Dr Holly Tootell

Faculty of Engineering and Information Sciences

University of Wollongong

Northfields Ave

Wollongong NSW 2522

14 October, 2022

Dear Dr. Tootell,

It is with great pleasure that we submit herewith the report on our front-end application that works towards solving an information systems problem derived from the United Nation's 2022 Sustainable Development Goals. This report has been prepared using the knowledge acquired from the Human Computer Interaction course, together with detailed research into the impacts of climate change and natural disasters on Australian rural communities.

In Part A, our analysis included researching the potential users of our system, allowing us to develop personas, scenarios and storyboards showcasing how our system can assist rural property owners to plan for and survive extreme climate events. Further, we evaluated existing applications in this domain, in order to create an innovative and comprehensive product. Based on our research, we compiled a list of initial requirements for our interface, *Cultivate,* which will form the basis of our prototype.

The newly appended section, Part B, documents the iterative process that we undertook to design the application prototype, progressing from a wireframe and wireflow level to a high-fidelity stage. It demonstrates our creativity and innovation in designing a User Interface and User Experience that can assist Australian farmers prepare for and recover from flood, fire and drought, as well as our consideration of the Human-Computer Interaction theory that underpins our design. Lastly, we present an evaluative assessment of our final design.

Thank you for allowing us to submit our research. Please review the attached report and respond with your thoughts. If you have any questions, please reach out to us at cultivate@gmail.com.

Sincerely,

Brad Watson and Jade Harris

cultivate

Documentation to Cultivate UNSDG Front-End Solution

The interface between climate and community

Recipient:	Dr Holly Tootell
Prepared By:	Brad Watson Jade Harris
Date:	14-10-2022

GLOSSARY/LIST OF ABBREVIATIONS

Property: In the context of a rural, typical agricultural property. A broader term to encompass large and often farming properties.

Property Members: All of the residents on a single property that uses the IoT sensors, often including the property owner, farmhands, and family members.

IoT (Internet Of Things): Network of devices that are connected through sensors to exchange data.

IoT sensor: Devices that detect changes in an environment, which is used in an Internet network.

Dashboard: A visual display of software data, in a way that provides all information at-a-glance.

Archetype: A classification that a represents a user group, for instance 'Property owner of a family-run farm'

Persona: A design technique devised by Cooper (1999), encompassing fictional characters who are created to represent user types (stakeholders that use an application in a similar way)

Storyboard: Representation of a sequence of actions that a user process through to achieve a goal (Sharp & Rogers & Preece, 2019)

Quantitative Data: Information that can be expressed as a certain quantity, amount or range. Often this is represented as numbers, such as rating a scale 1 to 5, though it also includes when two values can be 'yes' and 'no'.

Qualitative Data: information that is not easily scaled to a possible set of values such as numerical values, instead often represented as qualities and characteristics.

Functional Requirements: involve the critical tasks that the new system must complete to function

FURPS+: Terminology used to target the requirements to consider during the design of the system. The acronym represents Functionality, Usability, Reliability, Performance and Security, as well as + which refers to (x) amount of further criteria (appendix A)

Design (Non-functional) Requirements: Refers to the other criteria contained in FURPS+ aside from functional requirements, such as usability, reliability, performance and security, and generally require less priority.

Usability Requirements: Operational characters related to users such as the user interface and experience, work processes, online help and documentation. Additional considerations include menu format, colour schemes, use of logo and multi-language support.

Reliability Requirements: Describe the dependability of a system in terms of how it detects and recovers from service outages and incorrect processing

Performance Requirements: Operational characteristics related to measures of workload, such as throughput and response time.

Throughput: The rate that the amount of something (data) passes through something (system)

Security Requirements: Describe the mechanisms to control access and protection to data during storage and transmission.'

Fidelity: The level of detail a design is presented at.

Prototype: A prototype is a draft model of a product which assists innovators in planning and gauging the feasibility of designs rapidly, as well as collect feedback from team members, clients, and other perspectives.

Low Fidelity Prototype: A classification for a prototype that contains minimal detail, typically used to plan functionality and rapidly changing aspects of the system.

High Fidelity Prototype: A classification for a prototype that contains greater detail, encompassing prototypes at a level of detail that is almost photo-realistic.

Design Pattern: Interface solutions that can solve common problems faced during the application design process. They are often compiled by as a part of academic research or formed from industry experience, offering designs a proven level of quality.

EXECUTIVE SUMMARY

Australian rural industries must better manage natural disasters if they are to grow and be environmentally sustainable (GHD Pt Ltd, 2020). Climate change is a complex environmental, social, and cultural issue, exacerbated by extremes of floods, fires, and droughts worldwide. This has caused instability in the Australian agricultural landscape, emerging through losses of land, livelihood, and lives. Through the integration of emerging technology in the agricultural domain, such as Internet of Things sensors, Australian property owners have been able to monitor, protect and sustain these lands. The Cultivate interface has leveraged this shift to research and produce a unique application that offers property member the opportunity to quickly view statistics. It is intended to increase productivity and security of Australian agricultural output.

Furthermore, Cultivate inspires united growth, encouraging local communities to assist those in need. The connectivity the software provides leverages modern innovations in social media and offers users a wide variety of resources and utilities to support each other.

At a high level, the Cultivate interface system will allow rural property owners to:

- Manage and maintain statistics of their property through an interface that connects to IOT sensors installed on their location.
- Increase output efficiency by automating tasks and checks relating to flood, fire, and drought, namely atmosphere temperature, .
- Connect local communities through a central 'request and fulfil' system and community board.
- Assist in addressing the 2022 United Nation Sustainable Development Goal 13: Climate action, specific target 13.1, 'Strengthen resilience and adaptive capacity to climate-related hazards and natural disasters in all countries' (UN 2022).
- As supplementary benefits, our interface will address Goal 9: Industry, innovation and infrastructure, Goal 15: Life on Land research and Goal 17: Partnerships for the goals.

It is deduced that the application will employ a preventative approach to significantly reduce the physical, financial, and emotional and impact of climate extremes. This will offer consistency and security to rural property owners and communities, maximise business profitability and improve Australian prosperity. The development team of Cultivate has evaluated the problem domain and surrounding research in depth and are confident that our system will deliver impactful outcomes for its shareholders, and our world.

TABLE OF CONTENTS

Glossary 3
Executive Summary 4
Introduction 6

Part A: Users and Research 7
Problem Domain 8
Current Systems Evaluation 10
Stakeholder Analysis 16
Users 17
Primary 17
Secondary 21
Tertiary 23
Non-users 24
Personas 26
Scenarios 31
Use cases 41
Initial Requirements 47

Part B: Prototyping and Design 45
Iterative Prototyping and Interfaces 46
Storyboards 48
Wireframes and Wireflows 55
Low Fidelity Prototyping 76
Discussion on acceptance of key design patterns 95
Design Iterations 98
Final Design 114
Review of Design 116
Evaluation Assessment 117
Testing Considerations 118

Conclusion 120
References 121
Appendices 125
Appendix A: Style Guide 125

INTRODUCTION

Australia has been shaped by extreme temperatures, droughts, floods, and fires (Freund & Henley 2017). The 'Black Summer' bushfires of 2019/ 2020 demonstrated the scale at which this unpredictable phenomenon can impact property, destroying some 3000 homes, businesses, and farms (Gissling 2020) and burning 5.5 million hectares of land (Australian Institute for Disaster Resilience n.d.). Further, they showcased the impact of a natural disaster on life and community, resulting in 452 fatalities (either directly from the fire or from smoke inhalation) and another 3151 hospitalisations (Borchers Arriagada et al. 2020). This extreme fire season was compounded by three preceding years of drought conditions causing low moisture levels in bushfire fuel.

The Queensland floods of 2010-2011 caused damage estimated at around $2.38 billion (Australian Institute for Disaster Resilience n.d.), and in 2012 flooding resulted in more than 75 per cent of New South Wales being under water or threatened by floodwaters (Australian Institute for Disaster Resilience n.d.). More recently, the July 2022 flooding in New South Wales and Queensland placed over 50,000 property owners on evacuation alert (Turnbull, 2022), and damaged irrigation infrastructure, farm equipment and machinery, structures such as houses, sheds and fences, and infrastructure such as roads and bridges.

Drought can have dramatic and long-term effects on soil and can impact future production (NSW Dept. of Primary Industries 2020). An estimated 55 million people globally are directly affected by drought every year, making it the most serious hazard to livestock and crops in nearly every part of the world. (World Health Organisation 2021)

The United Nations speculates that medium-to-large scale disasters will increase by 40% from 2015 to 2030. (United Nations, 2022). The unpredictability of these extreme climatic events exacerbates the damaging impact they have on Australian communities, farmers, businesses, bushland, and wildlife, leading to financial stress, livelihood insecurity and psychological trauma. The Australian Bureau of Statistics (ABS) identifies the consistent impacts of extreme weather events as reduced production and income, and damage to and destruction of assets (Australian Bureau of Statistics 2020).

In the following sections, we present an insight into our research and understanding of the requirements for the Cultivate front-end application. Information gathered about the system has been garnered from our study of Human Computer Interaction, the United Nations Sustainable Development Goals, as well as from industry white papers and Internet research. The report is divided into two parts: Part A - Users and Research, and Part B – Prototyping and design. Firstly, through defining our problem domain and solution system, we conduct market analysis into benefits and downfalls of systems that current exist in the field. Then, stakeholder analysis identifying data gathering techniques for primary, secondary, and tertiary users, as well as non-users, refines an understanding of system requirements through a user-centred lens. This forms the basis of personas, which are used in scenarios and storyboards, subsequently leading to more technical use cases. In preparation for prototyping, a final list of initial requirements summarises our application. Then, Part B showcases the iterative design process behind our final high-fidelity prototype. The creation of wirelows, low fidelity prototypes, and finally the presentation of our final design reflects the refinement of our creative process. Subsequently, an assessment of our design is conducted to close the project. An explanation for the technical purpose precedes each section and document. Finally, the conclusion discusses the future design and implementation activities built upon the research and prototyping of the solution system.

Further, the project title 'Cultivate' incorporates the agricultural term 'to foster the growth of' to emphasise and 'play-on' the impact of our interface in both the agricultural sector and building of a sense of community.

cultivate

Part A

USERS AND RESEARCH

Problem Domain 8
Current Systems Evaluation 10
Stakeholder Analysis 16
Users 17
Primary 17
Secondary 21
Tertiary 23
Non-users 24
Personas 26
Scenarios 31
Use Cases 37

PROBLEM DOMAIN

Our problem domain is derived from the 2022 United Nations Sustainable Development Goal 13: Climate action, which calls for contributions to mitigate climate change and its impacts. The specific issue that Cultivate aims to explore is the ongoing damage to agricultural and rural communities, affected by issues such as property drought, flood and fire. This correlates to Goal 13 target 13.1, 'Strengthen resilience and adaptive capacity to climate-related hazards and natural disasters in all countries' (UN 2022). Drought depletes water stores for stock as well as for irrigation, leading to weakened and dying livestock, spoiling crops, destroying harvests, and affecting soil quality for future harvests. The effect of flooding includes losses of livestock, crops, buildings, technology, and machinery. Fire can destroy homes, businesses, and farms, kill stock, destroy harvests, and negatively affect soil quality. All of these phenomena impact the lives and livelihoods of farmers.

CURRENT STATISTICS IN THE CLIMATE-INDUCED DISASTER AND CHANGE DOMAIN:

- By 2030, damage-related loss in property values from climate hazards across Australia is estimated to reach $S571 billion. (Statista Research Department 2022)
- In the last 20 years, the risk of very low farm profits due to climate variability is double that of the
- prior 50 years. (Hughes & Gooday 2021)
- Changes in climate since 2000 have reduced average annual farm profits by 22%. (Hughes, Galeano,
- & Hatfield-Dodds 2019)
- The Australian 2019/2020 bushfire season damaged over 2.45 million hectares of agricultural land. (Statista 2022)
- The deleterious effects of an intense bushfire on soil (including loss of organic carbon and nutrients, water repellency, and increased erosion) can last for decades. (Mosley 2020)
- Flooding in NSW costs the economy an average of over $250 million every year. (NSW Government
- n.d.)
- In the far north Queensland floods in 2019, an estimated 600,000 head of cattle were lost. (Australian
- Institute for Disaster Resilience n.d.)
- The past three years have been the driest on record for New South Wales. (NSW Government n.d.)
- In the 2006/ 2007 drought, production of Australia's three main resource crops – wheat, barley and canola – was cut by more than 60%. (Ploegmakers 2007)

CURRENT TECHNOLOGIES IN THE CLIMATE CHANGE AND NATURAL DISASTER DOMAIN:

- Incorporation of technological devices into disaster relief, such as Open Relief UAV (Unmanned Arial Vehicle), where drones and robots assist in management.
- Role of social media as an early warning alert system and for data collection (GHD Pt Ltd, 2020)

As supplementary benefits, Goal 9: Industry, innovation and infrastructure, Goal 15: Life on Land; and Goal 17: Partnerships for the goal are also incorporated in our research. Goal 9 aims to build resilient infrastructure and promote sustainable industrialisation, supported through our implementation of technology in the traditional agricultural domain that facilitate innovative ways of farming, Goal 15 promotes the protection, restoration, and sustainable use of ecosystems, promoted through our easy-to-use system and reliable IoT software facilitating renewable practices on farmland. Additionally, Goal 17 aims to support the successful development of the goals by forming progress upon a shared vision, achieved by connecting community members on a local level to centre human assistance and united change.

APPLICATION SUMMARY

Australian property owners can sign into our system and request installation, and we will allocate an installer to travel to the designated property and install IoT sensors that measure the presence of liquid (monitor drought and flood), and temperature sensors (monitor fire - intensity and location). These will connect to a central dashboard interface on a smartphone and tablet interface, allowing users to monitor and realise opportunities to bolster their current agricultural activity and productivity. This application thus allows users to predict and prepare for climate-induced disaster, allowing property owners to observe the status of drought (relevant to 2020), fire (relevant to the 2021 bushfires) and flood (relevant to QLD, NSW and recent). Information regarding connected concerns including the weight of livestock feed available (for places in the paddoc, charge in electrical generators (to test back-up technology), and water tank levels are also displayed. This offers accuracy to the measurement of vital property characteristics, and ensures important parts of the everyday routine is met.

Additionally, the app includes a community section, allowing community users to connect and support one another through the challenges of climate change. By leveraging the benefits of social media to facilitate connection and collaboration, the application provides a platform that supporting communities in building ongoing resilient systems. To promote the installation of plans and disaster relief, individuals can connect with other users on their community board to request help and offer aid. Furthermore, meteorological personnel, government departments and emergency responders can contribute to community boards during emergency events, offering a central location for information and coordination. Each community board can also include a 'walkie talkie' audio system, where a facilitated voice channel enables rapid communication within a local range.

SUMMARY OF FEATURES

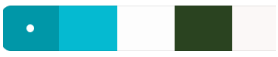
- 'My Property' dashboard, that offers:
    - Users to register a new property
    - Users to add other members to their property
    - A central dashboard that all members of a property can view, offering statistics gathered from IoT sensors installed around their property. Users can select each IoT sensor, which demonstrates statistics from the following list:
        - Measure of water level (for instance livestock troughs, and property water tanks)
        - Measure of the weight of feed (for instance livestock paddock feeders, and property barn or silo stores)
        - Measure of electricity that can be emitted in an emergency generator (indicating the functioning of a backup generator).
        - Measure of soil moisture (returning insight into flood and drought)
        - Measure of atmospheric temperature (returning insight into bushfire)
- 'My Community dashboard, that offers:
    - Users to post a 'Request for assistance' where other users can reach out if they have the tools or labour to assist the Requester
    - Users to join a 'Community' where they can receive information and alerts from the 'Community board'. Additionally, government and meteorological personnel can insert official updates.
    - Community's can have or open a collaborative voice channel where users can quickly communicate in a 'walkie-talkie' fashion

CURRENT SYSTEMS EVALUATION

Our market analysis involved exploring and evaluating the benefits of existing systems in the climate extremes and community-driven domain, affording insight into the current industry processes. By reviewing equivalent systems that offer a property dashboard, as well as less-explicit applications such as social media platforms, our research facilitates a multifaceted approach. The section presents a comprehensive analysis through a systematic matrix for each system, following by an expanded instance of shots of each interface.

MY PROPERTY

Existing Systems

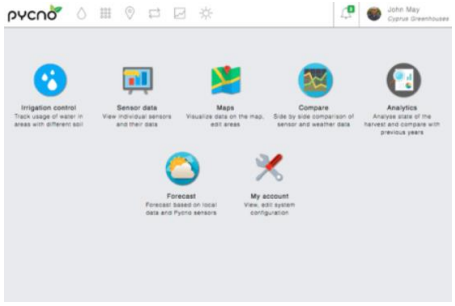
Criteria	Pycno	All METEO	Fires Near Me NSW
Summary	The Pycno app has a somewhat efficient user interface, as although the dashboard labels provide useful functionality and suits the apps' style, it is convoluted and can possibly be shortened to better suit the user.	The All METEO app has an efficient user interface as it has a minimalistic design whilst still providing the user with the appropriate functionality needed to suit the users' needs. The minimal amount of effort a user needs to navigate the app is effective and provides a better experience for the user.	The Fires Near Me NSW app has a somewhat effective user interface, as although the 'main screen' it presents to the user provides useful information, the app also has too many labels to suit the purpose of the app and the users in general. If the labels were condensed into fewer, more efficient labels, it would prove a better user interface.
Colour Palette	 #2596be, #0493a4, #fdfdfd, #2c4424, #d6d8d8	 #2c4424, #e4e4e4, #fcfdfc	 #C3dbbc, #4c94db, #ebf14b #f3f3f3, #d2242d,
Dashboard Labels	Irrigation control, sensor data, maps, compare, analytics, forecast, my account. The app has efficient naming and placing of labels; however, it could be confusing as there are quite a few labels. This is only acceptable as the app is quite large, however it could be condensed to allow better navigation.	Dashboard, my stations, world, preferences, logout. There is a good use of labelling placement and naming conventions, and has a minimal number of labels which enable efficient navigation of the app.	Incidents, my watch zones, emergency information, fire danger ratings, hazard reductions, map legend, disclaimer. There are too many labels for an app specifically designed to efficiently assist users in an emergency, however the labels are informative and able to allow users to navigate a larger app if needed.
Navigation Style	User can somewhat efficiently navigate through the app by clicking on each label on the dashboard. The only	Through the appropriate number of labels and naming conventions of them, the user can effectively	The layout of the app is simple yet works well, as the user can navigate easily through the app. The functionality of the app

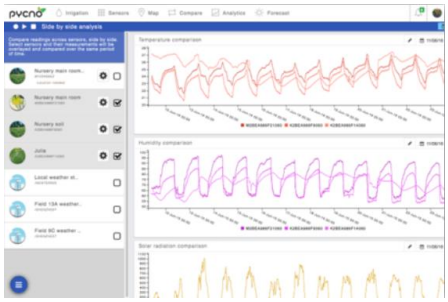
	issue is that since there is a considerable number of labels, it can lead to less efficiency when navigating the app.	navigate the app. The layout is straightforward, and the navigation style is satisfactory.	allows the user to properly navigate app in an emergency or when needing quick results, thus being efficient in app navigation.
Dashboard Interface	The dashboard is set out appropriately and displays relevant information for the user. It is effective at displaying different tools/parts of the site the user can interact with.	As the style of the app is minimalistic, the dashboard is intuitive and easy to use, as there aren't many options in the first place. This is effective, as it allows the user to quickly navigate the app and is visually pleasing.	There isn't a dedicated dashboard for the app, however the 'main screen' displays the most important information the user needs and is therefore providing the user with an effective user interface. The main screen suits the apps' functionality and is useful for the user.

SYSTEMS SCREENSHOTS

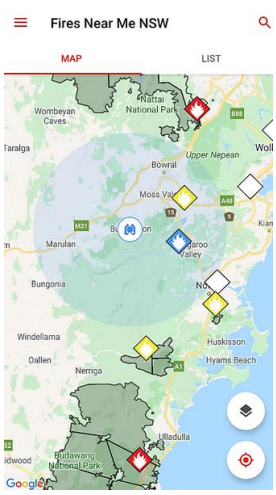
All METEO

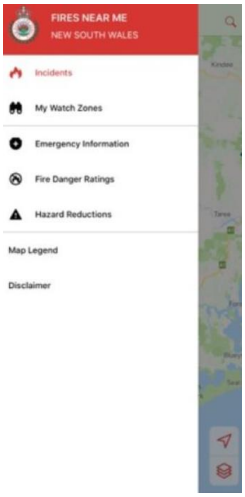
Pycno





Fires Near Me NSW





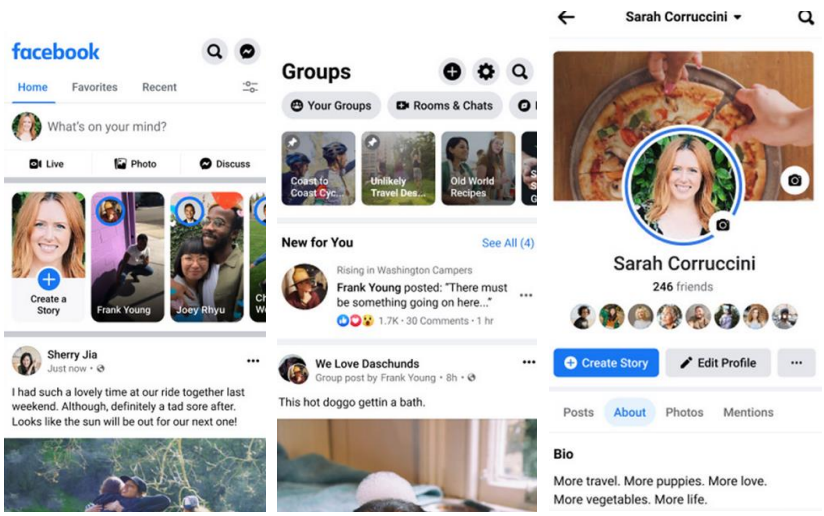
MY COMMUNITY

Existing Systems

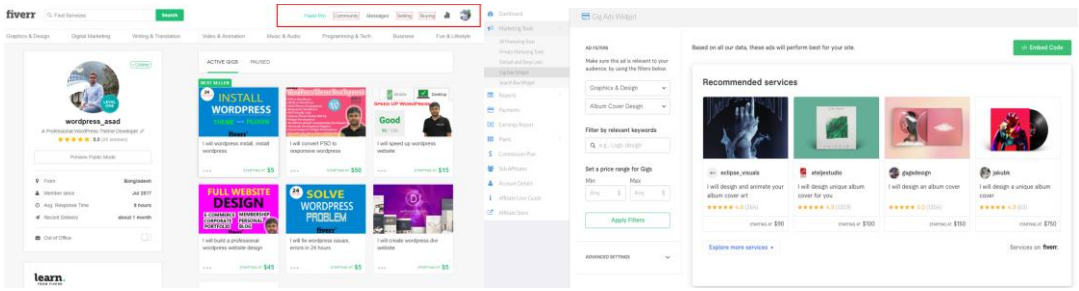
Criteria

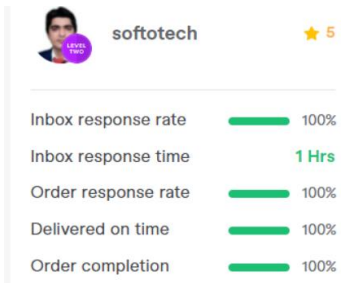
	Facebook	Fiverr
Summary	The Facebook app is a professional and well-made service which has an almost perfect user interface/ experience overall. The personalised recommendations suited towards users for listings appropriate to the user is useful and convenient. The general community driven aspect of the site enables professional and safe selling/ purchasing of products, overall making Facebook a near perfect example of good user interface/ experience.	Fiverr is a well-made and visually pleasing community-driven app which is suited to those offering specific services and those in search for specific services. Fiverr has made searching for services simplistic and easy to understand, whilst also enabling users to provide feedback through a rating system which can further allow for better customer experience. The site has made an intuitive app which benefits both sellers and buyers, demonstrating a great user experience.
Colour Palette	#2596be, #34548c, #fbfbfb	#f4fcfc, #1cbc44, #fbfbfb
Post Types	When posting, you're able to attach videos, photo's text and tag friends. In the marketplace, you're able to post photos of the product, specifics, location and contact number of sellers. The app supports a broad range of capabilities that are presented to the user provide an effective user interface, allowing for a better community environment for the user.	The website allows you to post listings including the type of service available, cost, specifics, reviews and more. As a seller, the user can appropriately post content advertising their listing and allow it to reach appropriate users. It is a site which hosts comprehensive listings of services available to the user and is overall an incredibly useful service with a great user interface.
Request Format	Users can, instead of purchasing the product directly, interact with sellers to meet up, discuss the product, or see similar ones through the marketplace. The buyers are also able to make a	Users can specifically request certain things when they purchase services. Although the interaction between buyer/seller in the beginning isn't too broad, the site encompasses anything needed for buyers to purchase services that can include more

	request/ offer to the seller, engaging with the community and all its members.	specific if the seller is able to provide it. This allows for stronger interaction between buyers and sellers, which is effective for the user experience of Fiverr.
Profile Display	In the Facebook Marketplace, users can see the sellers' reviews to determine whether other buyers recommend them, or if there are any issues with the seller. User profiles are also a way to interact more with the community, by allowing others to see personal info if the user decides to have a public profile.	Through Fiverr, users can see a sellers' profile which includes previous ratings from other buyers, as well as listings and a detailed description of the buyer and their background, I.e if a seller is teaching computer science, their profile would state they have a degree in computer science. This feature both promotes integrity through the community and allows for a better user experience.
Discovery System (How users discover useful profiles)	Users are recommended to others who may seem similar or useful to them. Advertisements in the marketplace are also directed towards users who have searched for similar listings. Users are also recommended both other users' profiles and groups like the user, allowing for a closer community and an efficient discovery system.	Fiverr recommends listings and buyers to users who have searched for similar aspects. It is an effective system which creates a more tailored environment for its user base which creates a better user experience. If a user searches for a term, they are more likely to be targeted by more appropriate advertisements and listings which can assist the user in finding what they need.
Feed Layout	The users feed allows for targeted advertisements, enabling users to better come across listings that suit them. The feed layout for Facebook is useful and efficient, it allows users to also set filters to enable better feed results. Overall, the feed layout adds on to Facebook's professional user interface.	There isn't much of a traditional feed for this site, however there is a dashboard with a wide variety of recommendations for users of different listings and users. Whilst the 'feed layout' is minimal, Fiverr makes up for it by having a strong recommendation system to suit the users and is therefore still providing users with a proper user interface and a considerable user experience.
Ranking System (How users are incentivised to use the system)	There is a 'star system' in place which ranks sellers on specific criteria such as price of the product they are selling, friendliness, product description and more. This ranking system allows for a better user experience and is a useful add on to Facebook.	Much like similar apps, sellers can be rated by buyers on specific criteria such as the services they are selling, if the buyer liked the seller, if the services lived up to expectation, etc. This system provides a better user experience and is assists the community in providing a more professional and secure platform to conduct business on.


facebook
Home
Favorites
Recent
What's on your mind?
Live
Photo
Discuss
Create a Story
Frank Young
Joey Rhyu
Sherry Jia
Just now
I had such a lovely time at our ride together last weekend. Although, definitely a tad sore after. Looks like the sun will be out for our next one!
Groups
Your Groups
Rooms & Chats
Coast to Coast Cyc...
Unlikely Travel Des...
Old World Recipes
New for You
See All (4)
Rising in Washington Campers
Frank Young posted: "There must be something going on here..."
1.7K · 30 Comments · 1 hr
We Love Daschunds
Group post by Frank Young · 8h
This hot doggo gettin a bath.
Sarah Corruccini
246 friends
Create Story
Edit Profile
Posts
About
Photos
Mentions
Bio
More travel. More puppies. More love.
More vegetables. More life.

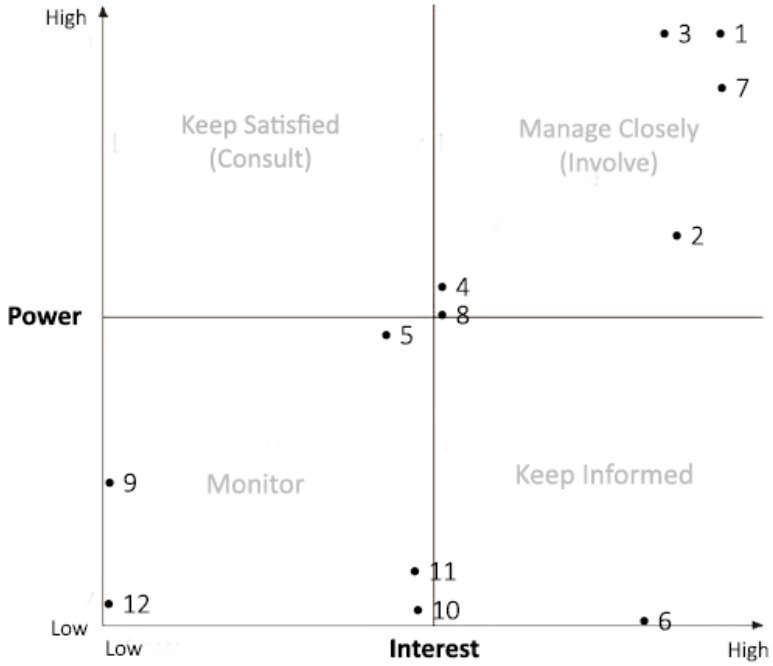
FIVER


fiverr
learn
Recommended services


softotech
5
Inbox response rate
100%
Inbox response time
1 Hrs
Order response rate
100%
Delivered on time
100%
Order completion
100%

STAKEHOLDER ANALYSIS

All stakeholder groups relating to cultivate are identified and considered through classification of their needs. This is firstly presented graphically on a Power-Interest Matrix (**APPENDIX A: Power-interest grid**), which offers a framework to organise and manage each stakeholder appropriately. In the following section, personas, fictional characters who represent an archetype of each primary user, will be presented. These are represented on the power-interest grid with a small version of their profile images. Following this framework, an explanation of needs for the user, data gathering techniques, and a professional list of data gathering is questions is presented for each stakeholder.



1. **Australian property owners**
2. **Additional members that operate the same property**
3. **Community members who may not own a large property but use the system to connect**
4. **Meteorological personnel, government departments and emergency responders**
5. **External property owners and investors**
6. **Rural communities**
7. **Cultivate development team**
8. **Insurance and banking personnel**
9. **Additional Non-User Stakeholders (Friends and family of property owners and local businesses)**

USERS - PRIMARY

Our primary users are Australian property owners, additional family and farmhand members who reside on the property, and community members who do not own a property with IoT sensors installed, but use the system to help other communities and remain informed on community events.

AUSTRALIAN PROPERTY OWNERS:

Considering that the average age of farmers in 2018 was 59 (Marwood & Hicks & Berry, 2018),, the demographic of Australian property owners ranges from 40 to 70 years old. Our specific target are users located in rural areas that are affected by climate change and climate extremes, such as Coomba, NSW; and Stanthorpe QLD.

Through our use of the power-interest grid, Australian property owners would have a high power and high interest. Thus, the optimal management strategy is to manage closely (involve). This evaluation is represented in our choice of data gathering techniques below.

Requirements:

- These users need Cultivate to offer them information on their property in a clear and consistent way.
- Be able to frequently interact with Cultivate to access the information for each sensor's collect on the property dashboard, in a rapid and precise way.
- Be able to frequently interact with the 'My Community' to submit requests and aid their local community, as well as get information from community leaders and meteorological personnel. This is particularly relevant as 'the inherent nature of the agricultural occupation is independent' (Karbowski, 2020)
- May possess a low to moderate experience with technology, though generally lower due to their busy schedules and remote location. Thus, the application must be suitable for users with a varying range of experience.

Data gathering techniques:

- Conducting face-to-face interviews with property owners in the NSW region (such as Camden), and online interviews with more remote owners.
    - Effective and engaging method for data gathering. Higher cost to execute, though rich data is gathered. As property owners are primary users, the effort is justified.
    - Recorded with audio and video if the participant is comfortable, otherwise the interviewer notes can take notes following the interview. By taking notes after the interview has been complete, it ensures that interviews are engaged while asking questions.
    - Semi-structured interviews will facilitate greater information given the user-focused and rich nature of the domain. In this way, owners can feel confident to share personal stories that illustrate the effect of climate extremes on their property. This will allow us to understand the emotional requirements of our interface. Further, the combination of closed questions to gain quantitative data and open questions to collect qualitative information will allow the data to be compared and replicated, whilst offering some exploration.

        - Further, property owner's will feel greater comfort in controlling the length of the interview
    - Active observation will also be used to engage property owners through a property tour, optionally as they complete their daily checklists, offering insight on their regular routines. Passive observation while we are on site will offer insight into logistical information required for the application, such as IoT sensor placement and the structuring of important dashboard elements.
    - Additionally, revising business documents will offer insight into the quantitative financial and asset impacts of natural disaster.

Questions:

Closed questions (offering quantitative data):

- Do you own a property?
- What is the size (in hectares) of your property?
- Do you regularly check the status of generators, water troughs and feed storages on your property?
- How many devices do you own?
- How many agricultural streams of income do you have?
- Has climate change impacted your property?
- How many natural disasters has your property been affected by
- On a scale of 1 to 10, how prepared do you feel in the event of a bush fire?
- On a scale of 1 to 10, how prepared do you feel in the event of a drought?
- On a scale of 1 to 10, how prepared do you feel in the event of a flood?

Open questions (offering qualitative data):

- How has climate change and natural disaster affected your property?
- What does your daily routine involve? What tasks do you perform to regularly plan for climate extremes?
- How do you regularly check the status of generators, water troughs and feed storages on your property?
- What are frequent fears you face relating to climate extremes, and do you face them often?
- What information do you believe should be gathered to protect the property's condition?
- What repetitive tasks do you often neglect or complete ineffectively, but are imperative to protect your property?
- A climate extreme is predicted on your property. How do you prepare? Do you often have plans in place in advance?
- Following a natural disaster, how do you believe the damage affects your property?
- How do you perform the activities that protect your property?
- Do you often have the capacity to use your cell phone?
- What are the features on your farm that you check the most, for instance ground moisture, water trough level?

ADDITIONAL MEMBERS THAT OPERATE THE SAME PROPERTY (FAMILY AND FARMHANDS):

Additional family and farmhand property members encompass a demographic of ages 16 – 80. Family members are generally aged 16 – 80, encompassing younger and older generations. Farmhands aged 20-60 are also classified as 'additional members', interacting with the central property dashboard to help with managing the property.

Through our use of the power-interest grid, additional property members would have a high power and high interest. However, they would have slightly less power than property owners. The optimal management strategy is to manage closely (involve). This evaluation is represented in our choice of data gathering techniques below.

Requirements:

- These users need Cultivate to offer them information on properties that may be family-owned or related to their career. Thus, the data provided is critical due to their responsibility for a property that affects other people.
- Be able to frequently interact with Cultivate to access the information for each sensor's collect on the property dashboard, in a rapid and precise way.
- Be able to frequently interact with the 'My Community' to submit requests and aid their local community, as well as get information from community leaders and meteorological personnel. This is particularly useful for anticipating the needs of property owners who they work for.
- May possess a low to high experience with technology, though generally higher as farming is often a career or a chore leaving them time to be exposed to other tasks.
- Given the varying range of ages, the application must be suitable for users with a varying range of experience.

Data gathering techniques:

- To gather data from additional members that assist on a property, a combination of semi-structured face-to-face interviews with members that are available for a one-on-one interview, and group interviews with all members that work on one property will be most effective. Together, these techniques will offer insight into the dynamic of collaboration that supports properties at a suitable level.
    - The use of group interviews will offer a suitable level of cost and detail, given the large number of interviewers that all members working on one farm will encompass. This can collect information regarding how workers coordinate.
    - Semi-structured one-on-one interviews will offer an insight into each individual worker. This can collect more personal and emotional information.
- Active and passive observation during daily routine and in the case of minor farm emergencies will offer insight into the practical application of how these members work together.

Questions:

Closed questions (offering quantitative data):

- I find chores more productive cooperating with the other members on the property?
- On a scale of 1 to 10, how well do you find each member on your property cooperates to accomplish daily routines?
- On a scale of 1 to 10, how well do you find each member on your property cooperates during emergency events?
- Has your property group ever forgotten to do a task?
- Has your property group ever run into miscommunication issues?

Open questions (offering qualitative data):

- How do you distribute daily tasks and keep track of if they have been complete?
- How do you distribute emergency tasks and keep track of if they have been complete?
- How do you facilitate communication within your family?
- During emergency situations, how do you reach out to each other for help and stay updated on one another's' condition?
- Do you find it easy to communicate with other property groups?
- How do you manage the threat of climate extremes and natural disaster?
- How do you recover from climate extremes and natural disaster?

COMMUNITY MEMBERS WHO MAY NOT OWN A LARGE PROPERTY BUT USE THE SYSTEM TO CONNECT WITH OTHERS:

Local community members who may not own a property that is connected to Cultivate, but may regularly check their community board on Cultivate to assist their community are primary users of the system. They represent a demographic aged 20 – 60 who reside in communities that are frequently affected by natural disaster, often by having critical members who own farming properties.

Through our use of the power-interest grid, community members would have a high power and high interest. Thus, the optimal management strategy is to manage closely (involve). However, they would have slightly less interest than property owners. The optimal management strategy is to manage closely (involve). This evaluation is represented in our choice of data gathering techniques below.

Requirements:

- These users need Cultivate to connect them with their community. Thus, they bust be able to frequently and quickly interface with the 'My Community' section to submit requests, get information from and aid their local community. The interface must particularly offer a great sense of community.
- May possess a low to high experience with technology, though generally higher due to their experience with farming as career or chore, leaving them time to be exposed to other tasks. Thus, the interface should be customisable for varying technical levels.

Data gathering techniques:

- To gather data from active local community members, we will employ semi-structured sone-on-one interviews and focus groups with community members. This is most effective to gather data from a wide range and number of community members, offering a holistic view from the group meetings. Then, more personal stories to be focused on through semi-structured meetings.
    - Local community members can volunteer to participate in one-on-one meetings, reflecting the individual nature of installing the system.
        - Convenience sampling (Creswell 2012) refers to a technique where interviewees volunteered versus being selected. The use of convenience sampling enhances the quality of responses as users will be passionate about providing input.

Questions:

Closed questions (offering quantitative data):

- How big is your community?
- How many times a week do you visit your local town or community spaces?
- On a scale of 1 to 10, how much do you agree that community offers aid when needed?
- Do you feel a sense of community and belonging within your local town?
- Has your community been affected by climate disaster?
- Do you check meteorological information such as from the Bureau of Metrology?
- Do you often have left over stock and items in your garage that you don't use?

Open questions (offering qualitative data):

- How do you connect with neighbours and find a sense of community?
- Is there a central location such as a notice board you visit to find information about your community?
- A climate extreme is predicted within the community. How do you believe it impacts families and individuals in the community?
- Following a natural disaster, how do you believe the damage affects the community?
- During a natural disaster, how does your community coordinate help where citizens can?
- How do you offer your neighbours help?
- What do you do with left over stock and items in your garage?

USERS - SECONDARY

METEOROLOGICAL PERSONNEL, GOVERNMENT PERSONNEL AND EMERGENCY RESPONDERS

Our secondary users are personnel from institutions who will sometimes interact with the application to enter updates to community boards. This includes meteorological personnel from institutions such as the Bureau of Meteorology (weather related flooding and storms), State Emergency Services and NSW Rural Fire Service, who are able to interface with Cultivate to provide information on community boards regarding emergency information. Government personnel can also interact with our application to post critical notifications. Further, emergency responders may occasionally use the system communicate with property owners in the case of natural disaster. The demographic for these users is 20 – 60-year-old personnel. Emergency responders and meteorological personnel may have knowledge of the community they are providing information on, whereas government personnel may not and may reside in urban environments.

