

Final Documentation

EyeRS Development Team - ITSP200 (Deliverable 6)



Table of Contents

1. Group and customer information	5
2. System proposal	6
2.1. Overview	6
2.2. Aim	7
2.3. Objectives.....	7
2.4. User Requirements.....	8
Functional Requirements	8
Non-Functional Requirements.....	9
Technical Requirements	9
2.5. Hardware & Software Requirements.....	10
2.6. Project Schedule	14
Deliverable 1 Gantt chart	14
Deliverable 2 Gantt chart	15
Deliverable 3 Gantt chart	15
Deliverable 4 Gantt chart	16
Deliverable 5 Gantt chart (User Manual)	17
Deliverable 5 Gantt chart (Evaluation Report)	17
Deliverables Gantt chart.....	18
Full Detailed Gantt chart	19
2.7. Assumptions & Constraints.....	19
Scope	20
Time.....	21
Budget	22
Risk tolerance	23
Resources.....	24
Quality requirements.....	25
2.8. Feasibility Study	26
Operational Feasibility.....	27
Technical Feasibility.....	28
Economic Feasibility	28
Schedule Feasibility	29
2.9. Recommendations.....	29
3. Customer Sign-Off	30
Bibliography	31
Methodology	33
1. User Requirements Document	35
Introduction	35

1.1 Purpose	35
1.2 Scope	36
Benefits.....	38
Goals	38
1.3 Definitions, Acronyms and Abbreviations.....	39
1.4 Overview	43
2. System Requirements.....	44
2.1 System Perspective	44
2.2 Functional Requirements.....	45
2.1.1 Context Diagram (high-level DFD)	47
2.1.2 Entity Relationship Diagram (ERD)	48
2.3 Non-functional Requirements	49
Human-Computer Interactions (User Interfaces)	49
Security Requirements.....	49
Communications Interfaces.....	49
Hardware & Software Interfaces	50
2.4 Technical Requirements	57
2.5 User Characteristics	58
2.6 Operational Environment	59
3. Customer sign-off.....	60
Bibliography	61
Introduction	64
1. Information systems design.....	65
1.1. Logical design	65
1.1.1. EyeRS Entity Relationship Diagram	66
1.1.2. Context & Logical Process Model.....	66
1.2 Physical design.....	69
1.2.1 Investigation of technologies to be applied.....	70
1.2.2 System testing	72
1.2.3 System interface design	124
2. Customer sign-off.....	165
Appendices	166
Appendix A.....	166
Bibliography	167
Introduction	170
1. Information systems design.....	171
1.1. Logical design	171
1.1.1. EyeRS Entity Relationship Diagram	172

1.1.2. Context & Logical Process Model.....	172
1.2 Physical design.....	175
1.2.1 Investigation of technologies to be applied.....	176
1.2.2 System testing	178
1.2.3 System interface design.....	228
2. Customer sign-off.....	269
Appendices	270
Appendix A.....	270
Bibliography	271
Introduction	275
1. System test case results	276
2. Customer sign-off.....	333
Introduction	336
Getting Started.....	337
User Characteristics.....	337
Hardware & Software requirements	338
How to install eyeRS	340
How to.....	340
Register.....	340
Reset a forgotten PIN	340
Login	342
Add Item.....	343
Add a category	345
View the slideshow	346
Access help and tips.....	347
Access information about the app.....	348
Access the settings.....	349
Reset your PIN	350
Change Display Options.....	351
Change profile details.....	352
Configure sound settings.....	352
Sort the categories	353
Sort the items	355
Search.....	356
Exit.....	358
View items in a category.....	359
Edit a category	360
Delete a category	360

Edit an item	360
Delete an item	361
Share	361
Trade.....	361
Glossary.....	362
Acronyms and abbreviations.....	362
1. Evaluation Report	364
1.1 Introduction.....	364
1.2 Final System and Customer Requirements.....	365
1.3 Group Dynamics and Collaboration	366
1.4 Time Management.....	367
1.5 Lessons Learnt.....	368
1.6 Conclusion.....	370

1. Group and customer information

Group number	5
Group name	eyeRS Development Team
Members	Student number: M4DDK8SQ1 Name: Nathan Surname: Shava
	Student number: XQ9X3WV31 Name: Matthew Surname: Van der Bijl
	Student number: MB2015-0785 Name: Emilde Surname: Arsenio
	Student number: Z46WWQH76 Name: Andrea Surname: Cloete
	Student number: MB2015-0219 Name: Sajjaad Surname: Ishmail

Customer	Name: Ndai Mapaso Company: CTI Education Group Industry: Information Technology
-----------------	--

2. System proposal

2.1. Overview

Our company, eyeRS Development, was requested to design a mobile application. The mobile application is called eyeRS and will be designed by a team consisting of five members, namely:

- Nathan Shava;
- Matthew Van der Bijl;
- Emilde Arsenio;
- Sajjaad Ishmail; and;
- Andrea Cloete.

The reason why our company were appointed to develop this mobile application is because they noticed that multiple individuals in all age groups are finding it difficult to manage and organise their personal and professional lives.

The customer's name for whom the mobile application will be developed for is Ndai Mapaso, the lecturer of the Software Development Project 2 (ITSP200) module which forms part of the for BSc Information Technology course at CTI Education Group.

The solution to this problem mentioned above is to develop the eyeRS mobile application for Android operated devices. The application will be developed to run on this platform considering that the Nielsen Company (2016), proved that 53% of smartphone owners use the Android operating system. This will allow the application to target a larger market compared to any other mobile platform.

The application will assist individuals who need relief in their day to day lives. The mobile application will allow the user to catalogue their personal belongings and beloved items in an organised and easily accessible picture format. It will also enable the user to deal with admin, trading, sharing and the searching of personal possessions, all in one mobile application that can be accessed anytime and anywhere. The mobile application will provide the opportunity for its users to be connected to their possessions all the time.

Although there are similar applications like Closet+ and ONSIGHT, development of the eyeRS application will bring a new and improved way in the cataloguing of users' possessions. This will be achieved by implementing the objectives outlined in the requirements specification.

2.2. Aim

The proposed eyeRS mobile application (eyeRS app) is designed to allow users to catalogue their personal effects and beloved belongings while creating a platform to facilitate the trade of their items in the future.

2.3. Objectives

The objectives of the eyeRS mobile application are as follows:

- To deliver a successful and reliable personal cataloguing mobile application that will solve the day-to-day needs of its users by 23rd of October 2017;
- To develop a mobile application that will reduce the workforce required to execute the user's daily tasks;
- To reduce the complexity of completing all the functions that are provided by the mobile application by the 23rd of October 2017.
- To allow first time usage of the application to present users with an in-app tutorial on how to use and manage the application;
- To provide a user friendly application that will allow its users to easily navigate between different functionalities of the mobile application;
- To produce an affordable mobile application that will meet user's specific needs.
- To allow users to identify their items in a visual format (e.g via a picture);
- To allow users to state a short and descriptive note about each catalogue item;
- To develop a mobile application in an efficient and effective manner that is delivered on time; and
- The usage of the application can be done connecting to the internet as well as offline mode.



2.4. User Requirements

Functional Requirements

- The user must be able to add a new catalogue item to the application with the item's name, short description and an image taken using the built-in camera;
- The user should be able to add an image for a catalogue item from the device's local gallery;
- The user should have the ability to assign a PIN to lock the application for security purposes;
- The user should be able to connect the application to the internet in order to share catalogue data with other users via WhatsApp;
- A user should be able to share items on their catalogue via a unique QR code.
- The user would like to receive a notification/alert when another user has sent them a link for an item of interest that they wish to trade for;
- The user should be able to mark a catalogue item as being up for sale and share the information with other users via Bluetooth or externally via WhatsApp; and
- The user would want the application to backup data so as to allow recovery of their catalogue items should the application crash for any reason.

Non-Functional Requirements

- The user should be able to change the layout of the catalogue items (e.g. switching between grid/list view formats);
- The application should be available for use when needed (i.e. any time of the day);
- The response time of the application should be relatively fast to improve the overall experience and acceptance of the application;
- The application should store all local data in efficient manner;
- The application should have a user friendly interface where they can easily navigate between the different functions of the application;
- The user should have ability to give feedback and personal reviews of their experience with usage of the application or should they also encounter any challenges in which they would require assistance; and
- The user would like to have the ability of changing the colour scheme (i.e. theme) of the application.

Technical Requirements

- The user requires the application to make use of the external storage memory so as to avoid using up internal storage space;
- The application must make minimal use of mobile/network data;
- All data pertaining to the application must be stored on the local storage in the Android application data files (root\sdcard\Android\data);
- The application must operate efficiently without lagging when opening or running the application;
- The application must make use of *.json* files when storing local data;
- The mobile application is to support a minimum Operating System (OS) version of Android IceCreamSandwich (4.0.3) or higher;
- The mobile application should be suitable for any mobile screen size; and
- The application should consume minimal battery life.

2.5. Hardware & Software Requirements

The table below is a list of the minimum system specifications for developers to build the mobile application using the Microsoft Windows platform:

Hardware	Software
Desktop/Laptop processor requirements for accelerated emulator: <ul style="list-style-type: none">● 64-bit architecture; and● Intel processor with support for Intel VT-x, Intel EM64T (Intel 64), and Execute Disable (XD) Bit functionality.	Microsoft Windows 7/8/10 (32-bit or 64-bit).
QWERTY/Dvorak Keyboard & Optical mouse	Java Runtime Environment (JRE) 6 or higher and Java Development Kit (JDK) 8 or higher.
Network adapter with the following recommended specifications: <ul style="list-style-type: none">● Bus Type: PCI Express;● Industry standards: IEEE 802.11b/g/n;● Interface: Wireless/Ethernet; and● Maximum Data Transfer Rate: 802.11n: 300 Mbps.	Android Studio 2.2 or higher (1674 MB required disk space).
4 GB RAM minimum (1GB RAM for the Android Emulator and 8GB RAM recommended (Google, 2017).	Android Software Development Kit Manager (SDK Manager).
2 GB of available disk space minimum; and 4 GB Recommended (500 MB for IDE + 1.5 GB for Android Software Development Kit (SDK) and Android Virtual Devices (AVD) manager).	Android Virtual Device Manager (AVD Manager).
Desktop/Laptop display monitor with 1280x800 minimum screen resolution.	

The table below is a list of the minimum system specifications for developers to build the mobile application using the Linux platform:

Hardware	Software
Desktop/Laptop processor requirements for accelerated emulator: <ul style="list-style-type: none"> ● 64-bit distribution capable of running 32-bit applications; and ● Intel processor with support for Intel VT-x, Intel EM64T (Intel 64) and Execute Disable (XD) Bit functionality, or AMD processor with support for AMD Virtualization (AMD-V). 	GNOME/Unity or KDE desktop environment Tested on Ubuntu 12.04 LTS, Precise Pangolin (64-bit distribution capable of running 32-bit applications).
QWERTY/Dvorak Keyboard and Optical mouse.	GNU C Library (glibc) 2.19 or later
Network adapter with the following recommended specifications: <ul style="list-style-type: none"> ● Bus Type: PCI Express; ● Industry standards: IEEE 802.11b/g/n; ● Interface: Wireless/Ethernet; and ● Maximum Data Transfer Rate: 802.11n: 300 Mbps. 	Android Studio 2.2 or higher (1674 MB required disk space).
4 GB RAM minimum (1 GB for the Android Emulator (Android 2017)) and 8 GB RAM recommended.	Java Runtime Environment (JRE) 6 or higher and Java Development Kit (JDK) 8 or higher.
2 GB of available disk space minimum and 4 GB recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image).	Android Software Development Kit Manager (SDK Manager).
Desktop/Laptop display monitor with 1280x800 minimum screen resolution.	Android Virtual Device Manager (AVD Manager).

