The implementation of a mobile application to decrease occupational sitting through goal setting and social comparison

Submitted in fulfilment
of the requirements for the degree of

Master of Science

at Rhodes University

Moipone Tsaoane

Grahamstown, South Africa
February 2022

Abstract

Background: Feedback proves to be a valuable tool in behaviour change as it is said to increase compliance and improve the effectiveness of interventions. Interventions that focus on decreasing sedentary behaviour as an independent factor from physical activity are necessary, especially for office workers who spend most of their day seated. There is insufficient knowledge regarding the effectiveness of feedback as a tool to decrease sedentary behaviour. This project implemented a tool that can be used to determine this. To take advantage of the cost-effectiveness and scalability of digital technologies, a mobile application was selected as the mode of delivery. **Method:** The application was designed as an intervention, using the Theoretical Domains Framework. It was then implemented into a fully functioning application through an agile development process, using Xamarin.Forms framework. Due to challenges with this framework, a second application was developed using the React Native framework. Pilot studies were used for testing, with the final one consisting of Rhodes University employees. **Results:** The Xamarin.Forms application proved to be unfeasible; some users experienced fatal errors and crashes. The React Native application worked as desired and produced accurate and consistent step count readings, proving feasible from a functionality standpoint. The agile methodology enabled the developer to focus on implementing and testing one component at a time, which made the development process more manageable. **Conclusion:** Future work must conduct empirical studies to determine if feedback is an effective tool compared to a control group and which type of feedback (between goal-setting and social comparison) is most effective.

Keywords: Feedback, sedentary behaviour, digital behavioural change interventions, agile, mobile applications

Acknowledgements

I want to express my deepest gratitude to Mr Gys Booysen and the Telkom team for their support. I would also like to thank my supervisors for their continued guidance and patience throughout this process. Thank you to Tavonga Chifamba for his assistance and expert advice with the implementation and Senate Mangakane for editing the thesis. I am so grateful to the Rhodes University staff for their participation and adaptability during the pilot tests. Thank you to my family for their love and support and my sisters for always carrying me. To both my grandfathers who believed in me from the very beginning.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Sedentary behaviour is a risk for poor health. Prolonged sitting is linked to detrimental cardiometabolic health outcomes, including obesity, dysglycemia, diabetes, cardiovascular disease, premature mortality and stiffness (Homer et al., 2019). Prolonged sitting can also affect pulmonary function because of the positioning of the diaphragm (Kang et al., 2016). Some pulmonary functions that can be affected include a lowered lung volume, weak muscles when breathing out and airway obstruction (Kang et al., 2016). Despite its risks, prolonged sitting has become an everyday behaviour, as sitting is common at work, while driving or in public transportation, and in front of screens at home (Homer et al., 2019; Mirzaei et al., 2019). Office workers spend, on average, three-quarters of their workdays sitting (Brakenridge et al., 2016). There is, therefore, a need for solutions that can change this behaviour. One approach is the use of Behaviour Change Intervention (BCI).

Traditional Behaviour Change Techniques (BCTs) are often resource-intensive, time-limited and require regular travelling to experts (Stockwell et al., 2019). There is, therefore, a need for approaches that are low cost, scalable and less staff intensive (Stockwell et al., 2019). Information and Communications Technologies (ICT) are part of every societal domain and support most human activities (Patrick et al., 2016). An upsurge of researchers is attempting to change behaviour through these technologies. These are called Digital Behaviour Change Interventions (DBCI) and benefit from affordability, accessibility and scalability (Fulton et al., 2018). DBCIs "employ digital technology to promote and maintain health, through primary or secondary prevention and management of health problems" (Michie et al., 2017).

Feedback poses a possible solution for decreasing sedentary behaviour. Kramer and

Kowatsch (2017) state that "providing feedback is a technique to promote health behaviour that is emphasised by behaviour change theories." Feedback can contain information such as guidance, motivation and critical comparisons, which may prompt behaviour change (Mory, 2003). It has also been found to increase compliance (Kramer and Kowatsch, 2017). As such, feedback can be used in interventions to change sedentary behaviour. Literature describes different types of feedback, which may have differing effects on behaviour change (Lipnevich and Panadero, 2021). However, there is insufficient knowledge about which type of feedback is the most effective for decreasing sedentary behaviour (Kramer and Kowatsch, 2017). This project aims to create a mobile application that will provide different types of feedback depending on a user's mode to determine which type is the most effective. Well designed software system practices can provide the best tools and practices to develop applications with high acceptability and user engagement rates. On the other hand, ergonomics provides an understanding and application of theory to increase effectiveness for behaviour change. This project integrated the two fields, computer programming and ergonomics, to design and implement the mobile application.

The project is a continuation of the 2019 Honours project (Tsaoane and Zschernack, 2019), which aimed to determine the best content, functionality and aesthetic design of a mobile application that will reduce sedentary behaviour among office workers. The Theoretical Domains Framework (TDF) by Atkins et al. (2017) was followed to design the feedback application as an intervention. This framework was created to understand the determinants of behaviour in a theory-informed way (Atkins et al., 2017). The Behaviour Change Wheel (BCW) was used in the first step of the TDF to define and specify the behaviour (Michie et al., 2011). The COM-B model of behaviour was used alongside it, and its primary function was to help determine the factors which needed to change for the behaviour to change Michie et al. (2011). Once these factors had been identified, Behaviour Change Techniques were applied to the designing of the mobile application as suggested by Fulton et al. (2018). The designs were established through an iterative co-creation and co-production process in which potential users of the app were involved. Focus groups were set up to discuss the application's content, functionality, and wireframes. Each focus group followed a similar pattern: first, the researcher presented the proposed phase. Focus group members shared their feedback regarding the phase. Finally, adjustments were made based on the feedback and presented to the group for more feedback. After these focus groups, the designs for the mobile application's content, functionality, and wireframes were finalised.

This project implemented these designs into a fully functioning mobile application. Pilot

testing was conducted to determine the application's feasibility and overall functionality. The implementation process was done through an agile development cycle and covered the following phases:

Phase 1: Development of a mobile application

Phase 2: Iterative pilot testing and adaptation

Phase 3: Final pilot testing with a sample of Rhodes University employees

Chapter 2

Literature Review

This chapter reviews the primary literature that guides this project. Section 2.1 discusses sedentary behaviour, including its risks and the recommendations to decrease it. Behaviour Change Intervention (BCI), which can be employed to decrease sedentary behaviour, and related theories are discussed in Section 2.2. Mobile application development, best practices, technologies and considerations are discussed in Section 2.3. Finally, co-creation and the agile methodology that is followed during the development process are discussed in Section 2.4.

2.1 Sedentary Behaviour

Sedentary behaviour is defined as sitting, reclining or lying and has an energy expenditure of 1.5 metabolic equivalents (METs) or less (Carson et al., 2016). Metabolic equivalent is the energy that that one requires while at rest (Jett et al., 1990). Recently, sedentary behaviour has been recognised as an independent risk factor (Maher et al., 2014). It should therefore be acknowledged that sedentary behaviour is distinct from physical inactivity (PI); with PI being an insufficient amount of moderate-to-vigorous physical activity (Homer et al., 2019). For this paper, the main focus will be prolonged sitting rather than a lack of physical activity.

2.1.1 Risks and prevalence

Sitting is not hazardous unless it occurs in high volumes (van der Ploeg and Hillsdon, 2017). Research shows that risks start to increase significantly at around 7–8 hours of

sitting per day (van der Ploeg and Hillsdon, 2017). This is referred to as prolonged sitting, which is defined as unbroken bouts of 30 minutes or more (Hadgraft et al., 2016). Accumulated sitting in prolonged bouts is associated with cardiometabolic risks, obesity, dysglycemia, diabetes, intervertebral disk issues, passive stiffness, a decrease in lumbar range of motion, chronic diseases, premature mortality and cardio-metabolic risk (Homer et al., 2019; Kang et al., 2016; Hadgraft et al., 2016). Brakenridge et al. (2016) state that these bouts may be detrimental to cardiovascular and musculoskeletal health.

Cohort studies from Australia and the US reported an association between mortality and total sitting time (van der Ploeg and Hillsdon, 2017). Even when individuals live for longer, many live with disabilities and a lowered quality of life (Stockwell et al., 2019). Mirzaei et al. (2019) found an increase in the waist circumference of the participants with the longest sitting time per day. Waist circumference was normal for participants that only spent 1–3 hours of their day sitting. Lower back pain has increased among workers, especially call-centre workers, who spend up to 95% of their day sitting. Lower back pain is the third leading cause of self-perceived disability and presents a major economic burden (Bontrup et al., 2019). Sitting for longer than seven hours a day has been said to increase this risk significantly (Bontrup et al., 2019). In a study by Bontrup et al. (2019), 75% of the participants had some level of chronic or acute back pain. Credeur et al. (2019) reported an association between prolonged sitting and a decrease in the vasodilatory function in the leg (a decrease in the vessels' ability to dilate, which can increase blood pressure).

Sedentary behaviour is prevalent in modern society and is involved in everyday tasks such as transportation, eating, watching TV, video gaming and seated work (Homer et al., 2019). Due to changing environments and advances in technology, many adults are office-based and work in screen-and-desk-based spaces (Homer et al., 2019). Office-based workplaces are highly exposed to sedentary time, with many people spending between two-thirds and three-quarters of their workday sitting (Hadgraft et al., 2016). Brakenridge et al. (2016) support this claim, stating that office workers spend, on average, three-quarters of their day sitting. In a cohort study of 8000 workers, it was found that 77% of their waking time was spent in sedentary behaviour (Homer et al., 2019).

Sedentary behaviour during Covid-19

Sedentary behaviour has become especially common due to the Covid-19 pandemic. In an attempt to decrease the spread of Covid-19, people have been advised to stay home

as much as possible. Unfortunately, this meant that many opportunities to be physically active were suspended (Hall et al., 2021). Parks, school playgrounds and recreational sporting facilities were mainly closed, which decreased the chances of people being active (Hall et al., 2021). As a result, sedentary behaviour, which was already an issue before the pandemic, is an even bigger issue; most people are sedentary at home in front of computers, phones or TV screens (Davy et al., 2021). Decreased social interaction, especially among adolescents, has increased sedentary behaviour, as many activities were done with others. Men and women between ages 50–81 reported that there had been a significant decrease in their physical activity and an increase in their sedentary behaviour since the pandemic.

As a result of lockdown and social distancing, many office-based employees work remotely from their homes (Jain et al., 2022). Working from home may have implications for employees' sedentary behaviour. A study that investigated the implications of working from home in Midrand, South Africa, stated that there was a major change in transportation (Garg and Rijst, 2015). This may limit some employees' opportunities to be active if they walked or cycled to work. While some employees feel more productive at home, others were reported to experience distractions that hindered their productivity (Jain et al., 2022). Being productive may mean that employees complete their tasks quicker and therefore sit less, but it may also mean that they work more and sit for longer than they did in the office where there is a more regulated work-departure time. Participants who experience distractions may sit for longer in total, spending more time to complete tasks. On the other hand, they may have shorter sitting bouts due to these distractions. Another cause of shorter bouts may be the social influences from family, friends and neighbours (Jain et al., 2022). Remote work offers employees a more flexible schedule, allowing them to spend more time with families and advance their careers and interests, which may also lead to shorter sitting bouts (Garg and Rijst, 2015).

2.1.2 Recommendations and guidelines

A 24 hour time period can be distributed among behaviours on a low to high energy expenditure continuum (Carson et al., 2016). These behaviours range from sleep, which has the lowest energy expenditure, to vigorous-intensity activities, which have the highest energy expenditure (Carson et al., 2016). Each behaviour on the continuum serves an essential purpose; both rest and exercise are critical to overall health. However, it is vital that each of these behaviours is done for an appropriate duration. As previously stated, sedentary behaviour is distinct from physical activity and must each have appropriate

guidelines. Guidelines to increase physical activity are quite common. The World Health Organisation recommends at least 150 minutes of moderate-intensity aerobic physical activity, 75 minutes of vigorous-intensity or an equivalent combination of both throughout the week (van der Ploeg and Hillsdon, 2017). Research states that one hour of daily moderate to vigorous intensity is beneficial for both physical and mental health (Carson et al., 2016).

It should, however, be noted that following these guidelines does not wholly safeguard one from the risks mentioned above of sedentary behaviour (van der Ploeg and Hillsdon, 2017). These guidelines target the physical inactivity issue, which, as stated previously, is distinct from sedentary behaviour. A worker who does 30 minutes of moderate to vigorous physical activity daily meets WHO's recommendations but is still at risk if the remainder of the day is sedentary (van der Ploeg and Hillsdon, 2017). One can meet WHO's physical activity recommendations within just 2% of the day, but that leaves 98% of the day (van der Ploeg and Hillsdon, 2017).

It is, therefore, still important to determine how this 98% is distributed between sedentary behaviour and light-intensity activities. Light intensity activities are defined by an energy expenditure between 1.5 and 3 METs and include static standing and ambulatory activities (moving around) (Carson et al., 2016; Jett et al., 1990). Behaviours must be monitored across the energy expenditure spectrum instead of only focusing on higher intensities (van der Ploeg and Hillsdon, 2017). Moderate and vigorous-intensity physical activities can only effectively decrease the risks of prolonged sitting if they are done for 35,5 hours every week, which is over four times more than the WHO recommendations (van der Ploeg and Hillsdon, 2017). This does not seem practical as, already, 31.1% of adults fail to meet WHO's recommendations (Mirzaei et al., 2019). Homer et al. (2019) supports this claim, stating that the WHO physical activity guidelines are only met by 1 in 3 adults and that the average time spent on moderate to vigorous intensity activities was only 26.9 minutes per day. It should, therefore, come as no surprise that several national guidelines are starting to include sedentary behaviour recommendations (van der Ploeg and Hillsdon, 2017).

Recommendations regarding sedentary behaviour are not fully established yet, but research has made a few suggestions. A study with a sample of 4840 participants discovered that as little as 4000 steps a day could lead to lower all-cause mortality (Saint-Maurice et al., 2020). The study also asserts that the intensity of these steps is not of importance, implying that people can move in different environments and contexts, not only as a means of working out. The Canadian 24-Hour Movement Guidelines for Adults aged

18-64 years recommends the following (CSEP, 2021):

- No more than eight hours of sedentary time a day.
- Break up sitting bouts as regularly as possible.
- No more than three hours of recreational screen-time.
- Several hours of light-intensity physical activities.

The World Health Organisation (WHO) has published guidelines for people to stay active and reduce sedentary behaviour during Covid-19 (WHO, 2022b). WHO recommends that people take active breaks during the day and interrupt their sitting bouts every 30 minutes. Standing desks are also suggested and can be created by stacking books on a regular desk.

2.2 Behaviour Change Interventions

Effective, context-specific, and scalable strategies are needed to support and carry through the above guidelines. One approach to this is the use of Behaviour Change Interventions (BCIs). BCIs are "coordinated sets of activities designed to change specified behaviour patterns" (Michie et al., 2011). Developing, implementing and evaluating effective BCIs is key to the advancement of behavioural sciences and its attempts to change behaviour (Michie et al., 2015). A recommended approach to ensuring the effectiveness of these interventions is by rooting their design and implementation in evidence-based theory (French et al., 2012). Hedin et al. (2019) supports this recommendation, stating that interventions are better grounded when based on behaviour change theory.

2.2.1 Behaviour Change Theory

Behaviour scientists and implementation researchers collaborated and identified different implementation theories to create a comprehensive framework that can be used to design theory-based interventions (Atkins et al., 2017). The result was the Theoretical Domains Framework (TDF); a synthesis of 33 theories of behaviour, grouped into 14 domains. The TDF does not specify one theory for behaviour change but rather provides a means

through which researchers can determine the cognitive, affective, social and environmental influences on behaviour (Atkins et al., 2017).

Out of 83 behaviour change theories, only three were found to be comprehensive, with many being group-level and static (Hekler et al., 2016b). Interventions should be dynamic to match dynamically changing individuals and social and environmental factors that interact to cause behaviour change (Hekler et al., 2016b). As a result, Hekler et al. (2016b) created a new framework for behaviour change, which uses state-space representation. This framework defines when, where, for who, and in what state an individual has to be to achieve the behaviour change. However, a more commonly used framework was created by Michie et al. (2011), after evaluating the shortcomings of 19 existing behaviour change frameworks. This framework, the Behaviour Change Wheel (BCW), claims to be more comprehensive and provides a practical guide to design and evaluate Behaviour Change Interventions and policies. More and more researchers are choosing the BCW as a guide to designing Behaviour Change Intervention.

Murtagh et al. (2018) designed an intervention through the use of the eight steps of the BCW. The intervention aimed to promote physical activity among adolescent girls by engaging their mothers. This framework was said to provide an in-depth understanding of the target behaviours and allow the intervention phases to be completed systematically. One reported limitation of the framework is that it was time-consuming.

The Behaviour Change Wheel uses a "behaviour system" called the COM-B system (see Figure 2.1), which specifies the conditions for behaviour change. This system suggests that behaviour occurs as a result of the interactions between an individual's capabilities, opportunities and motivation (Michie et al., 2011), described below:

Capability The psychological and physical capacity to perform a behaviour.

Opportunity The social and psychological factors that are external to an individual and can prompt or enable behaviour.

Motivation The habits, decisions, emotions and other mental processes that direct behaviour.

Behaviour Change Techniques (BCTs) can then target an individual's capabilities, opportunities or motivation. BCTs are the smallest active components of an intervention and are capable of changing behaviour (Fulton et al., 2018). There is a total of 93 techniques,

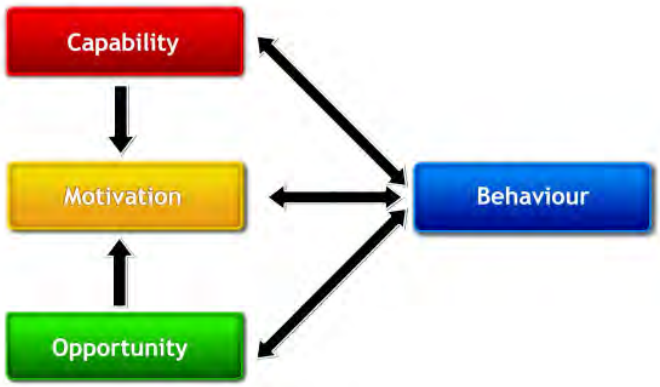

Figure 2.1: The COM-B system which is a framework for understating behaviour (Michie et al., 2011).

which are divided into 16 categories. The effectiveness of BCTs depends on their design, dose, duration and content (Fulton et al., 2018). One must also consider the provider, format, materials, settings, intensity, tailoring and style (Fulton et al., 2018).

Hedin et al. (2019) suggest that using multiple BCTs in an intervention will more likely lead to behaviour change. A minimum of three behaviour technique clusters is suggested for an improved outcome (Stockwell et al., 2019). The downside of this is that it makes it difficult to determine why a particular intervention worked (Hedin et al., 2019). Fulton et al. (2018) got around this issue through iterative bottom-up and top-down processes. The bottom-up process ensured that all necessary techniques were included in the interventions, and the top-down process then breaks down the intervention into its parts to gain insight into each (Fulton et al., 2018).

In studies that were aimed at changing behaviour in food sustainability, various techniques were used such as nudging, persuasive technology, visualisation, gamification and eco-feedback (Hedin et al., 2019). Techniques such as group walks, individual counselling and self-monitoring through pedometers increased moderate to vigorous physical activity (MVPA) by up to 56 minutes (Stockwell et al., 2019). A systematic review found feedback, social support, goal-setting and self-monitoring to lead to significant improvement (Stockwell et al., 2019). Goodwin et al. (2016) reviewed 22 papers that used BCTs in Coronary Heart Disease interventions and discovered the most common to be automated text messaging, provision of information, and goal setting.

Bondaronek et al. (2018) assessed the use of BCTs among 400 interventions that aimed to increase physical activity. The three most commonly used BCTs categories were feedback and monitoring, goals and planning, and comparison of behaviour. These categories are described below.

Feedback and monitoring

Feedback can be on behaviour (how it is performed) or on the outcome of that performance. A wide range of studies has proven that feedback and monitoring can successfully change behaviour (Hedin et al., 2019). The control theory provides theoretical reasoning for this outcome (Carver and Scheier, 1982). Feedback has also been seen to increase compliance (Kramer and Kowatsch, 2017). These benefits of feedback have led to researchers conducting feedback-based interventions that "seeks to alter habitual or ingrained behaviours through the targeted provision of information" (Hartley et al., 2021). A systematic review discovered that personalised feedback might be a key BCT in ensuring that interventions effectively change behaviour (Sherrington et al., 2016).

Goals and planning

This technique may be helpful for users who may be less likely to commit to intervention if their performance and behaviour goals are publicly shared (Yardley et al., 2016). A systematic review found that 28% of the analysed studies used goal-setting and planning (Hedin et al., 2019). The control theory suggests that feedback, when paired with goal setting, can effectively lead to behaviour change (Hedin et al., 2019).

Comparison of behaviour

Creating a supportive community with others who face similar risks may increase effective engagement (Yardley et al., 2016). Technology allows for behaviour tracking to be publicly shared, which can promote encouragement among peers (Yardley et al., 2016). Social motivation can be described as the drive for a goal, based on social influence (DiMenichi and Tricomi, 2015). Research found that participants increase their physical effort when they believe that they are in competition, which suggests that motivation can be increased by social factors (DiMenichi and Tricomi, 2015). Participants demonstrated increased attention and effort when they were playing against a competitor (DiMenichi and Tricomi, 2015).

2.2.2 Digital Behaviour Change Interventions

Digital technologies are accessible, scalable and more affordable, and therefore have the potential to address the limitations of traditional Behaviour Change Interventions (Fulton

et al., 2018; Winter et al., 2016). They are growing as a mode of delivery in behaviour change and are called Digital Behaviour Change Interventions (DBCI) (Fulton et al., 2018). DBCIs address current issues in healthcare such as poor access, uncoordinated care and costly health care (OConnor et al., 2016). They can improve health outcomes at a reduced cost because they can be scaled to reach large numbers of people (Michie et al., 2017; Munro, 2018). They have an improved experience, as they are often automated, interactive and can provide personalised feedback based on users' data (Michie et al., 2017). Users' data can be used to adapt and improve support as users' needs change (Hekler et al., 2016b). The last few years have, therefore, seen several studies which attempted to alter people's behaviour using DBCIs (Hedin et al., 2019). For instance, usage has shown positive outcomes when used to increase physical activity, manage type 2 diabetes, and reach weight loss goals (Higgins et al., 2019). A systematic review found that the use of DBCIs decreased systolic blood pressure, increased moderate to vigorous physical activity by up to 52 minutes a week, and decreased sedentary behaviour by 58 minutes per week (Stockwell et al., 2019).

DBCIs to decrease sedentary behaviour

Before digital interventions became available to the mass consumer market, sedentary behaviour was recorded using self-reports, which were said to vastly underestimate sitting time (Homer et al., 2019). Sophisticated, consumer-targeted technology is rapidly emerging and can enable people to track their activities in a more feasible, acceptable and cost-effective manner (Brakenridge et al., 2016). Research has shown that using pedometers in the workplace can increase activity and decrease sitting. In comparison to a control group, participants increased their walking when using a smartphone pedometer (Higgins et al., 2019). There are currently over 200 different activity trackers on the market (Bellone et al., 2016). Their benefit is that they are non-invasive, portable and relatively affordable (Bellone et al., 2016). Most of these trackers use accelerometers to track movement, which is achieved by measuring the force of inertia that is generated when mass is affected by a change in velocity (Bellone et al., 2016). Algorithms can then be written to calculate a user's movements, including standing, walking and running (Preuveneers et al., 2013). Accelerometers are embedded in most smartphones, offering pervasive, non-intrusive measuring (Preuveneers et al., 2013).

