Mattering the world of a mobile phone app: An anthropological study of design decision-making in a South African corporate context

By

Hannah Acutt

Student number: 17031118

Submitted in fulfilment for the degree Master of Social Sciences in Social Anthropology in the Department of Anthropology, Archaeology, and Development in the Faculty of Humanities, University of Pretoria

Supervised by

Dr Detlev Krige

May 2023

Declaration

I declare that the dissertation I submit titled 'Mattering the world of a mobile phone app: An anthropological study of design decision-making in a South African corporate context', for the degree of MSocSci Anthropology in the Department of Anthropology, Archaeology, and Development at the University of Pretoria, is my own work and has never been submitted by me for a degree at this or any other university. Secondary material used in the dissertation has been properly acknowledged and referenced in accordance with university requirements. I am aware of University of Pretoria's policy and implications regarding plagiarism.

Signature 

Date: May 2023

Abstract

How users integrate mobile phones and their apps into their lives is well-documented in scholarly literature (Kusimba, 2021; Miller et al., 2021), but their design and construction processes are understudied. As mobile apps increasingly become integral in human interactions globally, I suggest that an ethnographic examination of the processes and relations of their production in one corporate contact in South Africa is essential in 'mattering' and 'worlding' (Puig de la Bellacasa, 2017) apps. Furthermore, as the Global South is often absent in emerging discourse about technology's construction (Comaroff & Comaroff, 2012), this study seeks to contribute to understandings of app development in the Global South. The inequitable underpinnings of both the labour involved in creating technologies (Ashworth, 2022) and in how technologies are deployed to users have recently become controversial, evident in Couldry and Meijas' (2019) arguments about datafication and data colonialism. In this dissertation, I suggest that the political economy tradition of exposé anthropology in S.A. is a useful starting point for exploring the power relations in data colonialism. Specifically, Spiegel's (2005) call for exposé anthropology to progress from exposing relations of power and exploitation under colonialism and Apartheid to embracing an ethics of care, enabled by the theoretical developments on "thinking with care", allows me to deploy Puig de la Bellacasa (2012) and others to trace the harmonious and non-harmonious relations in the work of building an app in South Africa. In analysing the data I made through participant observation in this corporate, office environment from mid-February to mid-May 2022, which focused not on users and their exploitation but on design decision-making as member of the Design Team, I reformulate Latour's "network" towards Ingold's (2013) "meshwork" to position human and non-human actors involved in the making of apps. Specifically, I take ideas from more-than-human anthropology and Science and Technology Studies to depict Jira – a software task allocation system deploying 'agile methodologies' in the company that was the field site – as both a focal actor in the meshwork (Silva, 2019) and a companion (Haraway, 2003), showing the way in

which translations (Silva, 2019) and boundaries (Bowker & Star, 2000; Sachs, 1995) feature in decision-making processes and the worlds of apps.

Acknowledgements

I would like to begin by thanking my supervisor, Dr Detlev Krige. This dissertation would certainly "not be what it is today" without you; thank you for being my companion in thinking, writing, and worlding. In a similar vein, thank you to those who attended the UP–NWU writing workshop in 2022. Your suggestions and interpretations of that draft chapter opened up my dissertation to become something that is now far more interesting than it would otherwise have been. Finally, thank you to my family and partner for supporting me in so very many ways. You lent me the strength that I was convinced I did not have.

Contents

List of Figures

1. Introduction

The words ‘design’ and ‘designer’ can apply to many contexts: they can be used in architecture to refer to designing the plans for a building, and to design an interior is to map out the furniture and colours. It generally refers to drawing up a plan for how something should be created. I am most interested in how technologies are designed and, more specifically, mobile apps. Mobile apps are ubiquitous and entangled in almost every aspect of modern life in South Africa and globally: ‘There’s an app for everything’ has become a common phrase since the early 2000s. One source suggests that “20 to 22 million people in South Africa use a smartphone, which accounts for about one third of the country’s population” (Statista, 2021). This is predicted to grow to approximately 26.3 million users by 2023 (Statista, 2021). Smartphones are able to download mobile apps, whereas feature phones are not. Based on the above statistics, it is likely that a large number of South Africans use apps and that the proportion will continue to grow.

The way apps are now used has been of great interest to me – the various actions of being ‘social’, doing business, and tracking one’s life online have seemingly become so much more integrated and condensed into this one object that many of us carry around on our persons every day. I only really noticed this growing degree of integration between different parts of our lives and multiple mobile phone apps when I upgraded to a new cellular phone at the beginning of 2020. I had to re-sign into all of the apps I had registered for as a user, remembering passwords and other login details that I had not used or had to recall for years. The most difficult was resetting the banking app, which I had simply been using my thumbprint to sign into for the past three years. In the process, I completely forgot my password and had to call the bank’s customer service line to reset it. It was then that I realised just how much information I had been gradually handing over to all these apps, and the companies that own the apps, over the years without noticing it. As I gave these apps permission once again to access my contacts,

photographs, and microphone, the voice in the back of my head that had read academic scholarship about 'data colonialism' (Couldry & Meijas, 2019) and heard about the Facebook scandals over 'data mining' in the media slowly started to scream – but I needed these apps; I needed to be able to use them.

This learning experience, together with my growing concern about my own app usage and how easily I have been handing over information to the apps I use on a daily basis, dovetailed with but also diverged from my readings of the academic literature on the co-production of technology and social understandings (Govia, 2020), the importance of constructions of categories and classifications of medical diseases in Science and Technology Studies (STS) (Bowker & Star, 2000), making objects and art when design meets anthropology (Ingold, 2013), questions of seamlessness and seamfulness in technological design (Vertesi, 2014), and the call by some scholars to pull back the curtain on the design, creation, and production of various technologies. In thinking about my master's research, I started considering going back to the beginning phases of the life cycle of apps. Instead of taking their existence for granted as a given factor in our lives, I aimed to research their creation rather than how they are used, the focus of most of the existing anthropological literature on mobile apps (e.g., Kusimba, 2021; Miller et al., 2021). I came to realise that apps do not just materialise as fully-formed technologies in our lives; people make them, people make decisions about how to make them, and thus, I thought, their design and creation must involve relations of various kinds of labour, material dimensions, expert knowledge, and decision-making processes that could be the focus of an anthropological study (see Dumit, 2014).

Studying the design of an app, the processes and relations involved in its creation, as a way to 'matter' an app, means questioning the seeming autonomy of the app, its apparent

independence, solidity, and severance from all the relations that created it. As I will explain in greater detail, the idea of 'mattering' I draw from Puig de la Bellacasa (2017) to be building up the world of the app while paying critical attention to the, possibly exploitative, relations of its production. Mattering an app means paying attention to the relations, including the labour dimensions, that are implicated in its creation (Dumit, 2014). Consequently, I draw on scholarship about relations and relationality to matter the app. In his call for 'relational ethnography', Desmond (2014), for example, argues that a relational perspective calls for re-connecting relational severances while paying attention to webs of interdependent relations that result in the construction of, for example, an app. This means developing a perspective and insights that move beyond seeing an app like a user to see an app simultaneously from the perspectives of a user, designer, project manager, and anthropologist. As one of the designers, a research participant in this study, told me in an interview, "All those pixel perfect stuff that you see, it's always a mess in the beginning." It is aspects of this mess, these vital constructions and purposeful stitchings, that I aim to bring to light in this dissertation and that make me state here that making an app is messy, and I am purposely making the mess visible. Focusing on the messiness also allows me to show you that an app that appears solid and severed from multiple relationships is, in fact, not as solid as it appears to users.

I also draw on the feminist philosopher Maria Puig de la Bellacasa (2012), who, in reference to Donna Haraway, states that "nothing comes without its world" (p. 198); in other words, the world of an app, like other commodities and human-made objects, is rich in connections and interdependencies that challenge the fetishization of the app product. No commodity, person, or object exists without a world of labours and relations behind it that produced it – without the 'mess' – even if that world often remains invisible in the same manner that the 'fetishism of commodities', resulting from the commodity form and market exchange, renders invisible

the relations underpinning commodity production (Marx, 1887). The products of the world of technology often appear mystified, incomprehensible, and divorced from the human labour involved in their production, as powerful actors in this world seek to cultivate an ideology of 'seamlessness' – the aim of which is "to produce sensors, wireless networks, and biometrics that could combine to produce a unified sense of digital space, seamlessly integrated with the world of human experience" (Vertesi, 2014, p. 268). Apps, as a now commonplace piece of technology that promises to be infinitely adaptable to local circumstances, are highly subject to this ideological view of technology and its promises. However, I build onto Marx's sole focus on production, as the world of the app is not constrained only to processes of production. Instead, as Dumit (2014) states, any being or thing should "be teased open to show the sticky economic, technical, political, organic, historical, mythic, and textual threads that make up its tissues" (p. 349).

1.1. Technology and Power

Technology has long fascinated humans, producing both fear and admiration. A growing body of scholarship is emerging on technology and society (Miller et al., 2021; Vertesi, 2012). Recent critical scholarship shows how technology is characterised by a variety of unequal power relations that make up its construction. For example, scholars have written about the Digital Divide in how data is used and capitalised on. This term is used to refer to "the gap that exists between individuals who have access to modern information and communication technology and those who lack access" (Steele, 2019). This gap can be created by differences in education, income levels, geographical restrictions, and digital literacy (Steele, 2019), and it can often compound inequalities that already exist along these lines. For example, during the pandemic, "many [university] students [in South Africa] went from access to high-speed internet on campus, to being reliant on a cell phone with high data costs" (Ndebvu, 2021) and,

subsequently, all students being supported by the National Student Financial Aid Scheme (NSFAS) were offered subsidised data packages (Ndebvu, 2021). Such an initiative was a recognition of the negative effects of the Digital Divide and an effort to mitigate them.

Additionally, technology companies have become powerful actors in the global economy, as technology giants "have power that derives from their mastery of digital technology and which societies do not yet have measures to contain" (Swabey & Harraca, 2021). According to the London School of Economics, Microsoft, Apple, Amazon, Google, Facebook, Tencent, and Alibaba together account for two-thirds of the total market value of the global digital economy (Goel, 2021). Together with their platform power and the information asymmetry generated by the platform operators having all the knowledge and buyers, sellers, and market participants having "little insight into the platform's rules and logic" (Swabey & Harraca, 2021), technology giants are able to dictate the rules for large areas of the economy – acting like a kind of monopoly. Moreover, the power of these technology companies is not limited to the economy, but, according to Swabey and Harraca (2021), they have also have political power. The large, generally supportive consumer base of these large platforms means that they can mobilise their user base to advocate for their interests with regulatory and political bodies – as with Uber using their consumers to oppose regulatory changes (Swabey & Harraca, 2021). These platforms also exert political power "in shaping which media and other politically relevant information their users perceive" (Swabey & Harraca, 2021). Instagram, recognising this influence, removed the 'Recent' tab in the app ahead of the 2020 US elections to prevent the spread of misinformation on the platform (Browning, 2020). Thus, as a researcher, caring – using this phrase in terms of Puig de la Bellacasa's (2012) thinking with care – about power and relations in designing and building an app in South Africa means that I must also consider the possibly exploitative relations that shape the sphere of technology and how to move forward

in depicting realities that engage with these relations. This could mean, for example, lifting the veil created by the 'fetishism of the commodity' (Marx, 1887).

Other scholars have lifted the veil regarding the material and labour dimensions involved in the production of mobile phones. For example, in her article so viscerally titled 'The Little Hands of Labour Behind your Smartphone', Feza Tabassum Azmi (2021) recounts how a study conducted by UNICEF found that "more than 40,000 children work in mines extracting cobalt that powers the batteries of mobile phones and other electronic devices". She continues that, according to Amnesty International, children mine cobalt ore by hand and carry it on their backs down narrow tunnels without any gear to protect them from injuries or inhaling harmful dust (Azmi, 2021). Thus, I do not seek to lift the veil on exploitation that exists in the production of technology in this dissertation because, as I have shown above, many scholars have brought attention to this aspect of the world of technology and apps. I am aware of this research and the tradition of political economy, and I respond to it here in my introduction and my literature review. Thus, while I did not make data on exploitative relations, and thus cannot speak to it in this dissertation, I can connect the world of the app to this scholarship.

1.2. Theoretical Approaches

In the 1960s and 1970s, there was growing interest in Marxism among anthropologists, as Marxism emerged as a part of anthropologists' critiques of colonialism (Zunner-Keating, 2020). The main tenet that appealed was Marx and Engel's argument that "the Western cultural form was a product of specific historical events and ideas that could be critiqued and re-examined" (Zunner-Keating, 2020), thus rejecting the ethnocentric view of Western ideals and culture being superior. Responding to this critique of Western society and to the brutality of apartheid, that had been justified using the ideology of the superiority of White people over

Black people, South African anthropology developed exposé anthropology. This anthropology sought to depict the horrible effects of apartheid and racial capitalism on workers, mainly Black workers. South African anthropologists Gordon and Spiegel (1993) argued that "the strength of South African anthropology has been its ability to document, expose, and challenge the social contradictions and political-cultural myths so important in the making of South African society, both for the powerful and the powerless" (p. 92). The spirit of exposé anthropology I find useful in this dissertation for considering how to maintain anthropology's critical commitment to exposing relations of power and challenging the social contradictions of building technology in South Africa. The strength of exposé anthropology, and its focus on dismantling the ideology of apartheid, is that "apartheid as metaphor is a powerful means of explaining global inequality" (Gordon & Spiegel, 1993, p. 100). With the growing power of technology companies outlined above, I deploy exposé anthropology not in its traditional sense to depict the brutalities of Apartheid, but rather to draw on its critical analysis of inequalities, power, labour, and exploitation. The critique of data colonialism by Couldry and Meijas (2019) is representative of this return to exposé scholarship, as the authors seek to reveal the exploitative and unequal power relations that characterise the extraction of users' data by companies to sell it for profits. In this analysis, they depict this as a new stage of capitalism but also, as the term suggests, colonialism. This is because the aim of colonialism was to acquire, usually by force, large-scale resources from which economic value could be extracted, and it is this expropriation that makes these relations colonial (Couldry & Meijas, 2019). The colonial foundations of exposé anthropology are therefore well suited to an analysis of technology companies that still deploy colonial tactics to make their profits. However, the fault of a concept like data colonialism can be found in a parallel of Bruno Latour's (2004) critique of critique in his article 'Why has critique run out of steam? From matters of fact to matters of concern'. I suggest that Couldry and Meijas' (2019) critique of data colonialism transforms "the whole

rest of the world into naïve believers, into fetishists, into hapless victims of domination, while at the same time turning them into the mere superficial consequences of powerful hidden causalities coming from infrastructures whose makeup is never interrogated" (Latour, 2004, p. 243). Thus, instead of debunking the app and technology as something that *is* constructed and that, in this construction, is made up of asymmetrical power relations, I seek to consider *how* the app is constructed – what constitutes its gathering of matters of concern in a way that "adds reality to matters of fact and [does] not subtract reality" (Latour, 2004, p. 232). I seek to expose and reveal while simultaneously adding complexity and layers to the process of app production. The relations that are involved in the creation of an app are much more complex than simply involving exercises of power and domination over its uses in the colonial relations described by Couldry and Meijas (2019) or in the exploitative labour relations described by Marx (1887). Thus, as I will discuss further in my Literature Review, Lindén and Lydahl's (2021) suggestion for a double vision – focusing both on the potentialities of technology and on its dominations – aids me in my analysis and in depicting the complexity of an app's world.

My theoretical approach for depicting the construction of a technology in a corporate setting draws on suggestions from a few scholars for how to engage critically but responsibly in knowledge production, fleshed out more considerably in my Literature Review. First, in 2005, Andrew Spiegel made his argument for how South African anthropology could move on from its venerable tradition of exposé anthropology into new directions, such as an ethics of care. In that paper, Spiegel (2005) argues that anthropologists could not only focus on the politics of unequal relations as they have so productively done but also on depicting a care ethic; this would, in many ways, respond to Latour's suggestion for developing matters of concern. In adopting this care ethic, he argues that South African anthropology could move on but not abandon the "commitment to a situational ethical consciousness" (Spiegel, 2005, p. 136) that

exposé anthropology brings. Spiegel does not go into detail on how to bring the opportunities that exposé anthropology provides together with an 'ethics of care', but I argue that in taking a situated and 'worlding' approach to the study of technology, I am able to depict the construction and design of apps more fully than an "evangelical" (Nyamnjoh, 2015, p. 48) approach to anthropology. The "evangelical" approach seeks to 'give back' to and save marginalised communities rather than focusing on knowledge production. Thus, drawing on a double vision of care to bring together these concepts of critical, accountable knowledge construction for a study of technology, I can study technology in all its potentialities and dominations (Lindén & Lydahl, 2021) in a manner that is both critical of dominating power relations, but that also seeks opportunities for how to be more engaged in this world.

A further response of South African anthropologists to engage with the complexities of studying in South Africa is to be cognitive of the roots of the discipline and to be reflexive in choosing who or what to research. Gordon and Spiegel (1993) described how a "booming 'knowledge of the other' [drew] students to the discipline [of anthropology]" (p. 85), and this is reflected in more recent anthropology by Nyamnjoh (2012). He asserts that because of South Africa's history of Apartheid that degraded and oppressed certain groups along racial and 'othered' lines, "[South African anthropologists] study 'down' to marginal and disempowered people but rarely study 'up' to the privileged" (p. 70). In my research, I sought to 'study up' to examine the relations of power that characterise the construction of an app and, as Bowman (2009) argues, to "see connections between groups in society and to link groups and individuals to 'larger processes of change'…[to] inform our understandings of patterns of production, distribution, value, and power" (p. 2). Thus, 'studying up' in the world of the app allows me to link these app makers to processes of change, production, and power in the world of technology as a whole, and in the world of the app more specifically.

In my ‘studying up’ approach and my commitment to complexity, relationality, and liveliness, the process of designing an app in the company where I conducted field research involves many different actors and entities: clients, project managers/owners, designers, developers, Users, and, taking inspiration from more-than-human anthropology, I will also consider the labour management platform, Jira, as a non-human actor. Conceiving of the labour management platform as a non-human actor arrived much later in my project as I was writing up my data. I did not consider this to be a potential actor or ‘participant’ in my research proposal.

Because apps are designed to utilise visual means to communicate with users (Boxenbaum et al., 2018), how designers construct these visual elements of communication is a vital part of the process. The plasticity, malleability, and storage space (Henderson, 1998) that their designs provide also make the designers the nexus of this process. What this means is that apps are not only the outcome of the labour of developers who string together zeros and ones in a coding programme but also designers using artistic skills to create visuals that are central to user interfaces. Thus, I have placed the designers’ labours at the heart of the process, while also paying attention to their encounters with other employees in their company or with clients (Desmond, 2014) to depict the ‘world’ of apps (Puig de la Bellacasa, 2012). This was also shaped in part by the access I was able to negotiate during fieldwork, working as I did as part of the design team rather than a development team. Furthermore, I aimed from the start to do ‘anthropology with’ rather than ‘anthropology of’ (Ingold, 2013) the design team by becoming a member of the team and conducting participant observation in the second phase of my research. With regard to how anthropologists learn and make data during participant observation, Ingold (2013) states that “knowing is a process of active following, of going along” (p. 1), and as such is about being with participants and learning to see and do as they do

while learning these myself. I therefore came to learn and come to know by 'doing with' the designers rather than treating them as the objects of a study.

Finally, in reading in preparation for doing fieldwork for this dissertation, I realised that several conceptual binaries that structure our everyday speech and even critical thinking about the prominent themes in this study will not advance my thinking about the research. These are, amongst others, the idea that science and technology are separate from society and that humans can only interact with non-human entities rather than encounter and live with them. In this approach, I therefore draw on new materialism rather than Marxism. Marxism has made me attentive to…

> …the micropolitical working of power, politics and biopolitics[, but] the materiality addressed in these new materialisms is relational, plural, open, complex, uneven and contingent; cuts across dualistic boundaries between natural and social worlds; and for some new materialist scholars is invested with a vitality or liveliness. (Fox, n.d.)

Thus, I will consider technology, society, and power through the lens of matters of concern and care. I will expand on these in more detail in my Literature Review and explain why I do not think these conceptual binaries are productive for my study.

The following ethnographic vignette provides a description of my first experiences of designing an app and how it shaped what I came to pay attention to in this dissertation. I offer it as an entry point into design work and its complexities.

> It has been a few weeks since I began participant observation at App Company, the primary site of my fieldwork. I sat at my desk with the developers, pulled up a chair in the Design Team's office when they had meetings, and made tea in the small kitchen. I listened to the designers talk amongst themselves about how to make functionalities

work for users in an app together with the client's instructions, watched as these discussions manifested in a new set of buttons in the prototypes on Adobe XD, and observed the designers presenting these designs to clients. I now sit in the boardroom with Anika, Amanda, and Jessica as they review the design work they have done ahead of a meeting with a client the next day. The designers pull up a design they have been working on to present on the television in the boardroom. Amanda clicks on the Shopping Cart icon, and the view transitions to a collection of products that 'the user' has 'placed' in their cart. Not having encountered the elements of app design before, I am slightly bewildered by this seemingly ready-to-use and interactive form. However, in another presentation meeting with a client the next day, I realise this is far from the case. As they present to the client, there is a pop-up screen that no one – Designers, clients, or product owner – is sure the purpose of and there is a screen view that does not tie into any other screen or user journey. Like staircases leading to nowhere, these design ghosts haunt thought paths that were designed but never taken. They show that the designers must construct each element, ensuring that each staircase takes the user where they need to go and that one step does not jut out in the wrong direction. Beginning to realise the labour involved in designing an app, I decide to teach myself some elements of Adobe XD to better understand what aspects require the most toil, what tacit knowledge they draw on, and how they use the various tools afforded to them by the programme.

I download the free version of Adobe XD. I decided to start with Adobe XD on this day for two reasons. First, I am still seated in the development team's office, not having yet moved over to the design team's office. Because of this, I have wanted to be more of a 'participant' than an 'observer' in the Design Team's work. Second, some of the office

is off sick and some are working from home as they cope with a family issue. Consequently, the office is quiet, and interactions between team members are limited. As two of the members working from home are the director and the head of the design team, the office also feels much more relaxed and less structured, a good time I felt to allow myself to explore. With my back to the door and only Bongani sitting across from me and separated by the bank of monitors, I feel comfortable enough to teach myself how to use this tool that the designers are so skilled in wielding.

I open a new document and am confronted with the first hurdle: What size of canvas do I choose? They name the canvases according to 'standard' sizes they think the designer might want to use: 'iPhone 13', 'Web', and 'Instagram story', for example. With a standard pixel size attributed to each type of canvas, I assume that the aim is to simplify this very first element, which in many ways is one of the most important. The 'correct' choice should cause the device that mediates the user's engagement to fade away, as it is when the user has to scroll down a single screen on their smartphone that they become aware that perhaps the design was originally meant for a computer screen. Unsure what to choose, I go for a mid-range iPhone size. A single blank screen opens up, with options for drawing in certain shapes, adding text, and changing layers on the left side, and options for altering "Components" and changing colours and opacity on the right. Overwhelmed, I begin searching for YouTube videos and find one that teaches me how to make a login screen. Following it, I am delighted as I place a shadow on the oval and it becomes a login "button". As I design the rest of my app, I make decisions based on what I have seen other apps do: which options should be used on the menu bar (home, checkout, profile, etc.) and which icons to use for these options (a house, a shopping cart, a disembodied torso). As I design, I prototype, linking the screens and

buttons together, constantly checking that their flow "makes sense". My biggest accomplishment is when Luca helps me make components and change their states, thus allowing me to make it look as though my checkbox actually checks. This process involves creating two different forms for the checkbox, one with a check and one without, setting a state (default and toggle) for each, and prototyping to go between these states when the box is clicked. It is developing this piece of knowledge that brings me to my fullest realisation of the labour and toil involved in creating these interactive designs. This raises the first main question that I asked myself during my fieldwork: What is the purpose of creating such realistic and interactive designs? Why do the designers expend all this virtual sweat? And what is their relationship to these, their creations?

1.3. Research Methodology and Ethical Considerations

For this dissertation, I conducted an 'anthropology with' (Ingold, 2013) app makers with the Designers, Developers, and Product Mangers employed by a small technology company, App Company[1], based in South Africa, and more specifically in the provincial capital that is my hometown. My research participants also included project managers and designers who were employed by other companies that make apps, taking my cue from Desmond's (2014) argument to study different actors within the field. Some of the roles overlap, but broadly, at App Company, the sample was made up of three designers (Amanda[2], Jessica, and Luca), five project managers (Anika, Aphiwe, Connor, Tracey, and Vanessa), three developers (Bongiwe, Zintle, and Connor), and the director of the company (Wayne). Additionally, I interviewed

[1] The name of the company has been changed for the purposes of confidentiality.
[2] All the employees at App Company have been given pseudonyms to further protect the company's identity.

three Designers (Elize[3], Thulani, and Henrik) from another app company, Budgeting App[4], and three members of their management team (Stefan, Ryan, and Henrik). The rest of my sample consisted of three other designers (Arthur, Emily, and Erin), three developers (Marisha, Herman, and Chris), two members of a management team (Omer and Anna), and a client (Avril). In total, I interviewed 25 participants. Thirteen of my participants were men, and 12 were women. In terms of South African classifications of race (White, Black, Indian, and Coloured), 19 participants were White, 4 were Black, 1 was Indian, and 1 was Coloured. This system of classification arose during Apartheid and does not allow for greater diversity in the South African population. However, I use it here not as an analytical focus but because racialisation according to these categories is an everyday practice in South Africa. Maintaining these categories has also been used by the government and private companies like StatsSA to track economic markers like income, employment, and housing since Apartheid. Thus, I do not enter into any prolonged analysis of racial inequality in this dissertation, as I do not have sufficient data to do so, and it is not the focus of this dissertation.

1.3.1. Participant Observation

I began field research by conducting participant observation at App Company, so that I was an active part of the everyday decision-making processes of the designers. Participant observation gives the researcher "experiential knowledge…[and] puts [them] where the action is and lets [them] collect data" (Bernard, 2011, pp. 275-276). Participant observation was invaluable in this context since it gave me the opportunity to be a part of the decision-making process myself, observing the everyday decisions and conflicts that arose among varying members of the design team and between the design team and other actors. The focus of this investigation was the

[3] The names of those outside App Company have been changed according to their personal indication of whether they wished to have a pseudonym or not.
[4] The name of the app/company has been changed for the purposes of confidentiality

conversations the design team members participated in and how they negotiated about how to implement these conversations. I paid special attention, for example, to how the Designers discussed implementing a new design for a functionality and how they worked with the Product Owners/Managers, Developers, and Clients to develop the best design for this functionality. I did not pay attention to the more intricate issues of coding. This research was limited by the period of three to four months, so I mainly focused on the design team while finding greater space for analysing how they encountered the processes of labour enacted by the managers.

I spent three months in the company, from mid-February to mid-May 2022. I gained access to this company through the process of negotiating consent and an agreement that was made between myself as the researcher and the company's director, Wayne. Together, we 'workshopped' and developed a Memorandum of Understanding, which incorporated a Non-Disclosure Agreement detailing a maintenance of company processes' confidentiality and interactions that the company expected me to follow and how I intended to protect the company in my writing while maintaining my own research integrity – being able to represent my data fairly was the most important issue to me here. A part of this agreement was my provision of reciprocal labours to the company in the interest of fostering collegial relations. As a part of these labours, I wrote an article for the App Company newsletter, edited content for their website, and, at the end of my fieldwork, wrote a document in which I depicted my understandings and observations of how the company conducts its business and the interactions among Team Members. In my role as a participant observer, my first task was to reassure the employees that I was not there as a "spy" for the company's management, with Bongani still joking when I moved from the developers' office into the design team office that "You can go spy on them now." I thus did encounter some wariness of anthropologists and their role in a company setting especially, where the main concern was whether I would report back what the

employees said to Wayne. My main strategy for assuaging these concerns was to explain what anthropology as a discipline was, the purpose of my research, and to reiterate that my responsibility was to protect them as participants. I was also doing an 'anthropology with' my participants rather than an 'anthropology of' my participants, and so I was deeply interested in ensuring that our relations remained as reciprocal and horizontal (Bowman, 2009) as possible.

As a further part of my participant observation, I designed my own small app on Adobe XD, the programme the App Company designers use for their app prototypes and wireframes, as explained in my opening vignette. To understand the labour that they put into making these prototypes appear so realistic for the clients, I learnt the methods they used for depicting these functionalities using YouTube tutorials, with the help of the graphic designer Luca, and by playing around with the programme and teaching myself. Linking this to Ingold (2013), he suggests that anthropologists learn by engaging with their participants' worlds to know from the inside, and not simply observe their participants. Thus, I sought to understand and know about the designers' practices from the inside by doing with them, and not simply observing them as they worked.

1.3.2. Interviewing and Conversations

Participant observation was my principal method. However, I also conducted semi-structured interviews with the employees at App Company while I was doing field research at the company and interviews with research participants working in the design and technology field from December–May 2022. In semi-structured interviews, the researcher uses an interview guide, or a set of questions or topics to be covered (Cohen & Crabtree, 2006). The questions are still open-ended enough to allow participants "the freedom to express their views in their own terms" (Cohen & Crabtree, 2006). Important themes that I had not previously considered

then also emerged from interviews, such as time and labour tracking, classification, management, and how to 'see' or translate as a designer, developer, client, or project manager.

During my participant observation at App Company, I also sat in on different types of meetings to observe how tasks were delegated and how problems and conflicts were addressed. These meetings included the design team's daily stand-up meetings, sprint review meetings with the design team and clients, and sprint planning meetings with the head of the design team and the office's product manager. I will discuss these meetings later in the dissertation.

I also observed the ad-hoc conversations that team members had in between these meetings, while simply in their office, at lunch, or before the work day officially started. Free-flowing conversation allowed me to gain the 'in-between' information that I might not have thought of and to be part of the more intimate relationships that made up and make possible the everyday life within the company environment. I also made use of unstructured interviews, the main idea of which is that of "probe questions that are designed to be as open as possible [to allow] for spontaneity and for questions to develop during the course of the interview, which are based on the interviewees' responses" (LumenLearning, 2020). I found unstructured conversations useful for allowing me to see the concerns and decisions that the teams discussed in their everyday work. It was also when talking with each other that office workers would not feel the need to "explain" themselves, as they often did when talking to me, and the language that drew on their tacit knowledge would be more visible.

As part of participant observation, I developed some skills in prototype design, providing me with some of the knowledge that the designers have. I also developed my interviewing skills, learning when to prompt participants for more information and when they did not appear

comfortable enough for me to delve too deeply into their experiences. This was part of my new main skill: how to make participants feel more comfortable with my presence. As explained above, I began by explaining anthropology and my research as much and as often as possible so that my motives were not opaque. As I also became more comfortable in the company environment, I joked more with those at the company about my role and ventured to offer my opinions in conversations as the "anthropologist" 'who knew everything about any social issue'.

By conducting participant observation in App Company, I was able to do 'anthropology with' these app makers rather than an 'anthropology of', and thus I was able to learn to see as they do. Conducting interviews with the workers at App Company allowed me to gain a greater perspective on their understanding of the app-making process, and I cultivated connections and relationality by interviewing participants who were outside of App Company. The strength of participant observation was that it allowed me to gain an 'insider's' perspective of app creation and to both observe and be a part of encounters that I would not have been able to by only conducting interviews; I would not, for example, have realised the importance of Jira, the labour management software, in mediating their relations with other actors. The disadvantage of only being able to conduct interviews with the participants outside App Company is that I could not be a part of their everyday practices and could only draw on their descriptions and explanations rather than my own observations. However, by bringing these two methods together, I was able to make data on the process of making an app that provides a perspective of the production process.

1.3.3. Ethical Considerations

All research projects ought to be conducted in an ethical manner. This means that researchers ought to interact with participants according to ethical principles of conduct established by professional associations to protect both participants, the researchers, and the integrity of the research process. These principles are primarily informed by "respect for persons, beneficence, and justice" (Marshall, 1992, p. 1). Both Anthropology Southern Africa and the American Anthropological Association present guidelines on how to interact with participants, and these formed the basis of my own ethical conduct during this research. Briefly, the guidelines state that researchers should protect respondents from possible harm, make use of dynamic and informed consent, protect and preserve participants' data, and make findings available to participants at the conclusion of the research (American Anthropological Association, 2021; Anthropology Southern Africa, n.d.). I will now discuss each of these in my field research and in the writing of this dissertation.

First, an integral part of protecting respondents "may require the use of protective devices, giving pseudonyms to both persons and places, or offering anonymity" (Anthropology Southern Africa, n.d.). In conducting participant observation, I participated in App Company's day-to-day operations, which included information about how they conduct business that they do not wish to be made public. As such, I gave App Company the option from the beginning to be made anonymous to protect this information, which they opted to act on. As such, in writing this dissertation, I have either removed identifying details or explained my data in ways that will not give away confidential or sensitive information, any specific processes of coding or technology specific to the company, labour information, or personal details of employees. This is so that the company cannot be identified from these processes as per the Protection of Personal Information (POPI) Act. Broadly, the POPI Act "applies conditions for the lawful

processing of personal data of South Africans" (Privacy Policies, 2021). A large part of these conditions is accountability, openness, and security in the collection and dissemination of personal data (Privacy Policies, 2021). As I have indicated in the earlier footnotes of this dissertation, the name of App Company, Budgeting App, and all their employees have been anonymised, as per the managers' requests. For the interviews with designers and technology workers outside of these companies, I have followed their preferences as stipulated by their Letters of Informed Consent as to whether they will remain anonymous in this dissertation.

Consent is vital for any ethical research project. Informed consent "includes sharing with potential participants the research goals, methods…expected outcomes, anticipated impacts of the research, and the rights and responsibilities of research participants" (American Anthropological Association, 2021). Therefore, I obtained voluntary, informed, and dynamic consent from all research participants, both for interviewing and participant observation, ensuring that they understood and were willing to participate in this dissertation. They were able to withdraw their participation at any point, and there was no penalty for such withdrawal (Anthropology Southern Africa, n.d.). No participants chose to withdraw, but some participants chose not to have their interviews recorded, and I only took notes in my notebook. Consent was therefore repeatedly gained so that the consent-gaining process was "an ongoing dialogue and negotiation with research participants" (American Anthropological Association, 2021). As I explained above, a large part of gaining consent was negotiating the terms of my access to App Company.

I asked participants whether I could record interviews before each interview (Anthropology Southern Africa, n.d.). Participants were also made aware that I may use this data for future research, but that issues of anonymity and ethical consent will still apply to these future studies.

This is also a part of the POPI guidelines, which state that users have the right to know how their data are being used (Privacy Policies, 2021).

The participants whom I interviewed, the management of App Company, and the participants with whom I had conversations in the company were also given the opportunity to read a report based on my field research (Anthropology Southern Africa, n.d.). This document formed part of my efforts to maintain reciprocity between myself and App Company, as it gave them the chance to view their contribution and feel a part of the dissertation. This is also to prevent the sense that I have taken advantage of their trust by letting me become a member of the design team, even if only for a short while.

Lastly, I had to consider Covid-19 protocols for both the safety of myself and my participants. Although I was fully vaccinated, according to government guidelines for maintaining safety in the workplace (South African Government, 2021), I still wore my mask according to the company's guidelines, sanitised and washed my hands regularly, and maintained an appropriate distance from employees while in the office to reduce the chance of transmitting the virus.

2. Ethnographic Context

In this chapter, I provide some contextual information on where and with whom I conducted my participant observation to locate my research. Such efforts at considering location and positionality I take from feminist scholars like Haraway (1988) and her advocation for 'situated knowledges' or feminist objectivity; we need to name where we are and where we are not to develop partial, locatable knowledges (Haraway, 1988). Therefore, in this chapter, I situate the arguments that I will be making in this dissertation to add layers to my worlding of the app. A part of depicting situated knowledges is developing a sense of location. Cairns (2013) asserts that considering location "marks a shift from approaching the field as a backdrop or container in which research activities take place, to a spatial practice that actively constructs the people and places under study" (p. 326) or as, Madhok (2020) puts it, "a generative site in which concepts and persons come into being" (p. 401). Thus, in this chapter, I locate myself within my place of fieldwork, while also locating my place of fieldwork, to introduce the 'main characters' of my dissertation as a part of actively constructing their processes of becoming.