The table below is a list of the minimum mobile specifications for the customer and/or other users to run the mobile application:

Hardware	Software
Chipset: ARM-based or better.	OS Codename: Android IceCreamSandwich or better.
Memory: <ul style="list-style-type: none"> ● 750 MB RAM; and ● 2 GB Flash External; or better. 	OS Version: 4.0.x or higher.
Storage Type: <ul style="list-style-type: none"> ● Internal; ● Mini SD; or ● Micro SD. 	OS API Level: 15 or higher
Primary Display: <ul style="list-style-type: none"> ● QVGA TFT LCD or larger; and ● 16-bit/16M colours; or better. 	Browser: <ul style="list-style-type: none"> ● HTML; and ● Adobe Flash Lite or better.
Navigation Keys: <ul style="list-style-type: none"> ● 5-way navigation; ● 5 application keys; ● Power; ● Camera; and ● Volume controls; or better. 	Messaging: <ul style="list-style-type: none"> ● SMS; ● MMS; ● Email; and ● Push Email; or better.
Camera: 2 MP CMOS or better	
Sound: <ul style="list-style-type: none"> ● Loudspeaker; and ● 3.5mm audio jack. 	
USB: Standard microUSB v2.0 or better.	
Bluetooth: v1.2 or better.	
Network: <ul style="list-style-type: none"> ● (2G Bands) GSM 850 or better; ● (3G Bands) HSDPA 900 or better; ● 4G Bands LTE band 1(2100) or better; ● GPRS; 	

- 
- EDGE;
 - GPS; and
 - WLAN (Wi-Fi 802.11 b/g/n) or better.



2.6. Project Schedule

Deliverable 1 Gantt chart

Figure 1 below indicates the Gantt chart of the group and customer information and the system proposal that forms part of the deliverable 1.

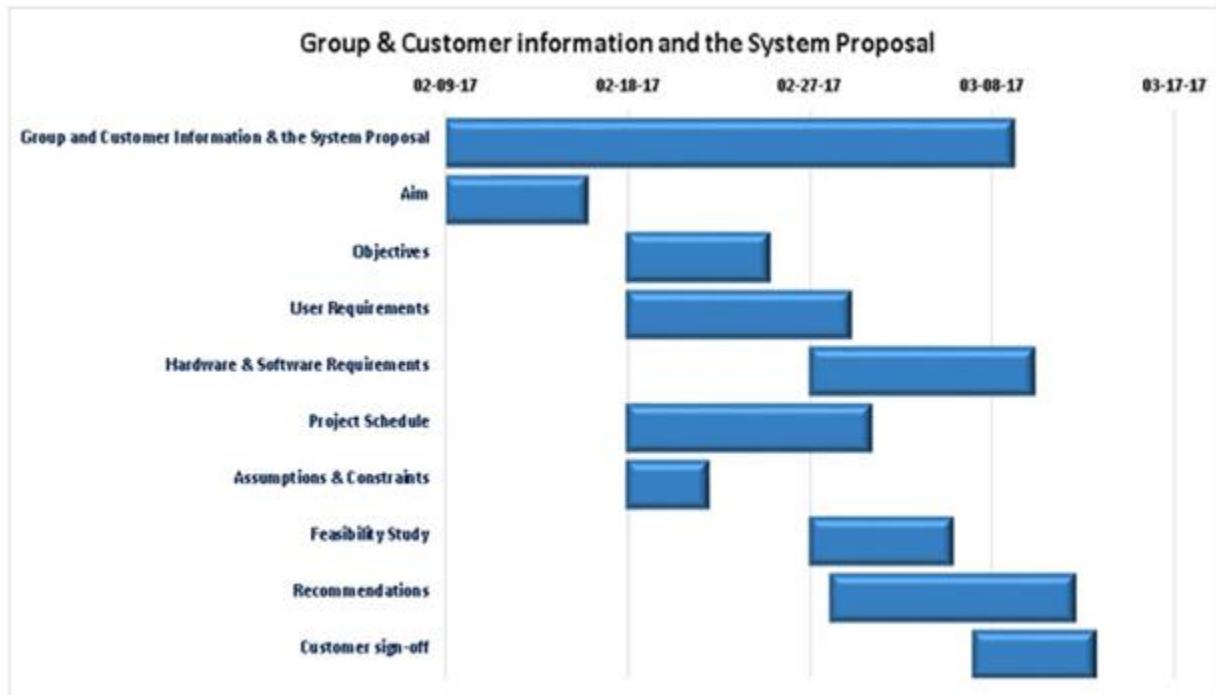


Figure 1

Deliverable 2 Gantt chart

Figure 2 below indicates the Gantt chart for the methodology and user requirements documentation that form part of deliverable 2.

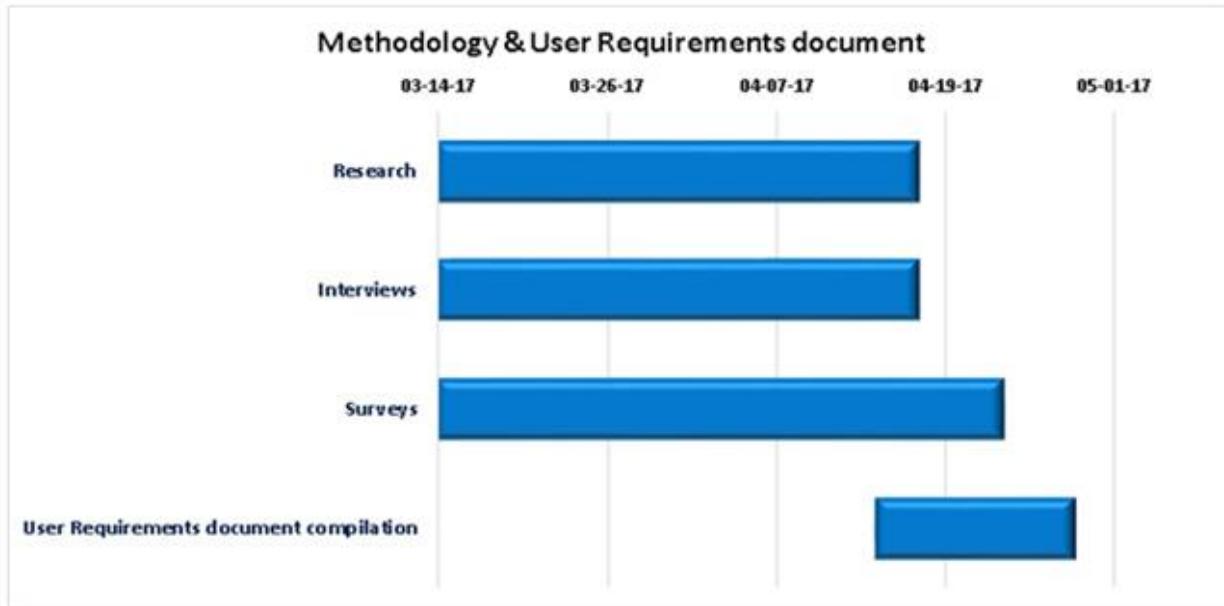


Figure 2

Deliverable 3 Gantt chart

Figure 3 below shows the Gantt chart for the system logical and physical design that forms part of deliverable 3.

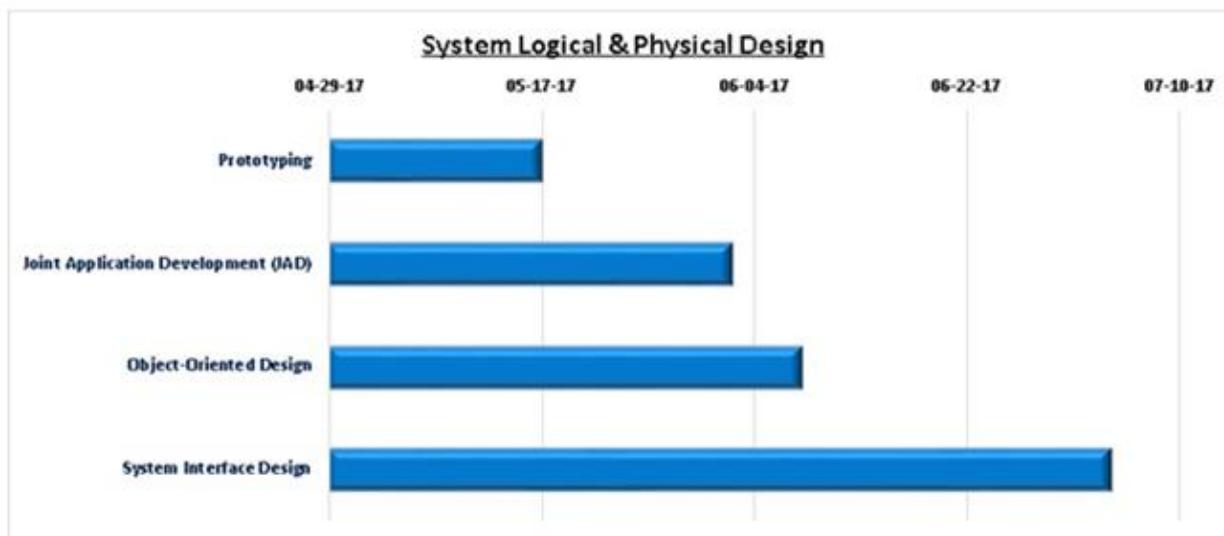


Figure 3

Deliverable 4 Gantt chart

Figure 4 below shows the Gantt chart for the system construction and testing that forms part of deliverable 3.