Although DBCIs are not limited to smartphone applications, the use of smartphones provides opportunities for DBCIs because they are persuasive, functional, user-friendly and accessible (Hedin et al., 2019; Kaur and Kaur, 2018). Tracking activities with dedicated

devices requires an additional cost, so smartphones are better as most people already own one (Higgins et al., 2019). They have features like location detection and are portable and lightweight (Baktha, 2017; Jayatilleke et al., 2018). They also facilitate access to information regardless of a user's location (Affonso et al., 2019). Many areas in the world have easy internet connectivity and a spread of mobile telephony, which presents an opportunity to improve health using smartphones (Higgins et al., 2019; Bellone et al., 2016). Smartphones and internet connectivity can break down the barriers to health provision as they provide access to over 325 000 health applications, which are often free (Higgins et al., 2019). Among these are applications for nutrition, chronic illness management and step tracking.

Limitations of DBCIs

Despite the benefits of digital interventions and the popularity of smartphones, adoption of and engagement with health-intervention applications are still relatively low (Bondaronek et al., 2018). Very few digital interventions are designed and developed to optimise user uptake, engagement, and sustained behaviour change (Fulton et al., 2018). For effective engagement, an intervention must have perceived benefits for the user (Yardley et al., 2016). Most studies in this field have problems of a methodological nature, and many do not show how they will lead to long term behaviour change (Hedin et al., 2019). The challenge in creating digital behaviour interventions is the application of theory that is commonly intended for non-digital interventions (Fulton et al., 2018). Only 52% of the studies reviewed by Winter et al. (2016) were informed by behaviour theory. This may be due to the lack of knowledge about which types of interventions work to change behaviour in different contexts and how to link theory to health outcomes (Munro, 2018). Literature does not provide much guidance regarding the parameters that can be used to design and develop DBCIs while still incorporating sufficient behaviour change theory (Fulton et al., 2018).

These challenges often lead to a decreased uptake of and engagement with health applications. It has been found that only 35–58% of individuals have at least one health application installed (Higgins et al., 2019). One study found that of those who installed a health app, 45% discontinued use (Higgins et al., 2019). Other reasons for low usage rates and discontinued use is that many people have privacy concerns, are unfamiliar with the platforms or do not think they need the applications (Higgins et al., 2019; OConnor et al., 2016). Additional barriers to usage include people being too busy, lack of personal

motivation to improve health and cost of technologies (OConnor et al., 2016). Some individuals have a lack of literacy and therefore low confidence (Higgins et al., 2019). Lastly, when interventions are used without human support (such as the researcher or a nurse), they are very likely to lead to dropout (uninstalling or ceasing usage) (Yardley et al., 2016).

Increasing effectiveness in behaviour change

DBCI rarely have a clear theoretical basis (Fulton et al., 2018). Theory can be described as a set of concepts that specify how phenomena relate to each other (Hekler et al., 2016b). One of the best practices in intervention design is the use of evidence-based theory (Murtagh et al., 2018). This is said to lead to interventions that are more likely to succeed and can be evaluated to determine the effectiveness of different components (Murtagh et al., 2018). Theory can strengthen data-based studies by identifying the problems that are worth studying and connecting one piece of evidence to another (Horn, 2008). Theory also helps in predicting data patterns in areas that are not yet studied and anticipating what aspects of a problem are likely to be the most critical (Horn, 2008). It explains and predicts generalisations and can provide insight into the future of a specific individual (Hekler et al., 2016b). In DBCIs, a sound theoretical basis is as important as the digital functionality (Fulton et al., 2018). Fulton et al. (2018) suggest looking at each BCT individually and brainstorming different ways in which it could be translated into an application feature and how it could be delivered digitally.

Increasing engagement and user uptake

No matter how well thought through and comprehensive, BCTs may still fail if they do not develop interest within users by having an attractive, user-friendly, and intuitive design. Engagement with health interventions is a precondition for effectiveness (Yardley et al., 2016). Engagement with technology can be defined by the quality of user experience (Fulton et al., 2018). Usability and the quality of use refers to the efficiency of learning, user satisfaction, communication between user and developer (through the interface), and applicability (Yuan and Marques, 2018). Engagement is a dynamic process that starts with a trigger such as a recommendation from a physician (Yardley et al., 2016). This is followed by the initial use, which can then lead to sustained use or disengagement (Yardley et al., 2016). Engagement also depends on the particular intervention, the user and their context (Yardley et al., 2016).

Yardley et al. (2016) argue that intervention usage is not always closely related to outcome. Some individuals may cease usage due to sufficient mastery, meaning that they do not need the technology for further improvement. In contrast, other users may continue rigorous usage due to dependency and low self-management. Additionally, devices might not accurately record offline usage. Focusing on effective engagement rather than sustained engagement is therefore vital. Engagement can be evaluated in several ways- qualitative analysis reports a user's experience with the intervention, while self-report questionnaires are convenient for large sample sizes and allow for standardisation and comparisons. The downside of both methods is possible bias. A solution to this is system logs on usage, which can provide the time that is spent on an intervention and help identify usage patterns (Yardley et al., 2016).

2.3 Mobile Application Development

As discussed above, smartphones are owned by a large proportion of the population, making them an accessible mode of delivery for digital interventions. This section discusses the approaches that can be taken to ensure that a digital intervention optimises the benefits of DBCIs while minimising its challenges. Section 2.3.1 introduces key points to consider about smartphones and mobile applications. Section 2.3.2 discusses some research-based recommendations for developing a mobile application. The two popular software development frameworks, Xamarin.Forms and React Native are compared in Section 2.3.3.

2.3.1 Considerations

For 52% of the time, users uninstall applications due to performance issues (Baktha, 2017). One challenge with mobile applications is the deficiency of resources such as processing, storage capacity, memory, battery life and expensive data transfer over 3G (Shehzaib et al., 2018; Baktha, 2017; Affonso et al., 2019). Higgins et al. (2019) found limited battery charge to be a concern with health applications. Mobile devices have limited computational power compared to conventional computers, and computationally intensive tasks require excessive battery power (Affonso et al., 2019). It is crucial to find ways to minimise the impact of these limitations while still facilitating access to information (Affonso et al., 2019). The core features to consider for an effective application are an engaging design, dynamic and interactive experience, high functionality and content that is based on reliable sources (Yuan and Marques, 2018).

2.3.2 Recommendations

Developers must keep up with high demands for applications while maintaining the quality and staying within budget (Kaur and Kaur, 2018). Applications must provide an intuitive user experience that requires little effort from the user (Fulton et al., 2018). Navigation must be simple, with a sequential rearrangement of content (Jayatilleke et al., 2018). Interface designs account for 58% of the reasons why users install applications (people usually look at the appearance before they commit to the download), so applications should catch the user's eye (Baktha, 2017). Baktha (2017) and Yuan and Marques (2018) suggest bright colours, cool textures, 3D effects, animation and use of multimedia. It should, however, be noted that too many images may flood the cache and disrupt performance (Baktha, 2017). In a semi-structured interview, users stated that icons must not be confusing, and text needs to be more concise and legible (Yuan and Marques, 2018). Jayatilleke et al. (2018) and Fulton et al. (2018) concur, stating that text must be in small chunks to accommodate the screen size and avoid over-scrolling and information overload. Non-patronising, everyday language (with key terms highlighted) is recommended (Fulton et al., 2018). Core features must be created with the target user in mind, and developers must not solely focus on the latest platforms (Baktha, 2017). Applications must be able to work offline, offer regular updates, and have responsive touch support (Baktha, 2017). On first log on, it is advisable to provide users with a virtual tour of the platform (Yuan and Marques, 2018).

Applications should be designed to decrease rather than add on to the burden of self-care and treatment (OConnor et al., 2016). They must be flexible enough to accommodate different demographics as well as changing conditions (OConnor et al., 2016). To accommodate different demographics, Yardley et al. (2016) suggest tailoring, which can be according to a user's lifestyle, interests or motivations. Previously, digital interventions were mainly used by people with higher education and income levels. However, accessibility can be ensured to people with lower literacy and socio-economic levels through tailoring. Tailoring also allows for individuals with special needs to be supported. Users' interests and preferences must be considered in the content and designs of applications (Han et al., 2020). Interventions should allow customisation by users as their needs and status change so that their immediate situation is always supported. The Self Determination Theory suggests that autonomy is a fundamental human need; giving a user an option to customise could, therefore, be motivating (Yardley et al., 2016; Mory, 2003). It should be noted, though, that tailoring and customisations should not be too alienating as that can be demotivating and lead to non-usage (Yardley et al., 2016).

Socio-contextual influences can support or undermine a user's engagement with technology (Yardley et al., 2016). Shehzaib et al. (2018) state that the reliability of a mobile application can be affected by dynamic environmental factors. Environments can be distracting, where usage is for brief periods on small screens and keyboards (Yardley et al., 2016). It is, therefore, essential to develop applications that are context-aware, stylish and adaptive, to accommodate the convoluted environments in which the applications are used (Shehzaib et al., 2018). Context-awareness allows the intervention to respond to a user's immediate situation in a proactive manner (Preuveneers et al., 2013). A user's context can be determined through a smartphone's time, location and activity tracking (Preuveneers et al., 2013). Usage of digital devices, social media and websites provide a wide range of data about users, including their interests and preferences (Hekler et al., 2016b).

Preuveneers et al. (2013) argue that intelligibility is crucial; applications must be smart but not too smart. There must be coherence between what the user knows and what the application knows; otherwise, the user may become frustrated. Automated actions must provide users with clear communication of the internal reasoning. One way of creating an intelligible application is through data visualisation.

Adding human support and facilitation can also increase engagement (Yardley et al., 2016). Many studies did not report support from health professionals (Yuan and Marques, 2018) It is vital to determine when support is necessary, as smaller interventions can be effective without guidance. Human support is mostly necessary when users need expert guidance and reassurance (Yardley et al., 2016). Creating an intervention that is user friendly, adaptive and enjoyable may also eliminate the need for human support (Yardley et al., 2016). In some cases, remote coaching through phones and email may suffice (Yardley et al., 2016).

Mobile Application Acceptability

According to the technology acceptance model, attitude towards technology and interaction with technology are influenced by its perceived usefulness and ease of use (Fulton et al., 2018). Cognitive predictors of engagement, emotion in adoption and continued use of digital interventions are imperative (Fulton et al., 2018). An acceptability study by Shehzaib et al. (2018) extracted user feedback data of 206 applications from Google Play and analysed it through statistical and machine learning methods. It was found that

applications for personalisation, libraries, and entertainment were popular, while applications for education, widgets and lifestyle were the most unpopular. It was also found that health and fitness applications were among the cheapest. The price of an application was found to be influential; users regularly complained about prices and hidden costs. As such, free applications have larger numbers of users. Smaller applications were preferred as they allow users to install more applications on their devices. The challenge with this is that making an application small can restrict the display and interactivity, which can, in turn, negatively affect the application's usability.

Application noticeability is key to its acceptability (Baktha, 2017). The market is growing rapidly, with over 80 billion application downloads since 2016 (Tavakoli et al., 2018). Applications must be made in such a way that people want to download them (Baktha, 2017). App store optimisation is one way for applications to get noticed (Baktha, 2017). The colours and designs of icons must be carefully considered as this is the first thing users see. Studying similar applications can help in the design of an intervention (Han et al., 2020). Applications such as Samsung Health, Google Fit, Nike Training Club can be considered for the physical activity field (Shehzaib et al., 2018).

2.3.3 Software frameworks

Android and iOS are the most popular platforms in the smartphones market, and each has its own design patterns and user interface designs (Baktha, 2017; Adinugroho et al., 2015; Dorfer et al., 2020). However, it is expensive and time-consuming to develop a native application for each platform (Baktha, 2017; Adinugroho et al., 2015). A more effective approach is to use a single environment that can target different platforms and screen sizes (Baktha, 2017). This approach is called Mobile Cross-Platform Development (MCPD), and it allows developers to write code from a single codebase and port it to different platforms (Dorfer et al., 2020). With MCPD, an intermediary language can abstract the developer from the different platforms, allowing them to focus on the business logic (Adinugroho et al., 2015). However, it is worth noting that cross-platform applications are more resource-intensive than their native counterparts, as they use different technologies to compile user interfaces and respond to user input (Dorfer et al., 2020). Even more resources are consumed for peripheral devices such as sensors and cameras as they need to be wrapped (Dorfer et al., 2020). As such, native applications provide faster and smoother user interaction due to lower resource usage and also have the advantage of persistent local storage (Preuveneers et al., 2013; Dorfer et al., 2020). The two commonly used frameworks are discussed and compared below.

Xamarin.Forms

Xamarin.Forms is an open-source UI framework that enables the development of Android, iOS and Windows applications from a single codebase. The applications are compiled on different devices with native controls. The back-end is written in C# and the UI in XAML. XAML is more readable and concise than equivalent code and has great visual clarity due to its parent-child hierarchy. The platform supports the Model View View-Model (MVVM) architecture, which separates the UI (view) from the back end (Model) and links them through data bindings. Xamarin.Forms provides a cross-platform APIs library called Xamarin.Essentials, which offers access to the accelerometer, geolocation, email and more (Microsoft, 2020).

React Native

React Native is a JavaScript library that combines Native development with the React framework. React.JS and React Native are not the same. React.JS is a JavaScript-based framework for designing user interface (UI) in a declarative manner using JavaScript XML (JSX). JSX is simply a syntax extension to JavaScript that is similar to HTML in structure but still has access to all the power of the JavaScript language. React uses component-based architecture to design interfaces. Each component is composable and manages its state, allowing for complex UIs to be created. React is platform-independent as it simply describes the structure of a user interface and the data it encapsulates but does not compile the user interface itself. React Native is an extension of React.JS, and it renders React and JavaScript to native code, allowing for mobile development. React Native allows developers to build mobile applications by using a Native Bridge. The Native Bridge implements a JavaScript thread that translates JavaScript code to native code. Native code can send data back to the JavaScript through registered callbacks to events (Native, 2022).

Platform comparisons

Xamarin.Forms applications provide a more native user interface and user experience and was reported to have a code reuse rate of 85% (Corbaln et al., 2018). The disadvantage of Xamarin.Forms is that the developer is limited to components from the .NET framework as there are not many open source libraries (Gerasymov, 2021). Xamarin.Forms has

stricter compatibility restrictions, which usually make it difficult for developers to implement third-party tools. React Native applications are developed using web technologies, which is advantageous because developers do not need to have in-depth knowledge of the underlying native platforms (Dorfer et al., 2020). Compared to Xamarin.Forms, React Native has a larger community of active developers on platforms such as Github (Gerasymov, 2021; Dorfer et al., 2020).

2.4 An agile, collaborative approach

This section discusses a methodology that can be followed to ensure that the software development process is done efficiently. Collaboration and its importance to the development process are introduced in Section 2.4.1. Section 2.4.2 describes the agile methodology, its relation to collaboration, and its benefits to intervention design. Lastly, in Section 2.4.3, evaluation of the intervention is briefly discussed as this is an essential step in determining its efficacy.

There is a rapid change in technologies that support digital interventions (Patrick et al., 2016). User engagement is changing just as rapidly as these technologies; assessment methods and design interventions must keep up (Michie et al., 2017). There is a need for new methods in research and practice that accommodate the complexity of health problems as well as the volume of data from digital technologies (Patrick et al., 2016). One of the challenges with achieving this is the reluctance to abandon traditional methods, most of which were used in the era of self-reporting data and cannot support the volume, velocity and variety of current data (Patrick et al., 2016). Technologies often change much faster than scientific methods (Hekler et al., 2016a). Patrick et al. (2016) suggest looking to other fields such as engineering and also using an agile science to determine which elements of an intervention work. This suggestion is supported by Michie et al. (2017) who state that collaboration between behavioural scientists and computer scientists plays a vital role by enabling theories to be specified in precise sets of constructs. This highlights three noteworthy aspects of implementing an intervention; collaboration, an agile process and thorough evaluation.

2.4.1 Collaboration

Many multidisciplinary research fields leverage both ICT and health care disciplines (Preuveneers et al., 2013). In a study designing a learning management system, five

groups of stakeholders collaborated to produce an effective mobile application (Jayatilleke et al., 2018).

1. Content experts from nursing, pharmacy and medical laboratory science validated all content.
2. Educational technologists checked the alignment of technical and pedagogical features
3. Novice users checked the overall effectiveness of the application
4. The developer developed the application
5. Researchers identified the impact of the innovation

Similarly, a study by (Urruticoechea-Arana et al., 2020) united experts for an autoimmune disease workshop. Family doctors and rheumatologists created diagnostic algorithms for each symptom. Educators and coordinators assessed the algorithms, and an external IT company adapted them into a free mobile application. By May 2019, there had been 2419 downloads.

(Yuan and Marques, 2018) followed the five-stage programming paradigm to develop a self-care application for patients with fibromyalgia. Physical therapists, patients, a designer and a programmer analysed the requirements and content, then set software objectives.

1. The designer created screen layouts
2. The programmer developed a prototype
3. Ten patients with fibromyalgia pilot tested the application to determine its quality
4. The designer proved the interface, and the programmer built the final product The next step is a randomised trial to assess the application's efficacy.

Co-creation

It is equally important to involve potential users of an intervention in this collaboration to optimise its benefits to them (Patrick et al., 2016). Fulton et al. (2018) defines co-creation as as collaborative design and production (co-design and co-production) process that

involves the end user. Participatory, user-centred design is key to ensuring an engaging and effective intervention (Michie et al., 2017). Involving end-users in the process enables insight about the target population that was previously unknown (Fulton et al., 2018). It grounds the intervention in an understanding of their knowledge, skills, behaviour, motivation, culture and views of the BCTs (Yardley et al., 2016). To recruit participants, Han et al. (2020) posted fliers, while Urruticoechea-Arana et al. (2020) used conferences, workshops and websites.

User-centred studies can take the form of semi-structured interviews or quasi-experimental designs. These studies can then determine the feasibility and acceptability of a mobile application (Han et al., 2020). Han et al. (2020) conducted preliminary testing on users with smartphone experiences. The tests determined user interface satisfaction, ease of learning and system errors. This was followed by a one-group pretest-posttest quasi-experimental pilot study, which aimed to determine feasibility, acceptability and effectiveness. Platforms such as Google Play involve users in the design and development of applications by providing them with the platform to rate and review them based on bugs, functionality, performance and quality (Shehzaib et al., 2018). Developers can then use the feedback to remove bugs, add new features and improve the overall user experience (Shehzaib et al., 2018). This feedback must be analysed with strict quality control (Tavakoli et al., 2018).

Fulton et al. (2018) created a digital intervention called StopApp, a web application that is designed to create greater access to and attendance at the UK Stop Smoking Service (SSS). StopApp is accessible through all digital platforms and does not need to be downloaded as it is a web application. A series of Behaviour Change Interventions were used for the web Application's content. The aim of the interventions can be summarised as increasing motivation, enabling action and prompting behaviour change without excessive thinking. Instead of the traditional expert-driven approach where researchers drive the design and content of the intervention, Fulton et al. (2018) employed co-design and co-production (co-creation) to increase engagement with the digital intervention. The participatory approach was taken beyond the assessment of acceptability, usability and satisfaction of the web application. Iterative and continuous user testing was said to have revealed many necessary changes that would have otherwise been overlooked (Fulton et al., 2018).

2.4.2 An Agile Process

The software community is being driven by the rapid adoption of agile methodology (Kaur and Kaur, 2018). This methodology leverages the advantages of the iterative and

Figure 2.2: The four key values of the Agile Manifesto (Beck et al., 2001)

.

incremental models (Barjtya et al., 2017). According to the agile manifesto, its four central values (displayed by Figure 2.2) are interactive teams, working software, customer collaboration and responsiveness to changes (Beck et al., 2001). An agile process consists of self-organising, cross-functioning teams where software is divided into components, and on each iteration, a working model of the component is delivered (Barjtya et al., 2017; Kaur and Kaur, 2018). Waja et al. (2021) state that these components can be delivered in a short space of time. Here, developing and testing are not seen as different phases (Kaur and Kaur, 2018). After each iteration, the product is tested to determine acceptability, and all errors are rectified before the next iteration(Barjtya et al., 2017; Waja et al., 2021). Before development starts, the "planning" stage of the agile process allows researchers to specify what needs to be implemented and how and the duration, costs, and stakeholders. At this stage, researchers can also determine the sequence and length of iterations and prioritise technical requirements (Waja et al., 2021). Most developers find this process to be a natural fit as feedback, growth, and change are essential during the development cycle (Baktha, 2017; Kaur and Kaur, 2018).

An iterative process is also an appropriate fit to the complexity and many unknowns of behaviour change (Michie et al., 2017; Hekler et al., 2016a). It reacts very well to changing requirements and is highly focused on client feedback (Barjtya et al., 2017). Iterating through the development and evaluation processes allows interventions to be refined to meet user requirements (Yardley et al., 2016). An agile methodology also goes well together with co-creation as it encourages close collaboration between clients, developers, and testers (Barjtya et al., 2017). Iterative and continuous user testing is said to reveal many necessary changes that would have otherwise been overlooked (Fulton et al., 2018). Progress is measured dynamically; the overhead is very low compared to other models, and errors can be identified and removed immediately (Barjtya et al., 2017).

2.4.3 Evaluation

Currently, there is insufficient knowledge on how and why particular DBCIs work, and some not (Munro, 2018). Hedin et al. (2019) suggest that studies on DBCIs should move from being exploratory to having more rigorous testing. These studies must be well planned, well-executed, include control groups, and must be conducted over sufficient periods (Hedin et al., 2019). Hekler et al. (2016a) support this, stating that theory-based interventions must be carefully examined for their efficacy. Criteria that can be used to examine an intervention include effectiveness, usability, and desirability (Hekler et al., 2016a). Pilot studies prove to be a valuable approach to examining interventions and determining their potential effectiveness. This is especially true for the agile context, as each pilot study can provide helpful information to use in the subsequent iteration (Bell et al., 2018). Pilot studies are a smaller version of a bigger trial and can therefore determine if the main trial is feasible (Bell et al., 2018).

This project aims to implement a mobile application that uses feedback to decrease sedentary behaviour. As can be deduced from the above research, this project must find a balance between sufficient behaviour change theory to maximise effectiveness and following software development guidelines to maximise uptake and engagement. The following chapters will attempt this, and the intervention's feasibility will be determined at a later stage.

Chapter 3

Project Background

This section revisits the study from Tsaoane and Zschernack (2019), which was conducted to design the content, functionality and wireframes of a feedback-based mobile application that aimed to decrease sedentary behaviour. The designs were firmly rooted in behavioural, feedback and intervention theories to make the intervention more effective. The theoretical framing that was followed to design the implementation is outlined briefly, and the end results (functionality and wireframes) are shared at the end.

Despite being a risk for poor health, Sedentary behaviour has become very common in office work. Current solutions are focused around encouraging people to exercise, with the World Health Organisation recommending 150–300 minutes of moderate or 75–150 minutes of vigorous-intensity activities, or a combination of the two per week. However, increasing Moderate to Vigorous Physical Activity by 30 minutes daily only decreases sitting time to 9.5 hours per day (van der Ploeg and Hillsdon, 2017). So looking at the whole physical activity spectrum is a more feasible option. One way to do this is through everyday lighter intensity activities that are suitable even during office hours. The proposed solution is a mobile application that can offer a practical and fun guide to decreasing occupational sitting time through feedback (in the form of mobile notifications).