Requirements:

- As these users will occasionally interact with the application to enter critical data and push human-generated notifications, is essential that the interface is effective and rapid to use.
- Likely possess a moderate - high experience with technology given the innovative industries they work in. This means that the interface should be able to facilitate quickly entering information.
- The interface must incorporate short-cut processes to input data, as these users many updates may need to be posted or during an emergency, the interface must be forgiving under emergency circumstances

Data gathering techniques:

- Gather data from meteorological and government personnel and emergency responders can be achieved passive observation, group interviews and business documentation will be employed in conjunction.
    - Passive observation will be non-invasive given the important and sensitive nature of these workplaces, but will offer knowledge into the atmosphere of institutions during an emergency.
    - Conducting brief group interviews with employees will minimise the amount of effort and cost for data collection, whilst obtaining an appropriate level of insight into employees view on entering updates on the interface.
    - By reviewing the formal documentation and protocols that are followed in case of emergency, we can garner how to best integrate Cultivate into their existing workflow and application interface.

Questions:

Closed questions (offering quantitative data):

- What department do you help with in the event of a climate disaster?
- During an emergency event, how many updates do you often post: 1 – 5, 5 – 10, 10 – 20, 10 + updates?
- On a scale of 1 to 10, how well does your current information distribution systems support your workflow?
- How many climate disasters have you faced?
- Is flood, fire, or drought the most difficult event to assist with?
- Do you find communication with community members easy?
- How many systems do you input information into when an update is needed to be shared?

Open questions (offering qualitative data):

- What procedures do you follow to update interfaces with an alert in the case of emergency?
- What are your thoughts on the integration of your information into other systems?
- During an emergency, do you find entering data into your system effective? What changes would you make?
- If there were no physical limitations, what changes would you make to distribute information and aid more efficiently?
- How has climate change and climate extremes affected your perspective and experiences in your career?
- What type of information do you share when you do need to provide an update? Do you include attachments such as images and videos?

EXTERNAL PROPERTY OWNERS AND INVESTORS

Tertiary users of our system will be the external property owners who influence the funding and instalment of Cultivate. They typically invest in the property as another stream of income, but live remotely and hire farmhands to manage the farm. The demographic encompasses individuals, families, and corporations with users within a 40 – 70 years old age range. Generally, they will be located in metropolitan environments.

Requirements:

- As these users manage properties remotely, they require reassurance and security that their property is managed. Cultivate must offer them information on their property in a comforting way.
- However, as their routines outside of the property may be busy, or they may manage multiple properties, they must be able to occasionally interact with Cultivate to access the information for each sensor's collect on the property dashboard, in a rapid and precise way.
- Be able to interact with the 'My Community' section where necessary if issues on the remote property needs assistance.
- May possess a moderate- high experience with technology, given their remote nature. Thus, the application should provide relevant information with options for greater insight into their property.

Data gathering techniques:

- Collecting information will be best accomplished by conducting one-on-one interviews and questionnaires. Due to the personal nature of funding and investing, semi-structured interviews would offer an insight into the emotional side of the interface and how to offer their business security.
- Given the busy nature of investors however, the use of concise questionnaires to gather qualitative and quantitative information will be a more suitable approach.

Questions:

Closed questions (offering quantitative data):

- Do you find it fulfilling to own a property remotely?
- How many investment properties do you own?
- How many properties do you remotely own and manage?
- Have you invested in a management application before?

Open questions (offering qualitative data):

- Why have you purchased the investment or passion property?
- Why do you manage your property remotely? Do you trust the operation of your property from an external location?
- How do you manage the threat of climate extremes and natural disaster remotely?
- How do you recover from climate extremes and natural disaster remotely?
- What motivates you to invest in an application that assists you in managing your remote property?

NON-USERS

Non-user stakeholders are individuals who have interest and influence in the project, though will not use the new system directly. Non-user stakeholders of our system will be rural communities, the development team of the Cultivate interface, and the family friends of property owners who use Cultivate.

Data gathering techniques for non-user stakeholders:

To collect information from all of the following non-user stakeholders, the use of concise-to-extended questionnaires will be used to rapidly gather qualitative and quantitative information.

- Given the busy nature of these stakeholders and moderate-low interest in the project, this technique enables participants to offer an appropriate level of detail. Further, the questions will be catered to each stakeholder group and context.
- An optional focus group could be conducted to gain further information from participants that are passionate about the issue and representative of other members who may be greatly influenced.
- If necessary (due to the additional effort and cost needed), passive observation can offer further depth into the practical workflow of these user groups.

RURAL COMMUNITIES:

Rural communities will be affected by the introduction of Cultivate and IoT on properties in their location. This user group refers to the demographic of small townspeople generally ranging in the age 20 – 80.

Requirements:

- Given the small-town and rural atmosphere that is often appreciated by townspeople, Cultivate's technology must be implemented in a way that emphasises its role to support than overtake.

Questions:

Closed questions (offering quantitative data):

- Do you fear the incorporation of technology disrupting the small-town atmosphere?
- How many property members do you know in your community?
- Has the impacts of climate change and natural disaster affected your community?

Open questions (offering qualitative data):

- What concerns do you have with the introduction of technology?
- How has climate change and climate extremes affected your perspective and experiences?
- How has climate change and climate extremes affected your community and its infrastructure?
- How do you manage the threat of climate extremes and natural disaster?
- How do you recover from climate extremes and natural disaster?

DEVELOPMENT TEAM OF CULTIVATE

The development team of the Cultivate interface will be invested in the success of the application. This user group refers to employees such as designers, programmers, and analysts, generally ranging in the age 20 – 60.

Requirements:

- The development team should feel inspired by the project, and empowered to produce their best work in the Cultivate interface.

Questions:

Closed questions (offering quantitative data):

- Do you find the development of Cultivate fulfilling?
- How many hours do you work on the Cultivate development?
- What field are you involved in for development?
- Is climate change and natural disaster a personal interest for you?
- Have you been impacted by the events of climate change?

Open questions (offering qualitative data):

- What motivates you to develop on the Cultivate team?
- If there were no physical limitations, what changes would you make to distribute information and aid more efficiently?

Insurance and banking personnel

Furthermore, insurance and banking personnel (aged 20 – 60) are affected in terms of reimbursing the property impacts from climate disasters, and influencing property purchase and retention.

Requirements:

- Insurance and banking personnel require accurate ways to prove natural incidents. As Cultivate offers useful assisting evidence

Questions:

Closed questions (offering quantitative data):

- What part of insurance and banking data do you prove?

Open questions (offering qualitative data):

- What motivates you to develop on the Cultivate team?

Additional non-user stakeholders (friends and family of property owners and local businesses).

Broadly, family and friends (aged 20 – 80) in international locations will feel a sense of reassurance and community, and local businesses (aged 20 – 50) will also be affected by the more reliable and quality supply of local produce.

PERSONAS

Personas refer to a design technique devised by Cooper (1999), encompassing fictional characters who are created to represent user types (stakeholders that use an application in a similar way). Our creation of personas resulted in four fictional characters based upon our stakeholder analysis, featured in the previous section. Additionally, an archetypes refers to a classification that a persona represents a typical version of, for instance 'Property owner of a family-run farm'. The creation personas greatly assist in consolidating and understanding out research, and representing the use of our system in a practical way, acting as fictional characters in the subsequent sections of scenarios and storyboards.

NAME: MICHAEL TUCKER

"Nothing like farm and family life"

Archetype: Property owner of a family-run farm

Demographic

Age:	40 years
Location:	Scone, Upper Hunter Shire, NSW
Gender	Male
Education	Primary school and high school education at his local facilities.
Job	Farming both crops and dairy cattle
Family	Mitchell (12) Chad (13) Debbie (28) Ted (82)

Biography

Grew up on his family's small property with two sheep and two cattle, attending school and revelling in the small-town Australian culture. At 18, gained employment as a farmhand. This introduced him to the agricultural industry, and fuelled by his passion for the Australian land. At 22, he purchased large acreage advertisement on his town's local community board, and worked hard since to craft a flourishing livelihood and family.

Technological Experience

- Little technological experience, often busy attending to chores on the property
- Owns a Motorola G50, and occasionally uses the property desktop to conduct management duties.
- Often uses a walkie-talkie system to communicate with family members on the property.

Motivations to use Cultivate

- To support his family and preserve the longevity and harvest of their property
- Improve the quality of his produce
- Combat the struggle with climate extremes on his property as it faces droughts frequently

Property Size: 100 acres hectares

Property Objectives:

  - Breeds and maintains a herd of Jersey cattle for milking as a part of his profit sources
  - Produces a large harvest of wheat and canola.

Goals in using Cultivate

- To prepare for climate extremes in advance, and during emergencies remain updated in affected areas to manage his property accordingly.

Personal Goals

- Learn how to use technology to better understand the youth.
- Maintain and work at farms for years for the rest of his life.
- Keep buying land as he loves working with and on them.

- Through the use of IoT sensor and weather pattern analysis, prepare for drought
- Maintain a rapid overview of the status of property features, ensuring all farm chores are met
- Receive alerts in case of emergency, such as water tank leak
- Accurately measure the level of property features, such as water tank level

NAME: ANDY FALCONER

"I appreciate my cattle, but chores are the worst."

Archetype: Younger family member assisting on the property

Demographic

Age:	18 years
Location:	Julia Creek, NW QLD
Gender	Male
Education	Bachelor in Agricultural Studies from a local university
Job	Works on his family's farm.
Family	Jodie (44)

Biography

I have grown up on my families property, and throughout my life I've always found that I enjoy interacting with technology and am able to help those who are less developed with it. I'm currently studying in Agricultural Studies, and I found myself being recruited into a crop management job. I realised that it would raise efficiency greatly if I integrated my knowledge of technology and passion for farming by introducing an IOT system throughout the farm.

Personal Goals

- To have an in-depth knowledge on Agriculture.
- One day own my farm to integrate a powerful technological system into it.
- To become more social and interact/ assist others in the community.

Technological Experience

- Moderate-high technological experience, particularly using social media such as Instagram, Facebook, and Twitter
- Skill using desktop and laptop technologies for school
- Experience with some cattle monitoring interfaces, though haven't remained due to his sense of personal responsibility for his animals
- Owns an Iphone 8, Playstation, Xbox and Mac school laptop

Motivations to use Cultivate

- To alleviate the rigour of completing monotonous chores
- To find assurance and certainty that his animals can be well looked after in light of the recent unpredictable fire and flood seasons
- To connect with other community members from his local high school in a property-context, finding advice for looking after his cattle

Property Size: 40 hectares

Property Objectives:

  - Family-driven business raising, showing and selling cattle

Goals in using Cultivate

- To gain a sense of security regarding his cattle, being able to know the amount of feed that is remaining and water levels in their water troughs.
- Gain a sense of certainty with the upcoming extreme weather seasons, allowing him to remain updated on his community's information board supported by meteorological sources.
- Find advice and tools offered for his cattle and their highest quality.

NAME: ALYSSA BROOKS

"Share the serenity of rural living"

Archetype: Farm manager

Demographic

Age:	37 years
Location:	Gatton, S.E Queensland
Gender	Female
Education	Possess a Bachelor in Agricultural studies from
Job	High-level farmhand who assists with crop management for a remote property owner who uses produce as another income stream.
Family	-

Biography

Alyssa has revelled in rural living since her small family estate. As she grew, she aspired to combine her joy of the outdoors with her career, motivating the ideal opportunity when the well-off Johns family purchased a property in her town. Aspiring to own the property as a personal investment, Alyssa applied to be Property Manager. Since, she has worked on the farm for over 15 years.

Personal Goals

- Learn more about handling IOT and technology in general.
- To continue to live the rest of her life on the farm.
- Make more money following her passions.

Technological Experience

- Moderate technological experience from her studies at her local University and friendships with co-students who moved to the urban environment
- Familiar with social media platforms such as Facebook, but doesn't find a lot of time to use them.
- Owns a Samsung Galaxy, and operates the property Windows computer for management purposes

Motivations to use Cultivate

- Struggles to use IOT every day, as it requires too much maintenance and overlook.
- Is motivated to try an application to have a proper IOT set up and to make it easier for her to manage her farm.
- Wants an easier to use platform and to communicate with other community members more easily.
- Wants an easy-to-use platform to communicate and connect with other community members in a professional farm-related manner.

Property Size: 10 hectares

Property Objectives:

- Growing lucerne, carrots and lettuce

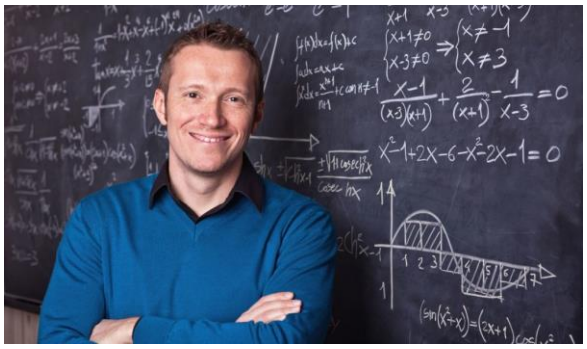
Goals in using Cultivate

- To improve her property management skills and preserve the John's income stream
- To create a better world through a strong farm output, and particularly the status of local producers
- To connect her community through her passion for the rural location

NAME: DAVE KNIGHT

"Community is central to solution"

Archetype: Active community member and environmentalist



Demographic

Age:	26
Location:	Bairnsdale, East Gippsland, Victoria
Gender	Male
Education	Completed a Bachelors in Mathematics Teaching at an urban university
Job	Teaches Mathematics at the local primary school
Family	Alice (26) Nick (2)

Biography

I have lived in a small town all my life, and I enjoy the simple things like interacting with my friends, teaching and learning mathematics, and giving back to the community. Once I completed my degree, I decided to follow my passions, as I have many connections to those who own farm properties and would like to assist those who have been affected by climate change.

Personal Goals

- Assist others and interact socially with them to connect more.
- Preserve the rural lifestyle and bring his passions into others' lives.
- Fight the increasing effect climate change has especially on rural farms

Technological Experience

- Is well-versed in technology given his education at an urban university
- Appreciation for the potential of technology to shift industries
- Leads mathematics classes that leverage graphing and statistical modelling through technology
- Owns an iPhone 11 and Windows laptop
- Uses Facebook marketplace occasionally to sell items.

Motivations to use Cultivate

- Does not own a property, however is connected to many community members who own them
- Passionate for the environment and the damage he has witnessed from climate change. Given his educated status, hopes to bring urban benefits to preserve rural lifestyle and community
- Is motivated to try our application as he would like to help his local community expand in terms of educational opportunities

Goals in using Cultivate

- Hopes to improve the local community by offering tools and aid on his weekends off work
- Facilitate individuals to build new connections by making suggestions to users of other users who could assist
- Assist in moderating the information board for his community, creating educational combined with fun posts to appeal to a younger demographic

SCENARIOS

A scenario is a simple informal narrative that represents one story about how an application may be used to achieve a certain goal. It incorporates the personas defined in the previous section to explore their contexts, needs and requirements, 'scenarios allow[s] the designer to explore the constraints… facilitators, and so on, that people operate under.' (Sharp & Rogers & Preece, 2019). Our system illustrates six scenarios that reflect the six core features of our application, introduced in the 'Application Summary'.

SCENARIO 1 - MY PROPERTY: REGISTERING A PROPERTY

The Tucker family reside in the Upper Hunter Shire region on a family-owned dairy property of three generations, enduring through severe drought. There are four family members: Michael (40), Debbie (38), Mitchell (12) and Chad (13). Their daily routines often involve important tasks to protect the property, such increasing the moisture on dry ground, checking their water stores for leakages, and checking the current livestock feed distributed around the paddocks stored in their barn. They often try to complete all of these tasks, but sometimes school and other responsibilities prevent their completion. One morning John is traveling past the community notice board when he discovers a pamphlet advertising a new application that assists with property management. He searches the application's name on the app store, and downloads it onto his Motorola G50. He registers an account and reads the contract for the IoT installation service, entering the initial set of requirements including the property address. Debbie, Mitchell and Chad also create accounts that are added to the farm dashboard as additional property members. Each member of the family subscribes to the Mudgee community information board on their individual profiles. The next day, an installer arrives and the IoT sensors are situated around the property. The installer completes the process, and each member receives a device-generated notification informing them of the successful connection of the new property to Cultivate, offering a tutorial for the application.

SCENARIO 2 - MY PROPERTY: VIEWING THE DASHBOARD

Every morning at 6:00 am, Andy (16) would wake up to his alarm to perform his farm chores before school. Although he is deeply passionate about the wellbeing of his cattle, he is finding less time to complete his chores with the increasing demands of school. Recently, his family successfully deployed the Cultivate application, adding his account as a member. This morning, Andy wakes up at 7.00 am. Andy checks if there are any urgent climate extreme warnings on his iPhone 8 lock screen. There are none, so as Andy gets ready for school, he proceeds to load the Cultivate application. This opens to his 'My Property' dashboard, which is designed so that all members of the family can interact easily and comfortable with it. Here, the interface presents a map that represents each IoT sensor as an icon, as well as a scrollable side-menu that categorises each of the sensors. The categories include water level (and moisture gauge), feed weight, generator charge and atmospheric temperature. Andy decides to use the map option. He navigates to each sensor placed on water troughs in the cattle paddock, clicking on the sensor icon which triggers an information box. Here, he can tell that none of the water level sensors record less than 1L of water, reassuring him that no troughs have developed leaks overnight, and that his cattle will remain watered for the day. Next, he navigates to each of the sensors placed under the hay that is placed around the paddock. They each record more than 5 kg of lucerne, indicating that his animals will remain content for the day. He also checks that the back-up electricity generator positioned behind the cattle stable is functioning currently (displaying sufficient electrical charge). Finally, he uses the interface to review the water level in the property water tank and the weight of hay that is stored in the barn, ensuring they

have enough stock for at least two weeks in the case of emergency. However, when viewing the dashboard, he noticed that one of the water troughs in the western quadrant was close to less than 1 L of water. Locating the correct sensor, he refills the particular trough before the local school bus passes his property.

SCENARIO 3 - MY PROPERTY: RECEIVING ALERT OF AN EMERGENCY EVENT ON THE PROPERTY

During droughts, Michael has a busy routine, prioritising his harvest of crops and operating his dairy farm to continue to sustain his business. Often, this leaves him a long list of tasks to complete in his routine that places property checks at the end of the workday. Through Cultivate, Michael has set up 'trigger' levels for each sensor measure: water moisture, water level, feed weight, generator charge and atmospheric temperature. This year, drought has been particularly challenging. One lunch time, Michael is relaxing after harvesting the best crops available during this difficult season and opens his Motorola G50. On his home screen, it displays two notifications from Cultivate: the water moisture has passed the lower limit of a sensor, and the atmospheric temperature has passed the 1000 °C[1] upper limit. He opens the Cultivate application, viewing the dashboard to locate the specific sensors where the emergency has occurred. If a sensor passes the triggering value, the IoT changes colour on his map and is pinned to the top of the dashboard, drawing attention. This allows him to isolate a small fire in the top-left quadrant of his property, as well as prioritise the limited supply of water to the wheat crops. Michael that he is managing the ground moisture sensor by clicking the check symbol, resolving the notification. On her device, Debbie (38) notices the unresolved temperature sensor. Immediately moving to the sensor, she extinguishes a small grass fire that could have intensified if it had not been noticed quickly, before clicking the check symbol.

SCENARIO 4 - MY COMMUNITY**:** UPDATE THE COMMUNITY BOARD WITH AN EMERGENCY

Alyssa often struggles to remain up to date with community news, as she is often busy revelling in rural life and managing the property for her external investor. This also means that she is often unaware of news relating to predicted weather forecasts and climate impacts. The owner of Alyssa's property installed the Cultivate application on the estate a week ago, which prompted Alyssa to make an account. While she was on the application, she subscribed to the Lake Conjola community board. One late morning after a series of rainy days, Alyssa opens her Google Samsung to notice an urgent notification from the community board. Clicking on the notification, it opens the Cultivate application and Lake Conjola community board page. Two announcements have been made regarding an unprecedented flood due to the overflowing of a local lake, one posted by a Lake Conjola board leader, and another by the Bureau of Meteorology. Pushing a notification to the other farmhands who are registered as members of the property, Joe (31) and Sally (31). The farmhands assist in deploying the farm's flood preparation protocol. Additionally, Alyssa joins the emergency voice channel using push-to-talk that facilitates her movement around the property to install flood barriers. Here, the community members collaborate, instructing Alyssa on the optimal placement for the sand blockades.

SCENARIO 5 - MY COMMUNITY**:** CREATING A REQUEST FOR ASSISTANCE

Michael has been researching into the installation of drip-irrigation technologies to assist with optimising the use of water for his crops during a drought. Two days ago, a drip-irrigation system arrived from an online supplier. However, Michael was unsure of the best practices to correctly implement the system. Michael goes to the

[1] https://ecos.csiro.au/bushfire-in-australia-understanding-hell-on-earth/#:~:text=Inside%20the%20turbulent%20diffusion%20flames,order%20of%201600%C2%B0C.

Scone community board. Completing a new post layout, Michael enters a request name: 'Assistance installing crop drip-irrigation system', type 'Non-urgent expertise', Description: 'Looking for a community member with expertise or confidence with installing these systems', and the expiry-date in five days. Michael revisits the post in an hour. Five comments have been added with the 'Non-solution' tag. Although this means that none of the comments have solutions, they mention the names of other members who they believe have expertise in the field. After another hour, Michael checks his request and a member Steven Allen (34) has submitted a 'Solution' comment that they are 'happy to help for two hours'. Michael closes the thread and privately comments his address on the post to Steven. The next day, Steven assists Michael in installation, teaching Michael new crucial skills.

SCENARIO 6 - MY COMMUNITY**:** FULFILLING A REQUEST FOR ASSISTANCE

On weekends, Dave often scrolls through the Bairnsdale Cultivate community board to view if there are any requests that he can assist with. This weekend, a new request is posted on the information board by a member Dean Copperdale (49) with the title 'Looking for a spare fire blanket' and a 'Non-urgent tool' priority-request type tag. The request type tag is used to draw attention to users if it is an urgent post which would generally be made during a natural disaster In this case. Dave has a spare fire blanket in his garage, and comments on the post 'I can drop a spare wrench around this afternoon', selecting a 'Solution' comment type tag to indicate that he can provide the user assistance. Dean closes the request, and responds to Dave in a private comment 'My address is 1 Bairnsdale, East Gippsland Victoria'. Later in the day, Dave drops off the fire blanket and builds a friendship with Dean, who offers him a jar of property-made honey. Later, Dean adds a community point to Dave to express his gratitude.

USE CASES

USE CASE 1 – MY PROPERTY: REGISTER A NEW PROPERTY

Use case name:	Register a new property	
Scenario:	User installs app, creates account, and needs to register property.	
Triggering event:	User clicks on the 'register property' button on the dashboard.	
Brief description:	User first starts using software, wishes to input their property, and clicks on the 'register property' button. They then fill in the appropriate information to better calibrate software and store information that may be useful for emergencies.	
Actors:	The user.	
Related use cases:	Will possibly interact with cases 2 and 5, as will need to register a property to view it and possibly request assistance depending on the type.	
Stakeholders:	Client, users, and possible community members.	
Preconditions:	The user has to download the app. The user has to make a basic account. The user has to own a property.	
Post conditions:	Property size, location, members of property are saved. Product saved details are associated with user.	
Flow of activities:	Actor 1. User wishes to register their property and enters property information. 2. User confirms and saves information.	System 1.1- System ties user with their account. 1.2- System asks for user input. 2.1- System saves user input. 2.2-System ties input data to user data.
Exception conditions:	1.1- Address isn't valid. 2.1- Basic customer data from input isn't valid.	

USE CASE 2 – MY PROPERTY: VIEW THE PROPERTY DASHBOARD

Use case name:	View the property dashboard	
Scenario:	User views the property dashboard.	
Triggering event:	User clicks on 'property dashboard.'	
Brief description:	The user has their property information saved with the app and wants to view the property dashboard, so they click the 'property dashboard' button which displays the dashboard.	
Actors:	The user.	
Related use cases:	Will interact with use case 1 as they user needs a property in order to view the dashboard, and possibly use cases 5 and 6 as users may need to be able to view the property dashboard for assistance.	
Stakeholders:	Client, user, and possible community members.	
Preconditions:	The user has to own a property. The user has to submit all details of the property to the system.	
Post conditions:	Dashboard must be displayed to user including all stats, graphs and other necessary components.	
Flow of activities:	Actor 1. User wishes to view their property dashboard. 2. User clicks on dashboard button.	System 2.1-System recognizes user is clicking button to view dashboard. 2.2-System retrieves users' information regarding property. 2.3-System displays dashboard including graphs and stats with user information.
Exception conditions:	1.2- User doesn't have a property tied to their account. 1.3- System doesn't have information to display about user.	

USE CASE 3 – MY PROPERTY: NOTIFY USER OF EMERGENCY EVENT

Use case name:	Notify user of emergency event	
Scenario:	A registered user is notified of an emergency.	
Triggering event:	An emergency occurs.	
Brief description:	A registered user has signed up to receive notifications of emergencies near their property. An emergency has occurred, and the user is notified that their property may be in danger.	
Actors:	User, emergency services.	
Related use cases:	Use cases 1,2, 4, 5 and 6 are related as the user needs to have inputted the property details, then those details are available to be displayed in the dashboard, which are able to be seen when an emergency occurs, notifying all relevant users and updating the community board.	
Stakeholders:	Investors, client, users.	
Preconditions:	The user needs to have relevant information in account. The user needs to have a form of contact in case of an emergency.	
Post conditions:	Notifications must be sent by the preferred form of contact to user in an emergency.	
Flow of activities:	User 1- User chooses preferred form of contact. 2- User receives notification.	System 1.1- System stores preferred form of contact. 1.2- System ties form of contact to account. 2.1- System is notified there is an emergency. 2.2- System sends notification to affected users.
Exception conditions:	1.1- User doesn't have a form of contact to be notified. 2.1- System may not be notified. 2.2- Users may have notifications turned off.	

USE CASE 4 – MY COMMUNITY: UPDATE THE COMMUNITY BOARD WITH AN EMERGENCY

Use case name:	Update the Community board with an emergency.	
Scenario:	Community board is updated due to an emergency.	
Triggering event:	An emergency occurs.	
Brief description:	If an emergency occurs, the community board should be updated to assist those who are affected as well as notify	
Actors:	User, emergency services.	
Related use cases:	Use cases 1 and 3 are needed in order to properly update the community board in an emergency. Users' properties need to be tied with their accounts as it will affect how users are notified.	
Stakeholders:	Users, investors, client, emergency services.	
Preconditions:	Users need an account in order to take part in the community aspect. Users should have basic details tied to their account in order to be notified.	
Post conditions:	The community board should be updated due to an emergency. Should be available to all users, despite relevance. Must work asynchronous with notifications being sent due to an emergency.	
Flow of activities:	User 1- User sees notification that there is an emergency. 2- User posts emergency to community board. 3- Another user visits community board to see emergency notification.	System 1.1 System receives emergency notification. 1.2 System notifies user. 2.1 System displays community board to user. 2.2- System posts user post. 3.1 System displays community board to another user.
Exception conditions:	1.1 User may not see notification or post on community board. 2.2 System may not update community board for all users.	

USE CASE 5 – MY COMMUNITY: POST A REQUEST FOR ASSISTANCE

Use case name:	Post a request for assistance	
Scenario:	User needs to make a request for assistance.	
Triggering event:	User interacts with 'request assistance' button.	
Brief description:	If the user needs assistance, they interact with the 'request' assistance button which then displays a pop up for them to fill in which type of assistance they need. They will then post their request to the community.	
Actors:	User and possibly other community members.	
Related use cases:	Use case 6 is relevant as users need to be able to post and fulfil requests in this community-driven app.	
Stakeholders:	Investors of the app and members of community.	
Preconditions:	The user should have basic details saved to their account. The user should include relevant details in the post.	
Post conditions:	A post should be available to see on the community board. The post should contain the important details.	
Flow of activities:	User 1. User creates post and enters post request information.	System 1.1 System inputs post request. 1.2 System inputs requests tag. 1.3 System inputs description. 1.4 System inputs request.
Exception conditions:	1.1 User may not input the required information. 2.2 System may not input the required information. 2.4 System won't post the post.	

USE CASE 6 – MY COMMUNITY: FULFILL A REQUEST FOR ASSISTANCE

Use case name:	Fulfil a request for assistance	
Scenario:	User fulfils a request for assistance.	
Triggering event:	A request needs to be filled.	
Brief description:	A user notices a request needs to be fulfilled and decides to fulfil it. They input the required information and post the response to the request.	
Actors:	User and community members.	
Related use cases:	Use case 5 is related to this one as users need to be able to post and fulfil requests in this community-driven app.	
Stakeholders:	Users of the app and other members of the community.	
Preconditions:	The user should have basic details saved to their account. The user should input the required details to the request.	
Post conditions:	The user which made the request should receive the fulfillment request. The system should notify all parties.	
Flow of activities:	User 1. User sees post and enters fulfillment information.	System 1.1 System displays post. 1.2 System prompts user to input required information. 1.3 System sends and notifies the user who put in the request.
Exception conditions:	1.1 System may not display post for all users. 1.2 User may not send required information. 1.3 IOT sensors may not send required information to system.	

INITIAL REQUIREMENTS

Initial application requirements compiled from the research of this report. These are defined by the FURPS+ (Functional, Usability, Reliability, Performance, Security +) framework.

CORE FUNCTIONAL REQUIREMENTS:

1. The interface should enable a user to add a property and add other users as 'members' of one property
2. The interface should enable a user to view the dashboard for their property and navigate to the statistics of a particular IoT sensor, such as viewing the level of moisture in the ground, measuring the weight of livestock feed in paddock and silo stores, viewing the surrounding temperature or measuring the electricity in the generator.
3. The interface should invoke device-generated alerts when a sensor passes a custom-set trigger point, indicating symptoms of drought, flood or fire. The dashboard should display alerts when a task has not been completed.
4. The interface should enable a user to post a request for assistance to a specific community, filling out the:
    a. Request name
    b. Request priority-type tag (draws attention to users if it is an urgent post, generally made during a natural disaster)
    c. Description
    d. Request expiry-date
5. The interface should enable a community leader to 'host' a push-to-talk 'walkie-talkie' voice channel
6. The interface should enable a user to comment on an existing post

ADDITIONAL FUNCTIONAL REQUIREMENTS:

To manage issues of scope creep, a secondary list of initial requirements has been developed to assist in the creation of a holistic interface. This list can be prioritised or adjusted accordingly. Unlike the core functional requirements, these are opportunities to further develop our design.

- The dashboard should be accessible by all members of the property
- The interface should enable a user to view their profile and edit their name, short biography, picture and optionally property location.
- The interface should enable a user to view another user's profile
- The interface should enable a user to comment on another user's post, indicating whether the comment is a 'solution' to the request or 'suggestion' on leads to follow regarding the request
- The interface should enable a user to search for a community registered to the service and view their information board
- The interface should allow a community leader to post an update for the community, inputting a name, type, priority, description and location
- The interface should allow meteorological personnel to post an update for the community, inputting a name, type, priority, description and location
- The interface should enable a user to connect to a community's push-to-talk 'walkie talkie' voice channel
- The interface should enable a user to become a member of a community

DESIGN (NON-FUNCTIONAL) REQUIREMENTS

- **USABILITY**
  - **Inclusivity:** The interface is required to be highly usable for individuals of varying education, technological experience, and ages (given the family nature of the dashboard)
    - Interfaces should use consistent and judicious placement of Cultivate's logo
    - The interface should use conventional behaviours for a smartphone app, responding to gestures of sliding, pinching, and expanding screens
  - **High memorability and learnability:** The interface is required to be used by varying demographics, as well as the everyday nature that the application will be used
    - Incorporate a help button on each page, with a brief explanation as well as an expanded explanation
    - Attach a hyperlink to an online version of the complete interface documentation
    - Include an optional concise tutorial for a learning user
  - **High utility:** The interface must have high utility for rapid use in emergency situations, and efficient use in regular routines
    - Feature quick navigation buttons, such as to the community 'walkie talkie' audio room
    - Feature a button that can quickly send a request to community members for help regarding fire, drought, or flood
  - **High effectiveness and efficiency:** The interface must have high effectiveness and efficiency to ensure users can quickly and correctly navigate
    - The dashboard has a clear layout with each IoT status, and a quantifiable as well as graphical representation of the statistic
  - **High safety:** The interface must have safety mechanisms, given the critical nature of climate extremes and property management
    - The interface offers a warning if an IoT sensor detects it has malfunctioned or measured inaccurately
    - A user can update, delete, and archive Community Requests
    - Community updates must be confirmed before being posted
    - A reset password mechanism with a password code should be implemented

- **RELIABILITY:**
  - **High reliability:** The interface must have high reliability for a long lifespan, as it will be relied on in everyday routines, and due to the critical nature of information during a natural disaster.
    - Indicate the deployment lifespan of IoT sensors
    - Detect when losing connection with an IoT sensor and display a warning message on the interface
    - Alert other users connected to the property that a user has become disconnected
  - In the event of a climate extreme, high reliability must be maintained.
    - Display the most recent IoT sensor and community updates if the application fails

- **PERFORMANCE:**
  - **High performance:** Cultivate must perform quickly under a high load of users in the event of a climate emergency. The interface must have a rapid response time to user's actions and display real-time statistics from the IoT sensors, as well as ensure that changes to the dashboard are shared for all property users. Additionally, the push-to-talk 'walkie talkie' system should have high performance when many callers are connected.
    - Response time of <1 second for all button presses, including updating dashboard property statistics

- Support simultaneous client connections to the application
- Support simultaneous client connections to the dashboard
- Push-to-talk system facilitates <1 second communication delay

- **SECURITY**
  - **High security:** Given the sensitive nature of property information and the critical nature of climate extremes, accounts and data should be protected accordingly.
    - Accounts must have a be password-protected. The password should consist of at least 10 characters, a capital letter, a special character and a number. The password should be obfuscated while the user is entering data
    - Community pages should have the option of being 'private', whereby members must be verified to join
    - Profile pages should have the option of being 'private', whereby their public profile and location of their property is concealed.

- **EXTENDED FEATURES**
  - **Design constraints:**
    - Function on Apple (iOS) and Android operating systems
  - **Implementation requirements:**
    - The front-end application should be presented in Figma
    - Documentation should possess a formal level of ceremony (presentation), and appropriate system diagrams
  - **Interface requirements:**
    - The application should successfully interface with existing meteorological applications, allowing data to be processed and displayed accordingly on the Cultivate interface
    - The application should interface with OS photo and camera applications (for users to upload images to their requests) as well as collect audio input
  - **Physical requirements:**
    - The system generally has no physical requirements given that it employs the portability of smartphone technologies. However, the application must generally be usable from any location and in varying conditions. Thus, it must use contrasting colours and readable fonts, as in emergency situations it may be difficult to process information and navigate around an interface.
  - **Supportability Requirements:**
    - Automatic configuration for installers and default settings for users to maximise the interface's performance on OS hardware.
    - Offer option to automatically download updates from a support server
    - Include optional error reporting mechanisms for users to submit the error context and additional data

cultivate

Part B

PROTOTYPING AND DESIGN

Iterative Prototyping and Interfaces 46

Storyboards 48

Wireframes and Wireflows 55

Low Fidelity Prototyping 76

Discussion on acceptance of key design patterns 95

Design Iterations 98

Final Design 114

Review of Design 116

Evaluation Assessment 117

Testing Considerations 118

Conclusion 120

References 121

Appendices 125

Appendix A: Style Guide 125

ITERATIVE PROTOTYPING AND INTERFACES

A prototype is a draft model of a product which assists innovators in planning and gauging the feasibility of designs rapidly, as well as collect feedback from team members, clients, and other perspectives. The are defined by characteristics such as minimal detail and rapid speed in producing them. They are used across many industries from automobile design to supermarket product design. In Interaction Design, prototyping is central to developing alternatives as well as iterations for application interfaces. Prototypes allow alternative designs to be quickly drafted and compared with minimal time and effort cost. Once an option is followed, the design is built upon in further prototypes.

There are two classifications of prototyping technique:

- **Vertical prototyping**: Design a wide range of functions, however, each contain minimal detail
- **Horizontal prototyping:** Design a limited range of functions, but with in-depth detail for each

In the following models, a balance of vertical and horizontal prototyping has been followed, without a focus on either. This is achieved in the 'Wireflows and Wireframes' section onward, through the presentation of the overall diagram, followed by an explanation of each section and function presented.

There are two further classifications for the amount of detail provided in a prototype:

- **Low fidelity:** Contains minimal detail, typically used to plan functionality and rapidly changing aspects of the system.
- **High fidelity:** Greater detail, encompassing prototypes at a level of detail that is almost photo-realistic.

Although the level of detail presented in a prototype is measured on an indefinite scale, the following models presented in this section can be classified as follows:

- **Low fidelity:** Wireframes and Wireflows, Low Fidelity Prototype
- **Shifting from low fidelity to high fidelity:** Design Iterations
- **High fidelity:** Final Design

An initial list of core requirements for the system were compiled from the research of this report, defined in **Part A**. Each of the prototypes to follow show consideration of all of the following core requirements, and refine their implementations to create an effective interface, evaluated in the Review of Design section. The success of the final design is measured by achieving each criteria. The effectiveness is evaluated by how effectively the usability each implementation. To define, usability refers to "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" (ISO 9241-11).

The requirements are defined as the following:

1. The interface should enable a user to add a property and add other users as 'members' of one property
2. The interface should enable a user to view the dashboard for their property and navigate to the statistics of a particular IoT sensor:
    a. Viewing the level of moisture in the ground
    b. Measuring the weight of livestock feed in paddock and silo stores
    c. Viewing the surrounding temperature
    d. Measuring the electricity in the generator.

3. The interface should invoke device-generated alerts when a sensor passes a custom-set trigger point, indicating symptoms of drought, flood or fire.
    a. The dashboard should display alerts when a sensor has not been handled.
4. The interface should enable a user to post a request for assistance to a specific community, filling out the:
    a. Request name
    b. Request priority-type tag (draws attention to users if it is an urgent post, generally made during a natural disaster)
    c. Description
    d. Request expiry-date
5. The interface should enable a community leader to 'host' a push-to-talk 'walkie-talkie' voice channel
6. The interface should enable a user to comment on an existing post

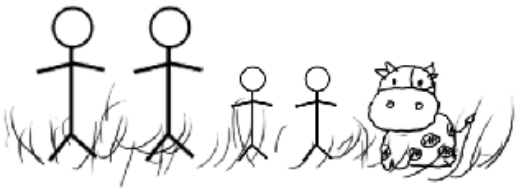
To manage issues of scope creep, a secondary list of initial requirements was developed to assist in the creation of a holistic interface. These are opportunities to further develop our design:

- The dashboard should be accessible by multiple members of the property
- The interface should enable a user to view their profile and edit their name, short biography, and picture
- The interface should enable a user to view another user's profile
- The interface should enable a user to comment on another user's post, indicating whether the comment is a 'solution' to the request or 'suggestion' on leads to follow regarding the request
- The interface should enable a user to search for a community registered to the service and view their information board
- The interface should allow a community leader to post an update for the community, inputting a name, type, priority, description and location
- The interface should allow meteorological personnel to post an update for the community, inputting a name, type, priority, description and location
- The interface should enable a user to connect to a community's push-to-talk 'walkie talkie' voice channel
- The interface should enable a user to become a member of a community

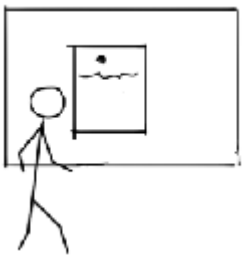
STORYBOARDS

The following subsection presents storyboards, which represent a sequence of actions that a user process through to achieve a goal (Sharp & Rogers & Preece, 2019). This offers a graphical representation of ways that the system will be used. Each of these storyboards have been generated by breaking down the scenario contained in the Part A section into a series of scenes. Used in conjunction with its scenario, our storyboards allow interactions with the system to be understood. Each storyboard frame contains detailed descriptions to explain the process. In the following 'Wireframes and Wireflow' section, the interface implements the requirements from these storyboards.

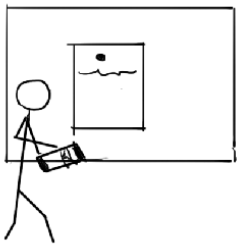
STORYBOARD 1 - MY FARM: REGISTERING A PROPERTY



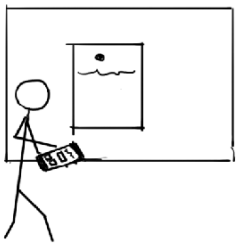
1. The Tucker family reside on a family-owned dairy property of three generations, enduring through severe drought There are four family members: Michael (40), Debbie (38), Mitchell (12) and Chad (13).