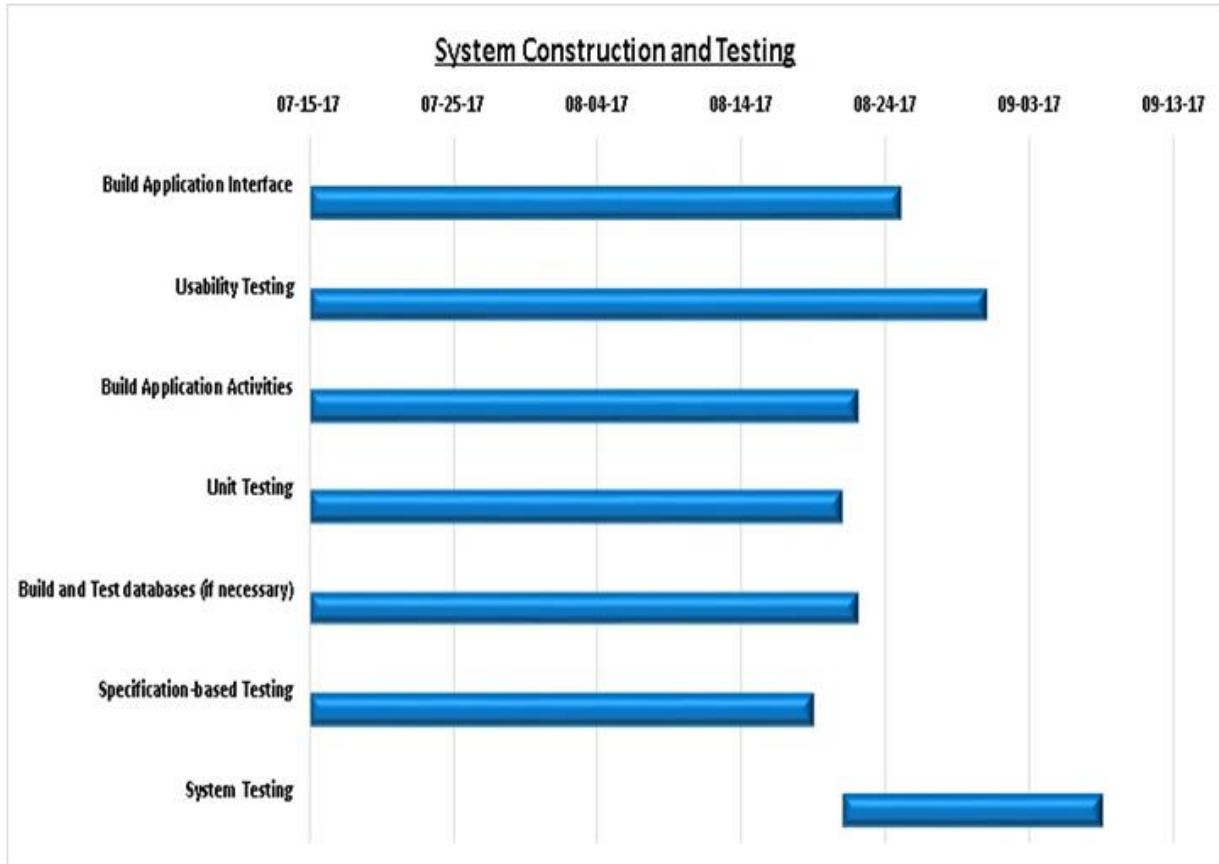


Figure 4

Deliverable 5 Gantt chart (User Manual)

Figure 5 shows the Gantt chart for the user Manual that forms part of deliverable 5.

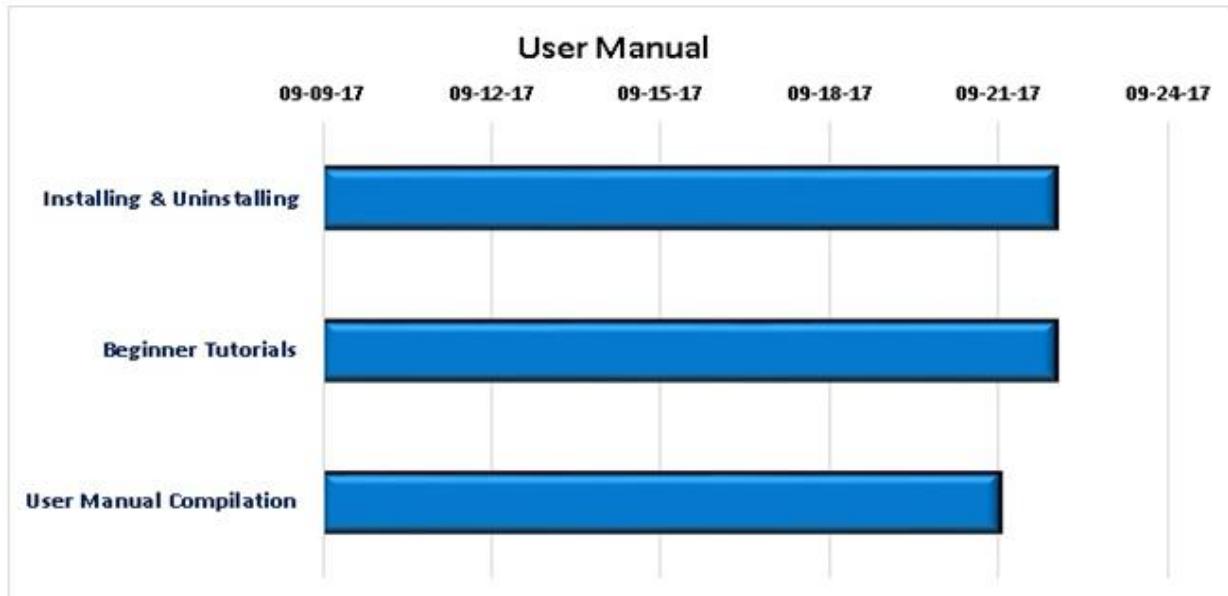


Figure 5

Deliverable 5 Gantt chart (Evaluation Report)

Figure 6 indicates the Gantt chart for the evaluation report that forms part of deliverable 5.

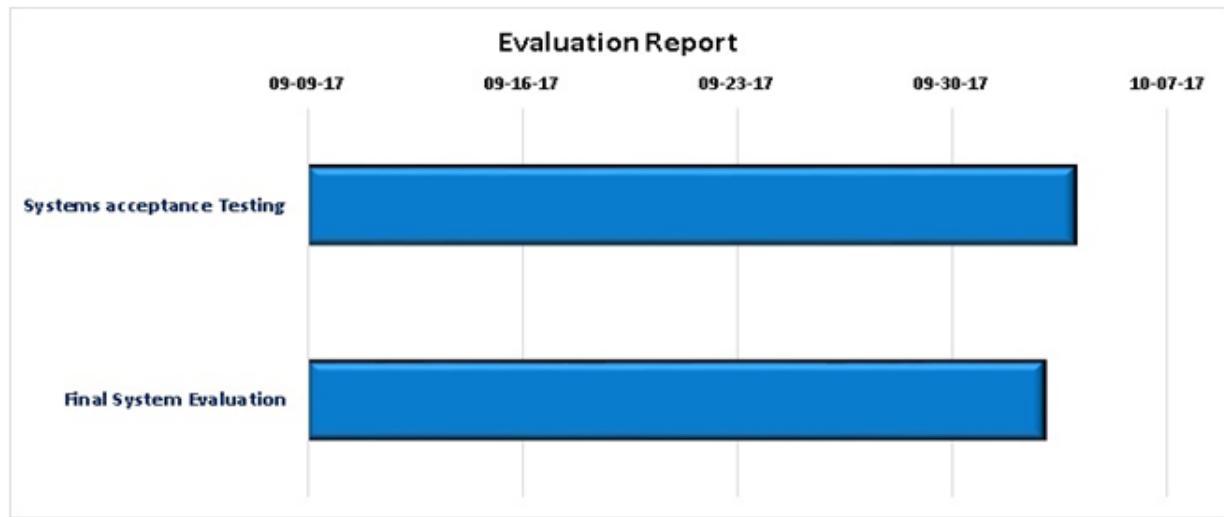


Figure 6

Deliverables Gantt chart

Figure 7 indicates the Gantt chart for the eyeRS Project Deliverables.

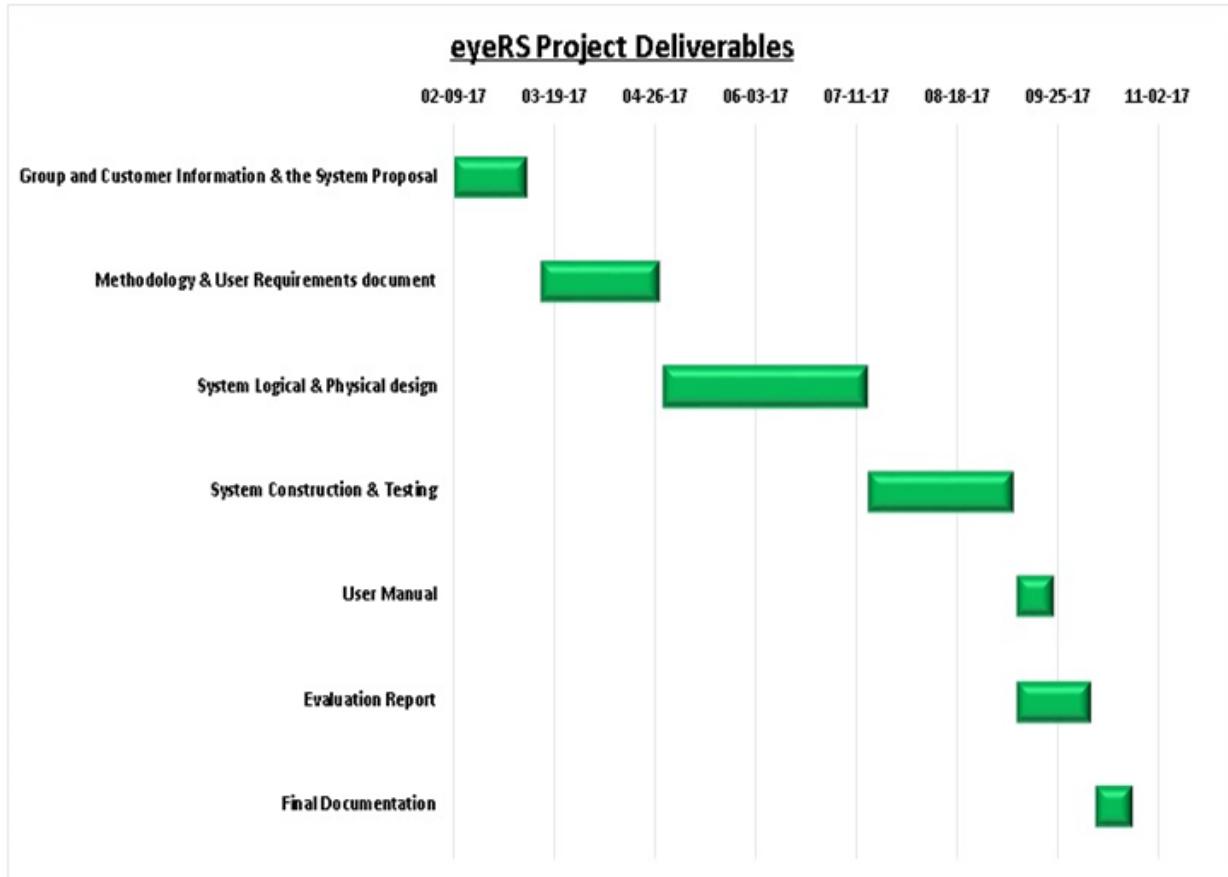
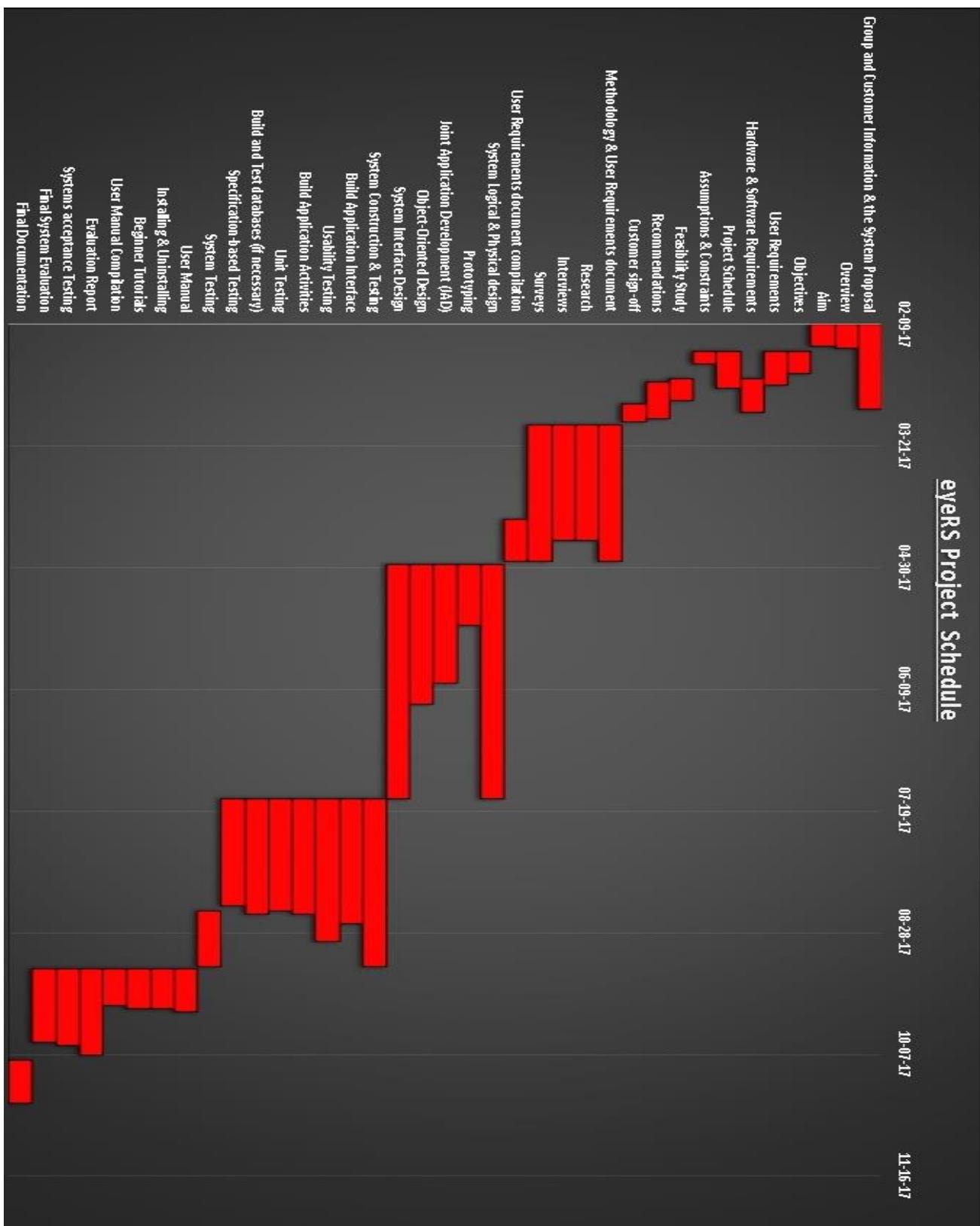