Higgins et al. (2019) stated that despite health apps having so many benefits, people tend to make very little use of them. Reasons include a lack of time, lack of interest, and is sometimes due to difficulties that users may encounter while dealing with technology. Applications that are based on theory are more likely to change behaviour; however, the designs are often overly clinical and with too much information (Fulton et al., 2018). On the other hand, some applications are designed well, with appealing user interfaces and a pleasant user experience. The issue with them is that they are often not rooted

Table 3.1: A guide to using the Theoretical Domains Framework of behaviour change as described by (Atkins et al., 2017)

Stage	Detail
1. Select and specify the target behaviour/s	Use documentary analysis or empirical research to identify and specify who should do what differently, to increase the uptake of evidence-based practice
2. Select the study design	May involve semi-structured individual interviews, focus group interviews, questionnaires, structured observations, documentary analysis or consensus processes
3. Develop study materials	Although materials from previous studies may be used as templates, materials should be adapted to be appropriate to the specified behaviour/s and context
4. Decide the sampling strategy	For exploratory studies, a maximum variation approach is appropriate
5. Collect the data	Published studies have used audio-recorded interviews (face-to-face or telephone; one-to-one or focus group) or questionnaires (paper-based or online)
6. Analyse the data	The objective is to identify the domains that are most relevant to the implementation problem being addressed and to populate those domains with context-relevant and behaviourally specific content
7. Report findings	For interview studies, report presents tables that include illustrative quotations, specific beliefs identified (with frequencies, if appropriate) and classification into domains

in behaviour change theory and therefore fail to be effective at changing behaviour. As such, there has not been a high acceptability rate of health applications in the global population (Higgins et al., 2019). However, there is still a vast potential for technology-enabled interventions in the health domain. The challenge is to perhaps determine how to design health applications to make them as appealing to users as they are effective at changing behaviour. In an attempt to do this, the application was designed first as an intervention to ensure proper use of theory. An agile development process (see Chapter 4) was then employed to ensure proper functioning and a pleasant user experience.

The Theoretical Domains Theory provides researchers with a way to design behavioural change interventions by gathering constructs from different theories (McParlin et al., 2017). Behavioural scientists and implementation researchers worked together to create the framework– Theoretical Domains Framework (TDF) which can be used to identify all the determinants of a particular behaviour, which can then be targeted as a means of changing the behaviour (Atkins et al., 2017). The framework provides seven stages (see Table 3.1), which researchers can use to conduct implementation research. The seven steps were taken to design the application as an intervention; each step will be discussed below.

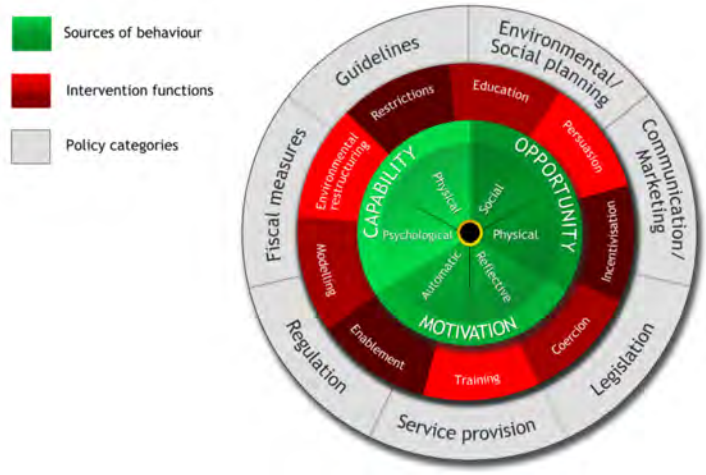

Figure 3.1: The Behaviour Change Wheel, which can be read from the inside out to determine the sources of a particular behaviour, intervention functions that affect these sources, and policies to support the functions at an organisational level (Michie et al., 2011)

Step 1: Select and specify target behaviour

This first step was completed with assistance from the first four steps of the guide to designing interventions which is based on the Behaviour Change Wheel (BCW) (Figure 3.1). The BCW is appropriate in this context as it provides a theoretically based lens to take a closer look at behaviour and ways to change that behaviour. The wheel is a synthesis of 19 behavioural change theories and is therefore quite comprehensive, ensuring that no critical factor is left out. It comprises sources of behaviour, intervention functions that can affect the different sources, and policies to support these functions at an organisational level Michie et al. (2011).

1: Define the problem in behavioural terms

The problem being looked at in this study is sedentary behaviour, particularly prolonged sitting during office-based work. Sedentary behaviour is associated with many health risks, but office workers spend most of their day sedentary, both at work and at home.

2 and 3: Select and specify target Behaviour

The target behaviour was reduced sitting and sitting bouts that were regularly interrupted by standing and light-intensity movement throughout and after the workday. The

target behaviour involved office workers being more aware of their sitting habits and more deliberate in how much they sat, both at work and home. This behaviour was targeted at an individual level so that each individual could manage their behaviour based on their personal requirements and limitations.

4: Identify what needs to change

The COM-B system is a model that was used alongside the Behaviour Change Wheel (BCW), and its primary function was to help determine the factors which needed to change for behaviour to change (Mayne, 2016). It should be noted that feedback was the predetermined and main factor. The COM-B system was used to determine the factors that would work alongside feedback to change behaviour through a mobile application. The COM- B system (green section on Figure 3.1) is illustrated in 2.1, and consists of Capability, Opportunity and Motivation, with an understanding that at least one of these factors need to be present for behaviour to occur. The two chosen factors were motivation and opportunity. Motivation, in this case, is the brain processes such as desires and habits that could lead to decision making and, therefore, direct behaviour. Opportunity refers to all external factors that could enable the behaviour, such as societal norms and financial or time resources. An attempt to change these factors was done through the use of Behaviour Change Techniques, which are the smallest active component of an intervention which are capable of changing behaviour (Fulton et al., 2018). Stockwell et al. (2019) suggest a minimum of three BCT categories for an improved outcome. The application was based on nine BCTs from six technique categories. The three primary categories for this intervention are discussed in 2.2.1.

Below, the nine techniques are listed in their categories. The techniques are defined briefly, followed by an explanation of how the techniques are implemented in this intervention.

Feedback and Monitoring Feedback on behaviour: Provide informative and evaluative feedback on performance. Self-monitoring of behaviour: Provide a method for people to record and monitor their behaviour.

Application

The application provides feedback in the form of mobile notifications. Users receive three daily notifications, informing them of their movement based on a goal they have set or in comparison to others. This is accompanied by tips (see Figure 3.2) on how they could increase their movement.

Goals and planning Goal Setting (behaviour): Allow a person to set a goal in terms of the target behaviour. Review Behaviour Goal(s): Review behaviour, consider goal modification every month, and allow users to reflect on the progress and change the goal if needed. Discrepancy between current behaviour and goal: Analyse current behaviour in light of the set goal.

Application

The application allows users to set a daily step goal, which they can revise and increase as their sedentary habits changed.

Comparison of behaviour Social comparison: Compare person's performance with the performance of others

Application

Users are divided into groups, their step counts are compared to members of the group, and they are ranked by position.

Self-belief Verbal persuasion about capability: Tell person that they are capable; discourage self doubt.

Application

This is represented through use of encouraging language in the notifications and throughout the application. Tips are also provided to encourage increased movement.

Reward and threat Social incentive: Inform reward if there is effort and progress.

Application

The application congratulates the user when they reach their goal or are first in their group.

Natural consequences Information about health consequences: Written or verbal info about behaviour's health consequences

Application

Awareness tabs for definition, stats, recommendations and risks are included with the application. "Did you know" segments are displayed on the homepage to teach users about sedentary behaviour.

A tip accompanied each of the three daily notifications that are discussed in the "Feedback and Monitoring" category. The use of evidence-based theory in intervention design creates interventions that are more likely to succeed (Murtagh et al., 2018). For this reason, a list (Table 3.2) of tips was created by reviewing the literature on sedentary behaviour

Table 3.2: An incomplete list of the tips on the left and the research their based on, on the right. These tips are sent to users from notifications as part of the feedback

Tip	Corresponding research
Place bin a few steps from desk	Bouts are recommended to be 29 minutes or less, as bouts of an hour or more can increase the risk for mortality (Homer et al., 2019).
Stand up for water, tea or coffee	Prolonged sitting can decrease pulmonary functions as quickly as after one hour (Kang et al., 2016)
Take a walk	Alternate between standing, sitting and walking as prolonged standing can lead to musculoskeletal problems such as lower back pain) (van der Ploeg and Hillsdon, 2017).
Cook to cut down on takeaways	Short bouts of activity are said to mitigate cardiometabolic risks that are associated with insulin resistance (Homer et al., 2019)
Walk out and watch the sunset	Positive changes have been seen in postprandial glucose metabolism when bouts of sitting were interrupted with brief physical activity (Homer et al., 2019).
Stand up and stretch	Substituting large amounts of sitting with standing throughout the day can lead to substantial differences in energy expenditure (van der Ploeg and Hillsdon, 2017)

studies. These tips provided practical ways for users to shorten their sitting bouts and decrease their total sitting time. Guidelines are often based on activities that require moderate to vigorous intensity activities such as exercising and jogging. Taking a walk is often suggested, but it is not always feasible at work. These tips provide lighter intensity alternatives that are appropriate during office hours.

Step 2 and 3: Select the study design and study material

The study used focus groups and information surveys that each member of the focus group completed. The material was collected from behavioural theory and feedback research for the application's content.

Step 4 and 5: Sampling strategy and Data Collection

The target adopters of the study were office workers at Rhodes University. More specifically, employees who work in the divisions of the Rhodes Registrar were invited to take part in the study. This group of people were highly exposed to the risks of sedentary behaviour as the nature of their work required sitting at a desk for extended periods. Data

was collected through a user-centred co-creation process. The co-creation process took the form of focus groups. The app's content, functionality and wireframes were presented to participants as done by Fulton et al. (2018). Data were collected on participants views, concerns and requirements regarding the content, functionality and wireframes. This data was evaluated, and each section was reviewed and presented back to participants. The functionality and wireframes that were presented to participants are detailed:

Functionality

Figure 3.2 illustrates the final functionality that was presented and accepted by participants at the focus groups. The main aim of this flowchart was to represent how users can navigate within the application. The application checks if the user is already signed up because first-time users need to select the mode (goal-setting or competition) as well as provide their gender and schedule. This will happen only once then users will be navigated directly to the homepage when they log in. However, users can select menu from the homepage and navigate to the profile page if they need to change any of the details. Lucid chart was utilised to design the flowchart.

Wireframes

Wireframes were designed next, using Axure RP. Wireframes gave users an idea of how each application page would look. The images were printed, and each mobile app screen was cut out in a similar size and shape to the average smartphone. This approach was useful because it allowed users to interact with the application and see the navigation and how the different pages relate. An incomplete list of the wireframes is displayed by Figures 3.3, 3.4 and 3.5. The profile pages in Figure 3.4 illustrate where the user will be prompted to provide the mode and schedule, and where they can later go to edit them.

Steps 6 and 7: Analyse data and report findings

During the focus group sessions, it was determined that participants engaged better with the wireframes but had a challenge engaging with the functionality flowchart as it was too abstract and technical. Therefore, a suggestion would be to present application navigation using the cut-out wireframes as well, rather than a flowchart. This will give participants a clearer understanding of the proposed application, leading to more constructive feedback

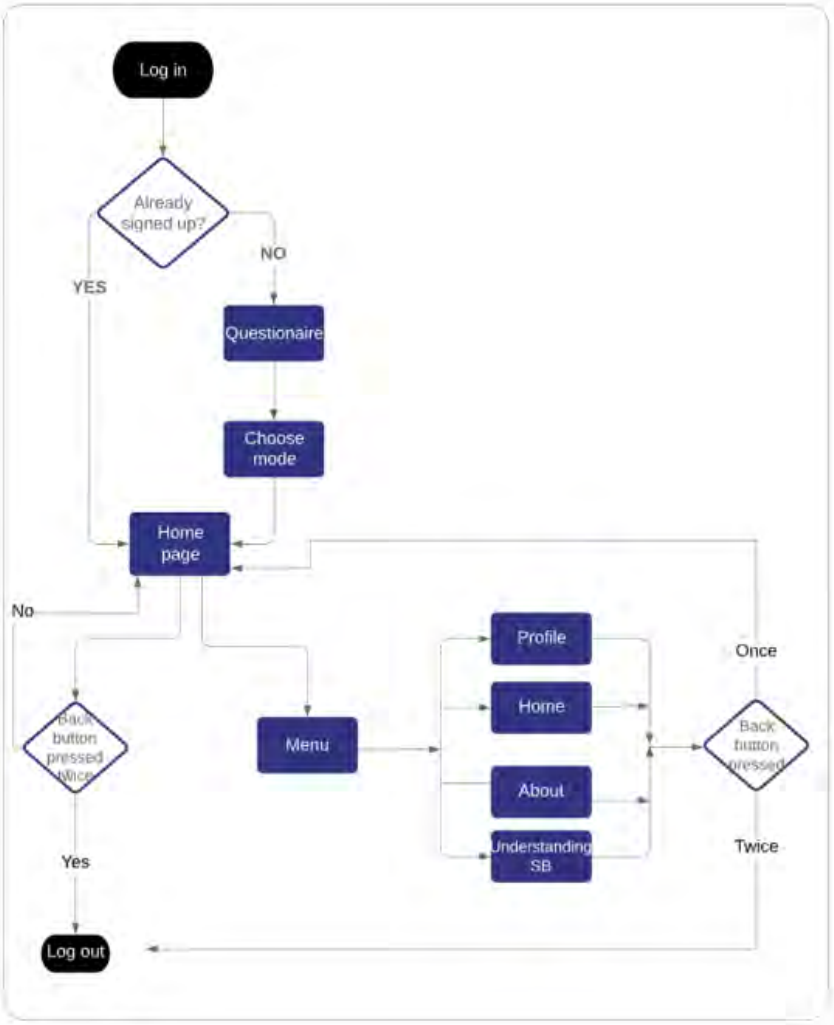

Figure 3.2: Proposed functionality of the mobile application

Figure 3.3: The proposed Sign up pages and advocacy page of the application

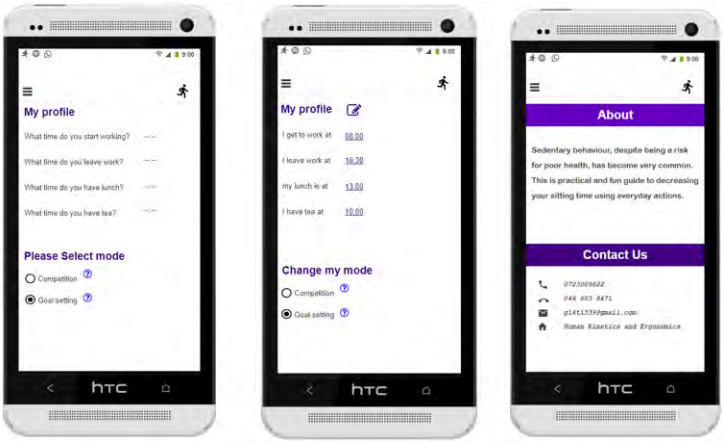

Figure 3.4: The proposed Profile pages and About Page of the application

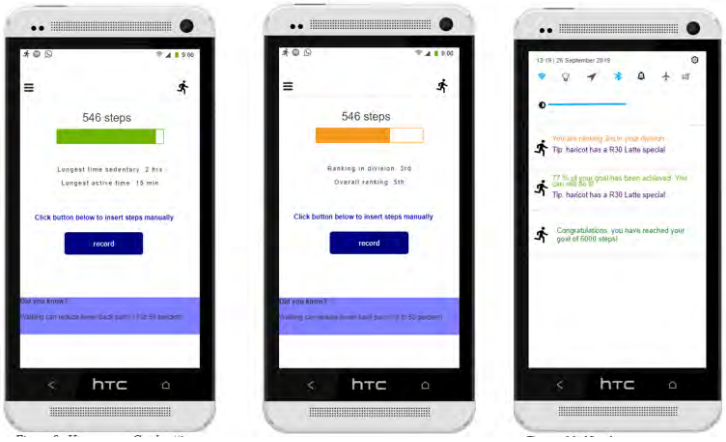

Figure 3.5: The proposed Home Pages for both conditions and the Notifications of the application

from them. At each focus group, feedback was received and used to make improvements. The idea of using tips in the notifications was received quite well, and one concern that participants had was that they do not always carry their phones around, which could make their step-count inaccurate.

Please refer to Tsaoane and Zschernack (2019) for the full report of all wireframes, and the detailed behavioural, feedback and intervention theories that informed this methodology.

Chapter 4

Method

This intervention was designed through a co-creation and co-production process where the content, functionality and wireframes of the application were designed together with the potential users of the application, as described in Chapter 3. Co-creation ensures a higher acceptability rate because user requirements can be determined and catered to. The designs were firmly rooted in behavioural change, feedback and intervention theories to increase the likelihood and effectiveness of behavioural change. However, no matter how effective the application may be in changing behaviour, it will not work if users do not have the desire to engage with it over a long enough period. This chapter used research-based software development guidelines and practices to ensure a higher acceptability and user engagement rate. Implementation of the wireframe designs into a working application was completed through an agile development process (See Figure 4.1). Agile development fits well with co-creation because it accommodates participant engagement and supports a quicker and easier adoption to change. The seven steps of the agile development process are discussed below, detailing the parts of the implementation that will be completed at each stage.

4.1 Plan

This mobile application offered a practical and enjoyable guide to decreasing occupational sitting time. The application provided feedback to users about their movement while they were at work. Their movement was determined through the device's accelerometer. The application provided two types of feedback- feedback on a user's progress based on a

Figure 4.1: The Agile Software Development Process from (D'Ambra, 2018)

goal they have set (goal-setting) and feedback on a user's progress compared to other users (social comparison). This progress was provided through three daily notifications. A user received a notification informing them of their progress, accompanied by a tip on how to increase their movement during and after office hours. The feedback and results from the focus groups were considered when planning the algorithms and code to ensure that the important requirements from the focus group process were catered for. Before implementation starts, important development guidelines and best practices for development will be discussed briefly.

Efficient use of resources The application must be written so that it does not drain a device's processing power, memory or battery. People use their devices for multiple tasks throughout the day; one application must not monopolise on the resources (Shehzaib et al., 2018; Affonso et al., 2019).

Application size There are thousands of mobile applications available on application stores. When an application requires too much storage space, it may discourage the user because device storage is limited (Shehzaib et al., 2018; Higgins et al., 2019).

Intuitive design The application's user interface must be attractive and easy to follow, and using it must not add to a user's cognitive strain. Designs on all pages must be consistent, and usage of colours, shapes and multimedia must be intentional (Baktha, 2017; Yuan and Marques, 2018).

Navigation Navigation must be simple, requiring minimal scrolling (Jayatilleke et al., 2018).

Context awareness The application must be able to adapt to a user's context. The device's location, time, Wi-Fi can be used to determine this context. For instance,

if a user is connected to Wi-Fi, the application can upload data to the cloud. When a Wi-Fi connection is not detected, data can be saved locally and only uploaded once a connection is detected (OConnor et al., 2016).

Connectivity The application must be able to work offline (Baktha, 2017).

Compatibility with older devices Developers must not only focus on current devices, as many users still use older models and operating systems. In other words, if an application is programmed for Android 12 (API level 31), it should be able to support down to Android 5.0 (API level 21) (Baktha, 2017).

Tailoring Applications have greater success when they allow users to customise them according to their own requirements and preferences (Han et al., 2020).

The challenge with this kind of implementation was to keep a good balance between theory and function. The application had to be based on as much evidence-based theory as possible without compromising these guidelines. This way, the application could have a high acceptability rate, ensuring long term use while effectively changing behaviour during this period.

4.2 Design

The wireframes for the application were designed in (Tsaoane and Zschernack, 2019) and briefly described in Chapter 3. Before coding, there were important requirements that needed to be considered and implemented into the application, as the focus group members suggested.

Use of health apps During the focus groups, it was found that 22% of users had health applications on their phones. 89% of the participant expressed an interest in using a mobile application that can decrease sedentary behaviour, and 11% said they were not sure. It was, therefore, crucial to make the application appealing to users who would normally be disinterested.

Understanding sedentary behaviour 78% of the participants indicated that they are aware of sedentary behaviour and its risks. The application, therefore, added brief info-graphics and definitions. This content was brief to avoid users having to over-scroll or being overwhelmed. However, a web-hosting service was used to provide more detailed content for users who want to read up more on sedentary behaviour.

Colour The theme colour that was decided on was the Rhodes University purple. In terms of code, this was Indigo on the colour picker or #4B0082 on the RGB scale.

Notifications 44% of the participants stated that they only stand up once or twice per day while at work. It was agreed that three daily notifications would be appropriate to encourage users to break their sitting bouts but not too many to irritate or overwhelm them.

Notification times Participants all had varying schedules- they did not all arrive at work at the same time, and all had tea and lunch at varying times, with some having lunch and no tea, some no lunch. The application accommodated this by requiring users to provide their arrival, tea, lunch, and knock-off time as illustrated by the wireframes in Figure 3.4. A generic time-picker was used for this. These times were then used to determine the time to send the three notifications. Generally, people stand up for lunch or tea but remain seated in between. The notifications were calculated to be sent in between these times. For example, if a user had tea at 10:00 and lunch at 13:00, they would get a notification at 11:30. This was achieved by calculating the average of the two times. The same was done for the other two notifications, one between arrival and tea, another between tea and lunch and the last between lunch and work departure.

Tips When tips (see Table 3.2) were presented, they were well received by participants. Tips accompanied each notification and gave participants fun ideas to break their sitting bouts. The tips were structured in a simple array and added to the notifications randomly.

Privacy concerns Several users expressed privacy concerns. At a subsequent focus group meeting, the researcher explained that the application would only require an email address and a password, and participants were comfortable with this.

Connectivity Some participants were also concerned about using the application when they did not have access to the university's Wi-Fi. Therefore, the application needed to be designed with caching and offline capabilities.

Accountability The researcher presented the 'Contact us' section of the application, which was received well as it gave users easy and convenient access to the researcher if any issues arose. The call, email and location APIs were used here to allow users to make contact directly from within the application.

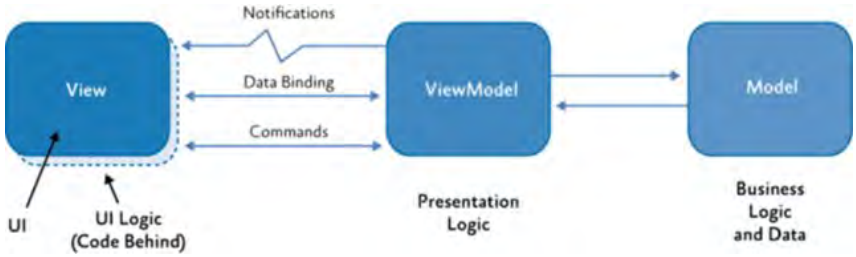

Figure 4.2: Structure of the Model MVVM (Model View / View / Model) architectural pattern (wibjorn, 2022)

4.3 Develop

As described in Section 4.3.1, the initial application was developed in c# using the Xamarin.Forms framework. Due to the challenges that the researcher experienced with the framework, a decision was made to develop a new application. This second version is described in Section 4.3.2 and was developed in JavaScript using the React Native framework.

4.3.1 Version 1: Xamarin.Forms and C#

The platform of choice to develop the application was Xamarin, created by Microsoft. Xamarin is a platform for building Android, iOS and Windows applications using the .NET framework and the C# programming language (Microsoft, 2020). The specific framework used for this project was Xamarin.Forms due to its broad developer support and cross-platform capabilities. Another benefit of Xamarin.Forms is that it provides access to the underlying Xamarin native frameworks for both Android and iOS, which gives the developer more flexibility (Microsoft, 2020).

