Think Make Learn Design Report

Jack Murley, Keshav Singh, Josh Hurley, Cédric Quenette

SIT216 – User Centred Design Assignment 3

Contents

Table of Figures

1.0 Introduction

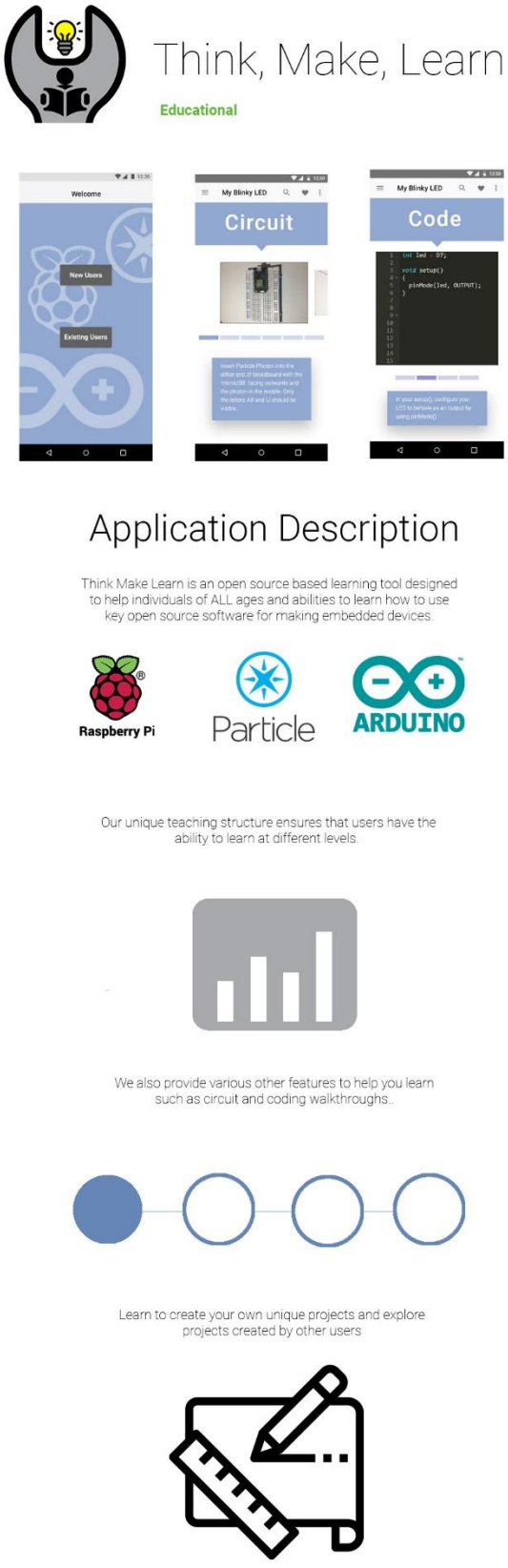


Figure 1 - Sample PlayStore Page

2.0 User Manual and Hierarchical Task Analysis (HTA)

2.1 HTA

Login

The way a user logs in is just like any other application that stores the users account as they require a valid account with a username and password in order to access the website. Furthermore, if they don't have a valid account they can then also register and instantly login off of that account and have that account automatically sign in the future for them unless turned off.

However also within our login, the users are able to login with either their Facebook or google account if they wish not to have to register through the website, this is to provide extra flexibility as well as to allow the users to be able to feel like the application is more family friendly and in the loop.

This not only ensures less work for the user and a higher level of verification but also for the backend system as it reduces the workload as it does not have to receive another and verify account message

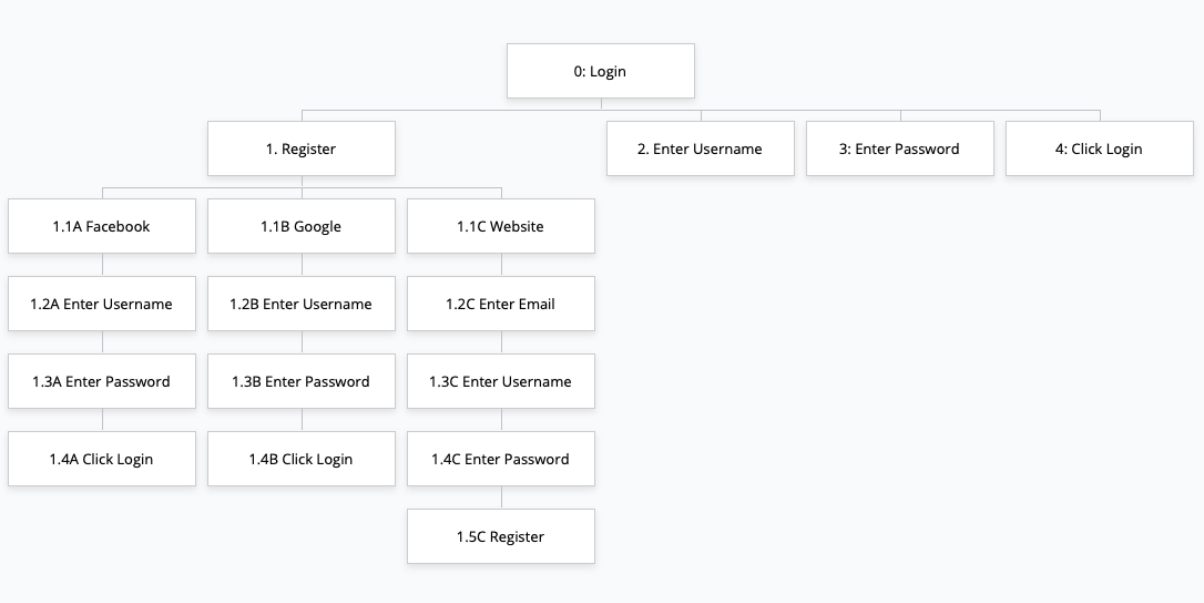


Figure 2 - HTA - Login

Search - Basic Arduino

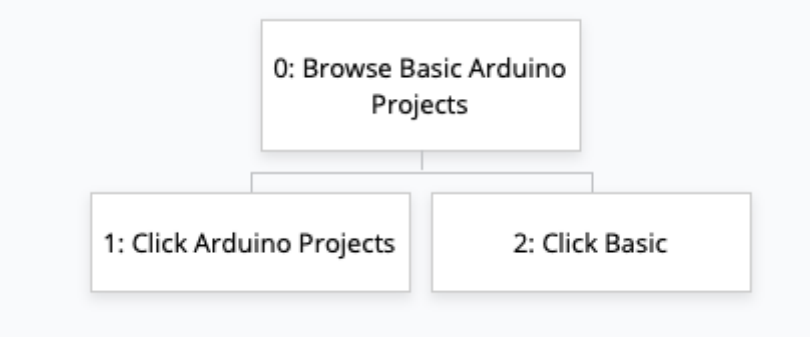


Figure 3 - HTA - Search Basic Arduino

Search - Intermediate Arduino

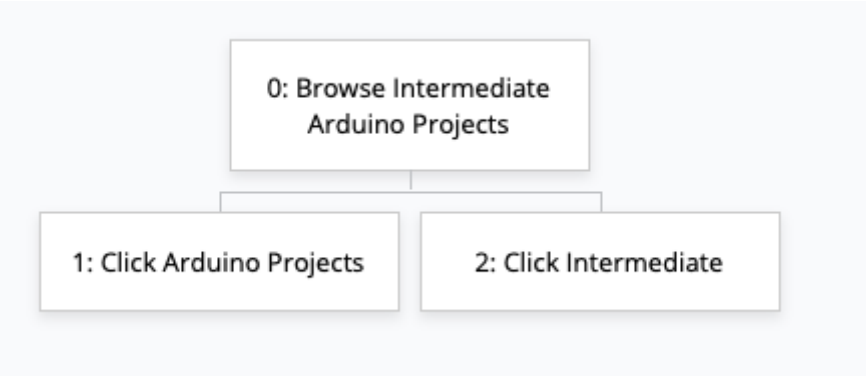


Figure 4 - HTA - Search Intermediate Arduino

Search - Advanced Arduino

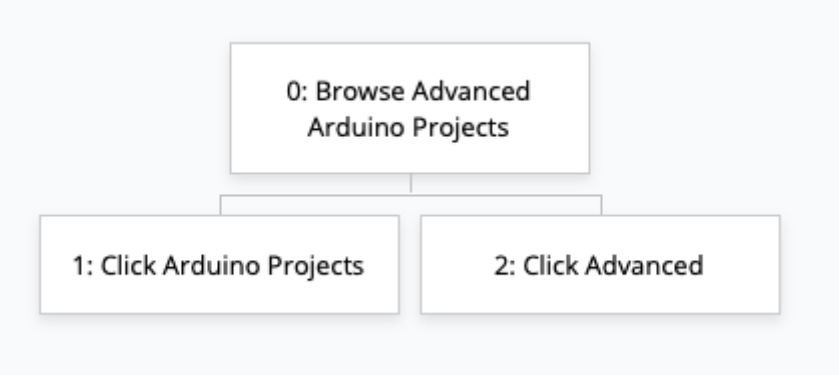


Figure 5 - HTA - Search Advanced Arduino

Search - Basic Raspberry Pi

Figure 6 - HTA - Search Basic Raspberry Pi

Search - Intermediate Raspberry Pi

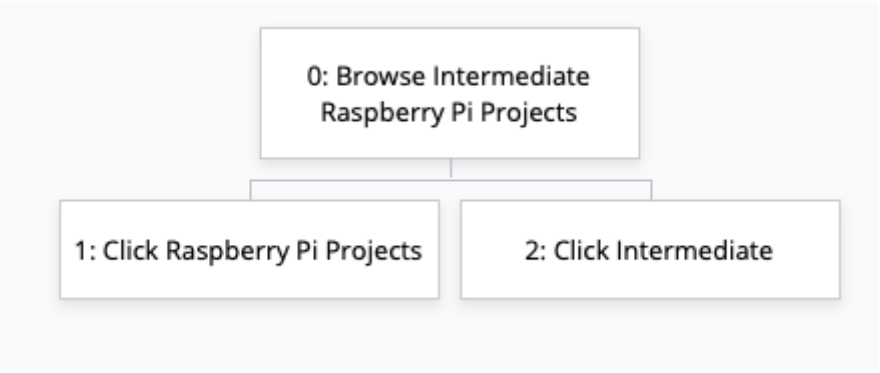


Figure 7 - HTA - Search Intermediate Raspberry Pi

Search Advanced Raspberry Pi

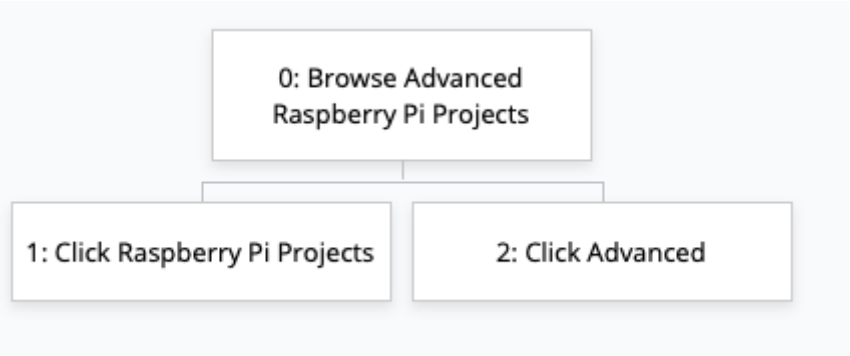


Figure 8 - HTA - Search Advanced Raspberry Pi

Search - Basic Particle Photon

Figure 9 - HTA - Search Basic Particle Photon

Search - Intermediate Particle Photon

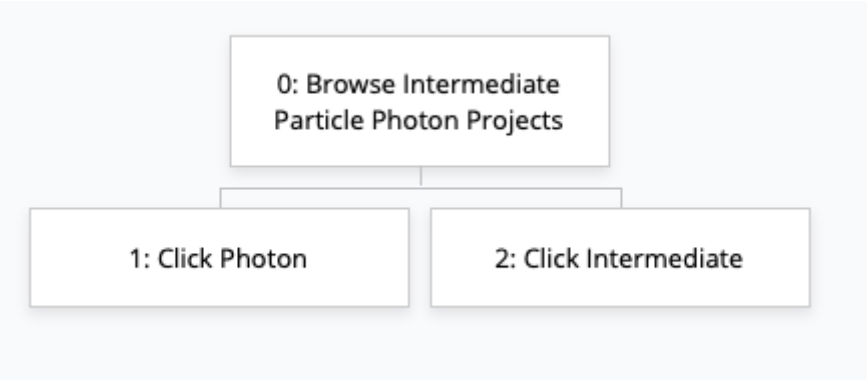


Figure 10 - HTA - Search Intermediate Particle Photon

Search - Advanced Particle Photon

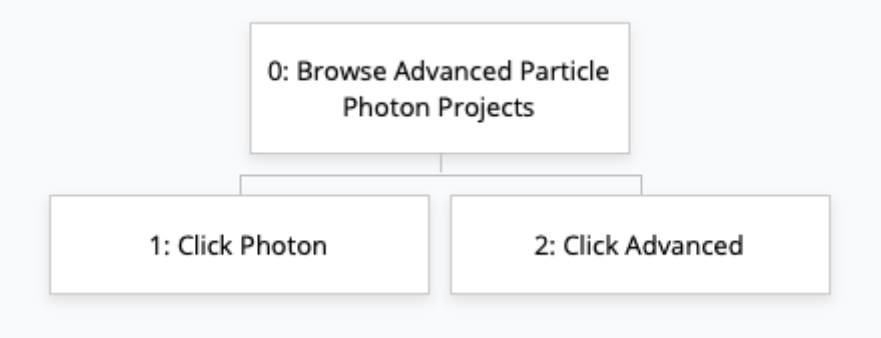


Figure 11 - HTA - Search Advanced Particle Photon

Search Function

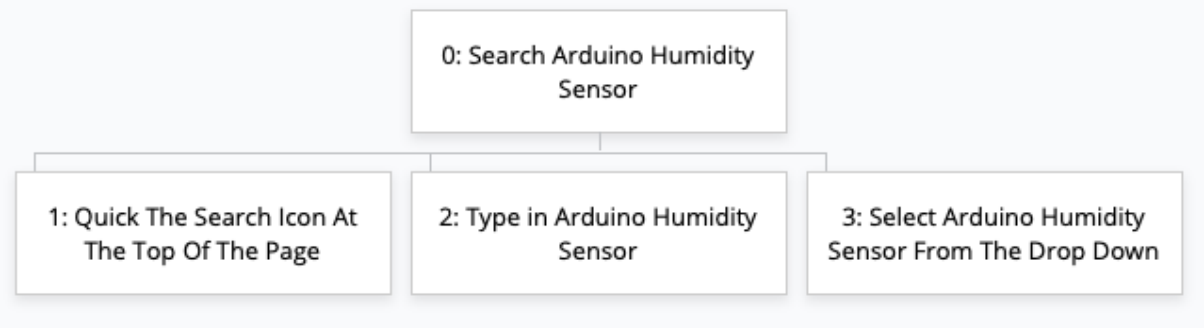


Figure 12 - HTA - Search Function

2.2 User Manual

Accessing the App

First Time User

If it is your first time opening the app, before logging in, specify whether you are an existing user or a new user, this will determine whether you create an account or log into your existing account. See fig. 12.