2. One morning John is traveling past the community notice board when he discovers a pamphlet advertising a new application.

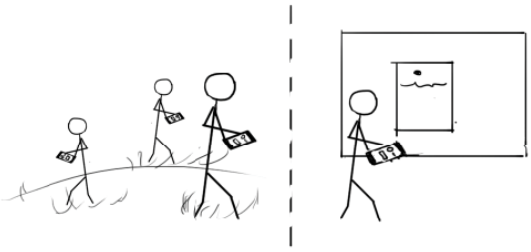


3. He searches the application's name on the app store, and downloads it onto his Motorola G50.

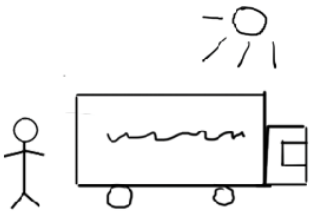


4. He registers as a property owner and reads the contract for the IoT installation service, entering the initial set of requirements including the property address..

5. Debbie, Mitchell and Chad also create accounts that are added to the farm dashboard as additional property members.



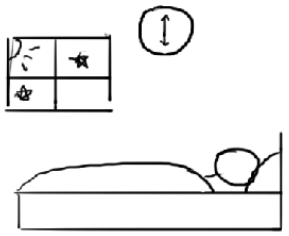
6. Each member of the family subscribes to the Mudgee community information board on their individual profiles.



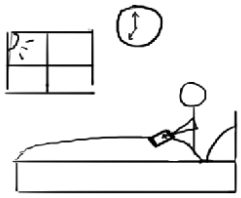
7. The next day, an installer arrives and the IoT sensors are situated around the property.

8. The installer completes the process, and each member receives a device-generated notification informing them of the successful connection of the new property to Cultivate, offering a tutorial for the application.

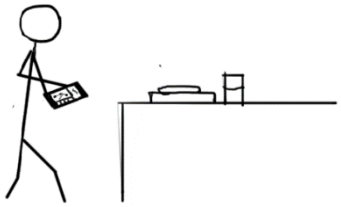
STORYBOARD 2 - MY FARM: VIEWING THE DASHBOARD



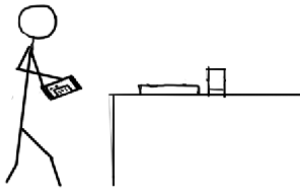
1. Every morning at 6:00 am, Andy (16) would wake up to his alarm to perform his farm chores before school.



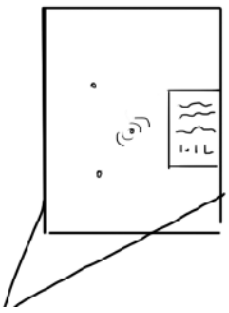
2. Recently, his family successfully deployed the Cultivate application, adding his account as a member. This morning, Andy wakes up at 7.00 am. Andy checks if there are any urgent climate extreme warnings on his iPhone 8 lock screen.



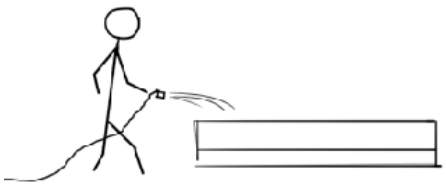
3. As Andy gets ready for school, he proceeds to load Cultivate, opening to his 'My Property' dashboard. Andry decides to use the map option to target each sensor.



4. Andy clicks on each sensor icon to trigger an information box for water troughs, weight of feed in paddocks, electricity generator charge, water tank level and barn feed store.



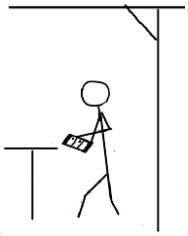
45 When viewing the dashboard Andy notices that one of the water troughs in the central quadrant was close to less than 1 L of water. He remembers the location of the sensor. (The above illustration indicates stick figure's hands)



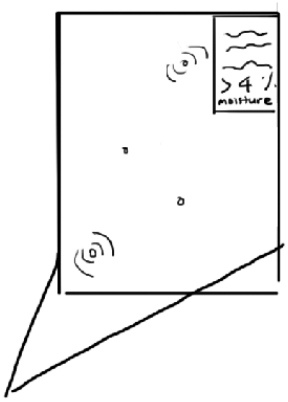
6. He refills the particular trough before the local school bus passes his property.

STORYBOARD 3 - MY FARM**:** SEND ALERT OF AN EMERGENCY EVENT ON THE PROPERTY

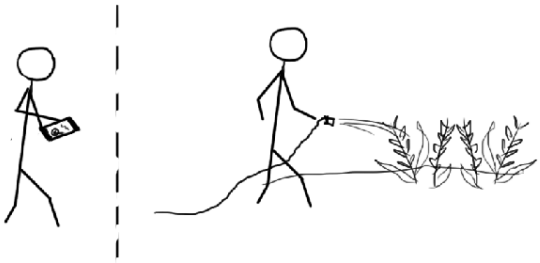
1. Michael sets up 'trigger' levels for each sensor measure: water moisture, water level, feed weight, generator charge and atmospheric temperature.



2. One lunch time, Michael opens his Motorola G50. On his home screen, it displays two notifications from Cultivate: the water moisture has passed the lower limit of a sensor, and the atmospheric temperature has passed the 1000 °C upper limit.



3. He opens the Cultivate application, viewing the dashboard to locate the specific sensors where the emergency has occurred. (The above illustration indicates stick figure's hands)



4. Michael indicates that he is managing the ground moisture sensor by clicking the check symbol, resolving the notification.

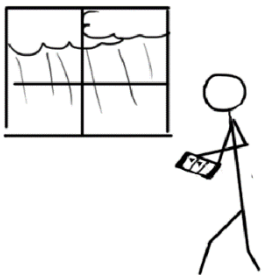
5. On her device, Debbie (38) notices the unresolved temperature sensor.

6. Immediately moving to the sensor, she extinguishes a small grass fire that could have intensified if it had not been noticed quickly, before clicking the check symbol.

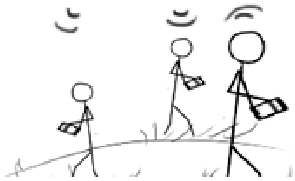
STORYBOARD 4 - MY COMMUNITY**:** UPDATE THE COMMUNITY BOARD WITH AN EMERGENCY

1. Alyssa often struggles to remain up to date with community news, as she is often busy revelling in rural life and managing the property for her external investor.

2. The owner of Alyssa's property installed the Cultivate application on the estate a week ago, which prompted Alyssa to make an account and she subscribe to the Lake Conjola community board.



3. One late morning after a series of rainy days, Alyssa opens her Google Samsung to notice an urgent notification on the Lake Conjola community board page. Two announcements have been made regarding an unprecedented flooding from the overflow of a local lake.

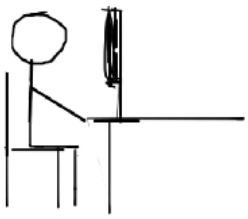


4. Alyssa pushes a notification to the other farmhands, Joe (31) and Sally (31).

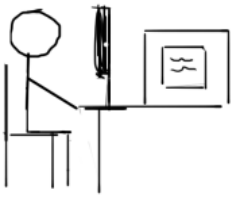
5. The farmhands assist her in deploying the farm's flood preparation protocol.

6.Additionally, Alyssa joins the emergency voice channel using push-to-talk that allows her to use the system while moving around the property. Community members collaborate, instructing Alyssa on the optimal placement for the sand blockades.

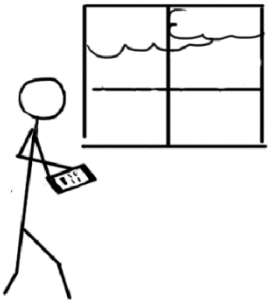
STORYBOARD 5 - MY COMMUNITY: CREATING A REQUEST FOR ASSISTANCE



1. Michael has been researching into the installation of drip-irrigation technologies to assist with optimising the use of water for his crops during a drought.

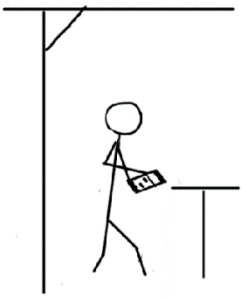


2. Two days ago, a drip-irrigation system arrived from an online supplier. However, Michael was unsure of the best practices to correctly implement the system.



3. Michael goes to the Scone community board and completes a new post layout, entering data for the request name, type (non-urgent expertise), description and request expiry-date)

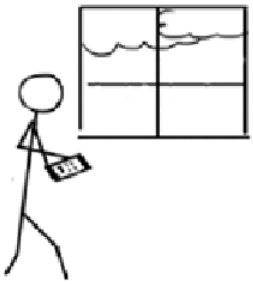
4. Michael revisits the post in an hour. Five comments have been added with the 'Non-solution' tag. Although this means that none of the comments have solutions, they mention the names of other members who they believe have expertise in the field.



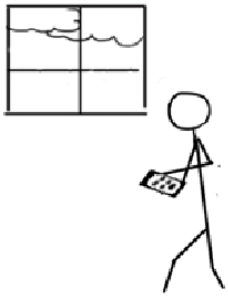
5. After another hour, Michael checks his request, and a member has submitted a 'Solution' comment that they are happy to help for two hours. Michael comments his property address.

6. The next day, Steven assists Michael in installation, teaching Michael new crucial skills.

SCENARIO 6 - MY COMMUNITY: FULFILLING A REQUEST FOR ASSISTANCE

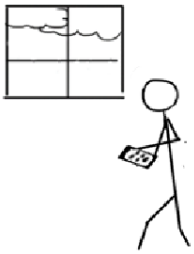


1. On weekends, Dave often scrolls through the Bairnsdale Cultivate community board to view if there are any requests that he can assist with.



2. This weekend, a new request is posted on the information board by a member Dean Copperdale (49) with the title 'Looking for a spare fire blanket' and a 'Non-urgent tool' priority-request type tag.

3. Dave has a spare fire blanket in his garage, and comments on the post 'I can drop a spare wrench around this afternoon', selecting a 'Solution' comment type tag to indicate that he can provide the user assistance.



4. Dean closes the request, and responds to Dave in a private comment 'My address is 1 Bairnsdale, East Gippsland Victoria'.

5. Later in the day, Dave drops off the fire blanket and builds a friendship with Dean, who offers him a jar of property-made honey.

6. Finally, Dean adds a community point to Dave to express his gratitude.

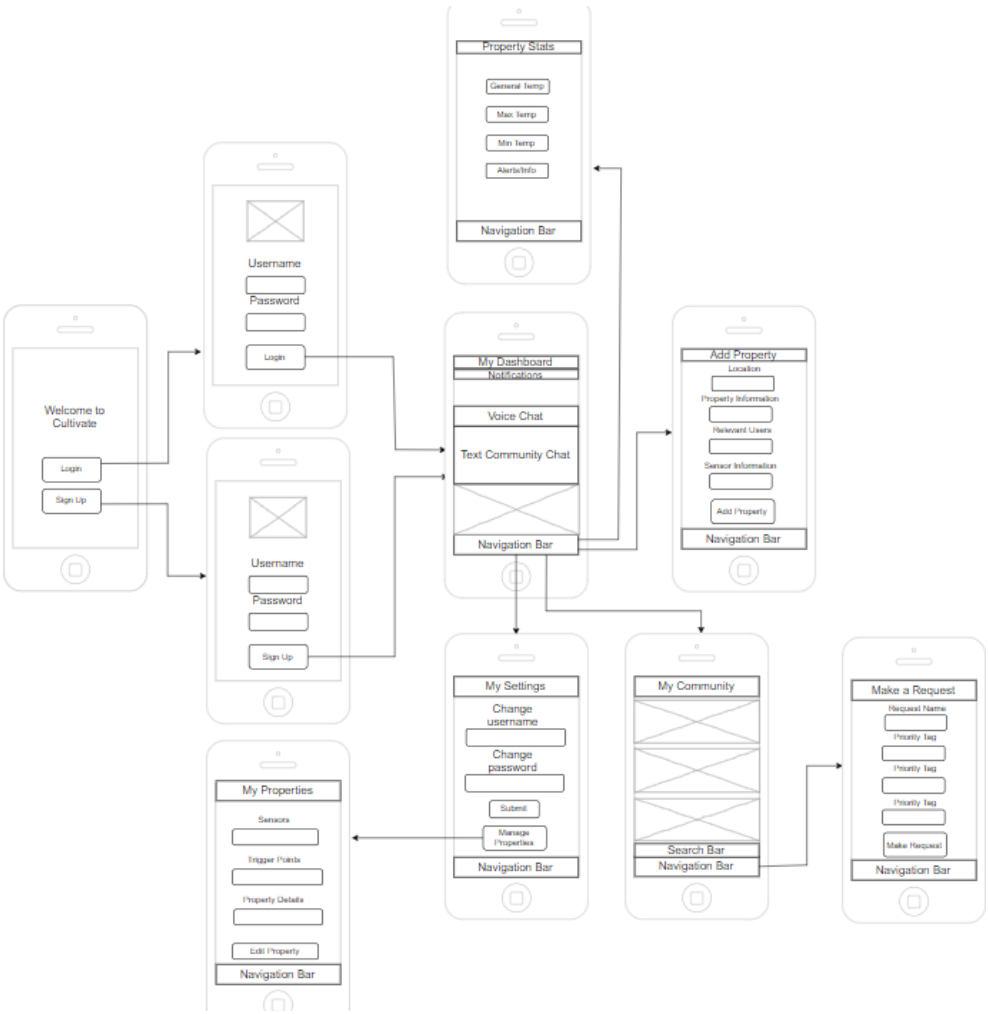
WIREFRAMES AND WIREFLOWS

A wireframe is a two-dimensional illustration of a page's interface that focuses on planning the functionalities available in an application and the structure of this content. They generally do not contain any colour or images, and are rapid to produce and refine a design. A wireflow combines many wireframes to demonstrate the movement between the wireframes, activated by elements such as buttons. The following subsection presents wireflows with each iteration of a section's wireframes. Each prototype presents consideration of all aspects of the system in a snapshot of the overall wireflow with each screen. This snapshot is accompanied by an overall explanation of the wireflow iteration and annotated with the section name for each significant component of the wireflow. Subsequently, each generation of wireflow is broken down with a detailed explanation of the scenario for the specific aspect of the system.

BRAD WATSON:

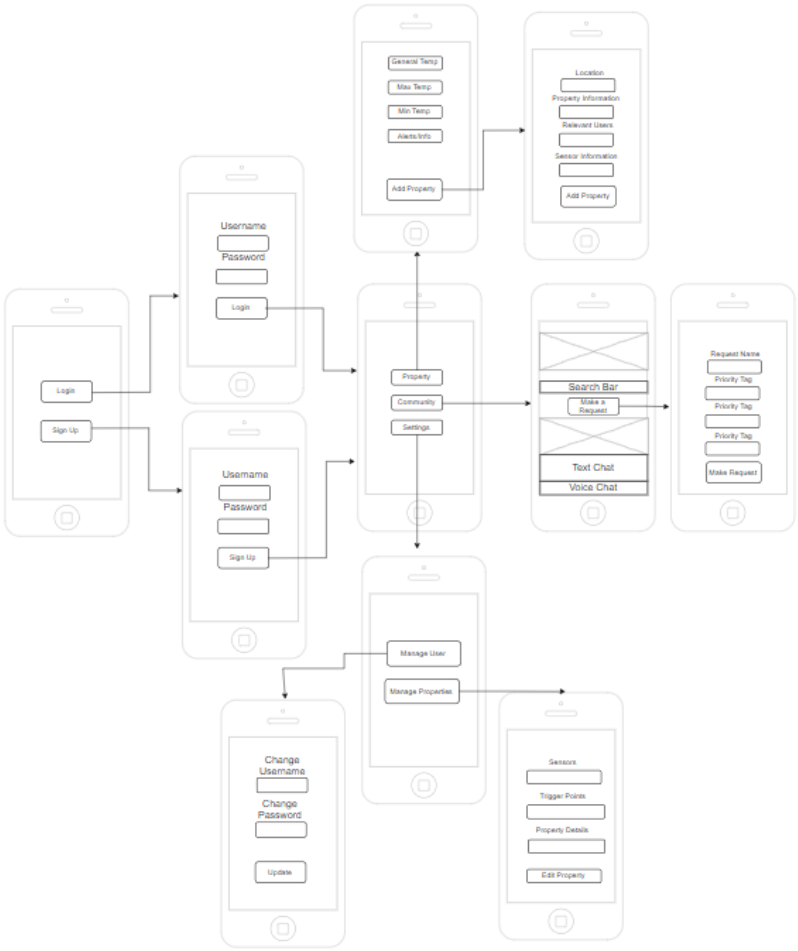
WIREFLOW 1:

This wireflow design was intended to be the first design to illustrate the key functions needed for our software outlined in our initial requirements. Different users may have different accounts which is why a login page and a sign-up page. Once a user has successfully logged into their profile or created one, they are then able to access the main dashboard. The dashboard is the most viewed screen in the app, which is why most of the core functions of the app (notifications and the chat) are available. A display at the top of the screen shows which function the user is accessing, and the user is able to navigate the app using the navigation bar at the bottom. For continuity, and a better user experience, the navigation bar is always placed at the bottom of each page, along with a title at the top to assist the user. From the dashboard, the user can access other main pages such as the community page, property page and settings page. The settings function holds the ability to change the username and password, along with current properties. The user is also able to navigate from the dashboard to either the property stats, or to add a property, each function having a simple box and form fill input for an easier user experience. The user can also navigate to the community page to see posts or make a request using the same form fill input. Basic functionality displayed via wireflow demonstrates an intuitive and simple user interface while still preserving the requirements of the app.



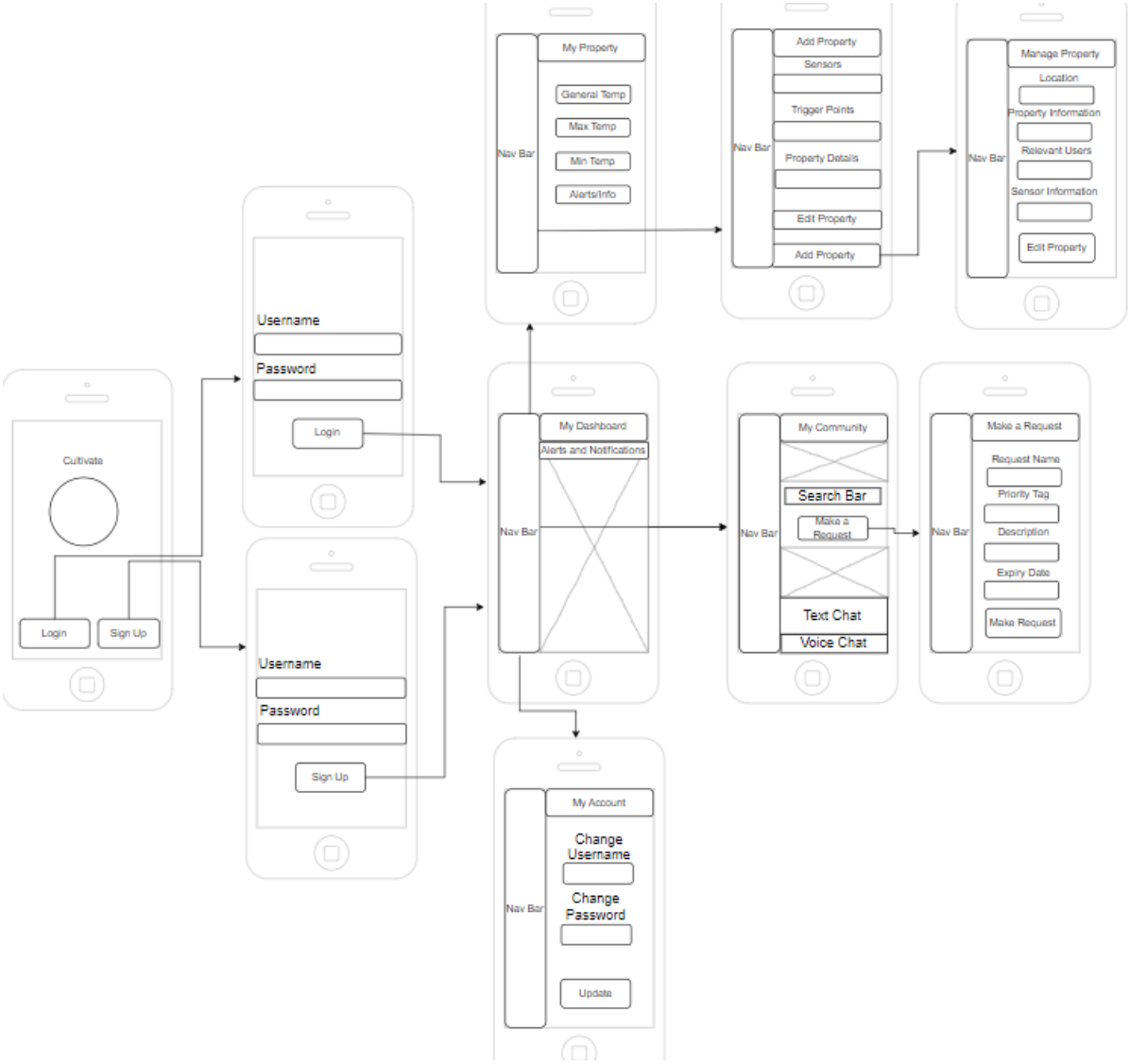
WIREFLOW 2:

The wireflow design was meant to be minimalistic, placing key core aspects of the functionality in the dashboard. The user can choose 3 buttons, either ‘property’ (add or view property), ‘community’ (view community posts or make a request) and finally ‘settings’ where the user can either edit existing properties or change account credentials. These minimalistic features offer an efficient, intuitive user experience, as it offers the basic requirements needed. Displaying only essential buttons and inputs eliminates risk of user error and better accessibility as there are no unimportant features. The user only needs to make a few choices to navigate the entire app, where although there are no titles to highlight which section the user has navigated to, the user is able to simply go back and navigate to the right page. Each form fill in lets the user input data, which is necessary to interact with the app, and is made as intuitive as possible by having the least number of forms the user is able to use while maintaining functionality. The wireflow demonstrates the basic outline of the app and how its design is supposed to minimise user error while enabling high navigation efficiency.



WIREFLOW 3:

This wireflow design provides the user with full functionality while having a sleek design. The navigation is found on the left, where the user is able to view pages such as the 'my community' page, the 'my property' page and the 'my account' page and access the functionality of the app through those pages. There is a header at the top of each page displaying the title so that the user knows which page they are accessing. This design takes up more room on the device and contains less white space however it provides the user with plenty of access to each form fill in and each button. On the dashboard, the most important and vital features of the app are displayed, such as alerts and notifications. In only a few clicks, the user is able to navigate the entire app, accessing all needed functions which is what the wireflow displays. The design is simplistic and intuitive, with features that highlight the functions of the app and incorporate a useful user interface while maintaining a professional look. The user is able to use the search bar in the community page to search for posts or users /communities, or they can interact with the text/ voice chat and make a request via a simple button. Any feature that requires user input for data uses a form fill in, utilizing continuity between pages in the app. The user is also able to use that design feature to edit/ maintain properties, edit their user credentials and more. Overall, the design is appropriate while presenting the user with the full functionality that's needed.



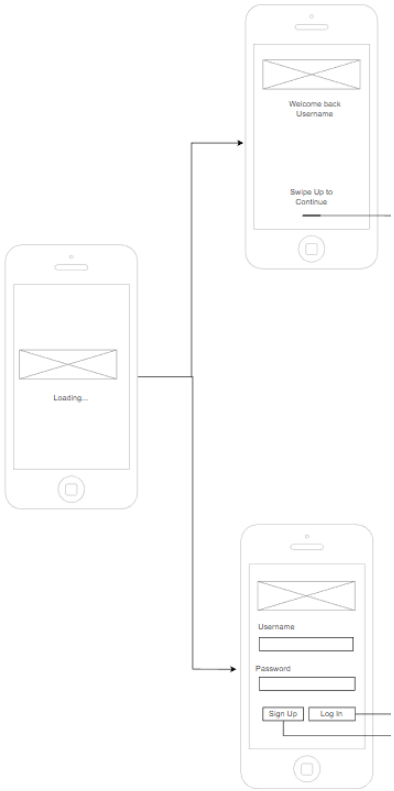
JADE HARRIS:

WIREFLOW 1: SIMPLISTIC INTERFACE

Preliminary wireflow that considers the functionality required to accomplish each system requirement by including one step to achieve a scenario on each screen.

In the first wireflow design, I created a separate screen that fulfilled each key step of the six initial system requirements. For instance, the screen to view a sensor's information, and a display to change the sensor limit was intentionally designed in two separate screens. In this way, it particularly explores Shneiderman's fourth rule, 'Design dialogue to yield closure', and the 'Sequencing' design pattern targeted at user perception and memory. This allowed me to break down the wireframe for each stage in the use case, ensuring that I considered all basic functionality and that the support for the functionality was included. Central design patterns include the prototyping of a 'Module Tabs' navigation system, the 'Form' input solution, and the 'Search box' workflow accelerator. Later workflow 2 then improves on this design by streamlining each requirement and condensing the isolated steps, economically utilising screen real estate. Changes include allowing all sensors to be viewable on a central map, and the information for a post condensed to the same screen as adding a comment. Then, workflow 3 incorporates more modern mobile application design patterns such as 'sliding overlay screens', to enhance the interface's flow for the user. However, the benefit of this preliminary wireflow is that its user interface is extremely simple, at times containing one button on each screen. This makes it suitable for mobile phone interfaces where often a user's fingers must interact with small interface elements.

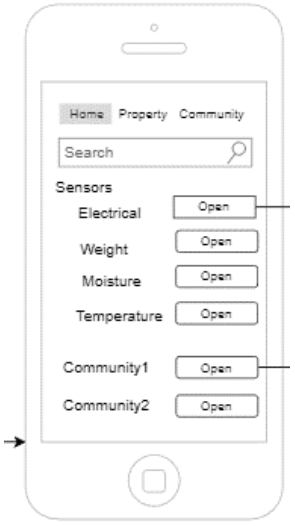
'Login and sign up' Wireframes



This wireflow begins by presenting users with a loading wireframe when they first load the application. This is a simple screen which will immediately establishes the branding of Cultivate. Additionally, it aligns with the first heuristic of Jakob Nielsen's 10 usability heuristics 'Visibility of system status', as often when they are first launched applications will be required to load data. Then, the loading screen branches off into two wireflow options: a 'Welcome Back' page if the user is returning, and a 'Sign up and Log In' page if you would like to log in with a new existing account, or sign up to Cultivate. The 'Welcome Back' page allows users to quickly access the dashboard interface. In retrospect, the incorporation of an 'Enter Password' wireframe even for an existing user would be essential to the security of our program. However, the simplicity of the wireframe presents all functions that are required. Similarly, the 'Sign Up and Log In' page presents all information that the application requires in a concise format. This screen was guided by the 'Account registration' and 'Login' design pattern. In future, the incorporated of the 'Password Strength Meter' design pattern would increase the security of accounts during the 'Sign Up' process.

'Dashboard' and 'Alerts' Wireframes

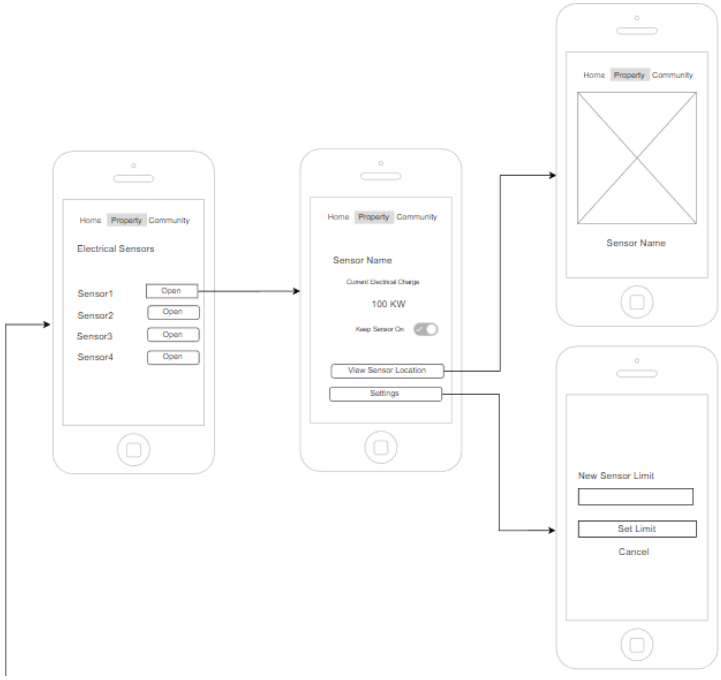
The 'Dashboard' wireflow that the user is presented with consists of one screen, drawing from the 'Dashboard' design pattern which allows the user to digest data from multiple sources at once. In this prototype, a central list of 'alerts' from 'My Property' and 'My Community' has not yet been considered. Instead, each of the sensors or categories that would be the source of alerts is presented quickly to the user. The shortcut to a particular type of sensor demonstrates the advantage of logical navigation that this simple prototype demonstrates. However, the lack of 'quick information' that this home screen presents, for instance a user being able to quickly view the most recent sensor alerts, highlights the primitive level of this wireframe and places for improvement regarding the 'Dashboard' design pattern. Also, the function of a user having the ability to join multiple communities was realised to be redundant, as this reduces the concept of centralising updates for a user's local area. Instead, user's will only be able to join one primary community, which also aligns with the scope of this project and streamlines the interface.



The menu in this wireflow employs a 'Module Tabs' design pattern. This is advantageous by separating content into each section which is accessed by the main interface, suiting the low to moderate technical experience of our primary target audience (property owners).

'My Property' Page Wireframes

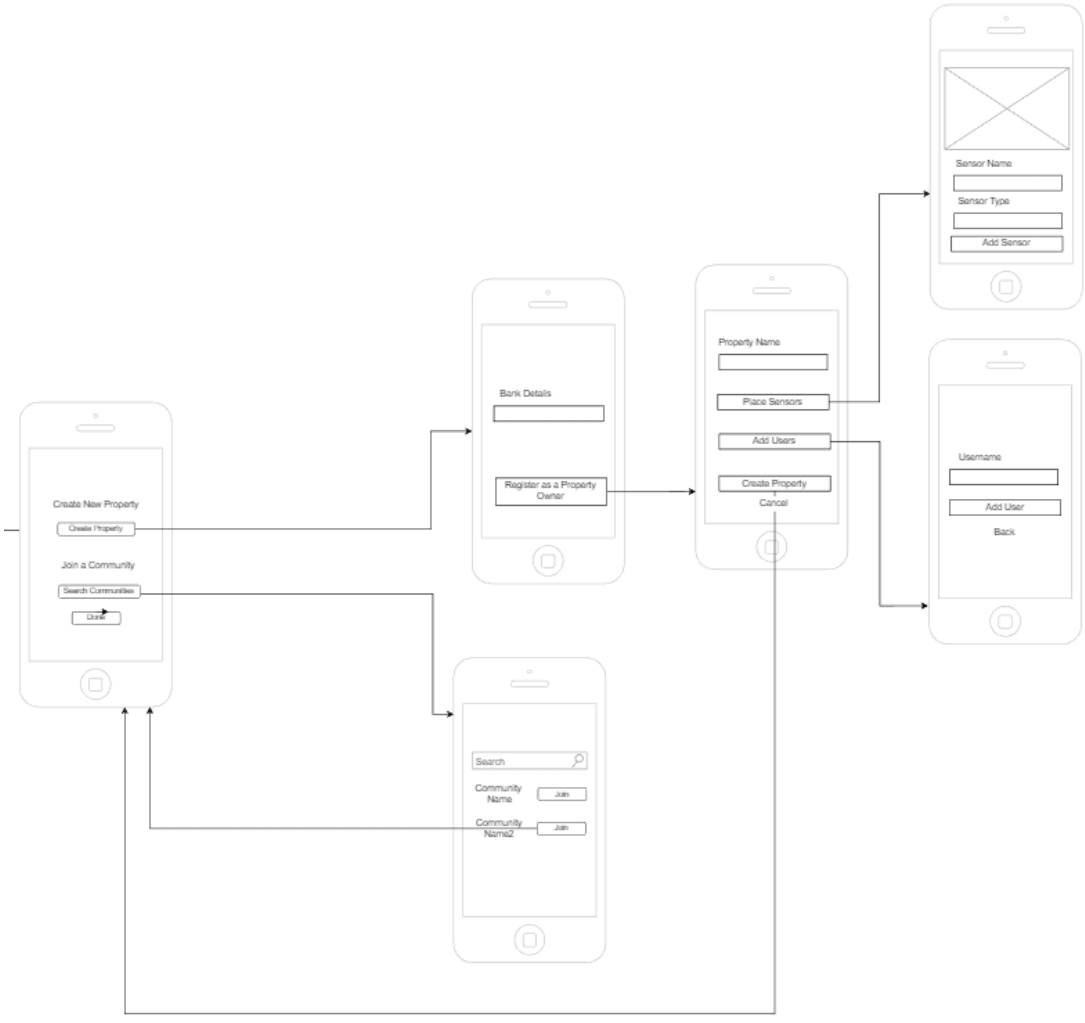
To access and customise the property, a sequence of wireflows are presented depending on the user's desired action. By providing an button that can access an interface with each step (such as the single action of accessing all of the electrical sensors, and the single action of viewing a sensor location), this wireflow is extremely concise. Whilst this does impair Nielsen's heuristic of 'Flexibility and efficinecy of use', as experienced users will need to follow this process of navigating to each separate page, it provides an extremely clear intention for new users. The first wireframe presents a simple design, presenting a list of sensors which a user can open to view greater information. Here, the wording of 'open' should be altered to 'view' or more clearly, 'view settings'. This would improve the wireflow in relation to two of Shneiderman's 8 Golden Rules of Interface Design, rule 7 'Support internal locust of control' as users would feel more prepared of what screen they are navigating to, and rule 3 'Offer informative feedback'. The screen presenting the information and options for each sensor is concise and presents all of the options for a user. Also, the use of the toggle switch design (Costa, 2022) suggests a mobile-phone friendly way for users to change particular settings. The map wireframe and updating a sensor limit display required functionality. A theme throughout this wireframe is the importance of creating concise interface messages, such as the confusing in the 'Settings' button directing the user to a 'New sensor limit' screen.



'Onboarding (Including 'Create a new property') Wireframes

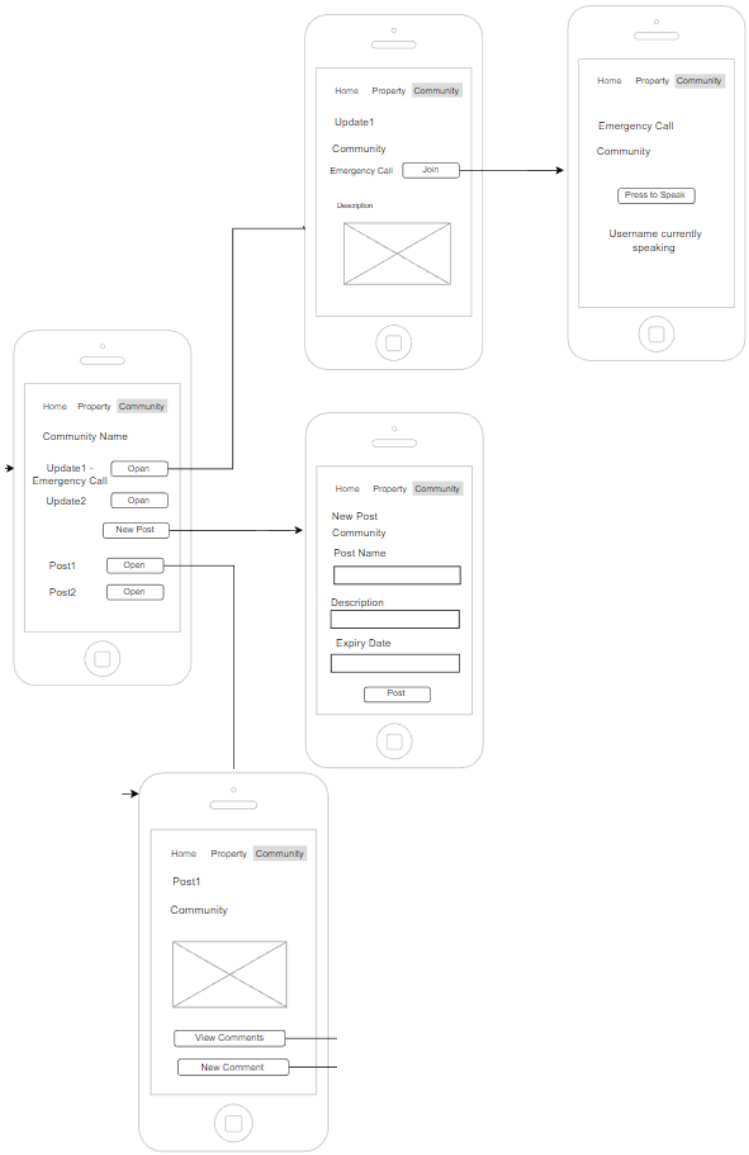
The sequence of steps to create a new property are considered in the below wireflow. This is accessed after a user has selected the 'Sign Up' button on the 'Sign up and log in' wireframe. These screens align with a 'Setup Wizard' design concept, whereby a single goal of creating a website has been reduced to each interface, making the registration process less intimidating for users. However, it is presented in a nonsequential order, where users return to a screen that shows which options in the sign up process they have completed. As shown, the wireflow of this registration process employs the use of buttons to present users with these options. The aesthetic intention of this is that the buttons will change into a different colour once the user has completed the step.

Users are presented with the option to create a new property or join a community. This allows new members to select both register a property with the system, become a community member, or to undergo both processes. After a user has completed a a path (for instance creating a new property), they will be returned to the initial wireframe, as indicated by the flow arrows. If a user decides to create a property, firstly they will be asked to enter their bank details which will be charged when the IoT installers visit their property. The next wireframe allows the user to provide their property address and add other members to their property. Due to a misunderstanding in the scope of the system, users will not be able to place sensors on their property, instead this screen is related to the installers. However, this version of the 'Steps left' design pattern provides a user with a great overview of their progress. Alternatively, the 'join community' option allows users to search a particular address or community name through the use of the 'Search box' design pattern. Additionally, the 'join community' button is inspired by the 'Follow' design pattern, which allows users to subscribe information that they find interesting. This provides a simple interface for users to then join the communities that they wish.



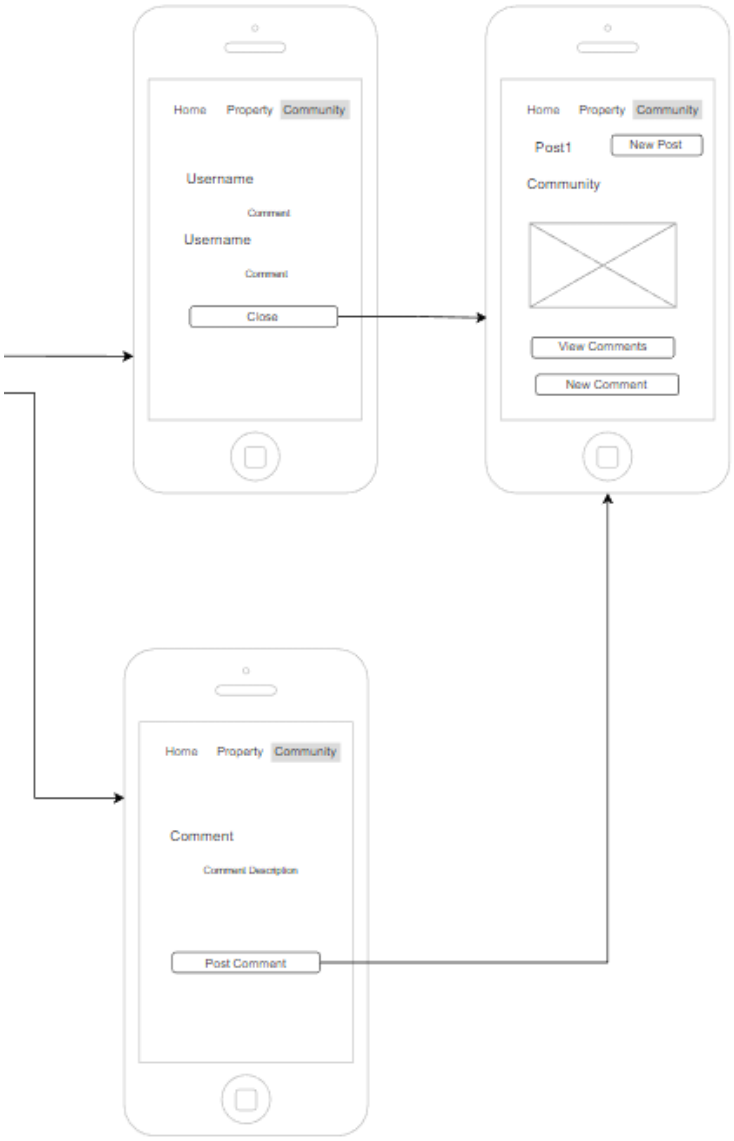
'My Community' Page Wireframes

The wireframes that support a communities 'notice board' is modelled below. The first interface portrays an intended community landing page, consistent with the dashboard which displays central shortcuts for user's to navigate to for further information. Mostly though, the main community page wireframe distinguishes between two community post types, an 'Update' and a 'Post'. This groups the interface into posts that are created by administrators of the community, 'Updates' and posts that the user can create. Viewing an update is presented in a screen that accepts the 'Blog Page' design pattern screen. It presents the update name, description and a potential image. Updates can take different formats, as seen through the selected update which contains a join the emergency call channel. This executes the walkie-talkie push to talk voice channel system. By positioning the emergency call as integrated into potential community updates, it ensures that users will check if the channel is open often. As well, the 'New Post' interface draws from the 'Form' design pattern to capture the input of a new community post. The presentation of necessary elements in the form provides a logical way for users to enter their desired information. Lastly, viewing a post presents the information in a simple interface.