Application Architecture

The Xamarin.Forms application used Model View View-Model (MVVM) as the design pattern. MVVM was designed to separate program logic and user interface controls. This design pattern works well with Xamarin.Forms as it allows the views (User Interface) to be independent of any particular model platform. The application followed this design pattern by abstracting the views, business logic, and data into separate namespaces “views”, “view models”, and “model”, see Figure 4.2.

User Interface Design

Xamarin uses XAML as the primary markup language to define user interface components. Syncfusion was also used to illustrate the Charts and Layout components. Charts can provide an easy way to display users' progress through visualisation. The Layout components provided simple interfaces to create and manage navigation views and routing through tabs and a fly-out menu. Syncfusion is free and has strong support from a large community of developers.

Back-end Architecture

Google's Firebase Service provided the backend. Firebase is a mobile backend as a service (MBaaS) product that Google offers to help developers quickly deploy and scale their mobile applications. While Firebase was designed to work with the native Android and Apple platforms, multiple wrapper libraries are written to work with other languages such as C#. An open-source library was used to save the models in the client application to the Firebase database. The database allowed for data persistence for users and data reporting for the researcher.

App-center is a Microsoft service that provides developers with Diagnostics and Analytics (in production). It works by using a .NET library that binds to the Xamarin application automatically. This service provides the developer with diagnostic information such as crash reports and exceptions. It also provides analytical data such as user retention, device usage and abstracted geographic data. This service helped in monitoring application performance and identifying bugs in production.

Xamarin.Forms Limitations

Xamarin.Forms is a solid platform and an excellent solution to cross-platform development. Nonetheless, several issues were encountered through different testing iterations (Testing is discussed in Section 4.4). These issues ranged from minor UI bugs to major resource consumption issues, which caused the application to be unusable for reliable data collection. These issues will be discussed in further detail in Section 6.2.

4.3.2 Version 2: React Native and JavaScript

As a result of the limitations of the Xamarin.Forms framework, a decision was made to write an new application using an alternative framework. The framework of choice was React Native, a JavaScript library that combines Native development with the React framework. React Native uses a component-based architecture to design interfaces where each component is composable. React is platform-independent as it simply describes the structure of a user interface and the data it encapsulates but does not render the user interface itself.

Development pattern and architecture

JavaScript is a multi-paradigm language that supports the functional paradigm. This differs significantly from C#, which uses the OOP paradigm. The functional paradigm was used as the primary paradigm for this application. Functional programming allows for more precise declarative code and has clear rules for working with data to avoid unintentionally mutating the state. The design architecture used for the React application is a mixture of two patterns; React's Component-Based Design and Model-View-Controller (MVC). The component-based design focuses on composing complex user interfaces using base components. In contrast, MVC focuses on dividing the logic into interconnected elements that separate internal representations of information from how information is presented to and accepted by the user. Services are added to this architecture to accommodate the application's use of sensors and functionality to run in the background and persist across reboots. This architecture is described below:

Models The models are the data containers for the application. They define the properties for the data objects and expose functions to read and update them from the local cache and the server.

Views Views are simply a related collection of components. Components are composed to make a view by encapsulating and relating the components using JSX and JavaScript definitions.

Components Components are the units used for composing UIs in React. They define small single-purpose containers that describe UI elements such as buttons, menus, or images. These are written in JSX.

Services Services are the abstraction for the logic that persists outside the application's life-cycle. The React Native application used services to access the device's sensors to count steps. This layer was also in charge of persisting the application logic on device restart by communicating with the native code.

4.4 Test and Deploy

The three main sections to test were step-counting, notifications and background tasks. A fully functional application was expected to allow the user to:

- Install it from Google Play Store onto their mobile device, install it virtually with their Google account from their computer onto their mobile device, or install it using an APK file.
- Create an account using an email address, username and password.
- Set up their daily schedule by providing arrival times, tea, lunch and work departure.
- Select their Mode (competition, goal-setting or control) and gender (male, female or other).
- Have their step-count recorded as they move. This should happen when the phone is active and sleeping.
- Navigate to the *Home* page to view their step count, and read different "did you know" facts regarding sedentary behaviour.
- Navigate to the *Sedentary behaviour* tab to read up on its definition, stats, risks and recommendations. They should also be able to click a link that will open a web page should they wish to learn about sedentary behaviour in further detail.
- Navigate to the *About* page to call, email or physically visit the researcher from the provided address.
- Navigate to the *Profile* page to modify their mode, gender, goal or schedule.
- Logout and log back into the application by providing their email and password.

Existing step-counters were used to test the accuracy of the step counter. Samsung health from a Samsung S7 and Huawei Health from the Huawei Watch Fit were used to determine the accuracy. The researcher recorded the step count reading on each platform immediately before and immediately after the test. The researcher tested the step-count accuracy during a casual walk, light jogging, running and when in the car. Notifications were tested in two different ways, first by manually changing the time on the phone and seeing if notifications arrive at the expected time. Secondly, it was done over a few days by setting up a schedule and determining if notifications arrive at expected times. Lastly, for the background tasks, the application's step-count was recorded while the application was opened. The application was then closed and later opened again and see if the was an increase in step count. The application was switched off and switched back on to determine if the step-counter would be initiated without the application being opened manually.

The application was published on the Google play store. Google play store is installed on most Android devices and allows for a one-click install which is more convenient for users. The Google play console has built-in update management, allowing for updates to be rolled out automatically, without users installing each version of the application manually. An apk file was provided for non-Samsung Android users. The researcher provided users with instructions on installing the apk and was available to help with installations in-person, should participants require it.

4.5 Review

Pilot studies were conducted to review the application. The pilot studies were in two stages: an expert tested the application in stage one, and novice users tested it in stage two. An expert in mobile application programming was approached to analyse the application for quality and feasibility. This test took the form of unstructured interviews. Ten participants were recruited through word of mouth, and social media and each participant installed the application and used it for a week. Each participant was required to share their experiences with the application throughout the week. After that week, all feedback was collected on usability and bugs that need to be fixed. The application was improved and modified based on the feedback.

Pilot Testing took nine iterations, with each iteration addressing a different issue from the expert or novice users. Figure 4.3 is a screenshot from the Google Play Console, which

Version code	Released	Status
6	Sep 3, 2021 5:22 AM	Replaced on Sep 21, 2021 8:14 PM
5	Sep 3, 2021 12:55 AM	Replaced on Sep 3, 2021 5:22 AM
4	Jun 2, 2021 12:14 AM	Replaced on Sep 3, 2021 12:55 AM
3	Apr 28, 2021 9:39 PM	Replaced on Jun 2, 2021 12:14 AM
2	Apr 27, 2021 10:54 PM	Replaced on Apr 28, 2021 9:39 PM

Figure 4.3: Screenshot from the Google Play Console of some of the different iterations of the pilot studies

illustrates some of the pilot test iterations. Once the application testing was completed, it was published for pilot testing. Feedback from this test was then collected for analysis. The researcher then went back to implement this feedback, and another pilot test was conducted. Some of the feedback from the pilot studies was UI based, while others were functionality based. After the last pilot test, where participants did not face any more issues with the application, the application was launched.

4.6 Launch

Aim of pilot implementation

Once the software development process was complete, the application was launched to determine its feasibility as an intervention to decrease sedentary behaviours through goal-setting and social comparison. The pilot implementation test group consisted of office-based employees, as this is the context in which is it intended to be used. Employees used the application in one of three modes, competition, goal-setting or control, and their recorded step counts were later analysed and compared.

Study design

The application's feasibility was tested in a field study setting to ensure that participants can, as much as possible, go about their regular work schedules. The participants were allocated to the different groups through strategic sampling. The Global Physical Activity

Questionnaire (GPAQ) was used to determine users' sedentary behaviour and levels of physical activity (WHO, 2022a). The questionnaire was distributed using a Google form for users' convenience and ease of analysis. Participants with similar GPAQ scores were placed in the same group for the social comparison mode to ensure that competition is at a relatively equal level. The researcher also ensured that the control and goal-setting groups had participants with varying GPAQ scores to minimise the possibility that participants' sedentary behaviour and fitness level would be a factor. Three groups tested the application in total; one group for each type of feedback, and the third group was the control group. The control group recorded their movements, but they did not receive any feedback. The control group did, however, receive notifications containing only tips to ensure that the difference in step-count results is due to the type of feedback (social comparison or goal-setting) and not due to the tips themselves.

Participants characteristics

The target adopters of this study were office-based workers at Rhodes University and, more specifically, those who work in the divisions of the Rhodes Registrar. Participants characteristics:
a) Inclusion criteria: Employees whose primary task is desk work, secretarial and other admin related work.
b) Exclusion criteria: Any medical conditions that might impair walking

Procedure and instrumentation

Recruitment was done through the Rhodes University mailing list. Support (e.g. endorsement and promotion) from an organisation can be beneficial for the success of program implementation and long term success (Brakenridge et al., 2016). Visible support of the program (even though it targets individuals) by the institution can increase participant numbers. The study was advertised (see Figure 4.4), outlining the details and aim. Twenty-five participants were interested and sent the researcher an email. The researcher responded to each potential participant with further details. First, all participants were sent consent forms to complete, and they could fill in the forms digitally or print the forms out and fill them in. Next, they were sent the GPAQ and required to complete it through Google forms; 17 participants submitted the forms. Due to the Covid-19 pandemic, participants were sent work location documents to determine if they work from the office, home or both. This information was then used to determine if a user's location would

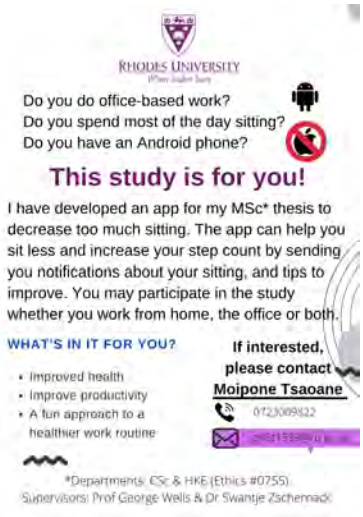

Figure 4.4: The advert that was sent through a mailing list to recruit participants for the study

affect their sedentary behaviour and how well the application can work to decrease it. Lastly, all participants were provided with a link to the application on Google Play (and an apk file provided) and requested to install the applications onto their devices. A total of 12 participants installed the application. For the Xamarin application, the mode was selected from the Firebase side by the researcher. For React Native, an email was sent where participants were instructed which mode to select. With both apps, participants in the goal-setting group were instructed to navigate to the *Profile* page to set a goal for themselves.

For the Xamarin App, the 12 participants were given a week to familiarise themselves with the application. In that week, they could contact the researcher regarding any issues they were facing with the application. Testing took one month.

Due to the limitations of Xamarin, the researcher decided to write a new application with a different platform. The second application's launch time was much later as the researcher required time to familiarise themselves with React Native and JavaScript. As such, the duration for the React native application was a minimum of one week for some participants and up to five weeks for other participants. At that stage, not all participants were able to join the study, and there was a total of 8 participants. Due to the small sample size, a decision was made to divide participants between goal setting and competition and leave out control.

Ethical considerations

All participants participated voluntarily and were able to opt out of the study at any time. All participants were required to install the application onto their phones and have their phones periodically send data to the researcher. The identity of each participant was concealed, and the data, which included personal information, was not used in the research paper. Potential benefits were that it offered a possible solution to the problems that many office-based workers faced due to sedentary behaviour. Society can benefit from this study as it provides an inexpensive and accessible solution to decreasing sedentary behaviour. Ethical approval was extended from the honours project (Tsaoane and Zschernack, 2019), with reference number 0755.

Data processing

The functionality of the different features of the application, feedback from participants and the data from users were examined to determine the feasibility of the application. To determine whether the collected data was reliable, a two-factorial analysis of variance was be used (to determine whether there is a significant difference between the means of the two populations; the goal-setting population and the social comparison population).

Chapter 5

Results

This study was conducted to determine the best way to implement a digital behavioural change intervention with a high user acceptability rate while effectively changing behaviour. According to research, the best way to go about this kind of implementation is through finding a balance between sufficient use of theory and following software development guidelines. A sound theoretical basis is as important as the digital functionality (Fulton et al., 2018). This study is deeply rooted in behavioural change and feedback theories and follows an agile software development process; implementation is done in iterations, each with a pilot test to ensure a participatory approach. Iterative pilot studies were conducted, the final pilot being with office-based workers at Rhodes University, where they were divided into groups to test the application's feasibility. Two applications were built during the implementation process. Section 5.1 details the first application, which was implemented in C# using the Xamarin.Forms framework. Due to its limitations, a second application was implemented in JavaScript using the React Native framework (Section 5.2). Section 5.3 outlines the process of going from an intervention design to a functioning application.

5.1 Xamarin.Forms Implementation

5.1.1 Application Architecture

To build the Xamarin application, the MVVM architecture was first structured. Views contained all the necessary files to define the User Interface and, therefore, mainly contained XAML files and the C# code-behind files, which would bind the view-models. The

```
22          public static void Initialize()
23          {
24              Barrel.ApplicationId = "com.companyname.dbci";
25              Barrel.EncryptionKey = encryptionKey;
26              BarrelUtils.SetBaseCachePath(Path.Combine(FileSystem.AppDataDirectory,"dbci")); // Set data file path to persist data longer than caching
27              currentBarrel = Barrel.Current;
28          }
29
```

Figure 5.1: This method was used to initialise the cache for easier, faster access.

view-model is where all the business logic was contained and included the binding logic to update the views using data from the models. Additionally, the view-model updated the models based on the user's input and the defined business logic. The models are simply data container objects meant to hold the properties of the data, such as the user profile data and step count. This application saves the data to a database, requiring a database encapsulation layer. However, as this is a small application designed to work with Google's Firebase as the back-end, the encapsulation layer has been combined with the data layer. The two significant data structures in this project were Step counter and Cache classes. The Step counter class is the model responsible for encapsulating the user's steps. It calls upon the step counter service to initialise itself and start counting steps using the device's sensors. The cache is responsible for caching all data locally for easier, faster access. It saves any user data type by serialising it to JSON and indexing it using a key. The data can then be fetched from the cache by providing the index key and deserialising it to the appropriate data type. Figure 5.1 shows the method that was written to initialise the cache.

Accessing mobile device sensors and running tasks in the background requires a service. A *services* namespace was created to abstract the creation and management of these services as they exist outside of the application's lifecycle. The services ran on a separate process to the main application and provided an interface to expose itself. The main application then used the exposed interface to interact with the services to obtain needed data. These primary services formed the core of the application and business logic:

Step Counter Service This service would access the native device's accelerometer. An accelerometer measures a device's position in 3D space. It is a standard sensor on most smartphones. After reading the values from the accelerometer, the service calculated a user's steps.

Notifications Service This service managed feedback that was sent to the user in the form of notifications. The notification service read a user's schedule and mode

from their profile and used them to determine the appropriate time to send them notifications and the type of feedback.

Foreground Service This service ran in the foreground and formed the base of the application's services lifecycle. It ensured that the user's step count would be resumed if the device restarted and that the context of the application was not lost even if the primary view was destroyed.

Business Logic and Services

This application's business logic is mainly in the view-models namespace. The view-models hold controller logic that is responsible for updating the model based on the rules defined by the business logic. They also contain data binding and commands exposed to the user interface. This layer holds all the logic that allows the user interface to communicate with the services. Services were the central part of this application as the view models relied on data that the services would stream to update the UI and the data models. The models and the database encapsulation layer were combined into one layer. Any updates to the local data on the user's device would be automatically pushed to the server and vice versa. This worked well for the context of this application as the user's steps were recorded and shared in real-time. A repository pattern (Figure 5.2) was used to keep the credentials of the user securely stored on the device to preserve their session across app restarts, and to ensure that the local cache and server were in sync at all times.

5.1.2 Application pages and navigation

The Application's navigation was built next. The overall navigation was through a flyout menu. Once the user created an account, they were directed to the profile page to save their schedule, mode and gender (the flyout menu was disabled at this stage). Once the user saved their profile, they were directed to the home page, and the flyout was enabled. From here, the user could navigate to the *About*, *Sitting* or *Profile* pages as well as exit or log out of the application. The App.XAML (Figure 5.3) contained all styling for the application and was used by all pages to ensure a consistent design throughout.

The *Sitting* and *About* pages were created next. The *Sitting* page provided the users with four tabs that displayed the definition, statistics, risks, and recommendations of sedentary behaviour. This was for users who wished to learn more on the topic. A separate web

```
12      public class LocalRepository : IUserRepository
13      {
14
15          public Task DeleteUserAsync()
16          {
17              Cache.Remove("user");
18              return Task.CompletedTask;
19          }
20
21          public Task<(UserInfo userInfo, FirebaseCredential credential)> ReadUserAsync()
22          {
23              var userString = Cache.Get("user");
24              var user =  JsonConvert.DeserializeObject<FirebaseUser>(userString != null? userString : "");
25              var userInfo = user?.UserInfo;
26              var credential = user?.Credential;
27              return Task.FromResult((userInfo, credential));
28          }
29
30          public Task SaveUserAsync(Firebase.Auth.User user)
31          {
32              var userString = JsonConvert.SerializeObject(new FirebaseUser(user.Info, user.Credential));
33              Cache.Add("user", userString, 0);
34              return Task.CompletedTask;
35          }
36
37          public Task<bool> UserExistsAsync()
38          {
39              return Task.FromResult(Cache.isActive("user"));
40          }
41
```

Figure 5.2: Repository pattern to preserve user credentials when the application restarts.

```
 9          <ResourceDictionary>
10              <Color x:Key="HeaderLight">#ffffff</Color>
11              <Color x:Key="ThemeDark">#4B0082</Color>
12              <Color x:Key="ThemeLight">#ffffff</Color>
13              <Color x:Key="HeaderDark">#4B0082</Color>
14              <Color x:Key="DYK">#EE82EE</Color>
15              <Color x:Key="Link">#0000cd</Color>
16              <Color x:Key="Body">#000000</Color>
17              <Color x:Key="ButtonBorder">#C0C0C0</Color>
18              <Color x:Key="ButtonBG">#ffffff</Color>
19          </ResourceDictionary>
```

Figure 5.3: A snippet from the App.XAML file which contained all the styling (colours, fonts, themes etc) for the application. The file is written in XAML.

```
78          public ChartSeries GetChartSeries()
79          {
80              UserProfile userProfile = UserProfile.FetchUserProfileAsync().Result;
81              if (userProfile == null)
82                  return null;
83              if (userProfile.UserMode == "Competition")
84              {
85                  ColumnSeries columnSeries = new ColumnSeries();
86                  columnSeries.XBindingPath = nameof(Model.Month);
87                  columnSeries.YBindingPath = nameof(Model.Target);
88                  LoadColumnChartData();
89
90                  Device.BeginInvokeOnMainThread( () => {
91                      columnSeries.SetBinding(ChartSeries.ItemsSourceProperty, "Data");
92                      columnSeries.ItemsSource = Data;
93                  });
94
95                  return columnSeries;
96              }
97              else if (userProfile.UserMode == "Self")
```

Figure 5.4: Conditional rendering to display relevant chart to the user

page was made to limit scrolling and cognitive overload that contained further details on the topic. This page was hosted through Firebase Hosting, and users could read it in their browser. The *About* page offered users a way to contact the researcher directly from within the application, should they have any queries. The Xamarin.Essentials APIs (phone dialer, email, and maps) were utilised to achieve this.

The *Home* page visualised the user's movements through a doughnut chart for goal–setting and bar chart for social comparison. The charts were implemented using Syncfusion, and conditional rendering (Figure 5.4) was implemented to display the relevant graph to a user depending on their selected mode.

Below the graph, users were shown brief 'Did you know' facts regarding sedentary behaviour; this was implemented as a simple array (Figure 5.5), and a single fact was fetched randomly and displayed at a time.

The *Profile* page allowed users to save and modify their schedule, mode and gender onto the Firebase Realtime Database. Code to save a user's profile is displayed by Figure 5.6

5.1.3 Firebase

To find the user on the Database, their email was first retrieved locally with *Email = new LocalRepository().ReadUserAsync().Result.userInfo.Email*, line 89 in Figure

```
129         public void SelectDYK()
130         {
131             string title = "Did you know? \n";
132             List<string> DYKlist = new List<string>()
133                             {title+"Walking can significantly reduce the risk for cardiovascular disease."
134                             ,title+"Walking works many major muscles in the body."
135                             ,title+"Varying your pace can make walking fun."
136                             ,title+"Varying the walking route can make walking fun."
137                             ,title+"A pedometer is a motion sensing device used to count steps."
138                             ,title+"There's been a 30% drop in physical activity over the last two generations."
139                             ,title+"Office based workers spend more than ¾ of their day sitting."
140                             ,title+"Walking can improve our mental health."
141                             ,title+"Walking is good for weight management."
142                             ,title+"Walking can reduce the risk of cognitive decline."};
143             Random rnd = new Random();
144             int item = rnd.Next(DYKlist.Count);
145             DYKLine = DYKlist[item];
146         }
```

Figure 5.5: Array of facts about sedentary behaviour

```
71          public async Task SaveToDB()
72          {
73              UserDialogs.Instance.ShowLoading("", MaskType.Gradient);
74              try
75              {
76                  Schedule schedule = new Schedule()
77                  {
78                      Arrive = Arrive,
79                      Tea = Tea,
80                      Lunch = Lunch,
81                      Leave = Leave
82                  };
83
84                  UserProfile userProfile = new UserProfile()
85                  {
86                      UserSchedule = schedule,
87                      UserMode = Usermode,
88                      Gender = Usergender,
89                      Email = new LocalRepository().ReadUserAsync().Result.userInfo.Email,
90                      StepGoal = this.StepGoal,
91                      UserGroup = userGroup
92                  };
93
94                  userProfile.saveToDB();
95                  SetUserProfileCreated();
96
97                  NotificationService.SetNotificationSchedule(schedule);
98              }
```

Figure 5.6: Save user data to the RealTime Database

```
115         public async Task Register()
116         {
117             if (string.IsNullOrEmpty(RegEmail) || string.IsNullOrEmpty(RegPassword)||string.IsNullOrEmpty(RegUsername) || string.IsNullOrEmpty(RegConfPassword))
118             {
119                 await UserDialogs.Instance.AlertAsync("Please fill in all entries and try again", "Empty values", "OK", null);
120             }
121             else try
122             {
123                 UserDialogs.Instance.ShowLoading("Loading", MaskType.Gradient);
124 
125                 if (!RegEmail.Contains("@") || !RegEmail.Contains(".")) throw new Exception("Invalid email format, please try again");
126                 else if (RegPassword.Trim().Length <6)
127                     throw new Exception("Weak password, please try again with a stronger password");
128                 else if (!RegConfPassword.Trim().Equals(RegPassword.Trim()))
129                     throw new Exception("Confirmation password does not match password");
130 
131                 await FirebaseAuth.getAuthClient().CreateUserWithEmailAndPasswordAsync(RegEmail.Trim(), RegPassword.Trim(), RegUsername.Trim());
132                 UserDialogs.Instance.HideLoading();
133                 Application.Current.MainPage = new ProfilePage();
134             }
```

Figure 5.7: Code to create a new user on Firebase.

```
75          public async Task Login()
76          {
77              if (string.IsNullOrEmpty(RegEmail) || string.IsNullOrEmpty(RegPassword))
78              {
79                  await UserDialogs.Instance.AlertAsync("Please enter a valid email and password", "Empty values", "OK", null);
80              }
81              else try
82              {
83                  UserDialogs.Instance.ShowLoading("Loading", MaskType.Gradient);
84                  await FirebaseAuth.getAuthClient().SignInWithEmailAndPasswordAsync(RegEmail.Trim(), RegPassword.Trim());
85                  UserDialogs.Instance.HideLoading();
86                  Application.Current.MainPage = new HomePage();
87              }
```

Figure 5.8: Code to log in to the application.