Figure 13 - First Time User

Signup

If you tap new user, you are taken to the signup page where you need to enter your email address and create a password then tap on signup. To finalise your account creation, go to your inbox and confirm that your email is valid by following the link in the email from Think Make Learn (check your spam or junk folder if you do not find the email in your inbox).

Alternatively, you may signup by tapping the Login with Facebook or Log in with Google buttons which will either take you to a page to log into your Facebook or Google account or if you are already logged into your accounts your account will be created straight away. See fig. 13.

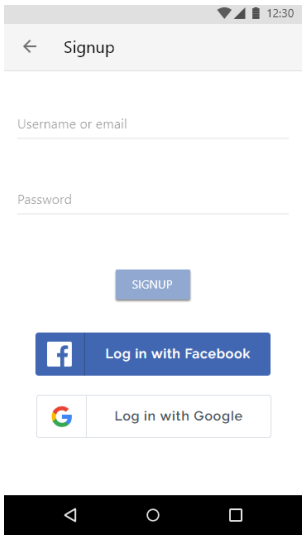


Figure 14 - Signup

Login

If you tap existing user, you are taken to the login page where you need to enter your email address and your password associated with your *TML* account then tap Login to be able to access the application.
Alternatively, you may tap on *Login with Facebook* or *Log in with Google* buttons which will either take you to a page to log into your Facebook or Google account or if you are already logged into your accounts, you will login straight away.

Once you are logged in, you will no longer have to manually log in unless specified in the settings page (see Settings). See fig. 14.

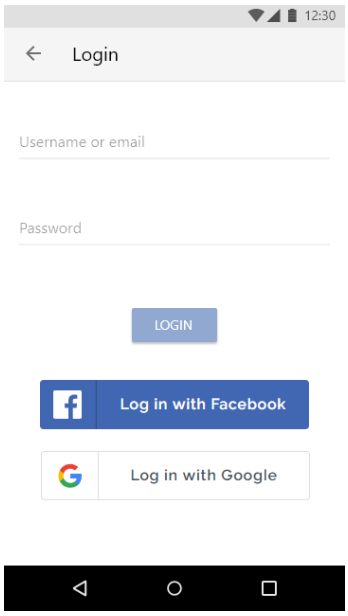


Figure 15 - Login

Competency Level

On the *Think Make Learn* browse page, you will need to specify your competency level before proceeding any further within our application. This is required for the following tabs:

- Raspberry pi
- Arduino
- Particle

Note only new users and existing users who have not done this before will be the only ones required to do this. This is especially important within our application as it ensures that users will be presented with projects that are at their own skill level. See fig. 15.

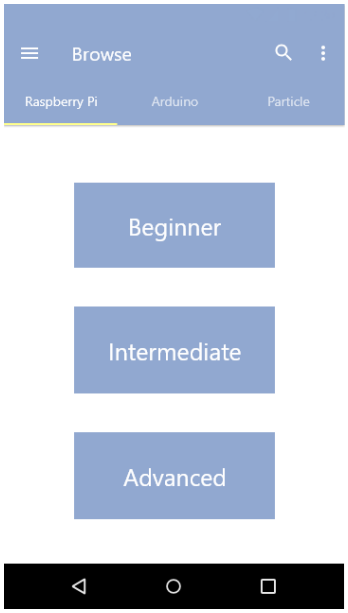


Figure 16 - Competency Level

Successfully completing a *Beginner* project unlocks *Intermediate* projects and successfully completing an *Intermediate* project unlocks the *Advanced* projects. Once you have selected your competency level within that tab by tapping one of the buttons, you are taken to a list of projects based on your competency level.

Search

Searching on the TML platform can be done by tapping the magnifying glass icon on the browse screen.

This will bring up a text box where you can enter your search terms through the onscreen keyboard which will automatically appear. Alternatively, you can tap on the microphone icon on the right of the text box and speak your search terms which will input your search for you based upon the words the microphone on the phone has picked up. See fig. 16.

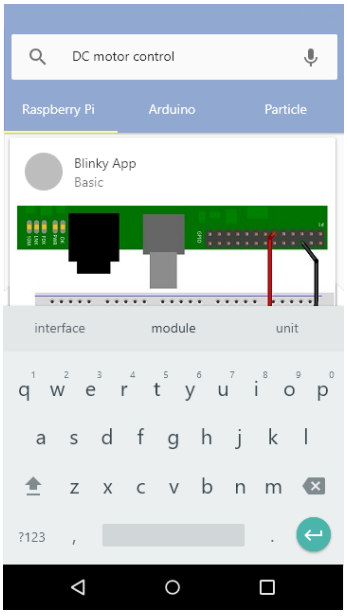


Figure 17 – Search

If no projects match your search term(s), you can tap *Go Back* on the dialogue box to be taken back to your previous screen.

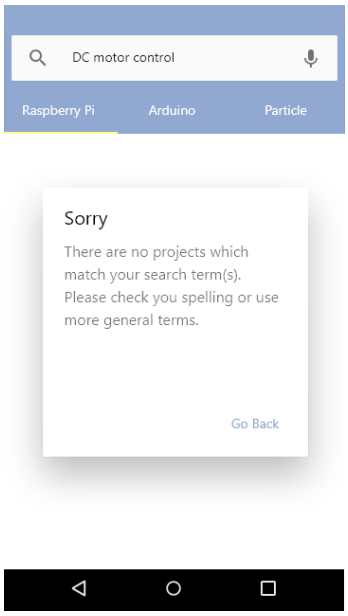


Figure 18 - No Search Results Found

Night Mode

To access the night mode/dark mode in the app, you must either open the hamburger menu by tapping on the three lines at the top left of your screen or swiping from left to right and switching the toggle for night mode. Alternatively, you can access the settings page from the hamburger menu and find a toggle for night mode there as well. See fig. 18.

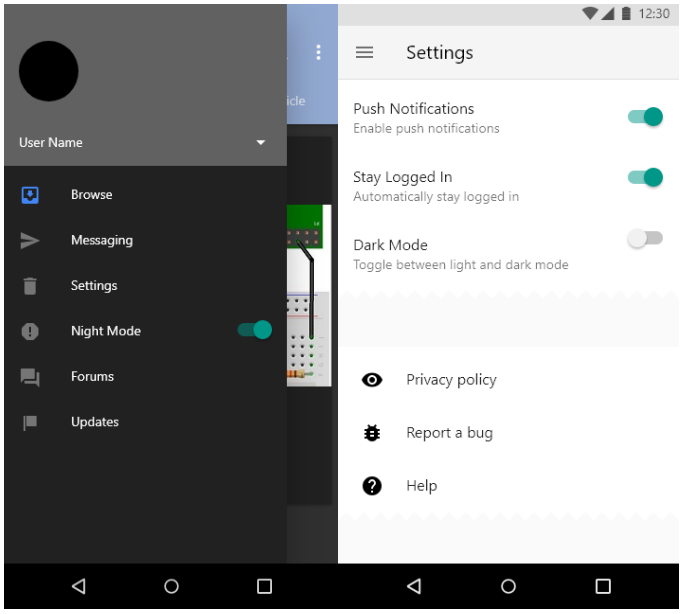


Figure 19 - Night Mode Toggles

Projects

Viewing a Project

To view a project, you must find one from either the browse page or from the search results. You must then tap View on the project's card. You will then be taken to the project page with a difficulty level, a description of the project, an option to view learning outcomes, materials required or to just start the project. See fig. 19.

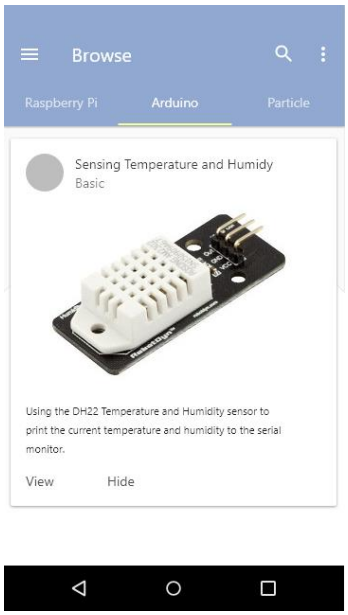


Figure 20 - View Project

Learning Outcomes

When tapping learning outcomes, you will be able to view what you will learn from the project. You are presented with the option to view the required materials for the project by tapping Materials Required. You can also jump straight into the project by tapping Start. See fig. 20.

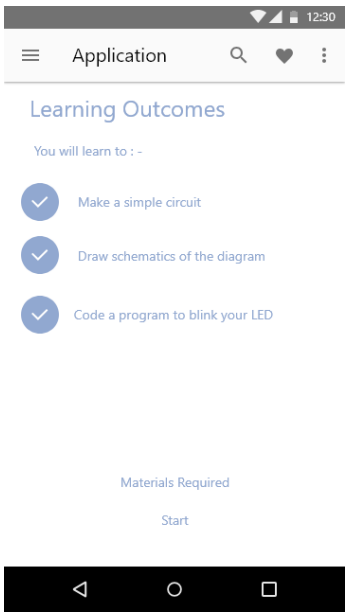


Figure 21 - Learning Outcomes

Materials Required

You will be presented a list of materials required to complete the project. Each material has a tick box where you can specify whether you have the required material. If you tick all the boxes, an option to start the project will appear which you can click to start the project.

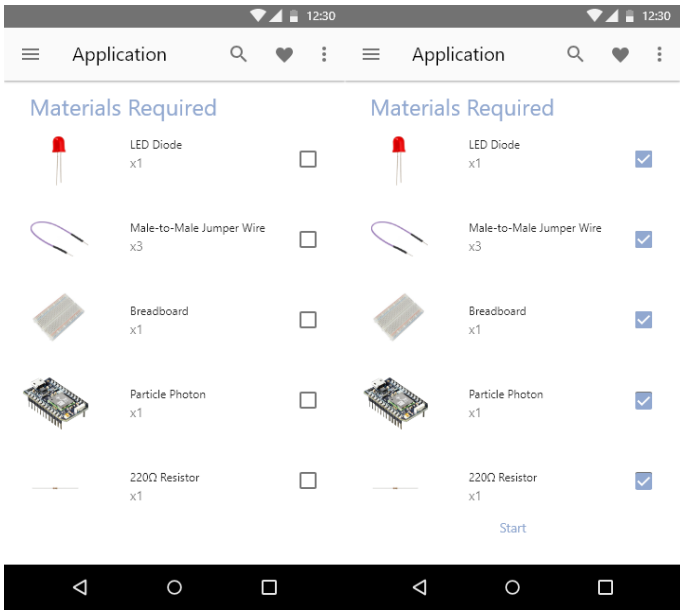


Figure 22 - Materials Required

Starting a Project

Once you have selected start, you will be taken to the first step of your project. To go to the next step, tap on the right of the screen. To go back to a previous step, tap on the left of the screen. See fig. 22.

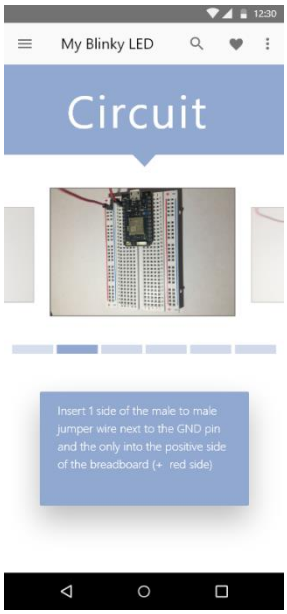


Figure 23 - Starting a Project

You may also click on the boxes to jump to and from steps. The darker box is the step you are currently on.

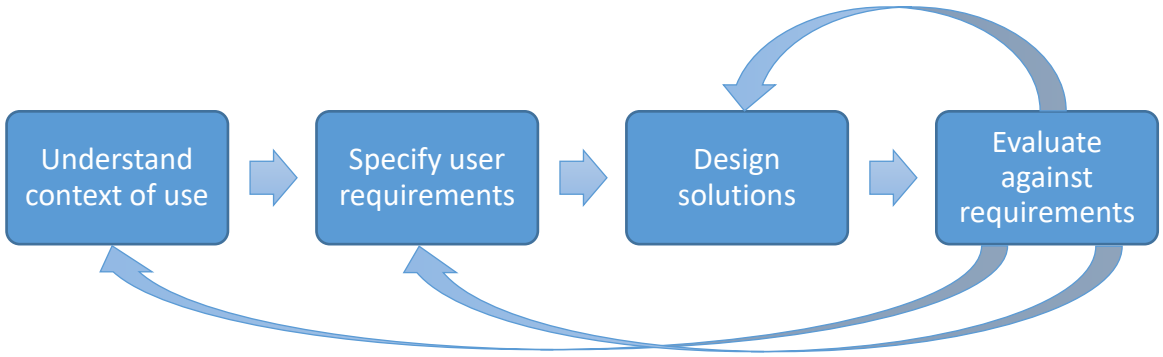
3.0 Design Justification

3.1 Introduction

Usability can be defined as an attribute of a system of a system relating to its ease of use and how easy it is to learn. The keys to usability are *affordance* (defined as how well elements in a system suggest their functionality achieved through mimicking the real-word) and *visibility* (defined as how easily interaction elements in a system are identifiable and how well they convey their effects). *Usability testing* relates to evaluating a system against the usability criteria outlined above.