2.1. App Company: Location

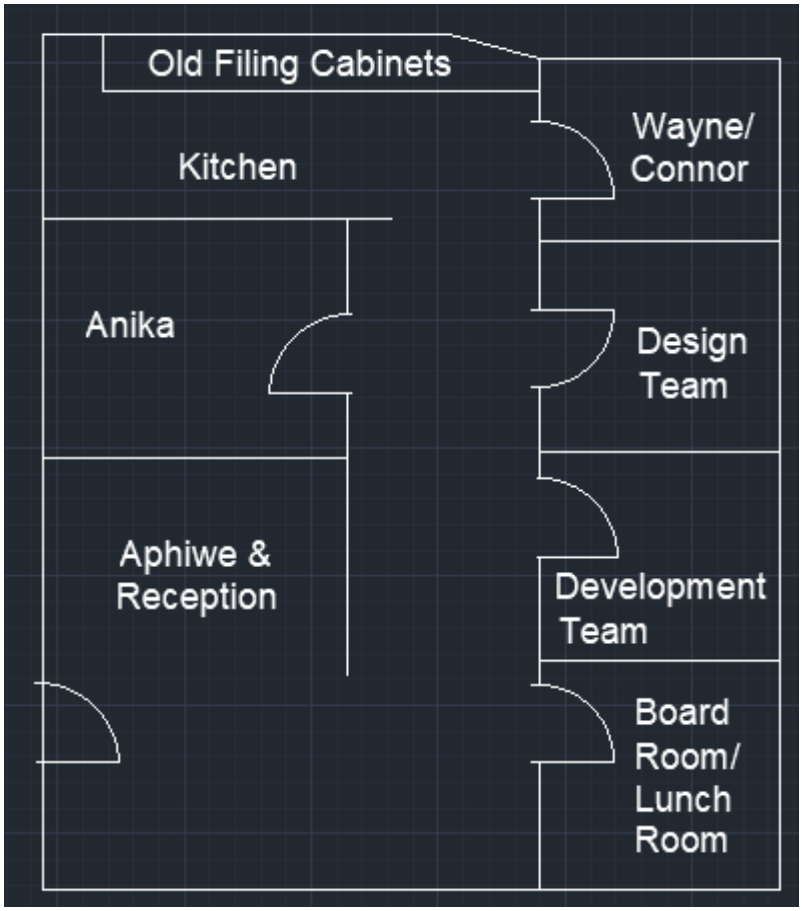
On the first day of doing field research at App Company, I follow the GPS on my phone as it takes me off the highway and out into the countryside. Eventually, I come to a small side road. Suddenly, the voice of my GPS tells me to turn right. I look around in confusion, not seeing another road and only some green fields with sheep grazing. I then notice the dirt track, seeming to lead off into the fields. Turning onto the track, I bump along it until I come to a building with a gate. I key in the gate code that Anika emailed to me and drive into the small parking lot, enough space for approximately 12 cars under cover and a grass space off to the side. The building itself is two stories high and modest looking. I later learn from my parents, who live in the area, that the building used to be an agricultural hub for farmers in the area. From 2012

to 2015, there were reports that the building was to be developed into a business park made up of light industrial sites, office spaces, and a shopping centre. The plans had certainly not materialised by the time of my fieldwork, as the building was still the same as it had been prior to its purchase for the scheme and was made up of only four or five businesses.

App Company registered as a business in 2015 (Companies and Intellectual Property Commission, 2018) and moved into the business park a few years later, allowing the company director, Wayne, to hire more staff. It was Wayne that I spoke to the most as I negotiated access to the company as a researcher. Thus, as I wait outside the office in the building's waiting area on my first day, early and first to the office, I am unsure if anyone besides Wayne will recognise me and let me in. Thankfully, a man – Aphiwe – arrives early to open the office. "Are you

Figure 1

Basic Schematic Floorplan of Downstairs Office Space



Note. Drawn by myself in AutoCAD.

Hannah?" he asks cheerfully. I gratefully reply that I am and get up to go inside with him. He unlocks the door to the App Company space, laughing about how I would beat him to his habitual status of being the earliest in the office, and begins to show me around. There is a type of reception desk, which also functions as Aphiwe's office since he is responsible primarily for support calls for a courier company. He then shows me the boardroom, quite small with a glass table in the middle and leather chairs around it, a large monitor at the front of the room, and a couch at the far end. He then shows me the office that I will share with Zintle and Bongani, explaining that they are a part of the development team at the company. Next is the design team's office for Amanda, Luca, and Jessica, about the same size as the development office, followed by Anika's, and lastly, the office that Wayne and Connor use, shown in Figure 1. Aphiwe explains that my desk is the one that Connor usually sits at, but he has been in Wayne's office for the past few weeks to learn more about certain tasks from Wayne. At the end is the small kitchen: a table with drawers, a kettle, a microwave, and mugs. There is no sink, which I later learn is a great annoyance for everyone in the office because they have to wash their dishes in the communal kitchen shared with another business. I sit down at my desk, slowly meeting the rest of the team as they trickle in for work. All of the team members are between the ages of 23 and 31, making for a fairly young office environment. Additionally, the head of the design team (Amanda) is married to the director of the company (Wayne), and one of the designers (Jessica) is married to the office manager (Connor), who is Wayne's brother, adding some kinship dynamics that overlap the co-worker relations.

2.2. Team Members' Roles and Designations

In a company that makes apps, there are different roles and designations for how tasks should be completed. In this section, I will depict these various functions as a part of 'mattering' the

connections and encounters (Dumit, 2014; Puig de la Bellacasa, 2017) that result in the emergence of the app.

2.2.1. Project Management and Support

As mentioned previously, Aphiwe handles all support queries related to the courier company and some project management for select clients. He is overseen most directly by Anika, whose main role or designation is that of Product Owner, for which her responsibilities include understanding what the client or user wants in their product, taking these requirements and turning them into instructions for other Team Members, and managing the workflow in the office by creating and assigning tasks. This can be for the building of a new product or for support issues on an existing product. She works with both the design team and the development team, as she creates and assigns tasks for both. Aphiwe and Anika work together as App Company's in-house management team, and their morning meeting is the first one that I sit in on.

The other member of the management team within the office is Connor, and part of his job is to work with the support desk to solve problems for clients when they have an issue with their product. He is also the office manager, so he is in charge of ensuring that "the office is running smoothly" or solving problems that arise in the office. Aphiwe and Connor work mainly with the development team, however, as support queries are typically related to development issues rather than design ones. The other members of the management team are Tracey and Vanessa, who work as consultants and not in the App Company offices. Vanessa owns her own business, based about half an hour from the App Company offices, where she primarily works through the ideation phase of building a mobile app with clients. The apps are then developed by App Company. Similarly, Tracey cultivates clients and brings the design and development work to

App Company while she works as product owner on these projects. I met Vanessa about a week into my fieldwork when she came into App Company offices to discuss conflicts she was having with a client, and she wanted Wayne's opinion on how to resolve them. Tracey I met only virtually on meetings that she had with Anika and in our interview. Vanessa understands her role mainly as one of facilitating the process of ideation, helping clients to map out the logic of their app. Tracey feels that project management is about working between all the various "humans" to bring a product vision to life in the best way possible.

The support group, who also project manage the labour of the office, works between and amongst both the development and design teams as they liaise with clients and relay information on tasks back to these teams. They act as the bridge between clients and the developers, and, as Aphiwe explained, they take the client's requirements "and then just translate them in a technological way in a way that the [development] guys can understand". They also help the client to understand the more 'technical jargon' by "translat[ing] into more human metaphors". Vanessa told me that clients need the most assistance with "the flow of things, so when people have got an idea, but their idea is made up of like a whole lot of words and a whirlwind, and how to get them into a structure that makes sense to a developer to be able to code."

The work done by the management team was, perhaps naively, an unexpectedly significant aspect of my fieldwork and the data I made. I had not anticipated that an entire team would be devoted to managing relations and communications between clients and the company's workers. Amanda explained that they implemented this role, hiring Anika initially, so that clients would have a dedicated person to talk to and not just "whoever picked up the phone". For the purposes of my dissertation, I became excited by the existence of this role, as I had

originally developed my ideas around how the different actors in the process 'translate' their requirements to each other, and in the field, I discovered there was an actor dedicated to such translation. The way in which Anika, as the main project manager in the office, allocated tasks to the team members also made explicit to me how time was valorised in the work environment, how the roles of the different teams and team members were defined and understood, and how she used her role as a kind of 'intermediator' to push a project forward to its completion. It was also in her main role as the "queen of Jira", as she dubbed herself, that she made tasks for the team, kept track of these tasks, and used the information contained in Jira to determine whether App Company could fulfil client requests. She and Jira worked together to shape and channel the labour of the teams at App Company, a force that I had not realised existed. I will introduce Jira in greater detail in the section 'Getting to Know Jira as a Companion', but suffice to say that Jira emerged as an important actor in my research. First, I will introduce the design team.

2.2.2. Design Team

The design team is made up of Amanda, Jessica, and Luca. Broadly, the design team designs all the images and graphics that the company needs – for functional specification documents, wireframes, prototypes, websites, and social media. Amanda explained that she and Wayne formulated the design team a couple of years after the company was started when she noticed a gap in the industry, where the design and development phases were split between different organisations. Bringing them both together in an "in-house environment [means that] people can come to one location and get everything from the get-go and move through the process."

Amanda is the head of the design team, overseeing all elements of the design process through quality control and team management. Currently, she still works on wireframes and prototypes, as she has the most experience of the team, but she hopes to transition the design work

completely over to the rest of the team so that she can focus on team management. She told me that the visual representation of products in the design process means that clients "seem to understand the process, and they can feedback a lot more productively on their product than if they don't have that…tactile element". Jessica mainly works on wireframes and prototypes for apps, taking instructions from what the product owners have sketched out with the clients. She fills in the sketches that were made with the client so that they become wireframes, adding in the client's corporate identity. For her, wireframes are an important part of the process because…

> …It's easier for [clients] to understand…this is what my app's gonna look like, rather than just giving them a sketch view, because then they want to know, how is my corporate identity gonna look on this? What does the button actually look like on my app…?

Lastly, Luca's main responsibilities are to design graphics for various social media platforms – Instagram, LinkedIn, Facebook, Twitter – and the App Company website. In terms of working with clients, he primarily does graphics, such as icons, buttons, and banners for websites, but he also helps with design work for app wireframes. He uses the Divi website builder to design webpages, Canva for online templates for designs, and Adobe Illustrator and Adobe Photoshop for creating vectors, graphics, and images.

In conclusion, the Design Team understands their role and focus in the company to be to give a visual body to the ideas that clients bring to the company, where the designs act as maps of intention for how the product will be developed. They are distinct from the development team, as they do not do any coding, and from the management team, as they typically do not communicate directly with clients.

2.2.3. Development Team

The development team consists of Wayne, Bongani, and Zintle within the office. Connor also does some front-end development work. There are also a few developers who work remotely, whom I did not meet. Wayne is both the head of the development team and the company director. As such, he is in charge of managing the development team and their tasks and reviewing the quality of their work. As the director of App Company, he also handles all matters related to running the business – finances, client relations, and oversight of all team operations, for example. Simply put, the role of the development team is to code the functionality of the app or website so that it works how the client intends it to and how wireframes depict the processes. Anika explained it to me as "dev is like a cave burrowed out behind the wireframes."

The front-end developers, Zintle and Connor, ensure that the development looks like the wireframes provided by the design team; for example, that images are where they should be and the correct size and that the font is correct and consistent. The back-end developers, Bongani and the remote workers, then "fill in the functionality" behind this that programmes what should happen when something is clicked, for example. Bongani explained the difference between the developers as it being the front-end developers who make the box on a form and the back-end developers who programme the functionality for what can go into that box and what happens if the information is entered incorrectly.

In conclusion, the main role and focus of the Development Team is to take the designs given to them by the Design Team and build them into workable code that will allow users to interact with the app as the designs display these interactions. If there are any issues or bugs with or

additions to the code that the client flags or requests at a later stage, then they must also implement these.

2.3. A Day in the Life with Designers

Inspired by Ingold (2013), I conducted an 'anthropology with' the designers at the company rather than an 'anthropology of' designers. This meant that I spent my days with them in their offices, listening to clients with them, joking with them, and learning about app design with them. With them and their guidance, I came to understand how apps are made. As I seek to further provide a description of technology creation that does not separate technology from society, I insist that their daily routines and interactions co-produce the designs they make, forming the structure on which they build and work. As Seaver (2018) explains, software systems in his case and apps in mine, "take the shape of the organizations that make them". I have chosen here to provide a description of one day that I spent at the company, a slice of their daily lives, which I will use to enter into a depiction of their general routines and rituals. I have chosen the second day of my fieldwork at the office, which was at the end of their two-week working sprint.

Amanda and Wayne were half an hour late in coming in that morning, and the Design Team meeting had to be delayed until Amanda arrived. As Amanda and Wayne are married to each other, their schedules are usually lined up. At the time, I wondered if the employees resented this late arrival and if they would consider their managers' marriage an inconvenience. As my fieldwork progressed, however, I learnt that Wayne would often work from home if he had a lot of work to do, and thus Amanda would, too. That morning, I waited anxiously at my desk in the Development Team office, concerned that I had missed the Design Team's daily meeting. I set up my laptop, pulled out my journal and pen, and waited. Eventually, Amanda and Wayne

came into the office, and Amanda called me to sit in on the meeting. I picked up my journal and went to their office, sitting in the extra chair. This meeting is called a “stand-up” meeting, a term that immediately makes me think of a kind of corporate energy where efficiency means moving and not sitting down. We were all sitting down, of course. In the meeting were Amanda, Luca, and Jessica. They reported on the work they had done the previous day and what they planned to do that day. Luca explained that he was carrying on with the graphic design work he has been doing for an app, and at the end Amanda stated that she will be doing “sprint planning” – planning out the work for the next two weeks. I appreciate that Amanda also makes herself accountable to her team.

I went back to the developers’ office, writing up my notes on the meeting, wondering what I would do for the rest of the day. To my excitement, Amanda came to tell me that they would be having a client feedback meeting shortly. In this meeting, they would explain to the client what they have been working on and receive feedback for moving forward. Amanda told me the meeting would be in the boardroom, and I entered the room at the appointed time expecting to meet with clients and explain my presence. There were, however, no clients. Amanda sat at the table, facing the television with her laptop out. Anika sat next to her at the head of the table, also with her laptop out. Jessica was on the couch at the far end of the room. I sat at the other end of the table, quietly writing notes in my notebook. I realised that the meeting would be virtual as Anika and Amanda log on to the meeting in GoogleMeet on Amanda’s laptop, and we waited without the client joining. Eventually, after some time of waiting, Amanda realised that they had clicked on the wrong link for the meeting. She quickly logged out, found the correct link, and entered the meeting, profusely apologising to the client. They immediately started the meeting to prevent further delays. Amanda presented the designs in GoogleMeet so that the client could see them while also putting the designs up on the television screen so that

Jessica and I could see them. Amanda and Anika presented changes that had been made and queried concerns with the client that came up as Jessica was designing. At the end, Anika told the Clients that she would send them the 'journey' – the set of app screens – link so that they could see how it looks on a phone and to add comments for any text changes. Anika and Amanda logged out of the meeting and talked about how they thought the meeting had gone. They told me that the clients were quieter than usual and sounded dissatisfied; Anika thought they just wanted to look through the journey or were upset that they were slightly late to the meeting. Picking up their belongings, they started moving back to their offices while deciding how long to give the client to read these changes. How long the designers should be given to implement changes was determined through an amiable discussion: Jessica made a suggestion, Anika made a counter-suggestion and checked how they felt, and they all agreed to the timeframe for the work.

Until lunch, I sat in the development team office, writing up notes and trying to listen to the chatter. We finally got to lunch: a formally designated half hour (13:00–13:30), even appearing on their working hours on the website. Most team members congregated in the boardroom for lunch, as the kitchen was not nearly big enough. After we moved to the bigger set of offices upstairs, the kitchen was big enough to accommodate most, so the boardroom became separated from the place of eating. While I was doing my fieldwork, Zintle and Bongani very seldom chose to eat with the other team members in the boardroom and then the kitchen, preferring instead to eat at their desks. I was never sure if this was because they did not want to interrupt their work, were more solitary, or did not feel comfortable eating with the others. This could be because of the 'family' dynamic that existed in the office, as I often noticed that Wayne, Amanda, Connor, and Jessica would discuss aspects of their family life at lunch: Wayne and Connors' parents or the dance class they were all going to that evening. This could sometimes

result in Anika, Luca, Aphiwe, and myself mostly sitting in silence until there had been a lull in the discussion or they suddenly became conscious that they were excluding the others. Then, the discussion would switch to work: talking about clients or the work that needed to be completed that afternoon. That particular lunch day, Anika and Amanda brought up in lunchtime conversation how I had joined the client meeting, noting that the clients were often more engaged than what they were that morning. They also complained about clients who do not listen to their explanations of why they completed an aspect of the design in a certain way. They were then interested in hearing my thoughts on the meeting: how did I think it went? Not wanting to make too many judgements on my second day, I said something about finding it interesting and looking forward to knowing more. As my fieldwork progressed, the conversations at lunch moved away from wanting to know my thoughts on their processes to wanting to hear my arguments on gender and gender roles. There were competing views in the office on (mostly binary) gender roles, with Bongiwe and Aphiwe firmly in the camp for designated roles in the home according to gender, Anika and Luca firmly opposing this and arguing for a distribution more according to ability, and Wayne and Connor opposing roles according to gender but acknowledging that perhaps some tasks are more naturally suited to a specific gender. As I entered the fray of these discussions, appalled at Aphiwe's assertions that men are simply harder working than women, for example, I managed to appoint myself as 'the resident anthropologist' and was often drawn into these types of discussions for my opinion based on my anthropology training. I found these discussions interesting as evidence of the distinctive viewpoints on social issues that are brought from 'outside' into the office. Lunchtime, as the space that is even set out on the website as a 'non-work' time, was the time when these discussions flourished, but they did often spill over into quips and inside jokes outside of this 'non-work' time. These 'social' issues were therefore not simply left at the door. At this lunchtime, however, these discussions have not yet started, and we moved on to discuss

how they were considering moving to the upstairs office space, which is bigger with more rooms. The proposed plan for moving upstairs is implemented about a month later.

The rest of the day was quiet, for me. For the App Company team, it was frantic and busy, as they were on the last day of a sprint, their two-week period of work. All the work that they had set out to do had to be completed before they ground over the next day onto their next period of work. At the end of the day, Wayne took me into the design team's office to show me how "cool" it was that they were all working on the same document at the same time on different computers, facilitated by Adobe XD. As we walked in, Amanda was twisted around looking at Luca's computer, hands in the air explaining something, and they immediately stopped work while Wayne explained, smiling awkwardly. I nodded, agreeing that it was interesting, and we stood there while the designers slowly picked up work again. As we turned to leave, there was stilted laughter, and Wayne asked about it. Jessica replied, "Well, it was kind of like a zoo: come look at the designers!" I laughed, too, having been aware myself of the awkward nature of the interaction that Wayne perhaps had not noticed in his excitement to share with me this team innovation. "That's me," I said, trying to ironically divert from this being my role, "I'm the anthropologist who's just here to spy on you." Thinking back, my ability to joke about my own role in the work environment was a means of mitigating the possible anxiety around my presence and what my role would be. The jokes, I hoped, would help make some of these anxieties explicit and allow me to subside them using humour.

My very last impression of the day, as I collected my backpack from the Development Team office, was Wayne telling Bongani that even every informal conversation must be logged in Jira, the labour management software – and, as I describe later, a non-human actor – otherwise "it didn't happen".

Now that I have introduced nearly all of the actors and provided contextual information, I want to review some of the relevant literature that framed my proposed research, which has made me curious and attentive to the ways in which these different actors encounter each other in the process of designing and constructing an app. More specifically, I will describe exposé anthropology in greater detail and link it to an explanation of Data Colonialism, what feminist approaches to care and ethics can bring to a study of technology, how understandings of classification and boundaries in information infrastructures can be used to depict the way that labour is organised, introduce the ideas of translation and transformation in the network, reflect on designing as a creative process within a corporate organisation, and, finally, consider how materiality and the visual can be used to explore the process of designing.

3. Literature Review

3.1. Framing My Theoretical Approach to a Study of Technology

In formulating my research questions for this research, deciding what to pay attention to in my fieldwork, and developing my arguments for this dissertation, I was guided by the following authors, conversations, and arguments. In presenting these conversations and discussions, I will be doing so in the spirit of Haraway's 'situated knowledges' and Puig de le Bellacasa's notion of 'thinking with care'. Haraway (1988) argues that situated knowledge includes "sustaining the possibility of webs of connections" (p. 584). Puig de la Bellacasa (2012) encourages us to recognise and build upon the webs of ideas and arguments crafted by other authors rather than just using such authors as background. Taking my lead from these two feminist thinkers, I want to connect the various arguments, thoughts, and conversations to help me reflect on how they have contributed to formulating my arguments for this dissertation to situate and locate their conversations in my own journey of reading and learning.

While there are many bodies of literature and concepts that are relevant to a study of apps, I will draw on the following in my discussion and contextualisation of my dissertation (I will introduce the main authors in the duration of the Literature Review): a feminist approach to capitalism; a discussion on the notion of 'data colonialism' with a reflection on using exposé anthropology; a consideration of the usefulness of the concept 'seamfulness' and 'seeing as' in studies of technology; situated knowledges; the concepts of matters of concern and matters of care, 'worlding', and 'thinking with care'; 'making' and 'anthropology with' rather than 'anthropology of'; scholarship on 'classification' and 'boundary-making'; the notion of 'digital materiality'; material organisations of space and tasks; translation and transformation; and how the visual mode is used in design and corporate organisations. I start my discussion of the literature with a review of a macro-sociological and historical perspective recently advanced

by Couldry and Meijas (2019), namely 'data colonialism', as it relates to providing the backdrop against which the designing of technologies like apps operates, and for a depiction of how exposé anthropology can be used to analyse and understand current technological relations.

3.1.1. Exposé Anthropology, Data Colonialism, and Seamfulness

As I started reading at the onset of this project, I encountered literature that reminded me of the tradition of exposé anthropology in South Africa. However, this literature was not about how colonialism and apartheid were rendering life for Black persons and families in South Africa intolerable and inhumane, as depicted by Spiegel (2005), but rather about how the relations surrounding and making up technology are becoming increasingly colonial and exploitative. For example, in their recent book *The Costs of Connection: How Data is Colonizing Human Life and Appropriating it for Capitalism*, Couldry and Meijas (2019) discuss how data and digital technologies have become such profitable tools of power accumulation for corporations and how the extent and scale of the labour and unequal relations that underpin these technologies remained obscured to users. Couldry and Meijas (2019) developed the notion of 'data colonialism' to describe the process of corporations "extracting value from data" (p. 188) created by users in everyday interactions with digital technologies, including mobile phone apps. Critical of this process, and attentive to issues of global inequalities in power and wealth, they argue that this process is akin to a type of colonialism since the human experience of users has become the target of profitable extraction, a colonisation by data (Couldry & Meijas, 2019, p. x). What marks this form of colonialism as new, however, is the ideology and imperatives of capitalism, which allows for "human beings [to] become not just actors in the production process but raw material that can be transformed into value for that production process" (Couldry & Meijas, 2019, p. xvii). Data colonialism, they argue, differs from historical

colonialism, such as that documented by exposé anthropology, in how and by whom it is enacted: data colonialism is enacted by corporations, whereas historical colonialism was enacted by governments using the ideology of civilisation and beneficence. Moreover, historical colonialism used guns and excessive forms of violence to subjugate, dispossess, and exploit, whereas the surveillance of data colonialism commits violence against the integrity of the self, as these tracked lives exist in a "space [that] is continuously invaded and subjected to extraction by external power" (Couldry & Meijas, 2019, p. 157). However, what their analysis misses here is the involvement of private corporations in the advancement of colonialism and exploitation throughout history. For example, the English East India Company, a private company that traded with East and Southeast Asia and India, "became involved in politics and acted as an agent of British imperialism in India from the early 18th century to the mid-19th century" (The Editors of Encyclopaedia Britannica, 2022). Thus, colonial-style exploitative relations have long gone hand in hand with the extractive powers of capitalism, and private companies have a long history of influencing state mechanisms of control.

In this vein of the interplay between colonialism and capitalism, in addition to the notion of 'data colonialism', Couldry and Meijas (2019) develop the idea of 'data relations', which are the relations that make data available for extraction by corporations. 'Data colonialism' cannot exist without 'data relations'. Thus, Couldry and Meijas (2019) take a relational approach to studying data colonialism, linking them to relational perspectives such as those put forward by Desmond (2014) and Puig de la Bellacasa (2012). These approaches will be explored in greater detail in the section 'Feminist Approaches'. The relations they refer to are the ways in which users of digital technologies interact with each other and the world, facilitated by digital tools. These interactions enable surveillance, monitoring, and social caching by the 'social quantification sector' (Couldry & Meijas, 2019). Data in this sense is not 'data' as South

Africans typically understand it, which is cell phone data that can be bought from cell phone companies to access the internet and apps on smartphones. Data as Couldry and Meijas (2019) use it is the data that is generated about users through their interactions on digital platforms, such as data on where they plan to go on holiday and their social demographics. This data can be constantly dis- and re-aggregated to form new packages of data to be sold to third parties (governments, companies, etc.), similar to the securitisation of financial assets (Kenton, 2021). Securitisation "occurs when an originator packages various financial assets into one group then sells this group of repackaged assets to investors" (Kenton, 2021). Such dis- and re-aggregation of data as commodity renders the histories and pathways of data even more obscure.

Within this process of 'datafication' and extraction of data, Couldry and Meijas (2019) argue that there does occur a type of violence against users. Rather than the brutal physical violence of historical colonialism, data colonialism can be better understood as a kind of abstracted and economic violence through the "gradual elimination of social spaces that can exist outside data relations" (Couldry & Meijas, 2019, p. 107). For example, if all grant payments to grant beneficiaries in South Africa had to be made through an app, then beneficiaries who previously managed to exist without a smartphone would find themselves forced to buy one. In using the app and the smartphone, they would suddenly also become a part of the web of data relations, where previously they had none or little to none, as they would receive their grant in person from a grocery store or via EFT. The authors also identify the extent of the extraction of users' personal data without their consent as a form of violence against their personal autonomy. As users are unaware of the ways in which their data are being extracted and sold through surveillance, the integrity of the self is under threat of erosion and dispossession (Couldry & Meijas, 2019). As I will discuss in the section on care, being attentive to these violent actions

is necessary for maintaining a critical view of technology, its production, and its uses. Worlding is not complete without paying attention to the violence that underpins "convenience".

Couldry and Meijas (2019) hint at a few strategies for resisting and dismantling this emerging new world order of 'data colonialism', among which is rejecting ideologies of datafication and thus the promise of personalisation of users' experiences on digital platforms – such as Netflix personalising users' watch suggestions based on users inputting whether they enjoyed a film or not. One strategy that they mention but do not explore in much detail, yet one that I find useful for my research, is 'seamfulness'. They borrow this term from STS scholar Janet Vertesi (2014) and her work on identifying seams in infrastructure. For Couldry and Meijas (2019), in the context of their book, seamfulness is the opposite to the seamlessness of data and technology that corporations seek to project, cultivated so that users desire a 'seamless' experience, in which users do not encounter friction when using a technology. Seamlessness is an aim of many material technologies, but it also expresses an ideology because, to achieve a seamless experience, users must give up greater and greater amounts of personal data so that the system can store it and use it for future experiences. Seamlessness aids data relations and furthers data colonialism. Since its aim is seamlessness, the external processing of data means that users do not know who this data is going to be used by or what it is going to be used for, and thus allows inequalities and surveillance to propagate invisibly. Couldry and Meijas (2019) suggest that paying attention to seamfulness (as opposed to seamlessness) could reveal or create obstructions where data flows are harmful. By drawing attention to the seams, the construction, labour, and relations that go into making these processes appear so seamless, users may take back their autonomy by being able to make decisions about whether they accept this part of the construction or not. I suggest that part of worlding an app could involve looking for seams and paying attention to seamfulness in production, design, and decision-making processes.

Whether users or other actors will take up this idea to resist datafication is another question altogether.

The designer and scholar Roussilhe (2018) explains that designers often aim to provide a 'seamless' experience, devoid of any friction, for the user. An example would be 'one-click' options available in many apps so that users do not have to repetitively input their payment details. Vertesi (2014) argues, however, that seamless design obscures the relations, infrastructures, and labour that have gone into stitching these varying elements together to appear coherent and a world on its own. In the words of Tim Ingold (2013), "processes of making appear swallowed up in objects made" (p. 7), and so users would not see any of the background mess that went into making the app appear seamless.

In contrast to furthering this ideology of seamlessness that clearly feeds 'data colonialism', Vertesi (2014) suggests paying attention to seamfulness, which "highlight[s] the seams between systems, instead of working toward their eradication" (p. 269). Highlighting the seams makes it possible to study, among others, the labour that actors put into manipulating various infrastructures to work for them at the local level and making these infrastructures appear "copresent". The example she gives is her experience with a group of scientists from the United States of America (USA) and Europe working together in a laboratory in Spain (Vertesi, 2014). For Americans to be able to plug in their American devices, they must plug in a transformer that steps down the power from the European 240 V grid to the American 110 V standard. For Europeans to be able to use their locally bought devices in a building that has only American plugs, they must plug in a European power strip. These workaround solutions result in scientists being able to use their different infrastructures by making them appear copresent, yet they are full of seams. Seamlessness in technology is not something that can just be assumed; it is

constructed and reproduced by designers on purpose, and it is often a messy overlap or entanglement of various technologies and infrastructures. These entangled infrastructures only appear copresent, or seamless, when they are interwoven tightly enough. In this collision of infrastructures, however, "their seams are visible in their many edges, endings, and exclusions" (Vertesi, 2014, p. 269).

Another strategy for engaging with concepts like data colonialism and its exploitative relations is through a rereading of the kind of exposé anthropology that was the hallmark of social anthropology in apartheid South Africa. Exposé anthropology arose during apartheid in parallel to the other dominant form of cultural anthropology at the time, namely "volkekunde", which sought to depict, in a very bounded and essentialist way, the various "cultures" in South Africa (Becker, 2007). This form of anthropology sought to essentialise, reify, and legitimise the cultural lines that the Apartheid government drew, taking the fact that these cultures existed as a given. Some anthropologists argued that this theoretical view would contribute to and justify the project of Apartheid and colonialism and opted instead to depict the impoverishment, suffering, and social stratification that Apartheid and colonialism produced, abstaining from reflections on culture and identity (Becker, 2007). Exposé anthropology therefore sought to reveal "micro-level 'beastliness' in order to expose the clearly systemic structures – some the continuing product of apartheid, others the product of global neo-liberal processes – that form the context in and out of which social-relational breakdown occurs" (Spiegel, 2005, p. 136). The political economy accounts that emerged from exposé anthropologists therefore played an important part in deconstructing the Apartheid government's premise of the benefits of separate development. [5]Anthropology in the political economy and exposé traditions remain useful for

[5] Separate development was a policy implemented by the apartheid government in South Africa to justify the forced geographical, political, and practical separation of different racial groups into separate 'homelands.'

the focus they place on paying attention to "micro-level" manifestations of exploitation – literally exposing or uncovering exploitative relations – but also for how these manifestations can be tied into macro-sociological processes like neo-liberalism. For example, Olukoshi (2015) recounts how structural adjustment programmes in Africa were used to tie African countries and leaders into committing to "'free' market-oriented governments" (p. 533) as a condition of receiving aid from the International Monetary Fund and the World Bank. However, these conditions of neoliberalism ultimately contributed to the decline of African economies, as the one-size-fits-all approach to reform did not account for the varying histories, contexts, and the nature of economic problems in the different countries (Olukoshi, 2015). Thus, the local manifestations of exposé anthropology can be used to speak to global trends.

Such approaches have become urgent in STS because of the mystification that surrounds emerging digital technologies, data colonialism, and how these are constructed as natural, inevitable, and positive (Klein, 2020). In this dissertation, I do not aim to show how data colonialism operates on a local level in the Global South and the kinds of domination and exploitation it engenders; as such, I do not use political economy as my main theoretical approach. However, I am concerned with how ideological aspects of data colonialism may work in the context of app development in the technology industry. I am also cautious about the macro-sociological arguments made by Couldry and Meijas (2019) and others, as they may be more appropriate when examining multinational and monopolistic corporations in the technology industry that are at the forefront of developing digital platforms used by users in the Global North and the Global South. I found it difficult to tie the data I made in the context of a small, privately owned technology company situated in the Global South to neat, macro-level arguments about data colonialism. As a result, I turned to Miller and Woodward's (2007) 'Manifesto for a study of denim' and their suggestion for how ethnography could approach

large-scale forces. They argue that anthropology, rather than simply exposing meta-sociological relations – such as the case of the assumed Americanisation of denim – can work to consider "the blindingly obvious" through ethnography for things that are so ubiquitous that we as humans and consumers are blinded by them. Rather than exposing this ubiquity of global denim, they use ethnography to discover why it is denim that is the chosen fabric used in such varied local contexts. In this, they argue that anthropology should engage with meta-sociological or global narratives of capitalism and Americanisation through local accounts, in their case, of how denim is worn. In taking this approach into my research on the production of apps, what is 'blindingly obvious' here is that these apps are often produced in corporate contexts that are run by the logics of a business – making profits, making labour as efficient as possible, and responding to other trends within the technology landscape for how to make and construct an app. Through my local account of a business in South Africa, I can engage with how and why these labour processes are enacted and how they affect the people who build the apps.

I will now discuss how I draw on feminist methodologies that provide pathways for how to conduct research that is accountable and embraces complex and heterogeneous realities, while remaining curious about how power works.

3.1.2. Feminist Approaches to Care and Ethics

A decade after Apartheid was dismantled legally, Spiegel (2005) argued that it was time for South African anthropology to shift its focus slightly. He suggested that anthropologists move on from exposé anthropology but not abandon the "commitment to a situational ethical consciousness" (Spiegel, 2005, p. 136). Rather than adopting an ethic taken only from "structural political concerns" (Spiegel, 2005, p. 136), he argued that anthropologists could

develop an attentiveness to context and situation. This echoes Becker's (2007) assertion that exposé anthropology often lacked reflection on the positionality of the researcher and on culture, even as the operations of power were exposed. A crucial aspect of the care ethic that Spiegel argued for is that care should be understood as a social process rather than something that happens only in an unformed and sporadic way in the home. Rather, "caring [can] be viewed as a species activity that includes everything that we do to maintain, continue and repair our 'world' so that we can live in it as well as possible" (Tronto, 1993, as cited in Spiegel, 2005, p. 136). Spiegel thus argued that it is necessary for anthropologists to consider how to give and receive care in both public and personal relations, as "care is almost always associated with inequality and vulnerability" (Spiegel, 2005, p. 137). Such a response promised to update exposé anthropology so that it would consider how to move theory forward rather than only exposing problematic political relations. To consider how to use exposé anthropology together with an ethics of care in the South African context, Nyamnjoh (2015) suggests that anthropologists, especially those in the Global South, should "move away from a propensity for vagaries [and "evangelism"] towards a more thoughtful, concrete navigation and critique of social processes, behaviours and people" (p. 60) Thus, exposé anthropology is useful for its critical examination of power and the resistance it evokes, but what Nyamnjoh is further speaking to here, I think, is the situatedness that comes from thinking with care and with feminist writers like Haraway and Puig de la Bellacasa.

Spiegel's arguments for developing an ethics of care developed in the early stages of social scientists' and anthropologists' engagement with the concept of care, yet much of his argument is now present in more recent thinking and writings on care. Similar to Spiegel's assertion that giving and receiving care involve conflict, Puig de la Bellacasa (2012) explains that in our inextricably interdependent world, caring should not be idealised. Rather, she argues, scholars

should engage through "everyday practical doings that engage with the troubles of interdependent existences" (p. 199). In her discussions on non-idealised care, she further argues that an important part of care-ful research is actively constructing the 'world' of the topic under study. This can also be called 'worlding' or 'mattering the world', and it offers me a conceptual hook to link together relevant ideas, arguments, and methodologies that I discuss in this chapter because of its commitment to seeing webs of connection and relation, processes of construction, and accountable or situated knowledges. In her interpretation, creating worlds, or worlding, involves thinking with care to actively cultivate relatedness to facilitate accountable knowledge construction (Puig de la Bellacasa, 2012, p. 205). Just as she explains that 'nothing comes without its world', I suggest that apps are not made in isolated contexts; apps have worlds, but these worlds are often rendered invisible. Puig de la Bellacasa's (2017) concept of 'matters of care' is useful here for understanding the merit of maintaining a critical approach that does not only seek to expose and debunk exploitative relations for the sake of it but, by maintaining an ethico-political consciousness, can address the complexities of an entangled world.

I take the idea of 'mattering' and 'worlding' primarily from Puig de la Bellacasa, who also took her inspiration from writers like Donna Haraway and Bruno Latour. In her *book Matters of Care: Speculative Ethics in More Than Human Worlds*, Puig de la Bellacasa (2017) seeks to further consider what it means to care in such an entangled world and how to "[stay] with the unsolved tensions and relations" (p. 5). Puig de la Bellacasa (2017) takes the origins of the term 'matters of care' from Latour's concept of 'matters of concern', which "stresses the troubled and unsettled ways, the more or less subtle ethical, political, and affective tremors, by which a gathering/thing/issue is constructed and holds together" (p. 35). Latour was in turn responding to the idea of 'matters of fact' in realist politics to examine how they are constructed and

assembled, not to critique and debunk them unnecessarily, but to emphasise "the concerns that attach and hold together matters of fact…to enrich and affirm reality by contributing further articulations" (Puig de la Bellacasa, 2017, p. 39). However, in response to the critique of critique posed by many scholars as they called for caring for the concerns in things, Puig de la Bellacasa (2017, p. 39) questions, "In a deeply troubled and strongly stratified world, don't we still need approaches that reveal power and oppressive relations in the assembling of concerns?". In response to this question, she suggests the concept of matters of care: an extension of matters of concern that engages with the world, with its strata of power and oppressive relations, to consider how to create more caring relationalities rather than "unveiling" matters of fact (Puig de la Bellacasa, 2017, p. 66). Thus, whereas exposé anthropology was concerned mainly with unveiling matters of fact pertaining to exploitation and domination, Puig de la Bellacasa (2017) suggests we move in the opposite direction, from matters of concern to matters of care.