5.6. The email is initially saved when the user creates an account (the profile page is only exposed once a user is signed into the app). Firebase Authentication is used to create accounts. Users are required to provide an email address, username, password and confirm the password (see Figure 5.7). The authentication code was also accompanied by exception handling, which was offered by *FirebaseAuthException* and provided checks to determine if the user already exists, if there was an HTTP error and so on.

Once a profile is created, users could log in using their email and passwords as shown in Figure 5.8, which was also accompanied by relevant exception handling through *FirebaseAuthException*. For their convenience, users remained logged in when they closed the application or switched off their devices. They were only be logged out of the application if they clicked the *logout button*.

For efficiency, data validation and exception handling were done from the client-side. This way, the application could inform the user of a password being too short, too weak or an email address format being incorrect without first attempting to create a profile on Firebase. Figure 5.9 shows the format shown to the user without client-side validation,

while Figure 5.10 illustrates the format after client-side validation.

Figure 5.9: Error format when data is validated by Firebase.

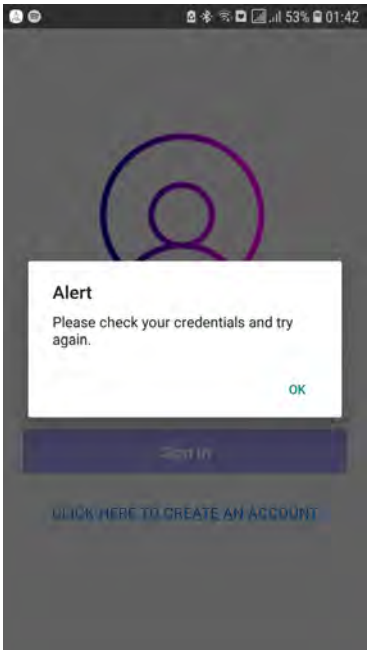

Figure 5.10: Error format when data-checks are handled from the client side

If a user forgot their password once they have logged out of the application or after reinstalling the application, they could contact the researcher for a new password. Firebase enables this functionality through a password reset link (Figure 5.11), which is sent to the user's email address. Figure 5.12 lists the other options that are related to this functionality, showing that Firebase can enable users to change their email as well and can also verify their credentials through both email and SMS.

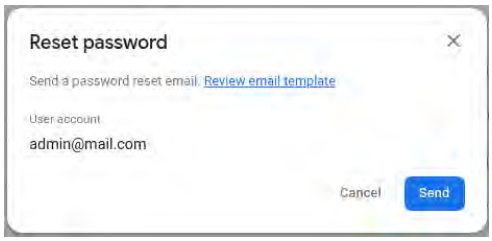

Figure 5.11: This functionality is provided from within the Firebase console and enables researchers to send password reset links to users when necessary.

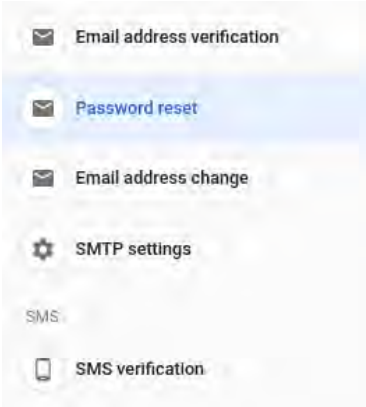

Figure 5.12: Firebase allows developers to assist users with their authentication through these approaches

5.1.4 Step-counter in Xamarin.Forms

The step counter was first created using an algorithm that accessed a device's pedometer sensor. This step counter worked quite well and was accurate when compared to the Samsung Health step counter. During one iteration of the pilot study, it was discovered that the step-counter worked very well on some devices and did not work at all on others. The step-counter code was then rewritten, using an algorithm from Github (Bagi, 2022) which used the accelerometer's changes in a 3D space while considering the effects of the gravitational force. Snippets of the algorithm variables and calculations are shown in Figure 5.13 and Figure 5.14. Figure 5.13 shows how initial values are set to calibrate the sensor and account for the force of gravity, while Figure 5.14 shows the calculations that were performed to get the step count. After testing the step-counter as discussed in Chapter 4.4, the sensitivity that worked best was 80.50f, which can be seen in line 48 of Figure 5.13.

5.1.5 Foreground tasks

Foreground tasks were necessary to ensure that the step-counter continued to record and share the user's step-count even when the application was closed (Figure 5.15). When the user's device died or was switched off, this foreground service initiated itself once the device was switched back on, without the users having to open the application (Figure

```
39          public AccelerometerStepDetector()
40          {
41              int h = 480;
42              mYOffset = h * 0.5f;
43              mScale[0] = -(h * 0.5f * (1.0f / (SensorManager.GravityEarth * 2)));
44              mScale[1] = -(h * 0.5f * (1.0f / (SensorManager.GravityEarth)));
45              OnStepTaken += stepTaken; //subscribe to step event
46              this.OnSensorChangedAction = SensorChange;
47
48              setSensitivity(84.50f); //adjusting sensitivity after personal testing
49          }
50
51          public void setSensitivity(float sensitivity)
52          {
53              mLimit = sensitivity; // possible threshold values: 1.97  2.96  4.44  6.66  10.00  15.00  22.50  33.75  50.62
54          }
55
```

Figure 5.13: Values set up to calibrate the accelerometer sensor.

```
67                      float vSum = 0;
68                      for (int i = 0; i < 3; i++)
69                      {
70                          float x = mYOffset + e.Values[i] * mScale[j];
71                          vSum += x;
72                      }
73                      int k = 0;
74                      float v = vSum / 3;
75
76                      float direction = (v > mLastValues[k] ? 1 : (v < mLastValues[k] ? -1 : 0));
77                      if (direction == -mLastDirections[k])
78                      {
79                          // Direction changed
80                          int extType = (direction > 0 ? 0 : 1); // minumum or maximum?
81                          mLastExtremes[extType][k] = mLastValues[k];
82                          float diff = Math.Abs(mLastExtremes[extType][k] - mLastExtremes[1 - extType][k]);
```

Figure 5.14: A snippet of the step count calculations.

```
34              if (isRunning)
35                  return StartCommandResult.Sticky;
36
37              var pendingIntent = PendingIntent.GetActivity(Android.App.Application.Context, 0, intent, PendingIntentFlags.UpdateCurrent);
38
39              var notification = new Notification.Builder(Android.App.Application.Context, foregroundChannelId)
40                  .SetContentTitle("Step Up")
41                  .SetContentText("Step Counter is Active")
42                  .SetSmallIcon(Resource.Drawable.Radio)
43                  .SetContentIntent(pendingIntent)
44                  .SetOngoing(true);
```

Figure 5.15: Part of the code that allows the application to persist

```
37              var serviceIntent = new Intent(ApplicationContext, Java.Lang.Class.FromType(typeof(StepDetectorService)));
38              serviceIntent.PutExtra("Title", "Message");
39              if (Android.OS.Build.VERSION.SdkInt >= Android.OS.BuildVersionCodes.O)
40              {
41                  ApplicationContext.StartForegroundService(serviceIntent);
42              }
43              else
44              {
45                  ApplicationContext.StartService(serviceIntent);
46              }
```

Figure 5.16: This code ensured that the application's step counter starts up automatically

5.16). It was also important to ensure that users received notification even when the application was closed. Notification channels, displayed by Figure 5.17 were created to enable this.

5.1.6 Notifications in Xamarin.Forms

Figure 5.18 shows the code to compose the notification message depending on a user's mode. Simple calculations were performed of the midpoints between their scheduled times to determine when a user receives notifications. This calculation allowed notifications to

```
48              if (global::Android.OS.Build.VERSION.SdkInt >= BuildVersionCodes.O)
49              {
50                  NotificationChannel notificationChannel = new NotificationChannel(foregroundChannelId, "Title", NotificationImportance.High);
51                  notificationChannel.Importance = NotificationImportance.High;
52                  notificationChannel.EnableLights(true);
53                  notificationChannel.EnableVibration(true);
54                  notificationChannel.SetShowBadge(false);
55
56                  var notifManager = Android.App.Application.Context.GetSystemService(Context.NotificationService) as NotificationManager;
57                  if (notifManager != null)
58                  {
59                      notification.SetChannelId(foregroundChannelId);
60                      notifManager.CreateNotificationChannel(notificationChannel);
61                  }
62              }
63
64              StartForeground(ServiceRunningNotifID, notification.Build());
```

Figure 5.17: A notification channel to enable the application to send notifications.

```
86              NotificationCenter.Current.NotificationReceived += Current_NotificationReceived;
87
88              string msg1, msg2, msg3, extraInfo = "";
89
90              if (UserProfile.FetchUserProfileAsync().Result.UserMode == "Self")
91              {
92                  var percentage = UserProfile.GetStepGoalProgressPercentage().Result;
93                  extraInfo = $"\n{percentage}% of your goal has been completed. Keep it up!";
94              }
95              if (UserProfile.FetchUserProfileAsync().Result.UserMode == "Competition")
96              {
97                  var rank = UserProfile.GetGroupComparisonRank().Result;
98                  extraInfo = $"\nYou are ranked {rank} in your division. Keep it up!";
99              }
100
101             msg1 = earlierTips[new Random().Next(0, earlierTips.Length)];
102             do
103             {
104                 msg2 = earlierTips[new Random().Next(0, earlierTips.Length)];
105             }
106             while (msg2.Equals(msg1));
107             msg3 = laterTips[new Random().Next(0, laterTips.Length)];
```

Figure 5.18: Notification message based on user's mode (goal-setting or social comparison).

```
181                 DateTime today = DateTime.Now;
182                 DateTime tomorrow = today.AddDays(1);
183                 while (tomorrow.DayOfWeek == DayOfWeek.Saturday || tomorrow.DayOfWeek == DayOfWeek.Sunday)
184                 {
185                     tomorrow.AddDays(1);
186                 }
187                 DateTime midnight = new DateTime(tomorrow.Year, tomorrow.Month, tomorrow.Day, 0, 0, 0);
188
189                 DateTime midArrivalTea = new DateTime(midnight.Ticks + TimeSpanMidpoint(schedule.Arrive, schedule.Tea));
190                 DateTime midTeaLunch = new DateTime(midnight.Ticks + TimeSpanMidpoint(schedule.Tea, schedule.Lunch));
191                 DateTime midLunchLeave = new DateTime(midnight.Ticks + TimeSpanMidpoint(schedule.Lunch, schedule.Leave));
```

Figure 5.19: This code schedules the three daily notifications

be sent between arrival and tea, tea and lunch, and finally, between lunch and work departure. (The notification times were calculated by the code shown in Figure 5.19). As seen in Figure 5.17, the notifications are integrated with the foreground service to ensure that they are sent even when the application is not open. The *SendNotification* function (Figure 5.20) was called to send these three notifications. Notifications were sent from Monday to Friday, and at the end of each day, notifications reset themselves for the next day. The visual outcome of the notification is shown in Figure 5.44.

```
215                 SendNotification(msg1, notificationIdCounter++, midArrivalTea);
216                 SendNotification(msg2, notificationIdCounter++, midTeaLunch);
217                 SendNotification(msg3, notificationIdCounter++, midLunchLeave, "Final");
```

Figure 5.20: Code to send the three scheduled daily notifications

Report history

Artifact	Stability	Performance	Accessibility	Security and trust	Tests completed	Report generated
7	5	3	5	0	6	Sep 21, 2021, 20:14
6	4	4	5	0	6	Sep 3, 2021, 05:22
5	3	0	0	0	6	Sep 3, 2021, 00:55
4	3	4	5	0	5	Jun 2, 2021, 00:14
3	3	2	4	0	3	Apr 28, 2021, 21:39
2	1	0	0	0	3	Apr 27, 2021, 22:54
1	3	0	0	0	3	Apr 24, 2021, 23:03

Figure 5.21: The seven pilot study testing iterations

5.1.7 Application feasibility

The final version of the application was Version 9.0 and followed eight iterations of pilot testing. Seven of these iterations were conducted with the Xamarin.Forms application. The application had several limitations that led to crashes and incorrect functioning on some devices. After the seven pilot study testing iterations as seen in Figure 5.21, it was determined that the application was not feasible—the red exclamation icons indicate fatal issues with the application. The significant issues from the Xamarin.Forms application will be discussed.

Step counting issue The first step-counter code worked very well on some devices and did not work at all on others. The second step counter was often inaccurate when compared to Samsung Health's step counter.

Black screen on navigation This UI bug caused the entire screen to go black and caused the application to become unresponsive to user input. The bug occurs randomly, and there is no way to tell when it might happen.

Resource consumption This bug caused the application to drain a user's battery or mobile data drastically.

Notifications bug For the application to provide user feedback, it used notifications. These notifications were sent on a pre-determined schedule calculated from the user's profile data. Xamarin.Forms contained a bug that would prevent these notifications from being displayed to the user as the notifications timer would stop working after an undetermined amount of time.

Table 5.1: Data captured for the Xamarin.Forms application, which shows inconsistent data due to issues with the application

	ID	P17	P1	P2	P5	P6	P9	P10	P11	P12	P14	P15	P16
6-Sep	Mon				1664	1294		14	1410	1352	5740		
7-Sep	Tue				3360	675	989	71	15994	15	1544		
8-Sep	Wed	13	1048	22	937	1288	1694		6680		161		1469
9-Sep	Thu	566	930		3418	3840	2837	14	1948		2505		6275
10-Sep	Fri	377	2781		95	889	2157	71	4831	499	1949		3050
13-Sep	Mon	820			1003	13			1769		776		3608
14-Sep	Tue	203			1158	827			1197		1090		3929
15-Sep	Wed	158		507	1157	1074			4551		85		5101
16-Sep	Thu	4	5955	513	3141	882			4199		1310		2603
17-Sep	Fri			253	1388	177			2402		1443		3576
20-Sep	Mon	1			1172	429	407		2		367		5226
21-Sep	Tue	889			1807	1029	1638		6520	226	3225		3805
22-Sep	Wed	38			1355	988	2382		2298	8	3579		4462
23-Sep	Thu				103	1980	414		2745		1766		3206
24-Sep	Fri					91			8229		769		1895

Chart rendering This bug is related to the UI component used to display the graphs and charts to the user. At times, the application could not render the charts to the home screen, and other times it did.

As mentioned, this application had quite a few issues; the challenge was that these issues were only on certain devices and at certain times, and there was no clear pattern. Table 5.1 displays the gaps in data due to crashes and inconsistencies with the application. Participants P16's application was consistent and quite accurate, while the values for participant P10 show the opposite: the application was inconsistent and the low values were inaccurately measured. Participant P15, who used an older phone model, could not use this application as it consumed too much of their phone's resources, especially battery and mobile data.

5.2 React Native and JavaScript

A new application was developed from scratch to respond to the issues discussed above. The application was hooked to the existing Firebase project by downloading a *google-services.json* file from Firebase and adding the API key to the project. This means that users could use the same authentication credentials on the React Native application as they did on the Xamarin.Forms. The critical features in this section are notifications and the step-counter; the general features of this second application will not be discussed in detail as similar concepts from the first application were used but with different code.

5.2.1 Application Architecture

The application was made up of four core aspects: Model, Components, Views and Services. The components were created using a Style-sheet (based on the traditional Cascading Style Sheets) to ensure a consistent design throughout the application. Each View (page of the application) had access to these components and imported relevant ones to build the look and functionality of that page. The components and views contained are illustrated by Figure 5.22 and Figure 5.23. In Figure 5.22, Header.js is the purple line below the page heading on each page, AuthButton.js is used to the buttons of the sign-in and registration pages. The name of the files in Figure 5.23 represent the page, for instance HomeView.js is the home page of the application where the user can view the step-count. MainTabView.js is the four tabs that display the definition, statistics, risks and recommendations of sedentary behaviour (these tabs are displayed in Figures 5.34, 5.35 and 5.36).

Figure 5.22: Components of the React Native application

Figure 5.23: Views of the React Native application

React Native has a different approach to handling data and states. The step counter of this second application has a different algorithm but follows the same principles as the first application since they both used the same underlying Operating System. The module accessed the device's sensors through a library that exposed the interface. The project used asynchronous storage, (Figure 5.24), to cache local data for easier and faster access. Asynchronous storage is faster than synchronous storage and allows the application to be responsive to the user even while loading data in the background.

```
import AsyncStorage from '@react-native-async-storage/async-storage';

export async function GetUserFromStorage() {
  const userData = await AsyncStorage.getItem('user');
  console.info(`fetching user data\n ${userData}`);
  return JSON.parse(userData);
}

export async function GetDisplayNameFromStorage() {
  return AsyncStorage.getItem('displayName');
}
```

Figure 5.24: Caching system for the React Native application

```
      const config = {
        default_threshold: 15.0,
        default_delay: 150000000,
        cheatInterval: 3000,
        onStepCountChange: stepCount =>
          (async () => {
            console.info('Step count changed ' + stepCount);
            const nowDate = new Date();
            const [nowDay, nowMonth] = [nowDate.getDate(), nowDate.getMonth()];
            if (nowDay !== day || nowMonth !== month) {
              await RestartCounter();
              return;
            }
            await handleStepCountChange(stepCount, sessionPath);
          })(),
```

Figure 5.25: Step counter algorithm from Shukla and Chauhan (2022)

5.2.2 Step counter

The step counter code (Figure 5.25) was found in Shukla and Chauhan (2022). The reading was quite accurate when compared to the Samsung Health mobile application and the Huawei Watch.

5.2.3 Foreground services

Similarly, foreground services were crucial for the React Native application to ensure that the notifications were sent and the step-counter continued to record even when the application was not open. The step counter service (Figure 5.26) and notification service (Figure 5.27) were used to interface with the device outside of the application's lifecycle. In Figure 5.26, the *handleStepCountChange* function was called every time a user took a step. It is a stateless function that takes the step-count and data for the current session and saves it for the user at the right place. If it did not find a profile, it saved the state and incremented it to a user's steps once they had completed creating a profile. For increased

accuracy, JavaScript's *Backgroundtimer* was run every 60 seconds to check if it was time to send a notification to the user. It is worth noting that despite the check being done every 60 seconds, the JavaScript React engine remained efficient (there was no noticeable increase in resource consumption).

5.2.4 Notification in React Native

The messages sent to users through notifications were composed using the code in Figure 5.28 and the code in Figure 5.29 was written to determine when these notifications would be sent. The Notifee API from Diarmid et al. (2022) was used to format and send the notifications. This API simplified and abstracted the notification channel, behaviour and appearance as seen in Figure 5.30. The visual outcome of the notification is in Figure 5.45.

5.2.5 Implementation outcome

The application was published on the Google Play Store. Users with Play Store on their devices could download it from there or download it from their computers if they were logged in with a Google account. An apk file was made available for users whose devices do not support the Play Store. This section displays different screens that formed part of the application.

Applications layout

The main menu for this application was a fly-out menu (Figure 5.31), which allowed users to navigate between *Home*, *Profile*, *Sitting* and *About* pages. The user was required to create an account or log in to access the menu. Figure 5.32 shows the screen where a user could enter their details to create an account. Profile creation required a schedule, Figure 5.33 displays the screen where users could select their times for the schedule.

```
async function handleStepCountChange(stepCount, sessionPath) {
  let displayName;
  let isProfileCreated;

  displayName = displayName ? displayName : await GetDisplayNameFromStorage();
  if (!displayName) {
    return;
  }
  isProfileCreated = isProfileCreated
    ? isProfileCreated
    : await GetProfile(displayName);

  steps = stepCount;
  const prevSteps = await GetTotalStepCount(
    displayName,
    (await GetCurrentSessionNumber(displayName)) - 1,
  );
  console.info(`Prev Steps:${prevSteps}`);
  stepEmitter.emit('stepCountChange', prevSteps + steps);
  if (displayName && isProfileCreated) {
    await SaveStepCount(displayName, stepCount, sessionPath).then(() =>
      console.info(`Saved step count:${stepCount}`),
    );
  }
}

function invokeListeners() {
  for (let listener in onStepChangeListeners) {
    listener(steps);
  }
}

function GetStepCount() {
  return steps;
}

async function RestartCounter() {
  stopCounter();
  StepCounter();
}
```

Figure 5.26: To ensure that the step-counter continued to calculate a user's movement even when the application is closed, this segment of code was written.

```
export async function StartNotificationService() {
  if (isStarted) {
    console.info('Notification Service already started');
    return;
  }
  console.info('Starting Notification Service');
  BackgroundTimer.runBackgroundTimer(async () => {
    try {
      //code that will be called every 60 seconds
      let shouldSendNotification = await CheckIfTimeToSendNotification();
      if (shouldSendNotification) {
        await SendNotificationTip();
      }
    } catch (error) {
      crashlytics().recordError(error);
      console.error(
        `Notification Service failed to send message:\n${JSON.stringify(
          error,
        )}`,
      );
    }
  }, 1000 * 60);
  isStarted = true;
}
```

Figure 5.27: To ensure that the user received notifications even when the application is closed, this code was written.

```
async function SendNotificationTip() {
  const notifications = earlierTips.concat(laterTips);
  const randomIndex = Math.floor(Math.random() * notifications.length);
  let tip = notifications[randomIndex];
  let notificationText = '';
  switch (await GetUserMode(displayName)) {
    case 'Self':
      notificationText = `You have reached ${Math.round(
        await GetStepGoalProgress(displayName),
      )}% of your goal today`;
      break;
    case 'Competition':
      notificationText = `You are ranking ${await GetGroupRank(
        displayName,
      )} in your division`;
      break;
    case 'Control':
      break;
  }
  notificationText = `<p style="color:#32bb12;">${notificationText}</p>`;
  tip = `<p style="color:#4b0082;">${tip}</p>`;
  await SendNotificationText(`${notificationText}${tip}`);
}
```

Figure 5.28: This code was written to compose the notification message depending on the user's mode

```
async function CheckIfTimeToSendNotification() {
  displayName = displayName ? displayName : await GetDisplayNameFromStorage();
  isProfileCreated = isProfileCreated
    ? isProfileCreated
    : await GetProfile(displayName);
  schedule = await GetUserScheduleFromStorage();
  if (!displayName && !isProfileCreated && !schedule) {
    return false;
  }
  let timesToCheck = await GetNotificationTimes(schedule);
  const currentTime = GetCurrentTime();
  return timesToCheck.some(time => isArrayEquals(time, currentTime));
}
```

Figure 5.29: The application performed checks to determine times to send notifications

```
async function SendNotificationText(text) {
  const channelId = await notifee.createChannel({
    id: 'tips',
    name: 'Default Channel',
  });

  await notifee.displayNotification({
    title: '<b>Step Up</b>',
    subtitle: '👣',
    body: text,
    android: {
      channelId,
      color: '#4b0082',
      smallIcon: 'notification', // optional, defaults to 'ic_launcher'.
      style: {
        type: AndroidStyle.BIGTEXT,
        text: text,
      },
    },
  });
}
```

Figure 5.30: The Notifee API from Diarmid et al. (2022) was used to format the notification.

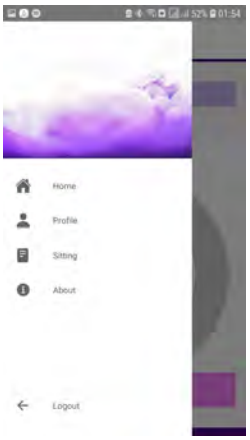

Figure 5.31: Fly-out menu

Figure 5.32: Create user

Figure 5.33: Set schedule

Understanding sedentary behaviour

This section was key to helping users learn more about sedentary behaviour and made them aware of its risks, but more importantly, offered recommendations of how to decrease it. The definition, statistics and recommendations of sedentary behaviour are represented by Figures 5.34, 5.35 and 5.36.