In our app, we made use of a variety of usability guidelines and principles to enhance the app's usability. We have then planned usability testing through user testing to ensure that our design decisions are validated in the real-world before pushing the product to market. We have also made use of two *user personas* to think about when making our designs. This ensured that we kept our target users in mind and that we catered the design to their needs and the way they would prefer to use the platform. Keeping context in mind - the goal is to keep out of the user's way as much as possible and ensure that they are able to perform the actions and tasks they need with minimal friction from the app.

We made use of an iterative process in our design. The process is as follows:



This process being iterative means that it is not linear. We could go back to previous steps as required given new information through discussion, research and testing. The design process being iterative has the advantage of having a more usable end result as we have gone back and adjusted until we were fully satisfied with the solution. Previous iterations of our design can be seen in the appendix.

3.2 Justification

Now, let us look at how we have made use of usability guidelines and principles in our design to maximise usability through user-centeredness and understanding of context.

In general, we have adhered to Google's Material Design guidelines for Android app design. Following those guidelines benefits the user as it promotes recognition rather than recall, thus increasing the visibility of the design elements. This is because Google's Material Design interface elements are used across a large range of Android apps, thus, giving the design a familiar feel for the user. For example, our browse page makes use of Google Material Design tabs (see fig. 23). This is a design element which is present in a wide range of apps, from Google and otherwise, meaning users can recognise the design element and its purpose.

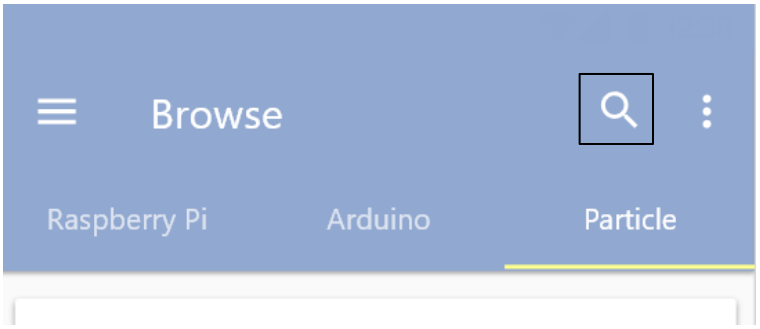


Figure 24 - Search Icon

Making use of Google's Material Design elements ensures that our design elements follow *affordance* through mimicking the real-world. For example, in fig. 23 above, we can see that the search icon is a magnifying glass, which is a recognisable real-world object which is used for finding, this means that our design elements suggest their functionality, improving usability.

Another advantage of using Google's Material Design elements is *consistency*. By using those design elements, we can be sure that they are repeated throughout the app, ensuring a consistent experience. This reduces the user's cognitive load as they do not have to figure out that different design elements perform the same tasks despite looking different. This is also true for the positioning of our design elements – they are always in the same place which also reduces cognitive load. Consistency promotes *learnability* – this is because design elements which are all in the same place across pages facilitate building muscle memory with frequent use. Furthermore, design elements which look consistent across pages mean that the user only has to learn one design per element. Consider fig. 24 below and fig. 23 above – the position of the magnifying glass for searching, the three dots for the dropdown menu and the three lines for the hamburger menu are in the same place in both pages (this is also true across all screens in the app where this design element is present). This makes for a consistent experience through the app which is easy to learn, thus increasing usability.

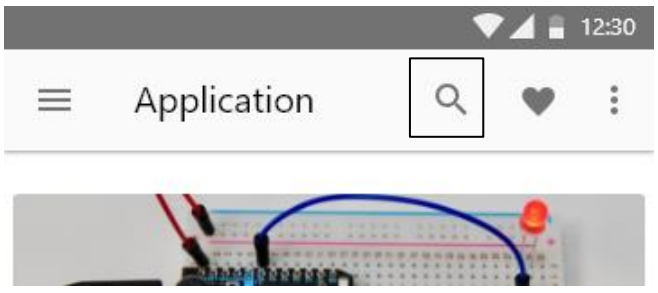


Figure 25 - Search Icon 2

Another advantage of consistent look and placement of design elements is error prevention. If design elements were placed elsewhere on a different screen, the user may tap where the element they are trying to reach was on the previous screen, which may cause them to accidentally tap the wrong button or menu, causing an action which they did not intend. Consistent look and positioning reduce frustration.

The colours we have chosen for the app follow the current Android app trend of using two colours. This is based on research which has shown that user retention drops when the app makes use of more than two colours. From a usability standpoint, fewer colours simplify the design of the interface, which matches our goal of keeping the interface as subtle as possible so that the user is

able to focus on the content. Following a colour scheme also increases consistency through repetition.

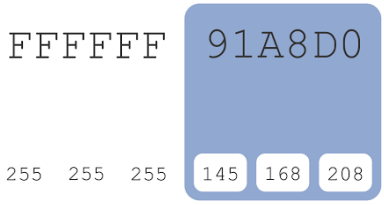


Figure 26 - UI Colours

The main colours we have chosen are white and #91A8D0 for our UI elements. #91A8D0 (Serenity) was Pantone's Colour of the Year 2016, the latest colour from them which would fit the context of the app. Those two colours have good contrast, increasing the visibility of elements. Furthermore, we chose *Serenity* because it is a calming colour and inspires trust. Given the context of the app, inspiring trust and promoting calmness works well as our project instructions need to be trusted before a user decides to follow them for their own project. The calmness also helps when the user is facing more challenging projects.

As for the user interaction with the app, we made use of a tap and swipe type interaction. This means that the user can tap buttons as well as swipe for navigation and interaction. Using both methods of interaction increases the *flexibility* of our app. This caters to a wider range of users based on their preferred interaction method. Both methods of interaction are also standard methods of interaction for apps on touch platforms. This means that the user will be presented with a familiar and recognisable way to interact with the app, making it more usable as the user does not have to learn a new way to interact with app.

Figure 27 – Tabs

In fig. 26 above, the Particle screen can be accessed by swiping right to left as well as tapping 'Particle'.

The structure of our design ensures that we group similar elements into zones as well as placing the most relevant elements at the centre. We also keep *standards* in mind by placing design elements where they would normally be in other apps, for example, in fig. x above, we can see that the three lines to access the hamburger menu are at the top left of the screen, which is where they would be present in most other applications. This gives a sense of familiarity, enhancing the usability of our application through recognition rather than recall.

The zones we have used are as follows:

A **menu/information zone** at the top where the user can access the hamburger menu, see the name of the page they are on, access the search feature and access a dropdown menu through the three dots.

Figure 28 - Menu/Information Zone

The **tabs zone** where the user is able to click to navigate to different pages based on their category.

Figure 29 - Tabs Zone

The **content zone**, centered on the screen and given priority. This zone is kept as minimal as possible as to not distact from the content.

Figure 30 - Content Zone

The use of zones also promote *consistency* as those zones persist across screens where relevant, improving the usability of the app.

We have also implemented *error recovery* in the app, a notable example of this is seen in fig. 30 where the user's search terms do not match any projects within the app. They are presented with a dialogue box with information about the system's state and an option to go back to recover from the error. They are also presented with advice on how to recover from the problem. This improves usability by keeping the user informed and giving them the option to go back from an issue.

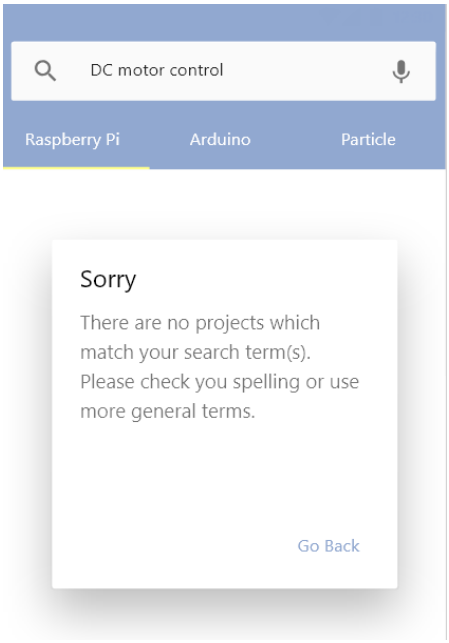


Figure 31 - Error Recovery

The TML app also features a night mode, which is a current design trend in mobile apps. This makes the design less hard on the eyes when viewing the content for extended periods of time due to darker colours generating less light. Furthermore, it gives the app more customisability, thus, improving user freedom and control. See fig. 31.

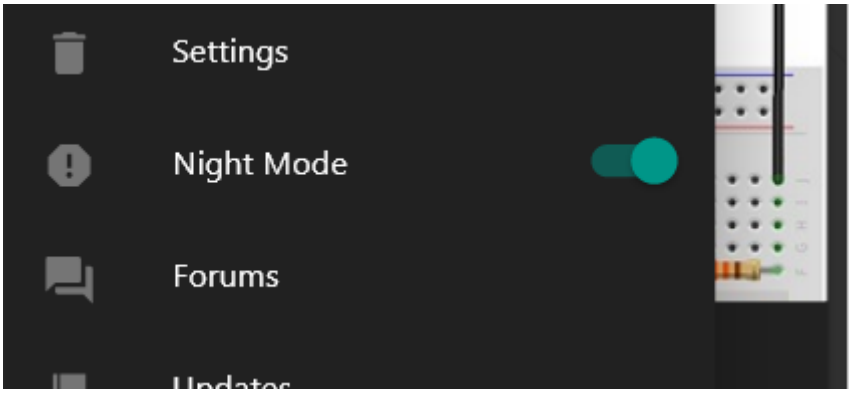


Figure 32 - Night Mode

3.3 Conclusion

Overall, the TML app makes use of an *iterative design process* to improve *usability* through the use of several *design principles* and *trends* to enhance *visibility* and *affordance* all while keeping the *context* of use in mind. The goal was to keep the interface out of the user's way and focus on the content. Following Google's Material Design, we have ensured *consistency* and *standards*, improved *recognition*, and enhanced *learnability*. We have structured our app in a logical manner which would be *intuitive* to the user and selected colours suited to the context of use and current trends.

4.0 Ethical User Testing

4.1 Finding/Gathering Participants

To initially find participants we would we want to find individuals from all demographics and levels of digital literacy and open source hardware skills to ensure that our mobile app is usable as well as has a high enough level of understandability and familiarity that users with little knowledge of open source hardware and as well as low levels of digital literacy are still able to use our app. However, we also need to ensure at the same time that we have a slight majority so we can test our most common users. For that exact reason we plan to have a 60/40 split of users with 60% have some/extensive knowledge of IOT open source technologies and then 40% with little/no knowledge of IOT open source technologies.

We are planning to test a large range of individuals to ensure that we do not discriminate against any groups of individuals/users that might possibly use our application either now in the present or the future.

In order to attract individuals towards our usability study as well as to keep the overall cost of the study down we would produce a flyer that would be sent out across the university (Deakin University - Burwood Campus) through all faculties and divisions to ensure that we get a large range of individuals with a varying degree of open source hardware skills as well as digital literacy levels. We will be providing the participants of the usability study a barbecue event to attend after all tests have been conducted as a thank you for participating in the study.

Furthermore we would then go outside of the university and go to various social gatherings as well as promote our flyer across Melbourne to try and encourage individuals to participate in our user testing to ensure that we also get a wide ranging age group as well as have individuals within our testing procedures that have a very low level of open source hardware as well as a low level of digital literacy.

After we have received and have 50 confirmed individuals of varying:

- Age
- Digital Literacy
- Open Source Knowledge
- Ethnicity

4.2 We would then start conducting our usability tests

The location of our usability testing will be conducted at Deakin University (Burwood Campus) in a closed environment (RIOT LAB) to ensure that not only does our testing meet university policy but also the participants of the tests will be in an encouraging and safe environment. The RIOT LAB can be seen in fig. 32 below.