Figure 7



Full Detailed Gantt chart



2.7. Assumptions & Constraints



Scope

Assumption

The eyeRS mobile application will consist of an inventory for household items. The user is able to upload these items onto the application via the built in camera or a local file source such as the gallery. A bid or buy system will be implemented for users who wish to sell any unwanted items. All inventory that will be put up for sale will be available to all users that have downloaded the mobile application. Contact information will be made available for the interested party. Personal inventory can be shared amongst other users via WhatsApp or bluetooth capabilities.

Constraint

Limited skills and time consumptions will lead to a very limited scope for the eyeRS application (Buttrick, 2013). Support for social media login would have been an added benefit if implemented, however due to the limited time, this requirement may not be met. An in app chat system would be convenient for the purposes of providing a online means for users to offer trade/sale of their items with other users. Due to the limited experience of the development team with the required task, this requirement may not be met.

Linking the application an external database to store the data would be less proficient than making use of the built in android database. It would, however, be beneficial in terms of sharing information pertaining to the database. This requirement cannot be met as the development team will not be able to meet the budget requirements to implement this solution. Once the scope of the project has been defined, there will be no room to add further features and requirements to the project (scope/feature creep). Failure to manage this can result in the development team failing to meet all the requirements (Buttrick, 2013).



Time

Assumption

The project schedule will have a fixed deadline. The Deliverables for the project are all time-constrained, therefore the client will need to sign-off each deliverable by the set deadline in order to ensure that the project remains on schedule. Certain deliverables will be completed in less time due to their scope. This will afford the team more time for the other deliverables. Other deliverables may require more time than others to be completed. Good time management is therefore vital in order to delegate the extra time acquired from early completion to the deliverables with a larger scope. The success of the project will also be measured on whether it is completed and delivered on time (Buttrick, 2013).

Constraint

Preparing for unforeseen circumstances should be done during the planning phase due to the limited availability of time to deliver the application. Although one cannot always predict the outcomes for every situation. For example team member becomes ill and can no longer participate in the project indefinitely. A contingency plan would have need to be set in place. The initial set deadline, may not be met as a result. There is no leeway to be given in terms of the schedule being changed to push back the deadline if any of the deliverables are not completed on time.

The deadlines for each Deliverable are as follows:

- Deliverable 1: 17/03/2017;
- Deliverable 2: 12/05/2017;
- Deliverable 3: 11/08/2017;
- Deliverable 4: 15/08/2017;
- Deliverable 5: 13/10/2017;
- Deliverable 6: 27/10/2017; and
- Release date: 03/11/2017.

The project Deliverables all have deadlines in which the customer is to sign-off and approve. Failure to meet any of the Deliverable deadlines will cause the project schedule to be modified. This may lead to exhaustion of the total project time required to complete the project due to the fixed time constraints.



Budget

Assumption

The budget will depend on a number of factors. Github will charge for security of the eyeRS mobile application code. The free version offers less security, however this option is sufficient for the scale of the proposed project. From planning and research which include surveys, this could become a potential expense. The budget is affected directly by the time taken to complete the project. Since spending more time on the project would require more resources and pay for the employees. Resources are the biggest factor in the budget for any project.

Constraint

Due to a limited budget, the project team may need to trim down some of the project work if it leads to the budget getting overrun (Buttrick, 2013). The nature of the tools being used to develop the system will be available for free use. No further costs are to be incurred in terms of outsourcing from any third-party software companies as this would simply overrun the budget (Buttrick, 2013). The costs incurred by the project team should fall within the allocated budget. Most of the resources and facilities required to successfully come up with a fully functional mobile application have already been accounted for (Buttrick, 2013).



Risk tolerance

Assumption

A system would need to be put into place to accommodate for any unforeseen employee turnover or retirement. In the case that a team member should remain absent indefinitely a replacement would be required to increase productivity. In doing so this could indirectly affect the budget in terms that we would need to search externally for a replacement. Another case would be if the budget runs out. Proper planning would need to be carried out to avoid this from happening. Proper planning and research will need to be done in order to run a successful project. That is the only way to avoid having to go back to any previous phases in the waterfall model. If proper planning and implementation is done, that will minimise any problems in the future if not eliminate them completely.

Constraint

It is vital for the customer to accept the risks that may be encountered within the project from the initial stage of the application development phase. Project risks will directly impact the project Deliverables and these need to be managed and resolved with a high priority to avoid making the project unfeasible (Buttrick, 2013).

For High risks, the project team will need to attempt to prevent and avoid that risk from occurring. The contingency plan will automatically be put into action should a risk classified as High, occur (Buttrick, 2013). For Medium risks, the project team may need to consider transfer measures and the contingency plan may need to be implemented (Buttrick, 2013). For Low risks, the project team may need to simply monitor the risks incase they escalate and become more severe over time (Buttrick, 2013).



Resources

Assumption

In order for any project to become successful, resource management and allocation will need to be handled efficiently. The resources that will be available would need to be identified as early as possible from the resources that are readily available to those that need to be obtained at a later stage. Resources that are available would be the people that will partake in the project which are the team members. In this instance 5 members. Each member of the team has a field of knowledge that will be beneficial for the progress and completion of the project (Schwindt 2006). Other resources such as tools and equipment that is required are the computer's (hardware) that are needed for the development of the application. Programmes (software) are also required in the development of the application should include Android libraries and development kits as well as an Integrated Development Environment (IDE) in which to do the actual development work.

Constraint

Problems may arise if there is a lack of resources required in order to carry out the project tasks required for each Deliverable (Buttrick, 2013). This could refer to the computer-aided software tools required to develop the application prototype, user interface or the application itself. In order to ensure the project is completed up to the required scope level, certain resources need to be made available at all costs (Buttrick, 2013). Some of these resources may include:

- Sufficient manpower to undertake each project Deliverable;
- Facilities to enable the project team to work effectively; and
- The project tools and systems are relevant and compatible with the project team's workstations.

There is not a large constraint on the resources available for the project team as there is access to various open source freeware available for download on the internet in order to successfully complete the project Deliverables (Riddle & Fairley, 2012).

Quality requirements

Assumption

The quality of the system will be of a great standard since a detailed scope is understood by all the members of the system development team. The project will also have a high quality standard because of the time that has been divided into efficient deadlines that will be executed. The quality of the system will be of a high standard since cost efficient planning were conducted.

Constraint

The quality of the system might be affected by the availability of time. The time given to the system developers might be abused which will lead to a shortage of time, which will then result in a decrease in quality. Ways in which the time might be abused is poor planning, unexpected setbacks, a loss in one of the development team members, insufficient contribution from members in the development team or any other cause that might take up time that was not planned for (Haughey, 2011).

Since time is money it will also lead to an increase in the cost of the development of the system. The increased cost will lead to a decrease of quality since the system development project has a fixed budget and the extra cost for the time exceeded will be deducted from other areas in the budget planning (Haughey, 2011).

The use of more resources may have a positive influence on the overall system but it might have a negative effect on the cost of the system development project budget and also on the time spent on the project. It might also influence the system in negative manner. The quality of the system might also be affected when the scope of the system changes. These changes will also influence the time, cost and resources used in the development of the application.

2.8. Feasibility Study

The feasibility study below will compare the proposed eyeRS mobile application against two pre-existing similar programs, namely:

- Closet+ (<http://closetapp.com>); and
- ONSIGHT (<https://www.onsightapp.com/mobile-product-catalogue-app>).

Both pre-existing applications are complete systems that have been published and are still in use.

Closet+ is a automated style assitant developed for Apple's iPhone that is designed to aid users in organising, cataloging and planning their wardrobes developed by MY/STATIC/SELF (MY/STATIC/SELF, 2017).

Onsight is a mobile catalog application designed for wholesalers, manufacturers and distributors to replace their paper catalogues (Onsight, 2016). Onsight's website (2016) indicates that they supports various mobile platforms, including: Android devices, Windows tablets and Apple iPads. The app's main purpose is for the salesmen to be able to show potential clients with descriptions and images about a desired product.

Operational Feasibility

A operational feasibility study involves an evaluation to determine how well a system has been proposed and whether its proposed operations are feasible, as well as how the system takes advantage of available market opportunities and satisfies the user's needs (Valacich *et al.*, 2015; Katimuneetorn, 2008).