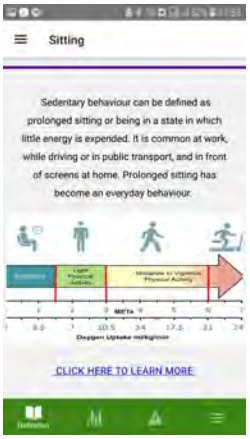

Figure 5.34: Definition

Figure 5.35: Statistics

Figure 5.36: Recommendations

Visualising movement (by feedback type)

The application visualised feedback differently depending on the user's mode. A doughnut chart (Figure 5.37) was used for goal-setting as it was ideal for illustrating progress. The

bar chart (Figure 5.38) was used for social comparison was each bar can represent a user, and all users could immediately see how they compare to the competitors in their groups. Figure 5.39 represents the control group, where their step-count was displayed but not visualised according to a type of feedback. Due to a small sample group, a control group was not tested for the second version of the application.

Figure 5.37: Goal-setting

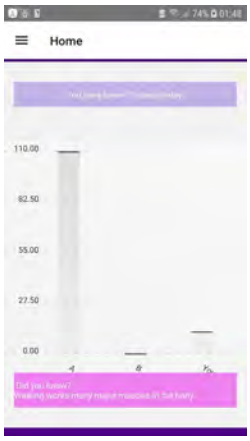

Figure 5.38: Comparison

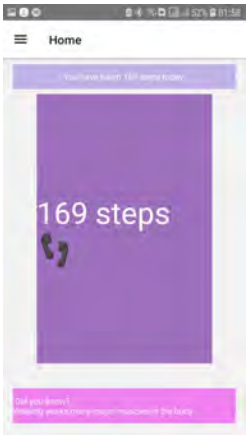

Figure 5.39: Control

5.2.6 Application feasibility

The final pilot study participants, who were Rhodes University office-based employees, installed and used the application, and their data was recorded to determine the application's feasibility. Figure 5.40 shows the individual average step count with their standard deviations. The step-count readings were compared to evaluate if feedback can be an effective tool in behavioural change and which type of feedback would be most efficient. The t-statistics for the comparison between goal-setting (S) and social comparison (SC) was calculated, and it was 0.190002151, which means that there is no significant difference between S and SC. The limitation here is that the study was quite brief and with a few participants, but Table 5.2 highlights that the application produces data that can be used for this type of comparison.

The application had some warnings, which were not crucial to the functioning of this particular application. However, Figure 5.41 and Figure 5.42 suggest that the application was feasible in its functionality – it did not have any crashes or fatal errors. This means that all major bugs were addressed at the final pilot study, and the application did not have fatal issues. Figure 5.41 shows the ninth and final iteration. The preceding artefact,

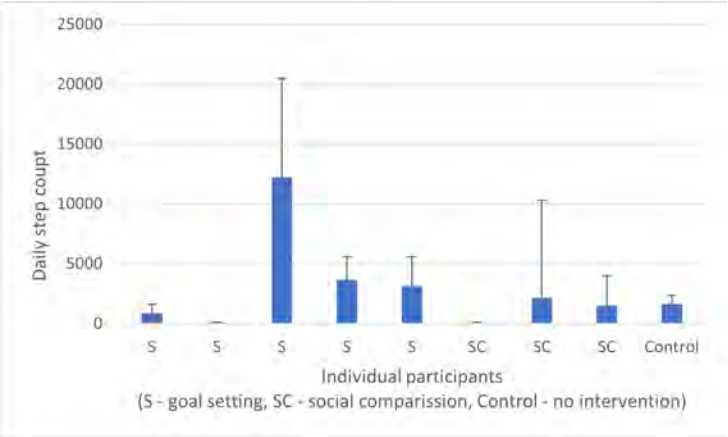

Figure 5.40: Individual results of participants with standard deviations (SC= Social Comparison, S= Self (Goal setting))

Table 5.2: This table illustrates results from the React Native application which were more consistent in comparison to the Xamarin.Forms application (5.1). Participant P9 was on leave during this period and started their test at a later stage. Participant P16 was still running the old Xamarin application.

	ID	P15	P3	P11	P6	P9	P10	P14	P52	P16
3-Dec	Fri		65	6852	1701		185	1801		2282
4-Dec			86	6546	5118		87	3460		1310
5-Dec			114	14070	3378		74	2468	102	2749
6-Dec	Mon		106	12015			61	1008	1232	2281
7-Dec	Tue	28	106	5570			62	1149	3813	970
8-Dec	Wed	497	56	4413			52	3975	10661	748
9-Dec	Thu	484		12037	1377		88	5429	1480	1000
10-Dec	Fri	956		13076	2937		46	3801	3962	1629
11-Dec		3055		12839	4834		246	54253	5623	
12-Dec		2051		7114	6774		115	4083	2867	
13-Dec	Mon	778		8984	2935		66	6649	7372	
14-Dec	Tue	723		15413	3666		48	4442	5296	
15-Dec	Wed	788		9764	4718		45	4215	4825	
16-Dec	Thu	610		3411	7724		39	0	793	
17-Dec	Fri	918		13988	4395		38	0	3918	
18-Dec		736		28937	2357		0	0	1184	
19-Dec		844		7656	3940		0	0	1126	
20-Dec	Mon	881		14320	3167		0	0	6839	
21-Dec	Tue	462		14097	919		3	0	1984	
22-Dec	Wed	876		0	5197		37	0	0	

Figure 5.41: Warnings of the React Native app (The warnings were mostly regarding SDKs which were not crucial to the project).

Figure 5.42: This figure demonstrates a 100% crash-free rate

which is not displayed, was the 8th iteration where the React Native application was tested and a few issues with the user interface altered.

Due to Covid-19, most people had the option to work at home (H), at the office (O) or a combination of the two (B). Users filled in a work location document (Figure 10 in Appendix) which could be used to determine if there is any relation between a user's work location and their sedentary behaviour. Table 5.3 shows individual participant steps, work location and mode. Most participants worked from both the office and home, and only one participant worked solely at home. Due to the limited sample, an analysis of variance was not possible.

Table 5.3: Participants average step-counts and work location. H= Home, O= Office, B= Both (Combination of the two)

ID	P15	P3	P11	P6	P9	P10	P14	P52	P16
Average	866	89	12216	3647	3133	61	2150	1467	1621
Location	B	B	B	O	H	O	B	B	O
Mode	S	S	S	S	S	SC	SC	SC	Control

5.3 From design to implementation

This application was developed from pre-existing wireframes designed prior to the implementation process. The final result looked different from the aesthetic that participants from the focus group had selected. The design of notifications are used to illustrate differences between the wireframe designs (which were made in Azure) (Figure 5.43) and the Xamarin.forms (Figure 5.44) and React Native (Figure 5.45) applications.

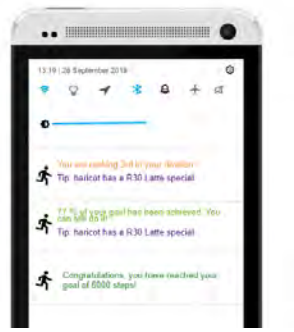

Figure 5.43: Wireframes

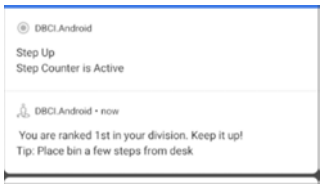

Figure 5.44: Xamarin.Forms

Figure 5.45: React Native

Chapter 6

Discussion

This discussion is divided into five sections. Section 6.1 discusses the overall experience with pilot studies and the agile method- the selected software development methodology. Xamarin.Forms and React Native will be compared and contrasted in Section 6.2. In Section 6.3, the process of going from theory to digital implementation will be discussed, including related recommendations. Finally, the implemented mobile application's feasibility will be discussed in Section 6.4. This section will also suggest future work for more rigorous testing of the intervention and effectiveness of feedback. To conclude, limitations and possible solutions will be outlined in Section 6.5.

6.1 Pilot studies and the agile method

With the agile method, applications can be developed and evaluated in chunks, rather than dealing with all errors at the end of the development process (Barjtya et al., 2017). Figure 5.21 illustrates this fact, showing the seven iterations that were followed while implementing the application. Each iteration had its fatal errors and crashes (marked in red icons). Each iteration had a different issue, depending on which component was tested during that pilot test. The agile process proved to be a suitable approach to an easily managed software development process, which concurs with Baktha (2017), who stated that most developers found it to be a natural fit.

The agile development process allows developers to implement the application in different features, where one feature can be designed, implemented, and reviewed before attending

to the next feature (Baktha, 2017). This made the development easier and less overwhelming, as issues did not need to be worked through all at once. During the implementation of the Xamarin.Forms application, the researcher started with the more simple components as an attempt to get them out the way and then tackled the more complex, bulkier code. In Figure 5.21, the rightmost column displays the different dates that the versions of the application were published to users. The fly-out navigation was set up first, followed by adding the user interfaces of the *About*, *Sitting*, *Profile* and lastly *Home* pages. The dates also show that new versions were rolled out within a much shorter period in the beginning stages of the development stage. This is in line with research by Waja et al. (2021) which stated that components could be rolled out in a short space of time.

The agile process allows developers to discover issues and cater to issues as they go (Waja et al., 2021). This project's implementation process concurred with this fact; the react native application only required one pilot study iteration, which had minor issues that needed to be altered. It was easier to foresee issues that were encountered during the Xamarin.Forms implementation and make the necessary provisions. While the platforms and programming languages are different, the concept of mobile device programming is quite similar at its core, and issues from one platform can give insight into another. This was especially true in this context, where both development frameworks are cross-platform. Use of the agile process will be discussed further, focusing on the experience with planning and the process' limitations and challenges. The Agile Manifesto will the be detailed to assess and summarise the project's use of the agile process.

Planning with agile

The planning stage of the agile process is crucial as this is where developers specify requirements and determine the duration and sequence of iterations (Waja et al., 2021). The plan for this project was to implement and test the UI for all pages, followed by their functionality. Once they were completed, the step counter and notifications would be tackled. This was to take advantage of the agile approach's ability to allow the developer to focus on these complex parts without distractions. However, the disadvantage is that if one of these complex features fails to work, a significant amount of time has already been lost. In the case of this project, the notifications failed to function as desired, and a dead-end was hit. At this stage, a new application had to be developed from scratch. Developing a new application was not accounted for in the project's timeline and hence pushed it back by a few months. This time was especially prolonged as the researcher needed to familiarise themselves with React Native and JavaScript. Had the Notifications

been the first feature to implement and test, its issues would have been found out much earlier, saving the developer time.

Section 5.3 shows the difference in designs between the wireframes and the two applications made with Xamarin.Forms and React Native. While there are many similarities between the wireframes and the application, they do not look exactly alike. Axure RP offers some components to design a wireframe that may not be feasible when implementing the application. There must be a closer relationship between wireframe design and user interface development to minimise the discrepancy between the two. The design of wireframes must be informed by which components the selected development framework will be able to support. In the future, a better approach would be to determine, during the planning stage, the core features for an application and then ascertain if the selected development platform can achieve these features as desired.

Limitations and challenges

The agile process is usually used to ensure that one feature can be implemented, tested and, once complete, be built upon with more features (Barjtya et al., 2017). Developing the Xamarin.Forms application contradicted this claim. One challenge that was faced with Xamarin.Forms was that a feature that was assumed to be completed broke down when code was added for a new feature at a later stage. The application's navigation from one page to another broke down during the implementation of the step-counter. The application would, at times, show a black screen during navigation to different pages. During one iteration of expert testing, it was discovered that this was a threading issue. When navigating from one page to another, a background thread attempted to modify the user interface which Xamarin.Forms does not allow. A simple solution to this was running the code on the main thread.

In a study by Jayatilleke et al. (2018), content experts, educational technologists, novice users, developers and researchers collaborated to create a learning management system. Similarly, Physical therapists, patients, a designer and a programmer collaborated in the production of a self-care application for patients with fibromyalgia (Yuan and Marques, 2018). The limitation of this study is that the researcher did the behavioural change and feedback theories research, designed the intervention and implemented the application. This may be a disadvantage as each section may not have received the amount of time it would have in a collaborative setting. In the future, a better approach is a collaborative

effort or allocating more time to the study to ensure that each of its aspects is done thoroughly.

The Agile Manifesto

To conclude, the project will be assessed and summarised through the four key values of the Agile Manifesto as described by (Beck et al., 2001).

Individuals and interactions The manifesto advises prioritising individuals and interactions over processors and tools. During the development process, feedback from the pilot studies and the mobile development expert was always prioritised and determined the development process's direction. Tools and processes were secondary and often adjusted to accommodate feedback from individuals.

Working Software While the software was documented, this was always secondary to ensuring that the code was working as expected. After the code was written, it was tested and further code written to fix bugs and add more features rather than allocating time to documenting and theorising the code.

Customer Collaboration This value was crucial to the project; user acceptability and a fully functioning application were the most important. Collaboration with users was flexible throughout all nine iterations of the pilot studies. It was not always possible to predict precisely when the next working application would be released. However, testing was organised and conducted as soon as there was a release, rather than following a pre-set schedule. This allowed for more rapid testing and coding process that minimised delays.

Responding to Change The initial plan for this study was to develop software using the Xamarin.Forms framework and C# language. However, when challenges were faced, a decision was made to rewrite the application using the React Native framework. It was more important that the application works optimally than sticking to a plan and using Xamarin.Forms. Plans should guide the development process but should be adjustable and not place unreasonable restrictions on the developer.

6.2 Platform comparisons

The most important functional requirements of the application are step-counting and notifications. The application was required to accurately measure a user's step count and provide feedback through notifications of this measurement. It was also important for the application to perform these feature efficiently- without over-consumption of a device's resources. Each framework's performance in meeting these requirements will be discussed.

6.2.1 Xamarin.Forms

For the Xamarin App, 12 participants were given a week to familiarise themself with the application. In this week, they could contact the researcher regarding any issues they faced with the application. One participant, who used an older phone model, had to uninstall the application due to the amount of battery capacity and mobile data is consumed. The empirical testing took one month. During this month, the researcher was working on a new version of the application, intending to fix the reported issues, which will be discussed.

Step Counting Issue This was not a bug as it was a general limitation with some smartphones. Older generation smartphones come equipped with accelerometers but not pedometers. Some smartphone companies only include pedometers in their flagship models. A pedometer is a sensor specifically designed to monitor a user's cadence and gait to calculate their steps. Some devices only have accelerometers, which measure the device's position and movement through 3D space (through the X, Y, Z plane). While they are similar, the accelerometer does not inherently measure a user's step count and require complex algorithms to do so (Preuveneers et al., 2013). An algorithm was found on GitHub (Bagi, 2022) which did this calculation. However, the step count would sometimes be inaccurate due to each device's difference in hardware quality and the accelerometer's sensitivity.

Black Screen On Navigation This UI bug caused the screen to go black and caused the application to become unresponsive to user input. This bug is a well-known issue among Xamarin developers and appears to have been around for several years. The bug occurs when a user uses a master-detail view, also known as side navigation or hamburger menu navigation, to navigate to any other application screen. The bug occurs randomly, and there is no way to tell when it might happen (other than when a user is about to navigate to another screen).

Resource Consumption This bug caused the application to drain a user's battery or mobile data drastically. In an attempt to minimise this issue, the application was designed to persist data offline and report to the database in real-time efficiently using the Firebase library (the code in Figure 5.2 and 5.1 indicates this). However, mobile data or battery consumption would skyrocket on some devices, thereby draining the device's resources. This bug was hard to replicate and test as some users did not encounter it. Cross-platform frameworks such as Xamarin.Forms are said to produce applications that are more resource-intensive than their native counterparts due to the technology they use to render user interfaces and react to user data (Dorfer et al., 2020). This becomes an even bigger issue when the application uses peripheral devices such as the accelerometer (Dorfer et al., 2020).

Notifications Bug This bug caused the notifications timer to reset after an undisclosed amount of time. These notifications were written to be sent at pre-determined times depending on a user's schedule (see Figures 5.19, 5.20 for the relevant code). Xamarin.Forms contained a bug that would prevent these notifications from being displayed to the user as the notifications timer would stop working after an undisclosed amount of time. This recurring bug in the Xamarin community does not have an official solution. This bug was very limiting because feedback provision through notifications is the core aspect of this project.

Chart Rendering This bug is related to the UI component used to show the graphs and charts to the user. The application could not render the charts to the home screen even though debugging tools showed that the data was there. This appeared to be due to an issue with the data binding in Xamarin.Forms. Fixing this bug was also a challenge because, at times, the chart did display.

Many of these bugs, especially the black screen and chart rendering bugs, were posted by other developers on coding forums like Stackoverflow. However, all responses were of developers facing the issue and none with a solution. During debugging, the code did not show any errors or indicate something was wrong. Gerasymov (2021) says that the Xamarin.Forms framework does not have a large community of active, open-source developers, as the framework has strict compatibility restrictions. As such, there are not many open source libraries available for notifications, step counters, and more. Gerasymov (2021) also confirms that it is challenging to implement third-party tools. Xamarin.Forms has been reported to be deprecated in November 2021 (wibjorn, 2022). This may be due to issues such as the ones stated above.

6.2.2 React Native

React Native was found to have the potential of solving the above challenges. Unlike Xamarin.Forms, React Native has quite a large community of active developers (Dorfer et al., 2020). This means that new solutions and libraries are constantly developed and released (Microsoft, 2020). The step-counter algorithm for React Native was more simple and did not have as much overhead. The Notifee API provides functionality for the channel, behaviour and appearance of a notification which makes it easier for the developer to create and manage notifications. As a testament to its effectiveness, many popular applications, including Coinbase, Airbnb, Uber and Facebook, are developed using the React Native framework (Native, 2022).

A few limitations to React Native were encountered while developing this application. The main hurdle was accessing device sensors and services without writing native code. React Native is a powerful platform; however, it still relies on native implementations for certain features. This is primarily a consequence of the design decisions made by the creators of React Native. This hurdle was overcome by writing native code for each platform to expose the interfaces or data required by the JavaScript thread.

6.3 Implementing DBCIs

The application was published on the Google Play Store, and an apk was also available, which can be shared via email and other social media. This meant that participants could join the pilot study from anywhere, as long as they owned an Android device. This aligns with Munro (2018), who stated that digital technologies could be scaled to reach large numbers of people. Scalability is essential to interventions because they can reach and positively impact larger numbers of people, compared to traditional interventions, which often requires face to face interactions. Another benefit of developing this application is that it had quite a low cost; both Xamarin.Forms and React Native were free, and a small fee was paid to store data on the Firebase Realtime Database and publish the application with the Google Play Console. Xamarin.Forms and React Native allow the developer to create applications for different platforms from a single codebase which saved the time it would have taken to develop and maintain two separate applications. Given these benefits of DBCIs, it is clear that they can have positive implications in the real world regarding behaviour change and intervention design.

The challenge, however, is that there are no clear guidelines that can be followed to design and implement digital interventions. There is a significant amount of behavioural change theory and many guidelines on developing a mobile application for high acceptability. However, there is insufficient knowledge of how to bridge the gap between the two. Fulton et al. (2018) expressed similar challenges, stating that the current theory is intended for traditional (non-digital) interventions.

6.3.1 Recommendations

Fulton et al. (2018) suggests looking at each behavioural change technique individually and brainstorming different ways in which it could be translated into an app feature and how it could be delivered digitally. As a result, Table 6.1 has been created to convert each BCT to an appropriate digital component. The table can allow researchers to formalise the translation from theory to implementation. It would also make it easier to evaluate an intervention, as this table can be referred to, to determine which techniques had been selected and if they were effective in attaining the target behaviour. This project's selected BCTs are used in the table to provide guidelines of how the table can be used.

In this project, The Theoretical Domains Framework from Atkins et al. (2017) was followed for intervention design, and the Behaviour Change Wheel (BCW) (Michie et al., 2011) for understanding and specifying behaviour. These theories were helpful as they provided an in-depth view of sedentary behaviour and insight into how to change it. This concurs with Murtagh et al. (2018) who used the BCW to design an intervention for physical activity. Murtagh et al. (2018) asserted that the framework provided a systematic completion of the intervention phases and an in-depth understanding of the target behaviour. Figure 6.1 outlines the suggested structure which can be used to design and implement a digital behavioural change intervention. This structure is an integration of the aforementioned behavioural change theory with the agile development method (the agile process and its benefits are described in Section 6.1). It is compiled to simplify the process and bridge the gap between theory and implementation.

The structure is in four main sections:

Selecting and specifying target behaviour This first section represents the first step of the Theoretical Domains Framework (Atkins et al., 2017). The first three steps of the guide to intervention design based on the Behaviour Change Wheel intervention development (Michie et al., 2011) are followed in this section to get a thorough

Table 6.1: BCT conversions. EachBehaviour Change Technique (BCT) is listed with a brief explanation as well as the corresponding digital conversion

Grouping	BCT	Explanation	Digital
Feedback and Monitoring	Feedback on behaviour	Provide informative and evaluative feedback on performance	Notifications on progress
	Self-monitoring of behaviour	Provide method for person to record and monitor their behaviour	Accelorometer API and form to manually record steps
Goals and planning	Goal Setting (behaviour)	Allow person to set a goal in terms of target behaviour	Set step count goal
	Review Behaviour Goal(s)	Review behaviour and consider goal modification	Every month: allow user to reflect on the progress and change goal if needed
	Discrepancy between current behaviour and goal	Analyse current behaviour in light of goal	Provide tips to catch up with goal and congratulate user on success (Table 3.2)
Self belief	Verbal persuasion about capability	Tell person that they are capable; discourage self doubt	Notifications with encouragement (Paired with Feedback)
Comparison of behaviour	Social comparison	Compare person's performance with the performance of others	Charts showing all competing users
Reward and threat	Social incentive	Inform reward if there is effort and progress	Congratulate user who tops that chart
Natural consequences	Information about health consequences	Written or verbal info about behaviour's health consequences	Awareness tabs for definition, stats, recommendations and risks. *Did you know* segment

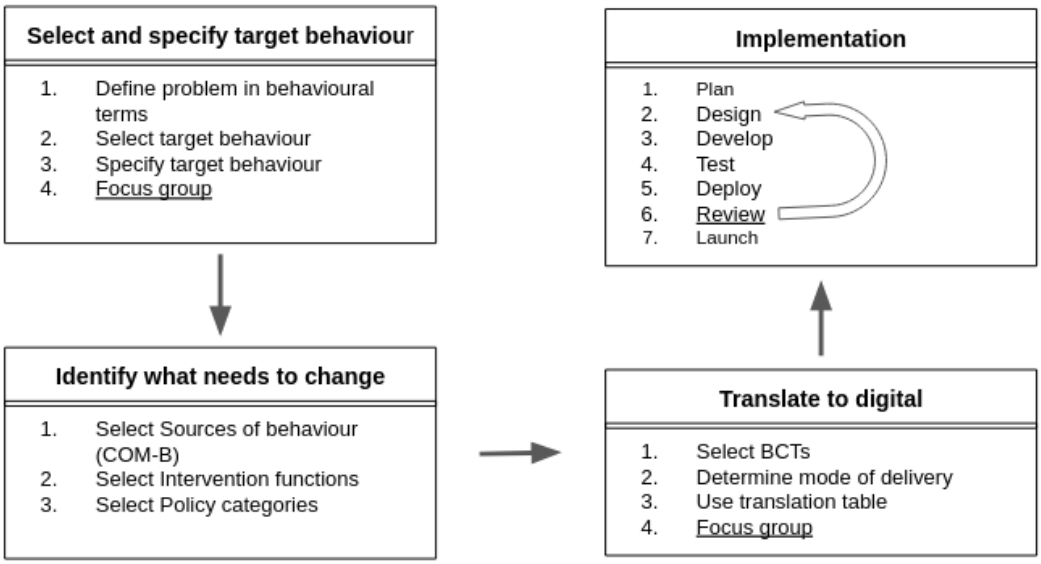

Figure 6.1: Suggested steps to take when implementing a DBCI

understanding of the problem and specify what the target behaviour is. First, the behaviour that requires an intervention must be described, including its problems and risk factors. The target behaviour must then be identified and lastly specified in as much detail as possible.