I take from matters of care the approach of articulating what I consider to be the concerns, of what matters, in making technology in the Global South – exposing how they are constructed and assembled – but also paying attention to how these assemblages affect their worlds. Puig de la Bellacasa (2017) is also in conversation with Haraway's (1988) arguments for 'situated knowledge' in its emphasis on location and accountability. Apps emerge as the result of or are the outcome of complex webs of interdependent relations and often invisible labour. I aim to 'matter the world of an app' in this dissertation; to actively cultivate and construct the relations, labours, and webs of interdependence that go into the making of an app, thus articulating the concerns that are a part of this world. Mattering the world of an app then becomes the process of 'making visible' the webs of relations, decisions, capital, and labour that go into making an app. There are other ways to matter an app, and other aspects to pay attention to. Here, however,

I have used the concept of care to shape my research approach, as I actively sought the relations that made up the world of the app, but I also use the concept of care to frame the work that the designers, developers, and project managers do. Understanding the work the app builders do as acts of care is useful because "to care is to be affected by another, to be emotionally at stake in them in some way" (van Dooren, 2014), and the app builders become affected by the client's vision, having a stake in the app's construction. Thus, a part of this dissertation is considering what I choose to care about as a researcher and examining what the participants choose to care about as makers of apps.

However, to care is not a smooth or harmonious process but rather one fraught with conflicts, troubles, vulnerabilities, and power (Puig de la Bellacasa, 2012). Therefore, an important part of 'worlding' is paying attention to issues of dissent and disagreement, which echoes Desmond's (2014) suggestions for a 'relational ethnography'. In this approach, he suggests that researchers foreground "fight and struggle, cooperation and compromise, misunderstanding and shared meaning-making between actors occupying different positions in a field" (Desmond, 2014, p. 555) and, as such, acknowledge that relations may experience friction. Taking the idea of dissent and conflicts in caring further, calls to acknowledge the darker side of care have arisen recently, echoing Spiegel's (2005) early arguments that caring can involve exploitation and inequality. Lindén and Lydahl (2021) describe researching with a 'darker side' to care as also "attend[ing] closely to [care's] exclusions, violence and marginalisations, and to what is enabled for us as the researchers to say and do by (attending to) those exclusions and/or that violence" (p. 4). Violence in care is further articulated in the conservation literature, as "care for some individuals and species translates into suffering and death for others, the 'violent–care' of conservation" (van Dooren, 2014). What can be taken from depictions of care in conservation literature is the idea that care is non-innocent; choosing to care for one actor or

objective can mean committing violence against another actor or objective in another context. This is in part what Puig de la Bellacasa (2012) means with her non-idealised and non-romanticised notion of care, and the practice of 'thinking with care'. In data colonialism, for example, corporations' choices to care about the profitability of data extraction mean that they might commit violence against users by damaging their integrity of self.

To reconcile both the opportunities for care and its darker sides, Lindén and Lydahl (2021) suggest taking a "double vision" of care. They take the term "double vision" from Haraway to mean seeing from both perspectives to reveal both dominations and possibilities. A double vision of care, therefore, is "both situated and critical, staying with the practices, specificities and potentialities of care while simultaneously critically interrogating those practices when needed" (Lindén & Lydahl, 2021, p. 8). Such an approach is especially relevant to the study of technology, a sphere that draws so much on narratives of possibilities and futuristic imaginaries. As the account of data colonialism suggests, however, these realities are also made up of relations of domination and violence.

However, in my views of these networks of relation, such as data colonialism and capitalism, I do not view these as systems that simply exist. Rather, like the authors who penned *Gens: A Feminist Manifesto for the Study of Capitalism*, I like to approach capitalism and data colonialism as unstable networks that are maintained through many decisions, conflicts, negotiations, and final resolutions, rather than set "formal economic models, practices, boundaries, and market devices" (Bear et al., 2015). Capitalism, then, is not 'out there' but is 'being made' here through processes of co-production through a network of human and non-human relations. Thus, I find this approach useful for my research in understanding the app builders as contributing to these networks in creative and varying ways that draw on trends like

'best practice' but rework them according to what the Client wants and local contexts. This is an especially appropriate approach for writing about technology construction in the Global South. As a feminist scholar located in the Global South, I realise that much of the discussion about large corporations, technology and Information and Communications Technologies are shaped by discourses prevalent in the Global North. Comaroff and Comaroff (2012) explain that scholars and popular culture typically view advances in technology as originating in the North and migrating to the South as an extension of the North's domination of the global economic structure. This view is an oversimplification and should be critiqued. Furthermore, the Global South is often depicted as the place where scholarly data is created and then refined in the Global North (Comaroff & Comaroff, 2012), and thus, the Global South is not where innovation and construction take place. However, using both Bear et al.'s (2015) generative approach to thinking about networks like capitalism, and using evidence from fieldwork, I will show that the outcomes of these processes and sets of relations are not predetermined but emergent.

3.2. Strategies for Working Across Difference

3.2.1. Classification, Boundaries, and Infrastructures

This dissertation also draws on literature in Science and Technology Studies (STS), an interdisciplinary field of study that emerged in the 1970s and which takes as its starting point the idea that science and technology should be studied in close conjunction with social structures, social relations, and social practices (Rohracher, 2015). This field of study has some roots in Bruno Latour's scholarly work on nature and society (Philip, 2015). In a seminal book for STS, *Laboratory Life: The Social Construction of Scientific Facts*, Latour and Woolgar (1986) argue that while social studies of science have historically dealt in abstracts, they aimed instead to "become part of a laboratory, to follow closely the daily and intimate processes of

scientific work" (p. 13). Subsequently, scholars working within STS have conducted significant and engaged research on how the human experience shapes and is shaped by technologies and boundaries they produce. Latour has since developed his thinking in different directions, from actor network theory to his investigation into 'modes of existence' (e.g., Latour, 2018). He has also become critical of the relativism of social constructivist scholarship in the critical tradition (Latour, 2004), approaches that at times overlap with what has been called exposé anthropology in South Africa. What makes STS an exciting field of study for me is that it foregrounds ideas of 'co-constitution' and the mutual shaping of science and technology and society. Similar to how the Gens scholars (Bear et al., 2015) refuse to accept "the discursive representations of 'the economic' as a domain", so scholars in STS typically refuse to disentangle science and technology from society. For example, for Donna Haraway (2006), constructing socialist feminist politics comprises a "theory and practice addressed to the social relations of science and technology, including crucially the systems of myth and meaning structuring our imaginations" (p. 12). While there is no space in this chapter to review the entire STS field, I want to point to two ideas pertinent to my dissertation.

First, the relationship between humans and technology is an old question that has been studied in many disciplines. Technology comes in many forms, such as in agriculture, everyday items like door handles, or vehicles. I am interested in the technologies of apps and their associated infrastructures as a way to world such apps. As my previous section discussed, I am arguing for an approach that looks for "webs of interdependent connections" to depict the everyday labours of technology construction, in line with Puig de Bellacasa's (2012) 'thinking with care'. Govia (2020) explains that "Technology is not constructed in isolation, but instead co-produced with 'social practices, identities, norms, conventions, discourses, instruments and institutions'" (p. 46). The second idea comes from Haraway's (2006) questioning of the 'immutability of

boundaries', as she suggests that scholars intertwine the 'boundaries' of science and technology and cultural constructions and pay attention to this process of construction.

In this, the other boundary or binary that I seek to complicate, if not dissolve, in this dissertation is that between humans and non-humans. Here, I draw on what is tentatively called more-than-human anthropology, with some approaches also known as multispecies ethnography, which challenges the theoretical and ontological primacy of human relations in anthropological thinking, or its human-centrism. Instead, it seeks to be "attentive to the agency of other-than-human species, whether they are plants, animals, fungi, bacteria, or even viruses" (Locke & Muenster, 2018). More-than-human anthropology has made me attentive to the agencies or affect of more-than-human actors like technological assemblages. In most anthropological thinking, agency is typically afforded only to humans and not to other and organic assemblages, and yet our lives are becoming increasingly entangled with them. From Abrahamsson and Bertoni (2014) and their article on vermicomposting, I take the idea that becoming-with in more-than-human approaches is not always neat and cosy but rather a "messy, yet constantly productive and on-going coexistence" (p. 125). Consequently, in this dissertation, I put these ideas of co-constitution and entanglement with technology to work by paying attention to the often messy social foundations (relations, practices, labour) upon which technology construction is dependent, showing how social interactions inform decisions about making a technology like an app. Simultaneously, I seek to depict some non-human actors implicated in app development, not simply as 'something' the app makers interact with, but actors they labour, live, and co-become with. Treating some of the non-human infrastructures in my field as actors in a network that works with humans in making an app was an idea that I only started developing in the writing up of my data. It was not an approach or idea that I developed in my research proposal. Again, I take my lead in attempting such a line of argument from the work

done by Janet Vertesi (2012) and her analysis of a group of scientists involved in the making and subsequent usage of the Mars' Rover. In her article, 'Seeing like a Rover: How robots, teams, and images craft knowledge of Mars', Vertesi (2012) describes how scientists, in deciding every day where to place the rover on Mars, began to 'see like a rover'; they would contort their bodies and viewpoints so that they embodied the rover and would see what it 'saw' to be able to better position the rover once it was moving about on Mars. Vertesi (2012) in a statement that strongly echoes Haraway's approach to situated knowledges, states that "seeing like a Rover is very much a question of seeing from somewhere, not adopting a view from nowhere, but still maintaining the power of a singular perspective" (p. 409). I will develop the idea of embodiment and 'seeing as' in relation to how the designers employ varying means to embody users and 'see as' users when they use an app.

The topic of boundaries and the construction of categories is useful to a study of technology construction in the workplace in two ways, taken primarily from Bowker and Star's (2000) book *Sorting things out: Classification and its Consequences*. First, the process of creating a system or infrastructure of classification is not a frictionless, linear process (Bowker & Star, 2000). In my dissertation, I aim to depict how the designers, developers, and project managers created and implemented their systems of classification as they created, described, allocated, and worked through tasks together with their 'labour management' software, Jira – an actor made of code that I am yet to properly introduce. Paying attention to how labour – or 'human resources' in company talk – was managed and decisions made meant that the "arguments, decisions, uncertainties, and processual nature of decision making [avoided being] hidden away inside a piece of technology or in a complex representation" (Bowker & Star, 2000, p. 135). The practical politics of classification for how categories and systems of organisation are arrived at decide what should be visible and what should be invisible in the system,

simultaneously constructing the community that will draw on this information infrastructure (Bowker & Star, 2000). Thus, I will show how designers, developers, and project managers organise their labour through their tasks, valorising an agile and flexible view of design and development work and how it could most effectively be conducted within an organisational structure. Bowker and Star (2000) discuss classification systems specifically in terms of workflows, and they conclude that the main challenges are in terms of comparability (that it is standardised enough across different settings), visibility (making 'invisible' work visible), and control (allowing some freedom of decision to employees). However, they explain that striking a balance between these concerns is a difficult and complex process, and trade-offs are required in the real world. For example, high comparability or standardisation of the system and greater visibility of tasks mean that the system takes some control or management away from employees. As I will show, despite Anika calling herself the "queen of Jira", Jira not only took away some control but also affected how Anika managed and classified labour for the rest of the employees.

Another cluster of concepts that is beneficial to understanding the data I made in this research is that of 'boundaries', 'boundary objects', and 'conscription devices'. As discussed previously, Haraway (2006) advocates for taking pleasure in the mutability and malleability of boundaries and paying attention to their construction. Boundaries are not as inflexible as the term itself suggests in limiting an area or space. Instead, as Vertesi (2014) argues, actors can construct boundaries to suit their needs and exploit the resources at hand. However, where boundaries are constructed and drawn, 'boundary objects' allow for communication across them and for managing tension between these different viewpoints. They "are able both to travel across borders [and are thus plastic] and maintain some sort of constant identity" (Bowker & Star, 2000, p. 16). I will argue in this dissertation that the wireframes and prototypes

the designers use to work with clients to create a visual plan for their apps can be understood as a boundary object, as they allow for communication and travel across the different groups (designers, developers, project managers, clients), while still maintaining their identity as a 'wireframe'. They are "capable of being read on different levels by different groups involved with the design and its final product" (Henderson, 1998, p. 200). However, in line with Henderson's (1998) argument, I further view wireframes and the "labour management" software Jira as conscription devices. This concept combines the information-capturing abilities of inscription devices and the plasticity of boundary objects (Henderson, 1998). Inscriptions, an idea taken from Latour, are "images that have been extracted from the laboratory to appear later cleaned, redrawn, and displayed as figures in support of a text. They are mobile, presentable, combinable with one another, and immutable" (Henderson, 1998, p. 32). Conscription devices combine these two concepts to create devices that require and facilitate group participation using visual media across groups. The conscription device, or drawing or wireframe, therefore becomes the means through which all communication must pass and acts as "a reference and collaboration ground to unite all these various forms of knowledge for negotiation" (Henderson, 1998, p. 54). As I mentioned previously, the visual dimension of mattering the world of an app is important not only because of my participation in the design team but also because of the important role of the visual in communication and app interfacing. The role of the visual mode for communication will be discussed further in the section 'Translation and Transformation in Design'.

3.2.2. Translation and Translators in Networked Decision-Making

Translation, a key concept in STS, is not about linguistic changes and interpretation. While I will use the more traditional understanding of translation to understand how the different actors convey meaning to each other using language tailored to each actor – what I will refer to as

translation-as-language – the understanding of translation put forward by STS is also relevant. This translation concept focuses more on how actors within a network are employed to address a certain issue. In this understanding of translation, there is an objective that needs to be achieved. Actors are assigned roles by the focal actor to reach this objective, these roles are distributed and, if the process of translation is successful, then actors take up these roles and enact them (Silva, 2019). In the *problematisation* stage, there is a central actor who sets themselves up as the actor through whom all negotiations about the issue must be routed: the "obligatory passage point" (Silva, 2019). They also stand in for all the other actors' perspectives. In the next stage, *interessement*, the focal actor attempts to induce other actors to be interested in the objective, thus creating the actor–network. If interessement is successful (i.e., the other actors are interested in reaching the objective), then the actors' roles are set up and distributed in the enrolment phase. In the final stage, *mobilisation*, actors are mobilised across the network to achieve the objective by fulfilling the role given to them by the focal actor (Silva, 2019). This is only possible if there is an actor–spokesman, the translator. The translator is crucial in bringing about mobilisation, as they give voice to the "[actors'] desires, their secret thoughts, their interests, their mechanisms of operation" (Silva, 2019, p. 403).

The translator must represent the needs of the actors in the network, balanced with the requirements of the objective. If the translator is not successful in accurately representing the needs of the actors, then the stage of *dissidence* could occur when the network is contested and the translator is deemed unfit to be representative of the needs of the other actors. An effective translation "involves speaking for others in one's 'own language'…; simplifying the complexity of actors' attributes in order to make them fit into the network's demands; [and] concealing the internal power struggles which led to the network's formation and consolidation (Silva, 2019, p. 403). Silva (2019) further posits that an important aspect of translation is

transformation, as the network is constructed "not by transporting a force that would remain the same throughout as some sort of faithful intermediary, but by generating transformations" (Silva, 2019, p. 404). According to Latour (Silva, 2019), actors in the network are effective when they induce others to act, and these are mediators who "transform, translate, dis-tort, and modify the meaning or the elements they are supposed to carry" (Silva, 2019, p. 403). This is opposed to intermediaries who "faithfully transport 'meaning or force without transformation'" (Silva, 2019, pp. 403-404).

I take from the literature on translation and translators the idea of achieving objectives within a network, how to effectively achieve these objectives, and the importance of representation and transformation for successful translation and network relations. However, in the world of app design and development in a corporate setting, it is not as clear within the network who the focal actor is; this shifts constantly. Using the concept of care in conjunction with the translation concept is therefore helpful for conceptualising such shifts and tensions. As I discussed previously, 'worlding' suggests that some threads offer potentialities and others dominations (Lindén & Lydahl, 2021). Who and what to care for – to achieve a certain objective – is thus a complex and constantly shifting set of relations.

3.3. Design Anthropology, Creativity, and Emergence in Making

As I was an anthropologist in a company environment, design anthropology provides a historical basis for examining the relations of production and design. This field "integrates anthropological methods, theory, frameworks and critique with design principles and practices to address an increasingly wide range of complex systems-level problems facing contemporary societies, institutions, and organizations" (Miller, 2017, p. 7). The most useful ideas for my dissertation from this field of study are its emphasis on the forms of collaboration that can take

place between a design team and the anthropologist, and its focus on the materiality in digital constructions. Studies that have documented collaborations between designers and anthropologists show that this approach foregrounds objects "reciprocally in the making" (Miller, 2017, p. 60). As Ingold (2013) further argues, similar to Haraway, "we know because 'we' are of the world" (Ingold, 2013, p. 5). Therefore, to matter an app, I worked together with the app makers.

Co-production, emergence, and processes of becoming are important concepts for understanding the creative process. Govia (2020) argues that technology, in her case Artificial Intelligence, is produced together with social practices, and not in isolation. The concept of 'co-production' then is useful to this study because of the emphasis it places on a back-and-forth between the designers and other aspects of the design process they have to consider, such as insights into the user that the anthropologist or other researchers might provide, expectations of functionality, and constraints on their time and budget. The co-production in such processes of making resonates with Ingold's (2013) discussion of designers corresponding with their materials to make an object that is emergent, rather than only imposing a pre-given form on the materials. He therefore speaks of 'makers' and 'making', in which making is a process of growth that emerges from a confluence of forces and materials. Thus, in addition to constructing, creating, and producing, I will talk about the designers as *making* apps to bring to mind this correspondence that they undergo with the technologies, tools, and materials in their world. Wilf (2014) furthers Ingold's (2013) arguments about the characteristic of emergence in making in his article 'The Semiotics of Creativity', which is useful to my understanding of how designers work. He explains that the Romantic notion of art is that it comes from nowhere, like divine inspiration. Instead, he argues that creative processes in practice are emergent and interactional. He reframes creativity as designers and artists

reworking a "given set of building blocks" (Wilf, 2014, p. 398) using improvisation, a process that is temporal, generative, and relational. Therefore, the design process in making apps can be understood as an interactional one that corresponds with designers' environmental constraints.

3.4. Visual Turn and Materiality of Office Space and Tasks

3.4.1. Material Space and Metaphors

Pink et al. (2016) argue that "the digital and the material are not separate but entangled elements of the same processes, activities and intentionalities" (p. 1). This approach is useful to me because it supports an engaged ethnographic practice with everyday actions comprising the material and the digital. STS scholars Bowker and Star (2000) argue that metaphors of "materiality" are useful for examining the relationships between the varying components of the technological system and how, in this case, designers perceive such systems. The material aspect draws attention to the physical entities involved – plug points and documents that have to be filled in or signed – and conventional arrangements – speed (of the internet, of computers) and how specifications are implemented (the materials used, the sizes of images and text that are decided on). I aim to depict the material metaphors (Bowker & Star, 2000) that the designers use to understand and depict the abstract technologies that they work with. Material objects are an important part of how employees conceptualise their work and organisational environment. For example, Wilf (2016) uses post-it notes to describe "the role played by graphic artifacts in organizational knowledge production" (p. 733). He situated himself as an anthropologist in an 'innovation workshop', where consultants taught business entrepreneurs how to use post-it notes to abstract meaning from research and deploy it to generate insights for creating new products. This argument is useful here in understanding how material objects are employed in organisational settings to convey meaning between actors, such as translators and central

actors. Wilf (2016) argues that documents translate and transform the meaning they carry, suggesting that they act as mediators in the actor network (Silva, 2019).

How material space is organised is also important to a study of office settings, a topic that features in organisational anthropology. This body of literature is useful for understanding the configurations of space in the office and for giving me the language to discuss them. Organisational anthropology "applies…[the methods of] anthropology…to illuminate the problems and situations we encounter in the everyday groups and organizations in which we all participate every day" (Penn LPS, 2019). Such observations are useful because they show how space has been delineated by the business and whether employees use the space as intended or remake it for their own purposes (Gibbard, 2016). I suggest that space could be treated as a type of boundary because it can be materially delineated with desks and sections (Sachs, 1995). Studying uses of space can also reveal hierarchal relationships between and within teams, relationships between co-workers, and everyday negotiations of common spaces. The rationalist and explicit view of work understands work as a set of discrete tasks that can be achieved through specified business functions (Sachs, 1995). This view is useful for documenting the specific and expected tasks of employees, such as those held in Jira. The activity-based view of work, however, further allows for "analy[sing] everyday work practices to demonstrate the ways employees actually make the business function effectively" (Sachs, 1995, p. 38). This view is evocative of employing participant observation to gain a sense of the tasks between the tasks, paying attention to how tasks are discussed or enacted. As such, I make use of both of these viewpoints and approaches to gain a broader idea of the practices and actions of the team members.

3.4.2. The Material and Visual Turn

The primary function of the design team is to create wireframes and prototypes that display the design for the app. Thus, an important aspect of this dissertation is the concept of the 'visual', as wireframes are visual representations of the app, and the app is primarily interacted with visually on a screen using virtual buttons; thus, the interface of the app as an object is material. The labour management software that App Company uses, Jira, also depicts their labour visually, so analyses of vision are also useful for understanding how this organises and affects their labour. For an introduction to the realm of visual design, I found Henderson's (1998) book *On Line and on Paper: Visual Representations, Visual Culture, and Computer Graphics in Design Engineering* incredibly valuable. Her book takes place within the context of engineering drawings that are then used to construct physical objects. While the topic of design is different, the similarities between the two types of design are striking.

I have already pointed to the idea of conscription devices that she developed to understand these designs, but there are two other aspects of her book that I find most useful for framing my research. The first is the idea of 'visual culture' and 'visual literacy'. She explains that visual culture is "a way of seeing that simultaneously both reflects and shapes how members render the world" (Henderson, 1998, p. 25). This 'culture' is constructed by a shared visual literacy, or ability to read encoded meanings, in collective practices of coordination and conflict. Because "a new visual culture redefines both what it is to see and what there is to see'" (Henderson, 1998, p. 26), a visual culture thus constructs what and how designers or engineers see. Similarly, Haraway (1988) argues for reclaiming vision. Historically, vision has been used as the 'marked gaze', to see and not to see – to Other. To reclaim vision, she suggests employing feminist objectivity, or situated knowledges, as it is vision that will allow us to "see through the visualizing tricks of modern sciences and technologies" (Haraway, 1988, p. 582).

As I pointed out previously, these wireframes can be understood as a boundary object and conscription device, as they are used between all the different actors (designers, developers, project managers, and clients) to imagine and communicate about how the app will look and interact. As boundary objects and conscription devices, wireframes and prototypes become vital parts of how the different actors communicate about and negotiate the elements of the app. As Henderson (1998) so aptly describes the process:

> Visual representations, including prototypes, are not only devices for communal sharing of ideas but are also a ground for design conflict and company politics, exactly because they facilitate the social organization of workers, the work process, and the concepts that workers manipulate to produce a collective product. (p. 10)

To build on this, Boxenbaum et al.'s (2018) argument for a visual and material turn is useful. Similar to Wilf (2016) and his post-it notes, they consider how "organizational actors engage with novel ideas and innovations...[through] materiality and visuality" (Boxenbaum et al., 2018, p. 598). Organisational research on materiality studies has focused on "how meaning is created, communicated and stored through visual means" (Boxenbaum et al., 2018, p. 602). Thus, the visual can be understood as a mode for constructing, communicating, and transmitting meaning. Such arguments can shed light on how wireframes and prototypes are used to convey meaning to the different actors and to the users of the app in how to use the app. Their discussions are further useful for me to consider how the material (visual) and social realms are connected and co-construct each other.

4. Introducing Jira: Does She Byte?

"I understand in theory you guys want to operate with your sprints and tickets [in Jira]...But this is practical," the client shouts. Anika and Aphiwe had suggested I sit in on the meeting with this client, a particularly difficult one who never seemed to be satisfied with the work that the team was doing. We are sitting in the new upstairs boardroom, Anika and Aphiwe sitting closely together by Anika's laptop, where they are meeting with the Client via Google Meet. Nobody has their camera on; I find myself wondering if the Client has spittle flying out of his mouth.

The office is experiencing loadshedding – a scheduled period when the electricity is turned off by South Africa's public electricity utility company because of high demand and low electricity supply – and the air-conditioning does not run on the company's inverters. The room is thus uncomfortably hot with the afternoon sun beating in. This lengthy meeting, meant only to update the Client on recent changes, is dragging on longer than expected. The Client's website is in its final development stages, and I am joining at what is meant to be the 'fine-tuning' stage; the client wants to launch the website in a matter of weeks.

Anika repeats her previous explanations for why they cannot simply add new tasks that he requests: once the scope of a project is locked, new tasks cannot be added in that sprint. "But this has nothing to do with custom dev," the client continues. "This is spacing...spelling. It defies logic what you're saying to me." Anika rolls her eyes at me and continues to attempt to calm him in the same monotone with the same explanations, which I assume he must know after a year of working with them. The meeting ends with the Client reiterating his disappointment at how the App Company team continues to miss the deadlines that they had agreed upon, stating, "If I could do it myself, I would…You've had more than enough time to get this done. Things can't just be

> bumped and bumped and bumped." Trying to understand how this depiction of such ineffective time management could be the case, when from my experiences in the team I understand them to take their client deadlines very seriously, I ask Anika if they keep missing the deadlines because the Client keeps adding things for the team to do. She shrugs. "You've hit the nail on the head."

Although a rather extreme example, this vignette is indicative of non-experts' general lack of understanding about software and development work – how exactly work is conducted within an office environment and why App Company chooses to structure it in this way. It is also indicative of how the agile workflow methodologies and Jira, as the labour management software that App Company ascribes to, affect and impinge on those outside the organisation, their clients. The Client has not been properly introduced to Jira; he does not adequately know her and so does not understand her language. From my data and research on job boards, it seems that there are trends in how labour is organised in technology corporations that have arisen in the past few decades, such as in agile methodologies. Jira enacts agile methodologies and makes those who use her adopt the methodologies, such as a flexible work structure and a language of labour. In their book *The Silicon Valley Model,* Steiber and Alange (2015) argue that "Markets and technologies in virtually every industry are now subject to frequent and unpredictable change" (p. vii) and that a more adaptable business model is required to respond to these changes. The 'Silicon Valley Model', as they term it, refers to companies in Silicon Valley that have succeeded past the start-up stage (e.g., Google, Tesla, and Twitter) and have employed a new business management model that prizes "qualities like innovation, adaptability, and rapid response" (Steiber & Alange, 2015, p. vii). This idea that the rapidly changing nature of technologies is now being reflected in flexible business models finds agreement in the argument made by anthropologist Seaver (2018): "social architectures,

software architectures, and physical architectures echo each other". In the context of this dissertation, the structure of company relations is the foundation upon which the app is built. As I will show in this chapter, most research participants felt that agile methodologies were well-suited to the constantly adapting nature of app development. However, I will also show that, contrary to the seemingly coherent formulation of agile methodologies – as put forward by the process of Scrum (Scrum.org, 2022) and the company Atlassian (2022a), for example – this method of labouring, mediated and affected by Jira, is much more contingent and subject to processes of negotiation and conflict. The encounter with the Client in the vignette about agile methodologies is further evidence of how the methodology only *appears* coherent – the employees at App Company construct and reify agile methodologies by maintaining their construction.

App Company imposes their system of organising time and labour – shaped by Jira – on the client, legitimising the company and Jira's ideology of effective labour, but, as the vignette shows, this legitimisation is not always successful. Furthermore, an analysis of my data shows that these methodologies – and, more specifically, the language offered by Jira – increasingly form the language of the organisation, as workers use 'Jirian' terms and understandings of time and labour to structure their work days and to explain to clients the work that they will be doing. In this vignette, however, the client does not understand this language – he finds it "illogical". There are two types of translation that are failing here: translation-as-language and translation-in-the-network.

I use translation-as-language to refer to the different speech types that the various actors employ for speaking about the process of building an app and through which the actors emerge as a particular type of actor – designer, developer, client, project manager – in the network.

Translation-in-the-network I use to refer to translation more as it is understood in STS: different actors that are placed in the network with varying concerns, attempting to achieve a common objective – in this case, building an app. Throughout this dissertation, I will show that these two types of translation are co-constitutive, as effective translation-as-language often results in effective translation-in-the-network with actors achieving their objectives. As the vignette shows, both these types of translation are often unsuccessful, and as I will show in this chapter, the methods of labouring affect and provide the foundation for these attempts at translation. Lastly, it is also in the interest of creating situated knowledges (Haraway, 1988) that it is important to contextualise and historicise these workflows, a follow-on from my efforts in the 'Ethnographic Context' chapter to locate and situate the labour of the actors in this dissertation.

4.1. Labouring in a Corporate Technology Context

Most of the research participants that I interviewed worked in companies that made use of workflow methodologies that have become common in technology companies: agile methodology, "Scrum", and a task tracking software. However, while agile methodologies are dominant and part of common parlance in technology companies (Steiber & Alange, 2015), this chapter's opening vignette shows that the function of and reason for favouring agile methodologies may not be well understood outside of technological corporate contexts. This lack of understanding could suggest a failure of translation-as-language, together with translation-in-the-network, where the actors within the company have not adequately translated App Company's methodologies to the client so that he understands the methodologies' significance. While he understands the words, he does not understand how they affect and mediate his relations with the company. The company's attempts to legitimise what I argue is an ideology of labour has failed. To enter into an analysis of agile methodologies and how they are legitimised by companies and their employees, I begin by providing a summary of agile

workflow methodologies, as they mirror the now ever-moving and agile world of technology (Seaver, 2018).

In 2001, a group of USA-based software developers proposed the "agile methodology" in their manifesto for developing a new workflow methodology. The developers sought to provide "an alternative to documentation driven, heavyweight software development" (Beck et al., 2001). The agile methodology the developers proposed was a response to corporate organisational structures that did not typically allow for processes of iteration and collaboration among employees and with customers, thus affecting the software and technologies employees were producing. Wayne described agile methodologies as being "about being as lean as possible and then filling in as you need to". For him, adopting agile methodologies in his company was to improve how the team approaches the labour that he and the product owners assign by negotiating labour that determines where they need to "fill in". The product owner at Budgeting App, Ryan, described the company's team structure as taking the "traditional sort of agile team framework". Their head of design, Henrik, described it as "tried and tested in the industry". For managers, agile methodologies are well-suited to the form of labour in their companies, which simultaneously meets the discourse of technology as constantly changing and adapting (Technology Practice Team, 2020), while providing companies with an approach that is 'proven to work'. Furthermore, by depicting agile methodologies as "traditional", these managers are reifying the logicality of this methodology, not questioning if there is another way of working that might be better suited. To them, unlike to the client in my vignette, the system does not need to be questioned; it makes sense.

Wayne explained to me in a conversation one afternoon that the app industry and technology in general are agile, and so apps, and the way App Company builds them, must evolve, too. He

attributes this to the fact that with newer computers and newer technologies, the automation of certain aspects of the labour process changes workflow methodology. In this explanation of the technology industry, Wayne affirms Seaver's (2018) argument that social and physical architectures shape software architecture. I interpret Wayne's explanation, together with Seaver's arguments, as being that the company must be agile because the industry, its ideology of methodology, is agile. Therefore, I argue that the technology industry has had great effects on the way in which labour is organised and deployed because, as Steiber and Alange (2015) recount, workers in technology companies must be able to constantly adapt to changing conditions and innovate in these conditions of flux. I argue that, by continuing to employ agile methodologies, technology companies legitimise the power of agile methodologies to constantly alter local labour processes.

Many of the research participants also worked in technology companies where the company owners had decided that the employees' labour would be constructed with the "Scrum" framework. The "Scrum process", a part of an "agile Methodology", was first articulated in a paper written in 1986 by Hirotaka Takeuchi and Ikujiro Nonaka titled 'The New New Product Development Game'. The authors took the term from one of the player formations in the professionalised contact and team sport rugby and reworked it to emphasise the importance of teamwork in product development (Verheyen, 2017). As Verheyden (2017) recounts, Hirotaka and Ikujiro posited that the best performance in a corporate environment is achieved when teams are small and self-organised and when teams are given objectives rather than tasks (Verheyen, 2017). In 1993, a team at Easel Corporation in the USA, made up of a few of the authors of the later Agile Manifesto, adapted this "Scrum process for software development" (knowledge hut, 2022) to enable its software development teams to deliver complex products in an adaptive, iterative process. "Scrum" contrasts with the "waterfall" model, the dominant

project management methodology before the rise of agile methodologies. The "waterfall methodology" began in the manufacturing and construction industries, where changes were not often made to designs and construction throughout the process because of the large costs this would incur (Asmo, 2019). Therefore, it is a sequential and linear process rather than a cyclical and iterative one (Hoory & Bottorff, 2022). In software development, the "waterfall methodology" meant that developers only took on feedback from the client after all work had been completed. Thus, the newer agile methodologies were developed to address the mismatch between the software and technology products that were being created in highly transitional environments and a "waterfall" methodology that was not well-suited to the more creative, adaptable process of creating software and digital technologies.

In addition to promoting and legitimating the highly flexible and adaptable form of labour in the ideology of agile methodologies, the Scrum framework sets out how to practically incorporate this ideology in terms of time and approaches to tasks. The framework sets time out in increments of two-week periods called sprints, with Scrum meetings happening daily to discuss objectives and each Scrum team member's status in the project (knowledge hut, 2022). It is used in a variety of corporate contexts around the world by technology giants like Apple, Google, and Adobe (professional development, 2022).

Many project management platforms have been developed that draw on these agile methodology frameworks (e.g., Trello and Jira, developed by Atlassian (2022b) and ClickUp (2022)). These platforms are software that users download onto laptops or mobile phones and market themselves as allowing users to follow the principles of Scrum and agile methodologies for greater productivity (ClickUp, 2022; Nextup.ai, 2020). The most frequently used by the participants were Jira and Trello, both owned by the company Atlassian. For this dissertation,

I will focus on Jira since I was able to observe and participate in how the employees at App Company encounter and work together with Jira. Jira was created by the American-based software company Atlassian in 2002 to help software companies track "issues" and tasks in product management (Product Plan, 2022a).

According to Atlassian, Jira is used in over 100 000 organisations, such as NASA and ebay (Atlassian, 2022b). It is free for up to 10 users and can go up to $145 a month for a Premium subscription, supporting up to 20 000 users (Atlassian, 2022a). Atlassian has patented over 250 specific processes – applied for and granted from 2017 onwards – related to the different types of software that they offer (JUSTIA Patents, 2022), effectively making it difficult for other software companies to mimic their labour management tracking processes. The scale of patents granted to Atlassian and their attempt to patent these processes is indicative of how project management has become a new market and avenue for profit-making. The platforms themselves are marketed as 'friendly' interfaces that help you to "easily manage even the most hectic project workflows" (Atlassian, 2022a). What this overtly helpful presentation of these platforms obscures is the proprietary software that has become so profitable for companies like Atlassian and ClickUp by marketing the flexible ideology of agile methodologies.

These project management platforms fit in with broader trends in tracking software that market themselves as providing users with a greater sense of control over their lives, as I explained previously. The trends in tracking software are especially evident following the remote working that was enforced during the Covid-19 pandemic and which now still continues at least partly for many companies globally. These tracking software are also often known as "bossware", as they allow 'bosses' to track their employees' productivity (McKeegan, 2022). The idea of task-tracking software is that it streamlines workflow processes and – in the case of Jira, Trello, and

ClickUp, for example – enables the enactment of agile methodologies. Jira's tagline is "Move fast, stay aligned, and build better – together: The #1 software development tool used by agile teams" (Atlassian, 2022b). Atlassian markets Jira as a product that will facilitate the enactment of agile methodology, collaboration, and dexterity. Since the product is primarily one that tracks workers' labour by enlisting employees to document where they are in the life cycle of the task, Atlassian is therefore suggesting that companies can achieve these goals simply by tracking what their employees do already – and the insights for how to 'work better' will become easily visible.

As Couldry and Meijas (2019) explain, the surveillance of people at work is not new, but "the intimacy of monitoring in today's datafied workplace is without historical precedent" (p. 154). Just this year, reports surfaced on how Amazon is tracking its warehouse employees as they work (Ashworth, 2022). Amazon tracks the activity of the workers' handheld package scanners for how much time workers spend on "time off task" and, if more than 30 minutes of "time off task" are accumulated, workers can be disciplined (Ashworth, 2022). Couldry and Meijas (2019) argue "that the constant watchability of our every thought and action by external forces [e.g., corporations] changes the field of power in which we exist, transforming a supposed order of individuals into a collection of living entities plugged into an external system" (p. 164). Thus, the effect of such corporate tracking is to threaten autonomy or, in terms of Bowker and Star's (2000) arguments about creating a classification system in workflows, the control of employees. It removes some of their autonomy in determining how they labour.

In the next chapter, I will return to a more detailed depiction of how the workers at App Company work with Jira to depict and manipulate time at the company. In this section, however, I will set up and argue that viewing Jira as a non-human actor as a part of the network

of app production is a productive way of overcoming the binary between human and non-human actors and that doing so provides valuable insights into what it is that Jira *does* in corporate contexts, as promoted by more-than-human anthropology. Jira is also a fairly invisible actor to clients and non-professionals outside the corporate context until they encounter Jira; I was not aware that there was something other than a human manager delegating and tracking tasks. For the client in the vignette, he feels that he is being mediated by an actor whom he does not know and resists this mediation, as he is unfamiliar with the actor. Thus, positioning Jira as an actor goes some way towards preventing me from not seeing the 'hidden' work of Jira – the very real effects that it has on employees, corporations, and their clients.