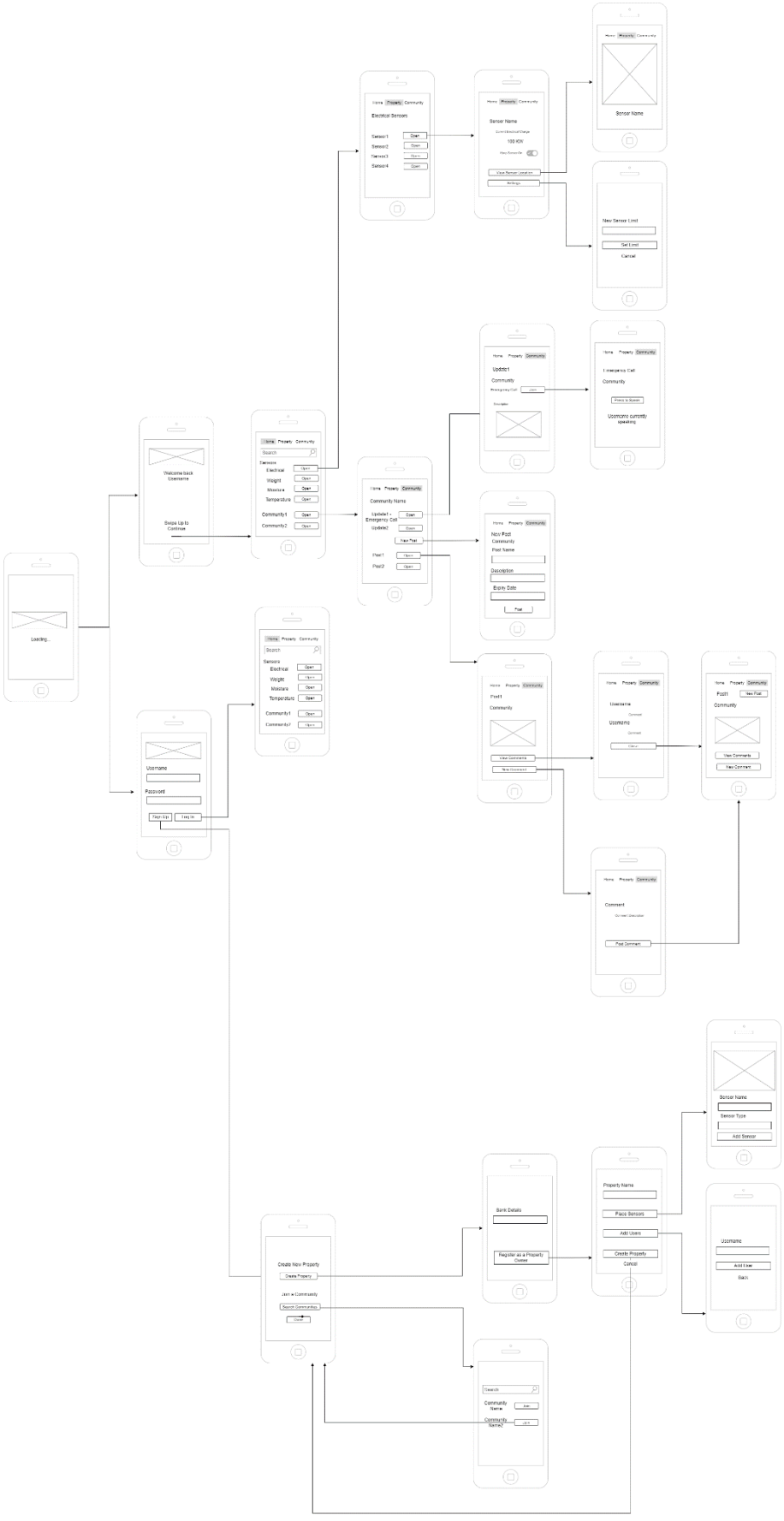


‘Add a comment’ Wireframes

Lastly, the wireflow presents a sequence of wireframes that allow a community member to comment on the post of another. This executes the ‘Comment Box’ design pattern in a simplified version, allowing a user to post a new comment. Additionally, a user can prompt a modal dialogue box (overlay). By separating all of the comments on an individual interface, users can quickly offer help to each other. Although separating the ‘New Comment’ screen into a separate interface is not optimal for the user to accomplish a task, as it may be most efficient for a user to refer to existing comments, it does simplify the interface and accomplish the functionality of adding a comment in a considered way.



Overview of Wireflow 1:



WIREFLOW 2: STREAMLINED INTERFACE

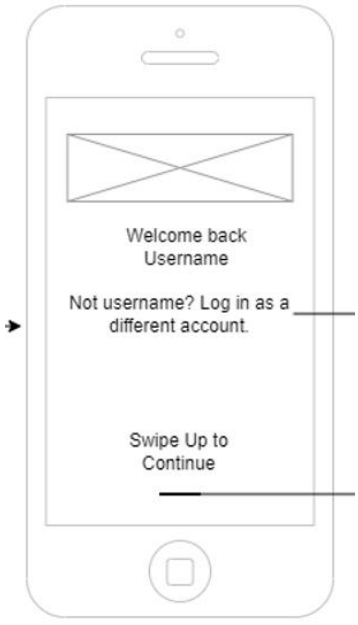
Combined features in each screen, and increased customisability for interfaces, such as the consideration of settings.

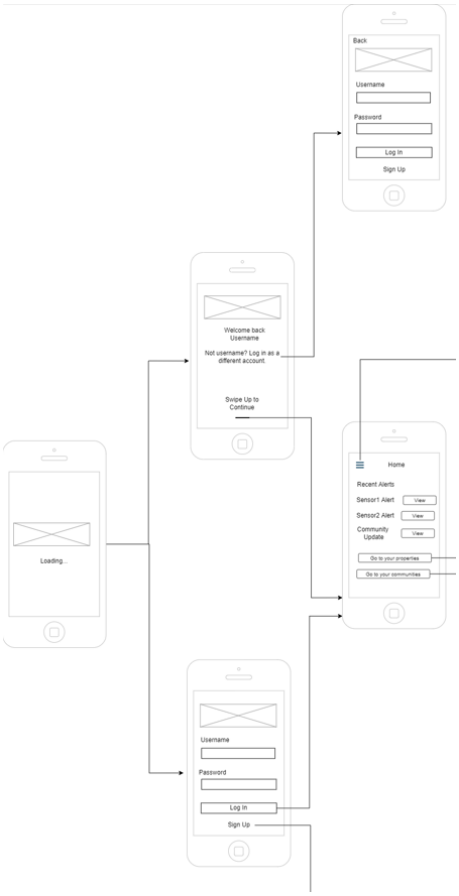
Workflow 2 varies from wireflow 1 by combining related steps to accomplish a core function into a single interface. In this way, the wireflow is approached from the perspective of the user, and improving efficiency for their work processes. It includes improvements such as combining the 'Blog post' design pattern for posts on the Community page into the same wireframe as the 'Comment' design pattern. This not only reduces the time required to navigate between features of the application, but supports the positioning of related information together. For instance, a user will be able to view comments whilst being able to read the post's description. As such, this wireflow accomplishes an alternative design that balances the 'Sequencing' and 'Chunking' design patterns to achieve a sleek interface. This wireflow also experiments with a different 'Retractable menu' design pattern, allowing further detail to be provided in terms of navigation around the interface without cluttering the wireframe displays. Further, the change from a nonsequential onboarding process, to the 'Steps left' design pattern provides an alternative to a simple user registration process.

Additionally, this wireflow has emphasis on the 'Notification' design pattern, through the incorporation of a 'general alerts' element that is displayed on the dashboard. Alongside 'Continuous scrolling' to allow the user to iterate through each alert in the menu, the use of the 'swiping' input motion will maintain a sleek interface. Other significant variation involves greater detail into the full functionality that is required to support the core system requirements, such as designs for an 'Edit sensor information' wireframe.

'Login and sign up' Wireframes

This wireflow is consistent with the loading, login and sign up wireframes from wireflow 1. This is due to the lack of need to change the concise process. However, this design alters the 'Welcome Back' wireframe to contain an option for the user to log into a new account (below).

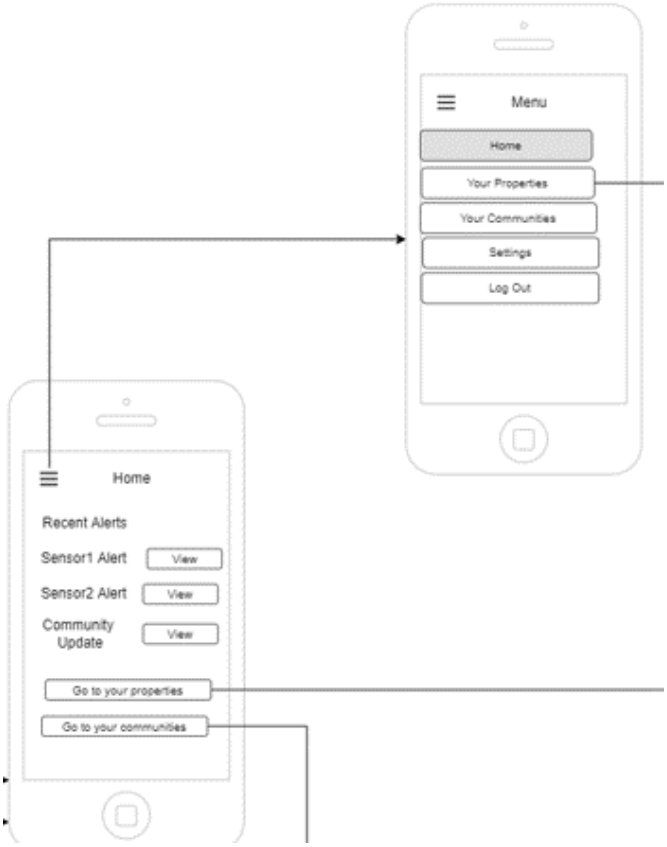




‘Dashboard’ and ‘Alerts’ Wireframes

This ‘Dashboard’ wireframe includes a single screen, drawing from the ‘Dashboard’ design pattern in a more explicit way than the home screen in wireflow 1. This ability to present the user with multiple sources of data is achieved through the inclusion of the ‘Notifications’ design pattern, whereby alerts triggered by sensors, new community posts, or new updates will appear beneath the ‘Recent Alerts’ section on the dashboard. This improves Shneiderman’s eighth golden rule of interface design, ‘Reduce short term memory load’, which is particularly essential given the critical nature of Cultivate in a natural disaster. For instance, a user will be reminded to check the status of the sensors often. This also appeals to the ‘Trigger’ timing design pattern, as cues are placed through the users regular path to motivate them to take action.

Additionally, experimenting with the ‘Retractable menu’ design pattern offers greater screen space to the design when the user does not have the menu open. This also provides greater options for number of menu options, and for some users, may make navigation easier given the increased menu icon size. On the other hand though, this menu style means that navigating between elements requries at least two interactions with the wireflow (one for expanding the menu, another for selecting the option).

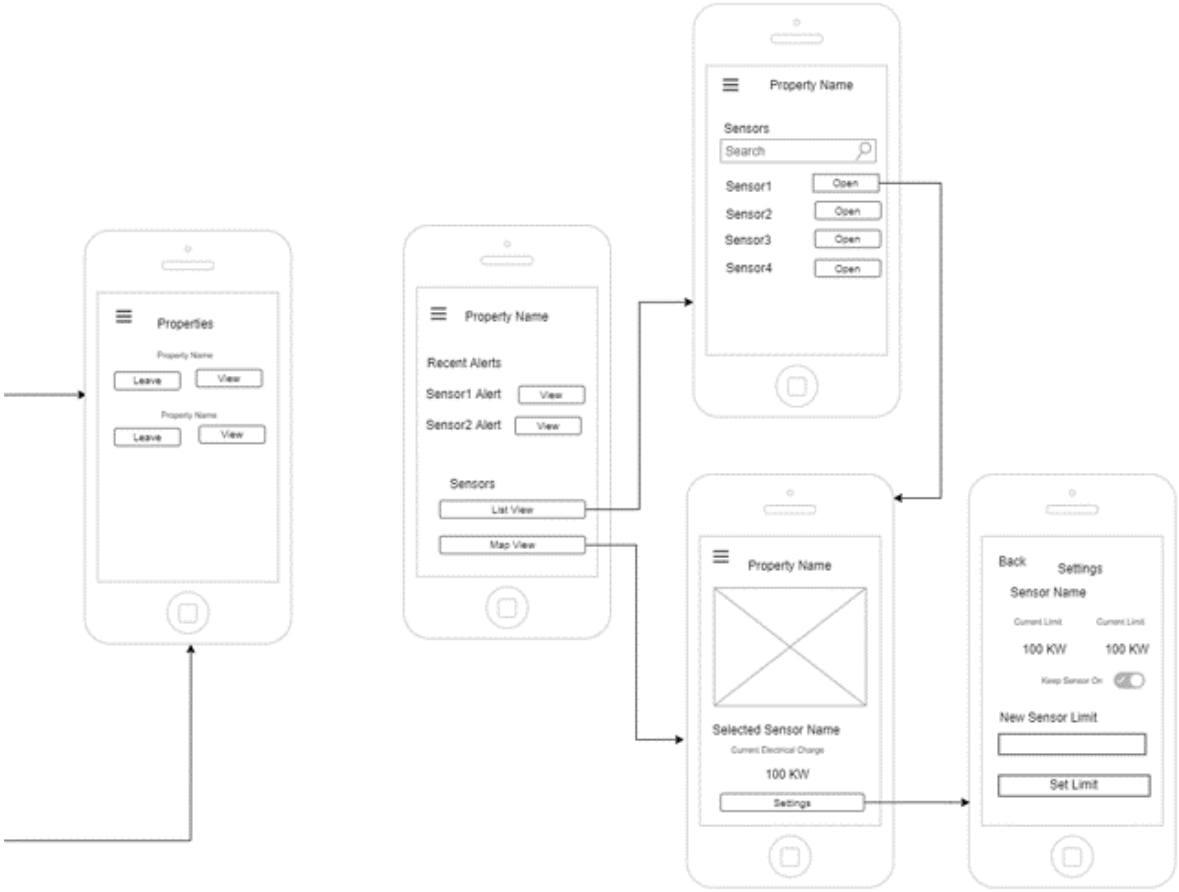


‘My Property’ Page Wireframes

The ‘My Property’ section combines multiple features into the same screen, instead employing the ‘Chunking’ design pattern to group useful functions together for the user. This design also experiments with the option of users as a member of multiple properties. This resulted in the addition of the ‘properties’ wireframe, where the user can view the information page for one specific property. As mentioned, overall, wireflow 2 leverages the ’Notification’ design pattern which has been used on the specific property’s main page. Unlike the general alerts that will appear on the dashboard interface, this allows the user to quickly view notifications relevant to a specific property and sensor. This improves Nielsen’s seventh heuristic ‘Flexibility and efficient use’, whereby users can achieve greater precision in which updates they would like to keep updated about.

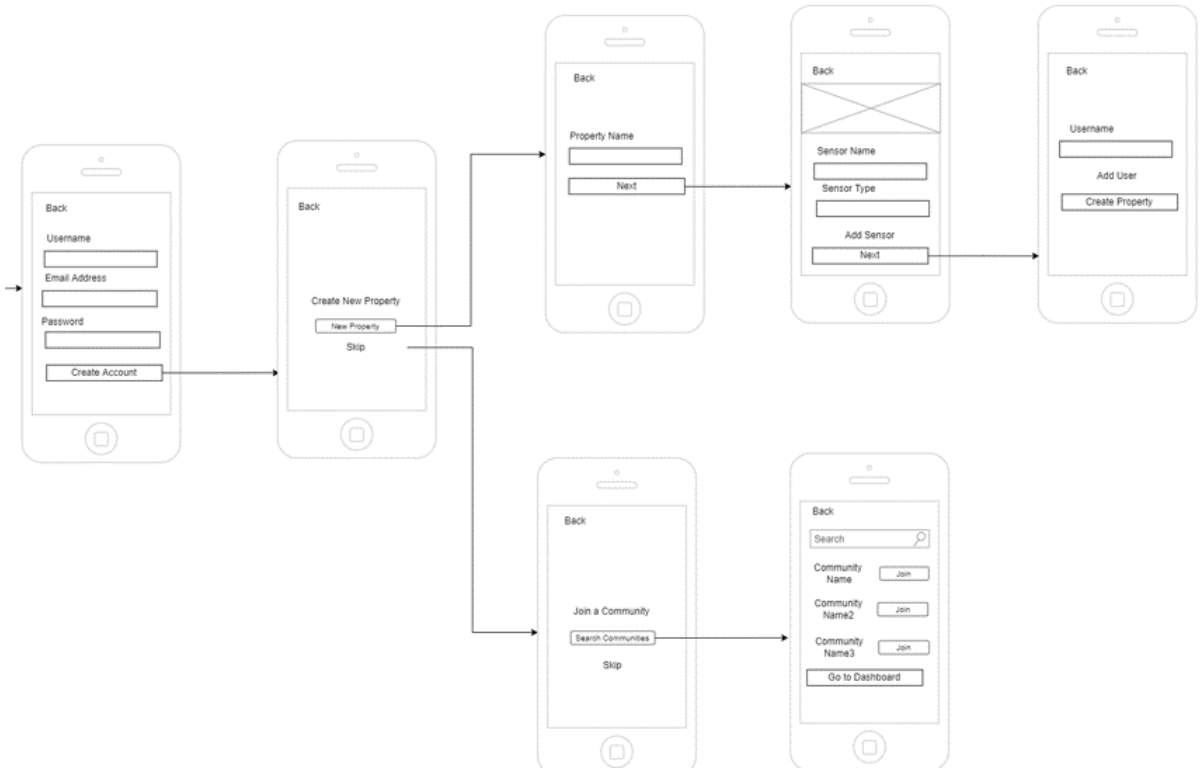
This wireflow also incorporates the ’Search bar’ design pattern in the ‘Sensors' wireframe, offering convenience in locating a specific sensor. When a sensor is selected through the ‘open’ button, its information page will be displayed. Unlike wireflow 1, on this screen the location of the sensor on the map is integrated with its details, reducing the users ‘short term memory load’, as defined by Shneiderman’s eighth golden rule.

This wireflow also includes a wireframe for the settings screen, illustrating the greater depth and customisability that the application supports. Whereas in wireflow 1, the IoT sensors would function with a predetermined limit, in wireflow 2 the user can set their own limit.



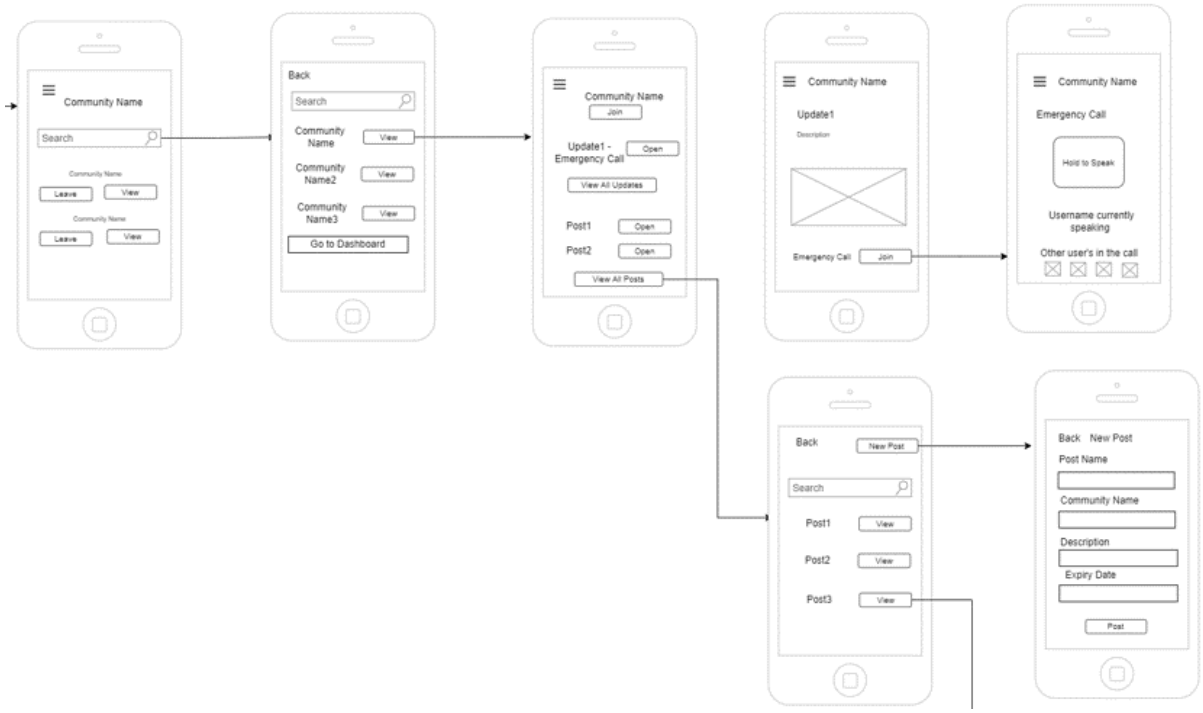
'Onboarding ('Create a new property') Wireframes

The onboarding wireframes for wireflow 2 employ the 'Sequencing' design pattern, to create an onboarding process more inviting than wireflow 1. Although both could work effectively, the idea of breaking one step onto each screen while the user is entering details will allow then to focus on each step. It also makes the process seem more tedious as rather than returning to a previous screen, the user is rewarded with immediately moving on to the next step. Coupled with the 'Steps left' or 'Completeness meter' design pattern, this will encourage users to finish the onboarding process as they receive gratification from their progress.



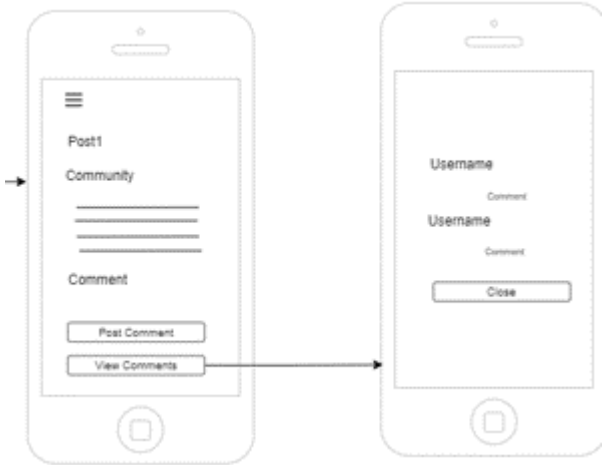
'My Community' Page Wireframes

The 'My Community' wireflow diverts from wireflow 1 through the combination of multiple actions on one page. In the leftmost wireframe, new functionality has been tested which will allow a user to be a member of many communities. It also includes the 'Search box' design pattern to support this feature. These adjustments create greater options for the user, allowing them to access a wide range of information from different communities quickly. However, it may impede important notifications from their own community. Once a user selects a community, the wireframe for a community page has a greater number of actions than wireflow 1, where a user can view one update or view an entire list of pdates which are displayed on a separate screen. The 'View All Updates' button is a replica of the 'View All Posts' button. By incorporating the 'Continuous scrolling' and 'Search Box' design pattern in a separate screen, users will be able to navigate between updates and posts effectively.

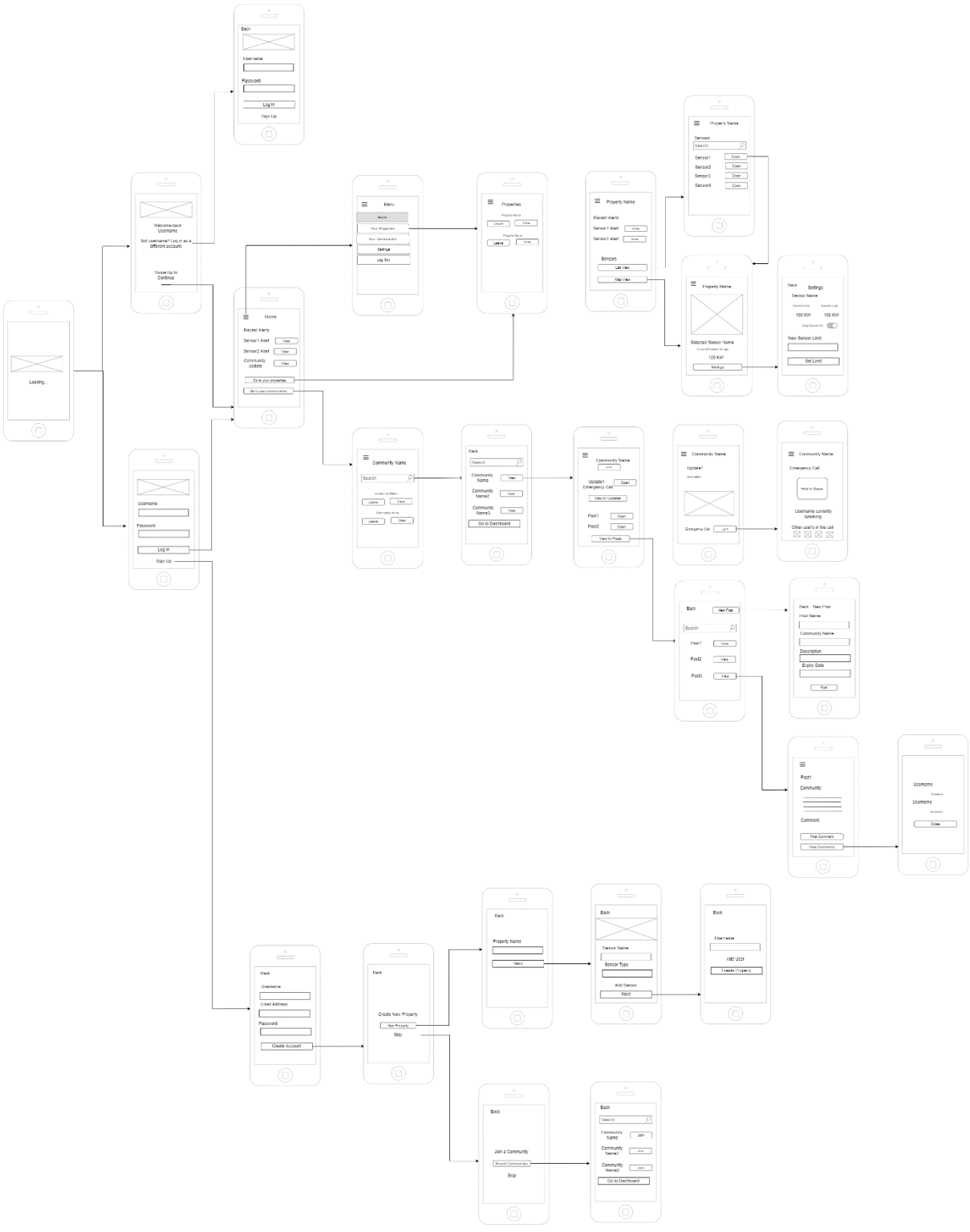


'Add a comment' Wireframes

Finally, this section of the wireflow offers a simple two-wireframe interface to the user. Namely, the main screen for each post offers a space for the user to immediately type and post a comment. This enables property owners to quickly type a comment of assistance, while viewing the original information that the poster included. Due to the increased control that users gain by seeing the 'Post comment' feature when they read a post, this wireflow appeals to the 'Autonomy' design pattern. By offering a sense of control to the community member, it enhances their tendency to respond.



Overview of Wireflow 2:



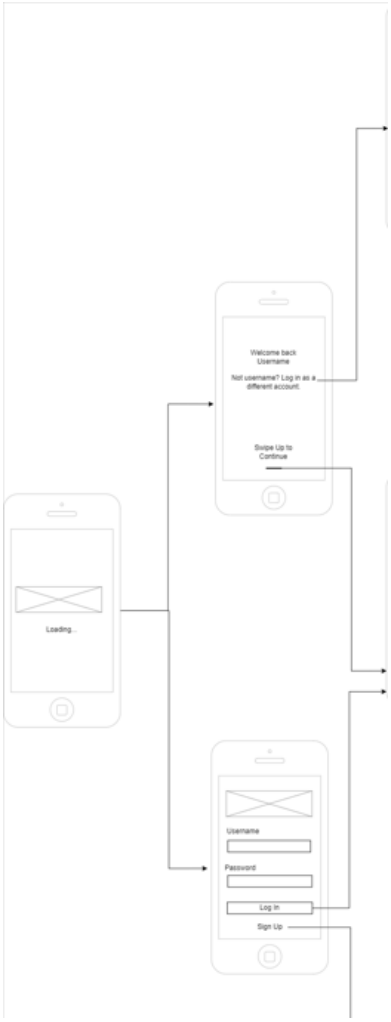
WIREFLOW 3: MODERN INTERFACE –

In-depth interface details supported by modern and dynamic visual features, such as 'Modal' draggable overlays

Lastly, the third wireflow presents a design that enhances the movement between core functionality, and considers the detailed features that must support them. Flow is enhanced between screens through the inclusion of modern and dynamic visual features, such as incorporating different user input methods including 'swipe', and updating the interface based on the user's context. This enables wireflow 3 to explore the use of overlays to create changing and connected interfaces, such as the display of sensor information as a layer atop the property map. These overlays are an adaptation of the 'Modal' design pattern. This persuasive concept draws the users attention in coherent way that varies from the 'Sequencing' cognitive pattern. Changes also include the shift to the 'Repeated Menu' design pattern at the footer of the interface, varying from the 'Module Tabs' interface of wireflow 1 in that the entire interface can change as opposed to a central panel on the screen.

'Login and sign up' Wireframes

This section of the wireflow is consistent with the design patterns explained in wireflow 2 'Login and sign up' wireframes, as no major interface changes were required to alter its fundamental functionality.



‘Dashboard’ and ‘Alerts’ Wireframes

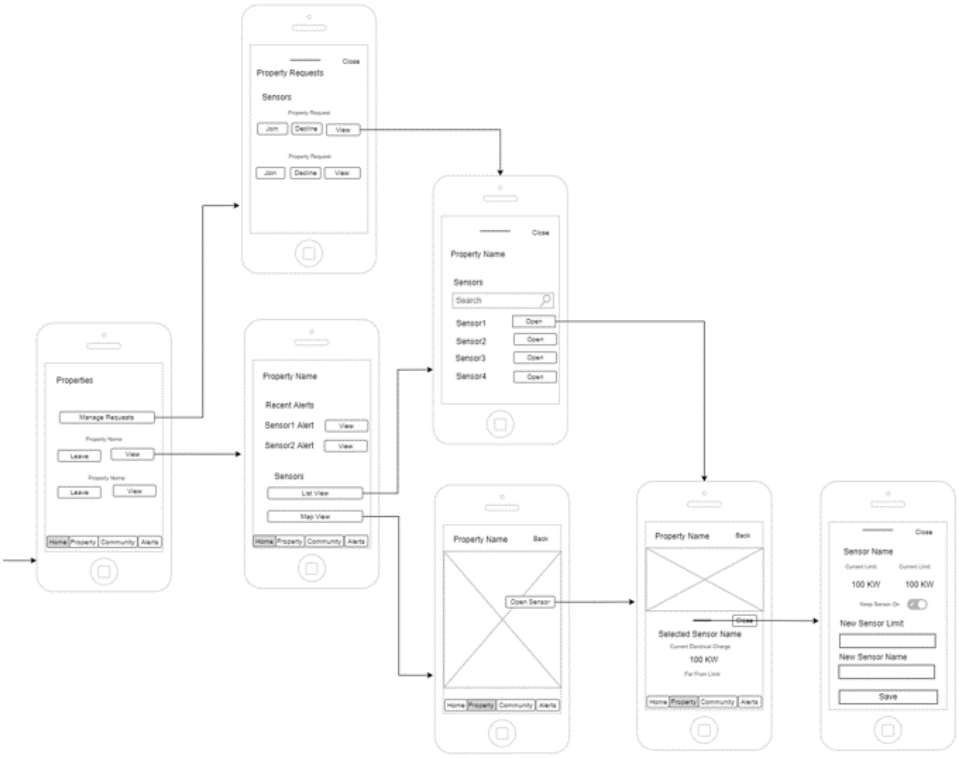
Significant changes in wireflow 3 emerge in the ‘Dashboard’ and ‘Alerts’ wireframes. Unlike wireflow 2, a ‘Carousel’ data pattern for presenting the notifications has been employed to offer greater space for alert previews presented to the user. The ‘Carousel’ pattern is especially used when the user should browse through a set of items. This aligns with the exploration of different mobile application input types, such as ‘swipe’, where the user will be able to swipe through the carousel. Versus the list view of wireflow 2, swiping horizontally can offer users greater precision in stopping on a particular item that they would like to select. Additonally, the acceptance of the ‘Progressive disclosure’ design pattern allows users to view all of the recent alerts if they wish to seek more details, as well as greater details of a specific alert (through the use of the ‘view’ button). This reflects the problem summary of the design pattern, whereby the user can focus on the task at hand while searching deeper into details if it is convenient. Also, the design of a customisable ‘Shortcuts’ section on the user dashboard aligns with Nielsen’s seventh usability heuristic, ‘Flexibility and efficinecy of use’. This will allow property owners to view specific information that is relevant to them quickly.

Additionally, the acceptance of the ‘Repeated menu’ design pattern divers from wireflow 1, where the repeated menu was positioned at the top of the screen. As a footer, the menu is less disruptive to the interface. In relation to wireflow 2, this design allows users to navigate more easily.

‘My Property’ Page Wireframes

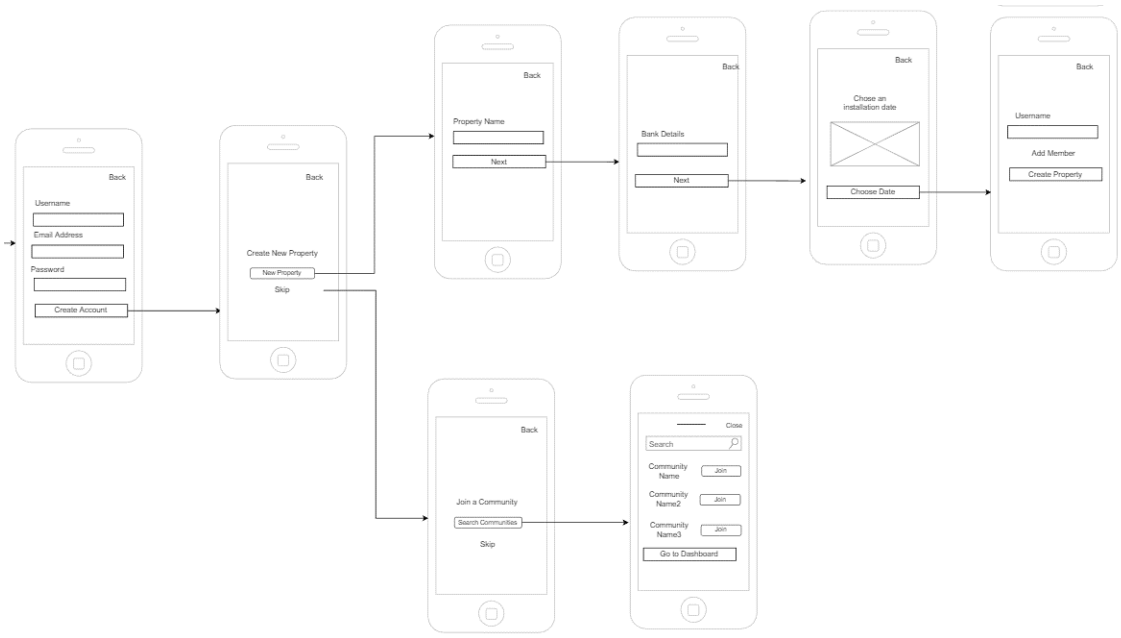
The ‘My Property’ page wireframes focuses on providing an interface that supports the full functionality and assoicated customisation for the users to control the sensors on their property. Wireflow 3 diverts from wireflow 1 and 2 through the extension of the ‘Property requests’ interface that it offers the user, and the expansion of the ‘Map’ design pattern which prompts a ‘Modal’ inspired overlay when a user selects a particualr sensor. This is evident when the user selects the ‘Map view’ button, leading them to an wireframe consisting of a full-sized map. As a full sized screen, wireflow 3 can incorporate a more detailed design pattern for this map with the ‘Map navigator’ design pattern. This allows users to find a location of choice on the map and therefore the associated sensor. As a modern adaptation, they can employ ‘swiping’ to navigate around the map, and ‘pinching’ to zoom into the image.

The ‘Modal’ influenced overlay containing the information for the selected sensor emphasises the dynamic design elements which are explored in the wireflow. The flow between user actions is demonstrated as users can drag up the overlay ‘movement bar’ to offer an expanded settings interface. This map and sensor system reflects Nielsen’s second usability heuristic, ‘Match between system and the real world’, allowing users to digitally navigate their property and monitor the sensors. Additionally, the eighth heuristic ‘Aesthetic and minimalist design’ is achieved through the improved transition.



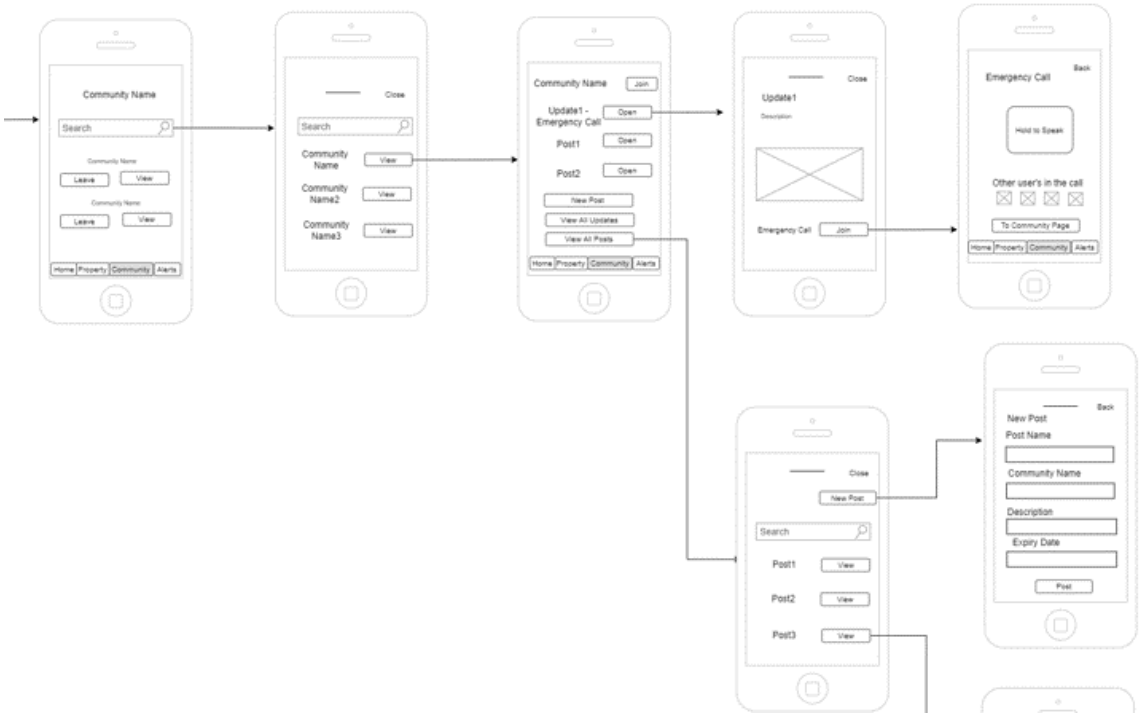
'Onboarding ('Create a new property') Wireframes

The wireflow for the onboarding process maintaims a sequential style to invite the user into the system, drawing from the 'Steps left' design pattern to seamlessly integrate all the steps required to set up an account. Namely, this includes the inclusion of a screen to retrieve the user's bank details, and a wireframe for the user to select a date for the IoT sensor installer to travel to their property. The calendar element draws from the 'Calendar Picker' design pattern, allowing users to submit information in simple visual way. Additionally, this onboarding wireframe will ensure that only valid dates are selected, and that dates where installers have already been booked can be represented easily such as though a red date colour, or a grayed out selection.