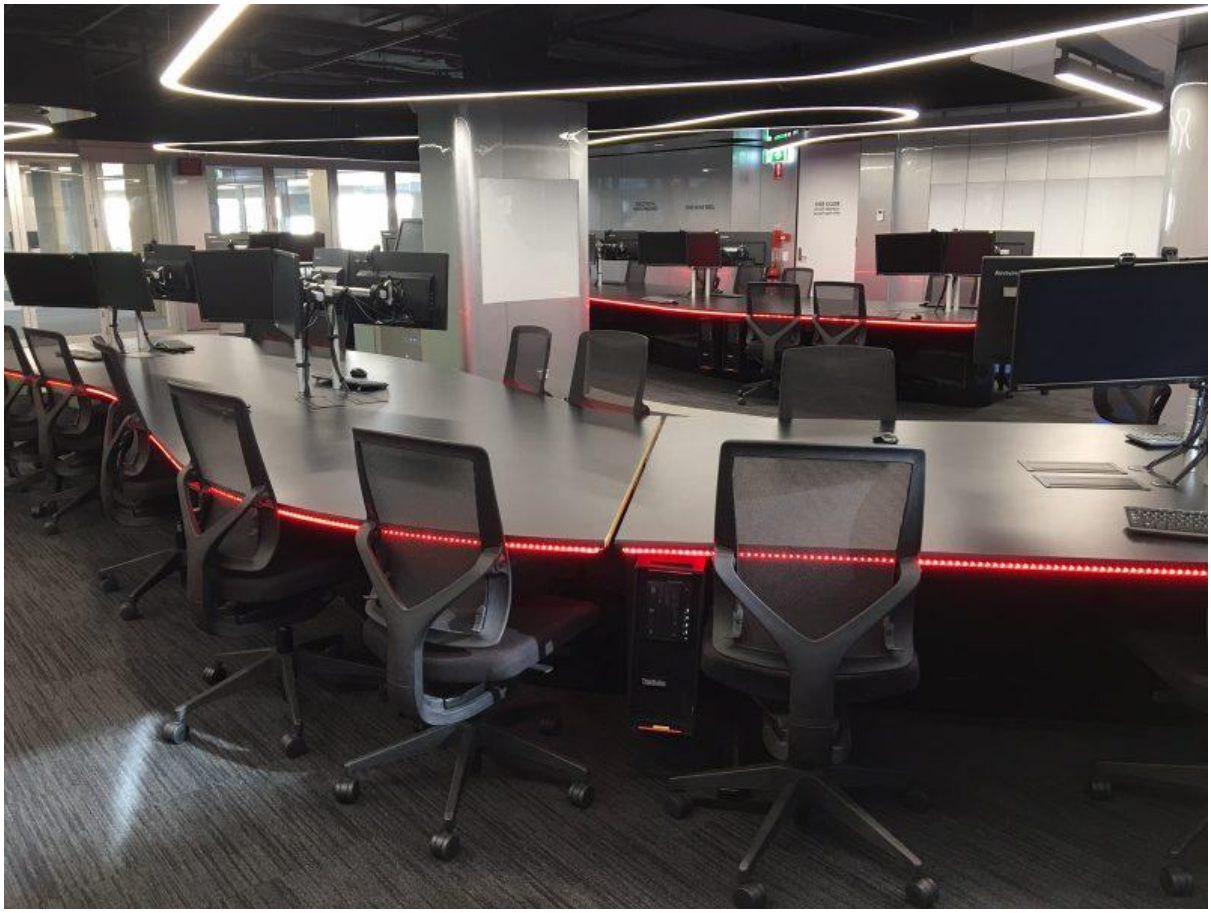


Figure 33 - RIOT LAB – Usability Testing Environment

Our usability testing would follow the Nielsen Norman Groups guidelines on why you only need to test with 5 users. Therefore following that guideline, we would conduct 10 usability tests with 5 individuals in each test as well as still ensuring that each test is different from the last to ensure that we improve our tests based upon the users feedback about the testing as a whole specifically not about the questions nor the tasks associated with the testing. This is to ensure that we gradually improve and see what works and does not work for users when using our application.

4.3 Storing and maintaining user data

Furthermore all data that is collected from the participants within our usability tests will be within accordance of the privacy act 1988, as a result we will also be following the subsequent 10 national privacy principles that follow with the privacy act 1988 to ensure that we not only conduct ethically sound testing but also to ensure that we are within the restraints of the law when handling both sensitive and non-sensitive user data. The ten specific principles that we will follow from the privacy act 1988 are as follows in order:

1. Collection
2. Use and disclosure
3. Information Quality
4. Information Security
5. Openness
6. Access and correction
7. Identifiers
8. Anonymity

9. Transborder data flows
10. Sensitive information

Ultimately while following these principles a specific set list of rules must to adhered to when gathering and disposing of user data. The list of rules is as follows:

1. No directly identifiable information will be stored permanently
2. All participants will be identified by a unique ID instead of their actual name
3. All high-level personal contact information will not be stored permanently
4. All hard copy information must be shredded after converted into the digital format and then disposed of appropriately after being verified as shredded
5. If a participant would want to change or delete their data, they will be able to regardless of the situation and if deemed that their data is unusable will be removed from the data set to ensure information quality.

Furthermore, before any testing is conducted on any of the participants, they will sign a consent form acknowledging that they are allowing us to not only use the data collected from the tests but also their personal information within various graphs such as ethnicity and background knowledge. As well the consent form also allows and gives the user the knowledge that they are able to withdraw any data collected if they would want to remove at any time. As this is important to ensure that we are in accordance to the 6th principle of the data privacy act.

Nevertheless, all data will be held on an offsite secure cloud service hosted by Deakins specific cloud servers as per university policy.

Furthermore, only three individuals will have access to ALL data held on the cloud storage facility with the exclusion of networking staff and others that might have subsequent access to the servers. These individuals are as follow:

- Lead Evaluation Testing manager
- Project Manager
- Lead Consumer Manager

The lead evaluation testing manager will have access to all the data held on the cloud storage primarily because they will be inputting the results from the user testing as well as analysing them and making recommendations towards the project manager based upon the results from the user testing. As a result, this user will need 100% access to all data surrounding the user testing.

The project manager also has full access to the data held on the cloud storage as if any member of the team needs access to a specific data set that they do not have access to they will have to go through the project manager to try and gain access. The project manager will also act as an admin staff and verify and track all access to all data on the cloud storage. Ultimately as a result requires untethered access to all data collected from the user testing to ensure that this user is able to efficiently and effectively perform their role.

As well the lead consumer manager will also need untethered access to all information collected from the user testing as the lead consumer manager will be required full access to be able to make the graph’s and usability requirements and improvements as they will be the main manager between the consumer and development team.

Nevertheless, the actual tests that are going to be conducted are the following which have been derived from the NNG’s definition of usability.

- Aesthetic appeal testing
- Usability Testing
- Mapping Testing
- Ethics Testing
- Responsiveness Testing
- Learnability Testing
- Error Recovery Testing
- Overall use of the application testing

4.4 Aesthetic Appeal Testing

The reason we are conducting testing on how aesthetically appealing our application looks is the following:

1. To ensure that they don't get turned off and uninstall our application from the way it looks
2. To ensure that our application promotes a neutral and friendly learning feel towards the user in the way it has been designed to by the front-end development team
3. To engage and want to make the user come back and continually use our application to further grow their knowledge of IOT
4. Recommend it to others and be proud to show it off to others around them

The way we have chosen to conduct our aesthetic appeal testing is to show the participants of the user testing a wide range of screens with our ultimate goal being that all of our screens within our application eventually get seen and tested by a range of different participants in the test. However, during our tests, we will prioritize and test our more frequently visited screens on our application the most to ensure that those screens are up the standards of a wide range of users as they are the most frequently visited.

Roughly each participant selected for this testing will provide feedback for 15 different screens on the application and provide the feedback based upon a Likert scale from 0-10 with 0 being not attractive at all and 10 being the most attractive application that they have ever seen. A sample Likert scale that might be used during our aesthetic appeal testing is shown in fig. 33 below.

Figure 34 – Likert Scale - How Aesthetically Pleasing Do You Find This Design

Additionally, we would then ask the users if they did not rate it 10/10 why and what would you change to make it 10/10 as to provide further feedback and to ensure that we get the most out of our appeal testing on our application.

4.5 Usability Testing

As with any user testing you always want to conduct a usability test on your application to ensure that it is usable by not only your target user but by any possible user. The way we have decided to conduct our usability testing is to break in up into 3 distinctive parts:

- Is it usable for you?
- Out of 10 what would you rate the usability of this application as
- Do you require any additional usability features for this application to meet your needs/ if no and if you did not rate it 10/10 why?

The reason behind us asking these rather abstract questions is for us to truly know and understand why they don't think it usable for them or why they believe that it is usable for them. This is to ensure that we avoid generic questions during our testing.

Furthermore, we will be using a Likert scale again from 0-10 to actively rate and gauge the usability of the application. With 0 on the Likert scaling being not usable at all whereas 10 on the Likert scale means the user did not encounter any usability issues at all during the testing phase. A sample Likert scale could be the following in fig. 34.

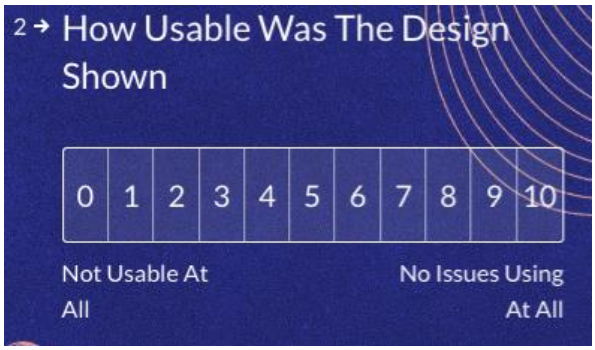


Figure 35 – Linkert Scale - How Usable Was the Design Shown

The reason that we have created this Likert scale so ambiguously is because it should technically be a yes or no question however a simple yes or no question is not enough to quantify and actually graph this question and make reasonable assumptions out of. As a result, we are measuring this on a Likert scale to better gauge where the user believes the level of usability within the application lays at.

Thus, the Likert scale is then compared to the participants answer to the following question. When the result is then compared to this specific answer it will enable us to see if the two answers line-up or not ultimately allowing us to make a direct assumption about that participants validity of his or her answer

Nevertheless, the questions surrounded around the usability testing is then focused on and around what the user actually rated which leads into the final question for this testing is “does this application require any additional usability features” to ensure that we get meaningful and quantitative data to graph and analyse.

4.6 Mapping Testing

The main reason we wish to conduct mapping testing is to ensure that our overall layout and connections within our application makes sense to not only users of a high technical background but also individuals with a low-level digital literacy. This is ultimately because we wish to target all

demographics with our application and as a result need to ensure that we don't overlook any aspect of our design when targeting this broad range of users.

We have decided to conduct the mapping testing by simply getting the user to do a set list of normal tasks that you would do in almost any application: These tests are:

1. Find the settings bar
2. Go to the basic Arduino projects page
3. Activate dark mode
4. De-Activate dark mode
5. View any project

The users will have a total of 30 seconds to complete each of these tasks. The mapping of that specific area/function will then be graded upon the amount of time taken for the user to complete it VS. their level digital literacy. Sample tables are shown below.

4.6.1 Low Level of Digital Literacy

Time (Seconds)	Points
0-10	12
10-15	8
15-20	7
20-25	4
25-30	2
30+	0

Figure 36 - Low Level of Digital Literacy Marking Rubric

4.6.2 Medium Level of Digital Literacy

Time (Seconds)	Points
0-10	11
10-15	9
15-20	8
20-25	4
25-30	2
30+	0

Figure 37 - Medium Level of Digital Literacy Marking Rubric

4.6.3 High Level of Digital Literacy

Time (Seconds)	Points
0-10	10
10-15	8
15-20	6
20-25	4
25-30	2
30+	0

Figure 38 - High Level of Digital Literacy Marking Rubric

The main focus around us focusing the lower level of digital literacy user is they will most likely have the most amount of trouble actually navigating our application and as a result we reward more points based on the lower level of digital literacy level. Ultimately what we would want to see is that it naturally scales out the point distribution between all levels of digital literacy, however we anticipate that this might not happen.

4.7 Ethics Testing

As a group we decided doing ethics testing primarily because we want to ensure that we don't offend anyone in anyway while they are using our application. As a result, we have ensured that within our user group we have a wide range of users from all different backgrounds.

We won't have any direct ethical testing conducted however at the end of the testing we will ask them the following two questions:

- Do you believe that your data you have entered on the website is secure?
- Did anything during the testing upset you or cause any level of discomfort, if yes, why?

Ideally, we would hope that our website does not offend anyone however in the real world this is not always a possibility. Thus, we are incorporating this within our user testing to ensure that participants have a dedicated time to talk about their experience about the testing as a whole.

4.8 Responsiveness Testing

As with most user testing now, we have also chosen to conduct responsiveness testing to ensure that our application is responsive enough towards a broad range of users. Conducting responsiveness testing is key towards our application specifically as we have a varying range of different sized screens and as a result the users might want to go into landscape mode instead of viewing it in the expected portrait mode.

As a result, responsiveness testing is key to our application succeeding within the marketplace. To effectively gauge how well our application responds to users wants and needs the users will be set a specific set of tasks that a user would normally do on our platform.

These tasks will be the follow in order:

1. Change the orientation of the screen
2. Does the screen scroll?
3. Go back to the previous page
4. Exit the application
5. Go to the next page using a different method

To evaluate the results of these tasks the outcomes will simply be either success or failure. Ultimately resulting in a final score out of 5.

Due to the natural simplicity from these tasks, if the overall result from the responsiveness testing is less than 95% more extensive testing and an overhaul of the design will have to happen to ensure we have an overall higher responsiveness testing score.

However, for the purpose of the first-round testing on users we will assume that our overall score will be greater than 95%.

4.9 Learnability Testing

To ensure that our platform is usable by individuals of all levels of digital literacy as well as numeracy and literacy levels we will conduct learnability tests of varying degrees. These tests are as follows:

- Do these interfaces look familiar to you?
- Is the settings bar where you expect it to be?
- Do you believe that this design is easy to understand?
- Where is the home button?

- Can you find the additional functionalities such as dark mode?

As a result of the open nature of questions they will be recorded as a yes or no so we are able to receive and record both quantitative and qualitative data for research papers. Much like the Responsiveness testing, if we receive less than a 90% approval rate for our platform in regards to just the learnability testing a more extensive and intensive form of testing will have to be conducted on specific user groups to understand exactly where our platform lacks and for which demographic.

Ideally however we expect our initial design to not exactly reach that threshold but very close to that threshold as we believe our high-fidelity model is not perfect however at a level where it should not greatly affect the overall or even the specific learnability results that would appear from the specified learnability testing as detailed above

4.90 Error Recovery Testing

At the heart of every application regardless of the marketplace or the intended or unintended proposed use of that said application if the error recovery and error clarity is not clear enough to the user than the application is almost destined to fail when it does not act like expected when errors appear.

Thus, we will conduct extensive testing within both error messages and error recovery. The overall initial tests will be as follows:

1. What is happening here (Show an error message)
2. Show the incorrect login screen error message
3. Demonstrate and make the user use a static screen and see how long it takes them to realize that the screen is frozen

If the results from all of these tests are not within the top 90% of all recorded results a complete overhaul and more extensive error recovery testing will be conducted to ensure that not only is our platform a level above the rest however also more user focused and truly user centric.