	Closet+	ONSIGHT
Performance	According reviews posted to the Apple App store about Closet+ is plagued with bugs and other glitches.	It performs very well according to their scope in which is an virtual advertising company based on pictures. It requires internet to work, affecting your battery life and costs (Onsight, 2016).
Information	Closet+ provides users with a grid base to provide users with a gesture-driven user interface.	It is not a system that any user would know what to do at the first place. It works in par with Windows as an admin and the mobile as users or employees. It is a one-to-many situation, Admin to employees. Windows system to mobile systems.
Control	Due to the natures of data that the app stores and the encryption standards associate with the Apple iPhone there is little risk of security issues. In addition to this Closet+ has built-in tools to backup and restore user data in the event of an error .	Nowadays safety is one of the strongest talking point when developing an system, due to data storage and payments. Onsight is exposed to any threat because its nature is data storage and payments online (Onsight, 2016).
Services	Due to the number of reported bugs present in the current build of Closet+ does not provide a reliable service to the user. As the app is designed for the cataloging of clothing alone it limits users to the clothing and other accessories though the system core system can be expanded upon greatly.	Due to its magnitude Onsight provides all different user guides in case of any uncertainty. Its' expansion depends on user's budget; package offers different features.

The implementation of the system will achieve the objectives set in the proposal's objectives. As indicated by the existence of other similar systems it is feasible to construct from an operational point of view. Ultimately, if the application is constructed and implemented correctly, the need outlined in this proposal can be met.

Technical Feasibility

A technical feasibility study is a process of analysing the logical aspect of a business operation to determine whether the organisation has the ability of constructing the proposed system (Valacich *et al.*, 2015). It involves assessing the development team's ability to physically construct a proposed system.

All the required technology outlined in the hardware and software requirements currently exist and the construction of the proposed system is feasible within current technology limits. eyeRS development process will be using different systems to maximise time efficiency as well as quick response in terms of solutions towards the final system in order to fully develop and launch the proposed system within the specified budget and schedule. There are already different systems in the market but we want to create something simple, trustworthy, friendly and affordable. We are fully capable to develop such system and succeed in different phases of the project development. Our team is equipped with different skillsets and roles during the application's development.

Ultimately it is feasible to construct the system from a technological standpoint. All software required to develop the system already exists and no new hardware components need to be produced to support the proposed system. The development team can be trained in the usage of the new technology and thus the development of the project does not need to be outsourced. The eyeRS development team has the ability to develop the proposed system.

Economic Feasibility

According to Valacich *et al.* (2015) an economic feasibility study is a process of determining whether the cost and financial benefits associated with developing a project are reasonable. The development team is well versed in the use of all software required and thus no external training is required and the creation of the project will not need to be outsourced.

Closet+ has a free version with limited features available for download with a full version available for \$2.99 (MY/STATIC/SELF, 2009). As Closet+ is free, a sales model where limited features are available for the free build of the software, it does not offer all the resources available until it has been purchased.

As any system in the market whose aim is to make profit and help people, they always offer the trial version for less than 30 days. This version comes with basic functionality. It is not a cheap system to buy it currently vary from R157+VAT to R5226+VAT per month depending on the package the user selects (Onsight, 2016).

The creation of the project is justified and it can be completed within the current cost and time constraints. The existing technology is cost effective. Once the project has been published there should be no additional maintenance costs. All data will be stored locally, therefore, no additional network costs will be incurred. The benefits of developing the system rather than outsourcing the development of the system ultimately outweighs the potential costs of developing the project.

Schedule Feasibility

Considering the time given and guidance from the gantt chart, our group will be able to achieve the different objectives in the system development. Our group is highly skilled in different sectors in the system development, which brings efficiency as well as a quick pace to the system development.

Closet+ is an application that has been in the market since the 14th April 2009; This was the launch of version 1.0 of the app. Different versions were created to improve the application as well as change the name of the application. From the 23rd of October 2009 to 31st of August 2010 the application was updated several times with different versions, from 1.3 through to 1.4.2 version where bugs were fixed and additional functionality to the app was added.

The application is continuously changing due to bugs, changes in the user's necessities and platform updates. Various versions have been launched in order to satisfy the user's needs and wants and improve the overall quality of service provided. On the December 12th of 2013 the application started to be called "Closet+" from closet 1.x; changes were made as the icon and updated design for iOS 7. The last update was made on the 2nd November 2015 where the developer updated the application to allow usability on the latest iPhone model (MY/STATIC/SELF, 2017).

2.9. Recommendations

In-house development has been recommended to the customer as it is ideally suited for the development of project. Communication within the development team is vital to the success of the project and would ultimately improve the overall productivity of the team as a whole. In-house development with a small team would facilitate this level communication.

If any team member is unable to contribute to the development of the project, a replacement would need to be found. Though this may affect the moral of the team it may be needed to make up for the loss of manpower. However, it all comes down to the budget and time prescribed initially.

The development of the project does not have to be outsourced as the team can be trained in the usage of the new technology and all tools that will be required for the development of the project. Ultimately, is-house development has been recommended as its benefits outway its costs.



3. Customer Sign-Off

<u>Customer Name & Surname</u>	<u>Customer Signature</u>
<u>Group Leader Name & Surname</u>	<u>Group Leader Name & Surname</u>

Bibliography

- Android, 2017. Download Android Studio and SDK Tools | Android Studio. *Android Studio*. Available at: <https://developer.android.com/studio/index.html> [Accessed March 1, 2017].
- Buttrick, R., 2013. *The Project Workout: The ultimate handbook of project and programme management*, Pearson UK.
- Haughey, D., 2011. Understanding the Project Management Triple Constraint. *Project Smart*. Available at: <https://www.projectsmart.co.uk/understanding-the-project-management-triple-constraint.php> [Accessed February 21, 2017].
- Katimuneetorn, P., 2008. Feasibility Study for Information System Projects. Available at: http://www.umsl.edu/~sauterv/analysis/F08papers/Katimuneetorn_Feasibility_Study.html [Accessed March 1, 2017].
- MY/STATIC/SELF, 2009. Closet+ on the App Store. *Apple App Store*. Available at: <https://itunes.apple.com/app/id309532414> [Accessed February 17, 2017].
- MY/STATIC/SELF, 2017. Closet+ / The Swiss Army Knife of iOS Style Assistants. Available at: <http://closetapp.com> [Accessed February 16, 2017].
- Onsight, 2016. Mobile Product Catalogue App - Onsight. Available at: <https://www.onsightapp.com/mobile-product-catalogue-app> [Accessed February 16, 2017].
- Riddle, W.E. & Fairley, R.E., 2012. *Software Development Tools*, Springer Science & Business Media.
- Schwindt, C., 2006. *Resource Allocation in Project Management*, Springer Science & Business Media.
- Valacich, J.S., George, J.F. & Hoffer, J.A., 2015. *Essentials of Systems Analysis and Design, Global Edition*,



Project Title: eyeRS

08/02/2017 - 20/10/2017



Methodology

The chosen methodology for the system development is the waterfall model as presented by Stair and Reynolds (2008). This model is used to develop a system in a sequential and linear way. The waterfall model allows development to be completed in a steady and downward fashion. A steady and downwards fashion means that each phase of the model is completed before moving to the next phase. The end result of each phase serves as input for the phase that follows. The phases of the waterfall model (Stair & Reynolds, 2008) that will be discussed are:

1. Investigation;
2. Analysis;
3. Design;
4. Construction;
5. Integration and testing and
6. Implementation.

During the investigation phase developers gather all possible information on the requirements that the product requires. The second phase is the analysis phase. During this phase allow developers gain a better understanding of the functions, problems and opportunities of the system being analysed (Rob & Coronel, 2004).

System design is the third phase. During this the design of the system is completed and structure. This includes technical specifications and requirements (Rob & Coronel, 2004).

The fourth phase is the construction phase. During this phase the identified requirements are implemented within the chosen system. The construction phase includes turning abstract ideas into a reality by coding the ideas (Rob & Coronel, 2004).

The fifth last phase is the phase of integration and testing. During this phase the units from the previous phases are integrated (Rob & Coronel 2004). Tests are performed to locate any faults (Rob & Coronel, 2004).

The final phase of the waterfall method is the implementation phase. During this phase the developed system is installed. After the system has been installed it needs to be tested and debugged until the system are ready for the customer (Rob & Coronel 2004).

The diagram below is a depiction of the Waterfall model presented by Stair and Reynolds (2008).

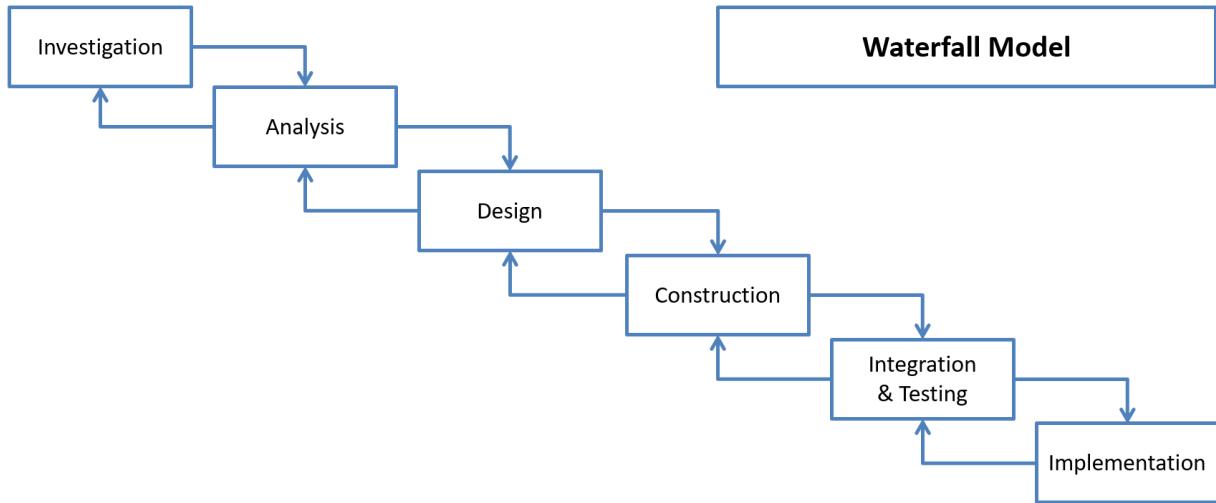


Diagram 1 Depiction of the Waterfall model (Stair & Reynolds 2008)

As can be seen in the diagram above, at the end of each stage a review is conducted to ensure that the requirements are met to a desired standard and quality. According to Sommerville (2011) once a phase has been completed it should be documented and evaluated. The next phase in the model should not be started until the current phase has been evaluated and approved (Sommerville 2011).

1. User Requirements Document

Introduction

This deliverable presents the user requirements document. This will describe the purpose, scope, definitions, acronyms, abbreviations and overview of the project. It also includes the system requirements that need to be focus on. The system requirements will contain the system perspective, requirements, user characteristics and operational environment.

1.1 Purpose

The proposed eyeRS mobile application (eyeRS app) is designed to allow users to catalogue their personal effects and beloved belongings while creating a platform to facilitate the trade of their items in the future. The eyeRS app is a Management Information System (MIS). A MIS is systems that use data to generate a report for the enduser to make routine decisions in response to a daily problem (Cress, 2017). In the case of the eyeRS app a summarised report is created for each item that the user can add to the system.

The purpose of the eyeRS app is to give individuals the power to manage and organise their personal and professional lives with less workforce. The eyeRS app achieves this control by allowing users to freely access and catalogue all of their belongings and items that are uploaded to the app. Upload can be done anywhere, on both an android mobile device. Each belonging is saved by means of photo identification and a short description.

The app allows users to be connected with friends and family. The connection is achieved by allowing users to share and trade items on their catalogues. The app also allows a user to be connected with their personal belongings anywhere.