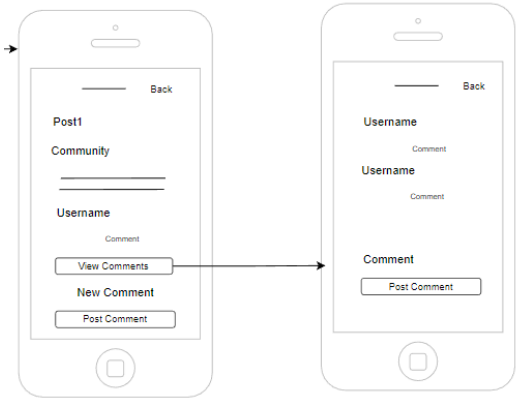
‘My Community’ Page Wireframes

The ‘My Community’ page is consistent with the design patterns in wireflow 2’s ‘My Community’ page.

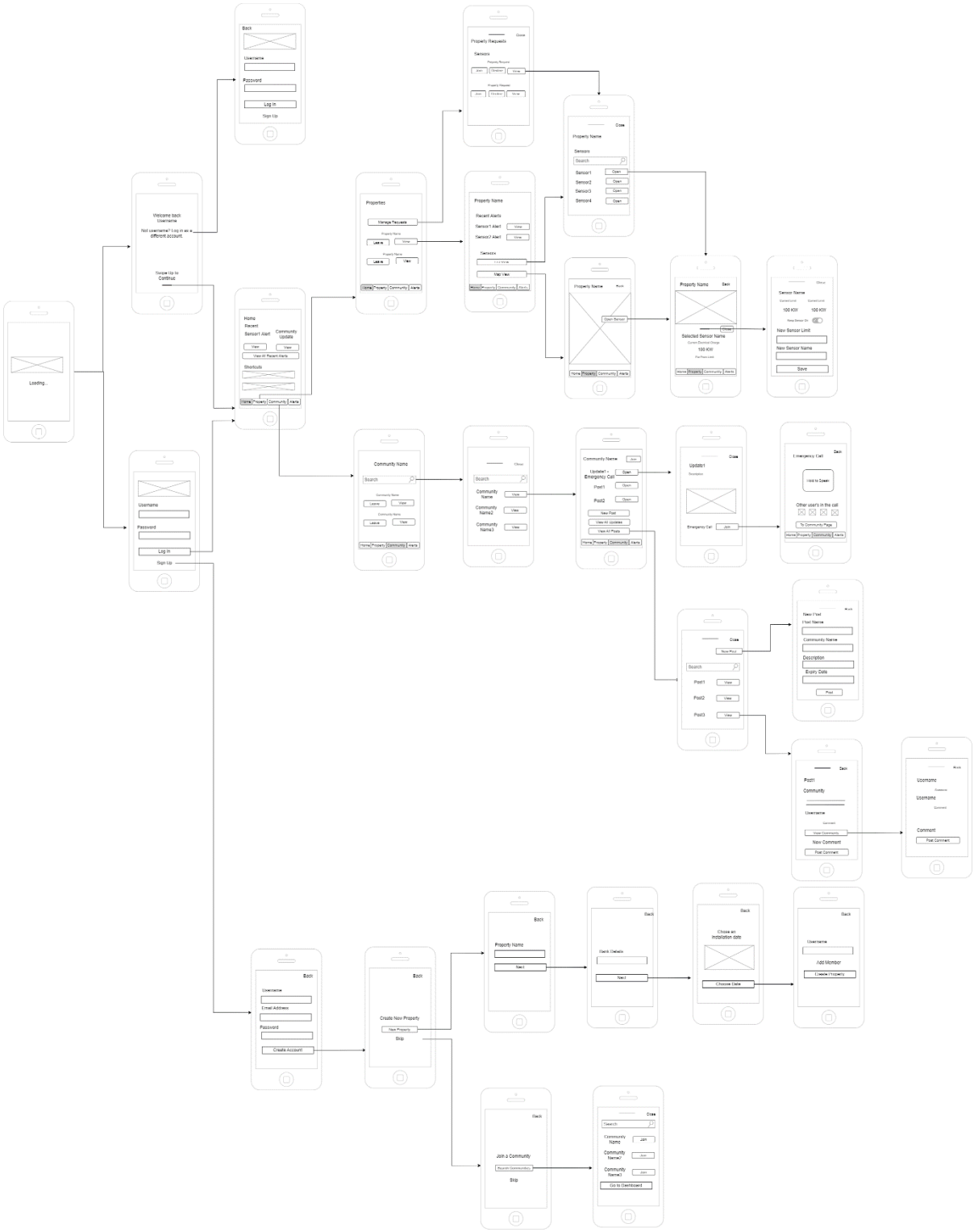


‘Add a comment’ Wireframes

Finally, the ‘Add a comment’ wireframe is consistent with the design patterns in wireflow 2’s ‘Add a comment’ wireframes, as no major interface changes were required. This includes the acceptance of the ‘Comment box’ element and the ‘Blog post’ set up. A further inclusion in this wireflow is the application of ‘Progressive Disclosure’ however, where users are able to preview a select number of comments on the post. This can assist them in the information that they require as they compose their own comment.



Overview of Wireflow 3:



LOW FIDELITY PROTOTYPE

BRAD WATSON:

Influenced by the principle of simplicity in each of the three presented wireflows, I created a low fidelity prototype in Figma by created combining the optimal elements from each core function.

'Login and sign up' Low Fidelity Section

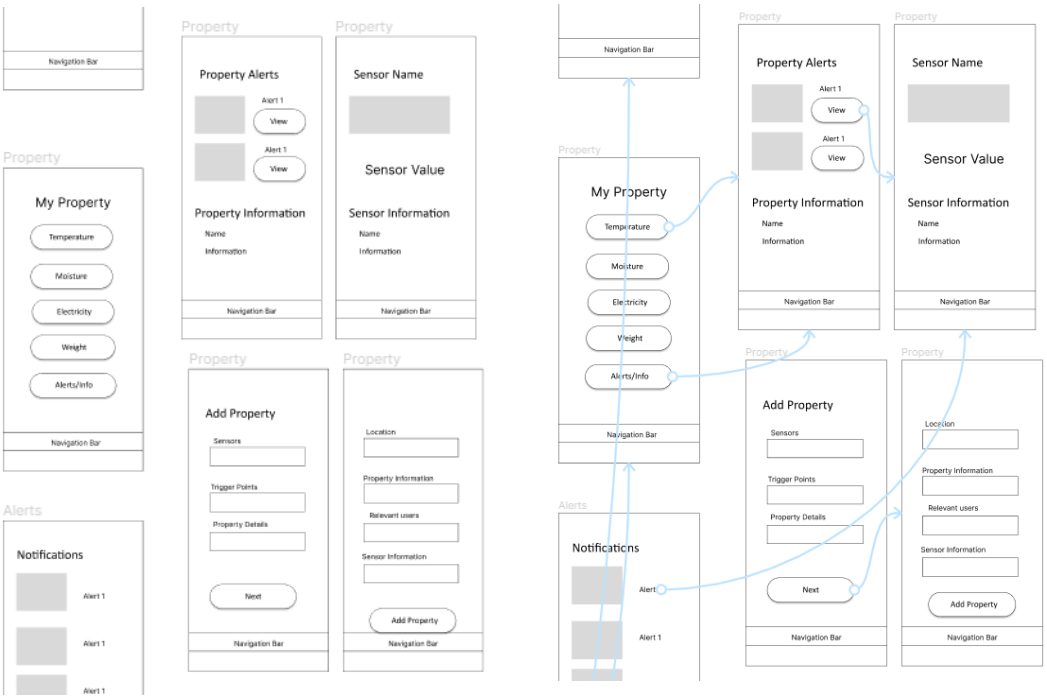
The 'Login and sign up' Low Fidelity section draws from Wireflow 1.

'Dashboard' and 'Alerts' Low Fidelity Section

The best dashboard design was selected from wireflow 2, in conjunction with the notification system of wireflow 1.

‘My Property’ Page Low Fidelity Section

Wireflow 3 was selected as the optimal design for the ‘My Property’ page section, given its aesthetic appearance.

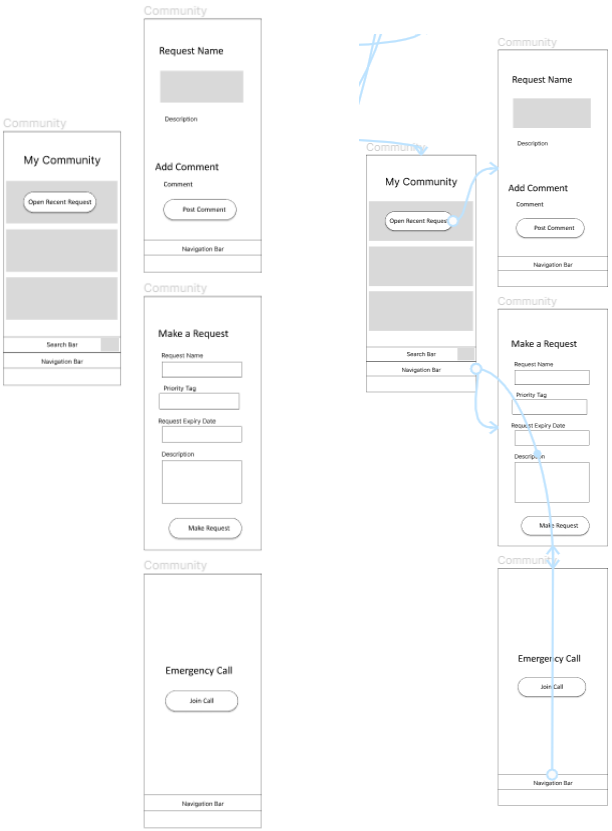


‘Onboarding (‘Create a new property’) Low Fidelity Section

Wireflow 2 was selected as the optimal design for the ‘Onboarding’ process section. When creating the low fidelity prototype, the idea to direct a new user to the dashboard, which will then prompt them to ‘Add Property’ or ‘Join community’ (replacing the navigation to the ‘My Property’ and ‘My Community’ until the user has finished the sin up process).

‘My Community’ Page Low Fidelity Section

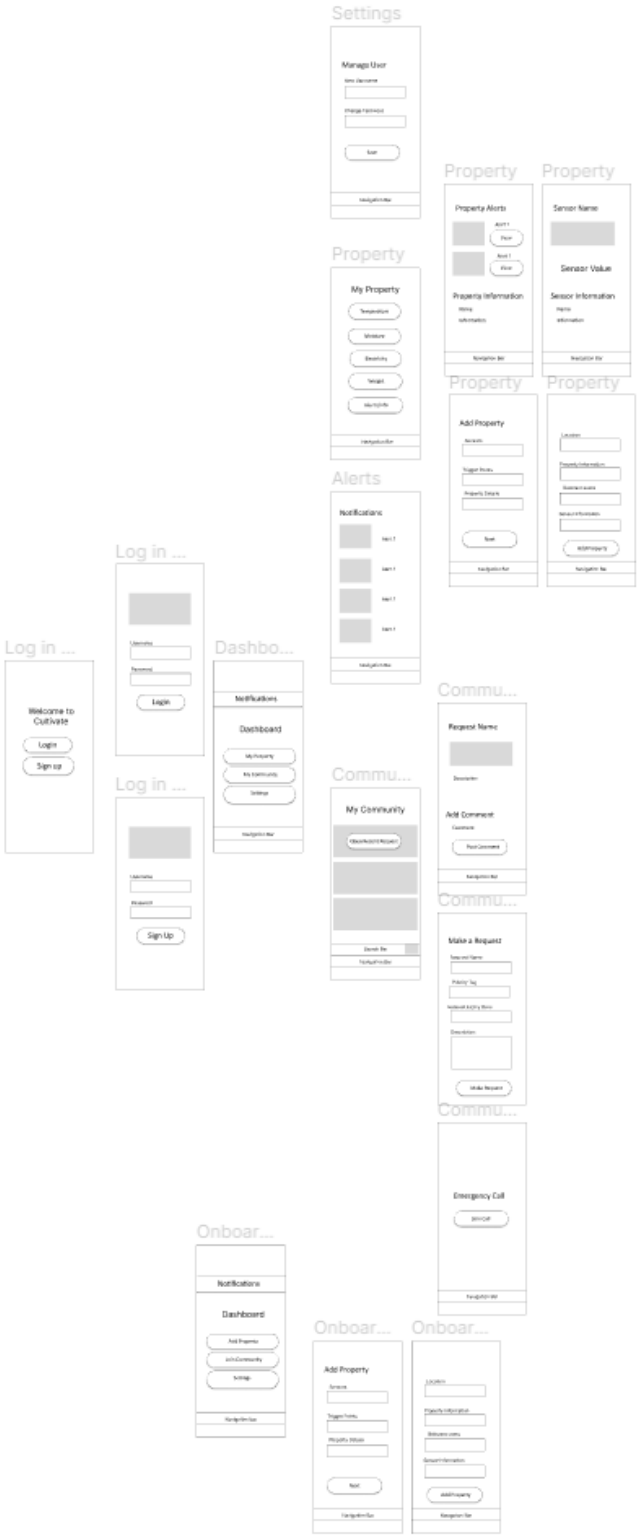
The optimal ‘My Community’ Page design was selected from wireflow 1. Additionally, during the design of the low fidelity prototype, the inclusion of the ‘Emergency Call’ function was incorporated.



‘Add a comment’ Low Fidelity Section

During the design of the low fidelity prototype, the inclusion of the ‘Add Comment’ function was incorporated, influenced by the post structure of wireflow 1.

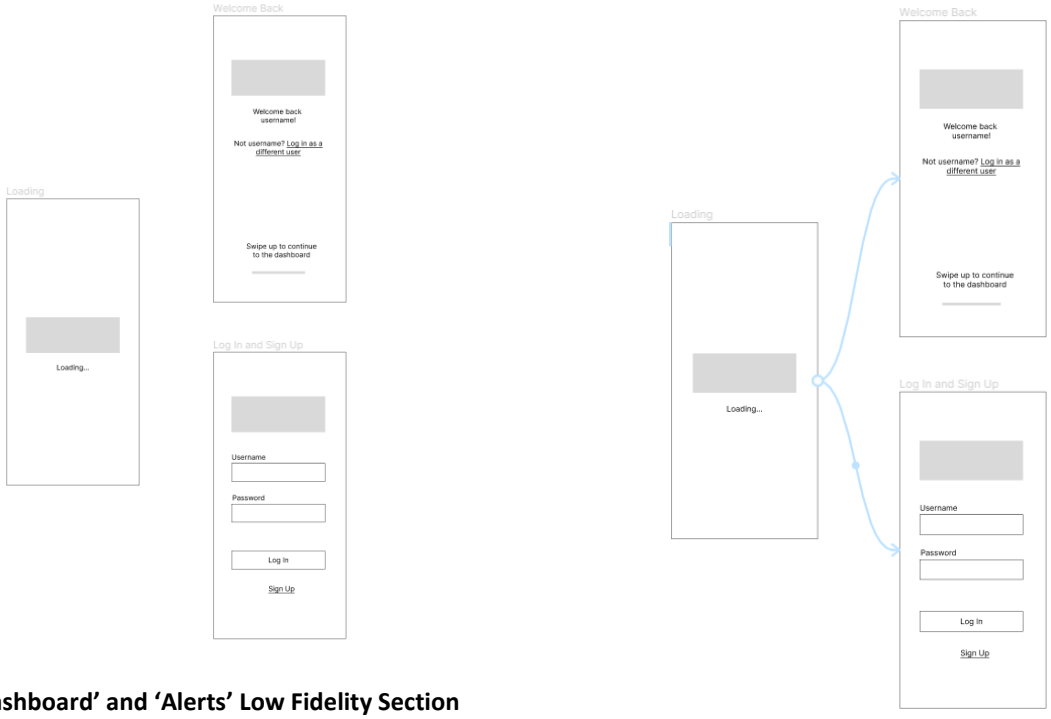
Overview of Low Fidelity Prototype 1:



Drawing from each of the three presented wireflows, I produced the following low fidelity prototype of the Cultivate interface in Figma, combining the most effective sections of design that achieve each of the core functionalities.

'Login and sign up' Low Fidelity Section

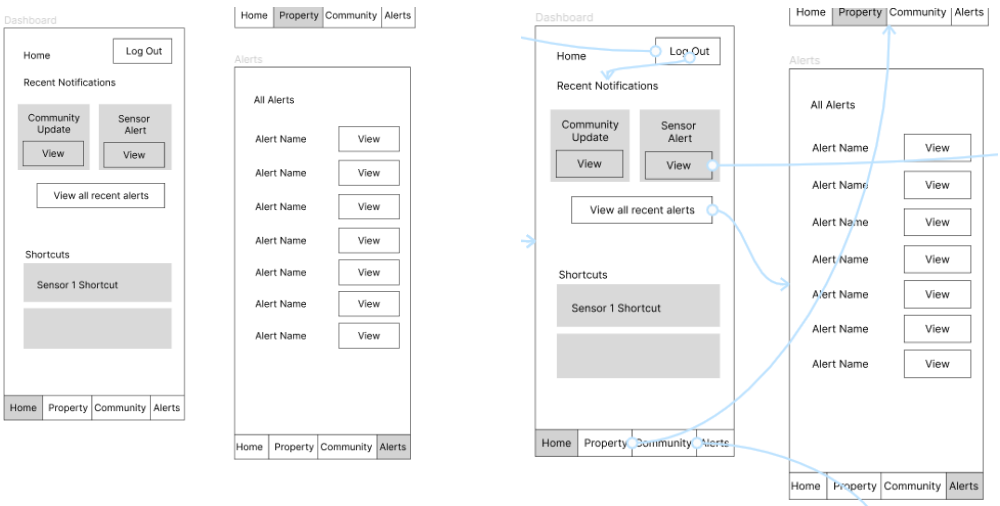
Wireflow 2 and 3 implement were selected as having the most functional and visually aesthetic login and signup wireframes. The low fidelity design incorporates the 'Login' and 'Form' design pattern that they employ.



'Dashboard' and 'Alerts' Low Fidelity Section

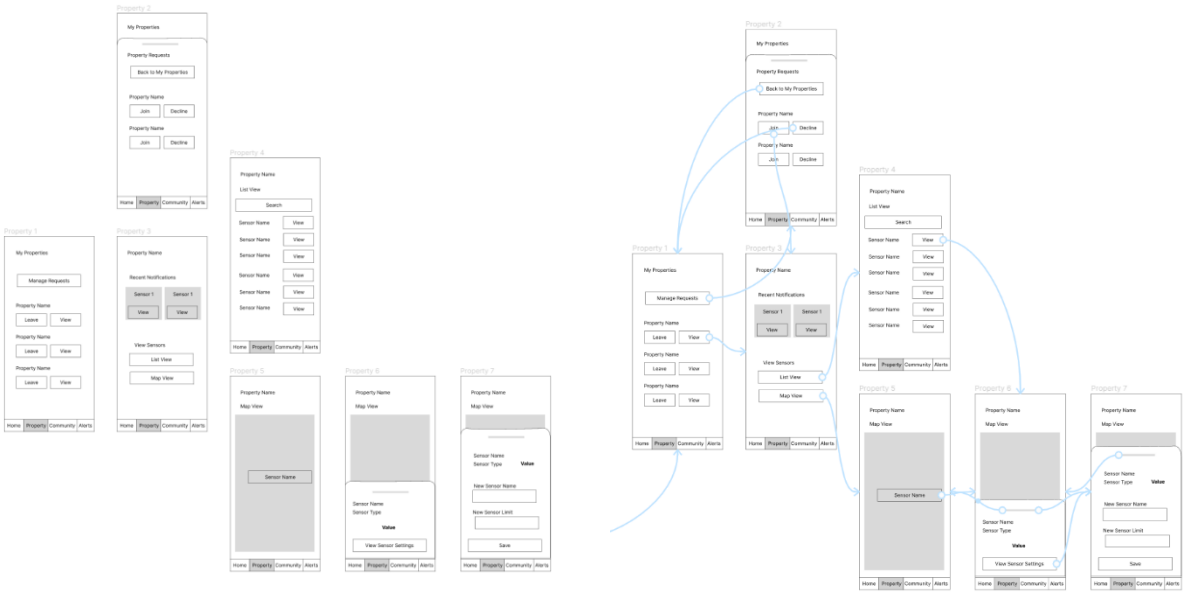
Wireflow 3 was selected for the 'Dashboard' and 'Alerts' section, as it achieves the most intuitive dashboard screen. This is achieved through design patterns such as 'Notifications', 'Progressive Disclosure' (alerts can be expanded), and 'Carousel' to present data. A customisable list of shortcuts is also offered to the user.

Additionally, the 'Repeated Menu' design pattern positioned at the bottom of each screen will was selected as the least obstrusive but still easily accessible navigation system for the application.



‘My Property’ Page Low Fidelity Section

The ‘My Property’ screen features of wireflow 3 was selected as optimal in conjunction with the specialised screens for editing particular settings that wireflow 1 offers. This combination is created through the ‘Modal’ overlay, which provides the user with further information on a selected sensor, which has a settings tab and other sequences that can be followed in the overlay- the ‘Sequencing’ design pattern of wireflow 3. However, during the design of the low fidelity prototype, a modification was made to the ‘Recent notifications’ section of wireflow 3. Instead, the most effective choice to display detailed notification data was the ‘Carousel’ design pattern characteristic of the overall wireflow, to display the notifications list. In this case, the list showcases alerts from the selected property’s sensors.

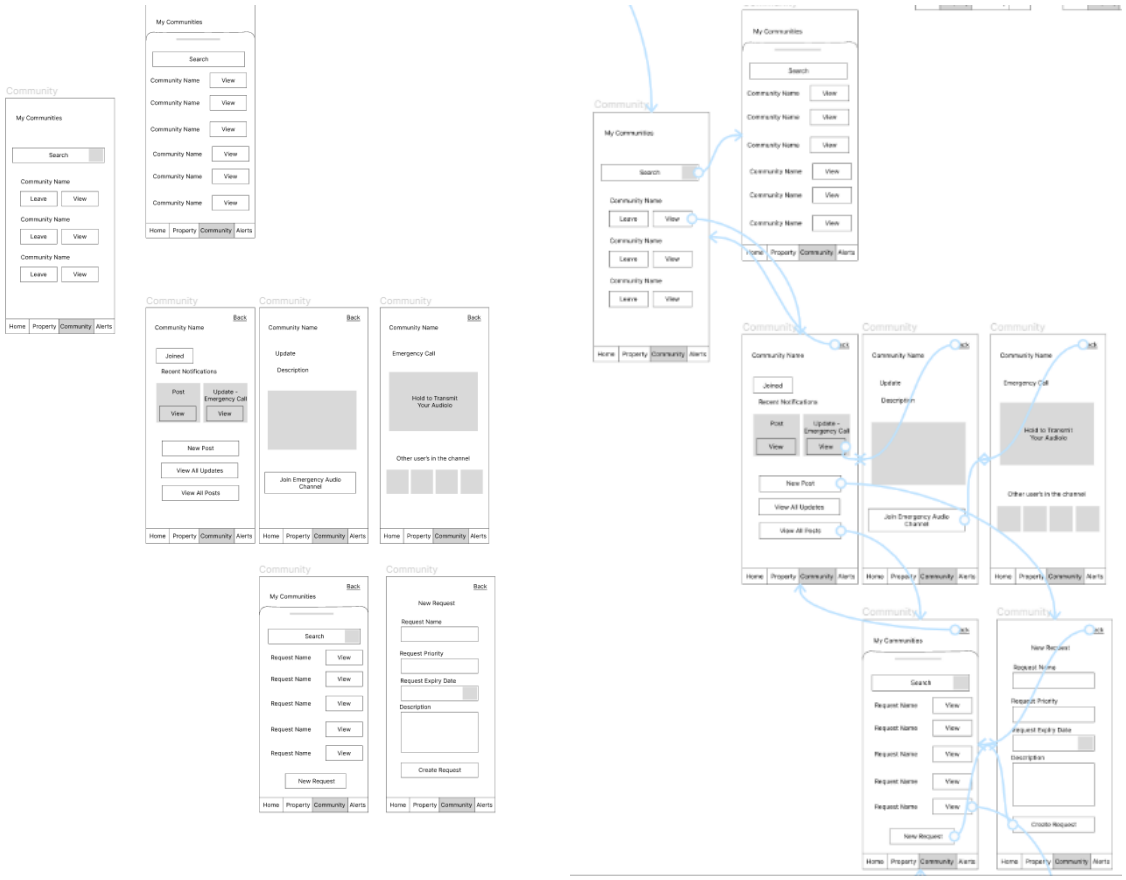


‘Onboarding (‘Create a new property’) Low Fidelity Section

During the onboarding process of wireflow 1, the nonsequential steps allow user’s to receive ‘closure’ of their progress through the registration process. This is shown through graying out the ‘Create Property’ button once the user has finished setting up a property, guididing them to the only available option left ‘Join Community’. This is effective in encouraging the user to continue the onboarding process. By combining this encouragement with the seamless sequential onboarding process of wireflow 3, the progress indicator ‘Step 1/8’ reduces feelings of overwhelm for the user. The more detailed registration inputs from wireflow 3, such as the installation date ‘Calendar Picker’ design pattern, and the request for the user’s financial account details, provide a fully functional onboarding process.

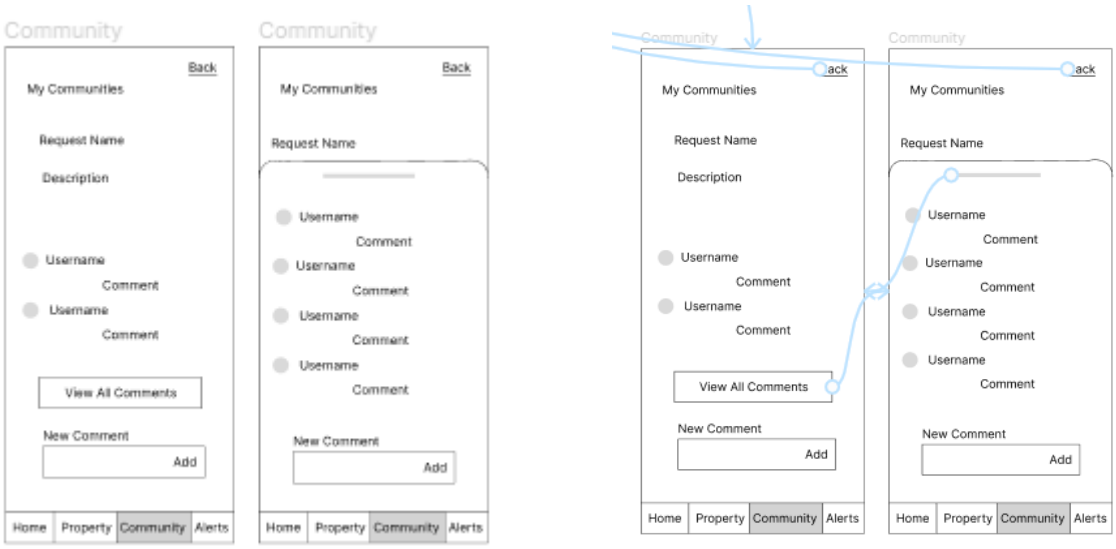
‘My Community’ Page Low Fidelity Section

Wireflow 3 is the most effective choice for the ‘My Community’ design in relation to functionality. Although wireflow 2 implements the ‘Chunking’ design pattern to group ‘Request’ related information and ‘Update’ related information, wireflow 3 combines the ‘Notifications’ into a central location. Thus, when a user is quickly observing the community page, they will absorb information from both sections. However, during the design of the low fidelity prototype, a modification was made to the ‘Recent notifications’ section of wireflow 3. Instead, the most effective choice to display detailed notification data was the ‘Carousel’ design pattern characteristic of the overall wireflow, to display the notifications list. In this case, the list showcases community notifications.

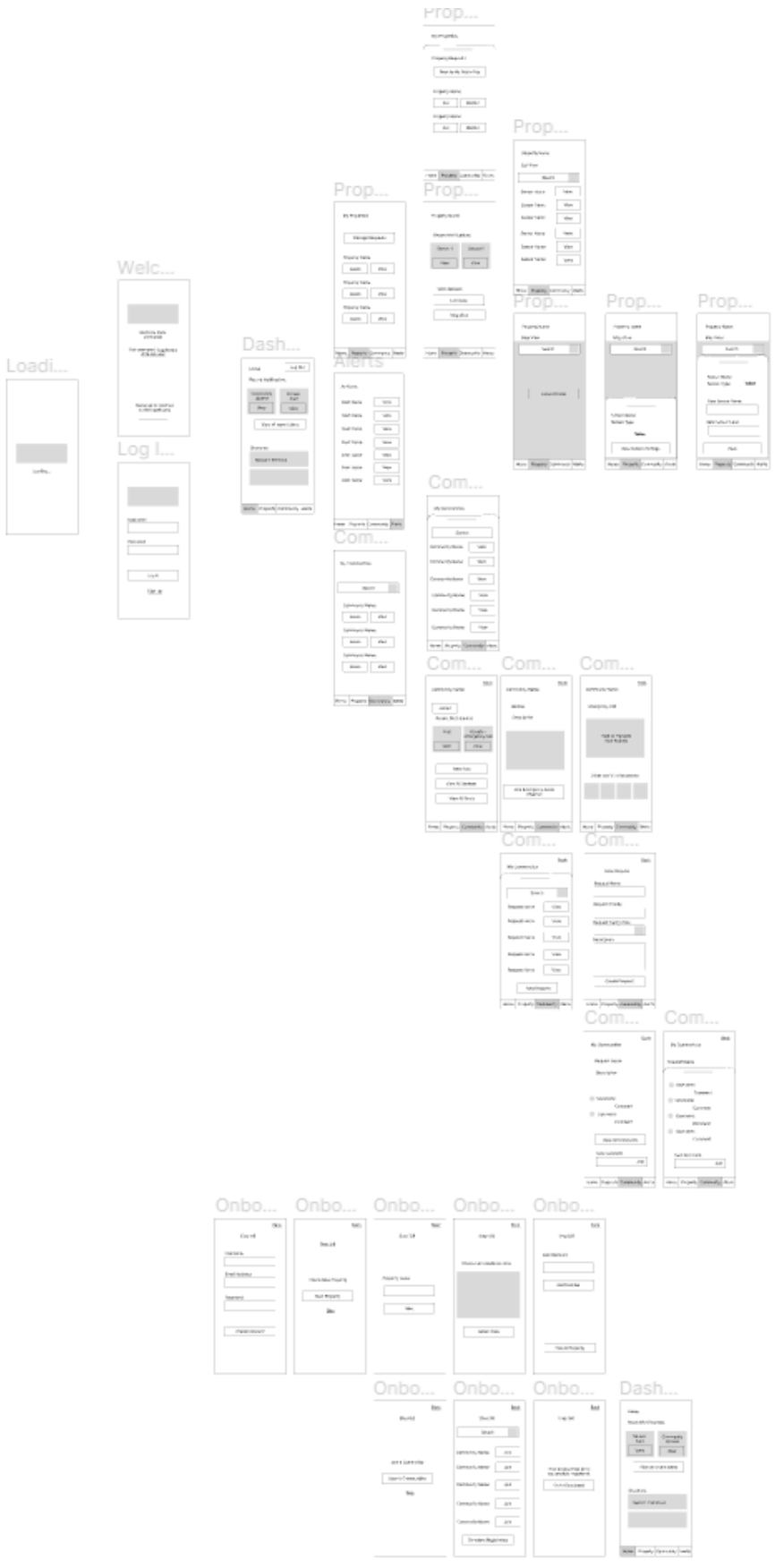


‘Add a comment’ Low Fidelity Section

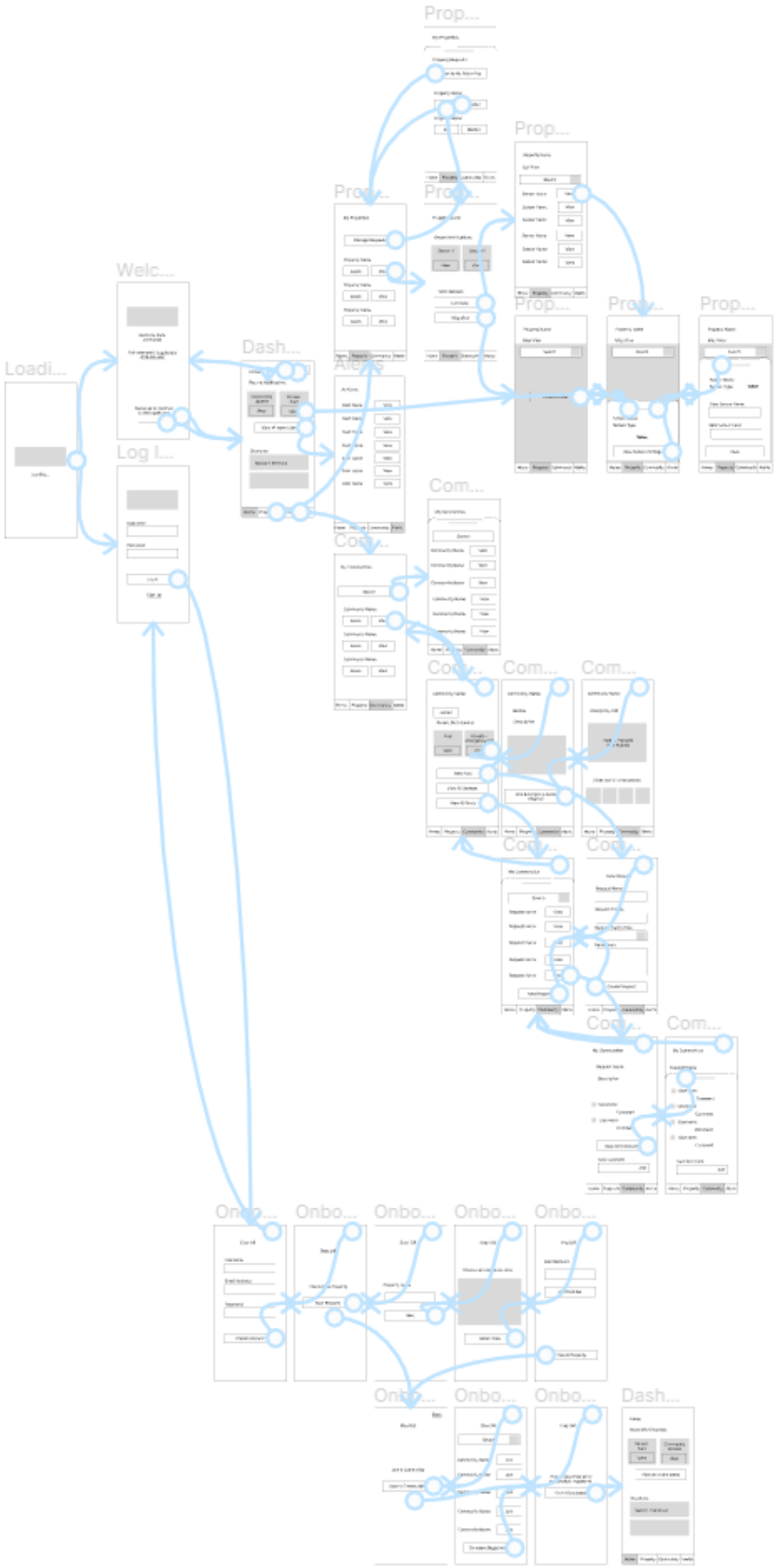
Wireflow 3 offers the most detailed consideration of the commenting system, incorporating the ‘Progressive disclosure’ design pattern to allow users to preview recent comments combined with the ‘Comment box’ pattern so that new comments can be efficiently made.



Overview of Low Fidelity Prototype 2:



With screen interactions:



COMBINED LOW-FIDELITY PROTOTYPE

This section outlines the design decisions during the merge of Low Fidelity Prototype 1 and Low Fidelity Prototype 2. The resulting low fidelity prototype was created from the combination of simplicity from Low Fidelity Prototype 1, and excessive detail in Low Fidelity Prototype 2. Ultimately, this convergence of conflicting design principles resulted in an effective balance. Additionally, through the creation of the combined prototype, some new features to support the core functionality of Cultivate was realised. A significant point to note however is that a lot of the features in the respective member's prototypes were common over both interfaces, thus deciding which presentation to follow for each section often depended on the consistency for that fragment with the entire prototype.

'Login and sign up' Low Fidelity Section

Low Fidelity Prototype 1 and Low Fidelity Prototype 2 both include similar designs for the Login and Sign up screens. The main variation is that Low Fidelity Prototype 1 features a menu initially for users to select between 'Login' and 'Sign up', creating a sequence of steps. Comparatively, Low Fidelity Prototype 2 allows the user to use the 'Login' feature with only one click (which will be used more regularly versus 'signing up', which will only be invoked once for every user). Therefore, the concept of the 'Login and sign up' page and the 'Welcome back' screen is incorporated from Low Fidelity Prototype 2. However, Low Fidelity Prototype 1 demonstrates the value of a clear interface to invite the responder into the application. Therefore, the 'Welcome back' interface has incorporated clear buttons influenced by Low Fidelity Prototype 1. The loading page is an additional feature, which has been improved by incorporating a friendly welcome message from Low Fidelity Prototype 1.

‘Dashboard’ and ‘Alerts’ Low Fidelity Section

The ‘Dashboard’ design pattern allows a user to digest multiple sources of information when quickly viewing the interface, essential given the number and critical nature of sensors and updates. Low Fidelity Prototype 1 includes important navigation features on the interface, creating a simple decision that does not overwhelm the user. By combining the navigational buttons offered on the homepage from Low Fidelity Prototype 1, and the ‘Carousel’ design pattern to highlight important notifications from Low Fidelity Prototype 2, an effective balance between information and action is achieved. Further, the translation of the ‘Notifications’ button in Low Fidelity Prototype 1 to the ‘View all recent alerts’ button in Low Fidelity Prototype 2 creates a considered design. As Low Fidelity Prototype 1 does not contain a navigational design pattern, the ‘Repeated Menu’ positioned at the bottom of the screen completes the combination at this section. During design, it was also realised that it is unlikely the user would log out of their phone. Instead, a button is positioned in the top right hand corner to access the user settings, adapted from Low Fidelity Prototype 1.