4.91 Overall use of the application

To measure the overall use of the application we will be evaluating it with a Likert scale from 0-10 with 0 be a truly awful application and with 10 being the best application ever. A sample Likert scale that might be used in the actual user testing is shown below:

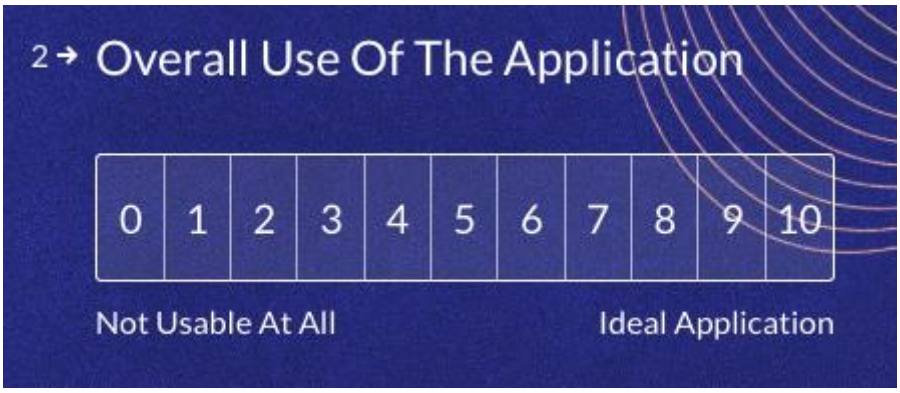


Figure 39 - Likert Scale - Usability Testing

The reasoning behind why the overall use of the application is kept to a single Likert scale result is that this data will be evaluated against the other answers that particular user has also submitted. Ultimately resulting in this answer being used as a benchmark to ensure that the users answers are

not only valid and consistent with regards to the normal range that has been calculated by a distance based machine learning algorithm to prove the legitimacy of the users data but also to give us an overall range of what the user believes of our platform by the end of the user testing.

4.92 Conclusion

Ultimately from our user testing we want to receive not only information that we can graph and demonstrate where we lack gaps but also gain insight into what specifics the individuals from the user testing would expect in a platform like ours.

However, we expect from our user testing to know roughly where we stand within the release state/schedule of the timeline when comparing it against the software development life scale focusing on the testing side. This is particularly important as it will greatly help us predict the expected release date of our platform.

5. Usability Guidelines

Based upon the current trends and the identified distinctive differences in functionality and the requirements needed within both an open source educational learning platform and standalone educational platforms a specific set of design guidelines have to be adhered too to ensure no key elements are left out during the designing phase. Thus, to ensure all key elements are included within baseline designs, for that reason we will be classifying and evaluating our designs against the following criteria:

- Usability framework (Nielsen, 2003)
- Learnability
- Efficiency
- Memorability
- Errors
- Satisfaction
- Usability Constraints (Mayhew, 2006)
- Cognitive, perceptual and motor capabilities
- User's physical and social environment
- Software and/or hardware constraints and system platforms

5.1 User Needs and Experiences

As mentioned earlier within the current trends at the heart of designing an educational IOT platform for individuals of all skill levels the user is the key focus and has to be constantly evaluated before, during and after the design as the users' needs and wants will constantly change in not only from an educational standpoint but also from an IOT point as well. Thus, in order to evaluate and understand what the user of our application wants and needs we first need to look at how best individuals learn.

Multiple studies get conducted/re-evaluated every year to ensure how best to educate individuals. One such study conducted by Julie Dirsen called Design for how people learn (2015) emphasises how individuals learn best from a combination of both theory work as well as practice work. As well as she emphasises how when this is combined with an external 3rd party such as a teaching assistant or professional in that field as they are able to answer all the questions they might have. As a result, increases and reinforces the knowledge they have learnt/implemented.

This is particularly important within our IOT learning application as if the individuals don't understand the theory behind how an embedded system works then then they won't be able to take that knowledge and repurpose it for another embedded system.

However, it also has to be noted that not only does Dirksen high suggest another external learning factor but also that you need to design your software in order to predict the user. The key areas she suggests to predict are the following:

- The individual's current knowledge level
- The individual's current skills in a particular area
- When you need to provide motivation towards the user to keep them studying and learning
- When best to integrate the software into the daily habits of the user in their own set study routine and if they don't have a study routine make one for them
- The current environment that the individual might be learning/taught in

These specific design factors all have to be taken into careful consideration when implementing the design of any educational software as if one or multiple factors are overlooked it can not only possibly cause the user to lose interest and have negative experience with the application but more seriously it can set off a spiral effect in which they eventually lose all motivation to study which is the main area we are trying to always actively avoid.

However, we also need to consider that there is a high possibility that we might deal with individuals of an extremely young age who might have a relatively low level of digital literacy. Thus, we need to place a significant emphasis on the familiarity of the software towards users of all abilities and backgrounds. This is particularly important within Australia as Australia is now a very multicultural country thus as a result has a wide range of users from various backgrounds

Furthermore, we also have to take into consideration how the user will be able to adapt the software towards their needs. This is with respect to individuals with either a physical or cognitive disability. As these issues have to be handled with the utmost care and importance as to ensure that we do not discriminate against anyone as the ultimate aim of our software is to help all individuals learn IOT in a fun and engaging environment

5.2 User Interaction

When considering a user's interaction within an education application specifically referring to IOT it can be broken down into the following screens:

- Login Screen
- Home Screen
- Specific Project View
- Resources needed that specific project
- How to make that specific project

As the result from these 6 screens will require the users of an IOT application to do drastically different things based upon the criteria they have selected. For that exact reason we will be considering them as their own category to ensure that no user interaction is missed or even possible overlooked during the design and usability phase of development within our application we are developing.

5.3 User Interaction Login Screen

When designing the login screen of our application we need to remember that we have a high likelihood of dealing with users who might have a low level of digital literacy. Thus, as a result need to ensure that we keep the login screen as simplistic as possible, however to also provide the user with multiple options to login to ensure that the process is as easy as possible. Thus, as a result the login screen will encompass and must have the following elements:

- The ability to have multiple ways for the user to sign in such as through their google or Facebook account, this is to provide the usability with enough flexibility when choosing how they want to pair with their account to our platform
- The ability for pre-existing users to skip the sign in page through using a remember me function, this is to ensure that the user has an increase ease of use when logging in within our application
- A distinctive clear and simplistic view to the login screen as to not confuse or overwhelm users to ensure that our websites provides that family friendly feeling towards all users
- Error messages, if a user tries to login and has entered an incorrect username or password, they will be provided with a popup message saying incorrect password or username. The reason we do not want to specify which is specifically is wrong is to increase an accounts security as it provides less information to potential hackers however it provides enough clarity to user that they have entered the wrong details

5.4 User Interaction Home Screen

When designing the home screen of the application we have to remember that once again on this screen we are dealing with individuals of a rather low level of digital literacy. Thus, we have to make the home screen as clear as possible as to not to confuse the user. Therefore, we have to ensure the home screen is fluent and as clear to the point as possible due to the wide range of users we might encounter within our application.

Therefore, we have to impose a set of strict guidelines that have be adhered to on the home screen. These guidelines will be the following:

- The ability to be able to browse various categories of open source technologies, this is to ensure that the user can see the various technologies we offer to the user
- The ability for the user to change orientation of the application only if they have enabled this. This is to ensure that the user does not get confused when the orientation changes all of a sudden.
- Error messages will be provided to the user if any error occurs while they are using the home screen. The error message will be provided within layman's terms in the specific language they have selected within the application.

5.5 Specific Project View

When designing the project views within the respective categories of specific IOT hardware we need to ensure that the user gets enough information about the project, however not too much that it deters them from doing the project. Furthermore, we also need to ensure that each project in the project view has an aesthetically appealing design that not draws the user to the project however also clearly indicates to the user how to view that specific project.

Therefore, the following list of guidelines should be adhered to too ensure that the user is not only attracted towards the projects but also are still provided enough information as to how to view the project further:

- The images of the projects not only look appealing to the users but also alluring to the user as to allow them to know that they can click on it to view the project further.
- The information provided about the projects need to be short and concise as to not overload the user and deter them
- The user must be able to view the overview of multiple projects once as to ensure that the number of clicks/swipes required by the user is kept to a minimum

5.5.1 Resources Needed in That Specific Project

When thinking about how the users will access and interact with the resources page of a specific project, we can categorize it within two specific areas. These areas are:

- Ticking off the required items needed to complete the specific project
- Viewing the required items need to complete the specific project

Therefore, we need to ensure that not only are the pictures of the required items are appealing towards the user but also the user interaction of the ticking of the items are large enough that users with big fingers can accurately tick the specific.

Therefore, the following suggestions should be followed to ensure that all users regardless of finger size are able to accurately tick off items while still having the required aesthetically appeal:

- The tick boxes should be far enough spaced apart to ensure that users with big fingers don't accidentally tick off two boxes accidentally this to not only reduce the possible amount of clicks however also decrease the chance of a user becoming frustrated
- The spacing between each of the images/text should be enough that they still correctly correspond to the tick box. This is to ensure that the user does not get confused as to what image/text is corresponding to the specific tick box

5.6 Project Build

When planning which page the user will spend the most time on, the project build page is where users will spend roughly 75% of their time. Therefore, very high emphasis must be placed upon how we design this page specifically.

Therefore, the a very strict and specific set of guidelines have to be adhered too to ensure that the not only the user interaction is meaningful but also aesthetically appealing to a range of users. The guidelines are the following:

- Applied Behavioural analysis has to be included within the design to ensure that we predict not only how the user might use specific pages however also where they might get confused or an error might occur to ensure that overall user satisfaction is high
- The language used to convey ideas and demonstrate specific procedures/ideas has to be clear and contain no jargon, which might possibly confuse them. This is to ensure that the procedures and ideas being taught is well received and understood as this is essential as our application is an educational learn tool
- The way the user receives the information has to be presented in a structured and logical format. This is to ensure that the user is able to actively take in the information

present, process it to make decisions based upon the project and then implement that information into the project they are developing

5.6.1 Interface design - Durable Interface for All Screens

When we are designing the interface and trying to predict how the user might/will interact and use the interface we need to ensure that the user does not lose any functionality for the compromise of size on the screen. Therefore, we have to design the interfaces for our application to predict the user's actions as well as where they might possible misuse/make a mistake within our application.

5.6.2 Interface design - Reactive Screen Size

Within the current day and age, the evolutionary trend of technology and the size of a phone's screen change constantly and rapidly, so much so that a new screen size for a phone comes out every year now. Thus, we need to ensure that the software has a reactive interface/screen design to ensure that all elements on the screen are perfectly placed and maintain their structure over various screen sizes and orientations to maintain the aesthetically appeal towards the user.

Through maintain the information's structure it not only helps organize all information in a logical order on the screen, however it also helps to contribute to a more fluent flow of how the information is organised which contributes to a quicker reading time.

Ultimately resulting in a greater user satisfaction with the application due to how structured and fluent the information is provided towards the user, regardless of the user's digital literacy level.

5.6.3 Interface Design - Minimalist Design

Due to the amount of information provided towards the user we need to ensure that this excess of information does not negatively impact the design. Thus, we need to make distinctive clear points in our information to ensure that we don't confuse new users especially those which might have a lower level of digital literacy.

Therefore, we will design our application with a minimalist design at the core with specific buttons and scrollable screens. We are going to do this to ensure that the program does not overwhelm new users, however at the same time still ensuring that the pre-existing users still have enough quality information to carry out the project. This is really important as various research papers and academic journals have continually shown that the majority of individuals especially young children become overwhelmed easily.

5.6.4 Interface Design - Navigation

When looking at a broad range of educational software they either have a very complex navigation system which is hard to understand or a very simplistic easy to understand navigation system. Thus, many fail as the user has to many options or not enough and this is particularly because the applications don't make use of Fitts law. Ultimately resulting in a lower level of familiarity on how to best traverse the software for the user.

Therefore, we will have to implement a navigation system that not only results in the user having to click less to require where they want to get within the application but also reduce the amount of aimless scanning required by the user. Furthermore, also helps maintain the minimalist design through fewer search buttons.

5.6.5 Interface Design - Consistency

At the heart of every effective and usable interface design is a consistent layout. This is because if the interface doesn't have a consistent look across all pages within the application than the user will

lose familiarity with the software. This is because the user will look in one place for something that is there on one page however on another it is somewhere else. In turn ultimately resulting in a lower level of user satisfaction as well as creates less meaningful human computer interaction

Therefore, for that exact reason the interface within our application will be designed to highlight specific areas of importance such as where the main navigation buttons are always shown as well as how much white space is shown at one place when additional functionalities/information is also shown at the same time.

5.6.6 Interface Design - Error Handling

Almost any application has the ability to the have the most eloquent and well thought out design however if that specific design does not include error handling which predicts the user's actions then ultimately the application will fail as a whole.

This is primarily because various studies have shown that when individuals, regardless of age encounter issues the large majority of them it will increase their blood pressure. As a result of a heightened blood pressure it would only take a few more issues/errors in the application until they create/have an extreme dissatisfaction with the application and it simply does not work like intended or predicted.

Therefore, error handling is essential to include within our application as our major objective is to keep our users entertained and engaged with the software and our platform is unable to do that if they have almost any level of dissatisfaction with the application. Thus, when designing the interface, we will place a high emphasis on not only how to prevent user error however also how to catch system errors that commonly appear when using older versions of phones which our applications framework might not work on.

6. Current Trends

Currently our application falls within the educational section of the app store. As a result, there is a varying range trends that we could of base our application off of.

However, we are going to be basing our trends off of applications within the programming category of the app store that also fall under education as our application is primarily a programming education application at heart with the touch of IOT added onto it. However due to the niche market that our application targets we will also be considering trends seen on websites that also meet the same criteria/categories that our application does such as the IOT side.