4.2. Getting to Know Jira as a Companion Entity in the Corporate Context

To set up Jira as a non-human actor in both my thinking and this dissertation, it is useful to think about how she is "lively" (Lupton, 2016) so that I can make visible the ways in which Jira acts and moves. Lupton (2016), in her consideration of how data is "lively", helps me with my thinking in this. Her definition of how data are lively is as follows:

> [Data] are about life itself (details about humans and other living species); they are constantly generated and regenerated as well as purposed and repurposed as they enter into the digital knowledge economy; they have potential impacts on humans' and other species' lives; and their commercial and research value contributes to livelihoods. (Lupton, 2016, p. 2)

Jira holds and is about the details of humans – it provides the names of employees, what teams they are a part of, and the details of the labour they are performing. The details and information held within Jira are constantly being altered and repurposed as Jira consumes the intellectual resources inputted by the employees and produces a depiction of their labour. Jira has real and

material impacts on the employees who use it, a point that I will develop further on in this chapter. The commercial value of Jira also contributes to the livelihoods of those who have built it and maintain it – the founders and employees of Atlassian. Jira is thus a lively entity.

Atlassian terms Jira a software and thus puts it in opposition to hardware. Those who work in technology often maintain these distinctions, as with Atlassian terming Jira a software. Marisha, who worked as an engineer at an electronic development company, also maintained this distinction. Marisha stated that "hardware is the actual physical device that you make…[and] the software is anything that runs on an external device". However, calling Jira *soft*ware seems to undermine the rather hardy and forceful work that she does and the extent of her liveliness. STS scholars have questioned the distinction between hardware and software. The terms arose in the early days of computers in the 1940s–1950s, as hardware was much bigger and thus more unchanging, while software changed much more frequently (Vahid, 2003). However, Vahid (2003) explains that "programmable processors and IC fabrics are making once inflexible hardware 'soft,' blurring the line separating software and hardware" (p. 27). I will not enter too deeply into this debate, except to state that because of the original distinction that turns software into a set of executable instructions rather than something that is configured and built (Vahid, 2003), I will henceforth not refer to Jira as software. This reduces the work that Jira does to something that only "runs on" a hardier entity, thus taking away Jira's agency and liveliness. Because Jira is simultaneously a tool for tracking and setting out labour and time, a conscription device, and also a lively companion to the human actors at App Company, I will refer to Jira as a labour manager and companion to restore her liveliness.

I take the term "companion" from Haraway (2003), who coined "companion species" to refer to humans cohabiting with animals. She argues that the relations of humans and animals are

characterised by a "significant otherness" – where neither actor is whole nor parts; their existence is mutually constitutive. In other words, the relationship between companion species "implies a two-way dependency…[where] neither species would be what it is today without the other" (Buchanan, 2018). I will simply refer to "companions" here to avoid the difficulty of defining technology as a species, since species typically refer to the biological understanding of species of "related organisms that share common characteristics and are capable of interbreeding" (Gittleman, 2022). Instead, I name Jira an actor and an entity. I argue that Jira can be understood as a companion entity, as Jira and App Company are neither whole nor parts, but manage the company's labour in a way that is mutually constitutive.

Jira would not 'be where it is today' without the App Company employees continuing to input the data about their labour or moving their tasks through the categories of labour according to Jira – essentially 'feeding' Jira. Similarly, App Company would not be where it is today in how they constitute their labour. For the App Company team, work does not exist without Jira. They recently implemented Jira, previously working by using time tracking on tasks to calculate how much to bill clients. The internal company documents state that "'If it's not in Jira, it doesn't exist.' This is to provide one source of truth for tasks." Similarly, in a conversation I overheard between Wayne and Bongani, Wayne told Bongani that every informal conversation must be logged in Jira; otherwise, it did not happen. Jira is therefore a vital part of the labour relations at App Company. Furthermore, as I discussed previously, Amanda explained to me that App Company is quite novel for how it has brought development and design labour together in one company. This is an organisational innovation that results in bringing different kinds of labouring and different ways of talking about labour together under one roof. Jira then works to mediate these different kinds of labour, allowing the designers and developers to hold different kinds of labour together in one place and talk about it in one common language: the

language of Jira. The employees previously did not use the language of Jira's sprints, tasks, and stories, but simply spoke in terms of work to be done; Jira makes their units of labour comprehensible to each faction of labourers. I will discuss the mediation and translation work of Jira throughout this dissertation, but it is worth noting here that the company 'would not be what it is today' without Jira. The modes of labouring that are now present and emerging in App Company have been mutually constituted by Jira and App Company, with Jira acting as a companion to this labour.

Additionally, for companion entities and actors, it is vital that each actor learns how to communicate across irreducible difference (Haraway, 2003). The language of Jira has become the way in which actors communicate across the difference between the human and non-human actor, as Jira recognises the words sprint, story, and "backlog" which the App Company workers have learnt to use to communicate with Jira. They now also use the language of Jira to communicate with other human actors – other designers, developers, project managers, and clients. Thus, how do human actors live with Jira, and how do they emerge out of their encounters with Jira?

I put these attempts at living with Jira at the forefront of this chapter because of how Jira mediates and manages the labour at App Company, actions that I did not expect to find when I began my fieldwork. As I continued my fieldwork, however, I became more curious about the ways in which this manager affected how the workers classified and understood their labour and, subsequently, I began to think about how entangling Jira with the workers at App Company would reveal how integral labour managers like Jira have become to processes of labour in corporate contexts. In other words, Jira is a part of the world of the app and to examine how Jira affects labour, I view the work of Jira through the lens of matters of care to reveal

"the more or less subtle ethical, political, and affective tremors, by which [the labour associated with Jira as a] gathering/thing/issue is constructed and [held] together" (Puig de la Bellacasa, 2017, p. 35).

Lupton (2016) continues that interacting with or, in the vocabulary of more-than-human anthropology, living with digital technologies has "material effects on [people's] ways of living and conceptualising themselves" (p. 3). In doing my own writing for App Company, Jira provided me with the task describing what I needed to write about, how long it needed to be, and how to run it through Grammarly and Search Engine Optimization (SEO) when I had finished writing. When I read this task, I had to assess in my head what I already knew enough to write about, what I needed to do some research on, and whether I should write in Microsoft Word or Google Drive. From this task, I also realised that I would need to ask Amanda about how to gain access to their Grammarly subscription and what exactly to do with the SEO, since this was not a term I had previously encountered.

Other employees would have to engage in a similar mental evaluation of the labour required to complete the task set out to them by Jira. Thus, Jira makes workers do tasks, and it affects how they labour by setting out the parameters for that labour – the objectives, the methods and tools to use. Accepting a task from Jira means that workers must labour to orient themselves to that task, engaging with an evaluation of what they know they are capable of, what knowledge they possess on how to complete the task, what knowledge they need to acquire to complete it, and what tools they will need to employ. While employees might discuss the broad strokes of a new task with their head of department, it is in reading the task in Jira that this introspective assessment occurs as employees read through the task's details. Thus, in this encounter with Jira, each employee reconfigures themselves as an individual labouring entity, capable or

incapable of the labour set out to them by Jira, while also emerging as a member of the team to which they have been assigned. However, Jira functions only if project managers input tasks, if employees move their tasks along its progress bar, and if teams comment in the comments box and tag each other. Anika, as the project manager, inputs all the tasks for the period of work into Jira; in other words, she feeds Jira. This could be why Anika refers to herself as the "queen of Jira", as she is the one who is most knowledgeable about feeding Jira, about how to live with Jira, so that Jira is a companion that helps their labour rather than hindering it by inaccurately representing their labour. Anika, like a queen, becomes a ruler over Jira because she is the one who feeds it and gives it much of its power over the employees. As a companion, Jira empowers Anika as the ruler of Jira and the ruler of tasks in return for this feeding.

However, once this allocation of tasks has been inputted by the Project Manager, it can only move through Jira in specific and set ways developed by Jira's coders. Jira will track labour, prevent other labour from taking place, and make the team move in a way that the clients might not like. The encounter in the vignette with the irate client was a failure both in translation-as-language and in translation-in-the-network. Because the client is unaware of the extent to which Jira mediates the labour of App Company, he also does not recognise Jira as an actor that prevents him from getting what he wants. He does not feel that he, the workers at App Company, and, although he may not name it, Jira, have the same objectives: he wants to finish his website without going through App Company's protocols of labour, while App Company wants to maintain its process of labouring with Jira.

The App Company team has developed a dependence on Jira, placing all the company's deadlines and tasks within her lines and borders. This dependence means that the workers are unwilling to meet the client's demands of abandoning the labour process that they have created

with Jira. Additionally, to be effective in their labour, employees must be attentive to Jira. When working at the company, as a part of my reciprocal labours, I wrote a couple of articles for the company's newsletter. I remember distinctly when I first appeared on Jira's board on our computer screens, suddenly visible in company operations. It was also in this moment that I became more of an 'insider' in company operations – I entered the network of company relations, and thus I also entered into relation with Jira. Until then, I had been doing my own work, accountable only to myself, and now I felt very exposed – Jira was tracking me. Amanda had spoken with me about a timeframe for the article and given my task "story points" – the amount of time in which I was expected to complete the task. As I moved the task from 'To Do' to 'In Progress', I felt like a countdown had started, like Jira would alert Amanda if I went over the time she stipulated, and I would be in trouble. In doing so, I also became attentive to Jira: what did I need to do to correctly reflect the labour I had completed; who did I need to 'tag' in the comments section; what links to my work did I need to provide? Being attentive to Jira involved practical labours outside of the work I was doing for the task itself and giving Jira the power to track my labour and be visible to the whole company.

Jira also ferries the App Company employees towards a form of labour that is more easily adaptable to circumstances and time constraints, as employees are able to move their labour around on Jira's board and work across different tasks throughout the day. Amanda explains that the previous labour management software they used was…

> …very labour-intensive. So, we were using the same sort of idea, but not nearly as effective as Jira. So, it has been a bit of a leap to get into Jira and get the whole team onboarded with it. But it has been very beneficial. The sense that you also applying Scrum processes.

It is the sense of organisation that working with Jira in the Scrum process provides that is appealing to App Company employees. The idea that the previous system was more "labour intensive" suggests that Jira has somehow reduced the amount of labour needed to produce an app. Therefore, it is a companion that the workers at App Company see as not only 'keeping track' of tasks but also having material effects on the labour output of the corporation. Whether it actually reduces labour or only makes workers feel like it has is debateable. Their output could be greater, but, as I have shown throughout this chapter, working with Jira requires extra labours to maintain and feed Jira. The employees have had to learn how to live with Jira, how best to utilise their new companion.

Learning how to work with Jira is a process of labour itself, and when Jira is used incorrectly, the production process is halted. When working on my article task, I did not assign my task over to 'QA review' because I felt we were still working on it, and I only tagged Amanda in the 'In Progress' section. Her silence when I explained this indicated to me that I should have sent it to QA. I challenged the process App Company had developed of working with Jira, and so my labour became invisible. Tracey pointed out that "Jira is just a tool, and part of the process. So, it really works well to hold the process. And Jira is only as good as the people using Jira". Anika echoed this sentiment by stating that problems that have arisen with Jira have been "user specific…and if they don't know what they're meant to be doing". However, I argue that instead of only being a "tool", the assertions that it is how the employees use and respond to Jira that determines how well it is used show that Jira works with the other actors to construct the network and the world of the app – it is these attempts to 'live with' Jira, to become literate in it, that affect the production processes of making an app.

It is also through Jira that the agile methodology is re-enforced and legitimised, as Jira causes the company to materially construct how they work, speak, and structure their time according to Scrum principles and language. The ceremonies of the workday in general and of the agile process in particular are useful for considering how these ceremonies create a kind of time infrastructure that Jira helps to construct. Ryan described how he assigns work to developers, "and they commit to the work and…there's that whole ceremonial element to this, the agile process". Emily, when explaining how the different units in the company she works for settle on a workflow process, stated that each unit will "decide their own ceremonies that they want; they will decide what kind of structure do they want. Do they want to follow agile, do they want to do a little bit differently, do they want to use Jira?". This suggests that the labour processes within these companies have a ceremonial element.

Why this use of the term "ceremonies"?. One way to understand this is to think in terms of rituals and the anthropological literature on ritual. Baur (1949), for example, writes of how "ceremonial behavior in competitive American businesses functions to integrate entrepreneurs for collective action so that a code of trade practices can be established to assure a secure and honorable place in the community" (p. 358). Adapting this, I argue that it is the ceremonies of the agile process facilitated by Jira that provides a shared language and routine to all the Teams, despite the different areas that they labour on. Ceremony is defined by Merriam-Webster (2022) as "a routine action" and "a formal act or series of acts prescribed by…protocol, or convention". There are general ceremonies for the working day – switching on lights and the WiFi when arriving at the office and ending the day by closing computer programmes.

However, Jira and the agile process add another layer of ceremony that establishes common trade practices within the network of actors. The agile methodology contains and demands a

daily ceremony of starting the work day with a stand-up meeting, a meeting that formally designates that the new work day has begun. It is in this meeting that the previous day is processed and formally filed away, as Team Members discuss what was completed the previous day and move on to what work has to be done that day. The sprint also sets up this time infrastructure by determining the rhythm of labour in the company. As Thulani at Budgeting App explained, he organised his labour in descending order of its importance within the two-week sprint. These daily and weekly rhythms of labour are common to all the teams in App Company and so Jira strengthens the relations between the different teams by providing them with a common vocabulary and time infrastructure.

At App Company, Amanda explained to me that one of the benefits of Jira is the greater "accountability" and "transparency", as "[all the] teams also know where other team members are in that project". Thus, this greater visibility is not only for company managers to track the labour of their employees but also for employees to be able to check where others are in their tasks. In conjunction with the possible dominating and exploitative dimensions outlined above, it is simultaneously possible that, when this tracking is visible to all employees, it could be used to foster and develop stronger relations across the network. The employees know how they are being tracked, but their managers are also being tracked by Jira. Jira, then, could function to flatten the hierarchy of the office, placing managers on par with designers, as managers are tracked and kept accountable in the same way. Furthermore, instead of completely dividing the phases of the production process in an assembly line so that employees only know about their phase of the process, the transparency of the tasks makes the phases and the labour involved in these phases visible to all employees, thus possibly retying previously severed threads between, for example, design and development work.

However, in directing employees to what labour they must perform, how they must perform it, and in controlling their output, Jira becomes a kind of boss, as it is in charge of some workers' labour. Anika, as the "queen of Jira", directs the boss, working as a companion with Jira to control employees' labour. Understanding Jira as a boss, whether directed by another actor or not, allows me to consider how Jira's control affects control workers. As a technical assemblage, Jira, like the time tracking for Amazon workers, is an unforgiving boss and a demanding one. As it is merely instructive code, it does not take into account illness, compassionate leave, or the contingency of working in creative contexts where something may take longer than expected.

Jira simply continues to tick over time and report that the work has not been completed. It turns a social relationship into a human–machine relation, at least in discourse, and mediates some of the managers' relations between the employees. It is Jira who most directly tells workers what labour to perform, complicating the supposedly neat chain of command in a corporate context; Jira is a boss, too – not just the human managers. As Jira is a technical assemblage, for human, social issues to be taken into account, someone else in the organisation must intervene to reassign the task or remove it from the board. Thus, it is by working as companions, technology and human together, that labour at the company is managed and completed.

In light of the depiction of the darker side of care by Lindén and Lydahl (2021), thinking with care also means exposing the seams of exploitation and domination within these labour processes. While exposé anthropology aims only to depict the exploitation and beastliness of colonial relations (Spiegel, 2005), the double vision of care also seeks to consider "what is enabled for us as the researchers to say and do by (attending to) those exclusions and/or that violence" (Lindén & Lydahl, 2021, p. 4). While these tracking 'bosses' like Jira and the

Amazon tracking present the possibility of a more efficient workforce that is capable of greater outputs, the darker underbelly of this relationship is one of domination over workers' autonomy. Instead of workers being able to delineate what constitutes a task and how they want to prioritise their workload, this organisation is provided to them via Jira, and Jira tracks their labour to ensure that it is performed according to that organisation. Knowledge of how to 'live with' Jira, of how to feed the beast, is a part of existing in the network and world of app production. However, 'living with' Jira may not always be comfortable. It can cause workers to feel exposed, as I did. It could also demand more labour from workers than they can offer, as Jessica and Amanda jokily complained one day about how many tasks were on each of their Jira boards: "You should never have that many tasks [91] on your board." Often, too many tasks – too much labour – being inputted into Jira resulted in a frantic last day of the sprint, with employees staying late to ensure that they finished all their work for that sprint before going home and the next sprint starting the next day. Thus, viewing Jira as an actor makes visible the complexity of Jira's relations within the network of app production.

I have provided an overview of the dominant workflow methodologies, trends, and software being used by some companies that build apps. It was slightly astonishing to me, as my interviews continued, how dominant agile methodologies are, as I had not expected there to be such a widespread workflow methodology in the technology industry. I will now argue that, as a central actor in the network, Jira's language becomes integral to the process of corporate app development.

4.2.1. Speaking with Jira

The Scrum framework is promoted for its ability to promote collaboration and iterative interaction between both phases of the production process, in contrast to the more 'waterfall'

production process outlined above. From my data, it is evident that designers, developers, and project managers were interested in bringing greater collaboration between themselves and their clients or users, depending on the structure of the company – they were interested in the success of the network in achieving its objective or resolving its concern. In terms of collaboration between other team members, the design team at App Company has begun to implement a consultation with the development team before client review meetings (in which work that has been completed in the sprint is discussed), so that the designers can enter the meeting with a realistic view. Amanda explained that "[the development team] can tell you this is going to be a very pricey feature...we can do it, but it's going to be [expensive]", and the client then questions if they still want the feature. In terms of the network, this pursuit of a stronger association between the two teams and the Client is the result of the company coming to realise the importance of understanding the capacities or potentialities of the different actors' within the network as these relate to the objective of producing the app. Despite App Company's intentions, this process of collaboration is not always harmonious and can be fraught with tensions and friction between the nodes of the network. Viewing the encounter between Anika, Amanda, and the client at the beginning of the chapter as translation-in-the-network is useful for depicting the tensions and 'growing pains' involved in constructing and maintaining a network. The network is at risk of pulling apart – dissidence – with Wayne exclaiming after a report of the meeting from Amanda and Anika that, "I've given him 40 grand worth of free dev[elopment]...He can walk."

An important difference between Callon's (1984) traditional depiction of translation-in-the-network and the translation efforts that app makers perform is that in Callon's depiction, all other actors must come to speak in the language of the focal actor. However, designers view their designs as a kind of bridge between all network actors that facilitates discussion through

a common visual language; I will return to this discussion in Chapter 6. Translation efforts, and efforts to use different languages, to create stronger associations in the network are also made by the project managers. Anika explained that she speaks to clients by "reword[ing], rephras[ing] how to say things in a simpler way than when a dev[eloper] might start talking in dev[eloper] language or dev[eloper] jargon to a client." Therefore, she views an important part of her job as correctly representing the client's vision to the rest of the network – the app makers – acting as a mediator (Silva, 2019), including in her entering of task descriptions into Jira.

The project managers act as mediators by not simply transporting exactly what the client said; instead, as Anika said, they "basically take [the Client's] simple human terms of their needs and requirements and [turn] it into a use case for the dev[elopment] team [or task for the design team]". Thus, the meaning is transformed so that it can continue to move through the network. Similarly, project managers 'translate' from the app makers to the clients. Vanessa described to me how she speaks to clients so that they can comprehend the more technical aspects of building an app:

> I come up with creative ways...to explain stuff that makes sense in their industry…for instance, the trucking company, I was using analogies to do with putting together a truck, and what trucks can do…I think that's also a lot of the times when people get excited about a technical idea, and then they get in touch with a Wayne of the world [a developer] and then Wayne will be excited with them but Wayne is not excited in the right language. And they get off the phone feeling flustered, overwhelmed, like, "Oh, I think that we can do it, but I'm not quite sure what he was saying."

This explanation is indicative of the importance of mediators – project managers – in building an app. Project managers must be capable of explaining more creatively, more "humanly", than Jira is able to because the tasks are written for experts in development and design. The project

managers are able to translate and transform speech meaning among the different actors, so that the different concerns of the network are accurately represented. The fact that "Wayne is not excited in the right language" shows that he simply transports meaning, conveying it exactly as a developer understands it, but does not transform it to be comprehensible to another actor, the client.

However, there is one language that all of the actors in the network must become fluent in if they hope to flourish in the corporate context of making an app. This is the language of Jira, and this is another reason why I argue here that Jira becomes a focal actor in the network. A focal actor, as I explained in the Literature Review, induces and mobilises other actors to act, enrols the actors in their position in the network, and becomes an "obligatory passage point" through which all negotiations must go (Silva, 2019). Anika, as Project Manager, is directly a focal actor: she mobilises the employees to act and labour, enrols employees in their position through providing them with labour, and all discussions about labour must go through her. I have depicted Jira as a bossy, non-human actor in the network that causes other actors to act and that affects employees' labour. However, Jira also indirectly performs all the functions that Anika does: Jira mobilises action by depicting tasks, enrols employees in their positions by providing these tasks according to function and design or development work, and all negotiations about labour must be routed through Jira. Because "work does not exist if it is not in Jira", Jira for the App Company team becomes the entity through which all conversations, decisions, and labour must be routed – making it a kind of conscription device but, even further, a focal actor.

This is most evident in the meeting with the irate Client, where he resists running all of their interactions and the labour required to create his product through Jira. The App Company team,

however, continues to bring all of their mediations back through Jira, as Anika asks him towards the end of the meeting to send screenshots of issues that he finds through the ticket desk that is linked to Jira. The other aspect of this specific form of labour that the Client questions is the language that App Company uses – one can almost hear the air quotes dripping with derision around sprints and tickets as he says, "I understand in theory you guys want to operate with your 'sprints' and 'tickets'...". This language the employees have taken from the Scrum framework in general and from Jira, more specifically.

I argue that by positioning Jira as a focal actor in the network through which all labour and decisions must go, the language of Jira becomes the language that all actors in the network must use – the language of sprints, tasks, backlogs, and stories. Jira is marketed towards and most used by technology companies – typically fairly abstract work that as a result can be difficult to break down into smaller tasks. Therefore, it is significant that the developers of Jira chose to use the language of storytelling – epics, stories, and tasks – that is so synonymous with myth, magic, and understanding time. The use of this language in a corporate environment piqued my interest as an anthropologist, as this is a phenomenon that is not new (see Dove, 2002). In Figure 2, the different levels of tasks show that Epics are the first level used to structure tasks. An Epic is a project to be completed for the client – the design team designing wireframes of an administration dashboard for an app, for example. Epics are broken down into stories and from there are further classified as tasks and put in the Backlog. Examining the effects of designating labour in this way bears a striking resemblance to Haraway's (2006) call to study "the systems of myth and meaning structuring our imaginations" (p. 12). Epics are typically long narrative poems that recount an adventure of a hero; stories are shorter and can take any narrative form; 'task' is perhaps not so obviously a part of storytelling terminology, but 'tasks' are a common element of fairy tales, where the protagonist must complete tasks and

then receive a reward (George A. Spiva Library, 2022). What is the function of choosing a medium that is arguably so fundamental to our experiences as humans?

Figure 2

Structure of task levels

Note. From Rehkopf (2022)

Kearney (2002) states that Aristotle argued that "the art of storytelling – defined as the dramatic imitating and plotting of human action – is what gives us a shareable world" (p. 3). Through the use of such language, the daily work of the app makers becomes a part of a shared world of the company and allows the employees to share their work in a way that makes sense to the other team members because it is a part of that world. Jira becomes the language of the app's production, and the team members must teach the clients this language so that they can become a part of the app's world. Jira provides the language for these relations, so the developers of Jira indirectly influence and shape this company and its labour. The ideas, methodologies, and even organisational shape of a company based in the Global South are being shaped by the developers of Jira from the Global North; these Global relations are mediated by Jira, bringing this new language into Africa.

Jira is a focal actor of all the encounters in the network, transforming the words and requirements from the client into necessary labour – making what the client wants intelligible

and understandable to the App Company teams. It is in creating these shared histories, stories, or worlds that there is a "transition from the flux of events into a meaningful social or political community" (Kearney, 2002, p. 3) or, Jira transforms the descriptions (events) into set units of labour that are comprehensible to the App Company employees. This is most evident when a team member challenges how the network has constructed the world of their stories. The elements of its creation are also revealed, as in each first meeting with a client, Anika must explain to them what a sprint is and how App Company works in "story points".

Stories also function to give structure to events – there are beginnings, middles, and ends. The act of storytelling turns "fragmented moments into a pattern, a plot, a mythos" (Kearney, 2002, p. 4). It is in making a pattern out of seemingly disparate tasks that acts of labour turn into a 'project' with a direction. Luca explained:

> When work is being done, there is tasks and a method to the way you're doing things. So you're not always left in the dark. For me, …that makes a huge difference, because then I can focus on the task and not stray off of that path.

Luca's assertion that he would not "stray off the path" is further evocative of the paths and journeys in fairy tales, where the protagonist must journey to an objective and is warned against leaving that path. If he does, unfortunate events will befall him. Stories provide humans with an inherent desire to stay on the path, to find the method of travel and stay with it. In this above example, Luca was contrasting the method of work at App Company with his previous experiences at other design agencies where he felt that how they structured work and the hierarchy made it difficult to "navigate". At Budgeting App, Stefan explained to me that he and Ryan were attempting to "[get] the design team structured in their thinking [because] the design team desperately needed a bit of process, a bit of structure...[it] makes a massive difference in their output". Learning with Jira and becoming fluent in Jira's language brought

a language from the Global North to a company in the Global South and structured their labour according to this language. Therefore, I argue that Jira's language of stories for organising abstract work functions to bring comprehensibility and structured patterns that are comforting if this structure is understood. For those who do not or refuse to understand the language of the app's production, it could seem confusing, as with the irate client. Thus, to achieve the objective of building the app, all actors must come to an understanding of this language. The irate client does not understand this language, so he struggles to communicate with the network.

Finally, I asked Anika about whether there have been times when a task has been misassigned or lost, and she replied:

> Within the team…there is times where [someone] create[s] a task and assign[s] it to someone…They can't find it unless they get the link, or they create a task in the wrong place. Yeah, [a] task was created in the wrong location.

After this, as the queen of Jira, she had to move the task to "the right place." Place and location are another important element of storytelling, as stories would be retold to make sense to a specific audience or group of listeners; it was in this retelling that the story became material and recognisable to that group. Thus, "putting a task in the wrong location" further reveals the constructed nature of App Company's retelling of how they labour with Jira and re-enforces Jira's power over the workers in causing them to tell the story of their labour and time in a certain way. This could make it more difficult for workers to resist the darker aspect of labour management tracking, which erodes their autonomy, as employees might feel that they share in constructing this world, and sharing in it 'correctly' provides rewards in the form of working harmoniously with co-workers and Jira.

I have argued thus far that Jira should be considered a non-human actor in app production so that the material effects Jira has on employees' labour is not lost. I began this chapter with an overview of agile methodologies and how they make the labour of employees in corporate technology companies more flexible and adaptable. I then introduced Jira as a non-human actor, companion, and focal actor in the network and world of app production. As a non-human actor, Jira affects how the employees at App Company labour: it structures their time, language, and methods of labouring. Understanding Jira as a companion allowed me to consider and depict how App Company would 'not be where they are today' without Jira and that Jira also requires feeding and input from the employees of App Company. As a focal actor, the language of Jira becomes the language in which all actors in the network must become fluent to contribute to the maintenance of the network. By using the very human and universal language of stories, Jira the bossy non-human brings together the different actors, interests, and phases of the fairly abstract work that is design and development to give these two teams a shareable world, one they then have to share with clients and users. Lastly, I depicted how the double vision of care suggests the darker side of project management tracking, which can erode the autonomy of employees. I will argue in the following chapter that making an app is not an abstract, 'magical' process, but rather one that is grounded in the materialities of space, objects, and infrastructure, and how to allocate and value labour.

5. The Material Substance of the World of Apps

Today, we are moving offices: from the smaller, downstairs ones to upstairs with more space. The space is now too small with more employees back in the office post-Covid, and everyone wants a real kitchen – not just the table in a cramped old filing room that currently serves as one. The office day begins normally, as the plan is to work until lunch and then start moving, but there are hints that everything is not operating as normal. It is slightly quieter without Anika and Wayne down the hallway, as they have already moved upstairs, and the internet is slower. The WiFi router moved the previous day with Wayne and Anika, so the rest of the office is sharing WiFi with the other offices downstairs. The slow WiFi is making it more difficult to get work done, and Zintle and Bongani continually grumble about something not loading quickly. Just before lunch, we start the process of moving. I am sure that no one expected it to be as involved a process as it was. In my head, moving would mean simply carrying all the furniture – desks, chairs, tables – and the computer monitors upstairs. However, I had forgotten about the inner workings of an office – and an office that specialised in technology at that. It begins with unplugging the power cables for all the monitors and laptops (approximately 20 cables) and spooling them to be carried more easily, then moving to the phone lines and the phone, the printer, and the printer cables. Each electronic device is carried and moved with cables and cords trailing out from the back, the exposed guts that are usually tucked away under the desks now cumbersome and inconvenient. The stairs seem to get longer with every trip, and everyone is sweating and panting. I am standing on the landing at the top of the stairs when Luca and Aphiwe start rounding up the stairs with the heavy glass table bound for the new boardroom. I flatten myself against the wall and watch them manoeuvre it through the doorway, then go downstairs to bring the printer, receiving jokes from Aphiwe about leaving the men

to carry the heavy things – a continuation of our banter around gender and gender roles. I happily point at Jessica, who is currently bringing the stack of heavy bookshelves up the stairs. After all this effort to deconstruct and move, it turns out that ripping things out is easier than putting them back in. As we get upstairs, only Zintle and Connor know how to plug the network wires back in, so we have to wait until they are available. I then help Zintle run the wires through the walls, pulling off the plastic compartments on the floor (the purpose of which I have only just learnt), and we thread the wires through all the offices. We crouch and crawl along the floor, using our fingers to pry into the compartments, to grasp at the wires as they become visible in this office's compartment, and then hammer the compartments back into place. As I contort between Wayne's office chair and the wires compartment, I think of Vertesi's (2012) description of how scientists contorted themselves to 'see like' the Mars Rover to position it correctly. I had never before lain on the floor where the wires are, or even thought about where they go. On this day, however, I gain a visceral understanding as my fingers sting and my knees chafe on the carpet. In contrast to WiFi cables, phone lines must go into the ceiling. This is a task that only the taller people in the office can do, as they even have to stretch to get to the ceiling while standing on a chair. Connor takes this task on, knocking out one of the acoustic ceiling tiles and throwing the lines through. Unable to help further here, I wander outside the office where Connor is working and find Aphiwe running an extension cord to his desk; there are no plug points out at the front of the office where he is set up. At the end of the day, most of the guts have been shoved back inside the walls, compartments, and sockets where they belong, reconnected to their power sources, and no longer trailing awkwardly.

In this chapter, I will depict some of the material dimensions that 'matter' the world of the app – that give the world its more feelable and traceable edges. I will show that an app does not simply emerge from nowhere; instead, it emerges from particular environments and locations that are grounded in everyday negotiations of space, infrastructure, and labour.

Narratives of technology advanced by technology companies typically present technologies as seamless, abstract, and mystified – users do not understand how they work or how they were made, and to maintain the veil over the possible exploitative relations surrounding technology, technology companies work to ensure that technology remains mystified (Couldry & Meijas, 2019). An example of this is the idea of the phrase 'high tech'. Henderson (1998) interviewed some engineers and designers, asking what the term 'high tech' meant to them. What emerged from this question is that there was no conclusive definition of 'high tech', but from their answers, she made the argument that 'high tech' functions as a myth. When consumers do not understand a technology, when it appears to work as if by 'magic', they deem it 'high tech', she argued.

As Arthur C. Clarke noted so astutely, "Any sufficiently advanced technology is indistinguishable from magic" (Marín, 2018). However, once consumers work with the technology in their everyday lives and become familiar with its workings, the technology loses its glamour and magic. It is demystified. Henderson (1998) continues: "The aura of high tech and the use of new technologies…continue to add status through the mystification of the mundane and messy work practices that are necessary to accomplish the goals of the job" (p. 196). 'Magic' has long been a theme of anthropological literature. Malinowski "thought that [magic] could be found when human beings were confronted with a lack of knowledge or ability to control something important to their lives" (Science Enclyopedia, 2022), and Evans-

Pritchard explored how the Azande tribe used magic to explain occurrences in their lives and to protect themselves from misfortune (Science Enclyopedia, 2022). In this chapter, I am less interested in thinking about 'high tech' as a form of magic, or as having magical functions. Instead, I am interested in how technology is constructed from material relations that challenge a view of technologies as being magical, inexplicable, or outside the realm of everyday matters.

My focus on grounded matters I take from the literature on care, which suggests that it is the "everyday practical doings" (Puig de la Bellacasa, 2012, p. 199) about which anthropologists should care. Therefore, I will reveal in this chapter how it is not through ineffable technologies that apps are built; it is not magic. As I showed in the vignette at the start of this chapter, the office space must be set up and constructed – workers must assemble the means for creating technology using their bodies and technology components that were made by other workers. These trailing guts are a part of the world of the app because, without these awkward material configurations, the app could not be made. Just as Vertesi (2014) argues that infrastructures at the local level must be configured in imperfect ways, so must the app makers work to assemble all the different systems, infrastructures, and programmes, in all of their functionings and failings, to build the app. Working in a company in South Africa also requires further workarounds for loadshedding, losing electricity, or slow WiFi. Thus, the fact that the app is made in an office (in South Africa) means that it is constructed out of a specific material space, both in terms of office furniture layout and how labour is ordered. How workers materially organise their offices and work day, utilise and construct their infrastructure, and draw on material metaphors that ground and make concrete their work in their endeavours to organise their messy labour (Pink et al., 2016) are useful for disrupting the ideology of the 'seamless' narrative of technology.

5.1. Material Organisation

As I showed in the vignette about 'moving day', the office space must purposely be organised and constructed. When it is packed up, to move it elsewhere, the elements of its construction are revealed; the edges of the infrastructures' seams become visible, and the efforts that are made to cram these seams back together are more obvious. In other words, the material edges of an app's world become 'feelable'. Agreeing with Seaver (2018) and his arguments, the physical and social architectures echo the software, or in this case app, architectures. Thus, examining how the workers and their bosses materially delineate the office space (Gibbard, 2016) and whether workers use the spaces as they are intended (Sachs, 1995) is indicative of how workers materially organise their world by creating associations and making boundaries to cut ties, and how these material configurations are reflected in an app. Understandings of constructing material space can also be applied to the work designers do in the app – designers move objects to suit their needs better and create associations and boundaries in the app.

5.1.1. In the Office

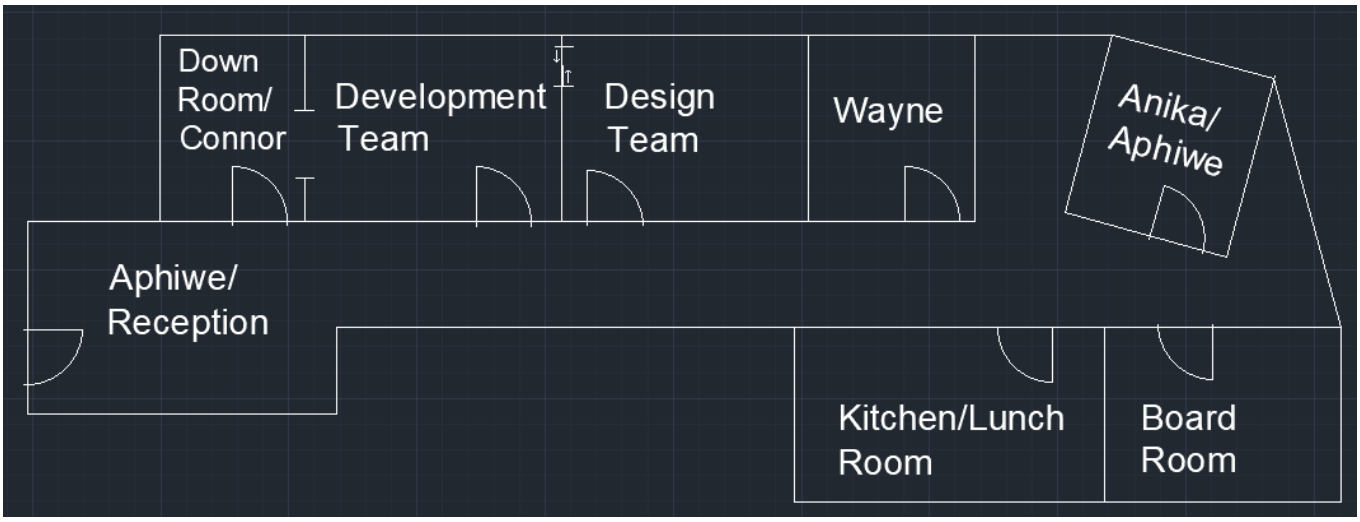
I will reflect here on the constructed nature of the office space, which I will utilise in the rest of this chapter to show the intentionality of choosing these spaces. The first is the negotiations that occurred around what to do with me, the anthropologist. As the anthropologist, I did not have an obvious working space. In my initial negotiations with Wayne about coming to the company, I was going to swap off days with Luca to share a desk and computer. However, as Connor moved into Wayne's office, Connor's desk in the development team office became available. Thus, despite my desire to sit with the design team, out of the necessity of how the space was constructed – there being no space or desk for me in the design team office – I had to start in the development team office.