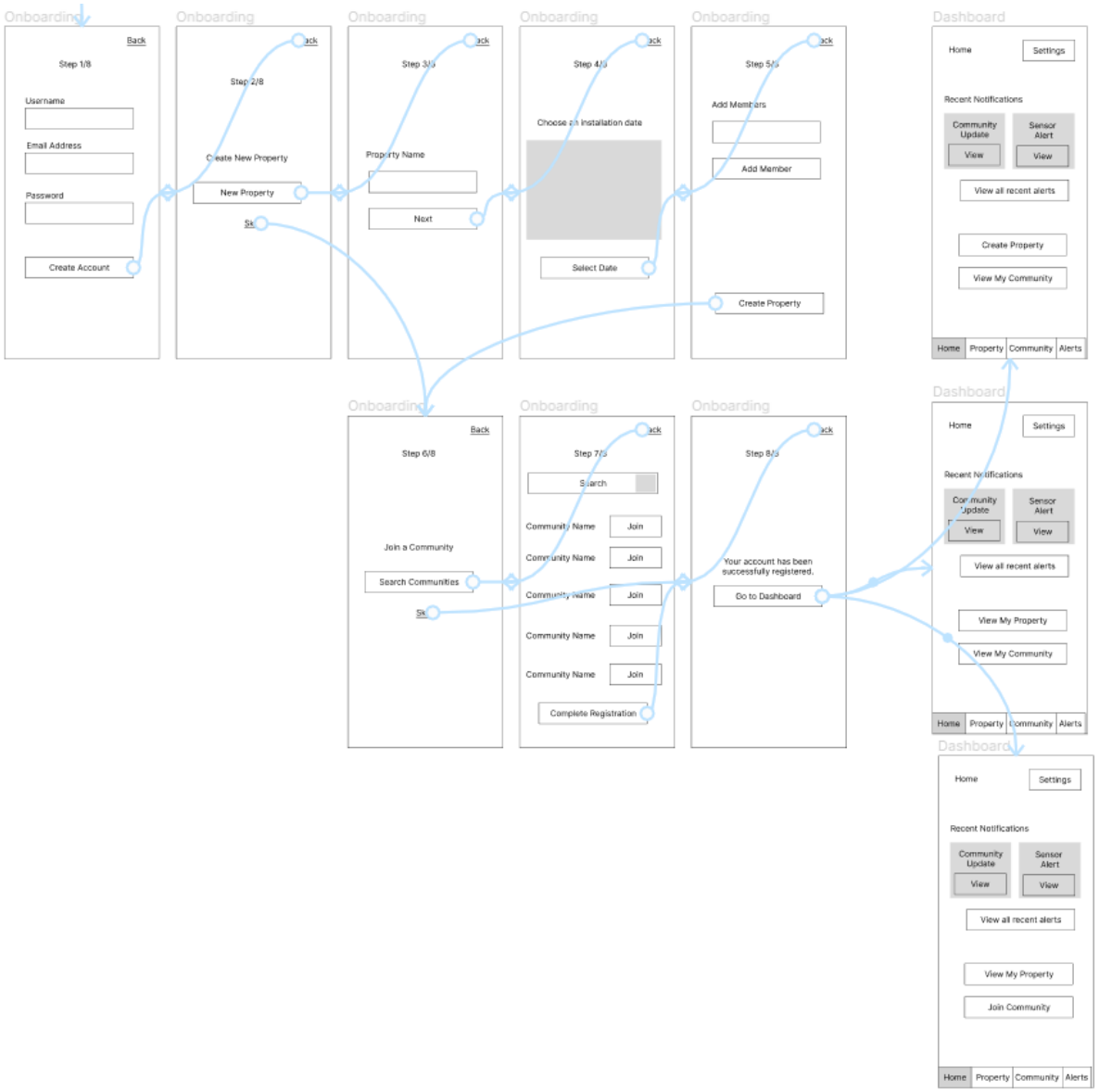
‘My Property’ Page Low Fidelity Section

The ‘My Property’ section of the application combines the practicality of Low Fidelity Prototype 1 with the innovative map function of Low Fidelity Prototype 2. Structurally, Low Fidelity Prototype 1 influenced the idea of each user owning a single property, as opposed to the property management system proposed in Low Fidelity Prototype 2. This aligns the small-community scope of Cultivate’s problem domain, and the application’s intention of providing a centralised space for property owner’s to remain updated about their properties. As a significant portion of the features in Low Fidelity Prototype 1 translate directly to Low Fidelity Prototype 2 (such as the sensor interface showcasing its name, current value, and further information), the ‘Modal’ design pattern overlaying the ‘Map navigation’ from Low Fidelity Prototype 1 are continued in the combined design. The separate ‘Property alerts’ screen, and ‘Property settings’ was incorporated form Low Fidelity Prototype 1.

'Onboarding ('Create a new property') Low Fidelity Section

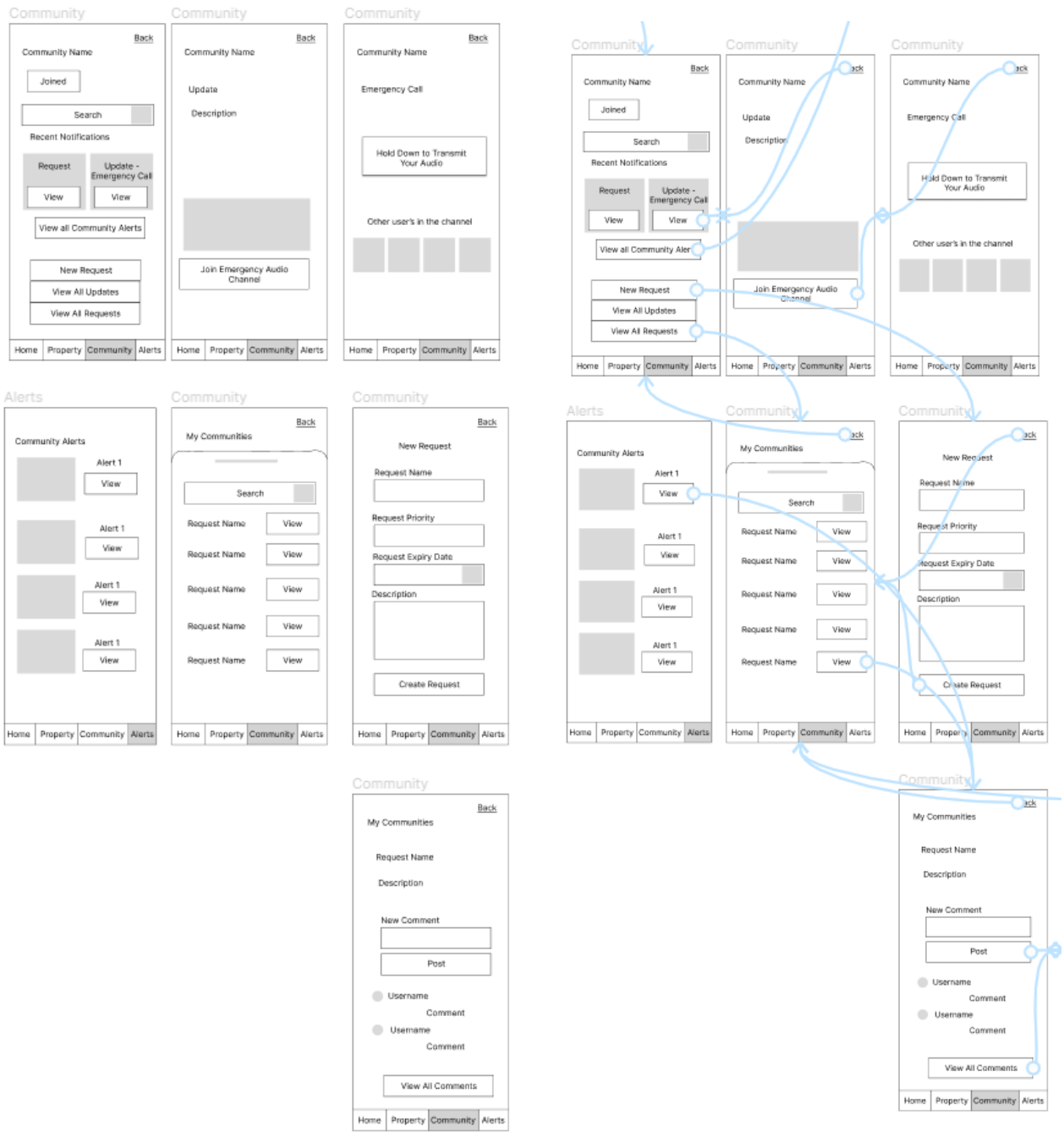
As mentioned in the introduction of this section, the common features and fields between the interfaces means that choosing either low fidelity prototype to pursue is ultimately equivalent. Both Low Fidelity Prototype 1 and Low Fidelity Prototype 2 present a 'Form' design pattern to collect the user's input and 'Create a new property'. The only point of difference identified is that Low Fidelity Prototype 2 breaks down steps into single interfaces, aligning with the 'Wizard' design pattern. In this way, the onboarding process becomes less intimidating. Therefore, the fields from Low Fidelity Prototype 2 are combined with the structure of Low Fidelity Prototype 1. The combined low fidelity prototype also kept other set-up required for a new user account such as 'Join a Community'. The inclusion of 'Choose an installation date' from Low Fidelity Prototype 2 instead of editing the 'Sensor details' from Low Fidelity Prototype 1 reflects our realisation in the scope of the project. Here, it was realised that installers would also connect the sensors to the dashboard, and therefore it is not in the scope of Cultivate. The signficiant influence of Low Fidelity Prototype 1 was the idea of changing the 'View My Property' and 'View My Community' dashbord according to whether the user had connected to a property and community. This is shown if a user skips a step as they are setting up.


Onboarding
Back
Step 1/8
Username
Email Address
Password
Create Account
Onboarding
Back
Step 2/8
Create New Property
New Property
Skip
Onboarding
Back
Step 3/8
Property Name
Next
Onboarding
Back
Step 4/8
Choose an installation date
Select Date
Onboarding
Back
Step 5/8
Add Members
Add Member
Create Property
Dashboard
Home
Settings
Recent Notifications
Community Update
View
Sensor Alert
View
View all recent alerts
Create Property
View My Community
Home
Property
Community
Alerts
Onboarding
Back
Step 6/8
Join a Community
Search Communities
Skip
Onboarding
Back
Step 7/8
Search
Community Name
Join
Community Name
Join
Community Name
Join
Community Name
Join
Community Name
Join
Complete Registration
Onboarding
Back
Step 8/8
Your account has been successfully registered.
Go to Dashboard
Dashboard
Home
Settings
Recent Notifications
Community Update
View
Sensor Alert
View
View all recent alerts
View My Property
View My Community
Home
Property
Community
Alerts
Dashboard
Home
Settings
Recent Notifications
Community Update
View
Sensor Alert
View
View all recent alerts
View My Property
Join Community
Home
Property
Community
Alerts



'My Community' Page Low Fidelity Section

Low Fidelity Prototype 1 and Low Fidelity Prototype 2 both execute similar features and a similar user-friendly interface in their design. Therefore, the combination of their designs was straightforward. Through the use of



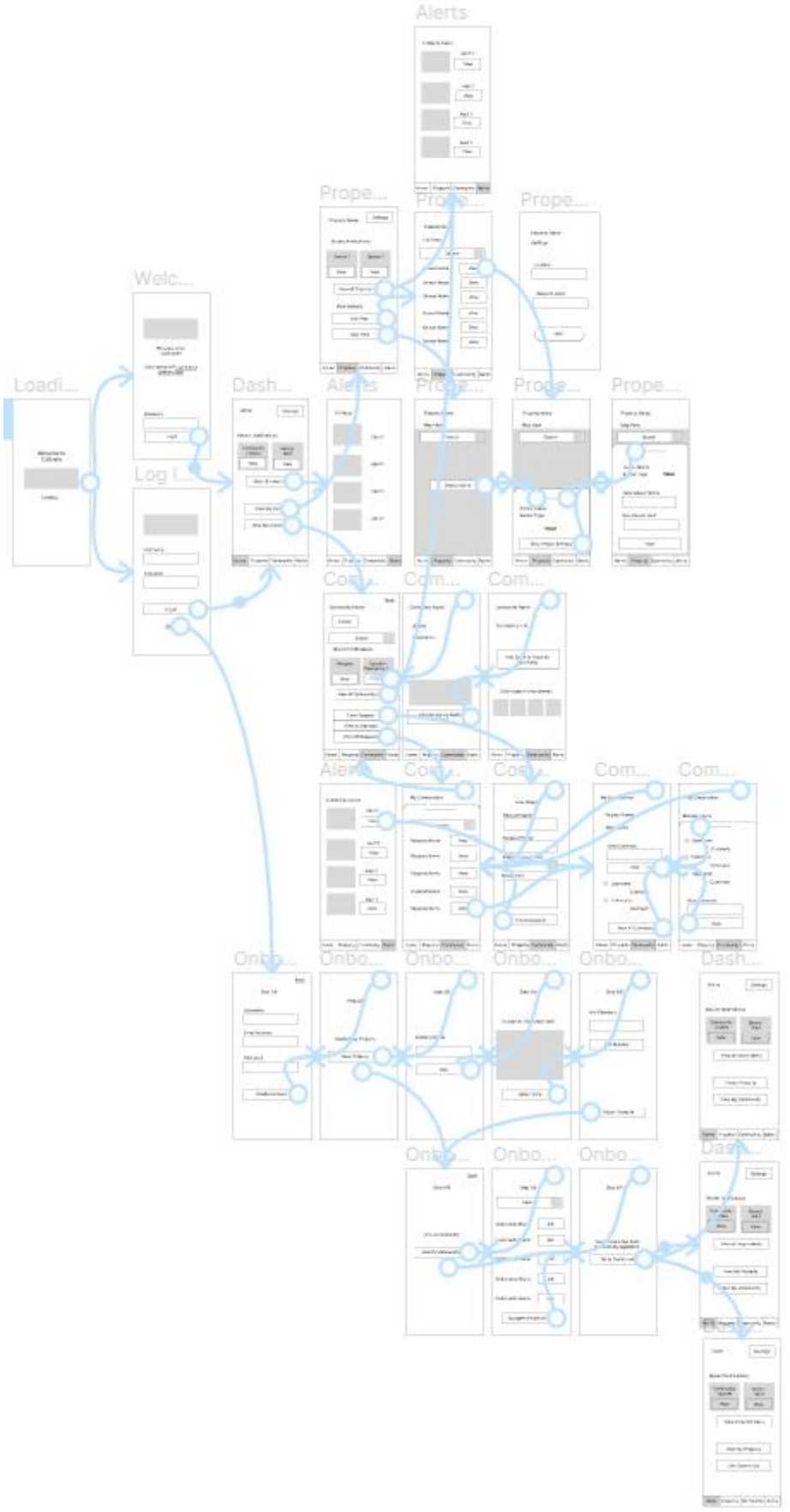
'Add a comment' Low Fidelity Section

The 'Add a comment' section is extremely similar across both prototypes, considering the common 'new comment' input and list of all comments. Therefore, in terms of presentation, the 'Modal' overlay contributes the modern interface features from Low Fidelity Prototype 2. The logical 'Chunking' design pattern used on elements organised in Low Fidelity Prototype 1 influences the placement of 'New Comment' directly beneath the post. This aligns with Nielsen's sixth usability heuristic, ' 6.Recognition rather than recall'.


Community
Back
My Communities
Request Name
Description
New Comment
Post
Username
Comment
Username
Comment
View All Comments
Home
Property
Community
Alerts
Community
Back
My Communities
Request Name
Username
Comment
Username
Comment
Username
Comment
New Comment
Post
Home
Property
Community
Alerts
Community
Back
My Communities
Request Name
Description
New Comment
Post
Username
Comment
Username
Comment
View All Comments
Home
Property
Community
Alerts
Community
Back
My Communities
Request Name
Username
Comment
Username
Comment
Username
Comment
New Comment
Post
Home
Property
Community
Alerts

Overview of Combined Low Fidelity Prototype

With screen interactions:



DISCUSSION ON ACCEPTANCE OF KEY DESIGN PATTERNS

Any successful software evidently needs to have key design patterns. The use of key design patterns creates a more intuitive, accessible, and robust user interface, defining elements and solutions to common problems faced during application design, compiled by academic research and industry experience. The acceptance of key design patterns has guided and accelerated our development process by emulating established design patterns.

The key design patterns incorporated in our combined low fidelity prototype are as follows:

1. **Dashboard design pattern**

In our combined low fidelity prototype, we accepted the dashboard design pattern as seen in diagram X. It uses a minimalistic and appropriate display of icons which allow the user to navigate the core functions of the app. The use of an operational dashboard offers the ability for users to view their data in greater depths while also focussing on the important aspects of our app (such as notifications and sensor data). It is vital that the most needed functionality of our app is displayed through the dashboard, thus why we chose the operational dashboard.

2. **Continuous scrolling design pattern**

Through the use of a continuous scrolling design pattern, the user is able to view more information than what would normally fit on an app page, which minimizes the amount of page-to-page navigation within the app. It reduces the need to click to more pages, instead allowing the user to simply scroll down by automatically loading the next page once the user reaches the bottom of the current page. This design pattern should increase the effectiveness of the user interface and allow for easier user interaction.

3. **In combination with Progressive Disclosure and thumbnail design pattern**

With progressive disclosure, the user should be able to navigate through the app without being distracted, presenting only the relevant information to the user. This should decrease the difficulty of learning how to use the app and let them focus on the necessities. Using the thumbnail design pattern is also efficient as displaying only the thumbnail of a picture instead of the entire picture both works well with progressive disclosure and allows the user to interact with the app without having to view full sized pictures, saving space for other aspects of the app. If the user needs to view a picture, they are able to use the thumbnail to see which picture to enlarge.

4. **Search Filters**

Search filters allow users to specify select posts and navigate the app more efficiently, proving to be a successful design pattern. Using this feature, the user experience is simplified and allows the user to specify what post/ user they are in search for. By allowing the user to search on their own terms, it increases effectiveness of the apps' search feature and allows for a better user interface. In a community driven app, it is vital that there is a proper search function, which is why this design pattern is useful and why we incorporated it into our software.

5. **Notifications**

This design pattern is essential in an application such as ours, as our app needs to send alerts to users in case of emergencies or other functions which need notifications. Informing users of events or posts is vital however it also has to work with other design patterns such as the progressive disclosure and thumbnail design pattern, as it is important to minimize distractions. We have resolved this issue by allowing users to change their notification preference in settings to increase efficiency within the app, while utilizing the core functionality of the software. The notifications the user may see will be adjusted to the user while keeping in mind their alert preference, drawing attention to vital updates/ alerts.

6. **Map navigator**

As per the app functionality, the user should be able to create, edit and manage properties/ sensors, which is a task easier done by integrating a map navigator. This also allows the user to visually see information and have a more interactive and intuitive user experience. Map navigators provide a clear and contextual visual input of information which allows the user to select and view properties/ sensors and gain data on them when needed. While this function isn't essential, it ties in with other key design patterns on the whole which solidify the user interface and increase efficiency in software usage and productivity.

7. **Follow design pattern**

This feature is an important aspect of the community side of the app, allowing users themselves to choose to receive updates on certain products, posts, or events. This personalizes the user experience and lets them create their own notifications. Using a following design pattern is useful for users who want to keep track of certain aspects within the app and can further be utilized with the appropriate notifications design pattern, allowing for a more useful user interface. This feature also increases the importance and functionality for the community aspect of the app, as users will have a better user experience by having access to a following design pattern.

8. **Steps Left design pattern (during onboarding)**

The steps left design pattern creates a better user experience by essentially splitting a user's process of filling out data into smaller steps and guiding them through the process. This is an intuitive feature which should benefit the user greatly and assist with maintaining the users attention and working with other design patterns that have been put in place to minimize distractions and increase efficiency, thereby creating a better user experience. Using a navigation block to illustrate how the process is going creates a better interface for the user as it informs them of how long it will take to fill in a form, which is why this design pattern is useful.

9. **Playthrough (during onboarding)**

This design pattern will be used to introduce users to the basic functionality of the software and allow them to learn how to navigate/ use the app before experiencing the whole software. By letting users learn how to use the functionality of the software by using a playthrough, it minimizes chance of error and increases the overall user experience. A skip feature will of course also be added to ensure users who are perhaps more experienced with this type of settings will not have to go through a playthrough to access the apps features. This design pattern is useful for introducing the core aspects of our app and assists users on learning how to use the app.

10. **Activity Stream (my community)**

An overview of the recent actions (history of actions) would be a benefit to the user as they are able to see the most recent changes for their sensors/property, or even updates to favourited posts or requests. Allowing users to show summaries of activities in the community page allows the user to stay relevant with the online community as well as being notified of activity within the community page. Showing a small summary of certain changes to posts, for example, fits well with the other design patterns as it gives the user freedom of choice whether to explore recent activities or not. It allows the user to engage more with the community more and is a beneficial design pattern.

11. **Trigger persuasive design pattern**

Through the use of 'recent alerts' and recent feeds positioned around the interface to remind the user to look further into new options, triggers are used to benefit the user experience through the app. The point of using a tripper persuasive design pattern (in the form of notifications), is that it makes the experience easier for the user by notifying them of something happening instead of them having to check manually. Due to the nature of our software, triggers are important as well as it can signal for emergencies, otherwise the user would have to check

the app constantly. Implanting these triggers were useful for both the user experience and work well with other key design patterns.

12. **Icon Menu and Repeated Menu design pattern**

An icon menu used throughout our app is beneficial as it displays to user a menu item in order to navigate the app and provides a better user experience since it is a more intuitive user interface. This design pattern also leaves more space for other features within the app and is a more efficient as a design for phones especially. A repeated menu is also used for both continuity of design as well as ensuring the user interface is consistent. This is useful as integrating both the icon menu and repeated menu designs will increase usability and decrease the navigation time spent for the app, as these design patterns introduce a more efficient user interface. These types of menus minimize user error and confusion as they are designed to be simple and intuitive, lessening the time the user needs to spend becoming familiar navigating within the app.

Therefore, through the acceptance of industry-identified design patterns, we have created a professional, modern software application which is designed to allow the user to quickly navigate to the appropriate aspect they need as easily as possible. This enables our interface to achieve principles such as consistency (employing robust methods of collecting user input and outputting system data), familiarity (achieved through a combination of User Interface design patterns and cognitive design patterns), and guidance (through streamlined solutions to application navigation, and user documentation).

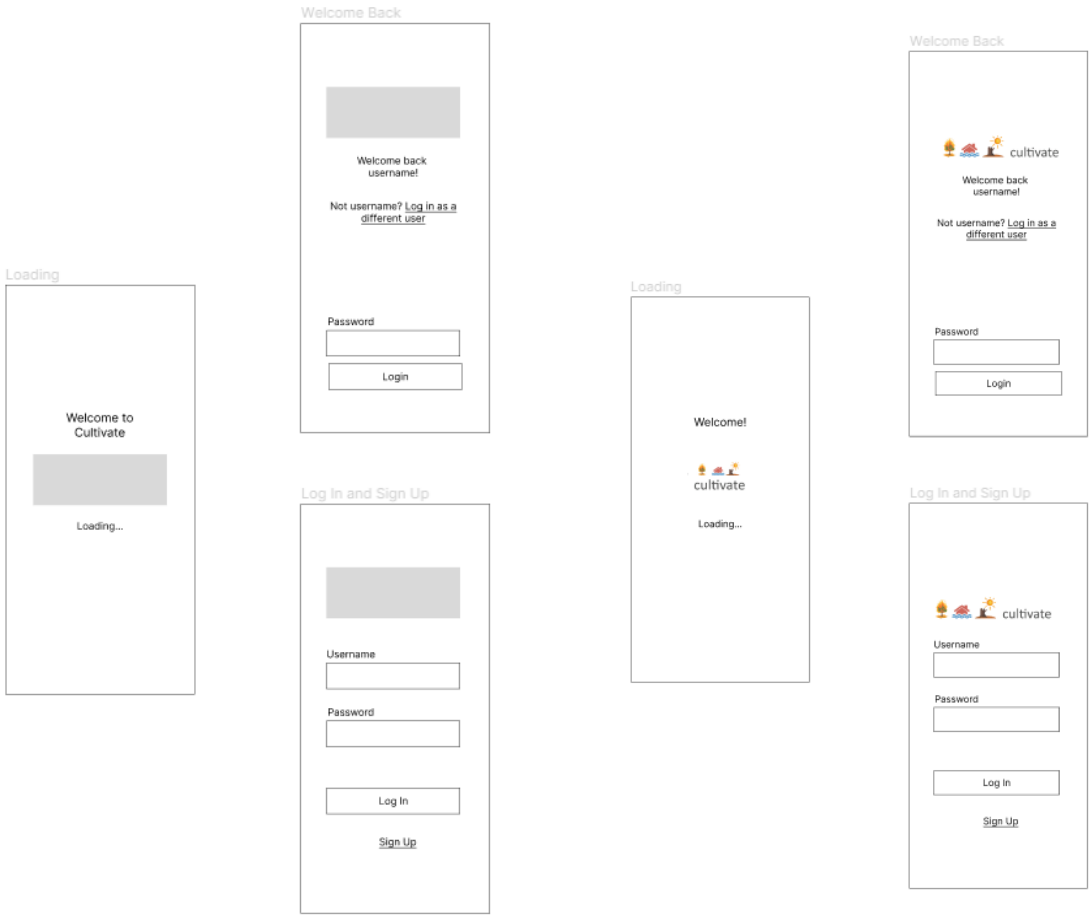
DESIGN ITERATIONS

Now that the combined prototype has been completed, the following section describes the iterations that our design experienced as we moved through the creative process, progressing a low fidelity prototype to our high-fidelity final design. Beginning this process, the design team agreed that the combined prototype successfully incorporated all of the necessary functionality. Therefore, this section primarily focused on the presentation, aesthetics, and usability of the application for each fragment of the prototype, aligning which each scenario.

'Login and sign up' Iteration Section

Beginning with the 'Login and sign up' interfaces, iterations were intended to ensure that Cultivate was inviting and straightforward for all members of the family and community to operate, particularly considering our target user of property owners aged 40-70. Located in rural spaces, such users may not have much technical experience. Therefore, the use of the 'Login' design pattern allows users to identify themselves using a combination of an email address and password, and the 'Registration' design pattern ensures that the data of users is secuired in a user friendly way.

The design of this interface does not vary much. Beginning with the Combined Low-Fidelity Prototype, the Cultivate logo is added to the gray low-fidelity markers. In inserting the Cultivate logo, it was determiend that the system dialogue message should present, 'Welcome!'.



Next, a greater level of colour was added to the interface to establish the Cultivate branding. It was deduced that the interface should have a white background so that it will maintain visiblity while users are performing manual labour or in dangerous conditions. The burnt orange colour in the logo, complemented by the blue

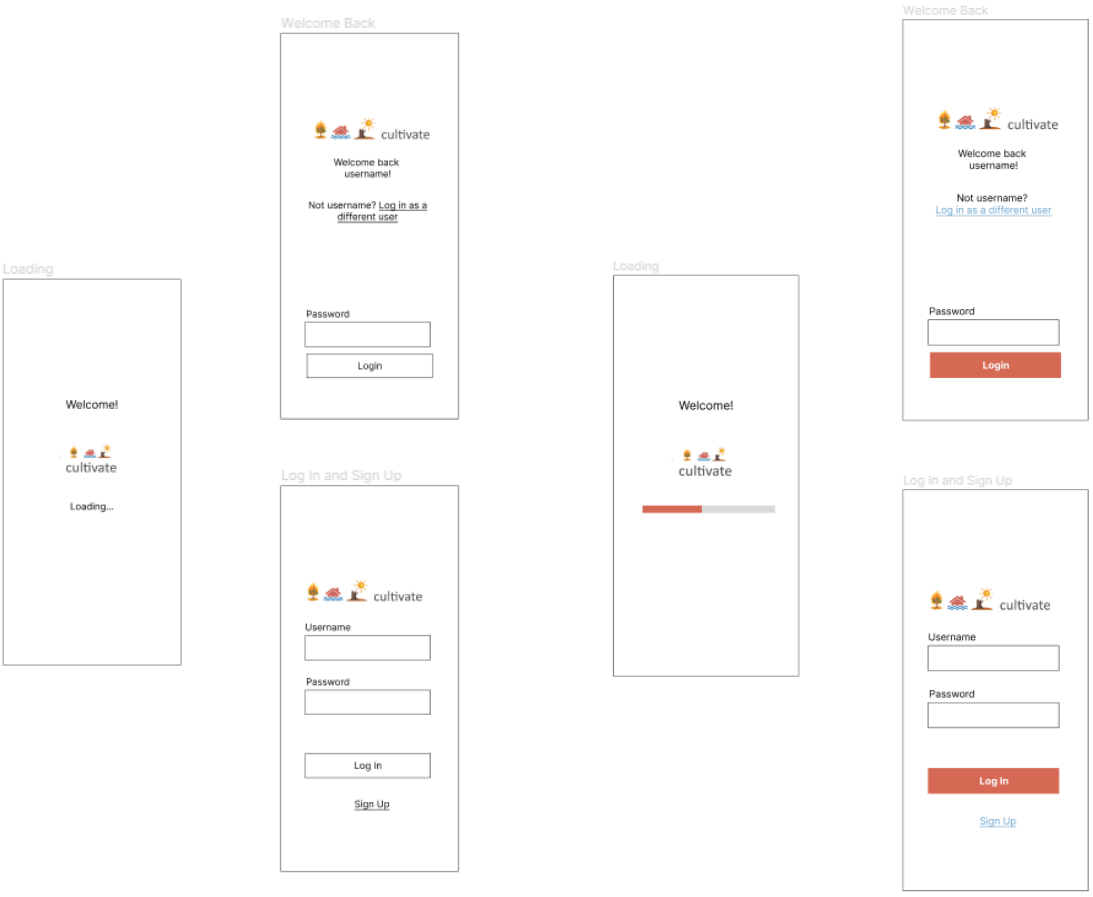
representing the flood natural disaster, were established as the interface colours. This allowed the interface to draw user's attention to specific elements, in an aesthetic way.

HEX: #68a2ce

RGB: rgba(99,159,204,255)

HEX: #d56953

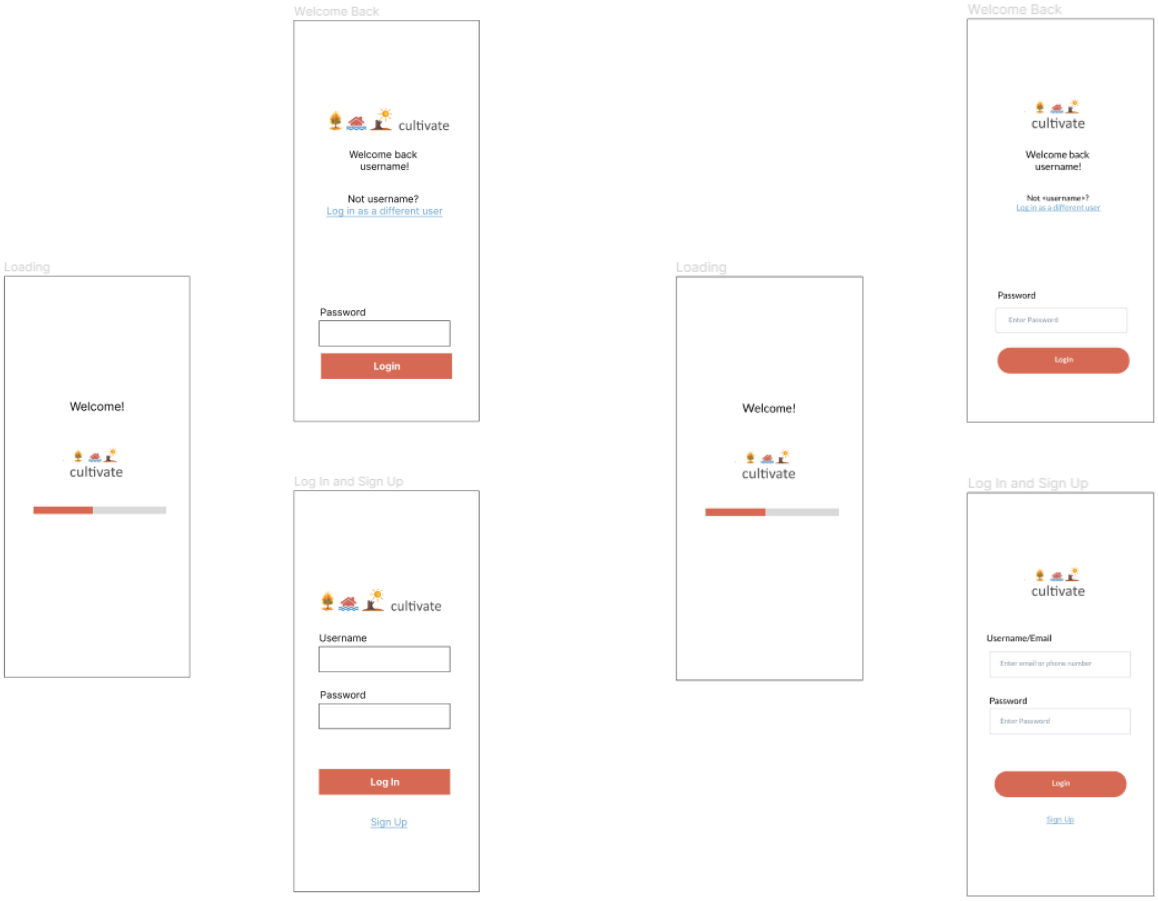
RGB: rgba(213,105,83,255)



Next, creating a modern and clean aesthetic in the interface was focused on. The 'Lato' font is currently an extremely "popular" (Sandu, 2020) font, featuring great style versatility such as 'Light' and 'Regular' to 'Semibold'. The interface shifts from the default Figma text font 'Inter' to 'Lato' 'Regular'. Additionally, the outline on the textboxes are reduced to a soft gray, and instruction text in an input textbox is added. This offers additional guidance to users in a way that does not clutter the interface. When the user begins to enter input into the textbox, their input will replace the instruction. The corners of the textbook were slightly rounded to further improve flow in the interface. The logo was also changed from the horizontal variation to enhance interface consistency. Additionally, a huge visual shift was made by changing the font size of title text to 18px, and the font of text in textboxes to 14px. Significant titles, such as 'Welcome!', was set at 20px.

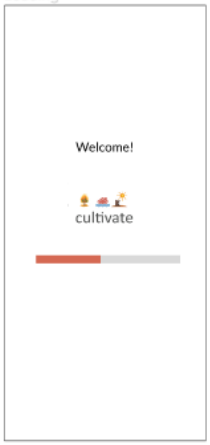
HEX: #c7cad9

RGB:
rgba(199,202,217,255)

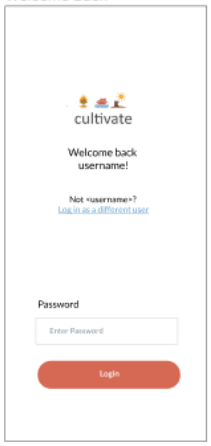


Finally, the 'Login and sign up' interface added icons from the 'Microsoft Fluent UI Icons by Iconduck' and 'Iconsax 6000+ icons', which are downloadable plugins on Figma. The addition of other common interface features such as 'forgot password' and allowing a password to be concealed were added. System messages such as 'sign up' were expanded. Also, drawing from inspiration from common applications, a Log In with Facebook and Log In With Google option is presented. These use icons from the aforementioend Figma icon plugins. The loading bar was also narrowed, as there was no need to take up furhter screen real estate.

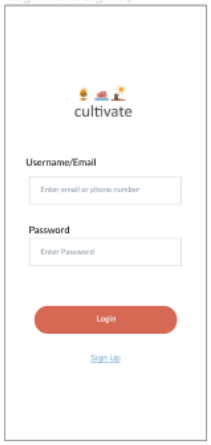
Loading



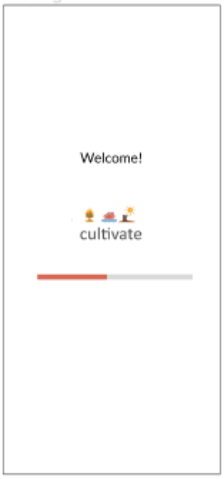
Welcome Back



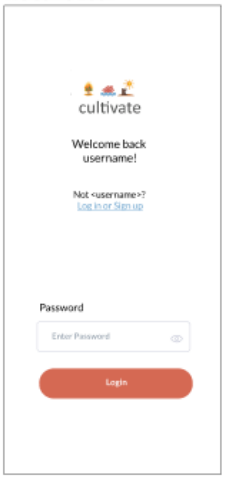
Log In and Sign Up



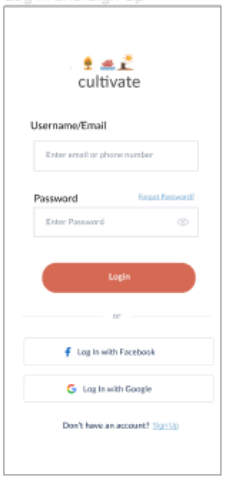
Loading



Welcome Back

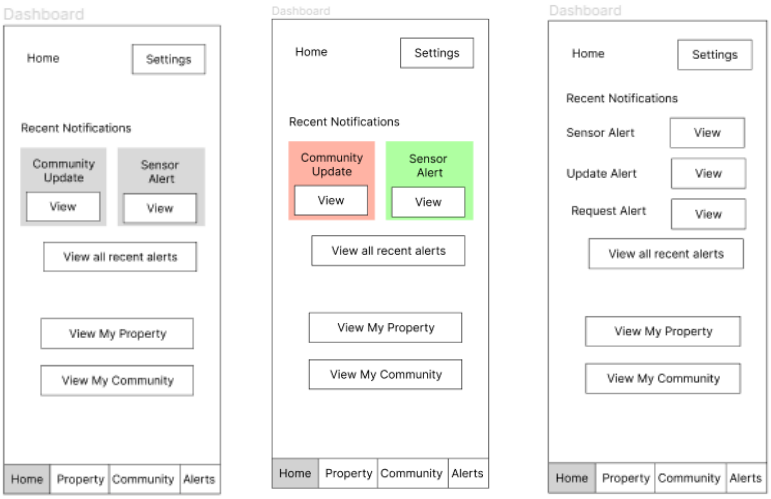


Log In and Sign Up

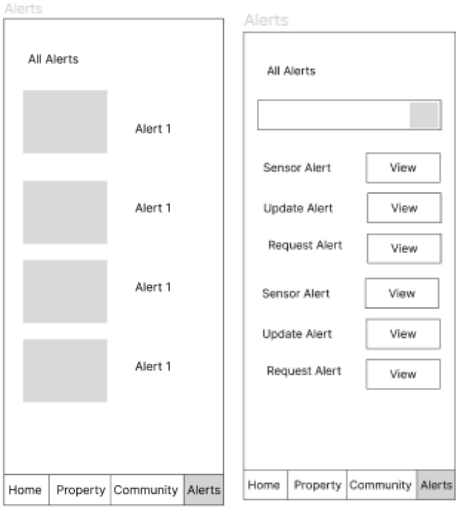


‘Dashboad’ and ‘Alerts’ Iteration Section

Next, the ‘dashboard’ and ‘alerts’ section of the system was built up. Beginning with the Combined Low-Fidelity Prototype, a coloured background is added to the gray low-fidelity markers. However, this intense colour disrupted the professional simplicity that the interface was aiming to achieve. Therefore, the ‘Carousel’ design pattern was replaced by the ‘Progressive Disclosure’ design pattern, displaying the data in a way that invites user to follow the button to view any data at all.

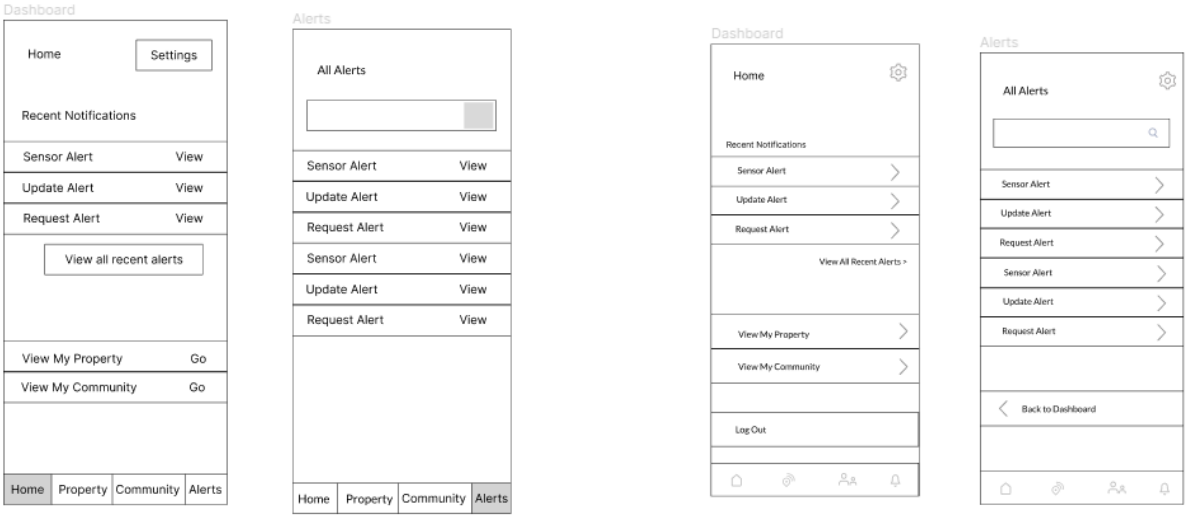


This shift was then reflected in the ‘view all recent alerts’ low fidelity prototype for program consistency. Addiitonally, the opportunity for the ‘Search box’ design pattern which allows users to navigate to a specific alert quickly and with efficiency was added.