However, it needs to be noted in particular that these trends will most likely change as this type of software/application is designed around users and when the user’s needs change so will the application. As a result, users currently require a more personalised learning experience tailored towards their needs and wants. Nevertheless, a single constant that has stayed the same is that they have always had a minimalistic and familiar design pattern for finding key information. The most common and relevant design trends are shown below:

Nevertheless, some key factors are currently emerging from all educational technology in general. These are the following:

- Use of Fitt’s Law - Almost every large-scale educational software and most software today implements Fitt’s Law into its design to ensure key information such as the registration, home button and help section is easy to find and use (Tang et al. 2017, p. 33-40). They do this to ensure the software has a high level of familiarity when a new

user starts to use the software. Which in turn results in the user spending less time looking for certain functionalities. Resulting in not only a better user experience but also more effective and meaningful human computer interaction. Ultimately resulting in the user learning more in a shorter period of time which is ideal.

- Personalised account feed - The majority of educational software in the modern age has a distinctive and adaptive account feed where the user is able to view their current progress. Within this account feed they are able to see how they are going and where they might need to improve based upon the answers and tasks they have completed within that specific software. This helps to ensure that the user is focusing primarily on the area's that they struggle and reinforces the knowledge they already know in small increments to make it less obvious to the user as to keep the user more entertained.
- Colour – The majority of educational software on the market place followed a specific colour scheme using colour theory generally focused around lighter pastel colours and only ever used neutral colours when displaying images of varied colours. As a result, created an aesthetically appealing look within the software. Most of the educational software uses this logic paired with applied behavioural analysis to ensure a fluent and aesthetically appealing design to ensure that the software is easy on the user's eyes which in turn ultimately causes the software being less strenuous on the user's eyes and decreases the likelihood of the user become angry of frustrated at the software due to eye pain or other factors. Furthermore, by creating less strain on the user's eyes it ensures that the user is less likely to close the educational software due to extensive strain on their eyes from staring at a screen for an extended period of time as this is a significant issue in today's day and age.
- Embedded teaching videos. Most of the educational software had embedded teaching videos in their software which appears after failing a specific question more than three times then it would pop up on the screen. This embedded teaching video would then explain the current aspect being taught whether it be variables or a depth first searching algorithm. Generally, it is a professional video in which the user is able to interact with it to ask questions to the pre-recorded video. Thus, it in turn allows the user to have a personalised feel towards the user as they a guiding the video where they need to go to learn more.

However, we need to consider that on average all educational software uses these trends and our application is an educational software aimed at open source software. Therefore, we also have to look at the trends in educational open source hardware software to ensure that we get a clear view of the current marketplace.

Currently key trends in open source software for teaching IOT are the following:

- Featured Projects - Within almost every open source software for teaching IOT they have a featured projects section within their software. They do this to not only showcase what the users could possibly make in the feature if they keep using their software but also the level of expertise the software will be able to provide them. Thus, as a result giving the user more incentive to stay with that software due to the generated idea of being able to build one of the featured projects
- Post Your Project - Once again a common occurrence is the ability for users to post their project to the website and receive feedback from experience and verified users about the project. This is important for the software as does it not only generate more content for their platform but it also provides the user with the sense of belonging and that they

are truly a part of that community which is essential to build and maintain an active user base within an open source context.

- Contests - To keep users constantly coming back and engaging with the website many platforms now run contests in which companies offer a cash prize to whoever can build a specific product for them. One example currently on an IOT website called hackster.io is a competition offered by Soracom has a cash prize of $10 000 for someone to design and build a miniature cell tower. While this might be too complicated for new users it provides even more incentive to learn IOT as they could have a monetary gain from their learning. Ultimately increasing not only their user base in the progress but also the websites affiliation with other major companies. Resulting in a win win situation for all three parties (User, Website, Companies) at stake.

7. User Personas

Persona 1 – Hernandez Garcia

Age: 19
Education: Completed High School
Occupation: University Student
Course: Bachelor of Computer Science
Location: Bogota, Colombia
Spoken Languages: Spanish (Primary), English (Upper Intermediate Level)
Family: 4 brothers and 2 sisters
Primary goal: To get a deeper understanding of the hardware side of developing for internet of things (IoT) devices.
Secondary goal: To show off to his brothers and sisters and introduce them to some cool projects.
Primary challenges: Not confident in setting up circuits for projects.
Secondary challenges: Sometimes forgets where he is at in the steps given, due to distractions from siblings.
Hobbies: Programming, video games, soccer
Computer literacy: High
Commonly used systems: Windows, Linux, Android phone, Raspberry Pi, Arduino

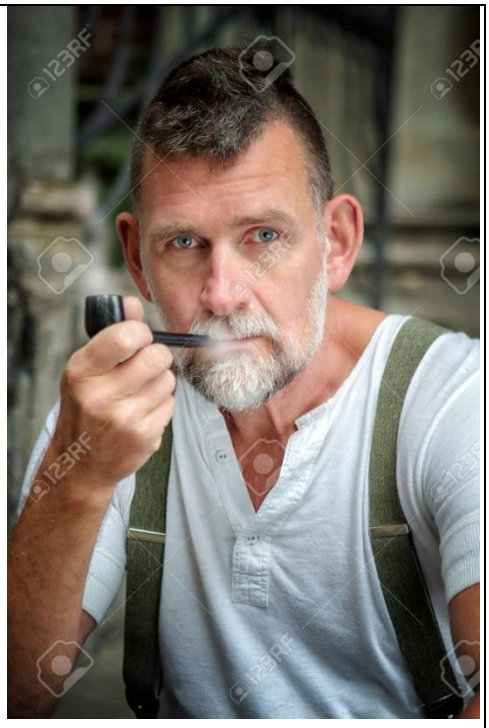
Hernandez is a first-year university student, studying Computer Science at the National University of Colombia. His first year consists of many introductory programming units, some math and data science, along with an introduction to data capture technologies.

Although he is able to complete the required work for his studies, the data capture unit takes a significant portion of his study times. It is not uncommon for him to spend many hours looking at circuit diagrams, trying to replicate them and just not quite getting the results he expects.

His loving family provide many interruptions, as being the eldest brother in the family, he has to sometimes help in chores that some of the younger ones can't do. When he comes back to what he was doing before, often times it is just easier for him to start from scratch due to not knowing which step he was at.

All of his brothers and sisters look up to him and like to try copy what he is doing, or at least understand what he is doing. The ones that are old enough have all joined his local soccer team at La Equidad. Hernandez hopes to show off some of his university creations and inspire them to start making things too.

Persona 2 – William “Willy” McLaughlin



Age: 51
Education: Completed High School and relevant work certificate (Cert IV in Train Driving)
Occupation: Train Driver
Location: Melbourne, Australia
Spoken languages: English
Family: Wife, Daughter and Grandson
Primary goal: Get greater control over the model train circuits, possibly even build a mini city
Secondary goal: Spend some time with his grandson teaching him about the things he makes
Primary challenges: Lacks basic programming skills.
Secondary challenges: Does not know where to look to find circuits or parts.
Hobbies: Model train collecting, woodworking, driving
Computer literacy: Low
Commonly used systems: Windows, Android phone

Willy is a full-time train driver working in Melbourne for Metro Trains. His work is shift-based, so his hours can be irregular for long periods of time.

He greatly enjoys driving trains, both the real ones and the model ones. There is great satisfaction in laying out track on a table, winding around all sorts of obstacles and just watching the trains go around. Willy wants to be able to independently control multiple trains on multiple tracks, but doesn’t really know where to begin. Setting up the circuits are not as much of a challenge compared to the control.

To go along with the model trains, Willy likes to work on building wooden structures to fit in with the construction of the tracks. It would be nice for him to be able to combine whatever systems he learns to make with some sort of wooden housing to contain any circuitry. Being able to hide cables from his grandson is important, not just so that things don’t get ruined, but also to prevent accidents involving electricity.

Programming is not something that he has been previously introduced to. This is not so much because of a lack of interest, but rather he had no reason to learn about them. He realises this will be an issue for his ambitions of creating a train controller, but so long as he can get some quick introductions to the basics first, it should be solved with time.

8. Appendix

8.1 Hand Sketching

To discuss the layout of our app, we drew some rough sketches outlining how the UI will look and feel and the positioning of design elements. This gave us an idea of the direction we want our app to go in.

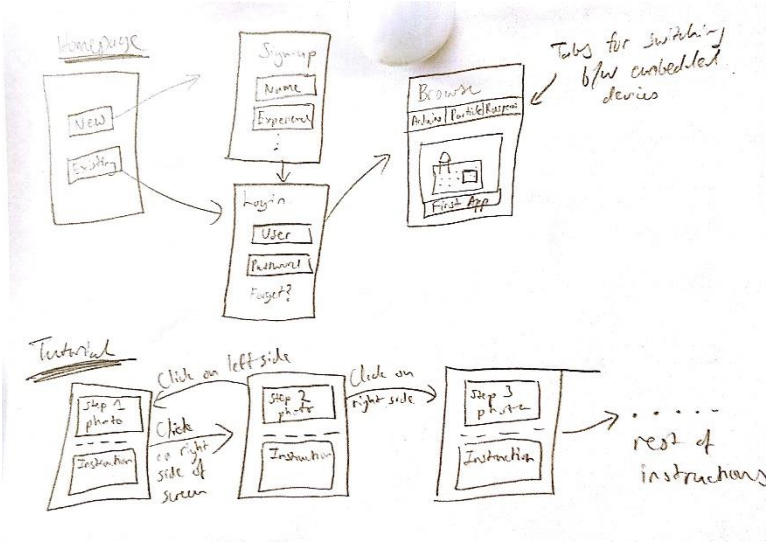


Figure 40 - Hand Sketch 1

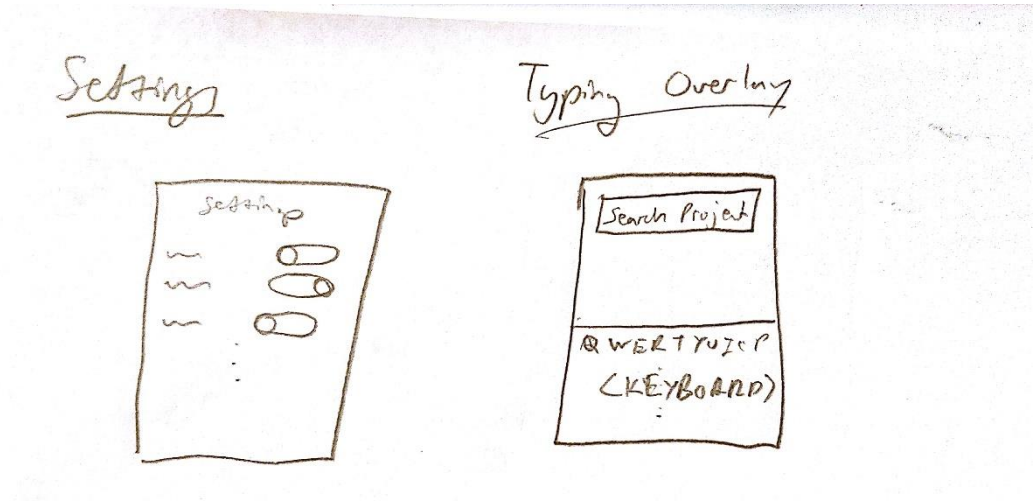


Figure 41 - Hand Sketch 2

8.2 Digital Sketches

Next, we digitised out sketches so we could have a clearer idea of how the interface of our app will look, however we abandoned using Microsoft Paint quickly and decided to just use Adobe XD.

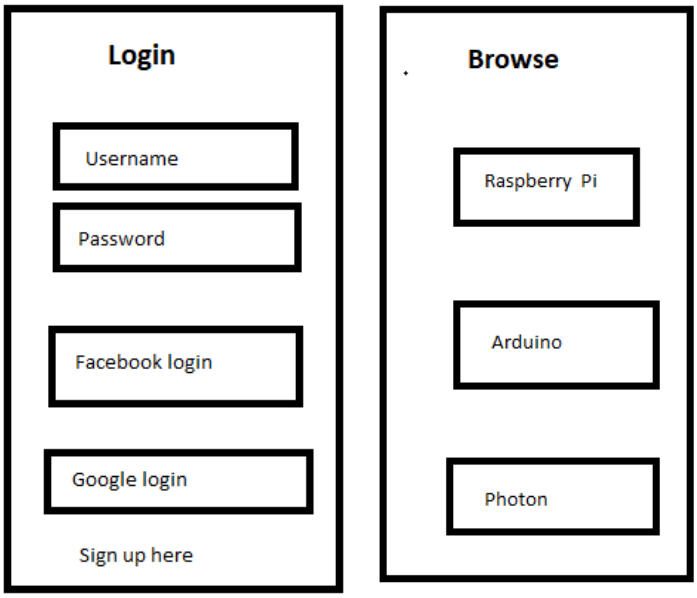


Figure 42 - Digital Sketches 1

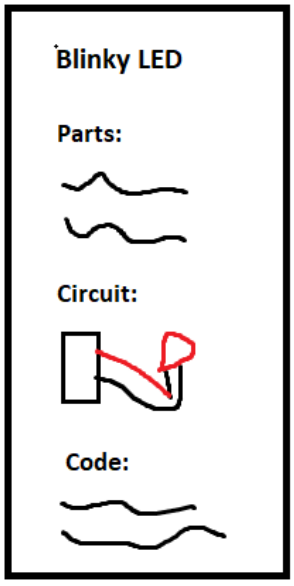


Figure 43 - Digital Sketches 2

8.3 Adobe XD Sketches

We moved to Adobe XD but manually drew all of our elements which was tedious and did not produce the desired result.