I could only move into the design team office when the office moved upstairs, the design team office was bigger, and a spare plastic table was available after they bought more, new desks. My new place in the design team office then disrupted the WiFi connection, as it was placed on my desk with Luca's jutting up against it and was constantly bumped. Eventually, Connor mounted it to the wall to keep it stable. In this way, the organisation of the office affected both my own and other team members' labour. The other way in which the office was constructed is very literal: building desks. With a bigger space and a new developer coming into the team, the company bought more desks for the development team office – which had to be assembled. Luca and I were tasked with constructing the desks, which took Luca away from the work he was doing for the design team. As we lined up the boards, screwed in nails, and slid in the drawers, we were literally labouring to build the office. Desks are constructed to make an office space, tables are moved into available space, and so the material conditions of building an app reflect the contingent and flexible nature of an app that is always being re-thought, re-designed, and updated.

I will start with team members who are not primarily designers or developers, as the organisation of their space is perhaps not as obvious a grouping as the designers and developers. At App Company, team members who straddle different categories of labour are grouped together or on the fringe of office rooms, indicative of how the organisation of the

Figure 3

Basic Schematic Floorplan of Upstairs Office



Note. Drawn by myself in AutoCAD.

office reflects the boundaries of labour that go into making an app. Connor, as office manager who also does some front-end development for websites, now occupies an in-between office space. He is still connected to the developers, but he has his own space so that he can talk with clients. It is connected to the developers' office with an arch and a separate door; the development team office can also be entered through this space. Previously, this space was designated as a kind of 'breakroom' or an extra space for eating lunch, with a couch to sit on. However, as the company spent more time in the new offices, no one used this space for their breaks since it was directly connected and visible to the developers. Instead, breaks were 'stolen' quietly by hanging around in the kitchen when making coffee and getting 'caught up' chatting with another team member. Because of the transparency in generally knowing what everyone is working on, made possible by Jira as outlined in the previous chapter, visibly taking a break could be frowned upon. Indeed, I myself, even though my work was not always visible

to others in the office, was influenced by this aura of productivity and would perhaps linger a little longer in the kitchen than needed rather than take a visible break.

Anika, as project manager, began her time in the new office with her own separate office space so that she could talk with clients, also denoting that her labour is understood to be different from the designers and developers. Aphiwe, who handled both support and some project management, started in a liminal space in the office – not enclosed in an office space, but setting up his desk in between all the other spaces. This initially worked well because, as he was in charge of support, he would play a 'reception-like' role. As it became unnecessary to have someone play this role, since clients only occasionally came into the office, he moved into Anika's office to further learn from her about project management. Finally, Wayne occupied his own office space, signifying his status as business owner. This set-up has only slightly changed from when the offices were located in the smaller space downstairs in the office building, as Connor shared an office with the developers and then with Wayne. Aphiwe also worked at the reception desk. The fact that Connor is not in Anika's office could reflect that Anika is higher in the office's hierarchy of roles, and thus explains why she was initially given her own office space. Aphiwe only moved into her office because she suggested it. Thus, as the duties of management are perhaps more varied and less 'obvious' than those of the designers and developers, their place within the office shifts more readily and often according to the current needs of the office.

For the design and development teams, the design team was set up in one office and the development team in an adjacent one; this was the same in the old set of offices. The two teams have thus been divided into the two main phases of app creation: design and development. The proximity of their offices – sharing a wall – indicates the close association between these two

phases in app-making, as teams from each phase affect the work that the other team does. As I discussed in the previous chapter, Jira affected the company's division in these two teams by mediating and providing them with a shared language to discuss and negotiate their labour. In conversations I had with team members, this boundary between the phases in app-making and their respective teams was a topic of discussion. Sitting with Zintle and Bongani, the two developers, in my first week at the company, I asked them about the difference between the designers and the developers. Zintle responded, "[The design team] just do; they don't think. Then we have to sit and think before we do it." For the developers, then, the work of designing does not involve mapping out logic or thinking about how their designs will affect the development of the app. By contrast, Jessica felt that…

> …"there's a lot of logic on [the designers'] side of the board, because the client's just like, "This app needs to do this. Please show me." And then [the designers] have to sit there and sketch out everything and decide how many buttons it needs and what's happening in the app. So basically, by the time it gets to the developers, they already know what the app's gonna do.

The separate office spaces are also indicative of the divide that the designers and developers felt between their labour on the app. However, they also recognised the importance of working together, and this became a greater emphasis towards the end of my fieldwork. Bongani felt that the proximity of the developer and design team offices was useful because it made it easier to resolve issues that arose and to collaborate across projects. The importance of physical proximity for effective collaboration was also indicated by Elize, now working at Budgeting App, as she described to me her previous experiences working in a larger office with more employees. In this company, she explained that the design and development team interacted a lot, and they worked in the same office space. The product team, however, sat…

> … "in a different section; they're not part of the design team. So getting information about the specific product [was] very difficult and tough…[Getting information] should be quick; it should be easy. But it really depends on the layout of the company."

Materially organising the layout of the different teams' workspaces is thus an important part of creating stronger relations in the network that reduces the pulls of tension and friction, as disconnects between the different teams and phases affect the product of the app. However, this close collaboration between designers and developers within the agile framework is a fairly new development that has arisen with user-centred design (Jones et al., 2016). Herman from EskomSePush recounted how, when he was developing apps for banks in 2012, "there's a lot of things always that are not designed, especially back then, like…how should the interaction be…should it move like this or like this." He also described how he would often disagree with designers he was working with at the time because designers in 2012 did not all have "this same love for mobile or something. So they wouldn't understand how a button should look."

The process of designing with developers in apps has therefore evolved to value collaboration, as in Bongani's statement about the proximity of offices, but there are still conflicts about how to achieve collaboration. Jones et al. (2016) found, in a survey of designers and developers working in agile software environments, that developers' main concern was designers being on call to provide immediate ad hoc feedback to developers. By contrast, the main concern for designers was having access to information to facilitate greater teamwork (Jones et al., 2016). Thus, while materially organising office space is an important part of collaboration, simply putting offices close together will not resolve collaboration issues, as shown in the following ethnographic example from my data about doors.

When thinking about office spaces anthropologically, the question of doors is important, as noted in the literature by Gibbard (2016). Workers can rework these spatial boundaries to suit their needs (Sachs, 1995), as doors act as a kind of moveable and malleable boundary, taking this view of boundaries from Haraway (2006). For example, Anika often left her office door open so that team members felt that they could always speak to her. Even though she was in a different office space materially, she did not want this to become a barrier to effective communication. From Elize's description of product managers above, it is clear that she thought that allowing this flow of work between different teams and spaces was important for maintaining a smoother process of work. For me, it seems that for an app to emerge smoothly from the office space, material boundaries between teams must not be too concrete but must be malleable when necessary. Wayne, as CEO, also noticed this, and to facilitate greater collaboration between the development and design teams, he suggested leaving the doors open between these two offices. In the new office, each of the development and design team offices has a main front entry door, and then there is a sliding door at the back that connects the two offices, shown in Figure 3. The designers and developers agreed that the sliding door, rather than the main office doors, would remain open as it was less draughty and disruptive. The argument was that it would make spaces feel more accessible, as it would require traversing less space. The door to the design team office would also stick and require a forceful shove, resulting in team members attempting to leave and enter the room as little as possible. Leaving the sliding door open – creating a gap in the boundary – was a workaround for the separations in material space, and it was decided on in a way that best suited workers' needs. However, this decision was constantly contested, just as boundaries are dissolved, moved, and reformed. The sliding door would often quietly be closed and remain closed for the rest of the day. When Wayne brought this up with the teams, Amanda explained that it was the development team who would close the door. Wayne's suggestion was that the development team use noise

cancelling headphones to cope with any noise from the design team's office. Even after this conversation, the sliding door would regularly be closed when teams gathered in the daily stand-up meetings – there could otherwise be a lot of clashing voices – and would not be opened again until Wayne noticed. Thus, while the teams value collaboration, this suggests that they also prefer a separation between offices when not working together so that a focus on their own work can be achieved. It is also indicative of how the contestation of boundaries between actors reveals tensions between different concerns.

Negotiations around office spaces were also revealed when moving into the new offices. However, contestations did not arise so much around which groups or figures to put into which offices, as spaces mostly followed the previous ones from downstairs and slowly migrated and changed according to the changing needs of the office. Debates instead arose over how to organise team members within the chosen office space. In the design team office, Jessica and Amanda organised their computer screens to avoid glare from the window near their desks, with Amanda explaining that she struggled with glare in the other office. These attempts came up against the issue of ensuring that, together with Luca's desk, there was enough space between all doors and desks to not feel cramped. In these discussions, it is clear that the hierarchy of the team does not materialise in the organisation of the inner office space. The hierarchy began with Amanda at the top as head of the department, then to Jessica as the head designer, and finally to Luca, who had only recently entered the company. The focus, however, was on gaining the configuration that was most conducive to creation – no light glare and a sense of openness. In the development team, however, the hierarchy of the team was made more explicit. Connor had first choice in picking his desk space, as he had been working in Wayne's office and had been at the company the longest – putting him closer to the top of the developers' hierarchy seniority of expertise and length of employment. Zintle contested this,

desperately wanting the desk by the window that Connor had chosen. In the end, this resolved itself as Zintle worked at 'Connor's' desk until he returned from Wayne's office, and Connor moved into the adjacent space not long after. The argument that I have been unfolding here is that it is not only Jira, the software queen, that shapes the design process but also the hardware and material things in the office space: tables, doors, routers, and electricity.

Virtual working is an interesting aspect of office work that arose following the Covid-19 pandemic, as hybrid and remote working options have been implemented by many companies. App Company has implemented a fairly flexible approach in this regard, allowing employees to choose to work from the office or from home (up to two days per week). Amanda explained how the company attempted to make remote work more possible and more flexible:

> We did used to have a lot of fixed stations, which we've now converted…to allow people the flexibility of more the hybrid workspace if need be and, I mean, Covid ongoing send-home commands. So yeah, we've now shifted pretty much everyone over…everyone's now on portable stations, so we can have hot desks…even though we're currently at a fixed desk, you have the ability to move around and work remote if the need arises.

While all employees preferred to remain at their fixed desks, rather than 'hot-desking', some employees would work remotely on set days every week, while others worked from home according to their needs for that week: one worker contracted Covid-19 at one point, four others became unwell with flu at different stages, and several had to deal with family issues. At all these times, they would work from home if they were able. In these cases, stand-up meetings would take place both over Google Meet and in the office. Workers also used Google Chat to ask team members questions if they were not both in the office. This affected the atmosphere in the office by making it feel much more relaxed and less structured; lunch time often went

on for an hour instead of the designated half-hour, as we simply continued to talk. However, Bongani argued that while virtual meetings and remote work can work, he preferred it when everyone was in the office physically because it saved time having face-to-face meetings. Similarly, Luca explained how he sometimes found talking over Google Chat frustrating because it was more difficult to receive feedback on his work; rather than a small "check-in" with Amanda, he would work on a task for a while, send it to Amanda on Google Chat, and find he was going in the wrong direction. Therefore, the material set-up of the office does have an effect on the work output of employees. At Budgeting App, the design team worked completely remotely, with the development team mostly in the office. They employed many programmes and strategies to make the boundary of distance more navigable, allowing communication across material space. Thulani and Elize work closely together as UI (User Interface) and UX (User Experience) designers. Thulani explained the importance of their remaining in constant contact, in conversation if not physically:

> We collaborate a lot; we're constantly communicating...So with [Elize], I work with her way more than the head of design. Because whatever work she works on…I need to understand what is she doing in order for me to be able to do my part of work. So we actually collaborate a lot on a lot of storyboards. We're constantly on ideation mode together; sometimes she'll just ping me if she's not understanding anything. And then same goes for me; I just ping her, just jump on a call if we have time...it's a seamless process that we currently have.

Therefore, the app makers have re-constructed their virtual space to be very material: at Budgeting App, they use virtual storyboards, a collaboration space that allows the designers to work together and 'brainstorm' as if they were together in the office, and the designers emulate quick chats in the office by using impromptu video calls. Not being together in a physical space is thus a boundary that can be worked around.

Lastly, a matter of concern and care that is a part of the world of an app in South Africa is the unreliability of electricity and WiFi. In South Africa, the supply of electricity is not guaranteed. The power grid took on a much greater load in post-Apartheid South Africa, as "post-1994, Eskom embarked on a massive electrification scheme, connecting townships and households that were not connected to the national grid during apartheid" (1922 - 2019: The rise and fall of Eskom, 2019). In the past decade, due to "a 'significant loss of critical skills', with 'poor quality of maintenance' and 'poor workmanship' causing breakdowns" and funds that are siphoned through corruption, Eskom has not been able to meet South Africa's ever-increasing energy needs. The result is unplanned outages and the no less inconvenient planned outages – loadshedding. This has had a major impact on South Africa's economy, as businesses must take extra, often costly, measures to maintain the electricity supply, such as buying generators and inverters. Such issues have a great effect on any business but perhaps most particularly on a technology company, where computers are their most vital tool, and work cannot be done without them. Amanda explained for them to be able to do their work effectively they require…

> …stable internet, which is a challenge that definitely has quite an impact on the entire office if the internet is either down or unstable, hence us having three [routers] to bounce between, so we have tried to create as stable a workspace as possible like having the inverter, having multiple connection points. But yeah...that is one of the bugbears of Africa.

The first thing that is turned on every morning are the inverters and WiFi and, when electricity goes out, only desktop monitors and WiFi are allowed to run on the inverters; even laptops must be unplugged if they are charged enough. All lights are turned off so that the office is completely dark, except for the blue glow coming from the screens, and no one can make tea

or coffee. Thus, when electricity goes out, as it inevitably will, computers – their tools – are prioritised over personal comfort, as work cannot be done without these electricity-fed tools.

5.1.2. In an App

Designing an app requires designers to consider how to lay out graphics and text in the app to be both visually pleasing to users and logical to navigate. Although this takes place on a computer, it draws on traditional understandings of material space and how to move through that space. I will historicise the disciplines of User Experience (UX) and User Interface (UI) design and how the original principles of these concepts reflect a consideration of materiality.

Just as development and design have historically been considered distinct and are now being brought into greater conversation and collaboration, so UI and UX design have evolved into co-constitutive elements of design. In many organisations, UX and UI designers have separate functions, while others advertise them as a dual UI/UX role. For the purposes of this section, I will consider them as two sides of the same coin. While UI design can be understood as designing buttons, logos, fonts, and UX refers to user experience when interacting with the product (Dumont, 2021), these can be seen to be co-constitutive, and their history evolved simultaneously. UI/UX design could seem fairly abstract, taking place as it does on digital screens and being mediated using a mouse and cursor. However, designers draw on principles that are not far outside our everyday realms of experience. The idea of ‘user experience’ has its roots in feng shui and ergonomics – ways of organising space (Stevens, 2021). Feng shui is about arranging objects in the most pleasing and user-friendly way, while ergonomics considers interactions between humans and “other elements of a system” (Stevens, 2021).

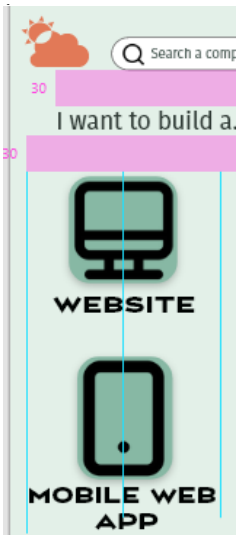
These descriptions place user experience squarely within the realm of both anthropological understandings of space and the co-production of human and non-human actors that is so prevalent in STS. In 1955, an American industrial designer stated that, “if people are made safer, more comfortable, more eager to purchase, more efficient – or just plain happier – by contact with the product, then the designer has succeeded” (Stevens, 2021). As personal computers were released in the 1970s, psychologists and engineers worked together on users’ experiences. Apple then had a big hand in developing the term and realm of user experience, as in the 1990s, their cognitive scientist with the title ‘User Experience Architect’ was the first person to have ‘user experience’ in their job title (Stevens, 2021). The history of UX/UI design can provide insights into how to analyse and understand space, the co-production of humans and technology, and how much of its grounding comes from engineering.

UX/UI designers organise space within apps to achieve the most pleasing and efficient layout. As I will show in Chapter 6, learning the conventions for this is a part of visual culture and tacit knowledge. From both my own experience of designing an app and observing other designers at work, current software has absorbed this understanding of space into its own design. For example, whenever a designer moves an object in Adobe XD, lines with centimetre

measurements on them state the object's proximity to other objects and put a crosshair in the centre of a section to help space perfectly, as shown in Figure 4. In encounters with design

Figure 4

App design displaying measurement marks



Note. From the app designed by myself in Adobe XD

tools that prompt designers to lay space out perfectly, designers will continue to design apps that have a particular understanding of space. This is also clear in how designers made decisions based on not 'cluttering up' the screen too much for users. In an encounter between a client and the designers at App Company I observed, the client wanted a hamburger menu, drop-down menu, and a menu bar, but the designers cautioned that this was not 'best practice' and would confuse users because of the overabundance of options.

The co-production of technology and humans is the main organising theme of this dissertation, but, more specifically in terms of UX/UI design, the evolution of user testing is a marker of the greater understanding that technologies are made *for* humans. As Tracey, a product owner, so aptly put it: "[I've been] working more within the UX realm and realising that there is no tech[nology] design that's not human-centred. You just can't do it without consulting the human because otherwise, what's the point?". Henrik explained to me that he started his projects by asking his design team…

> …to always start with, why are we doing this? From the customer's perspective…why do I as a customer need this feature? What will it do that, one, even makes me give it the attention that you want from me, since I already have other things that are…shouting for my attention? And two, if I start using it, what will it solve for me?

These statements evince the growing understanding that purposely *designing* apps is important, and that design should emerge out of a process of interaction with the users of apps. In many ways, this means that UX/UI design is only now beginning to follow more closely engineering design, since engineers have always needed to *design* their products – be it buildings, infrastructures, or machines. As the truncated history I provided above hinted at, engineering has been an inspiring force for digital UX/UI design for some time. This inspiration can be found in terms that app designers use daily, with wireframes and prototypes being the most apparent.

Prototypes have been in use longer, and understanding their origin makes the function of app prototypes much clearer. In engineering, prototypes are an early form of a product, so engineers can test its useability and performance (priorityprototypes, 2021). The focus at this stage is not so much on how the product looks but on making sure that it works. Prototyping in app design functions to show how users will interact with an app, such as where the screens will lead to. It gives designers the opportunity to test designs' workability before sending it to developers, as it is much easier to make changes to the prototype (the design) than to the product (the development). Amanda used the following analogy to describe prototypes in app design: "It's like buying a house online versus going and walking through the house and getting a feel for the height of the ceilings and the feel of the door handles. It's that tactile element to it." The use of the word "tactile" is telling here for the link between app prototypes and engineering prototypes. Amanda is aware here that the designs she creates will eventually be completely

tangible to a user, as users will employ their forefingers to tap on buttons she placed equidistant apart. Thus, the tactility of designs has become of greater importance with the advent of touchscreen phones, which means that the user is not mediated in their interactions with the app through the buttons on the mobile phone.

Bringing together the function of prototypes and principles of user experience, another way in which designers organise an app is in mapping where each screen, button, and tap should go. They demonstrate this in prototypes, but working this out is one of the first conversations they need to have with new clients. The designers and product owners work with the client to think through how to organise the space of the app – its world. Anika explained to me that clients will come to her and say…

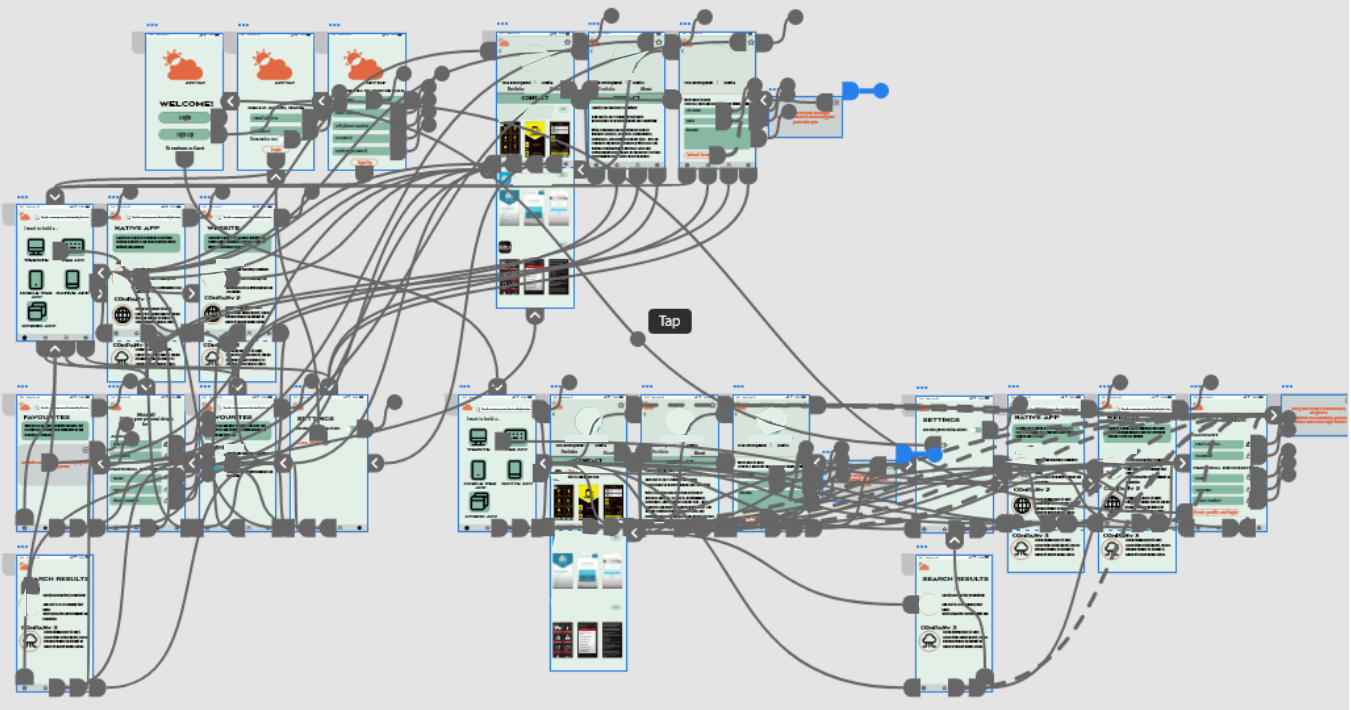
> …they want page 1 to 5. And then…me and the [design team], we've got to try and decide, okay, page one to five, how do they interconnect? How do they link? Where do you go from that page to there? How do you go back to the homepage?

Constructing the organisation of an app is thus a vital part of the 'design', but, like the cables trailing out of the walls when moving, users are not usually aware of the purposeful effort that has gone into making the app seem "logical" to them. The designers and product owners set out the threads of relation – the cables of infrastructure – that run behind the visual face – the walls – of their designs. Figure 5 shows all the threads of prototyping that run through a small app to connect all the screens and buttons, a kind of '2D' representation. They create associations between certain aspects – for example, taking the user to the home screen when they tap the login button. However, they do also create boundaries, or cut the ties of relation.

If the user does not have a profile, they might be prevented from accessing certain features of the app – a boundary has been put up to their entry.

Figure 5

Prototyping of an app - mapping the threads



Note. From the app created by myself in Adobe XD

5.2. Classifying, Organising, and Allocating Labour

The basis for considering how labour is involved with creating digital technologies I take from Pink et al. (2016), who posit that the digital and the material are "entangled elements of the same processes, activities and intentionalities" (p. 1). Agreeing with Puig de la Bellacasa (2012), although not explicitly, Pink et al. (2016) argue for accounts of how these digital–material entanglements emerge in processes of messy becoming. However, as I discussed in my literature review, the ways in which an arrangement of labour is documented and arrived at involves both social and political negotiations about what should be included and how to depict what is included (Bowker & Star, 2000).

The second aspect of material organisation is the way in which companies organise teams' labour and how this labour is valued. If this labour is not organised efficiently and valued correctly, then the cost of labour will be greater than the output of the company, reducing profits – an issue since good profit margins are the main goal of a private corporation. Thus, companies must fine-tune production processes and determine where and when to put workers' bodies and what to make them do. Additionally, paying attention to the modes of production involved in building the commodity that is an app implodes the world of the app, challenging the fetishization of the app product. These allocations of labour also clarify the distinction between designers and developers, as the labour of teams is allocated according to their respective heads of department and which product their team is currently working on.

5.2.1. Valuing and Evaluating Labour

As I discussed in the previous chapter, if the app is being built in a corporate environment, then work on this app will be organised according to what is known in the business world as a workflow methodology. Most of the research participants in this project worked in technology companies that followed the framework of Scrum, an agile methodology that focuses on iterative processes of feedback. In this section, I will discuss how this methodology affected how time was manipulated and assigned to labour. Central to understanding how I am using the term "labour time" is Marx's concepts of 'socially necessary labour time' and 'abstract labour'. Abstract labour presents labour in terms of a relation between labour and time: labour–time. Instead of measuring labour as concrete in terms of the goods produced, labour is measured by units of time to make it more quantifiable across goods (Bonefeld, 2018). Labour can then be paid for using money, as the labour–time is paid for according to its exchange-value rather than its use-value (Bonefeld, 2018). Abstract labour also contributes to the commodity fetish, as the outcome of abstract and commodified labour in capitalist production

is that the commodity appears to be severed from its mode of production, or the relations of production, so that consumers only value the exchange-value of the commodity. 'Socially necessary labour time' builds on the idea of abstract labour to depict the amount of time that is required for labourers "to produce the equivalent of their own livelihood" (Encylopedia of Marxism, 2018). Surplus labour is the labour workers complete in the latter portion of the day, at least in the factory model that inspired Marx's thinking, when this socially necessary labour time is complete, and they are working only for the profits of the employer. The greater the worker's productivity, the more surplus labour is completed. The concept of socially necessary labour time thus encompasses the evaluations that business owners must make for workers to both reach their livelihood needs and for the business to make a profit. How a technology company like App Company organises their workers' labour and commodifies it by receiving remuneration from clients is based on abstracting and subsequently valuing employees' labour. 'Time' is an important part of how App Company values workers' labour; however, it is not the only mechanism that is considered, and this process of valuation is not only made by the managers and company owner. As I showed in the previous chapter, Jira also shapes how labour and time in the company are understood and allocated.

The process of "Scrum" structures labour in a way that differs from the rhythm of more traditional working weeks: work will be planned to take place over a period of 10 days, with nine days of working and the tenth day used for a "sprint retrospective". A sprint is the work that must be accomplished within that period, and this work is determined in "sprint planning". The "retrospective" acts to evaluate how effective work and task completion were in that period, with employees suggesting ways to alter work in the next period for it to be more effective, thus drawing workers into evaluations of both their own labour and the assemblage of labour in the company as a whole. At the end of the period, the product owner and the design

team update the client on the work completed during the sprint in a "sprint review" meeting. At Budgeting App and other companies that work directly for users and not through clients, the sprint will be when they present their work to their managers for review. These meetings are thus the main points of gathering between the different actors in the network, where work completed or incomplete is put at the centre of the discussion and negotiations. The sprint organises work in a way that prioritises for workers what work has to be completed within a smaller, more conceivable period of time, thus artificially cutting time and making boundaries. Just like with Scrum, the term takes its roots from sports. A sprint is a short spurt of concentrated energy, usually in running – evocative again of movement and quick adaptation. These metaphors are therefore a part of the vocabulary that app makers use to construct the world of apps and creates the sense that labour must be efficient and productive.

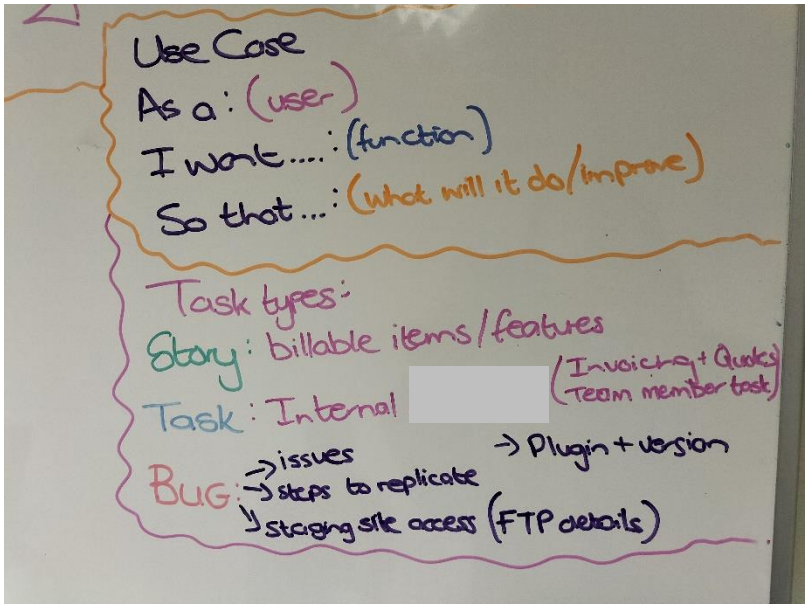
Bowker and Star (2000) argue that the final depiction of an information infrastructure does not display the "practical politics" that went into deciding what is displayed. Viewing Jira as holding the information infrastructure – the way in which the teams classify tasks and forms of labour – means that I must examine the activities of how these decisions about labour are arrived at, affected by Jira, to avoid hiding these negotiations behind the final decisions. I will begin with the "sprint planning meetings", in which the managers would evaluate labour that is required to complete the project as a whole, labour that has already been completed, labour that still needs to be completed, and place all this information against their understandings of what is feasible to complete within the company's designated period of work. At the beginning of a new sprint, Amanda sat with Anika in the "sprint planning" meeting to evaluate and assign tasks to design team members for that period of work. For the development team, this planning was about how to take a client's needs and requirements and turn them into a "use case" for the development team (shown in Figure 6). Sprint planning meetings often took place in the

conference room to avoid disturbing other team members. These meetings were usually scheduled to take place the day before the end of the sprint and took two to three hours.

I will briefly recount the different kinds of tasks as they relate to understandings of time. As I discussed in the previous chapter, Jira breaks down labour according to epics, stories, and tasks, and the App Company team employs this labour designation to distribute work. Direction for the Epic is taken from the agreed-upon scope that was drawn up with the Client, based on the "mock-up drawings" of what the app will require to function as the Client envisions it. Creating mock-up drawings is the first place at which the flow and idea of the app are made concrete; as I will explain in the following chapter, it is the malleability and means of visual communication that they provide that make them such a key part of the process of negotiating

Figure 6

Description by Anika of Labour Components



Note. Photographed and used with permission by Anika

with the client. Figure 6 was drawn by Anika to explain to Aphiwe the different types of tasks. What is interesting to note here is the ascription of some types of labour as being "billable" and others not. As Anika had to draw this diagram for Aphiwe to understand how to manage and assign different types of labour, the system of classification may not be as clear as the diagram

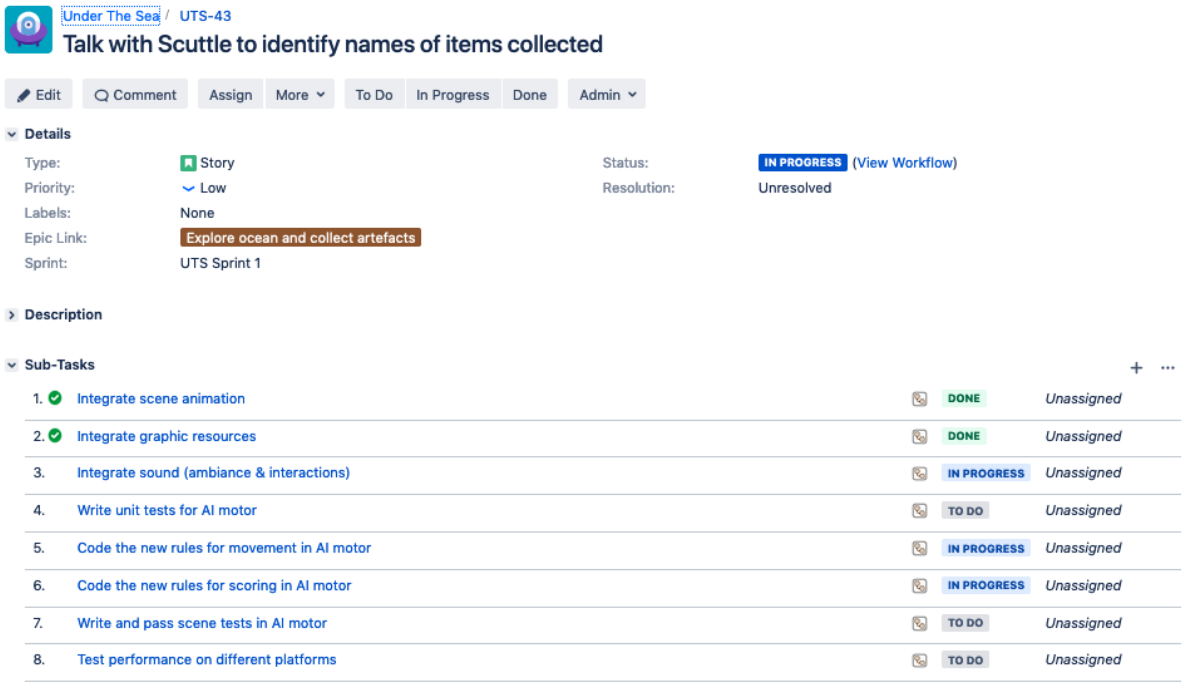
might suggest and, thus, what they charge clients for and what they do not may also not be as clear. I will return to a consideration of this later.

As a large part of this evaluation and assigning of labour happens with Jira, it is useful to consider how Jira reconfigures understandings of labour. Working with Jira, Amanda and Anika were able to take their evaluations of team members' skills and work to be completed to cut labour into smaller pieces that can further be abstracted, manipulated, and rearranged according to other constraints and needs. The main feature of Jira that teams laboured with was the Scrum board, a summary of all work being undertaken at a snapshot in time. In this section, I will discuss how managers allocated work to workers by drawing on the framework of labour provided by both Scrum and Jira. In Chapter 6, I will discuss how creating tasks, maintaining the Jira board, and abstracting the labour visually (Wilf, 2016) is a labour in itself that contributes to Jira's status as conscription device as well as companion.

The first way in which Jira affected the teams' labour was how it shaped the 'cutting up' of labour into more manageable pieces. For example, I sat in on a planning meeting, where the mock-up document was a dashboard that was convoluted with many more elements than they were used to working with. A dashboard is what the owner of the app uses to administer the app 'behind the scenes', where the owner can see all the information that users have inputted into the app on their computer screen. The document had been drawn up by Vanessa with a client who, by this stage, was not taking much guidance on 'best practice' in design, and the design was filled with incredibly detailed tables with pop-ups coming off of almost every column. Usually, each screen would become a task or story, but for this document, each screen had so many elements that Amanda and Anika were unsure. Amanda asked Anika if they

Figure 7

Story with sub-tasks



Note. From Campbell (2019)

should rather break down the screens, so that the table became the story or task and pop-ups be designated as sub-tasks. As Figure 7 shows, sub-tasks can be individually designated and tracked on the main task. Amanda explained in a meeting with another developer that, for the design team, she would "rather break [labour] down into achievable goal points with the stories and then each story has data...Otherwise…they can't track their progress." There are two things to take away from this quote. First, the way in which tasks are "broken up" is a subjective process that can be managed to suit the needs of the head of the department and their team – time is malleable. Second, while Jira tracks the App Company workers, they also use Jira to track themselves, and thus become entangled with Jira and the managers in an abstraction of their own labour.

However, a task does not necessarily mean a similar evaluation of time for both the design and development teams. Amanda explained that the design team seemed to work more "project-based", while the development team worked more "task-based". As "project" has connotations of being more comprehensive and a longer period of time than a task, I take from this that the design team tends to work in bouts of work that appear as one task but would take longer than a development team task. A task for the design team, for example, would be to design the user journey for the landing page of the app. For the development team, a task could be a "bug" that would take 15 minutes to do, while other tasks might appear short but end up taking much longer if the developer struggles with the code. Thus, for each team, tasks hold different understandings of the time required to complete them. In this way, time becomes a boundary that is more easily malleable in a system that allows it to be defined, calculated, cut up, and, most importantly, moved. From this, I argue that Jira is not only a task master but also the king of abstract time; although it is Anika who is the queen of Jira as she understands how to wield it, Jira has the ultimate power as king to influence how the company views time and labour as abstract.

The way in which labour is cut up is closely related to understandings of labour–time, and it is the abstraction of concrete labour into abstract labour that has facilitated the emergence of "story points". A story point, according to App Company documents, amounts to four hours of work and a set billable amount. The product owner and the head of the department give each story or task a story point, a decision made based on "work complexity, the amount of work, and risk or uncertainty" (Radigan, 2022) and whether the number of story points seems "doable" to team members. Story points are therefore not just about the time they will take but also the effort required to complete the task. For example, a conversation between Amanda and Jessica, a design team member, displayed this evaluation of both time and effort:

Amanda*:* Two story points for that work? Do you think that's fine?

Jessica: Yeah, because one of those pages will duplicate.