Conducting a focus group at this stage will provide insight into what potential intervention users think the problem is, if they have similar concerns or if there may be details of the behaviours that were missed. It would be more efficient to conduct a focus group study at this stage, before the sources of behaviour, functions and policies are identified, to ensure that they are identified appropriately. Yardley et al. (2016) assert that involving potential participants in intervention makes it more likely to be effective as it provides insight that the researcher may otherwise not have. Michie et al. (2017) add that a participatory approach makes for an engaging and effective intervention.

Identifying what needs to change Once the target behaviour has been specified, the next step is to determine what needs to change to go from the current to the target behaviour. The BCW (see Figure 3.1) can simplify this process. At the centre of the wheel are the sources of behaviour(COM-B). The COM-B system suggests that behaviour is a function of an individual's capabilities, opportunities and motivation. The middle ring on the wheel outlines the interventions functions which can affect different sources of behaviour. Lastly, the outermost ring of the wheel outlines the policy categories that can enable or support intervention functions if selected

appropriately.

Translation to digital Once there is sufficient knowledge of what needs to change, appropriate Behaviour Change Techniques should be selected to put the changes in motion. Fulton et al. (2018) confirms this and describes BCTs as the smallest active ingredient in an intervention capable of changing behaviour. The selection of BCTs can give researchers an idea of which mode of delivery to select. For instance, mobile notifications mapped quite well to feedback, which meant that a mobile application was a viable mode of delivery for this project. Once the mode of delivery has been decided on, Table 6.1 can be used to translate the BCTs into digital components for the selected mode of delivery. The table must be read from left to right, starting with the grouping which is the categories of the BCTs. Next, the BCTs are listed and a brief definition is provided for each. Once there is an understanding of the BCT, a digital conversion is provided. For instance, for the BCT "'Feedback and Monitoring', an explanation is provided followed by the digital conversion which is the use of mobile notifications, as notifications are a common and accessible means to provide feedback to a user.

At this stage, it would be helpful to conduct a focus group to present the plans to implement the digital intervention. This approach is time-efficient because if participants indicate that the intervention is not appropriate for their context, the researcher can revisit the plan instead of if they had already implemented an application.

Implementation If all the focus group participants approve the intervention plan, the implementation can start. The suggested methodology for development is the agile process. Section 6.1 detailed the benefits and considerations regarding this approach.

6.4 Feasibility of application

Figure 5.21 displayed red exclamation icons on all seven iterations of the first application. These icons indicate fatal errors and crashes in the application. Additionally, the notifications did not get to users daily as desired. This first application cannot be considered feasible; crashes and black screens hindered a successful user experience. The notifications did not work, despite being a key application feature. While there are some warnings (indicated by orange icons) in Figure 5.41, the final version (version 9) of the

second application did not have any crashes or fatal errors. This can be seen from Figure 5.42, where 100% of the users had a crash-free experience with the application. This indicates that the application functioned as required and is feasible from a functionality standpoint.

The main BCTs used in this project are Feedback on behaviour (in the form of notifications) and social comparison and goal-setting, which represent the two feedback types in the study. Feedback is the most commonly used BCT and is effective in changing behaviour according to Bondaronek et al. (2018) and Goodwin et al. (2016). The control theory suggests that pairing goal-setting with feedback is likely to lead to behavioural change (Hedin et al., 2019). DiMenichi and Tricomi (2015) found competition (a form of social comparison) to increase participant effort and motivation. Based on prior research, this intervention appears to have the potential to be effective at changing behaviour. Hedin et al. (2019) state that an intervention is likely to succeed when multiple BCTs are used. This claim is supported by Stockwell et al. (2019), who recommend a minimum of three technique categories for an improved outcome. This intervention uses eight BTCs from six categories, which can imply that the intervention has an even greater potential of leading to behaviour change. However, as Hekler et al. (2016a) suggest, behavioural change interventions must be thoroughly examined for their efficacy.

Table 5.2 showed that the application was counting steps more consistently for users. Participant P9 was on leave during this period and started their test at a later stage, which recorded their steps just as consistently. Participant P16 was still running the old Xamarin.Forms application. When the t-statistic was calculated from individual average step-counts in Figure 5.40, there was no significant difference between goal-setting and social comparison. This was due to a small sample space and the short duration of the pilot study. Therefore, future studies must compare these types of feedback more rigorously to determine the feasibility of feedback and which type is most effective. Hedin et al. (2019) concurs and advises that these types of studies must have a control group and be conducted over a sufficient period. An approach to conducting such a study is described in Section 6.4.1.

6.4.1 Future work

As a future study, an empirical study needs to be conducted to determine the effectiveness of feedback as a tool to decrease sedentary behaviour and which type (between goal-setting

and social comparison) is the most effective. The possible details of such a study will be detailed.

Research hypotheses This intervention is expected to decrease occupational sitting, and feedback is expected to be a valuable tool for this. The two types of feedback are expected to lead to different degrees of success. The following hypotheses are therefore proposed:

- A mobile application providing feedback about occupational sitting is effective for sedentary behavioural change.
- Social comparison and goal setting will have differing effects on the sedentary behaviour of participants during office hours.

Experimental design Data should be collected in a field study setting to ensure that participants can, as much as possible, go about their regular work schedules. This will give researchers a more realistic outcome regarding sedentary behaviour in the office. This would be an independent measures study, where each condition of the independent variables can have different groups of participants to avoid learning bias. It is suggested that participants are allocated to the different groups through strategic sampling. In a competition setting, people are more likely to put in an effort when they believe that they stand a chance (DiMenichi and Tricomi, 2015). Participants with a similar GPAQ score can be placed in the same group for the social comparison mode to ensure that competition is at a relatively equal level. The experiment would have three groups: one group for each feedback type and the third group as the control group. The control group can have their movements recorded, but they should not receive any feedback. However, the control group can receive notifications containing only tips to ensure that the difference in step-count results is due to the type of feedback (social comparison or goal-setting) and not due to the tips themselves.

Independent variables

The independent variable is the type of feedback that the user receives. The two types are social comparison and goal setting and are based on BCTs. Social comparison compares a participant's performance to that of others as a way to increase their motivation. Rankings can be displayed for participants to see how they are performing relative to their peers. Goal setting is when the participant sets a goal for themselves, which is the number of steps they want to reach per day.

Dependent variables

The two variables that will be determined during this study are step count and sitting, and will be analysed to gain insight into a participant's sedentary behaviour. Step count is the number of steps that the participants take in a day and is measured by using the accelerometer. Sitting time is the amount of time (in minutes) that a participant spends sitting and can be measured using the same sensor. Sitting time can be indicated by the time that the participant's device does not record any movement. It should, however, be noted that their measurements are not always accurate; movement can only be recorded if a user carries their mobile device. Sitting time might appear higher for users who do not carry their phones around.

Procedure and instrumentation

If the empirical study is being conducted at an institution such as a university or a company, recruitment should be done through the institution if possible. This is because support (e.g. endorsement and promotion) from an organisation can be beneficial in increasing potential participant interest (Brakenridge et al., 2016). Visible support of the program by the institution can improve participant recruitment.

Empirical study during Covid-19

If this study is conducted during the pandemic, it may help to record the participants' work location. This can then help to determine if a user's location would affect their sedentary behaviour and how well the application can work to decrease it in office settings vs home settings.

Individual differences

Some people are more likely to thrive in competitive environments and can thrive when given social comparison. For some people, however, competition induces anxiety, and they under-perform. The following surveys can determine the potential individual differences and how those relate to task performance. Depending on the purpose of the study, one or more of the below surveys can be provided to participants to determine or predict how they may be affected by goal-setting or competition.

- Hyper competitive Attitude Scale (HAS), which records individuals to reflect on habits and traits that might be associated with competitiveness.
- Personal Development Competitive Attitude Scale (PDCAS), which records individual preferences for situations in which competition can be used as a means to improve personal development. (males scored higher)
- Marlow-Crowne Social Desirability Scale (SDS), which investigates possible bias in participants' responses. The bias could be due to participants having unrealistic views of themselves or a participant's desire to please the experimenter.

Data processing

The dependent variable (step count) can finally be recorded and a two-factorial analysis of variance to determine whether there is a significant difference between the means of the two populations: the goal-setting population and the social comparison population. A comparison can also be made to the control group to determine if feedback is, in fact, a valuable tool for behaviour change.

Study variations

Furthermore, empirical studies can be conducted in varying contexts to examine the application thoroughly. Slight changes can be made to the code to only record a user's step count during office hours. This excludes the possibility that the bulk of a user's daily step count accumulates after work when they are in more domestic and social settings. Competition-related research by DiMenichi and Tricomi (2015) implies that males may perform better in competitive settings. The effects of different types of feedback can be compared between genders to test this.

6.5 Limitations

The first limitation is that some people do not use their phones regularly (or do not walk around with them). This is especially prevalent among adults compared to youth (Higgins et al., 2019). If an individual barely picks up their phone, it may appear as though they are much more sedentary than they are. One solution to this is providing a function on

the application to enter steps or distance duration of physical activity manually. Users could manually enter how long, in minutes, they were moving without a phone (duration) and the pace that they moved at. The duration and pace can be multiplied to determine an estimation of steps according to estimations by GHS (2022). The calculations can be done as follows:

- if pace = *slow*: then steps = duration x 59;
- if pace = *med*: then steps = duration x 77;
- if pace = *fast*: then steps = duration x 100;

A limitation to note with this solution is cheating — some users may enter false durations to increase their steps. Another possible solution, which was suggested during focus groups, is providing phone pouches to participants, which they can wear around the neck or waist, to encourage them to carry their devices more (Tsaoane and Zschernack, 2019).

Another limitation is Android devices that do not support Google Play Store. Generating an application from bundles provides an application with smaller storage size. The total storage space of the apk file was 35.8MB, but Google Play Store Console generated an application from app bundles rather than an apk, and the size was 13MB. Some Android devices do not support Google Play Store, so applications were installed from apk files, which took more than double the storage capacity. Shehzaib et al. (2018) state that people prefer smaller applications as that allows them to install more applications on their devices. There, the application size being significantly larger may deter some users.

Chapter 7

Conclusion

While many mobile applications on the market encourage physical activity, very few emphasise decreasing sedentary behaviour as an independent factor from increasing physical activity. Research has found that the risk factors of sedentary behaviour are not necessarily minimised by solely focusing on moderate to vigorous-intensity activity. This is especially true for office-based employees, who spend the majority of their day sitting. Research suggests the complete energy expenditure spectrum to be considered, particularly light intensity activities throughout the day in the case of office-based employees.

This application was implemented to provide means for employees to decrease their sedentary time while at work by encouraging movement on the light intensity spectrum. Research also states that feedback may be a useful tool in increasing movement and decreasing sedentary behaviour. As such, the application employed two types of feedback -goal-setting and social comparison- as the tools to decrease sedentary behaviour among individuals who work in office-like environments. The Theoretic Domains Framework, Behaviour Change Wheel and Techniques and feedback research were used to ensure sufficient theoretic grounding. Practical software development guidelines, the agile process and software architecture were then employed to maximise uptake and acceptability. However, little is known about bridging the gap between the theory and the implementation of such an intervention. The project suggests conversions from theory to a digital implementation through the provided table and diagram.

This study aimed to implement this application and test its feasibility through pilot studies. The application was first built in c# using the Xamarin.Forms frameworks and was not feasible due to crashes and fatal errors. A second version of the application was written in empirical JavaScript using the React Native framework. This version was

found to be feasible from a functionality standpoint because it worked as desired. Due to time constraints and a limited sample group, the study did not determine how much this application can decrease sedentary behaviour. Future work must look at empirical testing to determine if feedback is, in fact, an effective tool in decreasing behaviour when compared to a control group. The study must also determine which of the two types of feedback most effectively changes sedentary behaviour.

Bibliography

T. Y. Adinugroho, Reina, and J. B. Gautama. Review of Multi-platform Mobile Application Development Using WebView: Learning Management System on Mobile Platform. *Procedia Computer Science*, 59:291–297, 2015. ISSN 18770509. doi: 10.1016/j.procs.2015.07.568. URL https://linkinghub.elsevier.com/retrieve/pii/S1877050915020979.

F. J. Affonso, W. F. Passini, and E. Y. Nakagawa. A Reference Architecture to support the development of mobile applications based on self-adaptive services. *Pervasive and Mobile Computing*, 53:33–48, Feb. 2019. ISSN 15741192. doi: 10.1016/j.pmcj.2019.01.001. URL https://linkinghub.elsevier.com/retrieve/pii/S157411921830316X.

L. Atkins, J. Francis, R. Islam, D. OConnor, A. Patey, N. Ivers, R. Foy, E. M. Duncan, H. Colquhoun, J. M. Grimshaw, R. Lawton, and S. Michie. A guide to using the Theoretical Domains Framework of behaviour change to investigate implementation problems. *Implementation Science*, 12(1):77, Dec. 2017. ISSN 1748-5908. doi: 10.1186/s13012-017-0605-9. URL http://implementationscience.biomedcentral.com/articles/10.1186/s13012-017-0605-9.

L. Bagi. android-pedometer, Feb. 2022. URL https://github.com/bagilevi/android-pedometer/blob/bcff0d89ea0696e08eeb994434627dc33af7b857/src/name/bagi/levente/pedometer/StepDetector.java. original-date: 2010-04-27T06:37:16Z.

K. Baktha. Mobile Application Development: All the Steps and Guidelines for Successful Creation of Mobile App: Case Study. *International Journal of Computer Science and Mobile Computing,*, 6(9):15–20, 2017.

S. Barjtya, A. Sharma, and U. Rani. A detailed study of Software Development Life Cycle (SDLC) Models. 6(7):4, 2017.

K. Beck, J. Grenning, and R. C. Martin. Principles behind the Agile Manifesto, 2001. URL `http://agilemanifesto.org/principles.html`.

M. Bell, A. Whitehead, and S. Julious. Guidance for using pilot studies to inform the design of intervention trials with continuous outcomes. *Clinical Epidemiology*, Volume 10:153–157, Jan. 2018. doi: 10.2147/CLEP.S146397.

G. J. Bellone, S. A. Plano, D. P. Cardinali, D. P. Chada, D. E. Vigo, and D. A. Golombek. Comparative analysis of actigraphy performance in healthy young subjects. *Sleep Science*, 9(4):272–279, Oct. 2016. ISSN 19840063. doi: 10.1016/j.slsci.2016.05.004. URL `http://linkinghub.elsevier.com/retrieve/pii/S1984006316300232`.

P. Bondaronek, G. Alkhaldi, A. Slee, F. Hamilton, and E. Murray. Quality of Publicly Available Physical Activity Apps: Review and Content Analysis. *JMIR mHealth and uHealth*, 6:e53, Mar. 2018. doi: 10.2196/mhealth.9069.

C. Bontrup, W. R. Taylor, M. Fliesser, R. Visscher, T. Green, P.-M. Wippert, and R. Zemp. Low back pain and its relationship with sitting behaviour among sedentary office workers. *Applied Ergonomics*, 81:102894, Nov. 2019. ISSN 00036870. doi: 10.1016/j.apergo.2019.102894. URL `https://linkinghub.elsevier.com/retrieve/pii/S0003687019301279`.

C. L. Brakenridge, B. S. Fjeldsoe, D. C. Young, E. A. H. Winkler, D. W. Dunstan, L. M. Straker, and G. N. Healy. Evaluating the effectiveness of organisational-level strategies with or without an activity tracker to reduce office workers sitting time: a cluster-randomised trial. *International Journal of Behavioral Nutrition and Physical Activity*, 13(1):115, Dec. 2016. ISSN 1479-5868. doi: 10.1186/s12966-016-0441-3. URL `http://ijbnpa.biomedcentral.com/articles/10.1186/s12966-016-0441-3`.

V. Carson, M. S. Tremblay, J.-P. Chaput, and S. F. Chastin. Associations between sleep duration, sedentary time, physical activity, and health indicators among Canadian children and youth using compositional analyses. *Applied Physiology, Nutrition, and Metabolism*, 41(6 (Suppl. 3)):S294–S302, June 2016. ISSN 1715-5312, 1715-5320. doi: 10.1139/apnm-2016-0026. URL `http://www.nrcresearchpress.com/doi/10.1139/apnm-2016-0026`.

C. Carver and M. Scheier. Control theory: A useful conceptual framework for personality-social, clinical, and health psychology. *Psychological bulletin*, 92:111–35, Aug. 1982. doi: 10.1037/0033-2909.92.1.111.

L. Corbaln, J. Fernandez Sosa, A. Cuitio, L. Delia, G. Cseres, P. Thomas, and P. Pesado. *Development frameworks for mobile devices: a comparative study about energy consumption*. May 2018. doi: 10.1145/3197231.3197242. Pages: 201.

D. P. Credeur, S. M. Miller, R. Jones, L. Stoner, D. R. Dolbow, S. M. Fryer, K. Stone, and S. M. McCoy. Impact of Prolonged Sitting on Peripheral and Central Vascular Health. *The American Journal of Cardiology*, 123(2):260–266, Jan. 2019. ISSN 00029149. doi: 10.1016/j.amjcard.2018.10.014. URL `https://linkinghub.elsevier.com/retrieve/pii/S0002914918319714`.

CSEP. Adults 18-64 24-Hour Movement Guidelines, 2021. URL `https://csepguidelines.ca/guidelines/adults-18-64/`.

S. D'Ambra. What is Agile Software Development?, Feb. 2018. URL `https://www.cleart.com/what-is-agile-software-development.html`. Section: Web Development.

J. P. Davy, K. Scheuermaier, L. C. Roden, C. J. Christie, A. Bentley, F. X. Gomez-Olive, S. Iacovides, R. Lewis, G. Lipinska, J. Roche, A. Todd, S. Zschernack, and D. E. Rae. The COVID-19 Lockdown and Changes in Routine-Oriented Lifestyle Behaviors and Symptoms of Depression, Anxiety, and Insomnia in South Africa. *Journal of Physical Activity and Health*, 18(9):1046–1057, June 2021. ISSN 1543-5474, 1543-3080. doi: 10.1123/jpah.2020-0863. URL `https://journals.humankinetics.com/view/journals/jpah/18/9/article-p1046.xml`. Publisher: Human Kinetics Section: Journal of Physical Activity and Health.

M. Diarmid, Rusticman, and H. Ford. @notifee/react-native, 2022. URL `https://www.npmjs.com/package/@notifee/react-native`.

B. C. DiMenichi and E. Tricomi. The power of competition: Effects of social motivation on attention, sustained physical effort, and learning. *Frontiers in Psychology*, 6, Sept. 2015. ISSN 1664-1078. doi: 10.3389/fpsyg.2015.01282. URL `http://journal.frontiersin.org/article/10.3389/fpsyg.2015.01282`.

T. Dorfer, L. Demetz, and S. Huber. Impact of mobile cross-platform development on CPU, memory and battery of mobile devices when using common mobile app features. *Procedia Computer Science*, 175:189–196, Jan. 2020. doi: 10.1016/j.procs.2020.07.029.

S. D. French, S. E. Green, D. A. OConnor, J. E. McKenzie, J. J. Francis, S. Michie, R. Buchbinder, P. Schattner, N. Spike, and J. M. Grimshaw. Developing theory-

informed behaviour change interventions to implement evidence into practice: a systematic approach using the Theoretical Domains Framework. *Implementation Science*, 7(1):38, Apr. 2012. ISSN 1748-5908. doi: 10.1186/1748-5908-7-38. URL `https://doi.org/10.1186/1748-5908-7-38`.

E. Fulton, K. Kwah, S. Wild, and K. Brown. Lost in Translation: Transforming Behaviour Change Techniques into Engaging Digital Content and Design for the StopApp. *Healthcare*, 6(3):75, July 2018. ISSN 2227-9032. doi: 10.3390/healthcare6030075. URL `http://www.mdpi.com/2227-9032/6/3/75`.

A. Garg and J. Rijst. The benefits and pitfalls of employees working from home: Study of a private company in South Africa. *Corporate Board: Role, Duties and Composition*, 11:36–49, Sept. 2015. doi: 10.22495/cbv11i2art3.

O. Gerasymov. Xamarin vs React Native: A Short Comparison, 2021. URL `https://codeit.us/blog/xamarin-vs-react-native`.

GHS. Minutes in Motion 2022, 2022. URL `https://www.gundersenhealth.org/health-wellness/move/physical-activity/minutes-in-motion/`.

L. Goodwin, G. Ostuzzi, N. Khan, M. Hotopf, and R. Moss-Morris. Can We Identify the Active Ingredients of Behaviour Change Interventions for Coronary Heart Disease Patients? A Systematic Review and Meta-Analysis. *PloS one*, 11:e0153271, Apr. 2016. doi: 10.1371/journal.pone.0153271.

N. T. Hadgraft, G. N. Healy, N. Owen, E. A. Winkler, B. M. Lynch, P. Sethi, E. G. Eakin, M. Moodie, A. D. LaMontagne, G. Wiesner, L. Willenberg, and D. W. Dunstan. Office workers' objectively assessed total and prolonged sitting time: Individual-level correlates and worksite variations. *Preventive Medicine Reports*, 4:184–191, Dec. 2016. ISSN 22113355. doi: 10.1016/j.pmedr.2016.06.011. URL `https://linkinghub.elsevier.com/retrieve/pii/S2211335516300614`.

G. Hall, D. R. Laddu, S. A. Phillips, C. J. Lavie, and R. Arena. A tale of two pandemics: How will COVID-19 and global trends in physical inactivity and sedentary behavior affect one another? *Progress in Cardiovascular Diseases*, 64:108–110, Feb. 2021. ISSN 1873-1740. doi: 10.1016/j.pcad.2020.04.005.

J. Han, W.-O. Oh, I. T. Park, and A. Lee. Development and Effects of a Mobile Application for Safety Incident Prevention among Hospitalized Korean Children: A

pilot Study of Feasibility and Acceptability. *Journal of Pediatric Nursing*, 51:e69–e76, Mar. 2020. ISSN 08825963. doi: 10.1016/j.pedn.2019.09.022. URL https://linkinghub.elsevier.com/retrieve/pii/S0882596319302878.

K. Hartley, N. Lim, and C. Tortajada. Policy Note: Digital Feedback-Based Interventions for Water Conservation. *Water Economics and Policy*, 07:2071004, Jan. 2021. doi: 10.1142/S2382624X20710046.

B. Hedin, C. Katzeff, E. Eriksson, and D. Pargman. A Systematic Review of Digital Behaviour Change Interventions for More Sustainable Food Consumption. *Sustainability*, 11(9):2638, May 2019. ISSN 2071-1050. doi: 10.3390/su11092638. URL https://www.mdpi.com/2071-1050/11/9/2638.

E. B. Hekler, P. Klasnja, W. T. Riley, M. P. Buman, J. Huberty, D. E. Rivera, and C. A. Martin. Agile science: creating useful products for behavior change in the real world. *Translational Behavioral Medicine*, 6(2):317–328, June 2016a. ISSN 1869-6716. doi: 10.1007/s13142-016-0395-7. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4927453/.