1.2 Scope

The eyeRS mobile application consists of an inventory for household items. A user is able to upload these items onto the application via the built in camera or a local file source such as the gallery. A bid or buy system is implemented for users who wish to sell any unwanted items. All items that are put up for sale is available to all users that have downloaded the mobile application. Contact information is made available interested parties. Personal inventories can be shared with other users via social media WhatsApp or by means of Bluetooth connections.

Table 1 - Supported features

Supported Feature	Description
Single device support	The app will only allow users to make use of the app on a single device.
Tips & Help	When installing the app, a user will be shown a video tutorial on the basic features of the app. The user will have the option to refer back to this video should they require to do so. Onscreen help is available to assist users with navigating between features.
New category	The app allows user to create personal categories, which may differ from the prebuilt types.
Update existing data	The app allows users to make changes to existing items in their catalogue e.g. name changes.
Image capturing using built-in camera	The app makes use of a camera to capture images of new catalogue items.
Image gallery uploads	The app allows a retrieval of images from a user's personal gallery.
Bluetooth file share	The app allows users to share a QR code with another user for an item in their catalogue. Sharing can occur via a third-party messaging app or via Bluetooth.
File deletion	Users are able to delete existing category or catalogue items. There must always be at least one category by default however.

Item trade	The app allows users to trade their catalogue items with other users by sending them a link to the item on offer.
Add new item	The app allows users to add a new catalogue item, specifying its name, its description, and capturing an image of the item using the built-in-camera.
Third-party app interaction	The app can use other third-party apps to carry out certain actions within the bounds of the user-specified intent.

Table 2 - Unsupported features

Unsupported Feature	Description
Multi device support	Due to the project's time constraint, this feature is not be supported in the primary release. It may be available as a future update.
Revert changes	A user will not be able to revert back to their previous settings once changes have been made.
Video recording using built-in camera	Video recording will not be used in the app. The app will only allow the user to capture still images while using the camera feature.
NFC file share	NFC will not be utilised, as a sharing functionality is not loaded onto all mobile devices that support this technology.
File back-ups	The app will not be connected to any online database server, all data will be stored on the user's device.
Group messaging	The app will not allow a user to create a multi-user chat dialog.



Benefits

1. This is a reliable cataloguing app that will enable its users to manage all their belongings in a visual format;
2. Users will have access to the app anytime of the day;
3. The app can be used while connected to the internet or in offline mode;
4. The app will consume minimal battery life while in offline mode;
5. The app will be available as a free download from the Google Play Store; and
6. Users will be able to share offers with other users for items they want to trade.

Goals

1. The eyeRS team intends developing a user-friendly app that will enable the user to manage their personal belongings in a visual format;
2. The app will allow the user to be able to access the app whether they have Internet access or not;
3. The app will target a wide range of users with different backgrounds regardless of their demographics, culture, location;
4. A prototype for the app will be designed for use on any computer aided software tool.
The purpose of the prototype would be to get feedback from customers with regards to the look and feel of the proposed app before any development takes place;
5. The eyeRS development team will deliver a fully functional app that will be deployed on the Google Play Store;
6. The app will be available as a free download from the Google Play Store;
7. The user will be able to send app feedback to the development team in the form of ratings or written reviews; and
8. The user will be able to make suggestions of potential features to upgrade or fix (in case bugs are encountered).

1.3 Definitions, Acronyms and Abbreviations

Table 3 - Definitions used in the document

Definitions	
Term	Definition
Android OS	Is an operating system developed by Google design for mobiles cell phones to easily provide real actions as swiping, tapping and pinching to manipulate objects on screen (Rouse 2017a).
Catalog	A list or record of items systematically arranged and often including descriptive material.
Debugging	A process of detecting and removing errors and defects from a computer program (Hope, 2017).
Deliverable	Something that can be done and provided to a customer, especially something that is a realistic expectation (Martin & Tate, 2002).
Entity	A being or existence, a distinct, independent or self-contained object (Eastman, 1999).
Feedback	The process of returning part of the output of a device to the input, either to oppose the input (negative feedback) or to aid the input (positive feedback) (Coach 2015).
Functionality	A capability in which a device it is able to perform (Bass <i>et al.</i> , 2012).
Intent	With regards to Android development, an intent facilitates the performs a late runtime binding between different applications (Android, 2017).
Interface	Computer hardware or software designed to communicate information between hardware devices, between software programs, between devices and programs or between a device and a user (Sonmez, 2010).

Javadoc	A documentation generator created for generating API documentation in HTML format from Java source code.
Mobile Application	A portable computer program used for a particular type of job or purpose(Rouse 2017b).
Mobile Device	A portable, wireless computing device that is small enough to be used while held in the hand/handheld (Viswanathan 2017).
Prototype	The model on which a later product is based or formed (Rouse, 2017).
Waterfall Model	It is a progressive design process used in software development process where the process flows downwards through multiple phases (Rob & Coronel 2004).

Table 4 - Acronyms used in the document

Acronyms and Abbreviations	
Acronyms/Abbreviation	Expanded
3G	Third Generation
ADK	Android Development Kit
ADT	Android Development Tools
API	Application Programming Interface
App	Application
ARM	Advanced RISC (Reduced Instruction Set Computer) Machines
CPU	Central Processing Unit
DDMS	Dalvik Debug Monitor Server
DFD	Data Flow Diagram
DPI	Dots Per Inch
ERD	Entity Relationship Diagram
GPU	Graphics Processing Unit
HCI	Human Computer Interaction
HD	High Definition
HPROF	Heap/CPU Profiling Tool
HTML	Hyper Text Markup Language
IDE	integrated Development Environment
IO	Input/Output
JDK	Java SE (Special Edition) Development Kit
JRE	Java Runtime Environment
JSON	Javascript Object Notation
MIS	Management information systems

NFC	Near Field Communication
OS	Operating System
PIN	Personal Identification Number
RAM	Random Access Memory
SDK	Software Development Kit
UI	User Interface
USB	Universal Serial Bus

1.4 Overview

This section addresses system requirements. Included in system requirements is information users need to know about their software and hardware components in order to run the eyeRS app successfully.

The eyeRS app will be a conventional app when it comes to requirements, in which users will need to have any device that runs an Android OS. There are many apps on the market each one with its identifiers, look and functionality. eyeRS will help users organise their belongings (virtually) into categories, or to sell or trade and share items by using a single app.

2. System Requirements

2.1 System Perspective

eyeRS is relatively unique. There are not many applications that are used for the purpose of making an inventory of household items. Most inventory mobile applications, are mainly used for businesses to track stock, as well as provide sales and purchase information. Onsight is an Android app which is an existing product closest to the eyeRS system's intended functionality. Another app such as Closet+, involves cataloguing a user's clothing, operating like a mobile closet, in which a user can keep track of their clothing (MY/STATIC/SELF, 2017). Closet+ is a automated style assistant developed for Apple's iPhone that is designed to aid users in organising, cataloging and planning their wardrobes developed by MY/STATIC/SELF (MY/STATIC/SELF, 2017). A user is able to organise clothing under a number of categories. Onsight is a mobile sales application which corresponds with the eyeRS applications functionality of employing a means for selling items within the catalogue via the application (Onsightapp, 2014). Onsight provides a mobile catalog application designed for manufacturers, wholesalers and distributors to replace paper catalogues (Onsight, 2016). Onsight's website (2016) indicates that they support various mobile platforms, including Android devices, Windows tablets and Apple iPads. Onsight's main purpose is for a salesmen for them to provide potential clients with descriptions and images of products.

The factor that makes our application unique is a combination of allowing the user to catalogue items and have the option of putting them up for sale via the application.

2.2 Functional Requirements

Table 5 - Functional requirements

Identifier	Description	Source	Priority
301	A user must be able to search the app for a specific item in their catalogue.	eyeRS development team	High
302	The user must be able to add, remove or edit an existing catalogue item in a quick and simple fashion.	eyeRS development team	High
303	The app must allow a user to share information about a catalogue item via a third-party app externally (e.g. email, WhatsApp, etc.) as a link or an app generated QR code.	eyeRS development team	High
304	The app must allow a user to capture images of all the items in their catalogue.	eyeRS development team	High
305	A user must be able to browse through their catalogue items in the app.	eyeRS development team	High
306	The app must enable users to delete any existing items from their catalog. Once an item is deleted, changes cannot be reverted.	Client	High
307	The app must allow the user to manage and maintain existing items.	Client	High
308	The user must be able to lock the app with a pin code chosen by the user.	Client	High
201	The user must be able to search their catalogue using various search filters.	Client	Medium
202	A user must be able to receive help and tips on the app's features.	Client	Medium
203	The app must retain all user data and preferences once closed.	eyeRS development team	Medium

204	The app must have the capability to operate on any device that supports the Android OS.	Client	Medium
205	A user must be able to add descriptions for their catalog items.	eyeRS development team	Medium
206	The user must be able to view a history of the traded catalog items in the app.	Client	Medium
101	A user must be able to tailor the app to their requirements (i.e. adding, removing or renaming categories).	eyeRS development team	Low
102	The app must support both portrait and landscape views for all the screens.	eyeRS development team	Low
103	The app must be able to accept null values for certain fields when adding a new item to the catalog.	Client	Low

2.1.1 Context Diagram (high-level DFD)

The diagram below indicates the context diagram (high-level DFD) for the eyeRS mobile application.

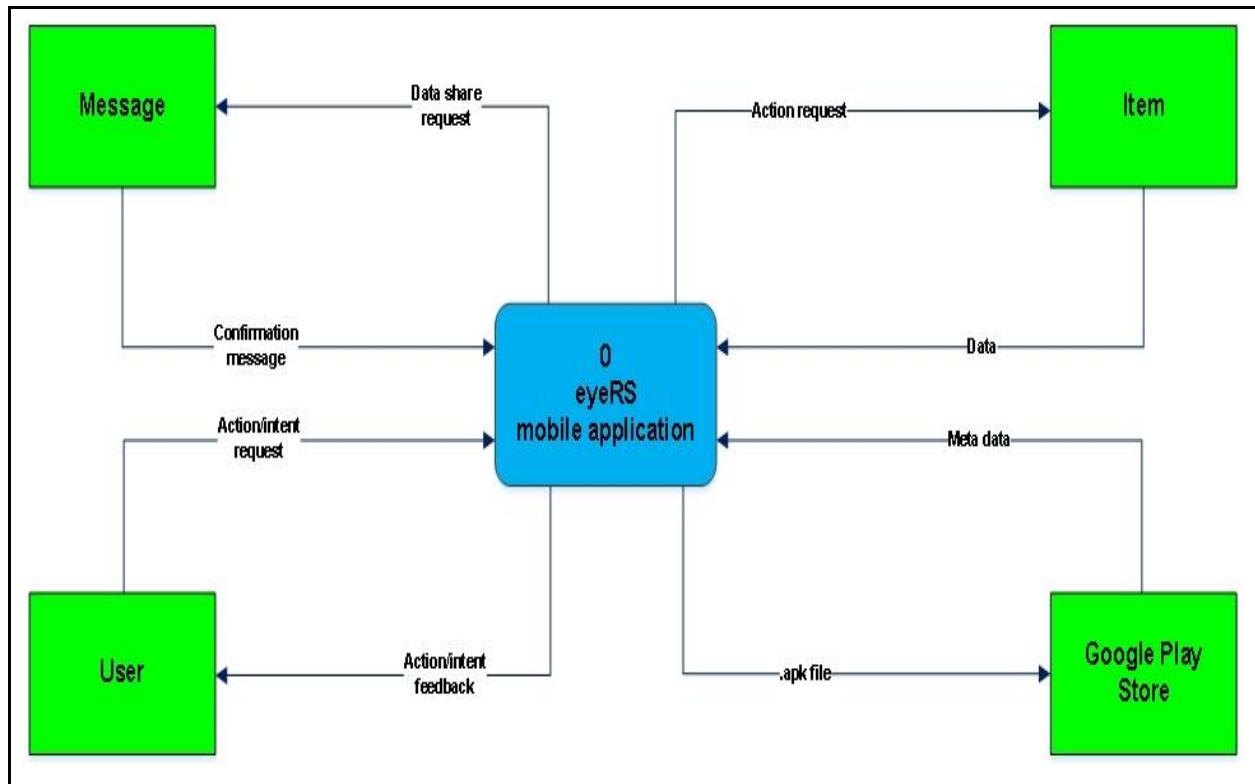


Diagram 2 - Context Diagram

Table 6 - Context DFD information

Entity	Data Flow from Entity	Data Flow to the Entity
User	Action/Intent request	Action/Intent feedback
Message	Data share request	Confirmation message
Item	Action request	Data
Google Play Store	.apk file	Meta-data

2.1.2 Entity Relationship Diagram (ERD)

The ERD below does not include attributes and follows Crow's Foot format as presented by (Connolly & Begg 2005).