Now, the apperance of the ‘Dashboard’ and ‘Alerts’ interface was improved on. Firstly, the buttons were extended across the screen to offer a greater space for users to click to activate the button. We also accepted the ‘Chunking’ cognitive design pattern, which groups related functions together so that users can access them more quicky. This is particularly important in relation to the ‘Recent Notifications’ information, where notifications should be accessible quickly. Additionally, multiple updates may need to viewed at once, thus by reducing the size that each notification takes up vertically but increases the screen space to select a button horizontally, the user can digest more information at once. This also allowed all information for buttons to be aligned to the left of the textbox, allowing users to scan one side of the screen when they are locating important data. In this way, Nielsen’s eighth heuristic ‘Aesthetic and minimalist design’ is hugely considered.

Next, the font sizes of text was changed to a readable and aesthetically clean size of 16px and the ‘Lato’ font replaced the current ‘Inter’ typeface. Additionlly, we decided that the ‘Repeated Menu’ design pattern should



be incorporated with the ‘Icon Menu’ concept, upon reflection at the cluttered appearnace of words to distinguish each navigation option. These icons were from the ‘Microsoft Fluent UI Icons by Iconduck’ and ‘Iconsax 6000+ icons’, which are downloadable plugins on Figma.

Next, the flow of the interface is improved through softening with gray colours, and rounded corners. Additionally, colour is added in a powerful way. This is expanding the interface's colour palette, aligning with Nielsen's second usability heuristic 'Match between system and the real world'. In the real world, red is often used as a warning colour, such as at traffic lights. Therefore, but applying the red to the 'Log Out' feature, the user's attention is caught.

HEX: #a2a2be

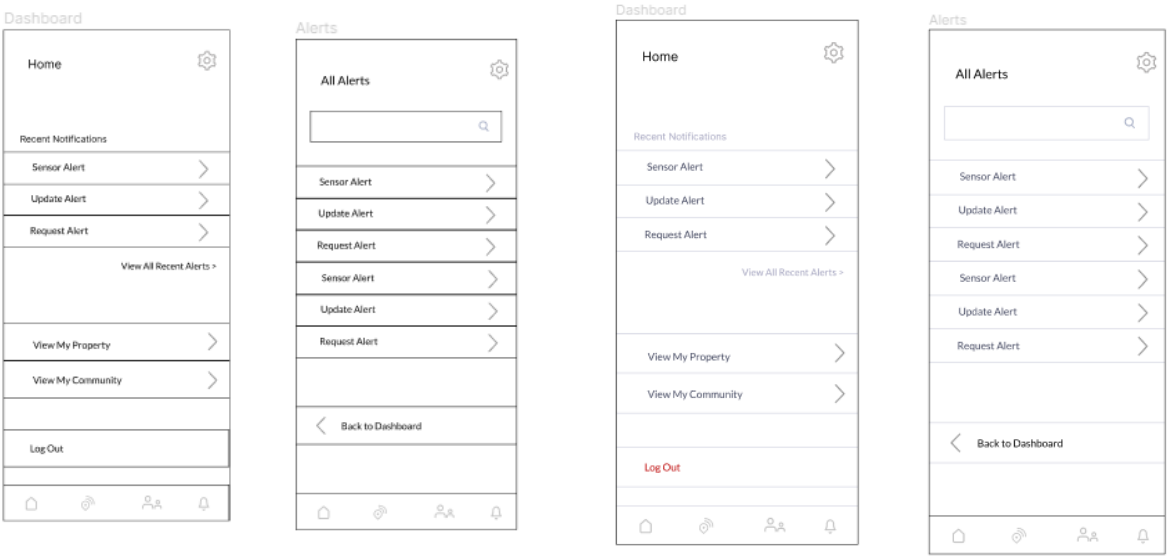
RGB: rgba(162,162, 190 ,255)

HEX: #bd0000

RGB: rgba(189, 0, 0,255)

HEX: #3f445d

RGB: rgba(63, 68, 93,255)



Lastly, colour is added to indicate the status of the alert, and the heading is changed to display the current user that is logged in. These are elements that contribute to the deep absorption of information at a glance of the dashboard. The final change is the use of a lighter grey to outline the dashboard buttons. HEX #f0f0f0 presents a lighter outline than #c7cad9, which is useful in textboxes to draw the users attention, but disruptive in navigational elements on the dashboard.

The colour of the text in the textbox was also changed for the login interfaces, after making this change.

HEX: #f49494

RGB: rgba(244,148, 148 ,255)

HEX: #f4ce94

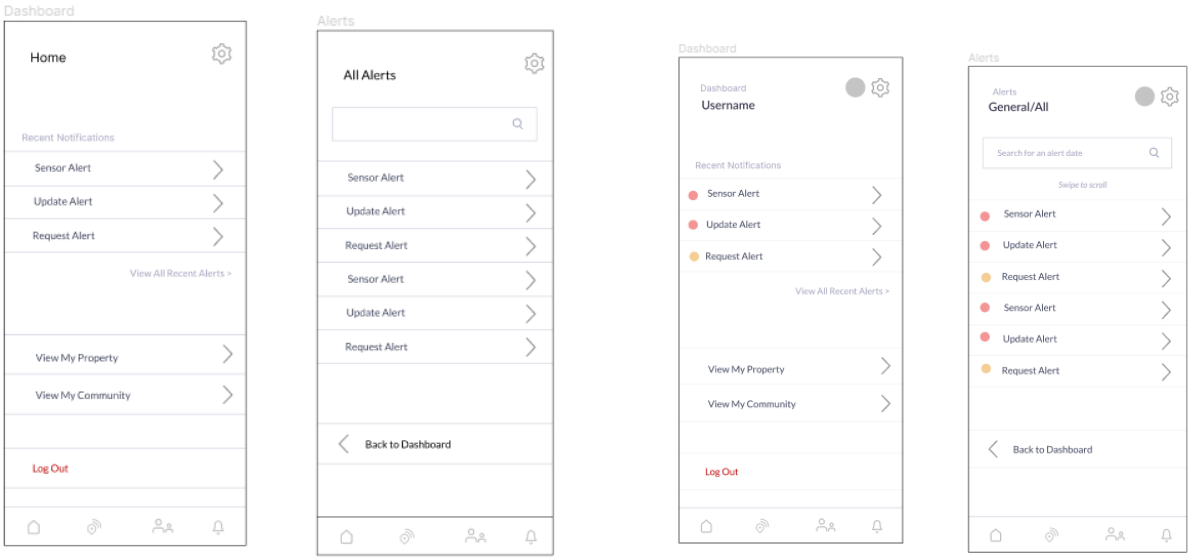
RGB: rgba(244, 206, 148 ,255)

HEX: #94f4ba

RGB: rgba(148, 244, 186 ,255)

HEX: #f0f0f0

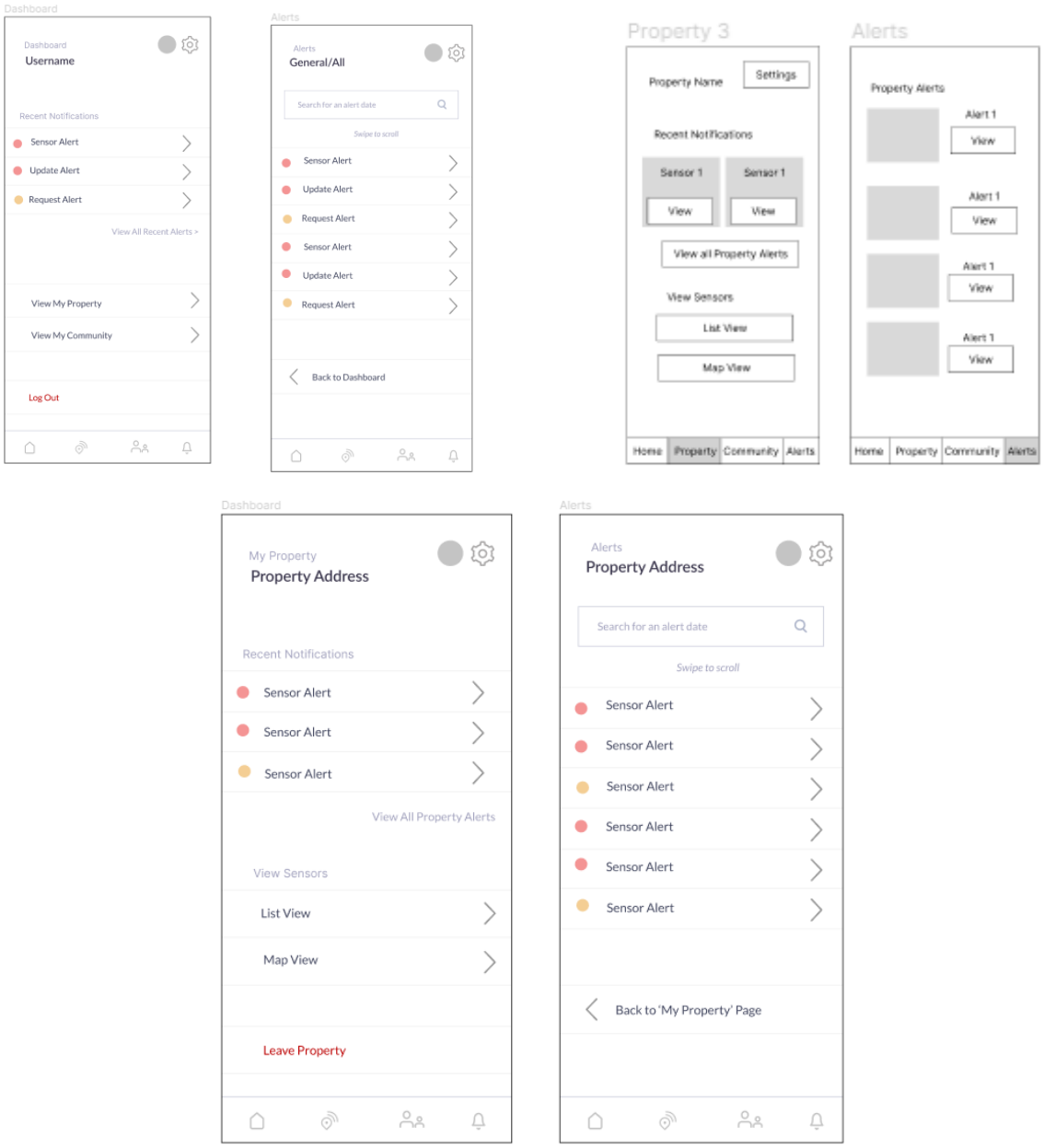
RGB: rgba(240, 240, 240 ,255)



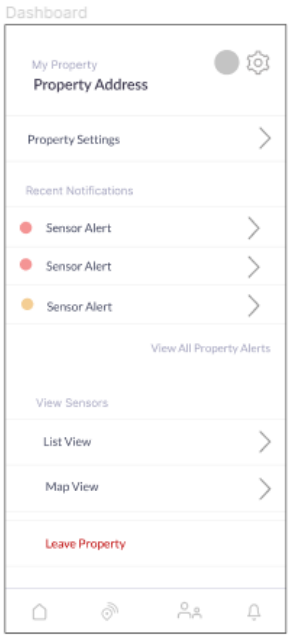
Once these core interface elements, colour guides and button styles were determined, design iterations mainly were the result of applying the style in the most user friendly way to the design patterns that were well thought out from the low fidelity prototype.

‘My Property’ Page Iteration Section

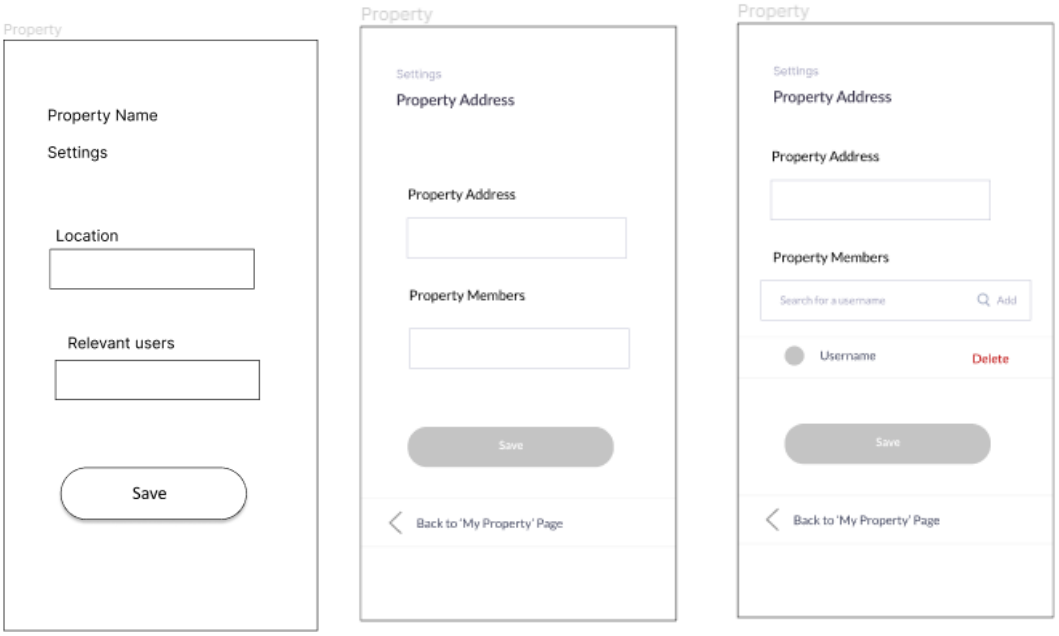
To promote consistency between the interfaces, the ‘My Property’ screen actually replicates the ‘Dashboard’ interface iterated on in the above fragment of the prototype. This combination of the low-fidelity prototype and a higher-fidelity prototype is shown below. The navigation elements to ‘My Property’ and ‘My Community’ are replaced by ‘List View’ and ‘Map View’ for the sensors.



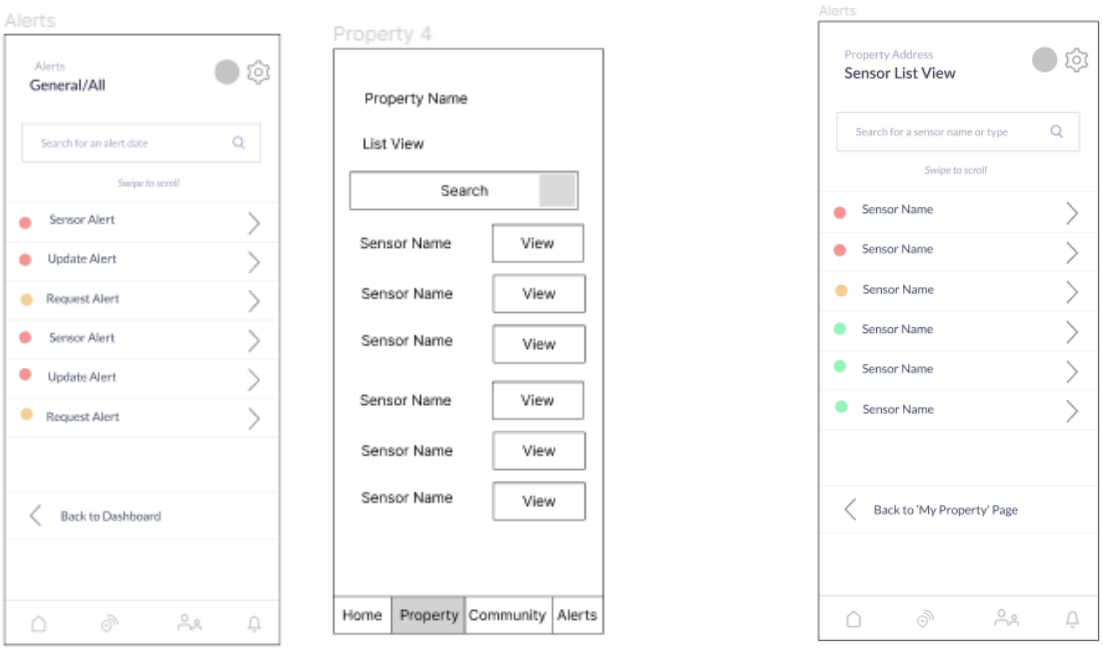
A later iteration occurred due to the realisation of a ‘property settings’ navigation bar being required. This was added as a button with the consistent style of the interface.



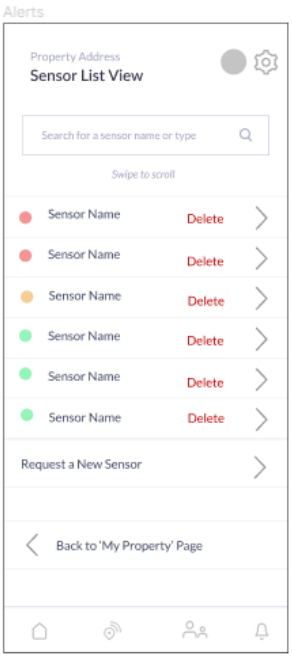
This was added, and the simple property settings page is as displayed.



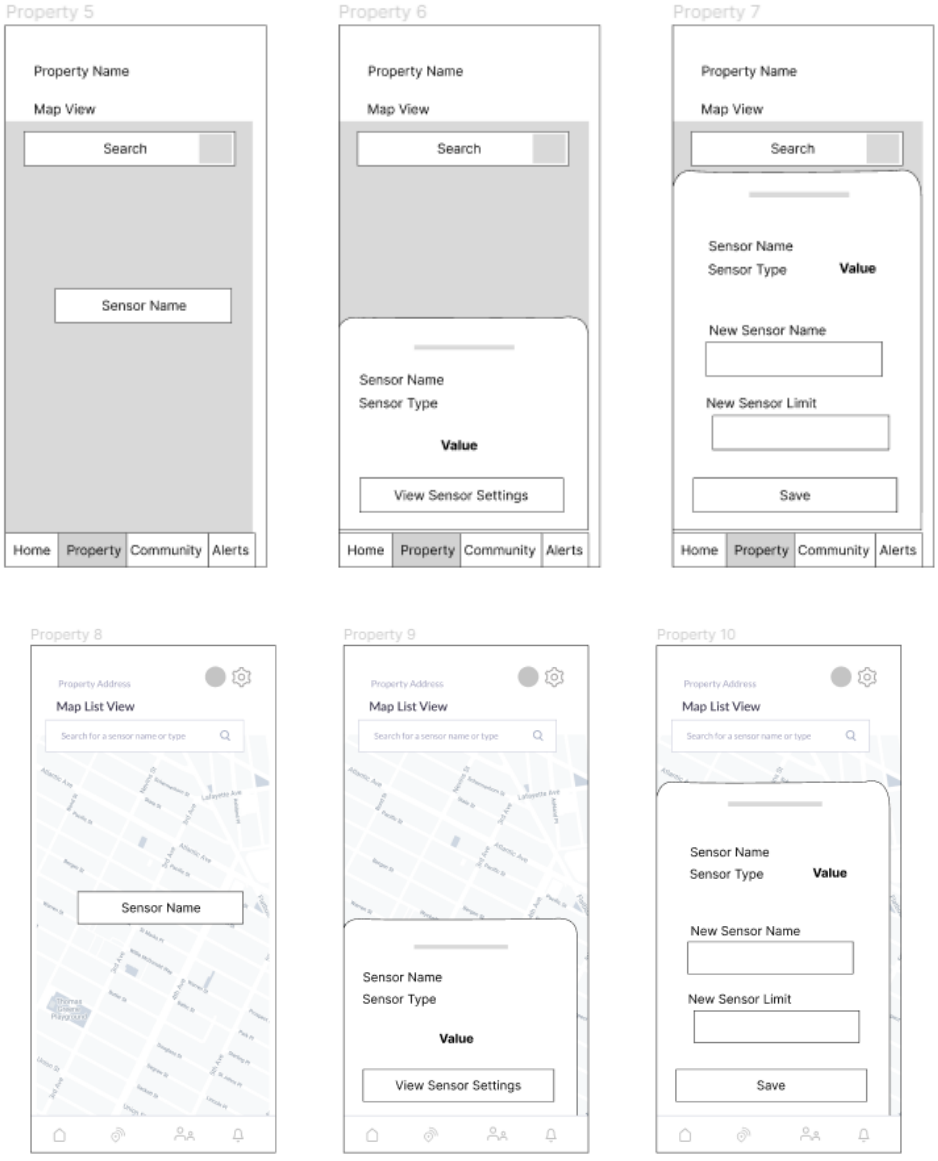
For the ‘Sensor list view’, the ‘Alerts’ layout with the ‘Search Box’ design pattern could also be duplicated. Then. the ‘List View’ displaying all of the sensors on the property can be achieved in a consistent, efficient and familiar screen.



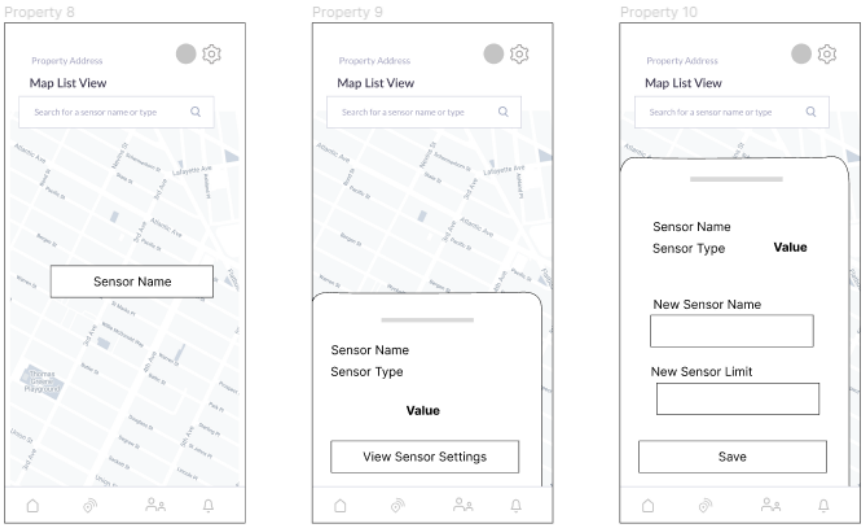
One further iteration on this design, is the seamless integration of management options for the user through the use of the conventional ‘red’ warning text, and the addition of the ‘Request a New Sensor’ function which was realised during design.

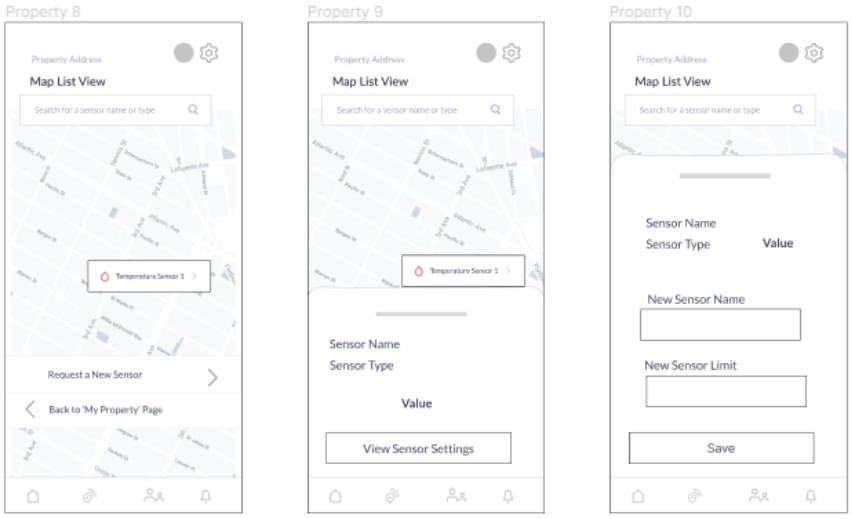


Lastly, the interface for the map property view was iterated. First, the grey box low-fidelity element was replaced with an image of the map. Other elements such as the 'Icon Menu' design pattern and the font type and size for the heading of the screen aligned with Shneiderman's first golden rule, 'Strive for consistency'. This includes also improving it with larger elements, such as replacing the search bar.

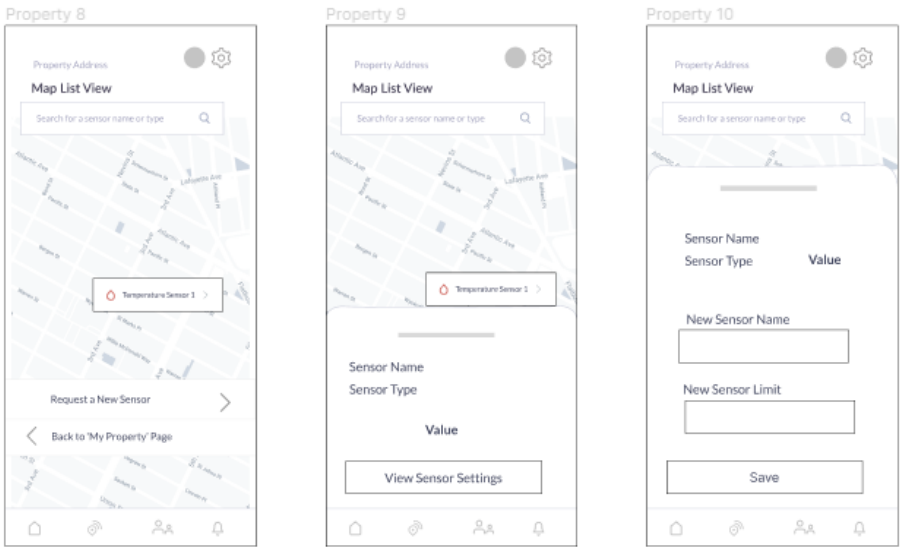


As a central component to the application, Nielsen's seventh usability heuristic 'Flexibility and Efficiency of Use' inteface is imperative. Therefore, the refinement of the 'Modal' design pattern to become a feature that allows the information beneath the overlay to remain visible is emphasised in the next iteration.

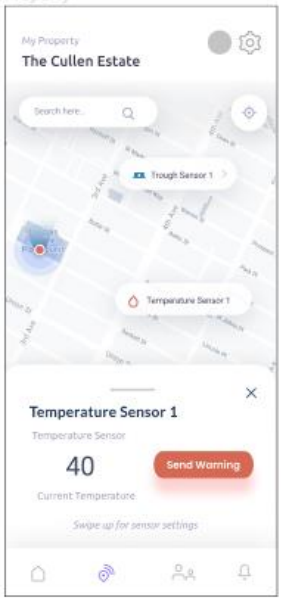




Lastly, the low-fidelity prototype is refined to a near high-fidelity state.

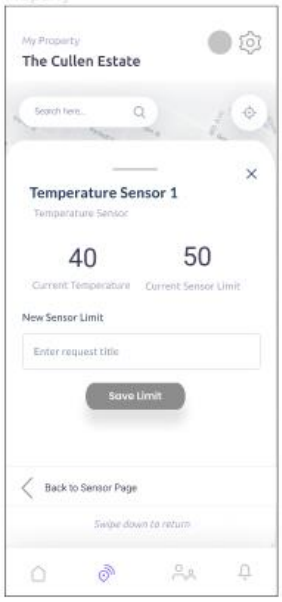


Property

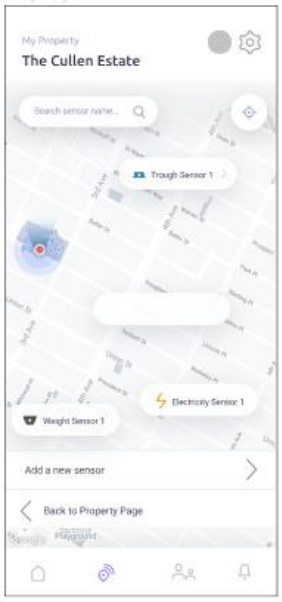


Property

Property



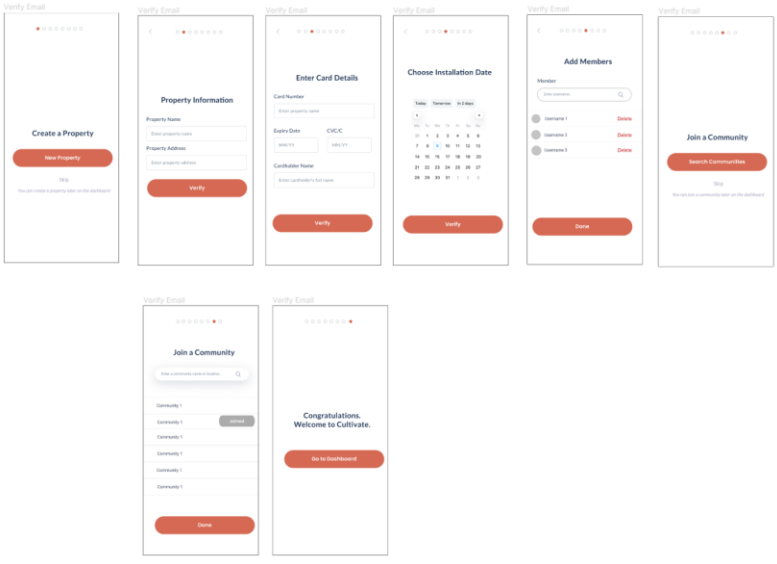
Property



‘Onboarding (‘Create a new property’) Iteration Section

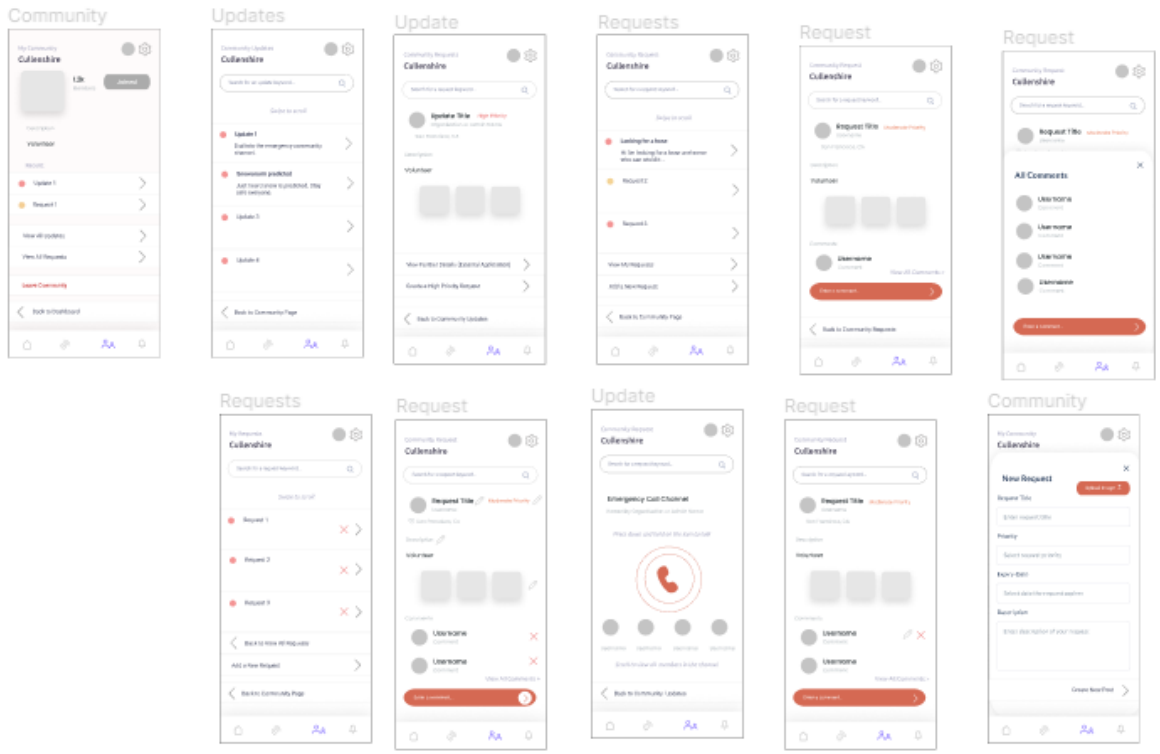
This aspect of the system was able to be designed to a near high-fidelty level quickly. The strong design theory of this wireflow is achieved through the refinement of the ‘Wizard’ design pattern to break the registration process into digestible steps, the ‘Steps Left’ design pattern to reward the user with their progress, and the ‘Sequencing’ persuasive design pattern which utilises the fact that users are more likely to proceed when they are presented with steps. The main iteration of the onboarding interfaces is the creation of consistency with the other interfaces of the application, and the use of Cultivate colours to establish brand and encourage the user to complete the onboarding process.

Not the application of the ‘Steps left’ design pattern shifted from a numerical value to visual dots. This shift to a visual representation makes the onboarding steps process appear even less overwhelming.



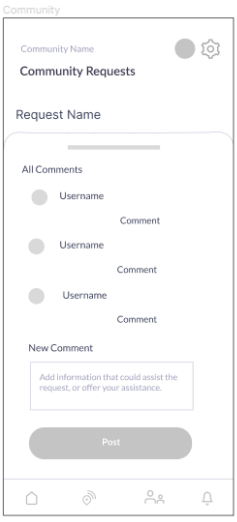
‘My Community’ Page Iteration Section

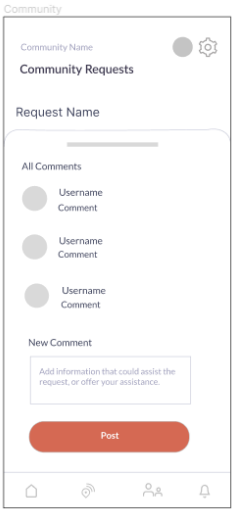
Similarly, this aspect of the system was able to be designed to a near high-fidelty level quickly. The strong design theory of this wireflow is achieved through the use of the ‘Blog post’ design pattern to allow users to make ‘requests’ effectively. This accomplishes the core functionality in a usable way. The main iteration of the ‘My Community’ interface is the creation of consistency with the other interfaces of the application, and the use of Cultivate colours to establish brand and encourage the user to complete the onboarding process. Additionally, colour and the consistency of font and use of icons brings this interface to the high fidelity stage quickly.



‘Add a comment’ Iteration Section

By applying the ‘Comment’ design pattern to Cultivate’s Community social system, users are encouraged to add a comment on the interface. The following iterations demonstrate the aesthetic applications that are applied to the screen structure that was consolidated in the Combined Low Fidelity Prototype.



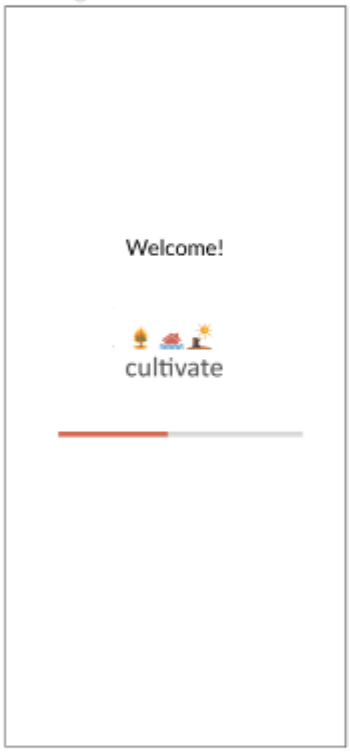


FINAL DESIGN

The following final design is the prototype of the Cultivate application presented at a high-fidelity stage. High fidelity refers to the amount of detail and quality included in the design. In this case, the high-fidelity prototype has been modified to near ‘photo quality’. This includes transition and interaction between screens to simulate the function of the Cultivate application. We have included the personas and the scenarios that they participate in, as outlined in Part A, allowing the function of our system to be visualised. The following final design has been refined from ongoing design choices and interface iterations as demonstrated in the previous section.