E. B. Hekler, S. Michie, M. Pavel, D. E. Rivera, L. M. Collins, H. B. Jimison, C. Garnett, S. Parral, and D. Spruijt-Metz. Advancing Models and Theories for Digital Behavior Change Interventions. *American Journal of Preventive Medicine*, 51(5):825–832, Nov. 2016b. ISSN 07493797. doi: 10.1016/j.amepre.2016.06.013. URL https://linkinghub.elsevier.com/retrieve/pii/S0749379716302410.

K. Higgins, C. Nesbitt, L. Horan, A. Curtis, K. Richard, C. Stallter, and S. Verela. Brief Educational Sessions to Promote Health App Use. *The Journal for Nurse Practitioners*, 15(8):e165–e167, Sept. 2019. ISSN 15554155. doi: 10.1016/j.nurpra.2019.04.003. URL https://linkinghub.elsevier.com/retrieve/pii/S1555415519301734.

A. R. Homer, N. Owen, and D. W. Dunstan. Too much sitting and dysglycemia: Mechanistic links and implications for obesity. *Current Opinion in Endocrine and Metabolic Research*, 4:42–49, Feb. 2019. ISSN 24519650. doi: 10.1016/j.coemr.2018.09.003. URL https://linkinghub.elsevier.com/retrieve/pii/S2451965018300115.

T. S. Horn. *Advances in Sport Psychology*. Human Kinetics, 2008. ISBN 978-0-7360-5735-6. Google-Books-ID: wxDkz6akXiwC.

T. Jain, G. Currie, and L. Aston. COVID and working from home: Long-term impacts and psycho-social determinants. *Transportation Research Part A: Policy and Practice*,

156:52–68, Feb. 2022. ISSN 09658564. doi: 10.1016/j.tra.2021.12.007. URL https://linkinghub.elsevier.com/retrieve/pii/S0965856421003219.

B. G. Jayatilleke, G. R. Ranawaka, C. Wijesekera, and M. C. Kumarasinha. Development of mobile application through design-based research. *Asian Association of Open Universities Journal*, 13(2):145–168, Sept. 2018. ISSN 2414-6994. doi: 10.1108/AAOUJ-02-2018-0013. URL https://www.emerald.com/insight/content/doi/10.1108/AAOUJ-02-2018-0013/full/html.

M. Jett, K. Sidney, and G. Blmchen. Metabolic equivalents (METS) in exercise testing, exercise prescription, and evaluation of functional capacity. *Clinical Cardiology*, 13(8): 555–565, Aug. 1990. ISSN 0160-9289. doi: 10.1002/clc.4960130809.

K. W. Kang, S. M. Son, and Y. M. Ko. Time-varying Changes in Pulmonary Function with Exposure to Prolonged Sitting. *Osong Public Health and Research Perspectives*, 7 (6):382–384, Dec. 2016. ISSN 22109099. doi: 10.1016/j.phrp.2016.11.005. URL http://linkinghub.elsevier.com/retrieve/pii/S2210909916301527.

A. Kaur and K. Kaur. Systematic literature review of mobile application development and testing effort estimation. *Journal of King Saud University - Computer and Information Sciences*, page S1319157818306074, Nov. 2018. ISSN 13191578. doi: 10.1016/j.jksuci.2018.11.002. URL https://linkinghub.elsevier.com/retrieve/pii/S1319157818306074.

J.-N. Kramer and T. Kowatsch. Using Feedback to Promote Physical Activity: The Role of the Feedback Sign. *Journal of Medical Internet Research*, 19(6):e192, June 2017. ISSN 1438-8871. doi: 10.2196/jmir.7012. URL http://www.jmir.org/2017/6/e192/.

A. Lipnevich and E. Panadero. A Review of Feedback Models and Theories: Descriptions, Definitions, and Conclusions. *Frontiers in Education*, 6, Dec. 2021. doi: 10.3389/feduc.2021.720195.

C. Maher, T. Olds, E. Mire, and P. Katzmarzyk. Reconsidering the Sedentary Behaviour Paradigm. *PloS one*, 9:e86403, Jan. 2014. doi: 10.1371/journal.pone.0086403.

J. Mayne. The Capabilities, Opportunities and Motivation Behaviour-Based Theory of Change Model. May 2016. URL https://www.researchgate.net/publication/301701597.

C. McParlin, R. Bell, S. C. Robson, C. R. Muirhead, and V. Arajo-Soares. What helps or hinders midwives to implement physical activity guidelines for obese pregnant women?

A questionnaire survey using the Theoretical Domains Framework. *Midwifery*, 49: 110–116, June 2017. ISSN 02666138. doi: 10.1016/j.midw.2016.09.015. URL https://linkinghub.elsevier.com/retrieve/pii/S0266613816301723.

S. Michie, M. M. van Stralen, and R. West. The behaviour change wheel: A new method for characterising and designing behaviour change interventions. *Implementation Science*, 6(1):42, Dec. 2011. ISSN 1748-5908. doi: 10.1186/1748-5908-6-42. URL http://implementationscience.biomedcentral.com/articles/10.1186/1748-5908-6-42.

S. Michie, C. E. Wood, M. Johnston, C. Abraham, J. J. Francis, and W. Hardeman. Behaviour change techniques: the development and evaluation of a taxonomic method for reporting and describing behaviour change interventions (a suite of five studies involving consensus methods, randomised controlled trials and analysis of qualitative data). *Health Technology Assessment*, 19(99):1–188, Nov. 2015. ISSN 1366-5278, 2046-4924. doi: 10.3310/hta19990. URL https://www.journalslibrary.nihr.ac.uk/hta/hta19990/.

S. Michie, L. Yardley, R. West, K. Patrick, and F. Greaves. Developing and Evaluating Digital Interventions to Promote Behavior Change in Health and Health Care: Recommendations Resulting From an International Workshop. *Journal of Medical Internet Research*, 19(6):e232, June 2017. ISSN 1438-8871. doi: 10.2196/jmir.7126. URL http://www.jmir.org/2017/6/e232/.

Microsoft. Xamarin documentation - Xamarin, 2020. URL https://docs.microsoft.com/en-us/xamarin/. Library Catalog: docs.microsoft.com.

M. Mirzaei, M. Askari, N. Namiranian, and H. a. Mahmoudi Kohani. Relationship between sitting time and anthropometric indices: Findings of Yazd health study (YaHS). *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, 13(3):2220–2225, May 2019. ISSN 18714021. doi: 10.1016/j.dsx.2019.05.015. URL https://linkinghub.elsevier.com/retrieve/pii/S1871402119302863.

E. H. Mory. FEEDBACK RESEARCH REVISITED. page 40, 2003.

G. Munro. *A realist review of digital behaviour change models for cardiac rehabilitation.* Feb. 2018. doi: 10.13140/RG.2.2.12746.26567.

E. Murtagh, A. Barnes, J. McMullen, and P. Morgan. Mothers and teenage daughters walking to health: using the behaviour change wheel to develop an intervention to improve adolescent girls' physical activity. *Public Health*, 158:37–46, May 2018. ISSN

00333506. doi: 10.1016/j.puhe.2018.01.012. URL https://linkinghub.elsevier.com/retrieve/pii/S0033350618300155.

R. Native. Introduction React Native, 2022. URL https://reactnative.dev/docs/getting-started.

S. OConnor, P. Hanlon, C. A. ODonnell, S. Garcia, J. Glanville, and F. S. Mair. Understanding factors affecting patient and public engagement and recruitment to digital health interventions: a systematic review of qualitative studies. *BMC Medical Informatics and Decision Making*, 16(1):120, Sept. 2016. ISSN 1472-6947. doi: 10.1186/s12911-016-0359-3. URL https://doi.org/10.1186/s12911-016-0359-3.

K. Patrick, E. B. Hekler, D. Estrin, D. C. Mohr, H. Riper, D. Crane, J. Godino, and W. T. Riley. The Pace of Technologic Change. *American Journal of Preventive Medicine*, 51 (5):816–824, Nov. 2016. ISSN 07493797. doi: 10.1016/j.amepre.2016.05.001. URL https://linkinghub.elsevier.com/retrieve/pii/S0749379716301386.

D. Preuveneers, Y. Berbers, and W. Joosen. The Future of Mobile E-health Application Development: Exploring HTML5 for Context-aware Diabetes Monitoring. *Procedia Computer Science*, 21:351–359, 2013. ISSN 18770509. doi: 10.1016/j.procs.2013.09.046. URL https://linkinghub.elsevier.com/retrieve/pii/S1877050913008399.

P. F. Saint-Maurice, R. P. Troiano, D. R. Bassett, Jr, B. I. Graubard, S. A. Carlson, E. J. Shiroma, J. E. Fulton, and C. E. Matthews. Association of Daily Step Count and Step Intensity With Mortality Among US Adults. *JAMA*, 323(12):1151–1160, Mar. 2020. ISSN 0098-7484. doi: 10.1001/jama.2020.1382. URL https://doi.org/10.1001/jama.2020.1382.

U. Shehzaib, J. Ferzund, and M. Asif. Mobile Apps Acceptability: A Meta-Analysis Model for Google Play. *International Journal of Information Technology and Web Engineering*, 13(4):1–13, Oct. 2018. ISSN 1554-1045, 1554-1053. doi: 10.4018/IJITWE.2018100101. URL http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/IJITWE.2018100101.

A. Sherrington, J. J. Newham, R. Bell, A. Adamson, E. McColl, and V. Araujo-Soares. Systematic review and meta-analysis of internet-delivered interventions providing personalized feedback for weight loss in overweight and obese adults. *Obesity Reviews: An Official Journal of the International Association for the Study of Obesity*, 17(6): 541–551, June 2016. ISSN 1467-789X. doi: 10.1111/obr.12396.

S. Shukla and A. Chauhan. react-native-accurate-step-counter, 2022. URL `https://www.npmjs.com/package/react-native-accurate-step-counter`.

S. Stockwell, P. Schofield, A. Fisher, J. Firth, S. E. Jackson, B. Stubbs, and L. Smith. Digital behavior change interventions to promote physical activity and/or reduce sedentary behavior in older adults: A systematic review and meta-analysis. *Experimental Gerontology*, 120:68–87, June 2019. ISSN 05315565. doi: 10.1016/j.exger.2019.02.020. URL `https://linkinghub.elsevier.com/retrieve/pii/S0531556519300877`.

M. Tavakoli, L. Zhao, A. Heydari, and G. Nenadi. Extracting useful software development information from mobile application reviews: A survey of intelligent mining techniques and tools. *Expert Systems with Applications*, 113:186–199, Dec. 2018. ISSN 09574174. doi: 10.1016/j.eswa.2018.05.037. URL `https://linkinghub.elsevier.com/retrieve/pii/S0957417418303361`.

M. Tsaoane and S. Zschernack. Designing a feedback app to reduce sedentary behaviour. Technical report, Rhodes University, 2019. URL `https://docs.google.com/document/d/1jWixkuycM6Z9RIDGp_foDlFXUNAsMxar/edit?usp=embed_facebook`.

A. Urruticoechea-Arana, F. Len-Vzquez, V. Giner-Ruiz, J. L. Andru-Snchez, A. Oliv-Marqus, M. Freire-Gonzlez, J. M. Pego-Reigosa, S. Muoz-Fernndez, J. A. Romn-Ivorra, J. J. Alegre-Sancho, F. Vargas-Negrn, M. Medina-Abelln, T. Cobo-Ibez, X. Mas-Garriga, J. Calvo-Aln, C. Costa-Ribas, R. Blanco-Vela, . Prez-Martn, E. Beltrn-Cataln, J. Forcada-Gisbert, M. V. Hernndez-Miguel, J. C. Hermosa-Hernn, J. Narvez-Garca, E. Nieto-Pol, and . Ra-Figueroa. Development of an application for mobile phones (App) based on the collaboration between the Spanish Society of Rheumatology and Spanish Society of Family Medicine for the referral of systemic autoimmune diseases from primary care to rheumatology. *Reumatologa Clnica (English Edition)*, page S217357432030040X, May 2020. ISSN 21735743. doi: 10.1016/j.reumae.2019.09.001. URL `https://linkinghub.elsevier.com/retrieve/pii/S217357432030040X`.

H. P. van der Ploeg and M. Hillsdon. Is sedentary behaviour just physical inactivity by another name? *International Journal of Behavioral Nutrition and Physical Activity*, 14(1):142, Dec. 2017. ISSN 1479-5868. doi: 10.1186/s12966-017-0601-0. URL `http://ijbnpa.biomedcentral.com/articles/10.1186/s12966-017-0601-0`.

G. Waja, J. Shah, and P. Nanavati. AGILE SOFTWARE DEVELOPMENT. *International Journal of Engineering Applied Sciences and Technology*, 5, Apr. 2021. doi: 10.33564/IJEAST.2021.v05i12.011.

WHO. Global Physical Activity Questionnaire, 2022a. URL https://www.who.int/ncds/surveillance/steps/resources/.

WHO. Stay physically active during self-quarantine, 2022b. URL https://www.euro.who.int/en/health-topics/health-emergencies/coronavirus-covid-19/publications-and-technical-guidance/noncommunicable-diseases/stay-physically-active-during-self-quarantine.

wibjorn. Developer tools, technical documentation and coding examples, 2022. URL https://docs.microsoft.com/en-za/.

S. J. Winter, J. L. Sheats, and A. C. King. The Use of Behavior Change Techniques and Theory in Technologies for Cardiovascular Disease Prevention and Treatment in Adults: A Comprehensive Review. *Progress in Cardiovascular Diseases*, 58(6):605–612, May 2016. ISSN 0033-0620. doi: 10.1016/j.pcad.2016.02.005. URL http://www.sciencedirect.com/science/article/pii/S0033062016300147.

L. Yardley, B. J. Spring, H. Riper, L. G. Morrison, D. H. Crane, K. Curtis, G. C. Merchant, F. Naughton, and A. Blandford. Understanding and Promoting Effective Engagement With Digital Behavior Change Interventions. *American Journal of Preventive Medicine*, 51(5):833–842, Nov. 2016. ISSN 07493797. doi: 10.1016/j.amepre.2016.06.015. URL https://linkinghub.elsevier.com/retrieve/pii/S0749379716302434.

S. L. K. Yuan and A. P. Marques. Development of ProFibro a mobile application to promote self-care in patients with fibromyalgia. *Physiotherapy*, 104(3):311–317, Sept. 2018. ISSN 00319406. doi: 10.1016/j.physio.2018.04.005. URL https://linkinghub.elsevier.com/retrieve/pii/S0031940618300762.

Acronyms

BCI Behaviour Change Intervention. 2, 5, 9, 10, 12

BCT Behaviour Change Technique. i, 3, 12, 15, 29, 81, 82, 84, 85

BCTs Behaviour Change Techniques. 2, 10, 11, 15, 23, 29, 81, 84–86

BCW Behaviour Change Wheel. iv, 3, 10, 28, 29, 81, 83

DBCI Digital Behaviour Change Interventions. 2, 13–15

GPAQ The Global Physical Activity Questionnaire. 45, 46, 86

ICT Information and Communications Technologies. 2, 21

JSX JavaScript XML. 20

MCPD Mobile Cross-Platform Development. 19

MVPA moderate to vigorous physical activity. 11

MVVM Model View View-Model. 20, 40, 49

PI physical inactivity. 5

TDF Theoretical Domains Framework. 3, 9, 27

UI user interface. 20

WHO World Health Organisation. 8, 9, 26

Appendices

Participants downloaded the application from Google Play Store or installed it through an apk file. The rest of the screens are displayed by Figures 1 to 6.

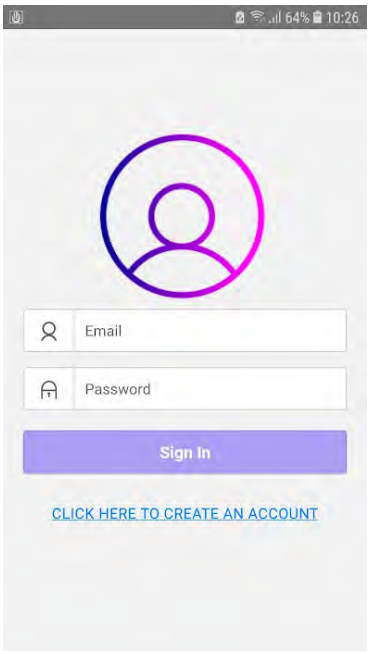

Figure 1: Login page

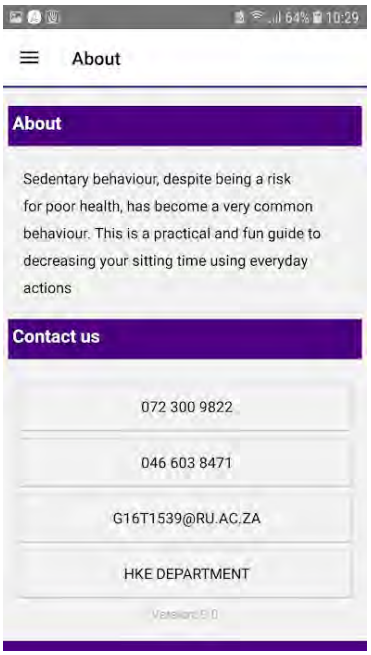

Figure 2: 'About" page

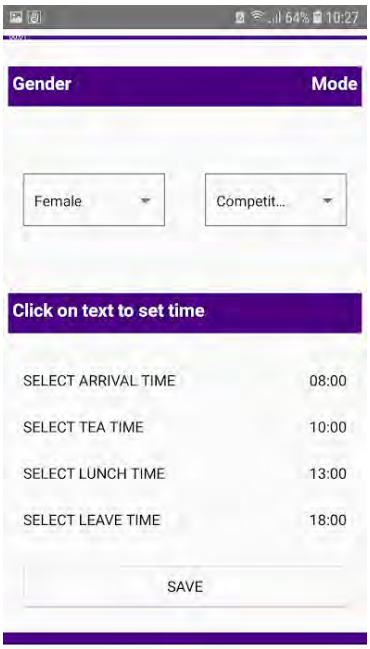

Figure 3: Profile page

Figure 4: Risks Tab

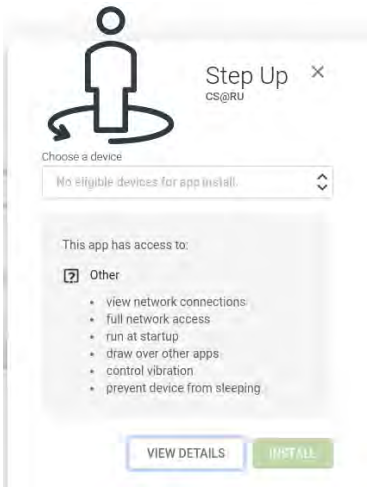

Figure 5: PC Install

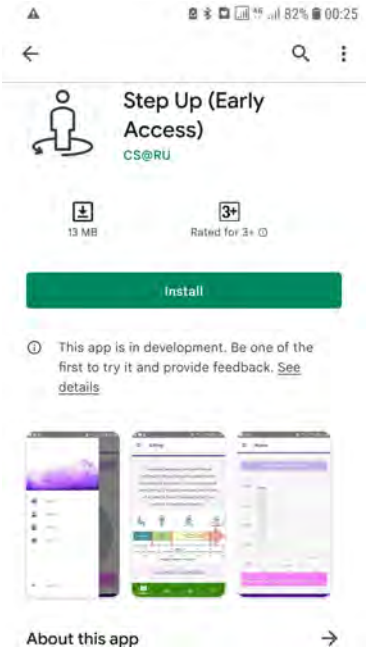

Figure 6: Google Play Store

The development and pilot implementation of a mobile application to decrease occupational sitting through goal setting and social comparison
By: Moipone Tsaoane
Supervisors: Prof George Wells & Dr Swantje Zschernack

INFORMED CONSENT DECLARATION

Moipone Tsaoane from the Department of Computer Science Rhodes University has requested my permission to participate in the above-mentioned research project.

I, _____ hereby agree to participate in this research project. The research project, which has been entitled above, has been fully explained to me. I am aware of the purpose of the project as well as what my involvement in it entails.

I have been made aware that:

- ☐ The purpose of the research project is to determine whether feedback can decrease too much sitting among office-based workers.
- ☐ The Rhodes University has given ethical clearance to this research project 0755 and I have seen/may request to see the clearance certificate by contacting Mr. Siyanda Manqele (s.manqele@ru.ac.za).
- ☐ I will need to download a mobile application onto my personal mobile device.
- ☐ I will need to register on the application, with my work email.
- ☐ My work email, created password and username will be saved on the cloud, but will not be shared with others.
- ☐ My daily steps will be recorded and possibly shared with others.
- ☐ I may receive notifications from the mobile application.
- ☐ My name will not be included in any records regarding this project, instead, I will be assigned a code which will identify my views and results.
- ☐ I have been made aware that participation in this study is fully optional, and that I have an option to discontinue my participation at any point.
- ☐ I am aware that I will have to fill in a survey
- ☐ Any further questions that I might have concerning the research, or my participation will be answered by Moipone Tsaoane (g16t1539@ru.ac.za/0723009822)
- ☐ By signing this informed consent declaration, I am not waiving any legal claims, rights or remedies.

Figure 7: Consent Form page 1

RU Ethics coordinator contacts: s.manqele@ru.ac.za

Participant:

_____	_____	_____
(Print Name)	(Signed)	(Date)

Researcher:

_____	_____	_____
(Print Name)	(Signed)	(Date)

Figure 8: Consent Form page 2

Dear participant

Thank you so much for your patience. The Step Up App is finally done - please click this link to download it from Google Play. If your device does not support Google play, please install the APK from this drive. (I will be on campus if you need any help installing the application)

You will be assigned a mode of either Competition, Goal-Setting, or Control (This will be done automatically and can be found under the *Profile Tab*). If your mode is Competition or Goal-Setting you will receive daily notifications informing you of your progress, as well as tips on how to improve. If your mode is Control you will not receive any notifications. You are not required to keep the app open as it will run in the background.

I have also attached a ***Work Location Document***. Please fill it in regularly to let us know if you're working from home, the office, or a bit of both. We only need this information for data analysis and will not be sharing it with anyone.

If you hadn't already done so, please fill in the consent form attached.

Kind regards
Moipone Tsadane

Figure 9: Email sent to participants

Date	Day	START	07:00-07:30	07:30-08:00	08:00-08:30	08:30-09:00	09:00-09:30	09:30-10:00	10:00-10:30	10:30-11:00	11:00-11:30	11:30-12:00
	Example	8:45	H	H	H	H		H	TEA	O	O	O
6-Sep	Mon											
7-Sep	Tue											
8-Sep	Wed											
9-Sep	Thu											
10-Sep	Fri											
13-Sep	Mon											
14-Sep	Tue											
15-Sep	Wed											
16-Sep	Thu											
17-Sep	Fri											
20-Sep	Mon											
21-Sep	Tue											
22-Sep	Wed											
23-Sep	Thu											
24-Sep	Fri											
27-Sep	Mon											
28-Sep	Tue											
29-Sep	Wed											
30-Sep	Thu											
1-Oct	Fri											
4-Oct	Mon											
5-Oct	Tue											
6-Oct	Wed											
7-Oct	Thu											
8-Oct	Fri											
11-Oct	Mon											
12-Oct	Tue											
13-Oct	Wed											
14-Oct	Thu											
15-Oct	Fri											
18-Oct	Mon											
19-Oct	Tue											
20-Oct	Wed											
21-Oct	Thu											
22-Oct	Fri											
25-Oct	Mon											
26-Oct	Tue											
27-Oct	Wed											
28-Oct	Thu											
29-Oct	Fri											
1-Oct	Mon											
2-Oct	Tue											
3-Oct	Wed											

Figure 10: Work location document