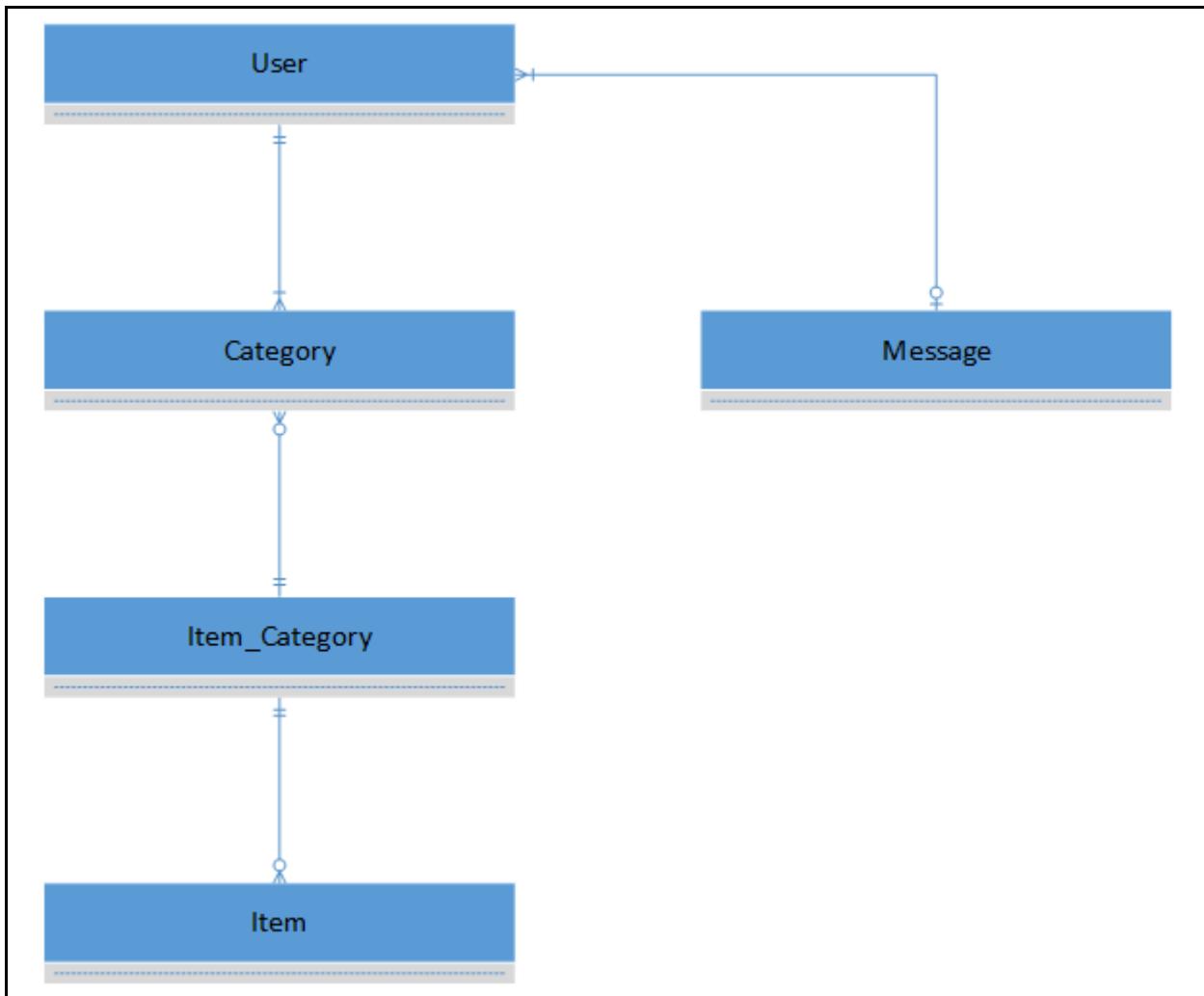


Diagram 3 - An ERD of the system

2.3 Non-functional Requirements

Human-Computer Interactions (User Interfaces)

The application should make use of Nielsen's usability heuristics and incorporate the following non-functional requirements:

1. Users should require no additional training in order to use the app;
2. The app must be accessible to all users;
3. The app should provide the user with productivity gains;
4. The app must not have a high error rate; and
5. The app must have accessible to all users.

Security Requirements

The app will be safe in terms of keeping the integrity of user's data safe. It will be secured by a login password or pin, depending on the user's choice. The storage platform is local which offers more control to the user and availability.

Communications Interfaces

The following non-functional requirements are applicable to the eyeRS mobile app communication interface:

1. The app must provide reliable interaction functions between the user and device.
2. There must be a simplicity factor related to the symbols that will be used in the eyeRS app;
3. The eyeRS app must provide an adaptable communication interface to the user;
4. The user interface must be understandable for the user to communicate to the app; and;
5. The eyeRS app must allow the user to modify the app to ease the communication with the app.

Hardware & Software Interfaces

Screen resolution and form factor considerations

A platform like the Android OS, which can run on many types of mobile and tablet devices, is likely to encounter various screen DPI issues (Android Developers, 2017). It is, however, important to keep a set of target screen resolutions while designing the user interface for the app (Android Developers, 2017). The following kinds of changes will be required for supporting different kinds of screen resolutions:

1. Font adjustment - The font size may be adjusted higher (for large DPI screens) or low (for small DPI screens) in order to ensure text remains readable;
2. Layout tweaks: The layout may need to be adjusted to increase or decrease the spacing between and around labels, and widgets shown on the screen. The ability to tweak will prevent information from becoming cluttered on high DPI screens or spaced apart too much on low DPI screens; and
3. Image changes - Background images or art may have to be provided in two different versions, namely a large size/high resolution version and a small size/low resolution version.

The app will need to utilize the OS and SDK mechanisms that have been provided to be able to cope with various screen size-ranges of devices operating on that specific OS (Android Developers, 2017).

Touch and non-touch screens: It is important to make sure that touch is activated for all the functionalities in it, when the app is running in touchscreen mode (Android Developers, 2017). Generally, API support will be present for the app to establish if the app is running on a touch-enabled screen or not (Android Developers, 2017).

Portrait and Landscape mode: Avoiding to add the support both portrait and landscape modes is necessary if the app will not require both modes to operate (Android Developers, 2017). However, if it turns out to be needed then it is vital to ensure that there is support for both modes on all the user interfaces in the app and not just a few of the interfaces in order to avoid users encountering a usability surprise (Android Developers, 2017).

CPU & Memory Characteristics

Android apps can operate on both a low-end Android phone with minimal amount of RAM available as well as a single core CPU. They can also operate on high-end devices with large amounts of RAM and even making use of a processor specified as an Octa-core (Android Developers, 2017). Should the app, however, be specified to require a specific minimum RAM and CPU power then it will, practically, fail on the low-end devices (Android Developers, 2017). If the app uses extensive arithmetic and logic functions such as those involved in streaming and decompression of audio and video and in rich animations, or it lets the user view and manipulate large sets of information or images then the minimum CPU and memory requirements for the app will need to be specified in precisely the same way as with similar desktop apps (Android Developers, 2017).

In order to objectively determine the requirements, every feature in the app can be evaluated from the perspective of its respective memory and CPU usage through operating via profiling software tool and arriving at a low tier for the whole app from the assessment (Android Developers, 2017). For the eyeRS app, the following profiling tools can be used to determine the CPU and memory requirements:

ADT comes with the DDMS which can let you view a number of data relating to the operating state of the eyeRS app like heap and thread data, LogCat display and processing data etc (Android Developers, 2017).

Several 3rd-party apps can be found on the Play Store to assist in measuring and analyzing memory, CPU and the battery performance of the app.

Network Condition Scenarios

There are four factors to consider with regards to the network condition of a mobile app:

1. Supporting multiple network protocols - Devices can operate with the network on one or more protocols like WiFi, LTE etc (Date, 2015). Certain features in the app may not operate well (or not operate entirely) on certain protocols. Supporting only the elite protocols like WiFi & LTE will create a risk of excluding the low-range devices which may not function on these protocols (Date, 2015). However, required or recommended protocols need to be specified as failure to do so may lead to users unintentionally trying to use features which require high-bandwidth on lower-tier protocol like GPRS, hence, causing themselves to become frustrated from using the app. For features such as streaming media, it will be advisable for users to make use of high-range protocols like 4G or WiFi (Date, 2015). It is also important to advise the user if a specific feature will become unavailable on a certain channel. Users will also need to know whether any additional usage charges may be incurred by selecting to operate over specific channels such as 3G so as to avoid getting billed;
2. Signal drop/strength reduction - Each network-enabled feature needs to be assessed in the app in the situation where the protocol, over which it is operating, becomes unavailable or its signal strength reduces (Date, 2015). The feature has to be either network-fault tolerant or fails gracefully in such a situation;
3. Network protocol transition - This network condition handles the app's behaviour when the device moves from a particular protocol to another. An example would be where the user leaves the work building and the device moves from WiFi connectivity to 4G (Date 2015). If a process was taking place in the app at this moment in time, *how will the app handle such a scenario?* It is therefore important to assess the app's capability in such scenarios and design it for a seamless transition to the other protocol or for a graceful failure; and
4. Support for multiple protocols - This network condition relates to the app's behaviour when there are multiple network protocols active at a time (Date, 2015). The Android OS today automatically prefers WiFi when available, as to using cellular data (e.g. LTE). It is important to enable listening to a single or multiple network-based events when a network protocol gets active or inactive and also when the strength of the signal for the active protocol is altered (Date, 2015).