Discussions about story points among team members explicitly revealed negotiations around labour–time and valuations of labour (whether the effort required related to the time allocation), and how this related to remuneration from the client (ensuring that story points were not over-allocated, resulting in billing the client too much for the work). Each team member could only have a certain number of story points on their board per sprint, and so the story points are again important here in structuring their labour. In the sprint planning meeting that I sat in on, Amanda suggested that a task under discussion would take only one sprint because they could reuse designs for previous screens. Anika cautioned, however, "Don't underbook". It was a fine balance between getting as many tasks done in the sprint as possible, and thus optimistically evaluating the labour time required, and ensuring that this did not result in over-allocating work that could not be completed. This tension is further understood by considering it in terms of socially necessary labour time, as the cost given to story points must encompass both socially necessary labour time and surplus value. The time paid for in story points by clients directly correlates to App Company's estimation of time on how long it will take to complete a component of the final product. In this case, time is not split up by days but by story points. The evaluation of employees' salaries is based on the number of story points they are expected to complete in a month, but they only receive part of the full value of that story point, with the rest being taken as profit. If the quality of an employee's labour is not to the expected standard or if a task is more tedious or complex than they estimated, then more time will be spent on the task (or story point) than budgeted. This cost must then either be absorbed by the business, eating into the surplus labour portion and reducing profits, or the client has to be charged for more story points. It is important that the company get as close as possible to correct estimations so that it does not forfeit profits, so Weber's arguments about rationalisation still

ring true: app companies attempt to classify and assign tasks according to structured time frames to determine the amount and value of labour needed to make profits. For Weber, modern bureaucratic life is characterised by an attempt to eradicate emotion and mystery and replace it with rational calculation (Rationalization, 2022). The aim is to bring the world under greater control and to make it predictable and calculable; in this case, making the cost of labour more calculable. Working with Jira is an attempt by App Company to bring the labours of the team members under centralised, visible control, where each task can be suitably cut up, labelled, and assigned a time frame. This gives the task predictability and calculability, making it possible to put a price on this labour based on clear calculations.

5.2.2. Contingent Labour

This 'rational' system of calculating and evaluating labour in Jira contrasts with App Company's previous system of evaluating and costing labour, in which team members would time how long it took them to complete a task, and clients would be billed based on this time-tracking. According to Jessica, it was not possible to tell a client, "This is how long it's gonna take; this is how much you owe us because we didn't know that beforehand." Amanda further explained that with Jira…

> …we can pre-allocate resources, which in this instance, are team members, so we can say that this batch of work is going to be X amount of story points, which equates to X amount of hours, and so you know that that team member is going to be booked up for the next two months, because they've got that allotted batch of story points associated to them.

Discussions about allocating story points are therefore central to the new labour process, and team members are involved in assessing and making decisions about their own work and labour effort. It seems that labour can be allocated more efficiently in terms of remuneration, meaning

that the client will, in theory, receive a more accurate estimation of the cost associated with the team's labour.

This description shows that managers understand that labour is a resource and, in a business context, it is vital for making a profit that resources are deployed and used as effectively and efficiently as possible. A system that allows for using team members (the resource) as efficiently (in the least time) as possible is therefore highly valuable to business owners for reducing labour costs and increasing possible profits. Such a rationalisation of tasks is especially interesting, given that the process of 'designing' is often thought to be a purely creative process that is often undervalued because clients do not have an understanding of the work involved. By breaking design work down into calculable tasks, it makes this visible to the client to give App Company's valuation of this work validity and legitimacy. However, such rationalisation is therefore in tension with two 'creative' strands: the design work involved in making apps and the language of storytelling used to structure tasks in Jira. Thus, taking inspiration from Haraway (2006), myth and tool mutually constitute each other: technology and the tools designers use to make technologies are co-constructed together with our metaphorical and imagined understandings of myth, stories, creativity, and making.

Additionally, I was uncomfortable with the fact that the smallest unit of time available for labour was four hours – what if the work should take fewer than four hours or not amount to the complexity associated with a story point? Did the App Company team only ever do the work that was originally set out in the tasks set out in the sprint planning meeting? I argue here that designing an app involves more contingency and ad hoc labour than this overly rational explanation of the labour system encompasses. I queried Anika about how they assess the story points, and our conversation is as follows:

Anika: So currently how we'd assess is, say I had like an article for you to write up, I'd ask you how long that article would take. And then you would tell me how many hours. Now, with one story point, it's up to four hours. So, then you tell me how long it would take you; if it takes you four hours to do that, or if it even takes you two hours, we'd say one story point…

Hannah: And if…it ends up being fewer hours in the end, or more hours, and then the story points are wrong?

Anika: Fewer is fine because then you've completed a task in the time. There are times where it's meant…we have allocated 10 story points, but it's actually 17. And that happens with some of our contract work, and then we carry over to our next sprint. So, we allocate…a client will have three story points per sprint, and say the work was five story points for that specific thing, so we'll do three story points worth of work in that sprint and the other two story points in the next sprint, and then we'll carry on in that way.

From this conversation, it can be seen that story points are, in many ways, more of an approximation of the work required, which can be manipulated and altered in contingent acts according to the situation at hand. The valuations of labour shift constantly as they encounter the actual labour required, and the story points are wielded and forged to be more reflective. There is an attempt to bring even this ad hoc labour under some rationalised control, as tasks can also be created "ad hoc" that were not included in the original backlog should the client raise a further issue that is not covered by the original, agreed-upon service-level agreement (SLA). There are then "SLA story points" and "ad hoc story points". "Ad hoc story points" cost the client more than "SLA story points", since they were not included in the original payment and estimation (the scope) of the project. These "ad hoc story points", however, do

not always cover all of the extra, contingent labour that the App Company workers do, as with Wayne exclaiming in the previous chapter that the irate Client received "40 grand worth of free dev[elopment]". Anthropologist Malaby (2009) also explains in his study of an organisation that worked with Jira that "Jira offered both more flexibility and more standardization…[but] as flexible as Jira was, it still could not 'see' certain kinds of nondiscrete tasks" (p. 82). Jira then is also an actor that 'sees from' a particular point of view – a concept discussed further in the next chapter – that constrains her ability to see past this explicit (Sachs, 1995) view of work. Malaby's statement also indicates that Jira is only capable of managing discrete and visible labour and cannot see the "more complex world of problem-finding, problem-solving, deciphering, decoding, understanding, and collaborating" (Sachs, 1995, p. 42) that has also been a focus of this dissertation. Additionally, it is this standardisation that Arthur – who, amongst other roles, has worked in UX design for some years – finds problematic. He explained to me that agile methodologies are "productionising creativity" and making it more "rational". I infer from this statement, responding to Sachs (1995), that he does not think agile methodologies take into account the contingent nature of creative labour and allow space for the tacit work practices that must occur between activity-based work. Thus, the highly adaptable and flexible form of labour promoted by agile methodologies and implemented with the help of Jira could suggest that this methodology is not suited to a more contingent, creative process that is not always rational or quick.

Furthermore, Jira and her seemingly total rationalisation of all tasks do not function alone. Rather, as I argued in the previous chapter, Jira works as a companion with the team members. Continuing with Weber's concept of rationalisation, he argued that modern capitalism promotes the idea that personal relationships should be removed in the process of rationalisation (Guru, n.d.), as they are not calculable. While the task board operates to

rationalise the teams' tasks, it does not act to make them impersonal. Research participants across the companies I interviewed expressed the idea that there is an intention not to have the "task board" be the only point of contact between team members. Behind the task board are discussions about tasks, demonstrations of how an element should look, and queries about the nature of the task – that all happen face to face or on private chats via GoogleChat or Slack. While the task board was important for regulating and tracking the labour of team members, it did not operate as the only means of regulation. There were numerous formal and informal meetings and conversations that occurred about work that allowed the Jira board to function.

Thus, the abstract, rational representation of work on Jira is built on somatic, social relationships in real life. This was recognised such as when Tracey explained, "at the end of the day, Jira is part of the process…very seldom can it be used in isolation; you need those daily stand-ups; you sometimes need briefing meetings for bigger tasks, all of it." The stand-up meetings took place every morning in each designated team group – design, development, and support and management team – in-person or via GoogleMeet with team members' referring to and questioning tasks in Jira. Therefore, these meetings provided additional space for checking team members' work progress and for expressing concerns or confusion about specific work tasks. Often, more complex aspects of the task that would be difficult to explain in the task description were discussed in this meeting, as it was possible to refer to the wireframes and physically show team members what needed to be done.

Less formally, questions about the task would arise throughout the work day if a team member needed clarification. If this conversation turned into something that Amanda needed to query with Anika or that Anika needed to query with the client, it was noted formally and physically, either in the "Comments" section on the task or in an email. Indeed, Emily, working for an

international online marketplace, complained that there are so many meetings – thus prioritising the importance of relationships and contact – that she struggled to complete her assigned work. In living with Jira, there are rich social worlds of relationships that the labour depicted in Jira is built on, without which Jira cannot function.

Finally, Jira affects and alters the way in which the App Company team members understand and depict time. First, Jira prompts teams to use time in the sense of "urgency" by project managers' assigning tasks a priority level: low, medium, high, highest. This indicated to the team member how soon in the sprint the task needed to be completed. These priorities further "break down" the sprint into smaller sections of time, meaning that team members not only plan their work across sprints but also within them. Classifying time in increasingly diminishing increments makes time feel infinitely malleable and controllable. Second, the sprint as a period of time alters the more traditional working week. sprints do not always fall neatly across the typical Monday to Friday work week, with sprints sometimes starting or ending in the middle of the week. There are then two classifications of a working week running concurrently, so the scrum classification of time is a boundary of time that pushes up against the traditional time boundaries of the Monday to Friday work week. The designers and developers then re-enforce the "scrum" time boundaries by naturalising this depiction of time. All of the team members I interviewed who spoke about sprints in how they worked felt positively about how the sprint structured work. Thulani saw the sprint as being that "we need to make sure that within the two weeks, we ideally finish those [set tasks] off. So if there's other work, ad hoc work, you know, I always just push them towards the end of the week." Thus, workers contribute to the construction of Scrum's time boundary by working within this structure. The extent to which classification of time is naturalised, and where the construction of this boundary is clearest, is when team members talk with clients outside the company who

are unfamiliar with “scrum”. When I asked Anika if they had to explain terms like sprint to clients, she replied:

> Most of the time, yeah. Even…with our processes, when we said sprints, a lot of the clients were not sure what we’re talking about. So we even created videos on what sprints are, what happens, and I’ve even told our clients we work in two week sprints, and this is how it is.

Therefore, as Bowker and Star (2000) explain, naturalising boundaries contributes to constructing group membership – in this case of working in an organisation that uses Jira – as members take Jira’s boundaries of time for granted. In this case, by naturalising these understandings of time, the actors emerge as app makers who are part of the corporate context. As the clients do not take these boundaries of time for granted and possibly even contest them, they emerge as actors outside of the organisation.

In this chapter, I have pointed to the differences in how the designers and developers set up their working spaces and how this set-up both creates opportunities for collaboration and creates a boundary between the two teams. I also depicted how the concept of material space can be used to understand how an app is designed and laid out. I then argued that Jira and agile methodologies construct the labour and time in a company to be abstracted and flexible, as tasks and labour are designated to employees in set and rationalised forms. However, I also showed that this rationalisation and the formal representation are constructed on social relations, informal conversations, and ad hoc, tacit labour that complicate a neat rationalisation of workers’ labour. Lastly, the main organising tenet of this chapter is that the abstract cannot work without the concrete: for the digital world of the app to be built, it must be grounded in material constructions of labour, time, and space.

6. What Vision Can Offer to an Understanding of Corporate Design Processes

I am sitting in on my very first meeting with Jessica, Amanda, and Anika. They are examining the wireframes they have been working on, checking that they have done all the work they agreed on with the client ahead of the upcoming meeting that afternoon. We are sitting in the boardroom, still in the downstairs office. Amanda has wirelessly connected her laptop to the television in the boardroom, showing on the screen aspects of the work Jessica has been doing on Adobe XD. The meeting is a long one, as they have done a lot of work in the sprint, but there are also many elements that they are unsure about and are trying to solve together. One of the discussions centres around the chat option in the app, where users can chat with a representative in the app to ask questions and get information. The topic is the colours of the chat bubbles: should the colours be different when receiving automated responses than when chatting live with a person? The other debate is whether the representative should be named or be called a representative of the app. As they talk, they refer to chat options from other apps: how they do their colours and how they refer to their live representatives. Jessica also brings up two options she has designed: one with the same colour throughout and one with differing colours, and they flick between these two designs. In the end, they decide on the "simplest" chat option: only two colours and a generic representative. I gather that this decision is the "simplest" for two reasons: it does not confuse the user with varying colours, and it will mean less work for the developers, as they can have the same name for all the representatives rather than having to determine how to programme in conditionalities for different names, and one colour can be attributed to each person in the chat. As I watch them clicking through the screens, seemingly pressing 'buttons' that work and scrolling through a 'cart' that has goods in it, this is my first introduction to wireframes. Is this the app? I find myself wondering. It seems to function like one,

> but when they move out of the 'app' view, there is a series of blocks laid out with lines snaking busily between them. As the meeting ends, I ask questions about wireframes. They explain to me that wireframes are the graphical presentation of the app, and then they joke that clients often think this is the completed app. As they laugh, Amanda looks at me and explains, "It's like getting an architect's plan for your house and then asking if you can move in."

In this chapter, I will argue that the visual mode is used to communicate between different actors in the network – namely, designers, developers, and clients. This communication is further made possible by the properties of the conscription device that wireframes and Jira's Scrum boards possess. I will also argue that Puig de Bellacasa's (2017) concept of vision-as-touch allows me to consider how designers 'see as' users. Finally, I will show that how Jira allows workers to visually abstract their own labour affects how they and their employers view their labour as malleable.

The opening vignette is indicative, I think, of at least two things. First, it shows how wireframes are used to communicate to the client and other team members how the app will look and function – allowing for visual problem solving. Second, it hints at the fact that, even though wireframes are a useful communication tool, there can still be things that are unclear to the client and, by extension, to the user. I draw here rather literally on Miller and Woodward's (2007) concept of the "blindingly obvious", where something is so ubiquitous that consumers no longer truly 'see' its construction and maintenance. In this case, every day users look at apps, these ubiquitous technologies, without seeing the 'why' behind the design choices. In creating these designs, and in viewing them, a visual culture is being co-produced that rests on a range of understandings of what visual elements signify.

In the term ‘best practice’, designers respond to a meta-sociological narrative of what is most understandable to users in the visual culture, thus reifying it. However, this narrative is negotiated on the local level as clients decide to ‘go against’ this best practice or in designers not quite being sure of what that would be in the context. The ‘darker side of care’ (Lindén & Lydahl, 2021) is also useful here for understanding work designers do as non-innocent, as they must necessarily marginalise another design option in choosing to follow ‘best practice’. Constructing wireframes and prototypes makes these negotiations visible and function, I argue, as conscription devices that enable different actors to comprehend concepts. In the first section of this chapter, I will discuss this first aspect of how visual elements are used to communicate meaning. However, drawing on Puig de la Bellacasa (2017) and her argument for “Touching Visions”, I will think about what embracing touch can mean for and contribute to these visual knowledges. I take her arguments both literally – by thinking of the embodied, touching engagement that comes with using an app on a smartphone – and in the more theoretical sense of thinking about “chains of touch that link and remake worlds” (Puig de la Bellacasa, 2017, p. 120), as “the haptic holds promises against the primacy of detached vision, a promise of thinking and knowing that is ‘in touch’ with materiality, touched and touching (Puig de la Bellacasa, 2017, p. 95). Considering vision as touch as Puig de la Bellacasa (2017) suggests is productive for my thinking about how designs affect both the actors – users and makers – as touching creates a sense of relation.

6.1. Translation and Transformation in Design

As I discussed in my Literature Review, visual culture is “a way of seeing that simultaneously both reflects and shapes how members render the world” (Henderson, 1998, p. 25). It is through developing a visual culture that the way in which humans see and what humans see is

developed. In line with Bear et al. (2015), this 'culture' is constructed through many actions and interactions of varying actors. As a part of developing a visual culture, in this case for app designs between workers in technology companies and users, it is necessary for actors to develop a shared visual literacy. It is only in constructing shared visual literacy that 'readers' (users) are able to understand the 'language' of the visual element being viewed. I will bring this discussion together with Boxenbaum et al.'s (2018) arguments for understanding how visual elements are used in organisational contexts as a mode of communication.

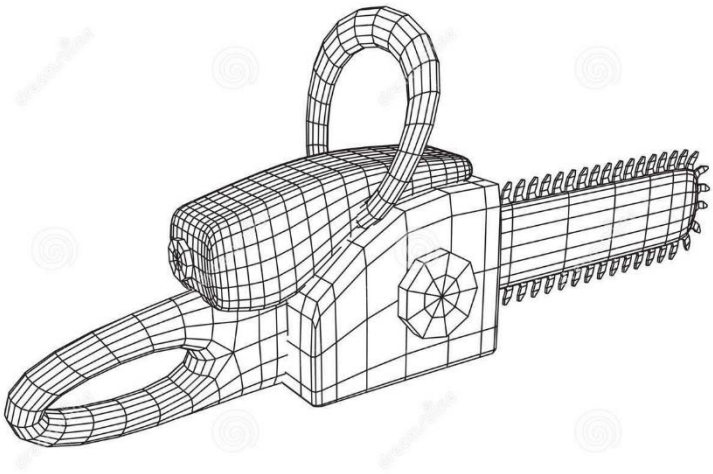
Therefore, I will discuss in this section how different actors in this corporate context work together to develop a shared literacy that conveys meaning to users and how these visual elements are used as a means of translation between the different actors. I argue here that viewing both wireframes and Jira as a 'gathering ground' for negotiating different forms of knowledge reveals the contestations that are facilitated using both visual and embodied means. Another important visual aspect in the organisational context of building apps is Jira. Jira is a highly visual information infrastructure, as I discussed in the previous chapter, but it is also a type of conscription device.

6.1.1. Wireframes as a Gathering Ground

I will begin with a description of how wireframes have been used historically and how this has endured in how wireframes are used in app design. The term wireframe has emerged more recently with the advent of Computer Aided Design (CAD) for showing 3D objects in manufacturing, with no detail so that it looks like it is drawn in wires (DevToolsGuy, 2015), shown in Figure 8.

Figure 8

Example of a CAD wireframe (chainsaw)

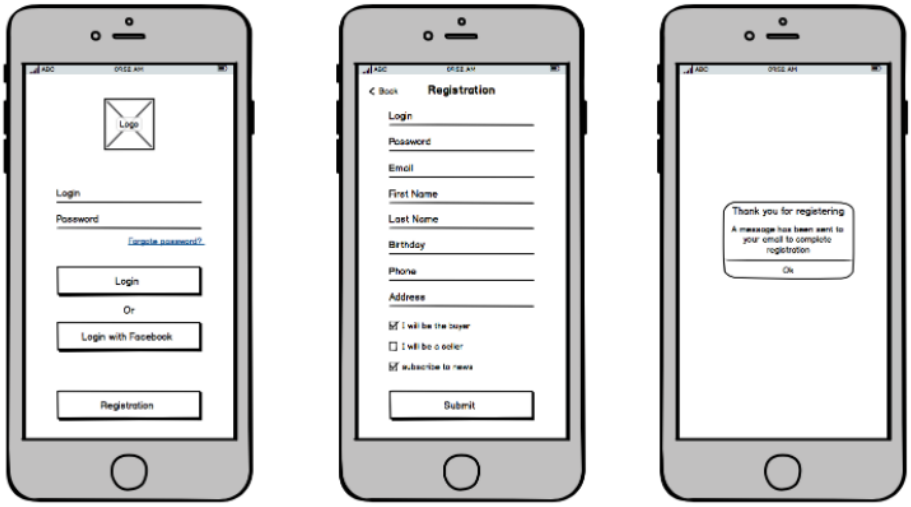


Note. From Newbi1 (2022)

For apps, wireframes are sketches or frameworks of an app design. Different levels of wireframes have evolved in app design, depending on the personal preferences and structures of companies and design teams. There are low-fidelity wireframes, which are the type of wireframe that most resemble those from the CAD drawings. As shown in Figure 9, low-fidelity wireframes are often pencil drawings that only have squares where the images would be, no text, and a sketch of where the buttons will go. App Company uses low-fidelity

Figure 9

Functional Specifications/Low Fidelity Wireframes



Note. From Product Plan (2022b)

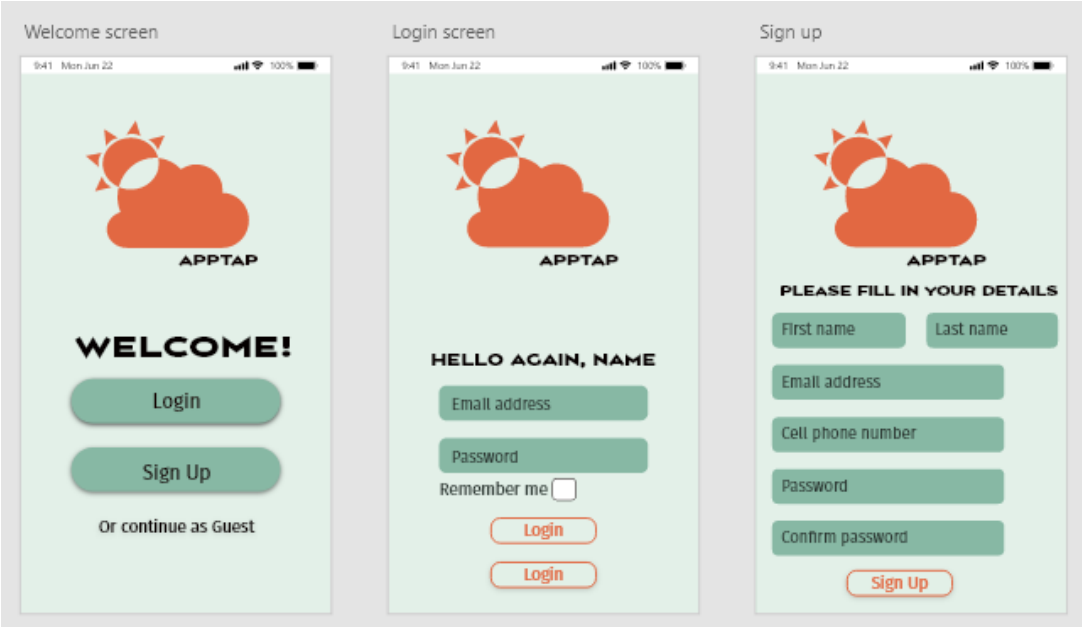
wireframes as a kind of "functional specifications document", as they call it, which is used to depict the functionality that the client requires in an app. In this document, each screen is drawn up, showing what functionalities it needs to contain and how each screen will link to others. This document is important for two reasons. First, it makes the agreed-upon scope clear, as the document acts as a physical, unambiguous representation of what an app will look like and what functions it will have. Any later disputes about function can then be referred to this document, as the client has signed off on it. Second, it helps the client become adjusted to thinking through the step-by-step logic required for developing an app. Low-fidelity wireframes are most associated with the early stages of the design process, as designers map out elements for the first time, arranging and rearranging them through various stages of ideation. When explaining to me the first stages of ideation that she does with clients, Vanessa showed me an A3 page that folded out into a messy sketch of arrows, boxes, and writing. She developed this document with her client, mapping out how the client would want users to move through an app. She later drew this up into a more identifiable 'phone screen'. There are then

high-fidelity wireframes, shown in Figure 10, which are intended to look close to how the app will be when it is developed, with fonts, logos, colours, and representative images filled in.

In contrast to prototypes, however, these are static. When prototyped so that clients can "click through" – referencing touch and tactility – the app, high-fidelity wireframes are so realistic that clients often think they are the actual app, thus resulting in Amanda's joke about thinking the architect's plan is the actual, useable house. It was around this time that I began to notice an emphasis on the language of sight used by participants: 'seeing the wireframes', 'seeing how long a project took', and 'seeing input from the developer'. I had also been playing with the idea of how different actors 'translate' their work and requirements for others. Both ideas come together in the concept of conscription devices and allow me to unpack how visual elements are used in an organisational setting that is made up of different actors. The distinction

Figure 10

High Fidelity Wireframe



Note. Taken from designs made by myself in Adobe XD.

between designers and developers and their different roles in the app production process is mediated by wireframes, as wireframes provide them with another common language; this time a visual one instead of Jira's language of storytelling. Furthermore, in thinking of vision-as-

touch, I can consider how these visual means of communication engage actors in a "world [that is] constantly done and undone through encounters that accentuate both the attraction of closeness as well as awareness of alterity" (Puig de la Bellacasa, 2017, p. 115). In other words, how do different actors come to 'see as' (Vertesi, 2012) others and 'see from' a specific vantage point as they design the app? Amanda's statement in the vignette about how clients think wireframes and prototypes are the actual app also shows that while non-professionals within specific industries have developed a cultural understanding of the function of architectural drawings, wireframes in manufacturing, and prototypes of objects, there may not be as much of a cultural understanding of these digital drawings (yet). Their powers for translation and negotiation are thus still being tested and developed, and an understanding of these drawings emerges out of encounters between designers and clients or non-experts in the field of digital design.

Low-fidelity wireframes are an important part of mapping out the making process for designers, as "They allow intangible ideas to become concrete – but still allow ideas to be reworked and renegotiated" (Henderson, 1998, p. 200). It is the malleability, the properties of the boundary object, of wireframes that makes them so valuable to designers and hints at the reason for the evolution of implementing a design process before development. The merit of examining wireframes so closely in how they are used in these corporate contexts is most evident in Henderson's (1998) assertion that…

> …visual representations, including prototypes, are not only devices for communal sharing of ideas but are also a ground for design conflict and company politics, exactly because they facilitate the social organization of workers, the work process, and the concepts that workers manipulate to produce a collective product. (p. 10)

Here, I will show how designers at App Company use wireframes to communicate with and resolve issues between developers and clients and to negotiate elements of the design. The nature of the boundary object is that it allows communication among different actors by being plastic enough that it can be altered to be understandable across varying situations, actors, and interests but without losing its meaning. This property further speaks to Henderson's (1998) assertion that visual representation is meta-indexical, as it is a gathering ground for multiple ways of knowing, seeing, and, indeed, touching. Wireframes can also be understood as conscription devices because conscription devices combine the information-storing properties of inscription devices and the plasticity and malleability of boundary objects (Henderson, 1998). As a conscription device, wireframes become a point through which all negotiations must pass because wireframes and prototypes have become "conscription devices that are so central to the design process that they must be used by anyone who communicates about the process" (Henderson, 1998, p. 200). This is first evident in how clients at App Company sign off on a functional specifications document that is formed based on a wireframe, constituting a kind of visual contract with App Company in addition to a more traditional written one. All future relations and negotiations are routed through this document, as designers refer to it for what and how to design and, if there are any conflicts with clients about what is within the scope of the design, Wayne will refer clients to this document. This is the first way in which wireframes are made central to the communication process. Moreover, high-fidelity wireframes are used as the main point of negotiation between all actors – between designers and developers to show the developers how to programme the functionalities, between designers and clients to determine these functionalities, and the product owner disseminates the wireframes among all the different actors. Thus, the first element of conscription that I will discuss is the way in which such devices allow diverse forms of knowledge to be held, negotiated, and conveyed.

As I see it, at App Company, clients typically come to the table with knowledge about the industry in which an app would be situated and what an app has to achieve for that industry. As Amanda explains, clients "may be aware of some new big development that now needs to be accounted for or modified to and [they] can get much more agile and adaptive during the process". Clients bring this industry knowledge, and it is the plastic and meta-indexical nature of wireframes that allows specific developments to be incorporated as the design process continues. This quote also shows that clients' knowledge of the industry becomes entangled with every aspect of an app. For example, clients tell the designers how to classify categories in an app because, as Jessica explained, "that's not really something we can just randomly come up with". The client's knowledge must then come together with the designers' knowledge of best practices in design and the developers' knowledge about how to programme an element for the best operation. Through engaging with the client and their own research about that industry sector, the designers become 'in touch' with that app's particular world, with Anika exclaiming that "We're basically becoming pharmacists" as they talk about an app situated in the pharmacy industry. Thus, designers are in a process of co-becoming of sorts with clients, in which designers must come to see both as the user and as someone who works as a part of the sector they are designing for. In this process of co-becoming and working across divergent concerns, various strategies of translation – as language, through the conscription device, and through touch and tangibility – allow for diverse types of knowledge to be negotiated.

These different perspectives, of 'seeing from' positions of alterity, is perfectly depicted in Amanda's description of the role of the designers:

> [Clients] won't necessarily understand all the jargon [of development]. It's a language on its own, and it's not just one language; it's about 12 [referring to programming

> languages]. So we found that communicating with the clients in more the design element has been much more effective than the developers communicating with them because they miss each other. A developer views the world in 2D typically, and a designer will veer to 5D. And clients normally view it in 3D or 5D, and they're much more big picture on stuff, whereas the developer will view a single element at a time. So design seems to have become a little bit of a bridge between the two.

In her statement that design "seems to have become a little bit of a bridge", Amanda is intuitively hinting at the meta-indexicality of wireframes and the way in which wireframes facilitate translation and communication among differently situated actors. When she explains that developers and clients "miss each other", she is pointing to a failed translation, in which developers are unable to effectively transform their knowledge into a form that is comprehensible to clients. Designers, on the other hand, are able to accurately transform developers' knowledge, depict it using their own knowledge of design practices, and thus effectively communicate with the client in a way that sustains the network's relations. Amanda's explanation also suggests that the designers' role emerges from these processes of translation as being responsible for balancing divergent vantage points and learning how to communicate across them – their role is embedded with a situatedness in which designers are aware of their positionality in the network.

The designers, wielding their wireframes and prototypes as elements of translation, are well placed to bring together these different positions of the network. The first function of wireframes and prototypes is to depict to clients or test users (outside of the company) how an app will look and interact to obtain feedback, as wireframes can be reworked and negotiated in real time (Henderson, 1998). Clients' first experience with this is in crafting low-fidelity wireframes, where the main purpose is to formulate the "logic" of an app. The "logics"

governing apps are not as clear to clients as they are to app makers, and the visual mode enables the designers to make this logic clear or comprehensible to clients. Anika explained to me that making an app requires tapping into human understanding and thought processes so that an app seems logical to users, "but outside users won't understand the logic, but once you get in this world, it's obvious". Vanessa explained that in the ideation phase, the part that clients need the most assistance with is with…

> …the flow of things, so when people have got an idea, but their idea is made up of like a whole lot of words and a whirlwind, and how to get them into a structure that makes sense to a developer to be able to code.

Designers thus take the ideas and words of the client and translate – transform – these into a visual mode of communication that both the client and the developers can understand. The low-fidelity wireframes and later prototyping make this flow visible to the client; it makes the logic that is "blindingly obvious" visible and thus, it perhaps also suggests that it is "blindingly logical". While the client has typically thought like a user, the client often does not have to think like a designer or developer. In my interview with Avril, a client who had had an app made for her residential estate, she was most surprised by the in-between steps that needed to be worked out in the process flow. She used the metaphor of starting a car: As a user, she was using an app like a driver would get into a car and turn it on. Developers, on the other hand, see it as going to the garage, unlocking the car door, opening the car door, getting in, looking at the mirror. She explained it as being "because we don't think in...the minutia of the steps that it takes to do something...we're so automated into doing what we do that we just go from A to C without thinking there's B in between". Seeing behind this process, however, and watching an app be made gave her a greater understanding of the multitudinous elements involved in making an app. This initial "sketching out" is indicative of how the app building

must pass through this initial drawing stage, allowing the client to negotiate in real time with the design team.

As the designs move into the higher-fidelity wireframes, negotiations of logic and flow between designers and clients continue, while taking on the added elements of colours, images, and phrasing of wording in an app. The vignette at the beginning of the chapter is indicative of this, as designers use wireframes to determine colour options for chats. At App Company, Adobe XD is preferred to other design software, as the design team feels that it works better for their purposes than a product like Figma. The App Company design team feels that prototyping is especially important in encounters with clients. Such prototyping enables the design team to find any difficulties that users might have with navigating through an app, and they raise any difficulties with the client. Working through prototypes with clients also makes an app more comprehensible to clients, as intangible discussions about screens and flow become concrete in buttons that can be 'tapped' and checkboxes that can be 'checked'. Designers therefore put in extra labour to make prototypes as realistic as possible because, as Jessica explained…

> …Otherwise, they'll be like, this is a checkbox, but it doesn't check, why? ...or you can't really add a product here; is the app gonna be like that? … It's easier for [the client] to understand, this is what my app's gonna look like, rather than just giving them a sketch view, because then they want to know, how is my corporate identity gonna look on this? What does the button actually look like on my app and stuff?

Designers thus simultaneously draw on and construct a mutual visual literacy with clients as a means of communication, as prototypes make the abstract goals and objectives of an app more concrete and tactile. The designers, product owner, and clients can then solve problems using wireframes and prototypes. An example of designers reworking and negotiating wireframes

following feedback from clients was when, in a meeting with a client, designers were presenting the login screen that had been designed; after logging in, the client would be taken to the Home screen. While viewing the specific prototypes, the client queried whether it would be possible to "simplify" the login, allowing users to jump straight from the login screen to anywhere in the app. Amanda explained that she was unsure, as it might undo the prototyping that designers had already done. As the meeting ended, the client brought up the login again, and Amanda made the change, moving the lines of the prototype to link the login button across the app instead of only to the Home screen. This encounter is also indicative of the undercurrents of shifting power relations, where, because the client is paying for the services of the company, what they want will ultimately show up in the final design. In another exchange about a client's request for a design, Wayne and Amanda were discussing where to place the tabs on a design:

> **Amanda:** But when I spoke to you about it on Friday, you said I must do best practice.
>
> **Wayne:** That's how they want it, so that's how it has to be done.
>
> **Amanda:** Damn it, that's such bad UX.

There are often tensions between what is considered to be 'best practice' in design and what clients want for their design. This is indicative both of divergent objectives and of competing ways of 'seeing' and perspectives linked to the actor's position in the network. As each actor 'sees' in distinctive ways, the main process of negotiation is in reconciling all these competing visions, and it is the main source of conflict between different actors, in this case designers and clients. Designers use wireframes as a means of translation (using vision as a kind of language, thus drawing on visual literacy) with clients, displaying knowledge to be negotiated with. To possibly resolve tensions between a 'best practice' design and what the client's idea is, Amanda explains how they help Clients make decisions by…

> …giving them both sides of a coin… if it's a situation like that [where there is a tension between best practice and what the client wants], we would give them both visual options, and also show them the journey as to how both of those would play out. And most of the time they…see the input from the developer, and they will sway to that side, because there's a reason as to why we say what we say.

In this way, designers are able to convey their knowledge about best practice in design and developers' knowledge about the best way to implement the functionality simultaneously with the client's own ideas and needs. Visual representations enable the client to make a decision, as wireframes are a highly malleable mode that the actors in the network collectively alter and shape (Henderson, 1998). Thus, it is "because of their meta-indexical qualities that drawings and prototypes can so easily function…as boundary objects that are capable of being read on different levels by different [actors]" (Henderson, 1998, p. 200). The plasticity of wireframes sets them out as a useful conscription device, as wireframes are constantly readjusted according to feedback from the client and the development team but still are able to convey the vision of an app across all the differently situated actors in the network.

There are other strategies that designers use, well-established in UI/UX design, to communicate meaning to users. Both from observing changes to designs that designers made and from my own experience with designing apps, I came to see past the 'blindingly obvious' to notice the small things that designers do to prompt users to understand how to use an aspect of any app. First is the use of a client's brand's colours. As Hynes (2009) explains, colour is an important part of establishing a brand's corporate visual identity, as colours are a part of making any brand recognisable to consumers. Designers are aware of these colour associations, and one of the first questions they ask clients concerns the corporate identity document – a page with the client's logo, brand colours, and fonts. Designers use these colours throughout an app to re-

enforce brand identity to users. as the designers explained to me in a conversation, users might download an app, see it has different colours from the brand's recognisable identity, and then delete it because users assume it is not a legitimise app since the colours do not match. Having coherent colour choices is thus a means through which designers communicate to users that an app belongs to a brand. Because "colour may play a role in imparting information, creating lasting identity and suggesting imagery and symbolic value" (Hynes, 2009, p. 546), designers also employed colour to communicate with users about how to use an app. Out of the client's palette of brand identity colours, designers would set 'active' colours so that, as users become familiar with using the app, they would recognise these colours as 'tappable'. Once again, this was not an element of design that I was consciously aware of, like the client, but it was one that designers were very adamant about. In an exchange with one client at App Company, the Client asked for the ratings section in an app-in-the-making to be purple and not yellow, making a comparison with the AirBnB ratings system. Amanda explained to the client that if they were set on purple, designers could use a lighter purple because the darker purple was their 'action' colour; in other words, the darker purple indicates to users that that element is "tappable". Designers also 'greyed out' unchosen options so that users would know what element they are interacting with. This was made explicit to me when Jessica asked Amanda if she could do an "active colour grey". Because of the existing visual culture that has been created in app development over the past decade around the colour grey in apps, literate app users now read grey as an inactive colour, an option that is not available or cannot be tapped. Jessica's concern was that users would not correctly 'read' grey as an option that could be tapped. In the end, Amanda suggested that the design team use it since they did not have many other colour options with that company's brand's identity.