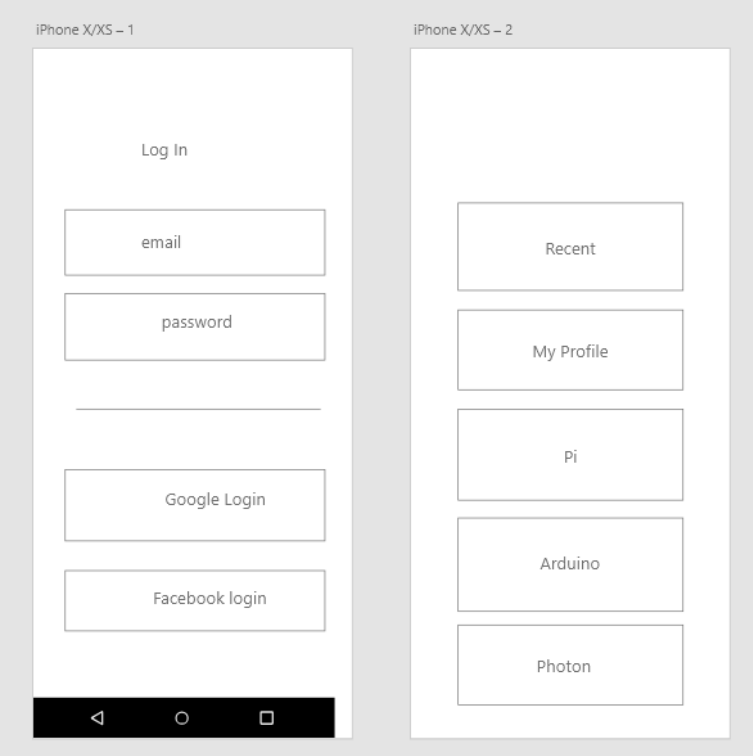


Figure 44 - Adobe XD Sketches 1

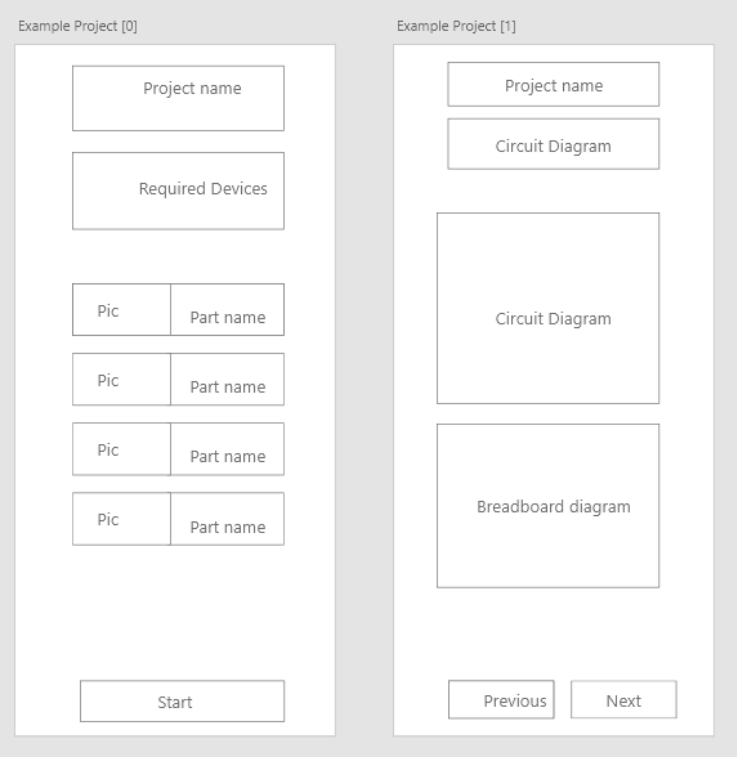


Figure 45 - Adobe XD Sketches 2

8.4 Google Material Design Stickersheet

We decided to use the elements from Google's Material Design stickersheet. This drastically enhanced the look of our interface as well as providing perfect consistency.

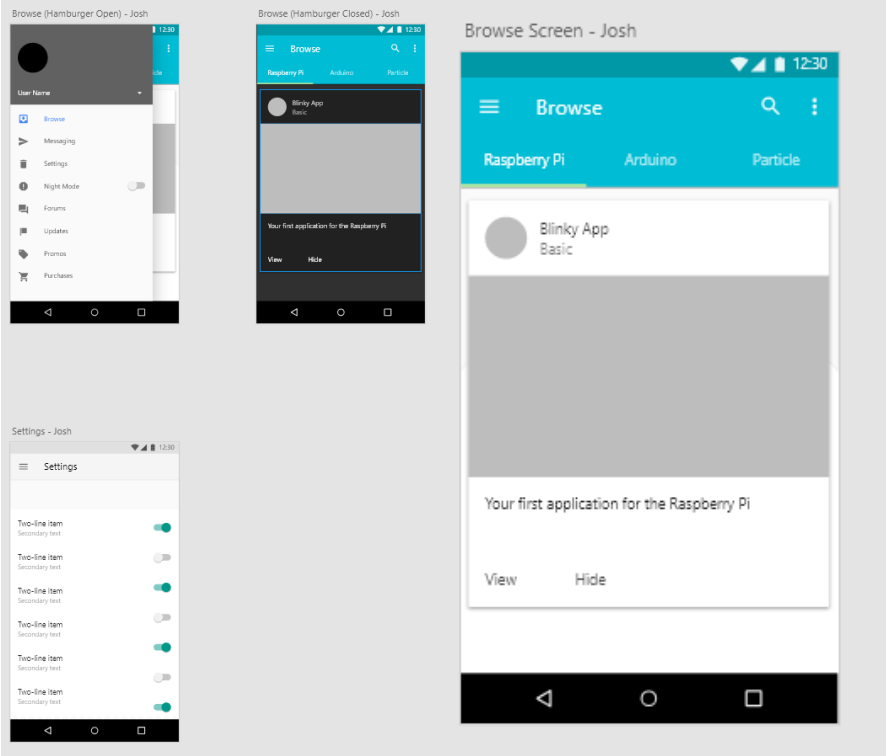


Figure 46 - Material Design 1

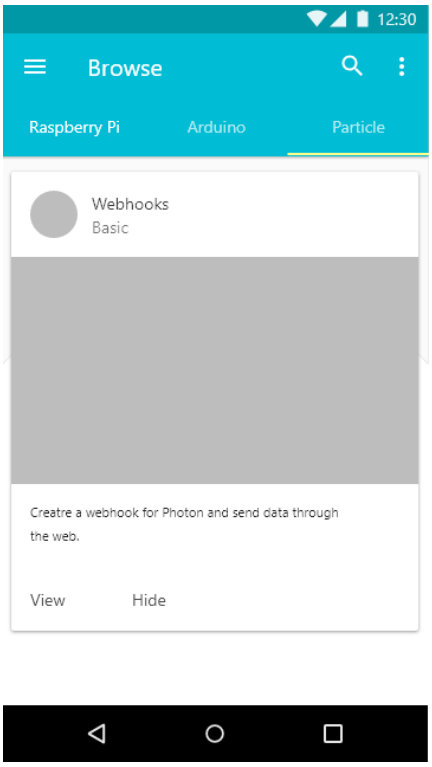


Figure 47 - Material Design 2

8.5 Incremental Changes

We started off with the default Material Design stickersheet colours:

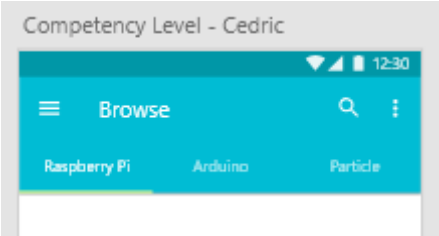


Figure 48 - Material Design Default Colour

We then decided to select a different colour scheme for our app, Serenity, Pantone Colour of the Year 2016.

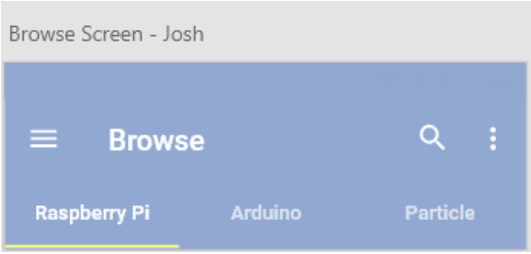


Figure 49 - Material Design Pantone Serenity

We started with the project pages with no features such as skipping steps or animations.

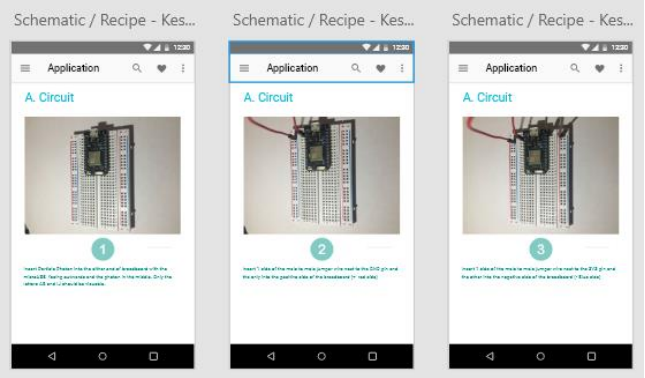


Figure 50 - Project Pages No Features

We then added those features in.