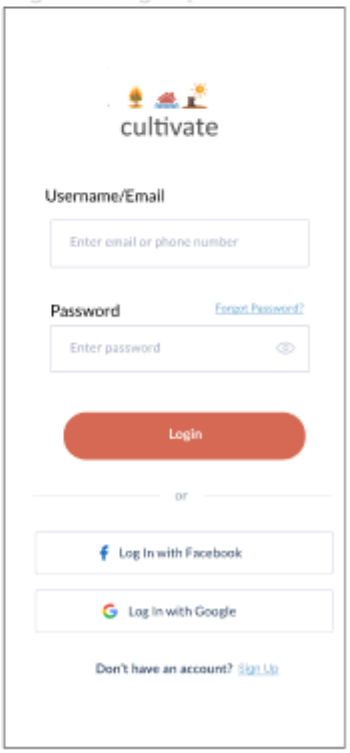
‘Login and sign up’ Final High-Fidelity Section

Loading

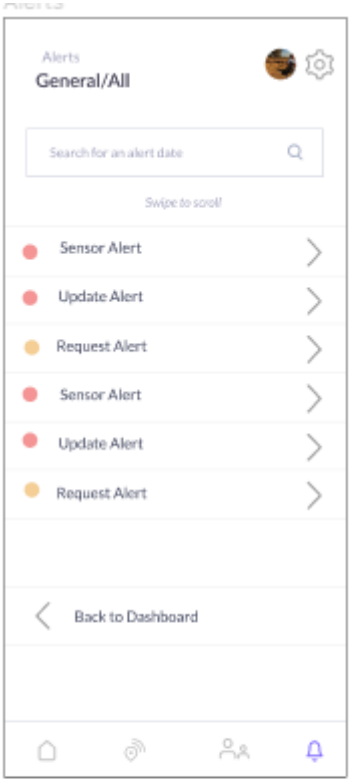


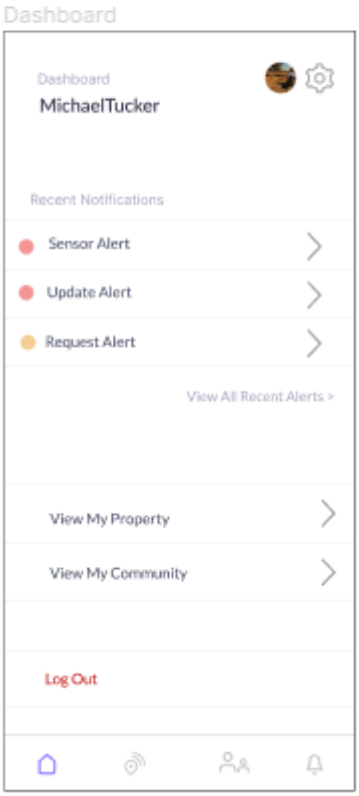
Welcome Back

Log In and Sign Up

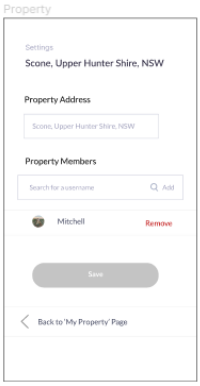


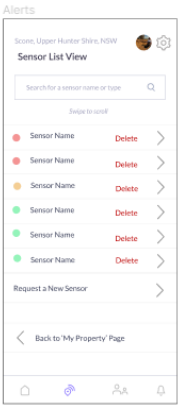
‘Dashboad’ and ‘Alerts’ Final High-Fidelity Section’

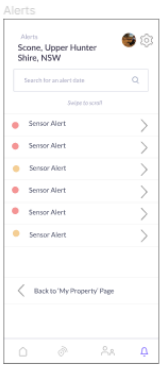


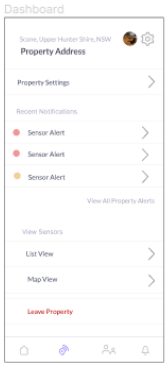


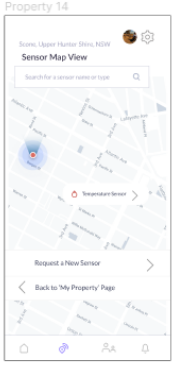
‘My Property’ Page Final High-Fidelity Section

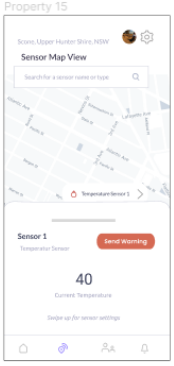


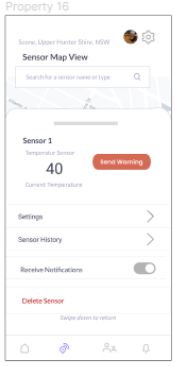


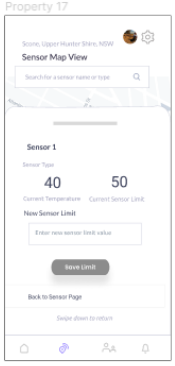


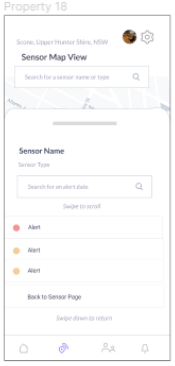




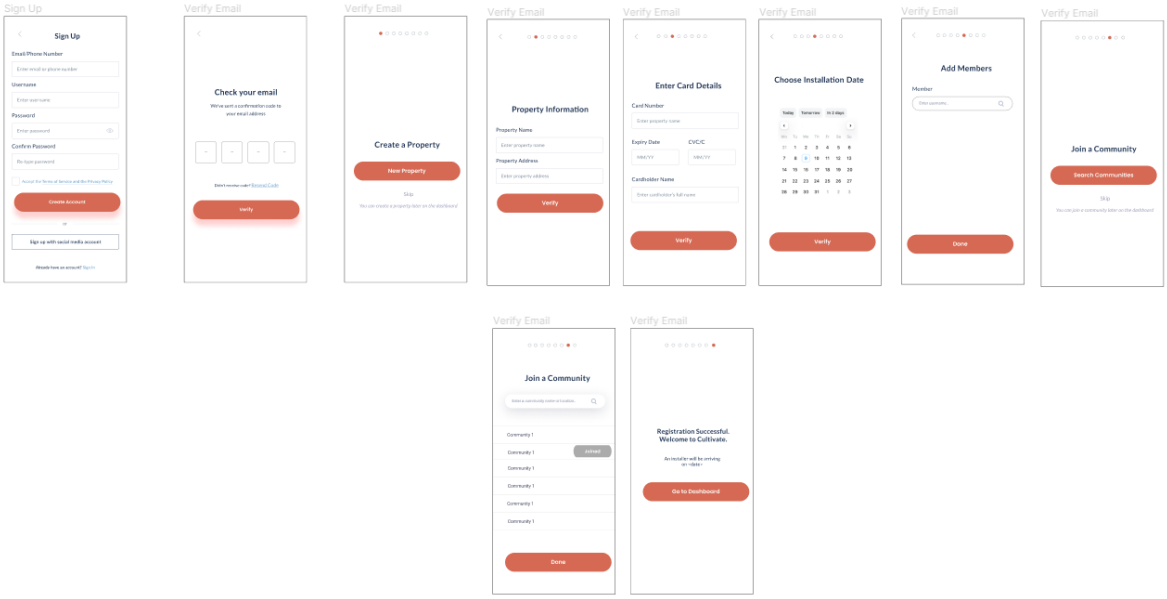




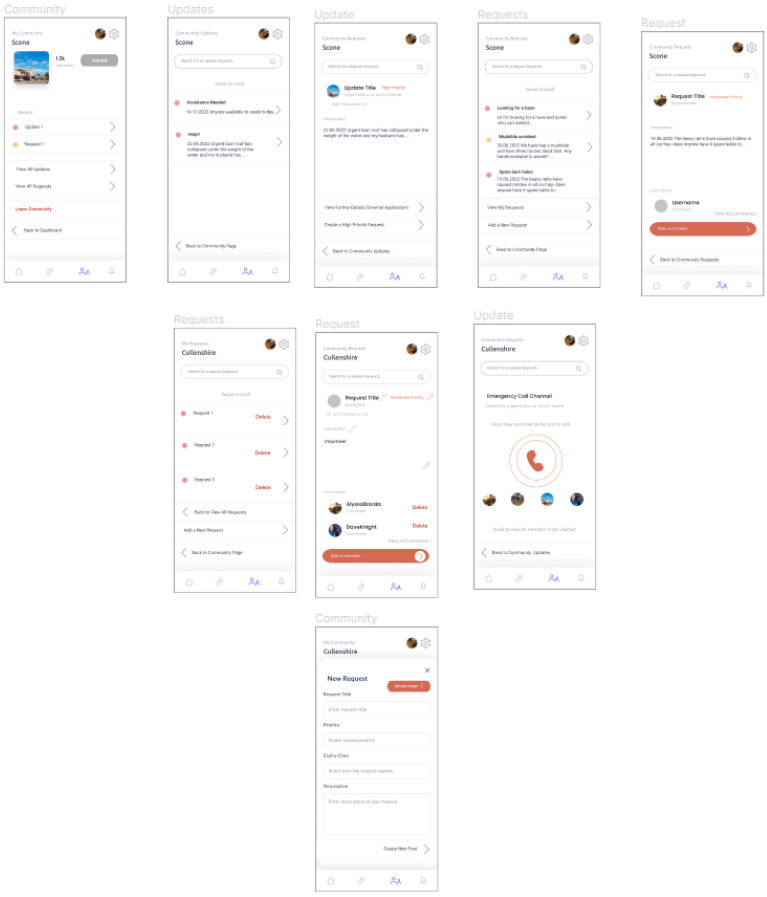




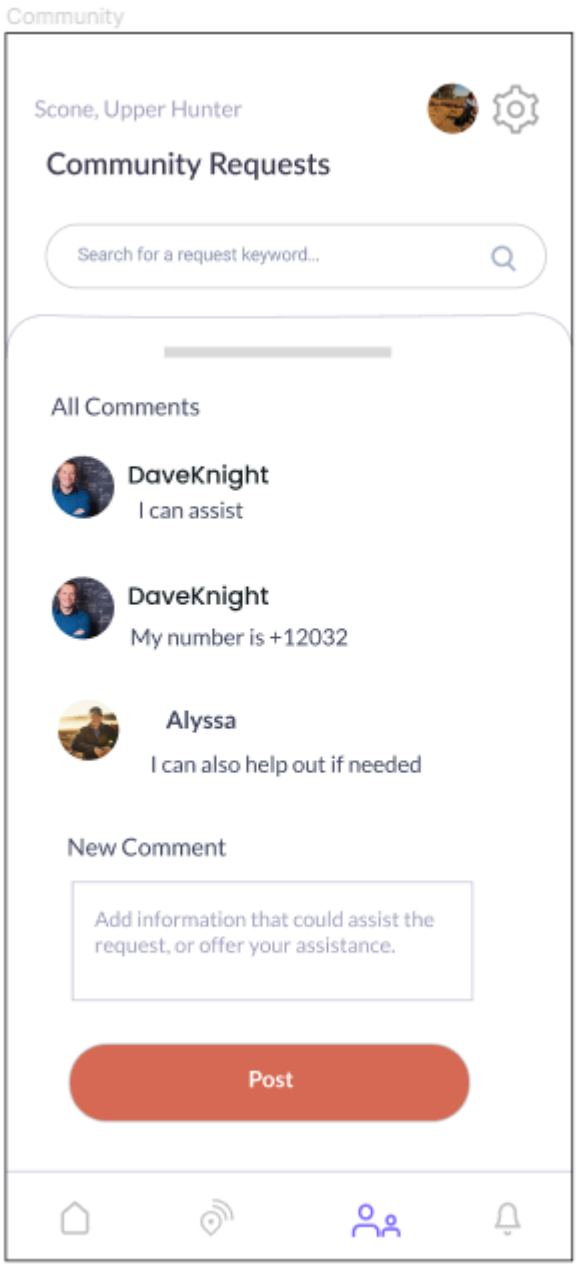
‘Onboarding (‘Create a new property’) Final High-Fidelity Section



‘My Community’ Page Final High-Fidelity Section



‘Add a Comment’ Final High-Fidelity Section



Overall review of the design:

Overall, the final design of the Cultivate application successfully integrated the core functionality requirements identified in Part A's initial requirements, garnered from our comprehensive problem domain and user research. These were incorporated in detailed and complete interfaces, supported by the consideration of design opportunities such as accessibility tools and user documentation, enhancing its overall quality and user experience. Moreover, the grounding of element selection in conceptual design patterns, and Nielsen's ten usability heuristics in conjunction with Shneiderman's eight golden rules for interface design solidifies a strong theoretic foundation for our design. However, through consideration of usability in the final prototype, the extent to which the application could assist users in real world natural disaster situations reveals areas for further improvement and opportunity.

Overall review of the design: initial requirements

As we deduced from our research in Part A of the report, there are six primary initial requirements for the system. Throughout the prototyping of our system, these six main core functionality requirements and features to support them have persisted. Instead, further requirements have been added to our secondary list of requirements, such as more detailed settings and customisability options, an effective onboarding process, and the scope of our project. Through evaluating the efficacy of the design in accomplishing each of the initial requirements, a comprehensive review of our final design is achieved.

1. The interface should enable a user to add a property and add other users as 'members' of one property

As shown, the interface provides an intuitive interface for users to manage and add a new property. Users can accomplish this during our onboarding process, whereby our interface offers a simple 3-stage-process to add a new property. This draws from the 'Sequencing' design pattern, supported by other user interface design patterns the strengthen our design. Each screen includes one step. The first step is to name the new property, then positioning the location of IoT sensors on a map of their property, and finally add members of the property which simplistically fulfills this criterion. Furthermore, features that support this requirement such as members to be able to accept a property request and members being able to search the username of a member result in the fulfillment of this criteria.

Our final prototype offers a concise single screen to allow users to create a property in an intuitive and user interface friendly manner. The prototype also allows for a way of navigation to a page which lets other uses act as a member of a property, fulfilling the initial requirement.

2. The interface should enable a user to view the dashboard for their property and navigate to the statistics of a particular IoT sensor, such as:

a. viewing the level of moisture in the ground

b. measuring the weight of livestock feed in paddock and silo stores

c. viewing the surrounding temperature

d. measuring the electricity in the generator.

Through our user interface, we have demonstrated through various diagrams that the user is able to navigate to the dashboard to view their property, and then click on any sensor connected to the property to view information of the sensor. Each page connects the user to a new feature, such as being able to view each sensor,

the history of the sensor, settings to change it etc. All of this is incorporated into a page each per function, which, although it can increase the amount of time it takes to navigate to a certain function, is beneficial as it doesn't overwhelm the user with too much information and data at the same time. This meets the initial requirement and is successful as displayed by our final prototype.

3. The interface should invoke device-generated alerts when a sensor passes a custom-set trigger point.

a. The dashboard should display alerts when a task has not been completed

Notifications are displayed on both the dashboard and the users' phone (if they have set their preferences to allow alerts in their phone). There is a set place for notifications to appear if a task hasn't been completed, or there is an emergency, or more specifically when a sensor passes a custom-set trigger point. There is both a page to view the sensor data, a page to change the trigger point and the dashboard where notifications appear. The dashboard not only displays alerts when a task has not been completed, it shows notifications custom set by the user either in regard to their property or the community functions. This successfully meets the initial requirement and is shown in our final prototype.

4. The interface should enable a user to post a request for assistance to a specific community, filling out the:

a. request name

b. request priority-type tag

c. description

d. request expiry-date

Our community page allows the user to first and foremost interact with the community by viewing posts and communities and being given the ability to like or comment on them. The user is then also able to visit the 'make a request' page from the community page, which allows the user to fill out the specific details above and post the request to the community for others to respond and view. The interface is intuitive and allows for any user to communicate with the local community via the app, thereby demonstrating that the user experience is optimal and has been made efficient through the design and through each aspect of the app. As shown by our final prototype, this requirement has been met and is a useful and fundamental feature of the software.

5. The interface should enable a community leader to 'host' a push-to-talk 'walkie-talkie' voice channel

In our community page, the user is able to navigate to a page which contains the 'voice chat' feature, called the 'Emergency Call Channel', allowing a community leader to host a push to talk voice channel. This increases user interaction and overall flows well with the functionality of the software. The community page hosts a large amount of community driven features, this emergency voice chat being one of them. The way this feature was designed was unique and although it is not normally found in other apps, it still provides an intuitive design and is a professional, well-made feature which will benefit the users in case of an emergency, meeting the requirement of a voice channel.

6. The interface should enable a user to comment on an existing post

Throughout the community page, the user is given numerous outlets to increase their interaction, namely with posts. From the community page, the user is able to navigate to the community request page to interact with posts and comment on existing posts in their feed. This is way users are able to assist other users, by commenting on requests. For example, if a user has posted a request to the community for a hose, a user is able to comment on the post and possibly communicate a resolution for the requestee. This function has been clearly outlined and adds to the interactivity of the software, creating a better user experience and a more useful community page function.

TESTING CONSIDERATIONS

Our software would need to be tested in order to ensure it meets the standards set by ourselves and our competitors. Testing the software would also benefit it greatly by eliminating possible common errors and unseen issues during the development process. Since the software deals with critical matters and often very important aspects of properties (notifying users of possible environmental disasters, etc), it is crucial that we conduct a thorough and detailed testing process, as it increases the quality of our product, while ensuring it works as needed. Despite the testing process being time consuming, due to the nature and use of our software, it is imperative this process is as comprehensive as possible before being released to the public for beta access.

To test our software, we would establish a usability lab in a controlled space, selecting five representative users to participate in different test cases which would be relevant to the everyday use of the software. We would first have to outline the end goals for the outcome (for example, the user should be able to make a request, edit property, view individual sensors etc) and determine the relevant users. Then the test environment should be created (for example, a user is placed inside the lab away from outside influence) and given access to a phone with the app installed. Once the tasks for the users have been developed, all users will be recorded performing the same test cases inside the controlled space. Once test cases are satisfied and enough user data is captured, each test case will be used to evaluate and analyse the data (for example making sure a user can receive an emergency notification properly). These test case results will then be used to correct all data and functions and is beneficial to the app since the software is fundamentally vital for users, which is the reason that testing is so important.

We will also conduct field studies which would essentially remove test cases and instead allow users to freely use and navigate the app, then evaluating those results. Allowing users to experiment with the app in an uncontrolled environment is also beneficial as it could help uncover issues, such as a difficult user interface or if users are unsure how to make a request. Using these results, we would be able to improve upon the app and change aspects which were unseen otherwise. By collecting logs and recording user input when they interact with the app, we could collect feedback on the software and then use that to improve upon the app. For example, if a user found the community page difficult to navigate or hard to find posts, we could use that data to alter that function and improve it. Conducting field studies would assist us further and create a better user interface/ experience when released to the public.

The use of predictive models could also be beneficial for us in the future, however it should be used in conjunction with other testing as although it is more cost effective than other ways of collecting user data and working with the results, since it is based on predicting common and usual functions (such as a user navigating to a different page from the dashboard, or a user requesting a post). While this doesn't directly involve users, predictive models are useful as they can conduct evaluations to determine the design and functionality of the software, which would assist us in the later stages of development. By using the keystroke level model (evaluating how long it takes to click on a button), we can determine how long it takes for users to click certain elements or navigate back home from a certain page. We can use this data and predictive models to define and improve the user interface and overall improve the software.

Through the use of evaluative models, we also plan to utilize these in order to develop our app and improve upon it via feedback. Methods such as the expert inspection, where we hire experts to review the software in terms of its user interface and overall general usability. This can be incredibly beneficial as gaining expert feedback can highlight certain criteria and aspects which may be overlooked during the development process. Although this means often using predetermined scenarios and test cases for the user interface, an expert view should explore certain aspects and highlight key areas for improvement. Using this evaluative model, combined with field studies, controlled lab studies and predictive models, the app should be explored in depth via expert and general users alike, improving and solidifying the user interface and general functionality of the app.

EVALUATION ASSESSMENT: *THEORETICAL FRAMEWORKS*

The use of two prominent evaluation theories in Human Computer Interaction (HCI) theory offers a deep review of the design. Firstly, comparison against the revised version of Jakob Nielsen's 10 usability characteristics (APPENDIX B: Jakob Nielsen's 10 Usability Characteristics) focuses on usability from the perspective of the user, based on heuristics distilled from an empirical analysis of 249 usability problems. This also includes evaluation aspects such as user documentation, encompassing the entire user experience. Next, an assessment is performed against Ben Shneiderman's 8 golden rules of interface design (APPENDIX C: Ben Shneidetman's 8 Golden Rules of Interface Design).

Revised Version (2014) of Jakob Nielsen's 10 Usability Heuristics (1990s)

Heuristic	Not evident	Neutral	Achieved	Strongly Achieved
1. Visibility of system status				
2. Match between system and the real world				
3. User control and freedom				
4. Consistency and standards				
5. Error prevention				
6. Recognition rather than recall				
7. Flexibility and efficiency of use				
8. Aesthetic and minimalist design:				
9. Help users recognize, diagnose, and recover from errors				
10. Help and documentation				

APPENDIX B: Jakob Nielsen's 10 Usability Heuristics provides a detailed explanation of each of the heuristics.

Shneiderman's 8 Golden Rules of Interface Design

Heuristic	Not evident	Neutral	Achieved	Strongly Achieved
1. Strive for consistency:				
2. Enable frequent users to use shortcuts				
3. Offer informative feedback:				
4. Design dialogue to yield closure:				
5. Offer simple error handling:				
6. Permit easy reversal of actions:				
7. Support internal locus of control:				
8. Reduce short term memory load:.				

APPENDIX C: Ben Shneiderman's 8 Golden Rules of Interface Design provides a detailed explanation of each of the rules.

it is ultimately deduced that with further improvements and iterations in the user-centred design process, as well as the conduction of usability tests with evaluators, the Cultivate interface could greatly assist in progress toward addressing

CONCLUSION

Therefore, through detailed and thorough investigation into the climate change problem domain, it is clear that Cultivate front-end application captures a greater demand for a more innovative, sustainable, and future-forward software that enables communities all over the globe to congregate further, as well as allowing those with a greater demand for IOT software to benefit their farms with a wide range of capabilities which support critical UN goals.

Cultivate has produced an extensive report which outlines the necessity of introducing a more intuitive IOT system for farmers which assists them with creating a more efficient environment for the farm. The report also delves into detail with the community aspects of the system. Through connecting users in local communities with well-researched and professional user experience practices, it benefits all parties and is a practical approach to encouraging users to purchase/sell/ interact with others. Users who have expressed interest in our system have provided us with details on their current IOT experience and have communicated their interest to participate in the services listed throughout our report. It is clear that Cultivate has successfully reinvigorated the use of IOT systems in agricultural production whilst also connecting local communities and engaging its members.

Therefore, through the rigiorous iterative design process, our system successfully achieves the initial core functionality requirements that we outlined will benefit any farm and create a more efficient IOT system. At the conclusion of evaluating our high-fidelity prototype, it is ultimately deduced that with further improvements and iterations in the user-centred design process, as well as the conduction of usability tests with evaluators, the Cultivate interface could greatly assist in progress toward addressing United Nation Sustainable Development Goal 13: Climate action, Goal 9: Industry, innovation and infrastructure, Goal 15: Life on Land research and Goal 17: Partnerships for the goals.

REFERENCES

Ahani, H 2022, Iconsax 6000+ icons, Figma, Viewed 2 October 2022, <https://www.figma.com/community/plugin/1157650182274906132>

allMETEO 2018, allMETEO, viewed 3 September 2022 <https://allmeteo.com/>

Arriagada, N, Palmer, A, Bowman, D, Morgan, G, Jalaludin, B & Johnston, F 2020 *Unprecedented smoke-related health burden associated with the 2019–20 bushfires in eastern Australia,* Medical Journal of Australia, viewed 31 August 2022, <https://www.mja.com.au/journal/2020/213/6/unprecedented-smoke-related-health-burden-associated-2019-20-bushfires-eastern>

Australian Bureau of Statistics 2020, *Measuring natural disasters in the Australian economy,* Australian Bureau of Statistics, viewed 1 September 2022, <https://www.abs.gov.au/statistics/research/measuring-natural-disasters-australian-economy>

Australian Institute for Disaster Resilience n.d., *New South Wales July 2019 - March 2020 Bushfires - Black Summer,* Australian Institute for Disaster Resilience, viewed 31 August 2022, <https://knowledge.aidr.org.au/resources/black-summer-bushfires-nsw-2019-20/>

Australian Institute for Disaster Resilience n.d., *New South Wales, January 2012 Flood - New South Wales 2012*, Australian Institute for Disaster Resilience, viewed 28 August 2022, <https://knowledge.aidr.org.au/resources/flood-new-south-wales-2012/>

Australian Institute for Disaster Resilience n.d., *Victoria November 2019 - February 2020 Bushfires - Black Summer,* Australian Institute for Disaster Resilience, viewed 31 August 2022, <https://knowledge.aidr.org.au/resources/black-summer-bushfires-vic-2019-20/>

Bible Vector Illustration n.d., *Burning bush not consuming by flames as narrative,* 123rf, viewed 28 August 2022, <https://www.123rf.com/photo_165067688_burning-bush-not-consuming-by-flames-as-narrative-from-bible-vector-illustration.html?vti=nlipje8czdhvooyk3r-1-4 >

Costa, R 2020, *Toggle switch design: the full run through,* Justinmind, viewed 6 October 2022, <https://www.justinmind.com/ui-design/toggle-patterns-examples>

Cultivate Farms 2018, *You Can Own A Farm,* Cultivate Farms, viewed 9 September 2022, <https://www.cultivatefarms.com/>

Facebook 2022, Facebook, viewed 3 September 2022 <https://www.facebook.com/>

Fiverr 2022, *Find the perfect freelance services for your business*, Fiverr, viewed 3 September 2022 <https://www.fiverr.com/>

Freepik n.d. *Drought Free Icon*, Flaticon, viewed 1 September 2022, <https://www.flaticon.com/free-icon/drought_491190>

Freund, M, Henley, B, Allen, K &Baker, P 2018, *Recent Australian droughts may be the worst in 800 years,* The Conversation Media Group, viewed 1 September 2022, <https://theconversation.com/recent-australian-droughts-may-be-the-worst-in-800-years-94292>

Gissing, A, Timms, M, Browning, S, Coates, L, Crompton, R & McAneney, J 2020, *Compound Natural Disasters in Australia: A Historical Analysis,* Bushfire and Natural Hazards Cooperative Research Centre, viewed 28 August 2022, <https://naturaldisaster.royalcommission.gov.au/system/files/2020-09/RCN.900.122.0001.pdf>

Hughes, N & Gooday, P 2021, *Climate change impacts and adaptation on Australian farms,* Australian Bureau of Agricultural and Resource Economics and Sciences, viewed 28 August 2022, <https://www.agriculture.gov.au/abares/products/insights/climate-change-impacts-and-adaptation#recent-changes-in-seasonal-conditions-have-affected-the-profitability-of-australian-farms>

Hughes, N, Galeano, D & Hatfield-Dodds, S 2019, *Analysis of the effects of drought and climate variability on Australian farms*, Australian Bureau of Agricultural and Resource Economics and Sciences, viewed 6 September 2022, <https://www.agriculture.gov.au/sites/default/files/documents/EffectsOfDroughtAndClimateVariabilityOnAustralianFarms_v1.0.0.pdf >

Iconduck 2022, *Microsoft Fluent UI Icons,* Figma, viewed 6 October 2022, <https://www.figma.com/community/plugin/1149932624908995432>

ImageIX 2022., *Boy with cows,* Zebrafish Labs, viewed 1 September 2022, <https://videohive.img.customer.envatousercontent.com/files/e58f19b4-2fbf-40c1-981c-5def2aa87fe7/inline_image_preview.jpg?auto=compress%2Cformat&fit=crop&crop=top&max-h=8000&max-w=590&s=55539cb442575227aac84953ce5ebe7a>

Insurance Council of Australia 2022, *Updated data shows 2022 flood was Australia's costliest*, Insurance Council of Australia, viewed 26 August 2022, <https://insurancecouncil.com.au/resource/updated-data-shows-2022-flood-was-australias-costliest/>

International Organisation for Standardisation 2018, *Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts,* ISO, viewed 6 October 2022, <https://www.iso.org/obp/ui/#iso:std:iso:9241:-11:ed-2:v1:en>

Lindsey, R & Dahlman, L, 2022, *Climate Change: Global Temperature,* NOAA Climate, viewed 26 August 2022, <https://www.climate.gov/news-features/understanding-climate/climate-change-global-temperature>

Mosley, L 2020, *Assisting to manage soils and landscapes post-fire,* Soil Science Australia, viewed 26 August

2022,<https://www.soilscienceaustralia.org.au/about/2020-fire-and-soil/>

New South Wales Government 2022, *Key Facts About Drought,* NSW Government, viewed 26 August 2022, <https://www.nsw.gov.au/drought-in-nsw/key-facts-about-drought>

New South Wales Rural Fire Service 2022, NSW Government, viewed 3 September 2022, <https://www.rfs.nsw.gov.au/fire-information/fires-near-me>

Nguyen, H, Wheeler, M, Hendon, H, Lim, E & Otkinb, J 2021, 'Weather and Climate Extremes', *Science Direct,* viewed 28 August 2022, <https://www.sciencedirect.com/science/article/pii/S2212094721000190#:~:text=5.-,Conclusion,objectively%20identified%20as%20flash%20drought.>

Nielsen, J 2020, *10 Usability Heuristics for User Interface Design,* Nielsen Norman Group, viewed 6 October 2022, https://www.nngroup.com/articles/ten-usability-heuristics/>

Panther Media GmbH 2013, *Laugh laughs laughing,* Alamy, viewed 28 August 2022, <https://www.alamy.com/stock-<photo-laugh-laughs-laughing-115204385.html?imageid=051DA0BB-3CEA-45DF-8918-6649B0D27CA0&p=335826&pn=1&searchId=8655d62b9f68380519484d19caa9ff54&searchtype=0>

Pixdeluxe 2017, *Living with Nature Right on my Doorstep,* iStock, viewed 28 August 2022, <https://media.istockphoto.com/photos/living-with-nature-right-on-my-doorstep-picture-id832988308?k=20&m=832988308&s=612x612&w=0&h=WK52X7hWLAHcXgILzLsFgHaf2w_N3MaWt1XbbQtl-hA=>

Pycno 2021, Pycno, viewed 3 September 2022 <https://pycno.co/>

Satzinger, J, Jackson, R & Burd, S 2016, *'Systems Analysis And Design In A Changing World'*, 7th Edition, Cengage Learning, USA, viewed 5 September 2022

Statista Research Department 2022, *Value of insurance losses from major weather events in Australia 2020,* Statista, viewed 28 August 2022, <https://www.statista.com/statistics/1098529/australia-insurance-losses-from-natural-disasters/>

Toxboe, A 2022, *Design patterns,* Learning Loop ApS, viewed 3 October 2022, <https://ui-patterns.com/patterns>

Tsegai, D, Medel, M, Augenstein, P & Huan, Z 2022, *Drought in Numbers 2022,* United Nations, viewed 31 August 2022, <https://www.unccd.int/sites/default/files/2022-05/Drought%20in%20Numbers.pdf>

Turnbull, T 2022, *Australia floods: 50,000 on evacuation alert after deluge hits Sydney,* BBC, viewed 28 August 2022, <https://www.bbc.com/news/world-australia-61991112>

United Nations 2022, *Take urgent action to combat climate change and its impacts,* United Nations, viewed 27 August 2022, < https://sdgs.un.org/goals/goal13>

Uses-flood – Icon, Nicepng, viewed 1 September 2022, <https://www.nicepng.com/png/detail/61-610363_uses-flood-icon.png >

van Welie, M 2008, *Patterns in Interaction Design*, Martijn van Welie, viewed 3 October 2022, <http://www.welie.com/patterns/index.php>

Wong, E 2020, *Shneiderman's Eight Golden Rules Will Help You Design Better Interfaces,* Interaction Design Foundation, viewed 2 October 2022, <https://www.interaction-design.org/literature/article/shneiderman-s-eight-golden-rules-will-help-you-design-better-interfaces>

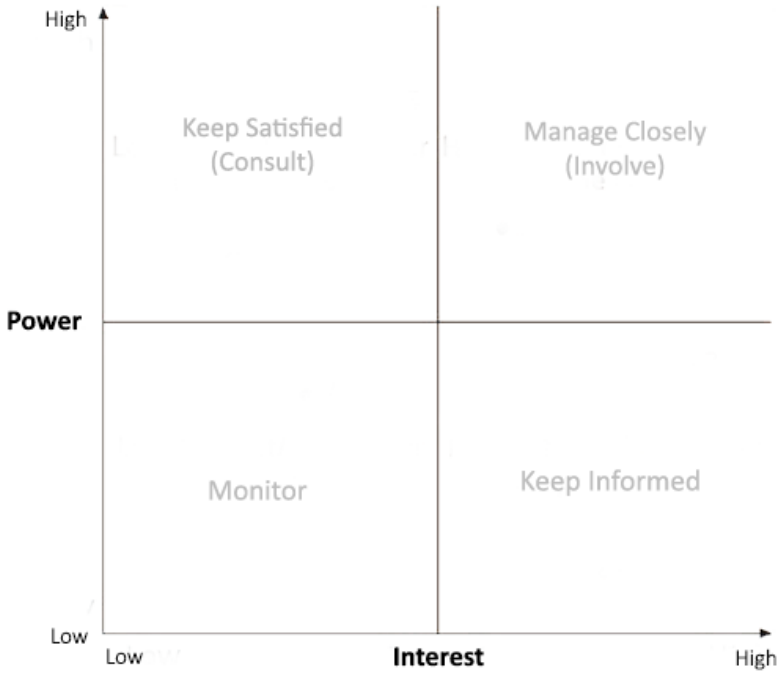
APPENDICES

The appendices contain lengthy materials that may compromise readability, or are unsuitable for the body of the report, but are helpful to understand the system proposal. Our figures involve diagrams, charts and other supporting documents.

APPENDIX A

POWER-INTEREST GRID

A matrix used to assess stakeholders based on their power or influence and interest in a project, established by Colin Eden and Fran Ackermann in their novel *Making Strategy*. This grid provides insight into management techniques for stakeholder communication and satisfaction.



APPENDIX B

STYLE GUIDE

Logo

The Cultivate logo is available as a primary logo, as well as a horizontal variation and a version with the icons alone. The logo will be used strategically, placed above headings and branding appropriately.

Refer to the logo subsection beneath image rules for the logo image rules.

Primary Colour Palette

The use of contrasting warm and cool colours symbolise the elements of water (relevant to drought and flood) and flame (relevant to property fire). As complementary colours, this enhances the aesthetic effect of our application whilst establishing a distinct interface.

HEX: #68a2ce

RGB: rgba(99,159,204,255)

HEX: #d56953

RGB: rgba(213,105,83,255)

Secondary Colour Palette

The use of natural Earthy colours as the secondary colour palette reinforces the sense of comfort our style aims to evoke.

HEX: #577e38

RGB: rgba(87,126,56,255)

HEX: #f1a52e

RGB: rgba(241,165,46,255)

HEX: #b1b6b9

RGB: rgba(177,182,185,255)

HEX: #494a49

RGB: rgba(73,74,73,255)

Interface Element Colour Palette

The use of modern colours as the secondary colour palette reinforces the sense of comfort our style aims to evoke in our application.

HEX: #c7cad9

RGB: rgba(199,202,217,255)

HEX: #a2a2be

RGB: rgba(162,162, 190 ,255)

HEX: #3f445d

RGB: rgba(63, 68, 93,255)

HEX: #bd0000

RGB: rgba(189, 0, 0,255)

Notification Colour Palette

The use of red, yellow and green draw from real world meaning, such as traffic light systems, to alert the user.

HEX: #f49494

RGB: rgba(244,148, 148 ,255)

HEX: #f4ce94

RGB: rgba(244, 206, 148 ,255)

HEX: #94f4ba

RGB: rgba(148, 244, 186 ,255)

HEX: #f0f0f0

RGB: rgba(240, 240, 240 ,255)

Fonts

We have used Lato fonts for headings and body text to achieve a modern, crisp, and clean look, creating an easy to read, clean and 'approachable' brand that evokes a sense of security, suiting the required comfort to soften the serious nature of our application. Additionally, paragraphs should have the justified alignment. This distributes text evenly between margins, achieving a professional look

Heading 1 (Lato, 20, bold)

heading 2: subsection (Lato, 14)

Subheading (Lato, 14, Regular, Grey)

Body text (Lato, 14)

Image Rules (Do's and Don'ts)

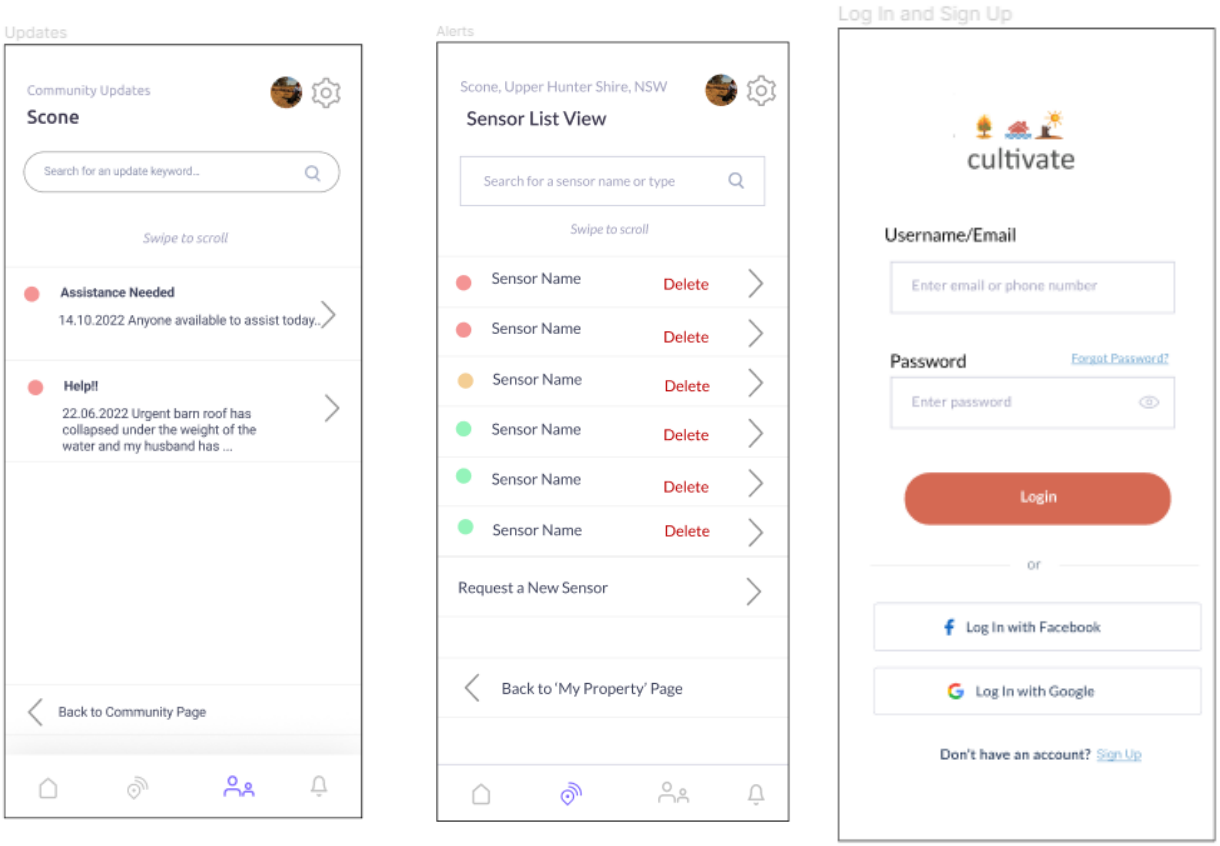
Do's:

- Images should be used judiciously
- Images should be spaced an appropriate distance apart on interfaces unless they are used in a gallery view.
- Images can be uploaded by users, and thus the application should allow images to be cropped to ensure that the image is not stretched or low resolution
- Images should be of themes relevant to the content of the application screen
- Logo can be placed on a transparent background.
- 'Cultivate' logo text can be changed to a different colour if it is placed on a dark-coloured background, but the logo symbols should not be changed.
-

Don'ts:

- Images should not appear with a stretched or blurred resolution
- Images shout not clutter the interface
- Images that are considered an icon should not be at an unreasonably large size (greater than 64 pixels).
- Logo should not appear with a stretched or blurred resolution.
- Logo icon order should not be changed.

Page Layouts



APPENDIX B

JAKOB NIELSEN'S 10 USABILITY HEURISTICS

1. **Visibility of system status (system status flags):** The system should keep users informed about what is happening with the system.
2. **Match between system and the real world:** Ensure the representations you use follow real-world conventions, offering familiarity and logic to the user.
3. **User control and freedom:** Allow users to undo and redo their actions, offering greater freedom and exploration without risk.
4. **Consistency and standards:** Ensure interface elements behave consistently.
5. **Error prevention:** Anticipate and plan for errors by eliminating them such as presenting users with a confirmation option, or recovering from them quickly
6. **Recognition rather than recall:** Minimise the user's memory load by making objects, actions, and options visible. Instructions for use of the system should be visible or easily retrievable where appropriate
7. **Flexibility and efficiency of use:** Quality of features that allow experienced users to perform regular interactions effectively, as well as customize processes to improve efficiency for themselves.
8. **Aesthetic and minimalist design:** Interfaces should not contain information which is irrelevant or rarely needed.
9. **Help users recognize, diagnose and recover from errors:** Error messages should be expressed in plain language, precisely indicate the problem, and constructively suggest a solution
10. **Help and documentation:** It is often useful to provide help and documentation to users. Any such information should be easy to search, focused on the user's perspective, list concrete steps to be carried out and present information concisely

APPENDIX C

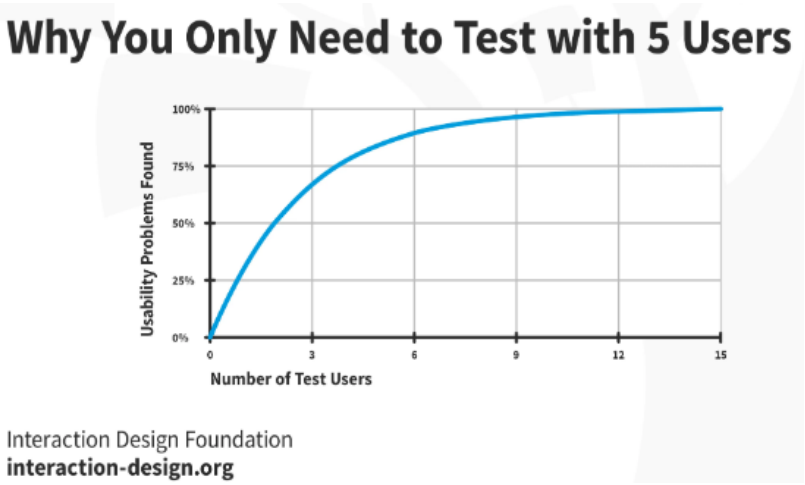
BEN SHNEIDERMAN'S 8 GOLDEN RULES OF INTERFACE DESIGN

1. **Strive for consistency:** Consistency should be achieved through a similar sequence of actions in similar situations; similar terminology in application information, consistent colouring, layout, and fonts
2. **Enable frequent users to use shortcuts**
3. **Offer informative feedback:** For every interface interaction, there should be application feedback at an appropriate level of detail.
4. **Design dialogue to yield closure:** Sequences of actions should be organized into groups with a beginning, middle, and end. Informative feedback at the completion of a group of actions gives users the satisfaction of accomplishment.
5. **Offer simple error handling:** An interface should be designed so that users cannot make serious errors; for example, graying out menu options that cannot be selected and ensuring only numerical values are entered into number-based information fields.. If users make an error, the interface should offer simple and specific instructions for recovery.
6. **Permit easy reversal of actions:** Allowing actions to be reversible reduces anxiety and encourages exploration of unfamiliar options.
7. **Support internal locus of control:** Locus control refers to an individuals belief that they have control over events in their life.
8. **Reduce short term memory load:** Avoid interfaces where users must remember information from one display and then use that information on another display.

APPENDIX D

JAKOB NIELSEN'S FIVE EVALUATORS RULE

Jakob Nielsen's research indicates that five evaluators will assist in discovering around 75% of the usability issues in an application. Beyond five users, with every additional evaluator, the proportion of new usability issues will be much smaller and usually will not compensate for the additional resources.



cultivate

The interface between nature and community