In setting these ‘active’ colours and being concerned about using grey, designers are employing routinised behaviour that enables them to overcome the tacit nature of aesthetic knowledge (Stigliani & Ravasi, 2018). Aesthetic knowledge is often tacit and difficult to explain – associated with ‘just knowing’ that something is right. Displaying aesthetic knowledge in wireframes and discussing among themselves and with clients why certain colours might not be used makes this tacit knowledge explicit and allows designers to articulate and work through otherwise hidden aspects of design. Another design principle that is a part of creating and drawing on visual literacy is icons in apps, in many ways a kind of communally derived body of tacit knowledge. Just as it is tacit knowledge that makes technological drawings usable among engineers (Henderson, 1998), it is tacit knowledge that makes icons designers chose usable and readable to users. If users are visually literate, with the necessary tacit knowledge, certain icons can immediately be recognised as denoting certain paths: a ‘house’ for Home or Main page or three lines for Menu. However, those who do not ‘see’ these through the lens of this visual culture do not yet understand this tacit knowledge. I witnessed this most clearly when trying to help my grandmother navigate through an app on her phone as our communication broke down in my attempts to explain to her what to do. When I told her to press the home screen, she looked for the word ‘Home’ instead of the house icon. For more options, I told her to go to the menu, intuitively knowing that this could be found by pressing the three dots in the corner, but she saw dots as no more than a collection of circles. Thus, as Henderson (1998) argues, in creating a visual culture, we restrict what there is to see and how we see; designers continue to use the three dots because they cannot see them as anything other than a menu, and I could not understand my grandmother’s hesitation because I have become literate in this visual culture.

In addition to allowing clients and users to visualise and materialise an app-in-the-making, wireframes are used to communicate with the development team about how to build an app. The development team uses wireframes for the artwork in an app and prototypes to know how to connect and programme the functionalities, again making wireframes a conscription device, as negotiations about developing an app with designers must pass through wireframes. As Anika explained, "instead of the client actually trying to draw up something with pen and paper, and the [development] team [not] understand[ing] what that means…we've taken that away to let us do that because we understand what the [development] team needs". The designers and product owners are thus able to translate more effectively for the development team than the client can, thereby sufficiently transforming the client's meaning. This is also because, in terms of translation-in-the-network, developers have contrasting concerns from designers and clients. As Thulani explained about developers at Budgeting App, "they speak a different language; they speak on functionality; the dev[elopers] don't really consider the users as long as it works, whatever that they're doing, they just want to see it work." However, while wireframes are capable of holding diverse forms of knowledge, there are limitations to their capabilities as the sole method of communication as a visual language. If the design team cannot display a function using the prototype, or if prototyping a function will make too much unnecessary work for the design team, then it must be explained in a "note to dev". These are attached like small sticky notes onto the wireframe with an explanation of what action should be achieved with that function in the back-end development. In the following conversation, Wayne and Amanda discuss whether a function needs to be displayed in the prototype:

> **Amanda**: "How much of this from a dev[elopment] side do you need depicted?"
>
> **Wayne**: "We don't need to show you filtering the graph. We just need to know the data."

He goes on to explain further that the design team should add a "note to dev" to display all the "lines of the graph" if the user chooses not to have a "filter", as the note will be clearer than having all the lines displayed on the graph in the prototype: "If it's not illustrated, it must be explained or it's invisible". This comment is indicative of Wayne's attempts to bring greater visibility to all elements of the design process: greater control in the language of creating a classification system for a workflow (Bowker & Star, 2000). Being in a corporate environment, Henderson's (1998) assertion that the visual culture created by visual representations "constricts and constructs the literal ability to see or imagine" (p. 26) is also useful here. In making wireframes act as the main mode of communication between the design team and the development team, it makes the edges and limits of visual language discernible. In the example above, what should be displayed visually and what should be written out – or, in other words, how much to rely on tacit knowledge and what to make more explicit – is still being negotiated.

6.1.2. Embodiment

A necessary aspect of designing an app is testing out designs to see how designs hold up when a potential user interacts with them. I argue here that this is a form of embodiment, as designers enrol their own bodies or those of others to test designs. I am reminded again here of Vertesi's (2012) discussion about "seeing like a rover", in which scientists working with the Mars rover would attempt to embody the rover by contorting their bodies to 'see as' the rover would, as "scientific seeing requires not only eyes and instruments, but hands and bodies as well" (Vertesi, 2012, p. 396). Wireframes help designers to "see like" a user, drawing on designers' own senses for validation of app designs. Puig de la Bellacasa's (2011) vision-as-touch is also useful here for considering how designers affect the way in which users engage with the world and how designers attempt to care for the worlds that they touch.

For example, Jessica uses the Adobe XD app on her phone to see "how what [she is] designing actually looks like on a mobile". Similarly, Thulani plugs his designs into an app on his phone, as…

> …It helps me because that's what the user is going to be seeing…how the colours appear, because the actual colours on the screen are not necessarily how it would look on mobile. So it gives me that true colour feel. And also, I can…see if the button that I placed here and the size, if it's reachable for the user, if it's not too small, if it's not too big. And if …the user will be able to see the text, if it's readable.

Designers not only employ their eyes for sight here, but they also 'envision', as it were. When looking at the size of the button, what they are envisioning is if it is "reachable" – in other words, will the user holding a smartphone in maybe one hand be able to reach the button to tap it with their thumb? In this embodiment, designers use their own thumbs to 'see' if they can easily touch the button. This was something I noticed in my own experience of designing an app, as a button would appear to be a good size when on my laptop screen, but when I displayed it through the Adobe app on my phone, it was immediately clear to me that it was not big enough – it was not "tappable" with my finger. I would then go back and resize the button, now looking far too large on my laptop screen. Thus, to 'see like' the user requires that the designers use multiple aspects of their bodies in making effective designs.

Designers and developers who worked directly with users rather than with clients made use of user research techniques to "validate [their] solutions", as Thulani put it. User research has arisen in the past decade or so to refer to digital technologies' users as part of the more established practices of market research, and, in many ways, it is a discipline that is devoted to 'seeing like' their consumers or users. Market research first began in the 1920s by simply asking people on the street if they had seen an advertisement in magazines or heard it on the

radio or television in later years (Kierlanczyk, 2016). In America, the consumerist culture grew as a part of their post-World War II boom, and it was necessary that producers understand their consumers' needs and choices (Kierlanczyk, 2016). Quantitative surveys remained popular with researchers, but researchers also developed more qualitative methods like focus groups and understanding users' motivations for buying and using products. From the 1960s–1980s, market research evolved to value consumers' mindsets when using a product in their lives, and focus groups became more popular (Kierlanczyk, 2016). Market research in more recent years has evolved to make use of both qualitative and quantitative methods – often gathering data using qualitative methods, but ultimately boiling this data down to statistics. At Budgeting App and for the other participants who worked with market researchers, they used AB testing, surveys, and comparison testing. The user researcher at Budgeting App, Elize, explained to me that AB testing is when designers create two prototypes with "slight changes on the screens and components…[but] the flow is the same". One group of user testers will then get prototype A and another prototype B and, based on how workable the prototype is, the researchers and designers will choose which prototype to send to the development phase. Comparison testing simply shows users two designs and asks which is preferred, while surveys ask users about possible design paths a design team is considering taking. AB testing is the preferred method, as users are not biased from seeing two possible prototypes, and the data about where users 'tap' can be used moving forward in the designs. The designers and researchers thus favour embodied forms of generating knowledge, as it constructs the most accurate picture of how users will encounter an app. Elize further explained that once test users have encountered the prototypes, she will "synthesise the data. So…I take back the insights, I put it into a document, and I make some suggestions. So, where we need to change, improve." She continued to explain that the purpose of these techniques is to be "the voice for the customers" and that having the data from this research…

> …gives you…a little bit of power towards [the product team]... Because at the end of the day, they're the decision makers…I can't just say, I want this this way, because I like it. But if we say, '…80% of the users that we tested with preferred this way', then you've got some data to back up.

Designers thus enrol the test users' bodies – how they touch – to make decisions about which design should be deployed to users but abstract this embodied information into something that is much less fleshy and much less complex. Just like how Wilf (2016) explains that post-it notes are used in corporate contexts to display insights gained from consumer research, boiling these insights down to statistics can be understood as 'pseudodata'. The statistics appear to represent the market – or the users' preferences – but the statistics are also so abstracted that they are actually decoupled from it. Thus, while user testing is useful to designers to gain greater feedback on designs, it does also decontextualise the results so that they are more easily incorporated into business and profit-making decisions.

6.1.3. A Confluence of Forces

In considering design as a visual mode of transmitting meaning, the design team is both involved in creative processes that respond to and rework both global and local matters, and their designs emerge out of this confluence of forces (Ingold, 2013). Some of these forces are exploitative, as with the extraction of users' data for profit-making purposes in data colonialism (Couldry & Meijas, 2019). As Lindén and Lydahl (2021) argue for focusing on the darker sides of care, I will pay attention to some of the non-innocent acts of designing an app. In creating prototypes, for example, what designers are practically doing is mapping out threads of relation. However, in caring about making an app appealing, instructive, and 'seamless', the ways in which users' data is being extracted is obscured. The concept of 'seamfulness' advanced by Vertesi (2014) is useful here. She explains that sociotechnical systems and

infrastructures overlap in messy ways and that actors must work hard to stitch these systems together to appear co-present. Couldry and Meijas (2019) used this idea of 'seamfulness' to suggest a possible avenue through which to resist the projected 'seamlessness' of data colonialism, as explained in my Literature Review. When designers are concerned about making the user journey as easy and logical as possible, the labour and choices of designers often become invisible. Designers and developers creating separate journeys through an app show how seams of prototypes and code are intentionally stitched together and then smoothed out. As I practised designing an app, I thought about making a guest user journey, thus allowing users not to have to make a profile to access the app. As I started doing this, however, I realised that to avoid confusion across the rest of the design's prototyping, I would have to replicate the entire app for the guest user, adding prompts that were specific to only the guest user. Sure that I must be doing this incorrectly, I asked Bongani about how they programmed a guest user journey. He confirmed that adding a guest to an app means that developers must create a separate journey for that user. From talking with Jessica and from my own app designing efforts, I learnt that for designers, this meant "building out" a whole distinct set of screens, with prototyping and sets of prompts specific to the guest login. Developers have to set up in code what kind of user is logging in and what should happen for each type of login. Guest users then jump across this constructed seam of division between the two journeys when a login is created and loop into the dominant journey. However, if seams are stitched together closely enough and if the new login journey loads quickly enough, users will never be aware that they were being taken on a different journey. Erin told me that "seamlessness is the antidote to friction". If seamlessness is an antidote, why is friction a venom? Designers care about reducing friction, removing venom, and smoothing away seams so that users do not realise what information is possibly being extracted from them in the hitches between these invisible seams. Is there a potentially violent side to this seamlessness, even if unintentional?

Visually pleasing designs could be obscuring uglier networks of data colonialism and exploitation that feed into its construction. The access key for users to be able to jump from guest to user so seamlessly is that users input personal information. Couldry and Meijas (2019) describe such extraction of personal data as a kind of violence against users, as it violates personal rights to autonomy. As Erin further explained, seamlessness is about whether designers want users to notice friction and be prompted to provide more information, or to integrate an app into their practices because there was no friction at all. The process of designing and building an app is not an innocent one of putting visual elements together to look pleasing. It is also a visual mode of transmitting meaning to users (Boxenbaum et al., 2018) that obscures relations with which users engage and oftentimes hides the context of an app's construction. This obscuring is most evident in the lack of understanding surrounding the labour relations involved in building an app and the function of app design wireframes.

However, designers do not only contribute to the exploitative networks of data colonialism, for example, in a uniform manner without any creative choices. The design team borrows from other designs, incorporates likeable aspects, discards invaluable dimensions, and contributes to these networks in new and creative ways that are not predetermined but are still shaped by the tools and materials available to them. This became clear to me when thinking about a conversation between Amanda and the design team, where she told them that "[she] had to add cards to [her] ApplePay last night and they had quite a nice onboarding process". She intended for designers to use some of these features in the onboarding process, but altering it to fit the app-in-the-making as needed. The designers themselves would encourage clients to bring ideas from other apps to show to the designers. Luca explained that using inspiration from other sources is helpful:

> Some clients come to you and they just say, "Oh, can you just design me this logo? …These are my colours. Be creative. You do what you want." And you'll go and you'll do it, and then they'll tell you, "No, actually, I don't like that." So what I like to do is, I will tell the client to go online to Google, find some images, find some inspiration for me...And that gives me a platform to start off of, because once I start heading in a direction, then we'll go back to the client and ask them,...how does this look to you?… and I won't waste my time, if they are not liking that direction that I'm going in. But if they are, then at least I can get feedback from them and make changes and carry on in that same direction.

In addition to the co-production and co-creation of the creative object that this quote suggests, it also points to different understandings of creativity from outsiders and creatives. Wilf (2014) explains that, as a result of the historical origins of the creative process typically located in the Romantic era, the notion of art is that it comes from nowhere, like divine inspiration. This is reflected in Luca's clients' understandings of his creative process and telling him just to "be creative". In contradistinction to this romantic view of art and its origin, as Wilf (2014) goes on to argue, creative processes are typically emergent and interactional, and by examining creativity as a process, creativity is reframed as the reworking of a "given set of building blocks" (p. 398). Thus, as a part of their creative process, designers would draw on and remake the "building blocks" to suit their purposes.

Designers must pay attention to issues of labour, trends in app design, and building apps for a South African user. This last point I witnessed in the first meeting that App Company workers had with a new client, where he was explaining the purpose of his app, and Amanda and Anika worked with him to determine the app's parameters. The main point here was the client, Anika, and Amanda deciding what form the app should take – a web or mobile app. The issue with a

web app is that it requires an internet connection to function, and some of the app's users would be likely to live in more rural areas. The designers thus convinced the client against a web app, as these users might live in areas without stable internet connections because of a lack of infrastructure that supports it, and mobile data is very expensive in South Africa. The designers must thus correspond with the world around them in a process of co-production and co-becoming to build an app that takes into account users' locations. Thus, as Ingold (2013) argues, the "maker…[is] a participant in amongst a world of active materials" (p. 21), and where making is a confluence of forces and materials out of which the object emerges. These designers from the Global South are therefore active makers, not just passive emulators.

6.2. Visually Abstracting Labour

The way in which tasks are created, classified, and assigned reveals important relations of how boundaries of arranging and formulating labour are made within a company, and it is labour boundaries that show how app makers understood their work and the object under construction. Viewing these tasks on Jira also reveals the types of labour required to build an app and how these requirements are understood and delineated by the team members. However, it is the conversations and discussions, both formal and informal, that occur before and around a task's creation that most display how team members are working together to build an information infrastructure (Bowker & Star, 2000). Working together with Jira, different actors construct an information infrastructure for the company and, in so doing, make visible and invisible aspects of its creation. As Jira makes the life cycle of creating tasks and labouring on and with them one that relies on vision, I can examine here how this labour is abstracted visually to be more comprehensible in this corporate context (Wilf, 2016). As the employees must learn how to 'read' and 'feed' Jira, relations with Jira also involve creating a visual literacy and visual culture about what and how labour is included and displayed.

The creation of tasks is a process made up of many discussions and negotiations, displaying the "practical politics" (Bowker & Star, 2000) that are involved in developing a system of classification. "The task" will grow considerably, web-like, throughout its life cycle, as making the task is an ongoing process that continues until it returns to – or dies in – the backlog. Making the task can thus act as a point of gathering or confluence for viewing interactions between different actors in the network: by observing discussions around initially creating a task, the company's understandings of material dimensions like time and labour expectations are visible; when viewing the task as it appears on the task board, the final result of these initial negotiations is clear, together with the ongoing negotiations as the task is worked with. For the employees at App Company, tasks were something to be viewed and worked off of – they were essentially inscribed objects. However, I also work here to de-objectify the task and view it as something that is constantly in a process of co-becoming within a network of various actors, as maintaining and caring for the task requires labours that are separate from the labour that the task inscribes. Thus, I will simultaneously view Jira as a conscription device through which all negotiations about tasks must pass, when paused in a moment in time, because all the negotiations about the team's labour must go through Jira to be recorded properly: "If it's not in Jira, it doesn't exist". I will also consider the tasks as emerging and co-becoming in the network in which Jira is a focal actor.

Understanding Jira as conscription device means that, like a focal actor, it is an obligatory passage point through which all negotiations must be channelled, but it also means that anyone hoping to discuss the team's labour must do so through Jira. Additionally, the fact that Jira can be understood as a conscription device means that this labour is negotiated through visual representations; thus, I will consider what representing the employee's labour in this highly

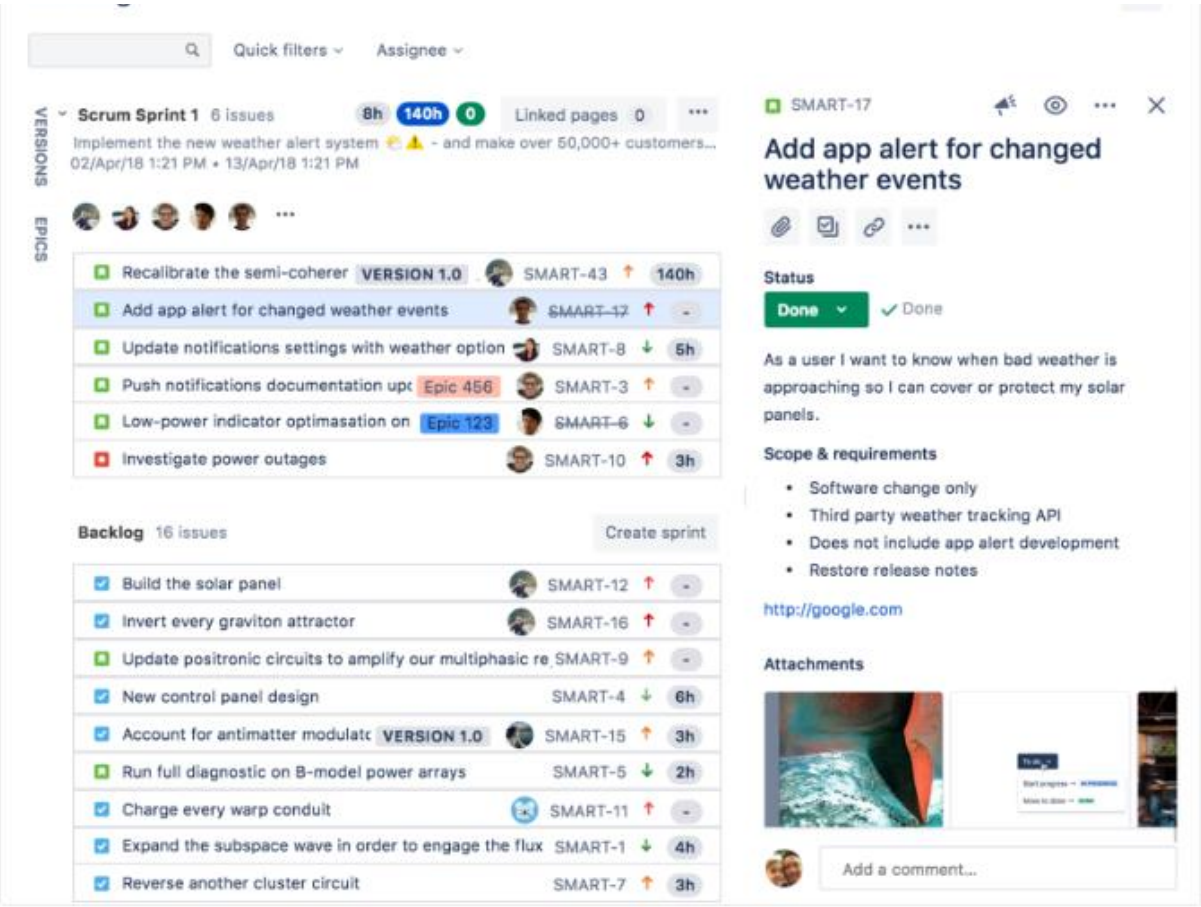
visual way does to how they labour. Rather than simply receiving verbal instructions from a boss about what to work on or the worker developing their own schedule in their head according to deadlines, Jira instructs the workers on what to do by showing them. The employee's labour becomes available for visible problem solving with many forms of knowledge from the different actors being held in a single task at one time – visibly, for all workers to see at all times.

The 'birthplace' of tasks was in the sprint planning meeting when, for the design team, Anika and Amanda created them in the backlog. Allocating tasks then meant that these tasks were taken from the backlog and assigned to team members and rendered visible on Jira's boards. Anika and Amanda ranked tasks in the "backlog" according to priority (i.e., the order in which tasks need to be completed), physically creating this ranking by moving "task tiles" up and

down in the backlog to create a visual representation of this ranking and prioritisation. As can be seen in Figure 11, Anika would label each task with an abbreviation of the project to which it related and a short description to identify it. In my interviews, a well-written description was felt to be important for avoiding confusion about what labour was required to be completed based on the task. As Tracey explained, this meant that there were two phases that must be

Figure 11

Elements of the Backlog



Note. From Jira Software Support (2022)

sufficiently addressed to ensure that team members completed the task according to the client's vision. First, the contract and scope needed to be clearly defined with the client so that the product owner had an accurate image of the client's requirements; if the product owner misunderstood the client's requirements, then the "task description" would incorrectly reflect these requirements. The product owner had to correctly translate and transform the client's objectives for their app into labour for the employees to perform. As Connor explained,

misunderstandings at this level might occur "because the client will say what they want, but they might not be explaining what they want versus what they have in their head…so the biggest problem is always that part of translating what they want into technical." As such, the process of the product owner creating a task and "task description" that can be understood by team members was an important part of avoiding conflict with the client at later stages. The second aspect of writing a "task description", as Tracey explained it, is "lack of communication…post brief" between the product owner and the designer or developer. The designer or developer must ensure that they understand "the task" as it is written and question the product owner if they are unsure of what is required. Tracey further emphasised the importance of "short feedback loops" at every stage of the process to correct misunderstandings at an early stage.

Properly defining or classifying the task is an important part of setting out the labour that the team must perform, but defining and writing this task is an ongoing process. As Tracey explained, to determine what the client wants in the design of the app, "I ask a shit-ton of questions. I stay curious. And I ask and I ask and I ask until we really unpack it properly". It is these questions that are used to develop a mental image of what the client is asking for, and the product owners then turn this mental image into visible task descriptions and requirements that appear on the Jira board. Thus, when a task is first created, Jira shows the outcome of these negotiations with the client but hides the initial labour that went into its creation. However, once the task is visible in Jira, it becomes open to contestation – it can be moved, altered, discussed over, and this is made possible by the fact that it is a graphical presentation that can be manipulated. Wilf's (2016) analysis of "the role played by graphic artifacts in organizational knowledge production" (p. 733), in his case post-it notes, is useful here for the idea of graphic mutability. Unlike Wilf's (2016) description of post-it notes, the aim is not to decontextualise

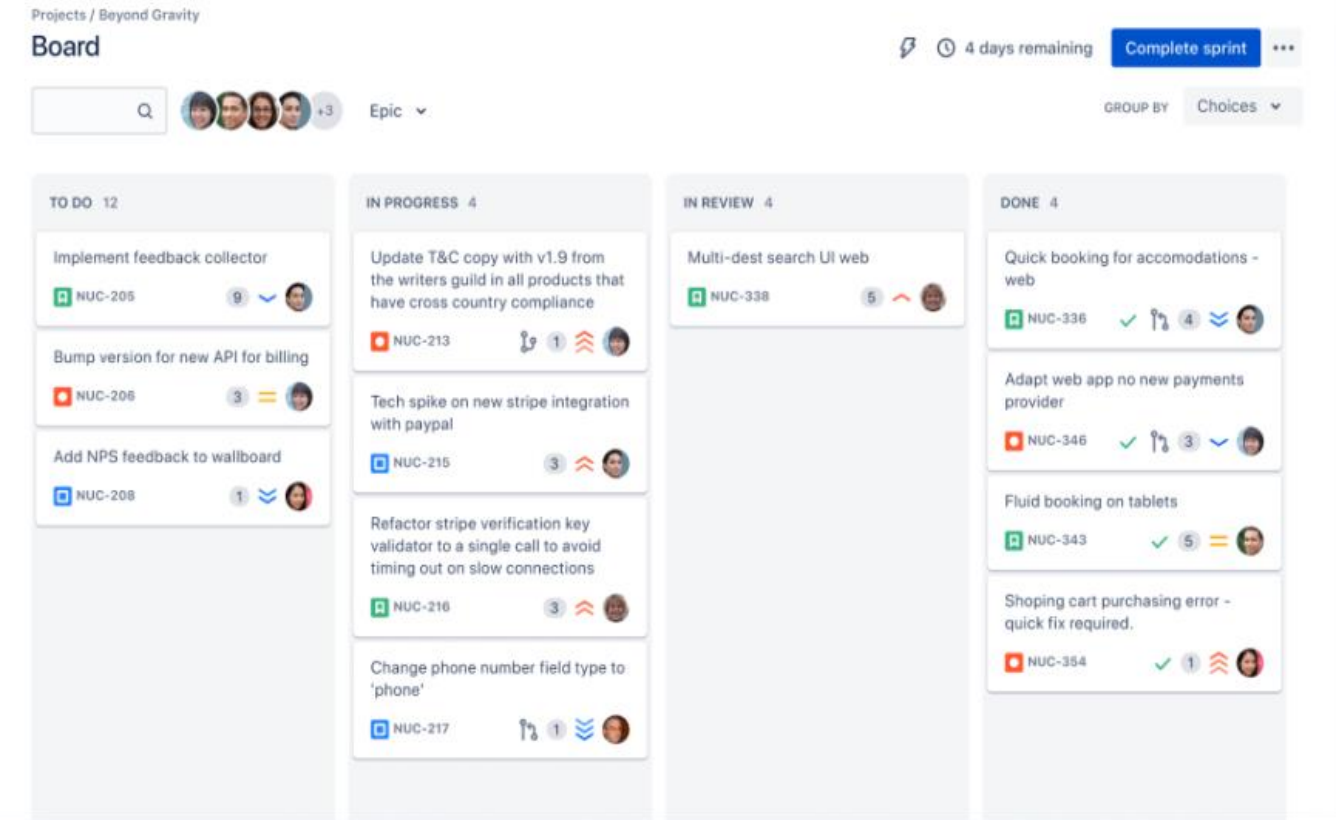
the information in Jira to be as few words as possible; there is an emphasis, as Tracey's explanations shows, on recording as much information as possible to reduce the possibilities of a misunderstanding about what is required from that task. While the tasks are possibly not as highly abstracted as the post-it notes, tasks must be abstracted to some degree to become visible, actionable labour – the requirements of the task are decontextualised from the contestations that went into its construction, as not all the conversations around creating the task appear in its presentation. However, viewing tasks moving through a life cycle in Jira shows how tasks are fed and maintained and how it is their moveability and plasticity that the team members constantly maintain.

At any one time, the Jira Scrum board, shown in Figure 12, holds many types of knowledge for negotiation, making it meta-indexical. Once tasks are made, they will appear on the board under the column “To do”, with a summary of information regarding the task on the task tile. The tile displays what the priority of the task is, as the priorities appear as coloured arrow markers on the task, making it quickly visible to team members what the work priorities are.

At a glance, team members could see the title of the task, what type of task it is, which Epic or

Figure 12

The Scrum Board



Note. From Atlassian (2022a)

project it is a part of, how many story points are associated with it, what its priority level is in the sprint, and who it is assigned to. By clicking on a task tile, Figure 11, the evolution of labour involved with that tile becomes visible, effectively recording the history of a task and creating an archive of sorts. The discussions around the labour required for that task would be visible, but only if the team members fed and maintained it. For example, once a team member

considered a task completed, it was marked for "QA review" and assigned to either Amanda or Wayne for the design team and development team, respectively. The team member had to then also write a "comment" that "tagged" the head of department that it was ready for "review"; Jira then automatically moved the task into the QA review column. After Amanda or Wayne reviewed the task, it could either be marked as "Done" if no further work was needed or assigned back to the team member under "In progress" if adjustments needed to be made; this could continue iteratively until the task was completed. There was also a column called "Blockers"; this is where tasks that were blocking the completion of other tasks were displayed. This process has evolved over time so that the "QA review" and "Blockers" columns can provide a more accurate representation of the workflow and where tasks are in the process. These columns could be considered more accurate because they depict more closely the associations that occur within the company between labourers: their labour is affected, as they become blocked by another person's labour or their labour is returned to them by others to be worked again. The Jira boards should, ideally, be a visual, moveable representation of all the work that was currently underway in that sprint, but to be an accurate representation, the team members must involve themselves in the labour of moving, tagging, and assigning – of associating with other team members. The team members put in the effort to do this labour of maintaining the tasks because it is expected of them by their heads of department, but the sense of organisation that it provides is also valued by employees. Jessica explained that in the current system of labour management with Jira…

> …it's a lot easier to keep track of where [the task] is in the process, whereas previously [before working with Jira], it would either look like it was done, or in progress, even if there was still work that needed to be done on it.

The workers at App Company thus reinforce the visual representation of moveable labour that Jira provides.

Another way in which the employees at App Company fed and maintained the tasks was by keeping the comments section alive. The comments section of a task most clearly displays the evolution of a task. As team members worked on the task, if they required further clarity on work to be done or a link to a document, for example, these discussions would appear in the comments section under the task. The one asking a question in this section had to tag the relevant team member in the request so that the person would be notified by email – App Company uses Gmail and the Google Suite – that a response is required on a question. Images for reference and links to wireframes would also be put in this section to help with labouring on that task. The comments section, therefore, acts as a space of communication where these short requests for information turn into a more formal record of interactions surrounding any specific task. Viewing the comments section of a task shows the messier labour and attentions required of a task, which does not appear when simply viewing the columns on the Jira board. Thus, as Bowker and Star (2000) argue, formal and informal systems of classification often run next to each other rather than being completely separate forms. This comment section showed who the task had moved between and when (as each comment was date- and time-stamped), could provide in-progress images of tasks since these were often attached to the task for review, and showed what clarification was required to complete it. Comments or requests from the client that were made in meetings or emails also often appear in the comments section so that they are tied to the task. The different forms of knowledge that were required to complete the task were therefore made explicit, and the comments acted as a storage place for the outcome of the negotiations that were made around wireframes, prototypes, and coding.

Having a graphical representation of the labour that accurately depicts the iterative process therefore enables team members to visualise work and steps to be followed, but at the same

time must work to maintain this visual representation. Jira, as a graphic artifact, allows teams to display and move information around as need be, as the labour is abstracted enough to be broken down into smaller, more mobile components. Workers' labour can then be deployed and managed more effectively because each labour component is visible at all times. However, when tasks were not properly fed or maintained, the tasks would not, for example, move through the life cycle of work, and thus would not always be totally representative of the labour that the teams do. This can be seen in Luca's statement that, "This isn't in the task, but I'm assuming they want us to do this." Outside of the work that is being visibly represented – abstracted – on the Jira boards, there is labour that employees do that is not always described in the task; their labour is more contingent and ad hoc than this graphic artifact might suggest. From my argument in the previous chapter, as Sachs (1995) argues, there is tacit work that occurs in addition to the discrete tasks as they appear in Jira. The tasks require that team members decipher and understand what labour needs to be performed – as can be seen in how Tracey puts such emphasis on accurate task descriptions – but tasks will also sometimes require ad hoc work that was not expected by the original requirements.

Thus far, I have argued that 'vision' and 'the visual' are useful for analysing how designers communicate with users and with the other actors in the network and that how employees and managers abstract workers' labour in Jira makes labour seem highly adaptable. In the conclusion, I will demonstrate the arguments of each chapter and how the chapter fleshed out a few inter-related aspects of an app's world in the corporate environment. I will also reflect on the theoretical and methodological contributions of this dissertation to the literature on the co-production of humans and technology.

7. Conclusion

In this chapter, I will provide a summary of my main findings and recommendations for further questions that could arise from this research. Thus far, I have argued that it is out of encounters between both human and non-human actors and entities that an app emerges. These encounters are multiple and varied, and I have focused only on some, guided by the questions I sought to answer. I started my proposal by asking myself the question, 'If nothing comes without its world, what world comes with an app'? I sought to 'implode' an app's world by focusing on processes of production to challenge the fetishization of the 'final' product of the app, which often severs this product from its world and makes an app seem as though it is floating untethered all on its own. By contrast, my data shows that an app's world is bursting at the seams, and I have made these seams visible as a part of worlding the app.

By drawing together some feminist and more-than-human approaches to studying this design process and the technology-in-the-making, I have considered aspects that matter this world and that matter in this world to provide a situated account of what it means to design and make an app in a corporate context in the Global South, situated as I am in South Africa. I have drawn on an approach that considers how to provide a critical account of technology that exposes the foundations of its constructions (Spiegel, 2005), while avoiding only providing an unmasking that does not go further with its analysis (Latour, 2004). Thus, I drew on Puig de la Bellacasa (2012; 2017) Haraway (1988), and Lindén and Lydahl (2021) to consider "what is enabled for us as the researchers to say and do by (attending to) those exclusions and/or that violence" (Lindén & Lydahl, 2021, p. 4). Such considerations are the result of paying attention to and exposing the (unequal) relations of technology production. I argue that maintaining this constant double vision that plays with exposé anthropology and a thinking with care approach is appropriate to studying technology production. Such an approach avoids falling into the trap

of viewing technology as either wholly a saviour – as with the reports that 4th industrial revolution technologies are driving productivity and growth across manufacturing and production (Luchtenberg, 2022) – or wholly exploitative – as with Couldry and Meijas' (2019) fairly dystopian depiction of Data Colonialism. Instead, I have provided an account of producing an app that provides "partial connections" (Haraway, 1988) – in which I constructed the relations of an app's world, location them within their specific business and labour context. I have also drawn on Miller and Woodward's (2007) concept of the 'blindingly obvious' to simultaneously depict macro-sociological trends in labouring in technological corporate contexts and the local manifestations of these trends in a single company.

7.1. Summary of Findings

In Chapter 2, 'Ethnographic Context', I situated my research within the company where I conducted the majority of my field research. I provided an account of the socioeconomic climate in which App Company is located and the general layout of the office. I also described my meeting with the different team members and described each team's role within the company. The aim of this chapter was to introduce the context and main characters I write about in this dissertation and to describe the context in which these apps are being produced. This was essential to argue that, to matter the world of the app, I must pay attention to the social and physical architectures (Seaver, 2018) on which the app is built. It is these relations and relationships that thread through and behind the app and hold it together as a seemingly coherent entity, but which I have revealed throughout this dissertation as relations that are contingent and constructed with care.

Chapter 3 reviewed the literature that shaped my thinking in formulating the topic for this dissertation and that I drew on and responded to while writing this dissertation. I began by

providing an account of data colonialism (Couldry & Meijas, 2019) as a macro-sociological concept that is an introduction into the exploitation that can exist in technologies and technology corporations in a capitalist and colonial configuration. I referenced exposé anthropology (Spiegel, 2005) to consider how I could respond to these modern-day examples of exploitation and power relations by drawing on critical reflection. I moved beyond the approach of exposé anthropology to incorporate feminist methodologies like thinking with care, matters and concerns of care (Puig de la Bellacasa, 2017), and situated knowledges (Haraway, 1988) to world the app in a way that was not only critical but also sought the complexity of an app's connections. I then represented how I would use the literature on classification, boundaries, and infrastructure (Bowker & Star, 2000) to understand the creation of a workflow within a company, ending the section with a description of translation and transformation in the network (Silva, 2019) and how I would use this STS concept to understand the relations between the different actors in the network. I followed this with a description of how I would approach the design work of the designers by understanding their work as emergent (Ingold, 2013) and creative (Wilf, 2014) and that draws on modes of visual communication (Boxenbaum et al., 2018; Henderson, 1998). Lastly, I depicted how I would use the literature on materiality (Pink et al., 2016) and material space (Gibbard, 2016) to world the matter of an office environment and ground the app in these everyday interactions and negotiations in material space.

In Chapter 4, I introduced Jira to the reader as a non-human actor in the design process or network – a companion entity (Haraway, 2003) to the workers at App Company, putting them in relations of mutual emergence (Buchanan, 2018). While I began this dissertation by focusing on the humans in this process of app creation, I found during my fieldwork and as I wrote up my data that this world of the app is made more complex by other technological assemblages

that feed into an app's creation in corporate contexts. Drawing on more-than-human anthropology (Locke & Muenster, 2018) made me attentive to the affect that non-humans also have in the processes of making an app. Understanding Jira as a companion allowed me to see Jira not only as software that the workers use to store their labour but also revealed the work that Jira does in mediating and shaping the labour at App Company as a bossy manager.

I also considered how, as a companion, the workers at App Company had to learn how to live with and become literate in Jira, labours that are not depicted explicitly as part of their 'job description' but which nonetheless are required for them to be successful actors in the network. Jira then also becomes a focal actor through which many negotiations about labour must go (Silva, 2019), which further affects how App Company structures their labour, the way employees speak about their labour, and their interactions with other, human actors. Because Jira becomes a focal actor, it is also Jirian language that must be used for the actors in the network to communicate, and this can cause both stronger and weaker relations in the network. Designers and developers are given a common vocabulary and time infrastructure that helps to bridge the divide between the different work that they do – the varying concerns they embody creating stronger ties in the network – but clients who are not yet familiar with Jira's language might contest Jira's mediation as a focal actor. The other effect of this common language is that relations in the Global South are affected by this language, which has been created by Jira's developers in the Global North. In this way, more relations are created between actors in the Global North and South. The structure of work in this company in the Global South is further affected by the Global North, as App Company draws on agile methodologies to configure their labour. Agile methodologies structure how labour is organised in terms of time, remuneration, and team collaboration (Dove, 2002). Through Jira, these agile methodologies are further enforced by drawing actors into structuring their labour according to agile

methodologies, using agile terminologies and framings of time on Jira's platform. I also argued here that it is the highly adaptable and flexible nature of labour that these methodologies create and enact that makes them well-suited to app production.