Figure 51 - Project Pages with Features

8.6 Storyboard Descriptions

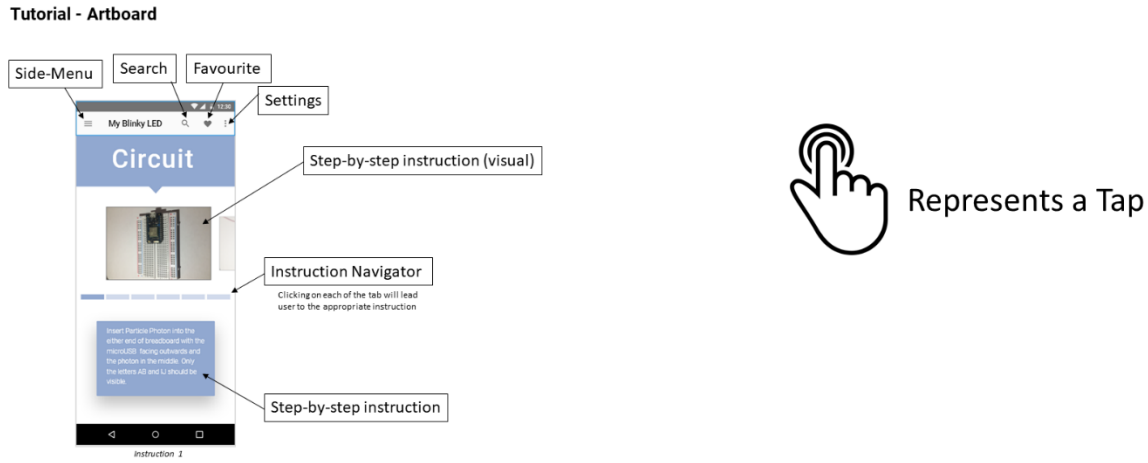


Figure 52 - Storyboard Description 1

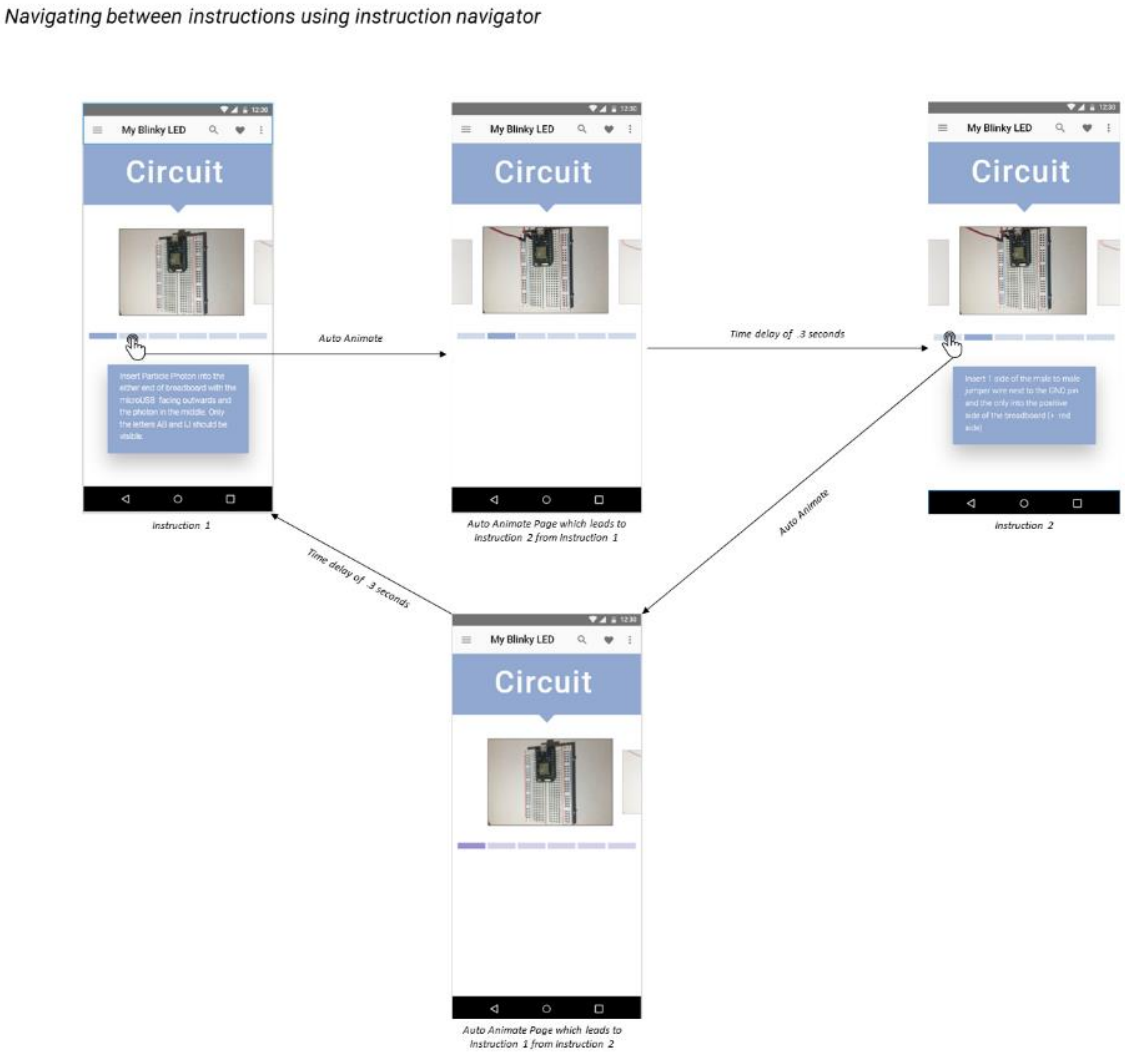


Figure 53 - Storyboard Description 2

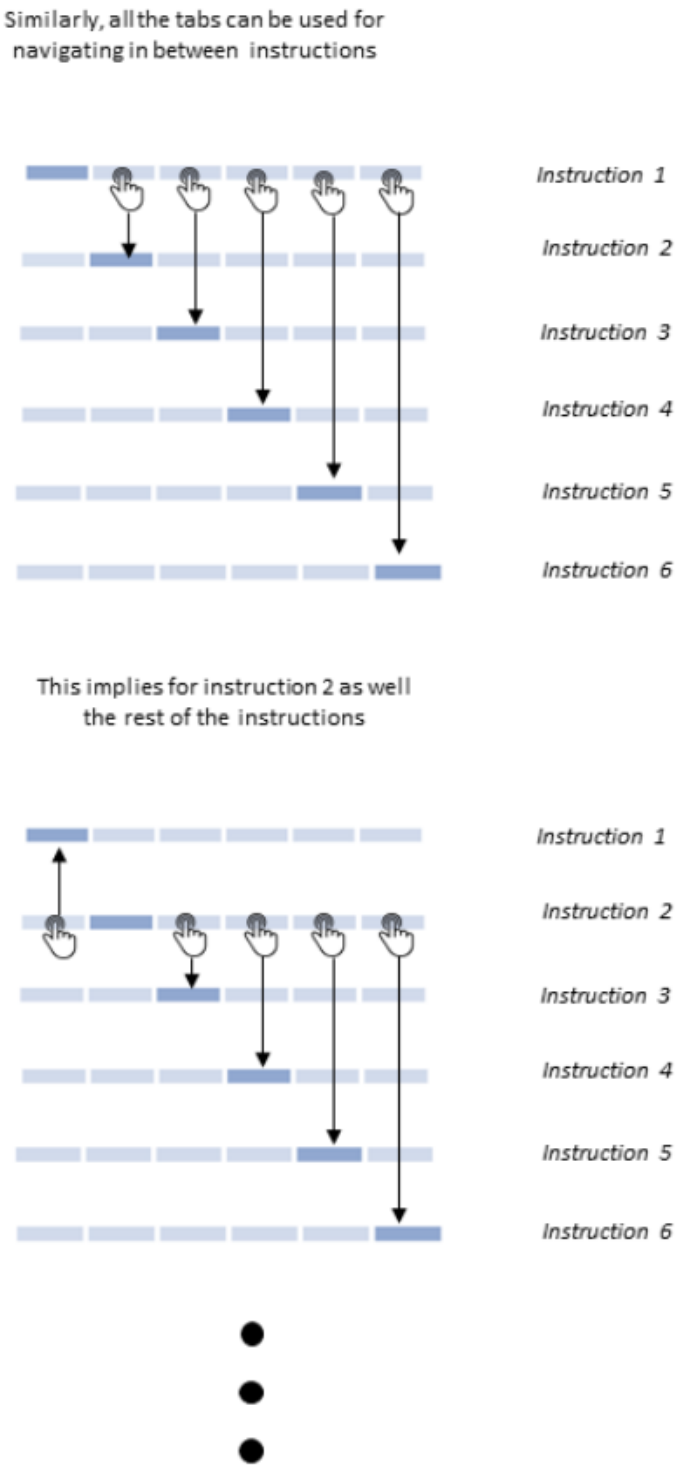


Figure 54 - Storyboard Description 3

Navigating between instructions by tapping on next/previous image of subsequent instruction

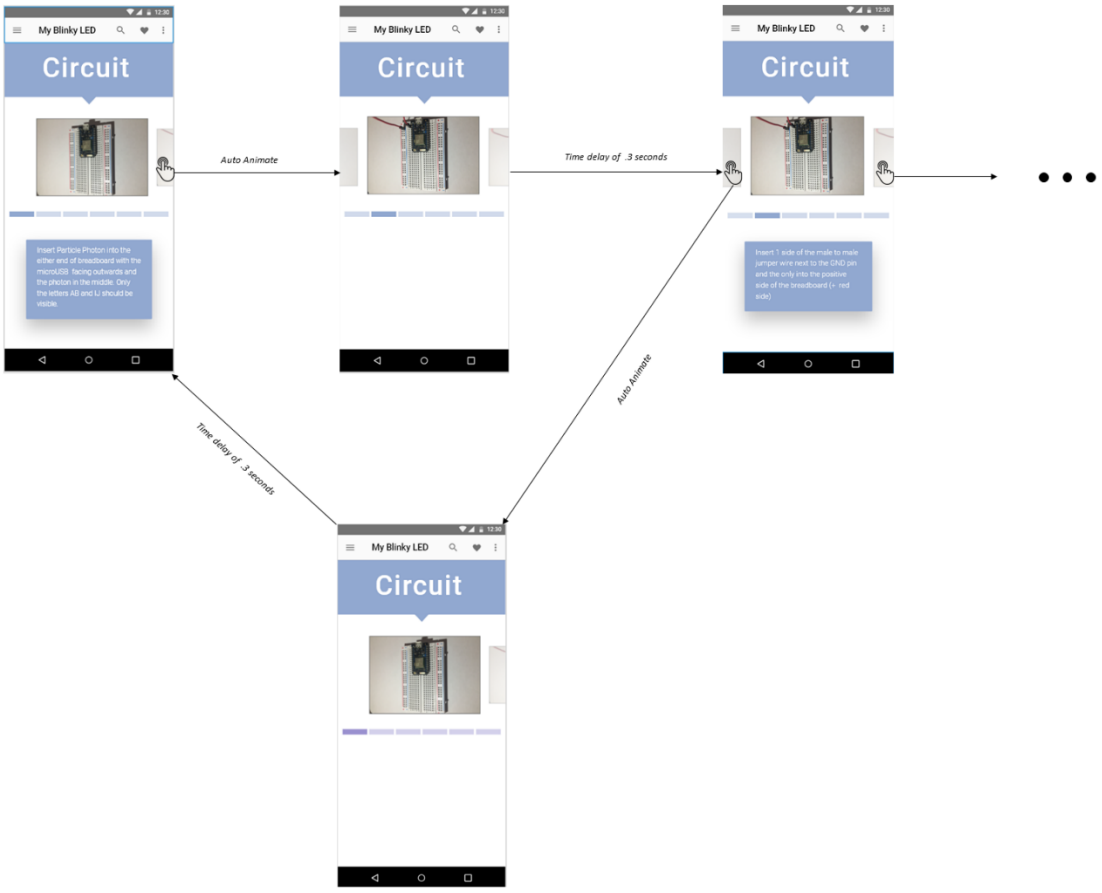
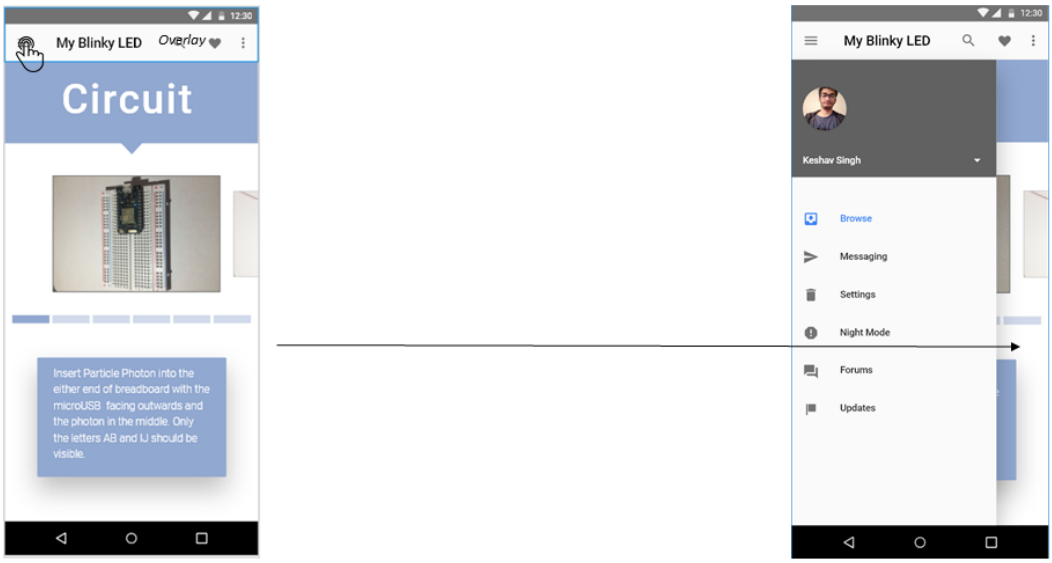


Figure 55 - Storyboard Description 4

Side Menu Interactions



Search Tool Interactions

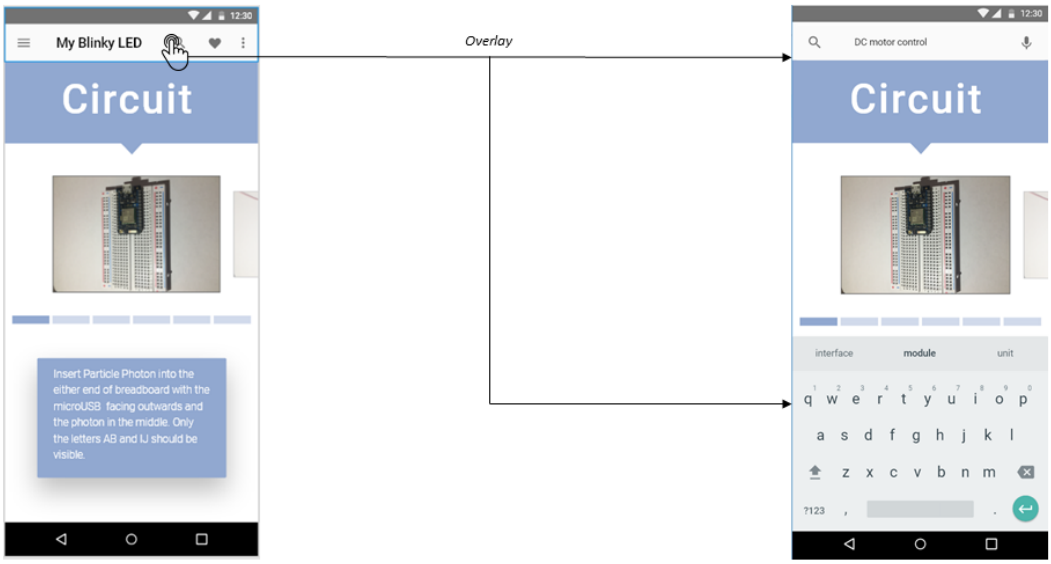
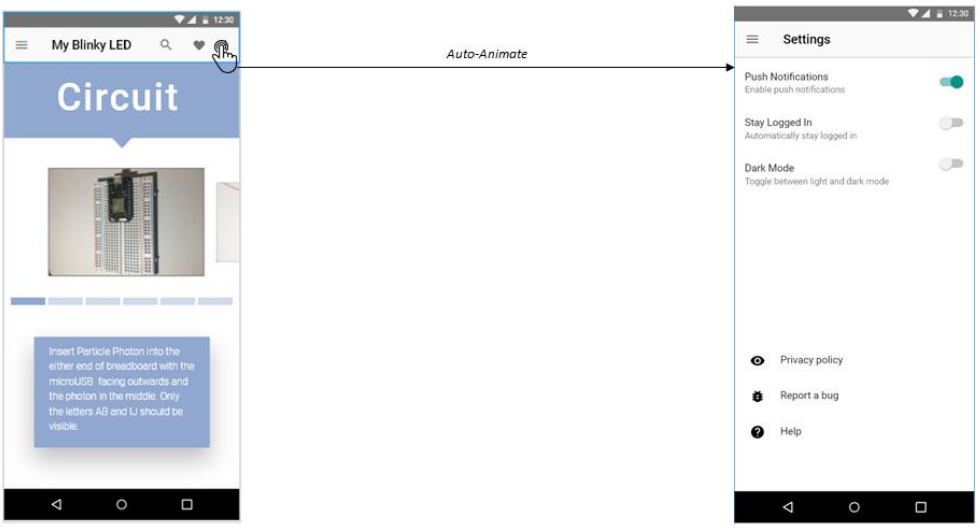


Figure 56 - Storyboard Description 5

Settings Interaction



Favourite Interaction

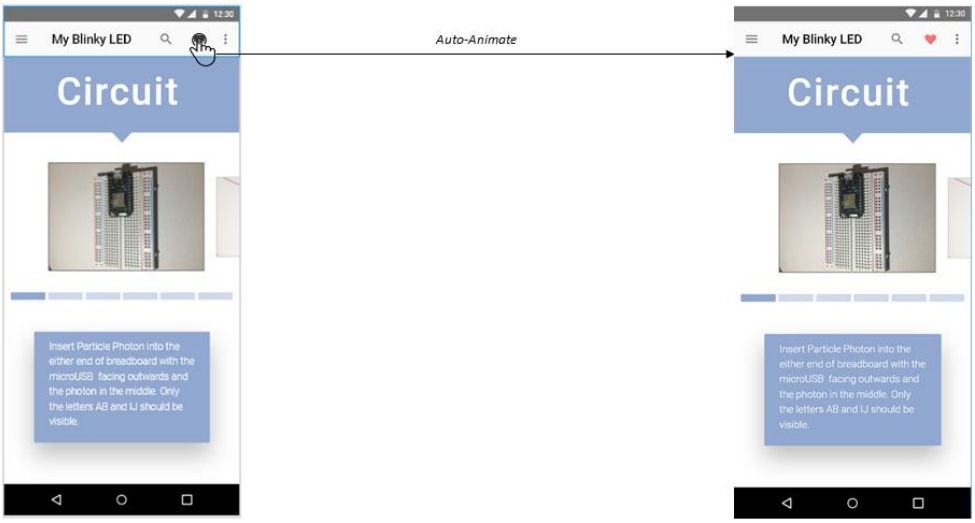


Figure 57 - Storyboard Description 6