In Chapter 5, I drew on Pink et al. (2016) for the idea that 'the digital' and 'the material' are not separated, but rather entangled aspects of the same domain. In Chapter 4, I introduced Jira as a companion that affects how App Company labours, and in Chapter 5, I built on this to argue that Jira has changed the organisation of App Company by providing them with a common time infrastructure and vocabulary and by facilitating an abstraction of labour that allows managers and heads of department to cut up, label, and assign labour in seemingly infinite malleable units of time. Jira shows that software is not just an add-on to hardware, as Jira comes to dominate the relations of App Company – software is a 'hard' infrastructure. Once again, I did not enter my fieldwork expecting to find any manager other than a human one, or a structuring of labour that made such attempts at rationalisation. I was therefore greatly interested in App Company's efforts to rationalise (Guru, n.d.) their labour using Jira and in the affects that this rationalisation had on how they spoke about and ordered their labour. Jira helped the App Company owner and managers to calculate the value of employees' labour to, seemingly, receive more accurate remuneration from clients and thus produce greater profits. However, I also argued that while App Company presented its labour structure as rational and completely calculable, its labour was more contingent; Jira could not encompass all of their labour and could not 'see' all of it (Malaby, 2009). The App Company workers perform labours outside of those depicted by Jira, such as in evaluating and abstracting their own labour and maintaining personal relationships in the office. I further argued that Jira can function only in conjunction with the social, somatic relationships that matter the office space, a continuation of the main argument of this dissertation that technology and humans co-produce each other. I

argued that while the narratives of technology present technology as severed from the everyday relations of their world, mystified, and coherent technology (Couldry & Meijas, 2019), the app is mattered by negotiations of labour, infrastructure, and space. My depiction of the production process of apps also aims to challenge a fetishization of the app to lift the veil on how technologies are constructed. It is not only Jira that shapes an app-in-the-making, but also the material aspects of the office: organisations of space, electricity, and doors. As Seaver (2018) argues, these material configurations shape the configuration of the technology they build. I showed that these material constructions evidence how the world of the app is messy and that, in employees' organising their office space, their different concerns are revealed and reflected in the production of the app.

In Chapter 6, I considered how 'vision' and 'the visual' can be used to understand the work that designers do and how they depict and conceive of this work. In continuing to matter the world of an app, I argue that Puig de Bellacasa's (2017) concept of 'touching visions' prompted me to consider how designers' design work touches users and threads them into the app's world. Such a viewpoint is especially useful with Lindén and Lydahl's (2021) darker side of care, as one of the functions of the designers is to remove friction – the venom of app design. As the designers work with intentionality to remove friction, they affect the worlds of an app and the users who use an app. As I showed in the previous chapters, constructing technology is grounded in everyday concerns of labour, negotiations with other actors, and possibly exploitative relations from these negotiations. Similarly, the visual mode of communication (Boxenbaum et al., 2018) that an app sets up with users is subject to possibly exploitative relations that obscure how users' data is being extracted and sold to third parties (Couldry & Meijas, 2019), for instance. I therefore interrogated the means by which designers convey meaning to users through their designs because, in constructing a visual culture with users, the

designers are restricting how we see and what there is to see (Henderson, 1998) – making some visuals matter and marginalising others.

Similar to my commitment in Chapter 5 to retie an app's world to the relations of its production, I argued that the designers' design work draws on and contributes both global and local matters – their designs emerge out of a correspondence with the world around them (Ingold, 2013), rather than from nowhere as if by creative magic (Wilf, 2014). It is not the designers alone who construct the app's designs, but they work with other actors in processes of negotiation and problem-solving to make the app. Thus, wireframes and prototypes become a visual means of communication – conscription devices (Henderson, 1998) – that are vital means for mediating relations between the actors in the network and their varying concerns. Wireframes also allow designers to 'see as' (Vertesi, 2012) Users, as they engage with their designs with their bodies, and thus the designers further touch and affect the world of the user by embodying it. I also argued that Jira, in addition to a focal actor, becomes a conscription device because of the visual means for engaging with labour that Jira provides. Because of this visual representation, designers and developers encountered their labour visually in Jira. Their labour thus becomes visible in a way that makes further abstractions (Wilf, 2016) and contestations of their labour possible.

The two main threads of this dissertation are that technology and humans are involved in encounters of co-production and that, to matter the world of an app, I focus on the mess, rich complexities, and threads of relation in addition to maintaining a critical approach that is attentive to the exploitative relations that often underlie the construction of technologies. By bringing these two threads together, I have argued that an app's world is made up of but cannot be reduced to some human actors in the network with varying concerns – e.g., designers,

developers, project managers, clients, and users – that are also affected by non-human actors like Jira. These relations are grounded in everyday negotiations and contestations surrounding labour, infrastructure, space, an interplay of local and global forces, and constructing visual culture. I will end this summary of my findings by reasserting that an app's world is messy – it is not seamless, and it is as affected by the relations of an interconnected world as any other product.

7.2. My Contributions and Possible Further Research

The first contribution of this dissertation is an analysis of the production process involved in building an app. While there are many studies that analyse the effect of apps on users and on how means of engagement are constructed around these apps – or their consumption (Kusimba, 2021) – studies that analyse the production process are not as common. Thus, I am contributing to the growing literature that focuses on the production side of technology and how this production is an imperfect and contingent process that does not reflect the narratives that these technologies create about themselves. Examples of this are artificial intelligence (AI), as Le Ludec and Cornet (2022) explained in their conference paper that it is human choices and workers that are vital in the work processes surrounding AI. In the modelling process of classification, it is the workers who classify the data and thus there is no automation – "only cognitive labour processes relying on all different levels of labour". What these studies contribute is an understanding that technology is still not divorced (at all) from the grounded relations of labour and human decision-making. My additional contribution to this literature is the more-than-human approach, which positions Jira as an actor in the process that the human actors encounter rather than only interact with. As project management is increasingly digitalised and put into software like Jira, studies that view these software as actors will provide

scholars with a means to engage with the software that does not negate the very real effects that they have on the labour and lives of employees – to view them as much more hardy.

My other contribution is the methodological approach I developed. By taking this double vision, situated, and 'blindingly obvious' approach to technology, I believe I have provided a means through which scholars in the Global South might engage with more established traditions of examining technology in the Global North. For South African anthropologists, Nyamnjoh (2015) states that "Anthropology's scientific status should derive from its methodical and systematic observation of socially constructed worlds in which anthropologists are as co-implicated as those they study, doubling as observer and observed" (p. 60). This echoes my approach in its commitment to a situated examination of how worlds are constructed so that the relations on which they are built can be revealed. However, in this examination, Nyamnjoh further suggests that the anthropologist must not distance themselves from their participants but, rather, as Puig de la Bellacasa (2012) argues, thinking in the world means acknowledging our own perpetuation of dominant values. Moving forward, studies of technology in the Global South could adopt such a methodology that is methodical and critical but that also provides space for reflection on the relations that make up that world. Additionally, the interplay of exposé anthropology (Spiegel, 2005) and matters of care and concern (Puig de la Bellacasa, 2017) offer a way to engage with technologies that are often so characterised by poles of great exploitation (Azmi, 2021) and narratives of possibility and futurism (Klein, 2020). Rather than approach these as poles that cannot be investigated together, I argue that the interplay of these two approaches brings the poles together in a kind of warping that turns them into two sides of the same coin; technology can simultaneously be broken down into its dominating components and built up by cultivating what is part of this world and what gathers around it.

Following the execution of this research, new questions have arisen for me that I was not able to explore in this dissertation due to the time and scope restraints of a master's project, which might be answerable in future research: How do designers', developers', and project managers' relationships with Jira differ? Is Jira 'hardy' enough to remain an actor in an industry that changes so constantly? What aspects of an app's world are still balled up that I might tease out, curious and critical about where the threads begin, where they knot, and what actors have wound and continue to wind them together?

8. Reference List

1922–2019: The rise and fall of Eskom. (2019). news24. https://www.news24.com/fin24/1922-2019-the-rise-and-fall-of-eskom-20190213

Abrahamsson, S., & Bertoni, F. (2014). Compost politics: Experimenting with togetherness in vermicomposting. *Environmental Humanities*, *4*, 125–148.

American Anthropological Association. (2021). *AAA statement on ethics*. https://www.americananthro.org/LearnAndTeach/Content.aspx?ItemNumber=22869

Anthropology Southern Africa. (n.d.). *Ethical guidelines and principles of conduct for anthropologists*. https://www.asnahome.org/about-the-asna/ethical-guidelines

Ashworth, B. (2022). *Amazon watches its workers and waits for them to fail*. Featured. https://www.wired.com/story/amazon-worker-tracking-details-revealed/

Asmo. (2019). *Waterfall project management: An overview*. Zenkit Blog. https://zenkit.com/en/blog/waterfall-project-management-an-overview/

Atlassian. (2022a). *Jira software*. https://www.atlassian.com/software/jira/features/scrum-boards

Atlassian. (2022b). *Our solutions*. https://www.atlassian.com/

Azmi, F. T. (2021). *The little hands of labour behind your smartphone*. The Wire. https://thewire.in/rights/child-labour-unicef-mines-amnesty-international-ilo

Baur, E. J. (1949). The functions of ceremony in the advertising business. *Social Forces*, *27*(4), 358–365. https://doi.org/10.2307/2572488

Bear, L., Ho, K., Tsing, A., & Yanagisako, S. (2015). *Gens: A feminist manifesto for the study of capitalism*. Society for Cultural Anthropology. https://culanth.org/fieldsights/gens-a-feminist-manifesto-for-the-study-of-capitalism

Beck, K., Beedle, M., Bennekum, A. v., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C.,

Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for agile software development*. http://agilemanifesto.org/

Becker, H. (2007). De-provincialising South African anthropology: On relevance, public culture & the citizen anthropologist. *Anthropology Southern Africa*, *30*(3–4), 85–96. https://doi.org/10.1080/23323256.2007.11499951

Bernard, H. R. (2011). *Research methods in anthropology: Qualitative and quantitative approaches* (Fifth ed.). Rowman Altamira.

Bonefeld, W. (2018). Abstract labour and labouring. *Consecutio Rerum*, *3*(5), 205–223. https://doi.org/http://www.consecutio.org/2018/11/abstract-labour-and-labouring/

Bowker, G., & Star, S. (2000). *Sorting things out: Classification and its consequences*. The MIT Press.

Bowman, D. (2009). Studying up, down, sideways and through: Situated research and policy networks. *The Future of Sociology: Proceedings of the Annual Conference of the Australian Sociological Association*, 1–11.

Boxenbaum, E., Jones, C., Meyer, R. E., & Svejenova, S. (2018). Towards an articulation of the material and visual turn in organization studies. *Organization Studies*, *39*(5–6), 597–616. https://doi.org/10.1177/0170840618772611

Browning, K. (2020). *Instagram tries clamping down on misinformation*. The New York Times. https://www.nytimes.com/2020/10/30/technology/instagram-tries-clamping-down-on-misinformation.html

Buchanan, I. (2018). *Companion species*. Oxford University Press. https://www.oxfordreference.com/view/10.1093/acref/9780198794790.001.0001/acref-9780198794790-e-765

Cairns, K. (2013). Ethnographic locations: The geographies of feminist post-structural ethnography. *Ethnography and Education*, *8*(3), 323–337.

Callon, M. (1984). Some elements of a sociology of translation: Domestication of the scallops and the fishermen of St Brieuc Bay. *The Sociological Review*, *32*(1_suppl), 196–233. https://doi.org/10.1111/j.1467-954X.1984.tb00113.x

Campbell, L. (2019). Automate sub-task creation in Jira with elements copy & sync. *valiantys*. https://valiantys.com/en/blog/atlassian-administration/automate-sub-task-creation-in-jira-with-elements-copy-sync/

ClickUp. (2022). *One app to replace them all*. https://clickup.com/?utm_source=google&utm_medium=cpc&utm_campaign=gs_cpc_t4_nnc_brand_trial_all-devices_cpc_lp_x_all-departments_alpha_kd-brand-t4-20220710&utm_content=all-countries_kw-target_text_all-industries_all-features_all-use-cases_clickup_exact&utm_term=e_clickup&gclid=Cj0KCQiA4aacBhCUARIsAI55maF0v0Ca-0ixGybuJouS-qAwKZsFXM_9iBylevzNSd9c7Iokuow7y6YaAsfnEALw_wcB

Cohen, D., & Crabtree, B. (2006). *Qualitative research guidelines project*. http://www.qualres.org/HomeSemi-3629.html

Comaroff, J., & Comaroff, J. L. (2012). Theory from the South: Or, how Euro-America is evolving toward Africa. *Anthropological Forum*, *22*(2), 113–131. https://doi.org/10.1080/00664677.2012.694169

Companies and Intellectual Property Commission. (2018). *CIPC entity search*. https://eservices.cipc.co.za/Search.aspx

Couldry, N., & Meijas, U. A. (2019). *The costs of connection: How data is colonizing human life and appropriating it for capitalism*. Stanford University Press.

Desmond, M. (2014). Relational ethnography. *Theory and Society*, *43*(5), 547–579.

DevToolsGuy. (2015). The history of wireframing & prototyping. https://www.infragistics.com/community/blogs/b/devtoolsguy/posts/the-history-of-wireframing-amp-prototyping

Dove, R. (2002). *Response ability: The language, structure, and culture of the agile enterprise* (1st ed.). John Wiley & Sons, Incorporated. http://ebookcentral.proquest.com/lib/qut/detail.action?docID=119597

Dumit, J. (2014). Writing the implosion: Teaching the world one thing at a time. *Cultural Anthropology*, *29*(2), 344–362. https://doi.org/https://doi.org/10.14506/ca29.2.09

Dumont, S. (2021). *The difference between UX and UI, according to designer and Maze CEO, Jonathan Widawski*. https://maze.co/blog/ui-vs-ux/#:~:text=What's%20the%20difference%20between%20UX,when%20interacting%20with%20a%20product.

Encylopedia of Marxism. (2018). *Necessary & surplus labour time*. https://www.marxists.org/glossary/terms/n/e.htm

Fox, N. J. (n.d.). *New materialism*. Global Social Theory. https://globalsocialtheory.org/topics/new-materialism/

George A. Spiva Library. (2022). *Fairy tales*. Missouri Southern State University. https://libguides.mssu.edu/c.php?g=185298&p=1223898

Gibbard, M. (2016). *The anthropology of work: Habits, trends and behaviors in the workplace*. https://www.linkedin.com/pulse/anthropology-work-habits-trends-behaviors-workplace-max-gibbard/

Gittleman, J. L. (2022). *Species*. Encyclopedia Britannica. https://www.britannica.com/science/species-taxon

Goel, D. (2021). The global digital divide is reminiscent of colonialism. https://blogs.lse.ac.uk/brexit/2021/05/06/the-global-digital-divide-is-reminiscent-of-colonialism/

Gordon, R. J., & Spiegel, A. D. (1993). Southern Africa revisited. *Annual Review of Anthropology*, *22*(1993), 83–105.

Govia, L. (2020). Coproduction, ethics and artificial intelligence: A perspective from cultural anthropology. *Journal of Digital Social Research*, *2*(3), 42–64.

Guru, S. (n.d.). *Theory of rational-bureaucratic society: Weber's classical theory of modernity*. https://www.yourarticlelibrary.com/sociology/theory-of-rational-bureaucratic-society-webers-classical-theory-of-modernity/39849

Haraway, D. (1988). Situated knowledges: The science question in feminism and the privilege of partial perspective. *Feminist studies*, *14*(3), 575–599.

Haraway, D. (2003). *The companion species manifesto: Dogs, people, and significant otherness*. Prickly Paradigm Press.

Haraway, D. (2006). A cyborg manifesto: Science, technology, and socialist-feminism in the late 20th century. In *The international handbook of virtual learning environments* (pp. 117–158). Springer.

Henderson, K. (1998). *On line and on paper: Visual representations, visual culture, and computer graphics in design engineering*. MIT Press.

Hoory, L., & Bottorff, C. (2022). *Agile vs. Waterfall: Which project management methodology is best for you?* Forbes Advisor. https://www.forbes.com/advisor/business/agile-vs-waterfall-methodology/#:~:text=The%20main%20difference%20is%20that,different%20phases%20of%20the%20project.

Hynes, N. (2009). Colour and meaning in corporate logos: An empirical study. *Brand Management*, *16*(8), 545–555.

Ingold, T. (2013). *Making: Anthropology, archaeology, art and architecture*. Routledge.

Jira Software Support. (2022). *Use your scrum backlog*. https://support.atlassian.com/jira-software-cloud/docs/use-your-scrum-backlog/

Jones, A., Thoma, V., & Newell, G. (2016). Collaboration constraints for designers and developers in an agile environment. *Proceedings of the 30th International BCS Human Computer Interaction Conference: Fusion!*(37), 1–9.

JUSTIA Patents. (2022). *Patents assigned to Atlassian pty ltd.* https://patents.justia.com/assignee/atlassian-pty-ltd

Kearney, R. (2002). *On stories*. Routledge. https://doi.org/10.4324/9780203453483

Kenton, W. (2021). *Financialization*. Investopedia. https://www.investopedia.com/terms/f/financialization.asp

Kierlanczyk, K. (2016, August 2). A brief history of market research. https://www.keltonglobal.com/perspectives/a-brief-history-of-market-research/#:~:text=The%20first%20true%20instances%20of,order%20to%20be%20considered%20effective.

Klein, N. (2020). *Naomi klein: How big tech plans to profit from the pandemic*. https://www.theguardian.com/news/2020/may/13/naomi-klein-how-big-tech-plans-to-profit-from-coronavirus-pandemic

knowledge hut. (2022). *Scrum history*. https://www.knowledgehut.com/tutorials/scrum-tutorial/scrum-history

Kusimba, S. (2021). *Reimagining money: Kenya in the digital finance revolution* (F. Wherry & J. C. Lena, Eds.). Stanford University Press.

Latour, B. (2004). Why has critique run out of steam? From matters of fact to matters of concern. *Critical Inquiry*, *30*(2), 225–248.

Latour, B. (2018). *An inquiry into modes of existence: An anthropology of the moderns* (C. Porter, Trans.). Harvard University Press.

Latour, B., & Woolgar, S. (1986). *Laboratory life: The social construction of scientific facts*. Princeton University Press.

Le Ludec, C., & Cornet, M. (2022, July 8). *Who is (really) producing artificial intelligence (ai)? Examining labour processes behind french commercial AI systems.* EASST 2022 – Politics of Technoscientific Futures, Madrid.

Lindén, L., & Lydahl, D. (2021). Editorial: Care in STS. *Nordic Journal of Science and Technology Studies*, *9*(1), 3–12.

Locke, P., & Muenster, U. (2018). *Multispecies ethnography*. Oxford Bibliographies. https://www.oxfordbibliographies.com/view/document/obo-9780199766567/obo-9780199766567-0130.xml

Luchtenberg, D. (2022). *The fourth industrial revolution will be people powered*. McKinsey & Company. https://www.mckinsey.com/capabilities/operations/our-insights/the-fourth-industrial-revolution-will-be-people-powered

LumenLearning. (2020). *Unstructured interview*. https://socialsci.libretexts.org/Bookshelves/Anthropology/Cultural_Anthropology/Book%3A_Cultural_Anthropology_(Evans)/04%3A_Methods/4.06%3A_Unstructured_Interview

Lupton, D. (2016). Digital companion species and eating data: Implications for theorising digital data–human assemblages. *Big Data & Society*, *3*(1), 205395171561994. https://doi.org/10.1177/2053951715619947

Madhok, S. (2020). A critical reflexive politics of location, 'feminist debt' and thinking from the Global South. *European Journal of Women's Studies*, *27*(4), 394–412.

Malaby, T. M. (2009). *Making virtual worlds: Linden lab and second life*. Cornell University Press. https://doi.org/10.7591/9780801458996

Marín, O. (2018). *Was Arthur C. Clarke right? An exploration of the magical nature of technology*. https://lab.cccb.org/en/arthur-c-clarke-any-sufficiently-advanced-technology-is-indistinguishable-from-magic/#:~:text=In%201962%2C%20in%20his%20book,technology%20is%20indistinguishable%20from%20magic%E2%80%9D.

Marshall, P. A. (1992). Research ethics in applied anthropology. *IRB: Ethics and Human Research*, *14*(6), 1–5. https://doi.org/10.2307/3563851

Marx, K. (1887). The fetishism of commodities and the secret thereof (S. Moore & E. Aveling, Trans.). In F. Engels (Ed.), *Capital: Volume 1*. Progress Publishers. (1867)

McKeegan, C. (2022). *I've managed a remote team for 10+ years. Here's why we don't use productivity monitoring software*. Inc. Africa. https://incafrica.com/library/carrie-mckeegan-ive-managed-a-remote-team-for-10-years-heres-why-we-dont-use-productivity-monitoring-software

Merriam-Webster. (2022). *Ceremony*. https://www.merriam-webster.com/dictionary/ceremony

Miller, C. (2017). *Design + anthropology: Converging pathways in anthropology and design*. Routledge.

Miller, D., Rabho, L. A., Awondo, P., de Vries, M., Duque, M., Garvey, P., Haapio-Kirk, L., Hawkins, C., Otaegui, A., Walton, S., & Wang, X. (2021). From apps to everyday life. In *The global smartphone* (pp. 79–102). UCL Press. https://doi.org/10.2307/j.ctv1b0fvh1.12

Miller, D., & Woodward, S. (2007). Manifesto for a study of denim*. *Social Anthropology*, *15*(3), 335–351. https://doi.org/10.1111/j.0964-0282.2007.00024.x

Ndebvu, H. (2021). *Data relief for south african students*. UbuntuNet Alliance. https://ubuntunet.net/2021/04/data-relief-for-south-african-students/

Newbi1. (2022). *Chainsaw wireframe model vector*. dreamstime. https://www.dreamstime.com/chainsaw-wireframe-model-vector-chainsaw-model-wireframe-low-poly-mesh-vector-illustration-image120150656

Nextup.ai. (2020). *History of atlassian – an unusual tech titan*. https://medium.com/@nextupai/history-of-atlassian-an-unusual-tech-titan-fdf3fcecae4e

Nyamnjoh, F. B. (2012). Blinded by sight: Divining the future of anthropology in Africa. *Africa Spectrum*, *47*(2–3), 63–92. https://doi.org/10.1177/000203971204702-304

Nyamnjoh, F. B. (2015). Beyond an evangelising public anthropology: Science, theory and commitment. *Journal of Contemporary African Studies*, *33*(1), 48–63. https://doi.org/http://dx.doi.org/10.1080/02589001.2015.1021214

Olukoshi, A. O. (2015). Structural adjustment programs. In D. Eyoh & P. T. Zeleza (Eds.), *Encyclopedia of Twentieth-Century African History* (pp. 533–536): Routledge.

Penn LPS. (2019). *What is organizational anthropology?* https://lpsonline.sas.upenn.edu/features/what-organizational-anthropology

Philip, K. S. (2015). Indigenous knowledge: Science and technology studies. In J. D. Wright (Ed.), *International Encyclopedia of the Social & Behavioral Sciences* (2nd ed.): Elsevier.

Pink, S., Ardèvol, E., & Lanzeni, D. (2016). *Digital materialities: Design and anthropology* (S. Pink, E. Ardèvol, & D. Lanzeni, Eds.). Bloomsburg Academic.

priorityprototypes. (2021). *Engineering prototype*. https://prototypes.prioritydesigns.com/complete-prototypes/engineering-prototype/

Privacy Policies. (2021). *South Africa's POPI act*. https://www.privacypolicies.com/blog/popi-act/

Product Plan. (2022a). *Jira*. https://www.productplan.com/glossary/jira/#:~:text=Atlassian%20introduced%20Jira%20in%202002,tracking%20software%20tool%20called%20Bugzilla.

Product Plan. (2022b). *Wireframe*. https://www.productplan.com/glossary/wireframe/

professional development. (2022). *Who uses scrum?* https://www.professionaldevelopment.ie/who-uses-scrum

Puig de la Bellacasa, M. (2011). Matters of care in technoscience: Assembling neglected things. *Social Studies of Science*, *41*(1), 85–106. https://doi.org/10.1177/0306312710380301

Puig de la Bellacasa, M. (2012). 'Nothing comes without its world': Thinking with care. *The Sociological Review*, *60*(2), 197–216. https://doi.org/10.1111/j.1467-954X.2012.02070.x

Puig de la Bellacasa, M. (2017). *Matters of care : Speculative ethics in more than human worlds*. University of Minnesota Press. http://ebookcentral.proquest.com/lib/pretoria-ebooks/detail.action?docID=4745533

Radigan, D. (2022). *Story points and estimation*. Atlassian Agile Coach. https://www.atlassian.com/agile/project-management/estimation

Rationalization. (2022). *Oxford reference*. https://www.oxfordreference.com/view/10.1093/oi/authority.20110803100405399

Rehkopf, M. (2022). *Stories, epics, and initiatives*. https://www.atlassian.com/agile/project-management/epics-stories-themes

Roussilhe, G. (2018). *A fundamental critique of seamless design*. http://gauthierroussilhe.com/en/posts/critique-seamless-design

Sachs, P. (1995). Transforming work. *Communications of the ACM*, *38*(9), 36–44. https://doi.org/10.1145/223248.223258

Science Enclyopedia. (2022). *Magic*. https://science.jrank.org/pages/10049/Magic-Functions-Effects-Magic-in-Classic-Anthropological-Works.html

Scrum.org. (2022). *Professional scrum™ training*. https://www.scrum.org/courses

Seaver, N. (2018). *What should an anthropology of algorithms do?* https://www.google.com/search?q=website+access+dates+apa+7th&oq=website+access+dates+apa+7th&aqs=chrome..69i57j0i8i13i30l2j0i390l2.6069j0j7&sourceid=chrome&ie=UTF-8

Silva, G. (2019). Traduttore-traditore all over again?: The concept of translation in the actor–network theory. In C. Carrasco, M. C. Munoz, & C. D. Carbajo (Eds.), *Traducción y sostenibilidad cultural* (pp. 401–406). Editions University of Salamanca.

South African Government. (2021). *Safety in the workplace*. https://www.gov.za/covid-19/companies-and-employees/safety-workplace

Spiegel, A. (2005). From exposé to care: Preliminary thoughts about shifting the ethical concerns of South African social anthropology. *Anthropology Southern Africa*, *28*(3–4), 133–141. https://doi.org/10.1080/23323256.2005.11499923

Statista. (2021). *Number of smartphone users in South Africa from 2014 to 2023 (in millions)**. https://www.statista.com/statistics/488376/forecast-of-smartphone-users-in-south-africa/

Steele, C. (2019). *What is the digital divide?* Digital Divide Council. http://www.digitaldividecouncil.com/what-is-the-digital-divide/

Steiber, A., & Alange, S. (2015). Preface: It's time for a new model. In A. Steiber & S. Alange (Eds.), *The silicon valley model: Management for entrepreneurship* (pp. vii–x). Springer.

Stevens, E. (2021). The fascinating history of UX design: A definitive timeline. https://careerfoundry.com/en/blog/ux-design/the-fascinating-history-of-ux-design-a-definitive-timeline/

Stigliani, I., & Ravasi, D. (2018). The shaping of form: Exploring designers' use of aesthetic knowledge. *Organization Studies*, *39*(5–6), 747–784. https://doi.org/10.1177/0170840618759813

Swabey, P., & Harraca, M. (2021). *Digital power: How big tech draws its influence*. TechMonitor. https://techmonitor.ai/policy/big-tech/power-of-tech-companies

Technology Practice Team. (2020). *Technology, adaptability and mindset to drive growth, future-proof business*. Kestria. https://kestria.com/insights/technology-adaptability-and-mindset-drive-growth-f/

The Editors of Encyclopaedia Britannica. (2022). *East India company*. Britannica. https://www.britannica.com/topic/East-India-Company

Vahid, F. (2003). The softening of hardware. *Computer*, *36*(4), 27–34. https://doi.org/10.1109/mc.2003.1193225

van Dooren, T. (2014, 20 October). Care: Some musings on a theme. https://www.thomvandooren.org/2014/07/19/care-some-musings-on-a-theme/

Verheyen, G. (2017). *Scrum, what's in a name?* DZone. https://dzone.com/articles/scrum-whats-in-a-name

Vertesi, J. (2012). Seeing like a rover: Visualization, embodiment, and interaction on the Mars exploration rover mission. *Social Studies of Science*, *42*(3), 393–414. https://doi.org/10.1177/0306312712444645

Vertesi, J. (2014). Seamful spaces. *Science, Technology, & Human Values*, *39*(2), 264–284. https://doi.org/10.1177/0162243913516012

Wilf, E. (2014). Semiotic dimensions of creativity. *Annual Review of Anthropology*, *43*, 397–412. http://www.jstor.org.uplib.idm.oclc.org/stable/43049582

Wilf, E. (2016). The post-it note economy: Understanding post-Fordist business innovation through one of its key semiotic technologies. *Current Anthropology*, *57*(6), 732–760. https://doi.org/10.1086/688952

Zunner-Keating, A. (2020). *Marxism and anthropology*. Medium. https://amanda-zunner-keating.medium.com/marxism-and-anthropology-b8172d55a101

Appendix A: Research Ethics Committee Approval Letter

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Faculty of Humanities
Fakulteit Geesteswetenskappe
Lefapha la Bomotho

Humanities 100. Since 1919

9 February 2022

Dear Miss HM Acutt,

Project Title:	Worlding an App: Emergent Webs of Conflict and Compromise
Researcher:	Miss HM Acutt
Supervisor(s):	Dr PFD Krige
Department:	Anthropology, Archaeology and Development Studies
Reference number:	17031118 (HUM043/0721 Line 1) (Amendment)
Degree:	Masters

Thank you for the application to amend the existing protocol that was previously approved by the Committee.

The revised / additional documents were reviewed and **approved** on 09 February 2022 along these guidelines, further data collection may therefore commence (where necessary).

Please note that this approval is based on the assumption that the research will be carried out along the lines laid out in the amended proposal. Should your actual research depart significantly from the proposed research, it will be necessary to apply for a new research approval and ethical clearance.

We wish you success with the project.

Sincerely,

Prof Karen Harris
Chair: Research Ethics Committee
Faculty of Humanities
UNIVERSITY OF PRETORIA
e-mail: tracey.andrew@up.ac.za

Room 7-27, Humanities Building, University of Pretoria, Private Bag X20, Hatfield 0028, South Africa
Tel +27 (0)12 420 4853| Fax +27 (0)12 420 4501|Email pghumanities@up.ac.za | www.up.ac.za/faculty-of-humanities

Appendix B: Example of Signed Letter of Informed Consent

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Faculty of Humanities
Fakulteit Geesteswetenskappe
Lefapha la Bomotho

Department of Anthropology and Archaeology

Participant Informed Consent Form: Interviews
Hannah Acutt, MSocSci Anthropology
Worlding an App: Emergent Webs of Conflict and Compromise

Introduction

The purpose of this document is to inform you about the purpose of my research project, and to inform you of my rights and responsibilities to you as the researcher. Once you have read this form, you will be given the opportunity to ask any questions about my research. Thereafter, you will be given the opportunity to decide whether you wish to participate in this research or not. If you do consent to participate, please sign this form or provide verbal consent of your agreement, as well as indicate whether you give permission for your contribution to be recorded. Please note that this recording, a transcript of it and notes that will be considered data for this project have to be stored securely at the Department of Anthropology & Archaeology for a period of 15 years. According to your preference, interviews will be conducted either via a video conferencing platform (such as Zoom) or in person at either your office or an agreed-upon public space. Should the Covid-19 pandemic escalate to a point where physical contact is either restricted or banned, all interviews will take place over a video conferencing platform for the safety of both participants and me, as the researcher. If interviews take place in person (Covid-19 restrictions permitting), all participants in the interviews will adhere to government guidelines on maintaining safety from Covid-19: masks must be worn at all times, and I will provide disposable masks should you, the participant, require one; we will sit at least six feet apart according to social distancing rules; and all individuals will sanitise their hands at the commencement of the interview and at its closing. Participants should also advise me, the researcher, if they display any of the identified symptoms of Covid-19 so that the interview can be postponed until the quarantine period of 10 days is completed, or so that the interview can be rescheduled to take place over a video platform.

Description of research

My name is Hannah Acutt and I am currently studying to gain my Masters in Anthropology from the University of Pretoria. My Masters research project is about a topic of my choosing and is titled 'Worlding an App: Emergent Webs of Conflict and Compromise'. The main aim of this project is to document and understand decision-making processes implicated in building/designing an app in a corporate environment. As part of my research, I plan on interviewing designers, coders, members of the management team, and clients who have been or are involved in the making, designing and building of an app in a company. This Participant Informed Consent Form is a prerequisite for conducting these interviews. I hope my project will add to academic understanding of the increasing role of apps in everyday life.

Voluntary participation

As a part of this study, I am requesting your permission to be interviewed for this project. If you agree, I would like to record our interview so that I can transcribe it at a later date. I expect the interview to last an hour. As per

---

Room 8-15, Humanities Building
University of Pretoria, Private Bag X20
Hatfield 0028, South Africa
Tel +27 (0)12 420 2595 | Fax +27 (0)12 420 2595
Email: sanna.sechele@up.ac.za | www.up.ac.za

the ethical guidelines from the University of Pretoria, and the Ethical Guidelines from the professional association Anthropology Southern Africa, I have to explain to you that your participation in this research project is entirely voluntary and that you can withdraw at any point should you feel uncomfortable or not wish to continue. No one can force you to participate in an interview. If you agree to participate, you are not under any obligation to answer all the questions that I will be asking, and there will be no repercussions to you should you choose not to answer a question or withdraw from the research. There will be no right or wrong answers to my questions.

21.04.2022

Confidentiality, benefits and discomforts

It is very common in anthropological research for research participants to state from the outset that they wish to remain anonymous and that a pseudonym be used in the final written research outcome (whether a dissertation or article). Should you wish, I will keep your identity anonymous. I will also not reveal your identity and personal information to other employees of the company where you work. This information will only be used for the purposes of academic writing.

There are no direct personal benefits that you will get by participating in this study. However, the study will enhance our knowledge on how the relationships and decision upon which apps are created, which may be used by other members of the scholarly community. There are no anticipated risks attached to participating in this study. However, if you have any concerns about how the interview was conducted, or any other concerns regarding your participation in this study, please contact my research supervisor.

Contact details of researcher and supervisor

Ms Hannah Acutt, hmacutt@gmail.com and contact number 083 302 4143

Dr Detlev Krige at detlev.krige@up.ac.za, Senior Lecturer in Anthropology, University of Pretoria.

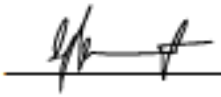
The department can be contacted via the Senior Departmental Administrator whose contact details appear in the footer of this document.

Consent

	Yes	No
I confirm that the purpose of the study has been fully explained to me	X	
I understand that no personal benefit will accrue to me as a participant	X	
I understand that I am participating freely	X	
I want to remain anonymous		X
I agree that the interview be recorded	X	

Room 8-16, Humanities Building
University of Pretoria, Private Bag X20
Hatfield 0028, South Africa
Tel +27 (0)12 420 2595 | Fax +27 (0)12 420 2595
Email sanna.sechele@up.ac.za | www.up.ac.za

I agree that data be stored for 15 years in the Department of Anthropology & Archaeology	X	



21.04.2022

Signature of interviewee Date

H.M.Acutt 21/04/2022

Signature of researcher Date

Room 8-16, Humanities Building
University of Pretoria, Private Bag X20
Hatfield 0028, South Africa
Tel +27 (0)12 420 2595 | Fax +27 (0)12 420 2595
Email sanna.sechele@up.ac.za | www.up.ac.za

Appendix C: Example Interview Schedule for Designers (App Company)

- Could you start by explaining what you studied, where you have worked, and the kinds of jobs you have done?
- Could you explain to me what your official roles and responsibilities are according to your contract? What tasks are you expected to complete for your role?
- In terms of making, assigning, and classifying tasks on the Jira board, do you think that this works well or are there any areas that you feel still need to be worked on? How has the system evolved since you've been working here?
- How do you understand how tasks are made on the Jira board? Do you find the classification and assigning of tasks to work well?
- Could you explain the set-up of your team and team members? Does the hierarchy help to provide a process for resolving conflicts, or does it make it more difficult to resolve conflicts?
- Do you have a work process that you follow, such as a way of assigning and defining tasks to team members? How do you split up work time?
- When you are designing an element, how do you balance the client's requirements/requests and what your design will mean for the developers? Could you explain your mental process for balancing these two needs, and when one might take precedence over the other?
- Why do you think it is important to make the wireframes look as realistic and functional as possible for the client? Why, for example, does it help to make the check box look like it can be checked?
- 'Seamless' is a big word in technology in general and in App Company's ways of speaking, but I'm interested in 'seamfulness', which is paying attention to how you make the seams of your work disappear for the user. So why do you think it is important that

an app/design appear seamless? Could you give me an example of a functionality you have tried to make appear seamless for the user?

- Have there been any examples of miscommunications between your team, the client, and the developers? How did this miscommunication arise and how did you resolve it?
- Is there a difference between how you speak with the developers about how to implement functionalities and with the client about what functionalities they require? If so, could you explain how you have developed this difference in ways of speaking, and how necessary it is to complete your task accurately.
- What electronic means of communication do you use? What are the advantages and disadvantages of this platform for communicating with other members of your team? What communications are made on private chats and what communications are made publicly?
- Which technologies that you use every day do you find help make your job easier to perform? Which technologies do you find can make your work processes more convoluted or difficult to perform?