Battery-consumption Considerations

On devices, the battery is a rare and important element. There are 3 main causes of battery drain in devices:

1. The device's main processor;
2. The device's screen (including the GPU); and
3. The communications processors.

On a device, the battery is supposed to be wholly available to the phone app which is used for initiating and reception of phone calls. The eyeRS app may then become unpopular or even get uninstalled, if it consumes a lot of battery life during use. The idea of what is meant by 'a lot' may vary, due to the nature of the app and the hardware capabilities of the platform it is running on. A good way to evaluate the battery usage characteristics is to evaluate the app against a 'control' app which is usually a 'well-known app' in its class, in this instance *Closet+*. There are 3 general battery-power tests that should be carried out:

1. Regular use test - Starting with the battery on 100%, use the eyeRS app for six to twelve hours and measuring the battery level after each half or hour mark (Qian, 2015). A dynamic testing tool may be used to perform this test, in order to ensure that the test will run for the specified time interval (Qian, 2015). The test then reveals how fast the eyeRS app consumes battery life while in 'regular' use, and also, ensuring background and foreground features in the app are functioning as they should;
2. Idle-run test - The device should have its screen locked and power saving mode switched off (Qian, 2015). The battery life should be on full and then the app should be kept running on its main-home or dashboard view as specified, measuring the battery life at ½ or 1 hour intervals. This test will measure the battery drain due to such things as intentional or unintentional automatic screen refreshes, and due to the background threads or services running in the app (Qian, 2015); and
3. Screen lock test - Performing an idle-run test again, although this time the device's screen should be locked (Qian, 2015). This enables testing the app to determine if it is using any network and, or, CPU resources (including the battery life) when the app cannot be viewed by the user (Qian, 2015).

In every instance, it may be useful to analyze the battery level against time so as to get a rapid visual depiction of the overall trend.

It is also important to ensure that, while carrying out the tests, no other app is operating on the device and to switch off the phone feature so as to ensure calls will not be received while the test is in progress (Qian, 2015). If possible, turning off apps like chat clients that usually run in the background will also be required.

Sensor Attributes

A lot of devices (including low quality phones) are equipped with an in-built camera, that could be specified as a particular sensor. Mobile devices & tablets usually have multiple types of sensors like GPS-location, gyroscope, ambient light & proximity and accelerometer sensors, for example. They are also capable of connecting to external sensors through Bluetooth or USB connectivity. While the eyeRS app may make use of a sensor to receive information, attention needs to be focused on the following attributes of the sensor:

1. The Maximum Sample rate;
2. The Operating range;
3. Sensitivity; and
4. Accuracy.

Media Functionalities

The device's OS and its hardware that the app operates on, will generally determine which audio and video capabilities the app will support. E.g, which audio/video formats the app supports; whether it will accommodate full HD, and if not, if it can facilitate multi-channel surround sound etc., which are all specified by the OS's capability and traits of the device's hardware. The following needs to be noted before a decision can be made as to which specific audio/video features are to be supported:

1. Details of the OS version that the app will operate on need to be known and whether it has support for any specific media features like surround sound that the app will need to support. The app needs to be targeted at versions only at and higher than that version. In a number of cases, it will also be a requirement to provide programmatic access to these capabilities, hence, ensuring that SDK support is available to run the features through the app's code is just as important;
2. If no out-of-the-box support exists while the OS supports the feature via its SDK, then, it is important that there is enough support through 3rd-party libraries. If not then there is the risk of often incurring significant costs of issuing support by creating a unique software library; and
3. It is also important to ensure the mobile hardware targeted for the app has the hardware for running the features required.

Support for various versions of Android OS

Various mobile OSs are designed to allow them to be ‘forward compatible’. The OS developers put in a lot of effort to ensure that as long as the app uses the official developer SDK in a prescribed manner by the SDK documentation, the app written for OS version 1.0, for example, will operate on version 2.0 etc.

Up to a certain extent, the forward compatibility stops and the app creator will be required to reconstruct the app for a higher version of the OS. Users generally update the OS on their devices often. It is therefore important to include forward compatibility within the app. If the app no longer operates after the user updates the OS to a better version, the user could decide to uninstall the app simply, and not be worried about getting the higher version of the app, although it may be available. Here are some of the considerations to ensure the app continues to operate on later OS versions:

1. Avoiding extensive use of 3rd party libraries;
2. Avoiding use of API's not recommended by Android;
3. Implementation of best practices while using APIs and avoiding non-standard usage of API methods;
4. Testing the app on all OS versions that are intended to be supported on;
5. If any features are OS specific, it is important to ensure they fail gracefully on prior OSs; and
6. Whenever the app starts, it is important to determine if a better version is available and urge the user to get the version. Usually, the Google Play Store should advise the user through a push-messages when a better version of the app is ready for download. It may be preferable to perform the version check each time the app starts. The user may fall back in the version of the app by what the latest available version is on the Google Play Store , therefore, it could be vital to urge the user to update the app prior to allowing them to use it.

Interrupts, notifications and multitasking

If a phone call, text messages or other types of notifications (e.g. calendar-reminders) occur, the device will probably inform the eyeRS app of the notification. Selecting to respond to the notification, the OS may then background the eyeRS app or, in the event of mobile OSs not capable of multi-task capability, they may just terminate the eyeRS app.

In either situation, the OS will most likely give the eyeRS app a chance to respond to the halt, termination or background event by issuing a handler method that should be implemented. It is necessary for the eyeRS app to handle the interrupt in such a way which will:

1. Not interfere with the OS's processing the user's choice to respond to the interrupt (like accepting a call or reading an SMS); and
2. Not lead to damages to the app's capability to operate as normal after the OS resumes the app after the user has finished dealing with the interrupt or if they decide to ignore it.

Each feature must be evaluated in the app from the perspective of determining how it could, and must operate if the app is pushed to the background by the OS, or made dormant while that feature is in execution-mode, and how it will recover from this interrupt situation after the OS or the user 'foregrounds' the app back after servicing the interrupt.



2.4 Technical Requirements

1. The app functionality must be developed using the Java programming language supporting a minimum JDK of version 7;
2. The app should support a minimum OS version of Android IceCreamSandwich (4.0.3) or higher (Google, 2017);
3. The mobile application should allow for future maintenance updates;
4. The deployment of the mobile application will require an active Google developer account;
5. The app should be developed in Android Studio IDE or other appropriate IDE;
6. The Android Studio SDK version used by the development team to construct the app is to meet the minimum requirement of version 2.2;
7. The app will have API documentation generated using javadoc;
8. Use of the SDK Manager will be required for testing the app on different API levels to target different OSs as well as installing, updating or uninstalling packages;
9. The development team will construct the app using machines that support a minimum JRE of version 7; and
10. To develop a prototype for the proposed end user to approve using Justinmind Prototyper or another equivalent prototyping tool.

2.5 User Characteristics

The users of the eyeRS mobile application should be one that requires an easy to use catalogue for day-to-day life. Just basically keeping track of household items and to improve on your home by upgrading, or getting rid of unwanted items. An application for users to make their house a place they can call home.

The users are not limited to a specific group. Whether they're novices or experts in handling a mobile device, bachelors / bachelorettes living on their own or with roommates or even families the application caters to all. According to (Matz 2013) there are a few preferences to be considered:

Table 9 - Preferences

Characteristic	Preference
Age	3+
Gender	Male/Female
Educational level (Level of experience)	A basic knowledge of operating mobile devices
Language	English
Computer skills	Novice; Technologically literate
Domain-related knowledge and skills	Interior design
Physical environment	Home, office, on the go
Social environment	Family members, friends, business

2.6 Operational Environment

The operational environment will consist of two environments. An environment for development and testing of the application and an environment for the prototype and final application. The eyeRS application will become available for download on android mobile devices.

The application development and testing will take place on a desktop with the Microsoft Windows XP or later versions of the operating system (tutorialspoint.com 2017). The workstations will require the following tools for development:

1. Java Runtime Environment (JRE) 6 or higher and Java Development Kit (JDK) 8 or higher;
2. Android Studio 2.2 or higher;
3. Android Software Development Kit Manager (SDK Manager); and
4. Android Virtual Device Manager (AVD Manager).

Android studio will be the main tool used for the app development and the Java language to will be used to develop eyeRS functionality.

The app will be operating on00 android mobile devices therefore the mobile devices will need to make use of a supported android version 4.0.3 (IceCreamSandwich) or higher.



3. Customer sign-off

Customer name and surname	Customer signature
Group leader name and surname	Group leader signature

Bibliography

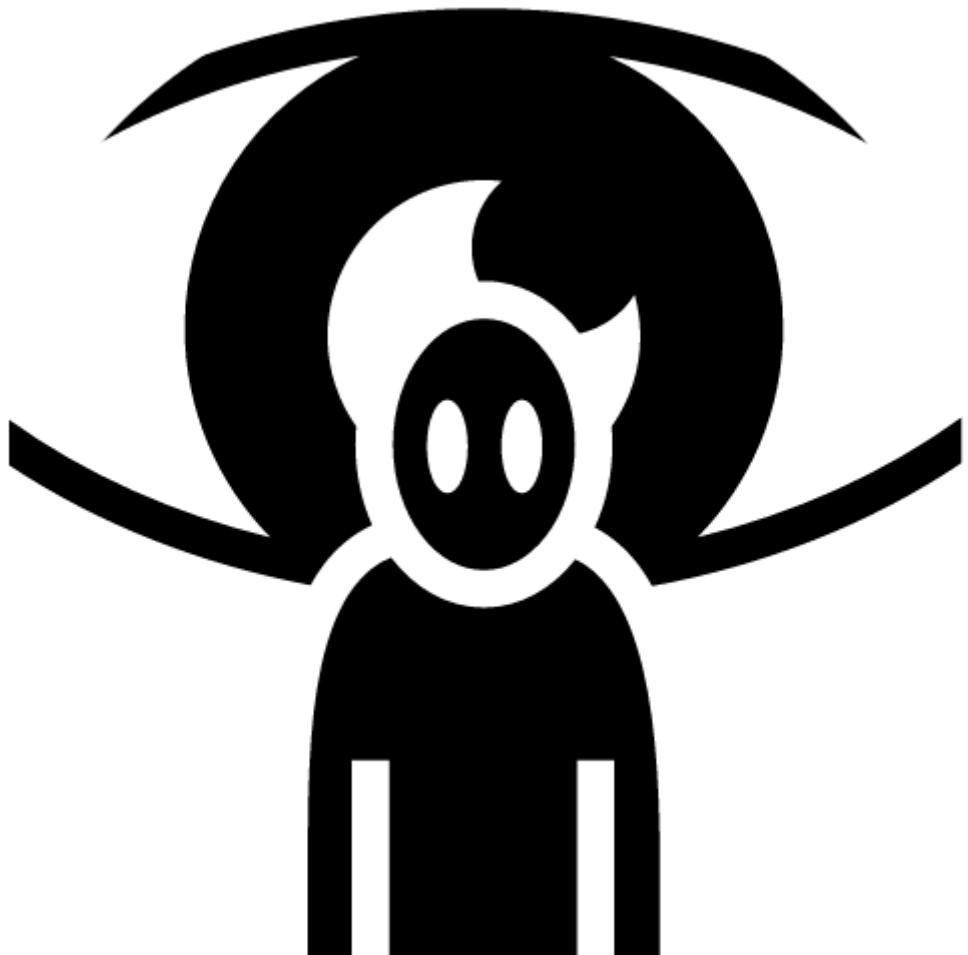
- Android, Intent | Android Developers. Available at:
<https://developer.android.com/reference/android/content/Intent.html> [Accessed April 13, 2017].
- Best Practices for Performances | Android Developers. Available at:
https://developer.android.com/guide/practices/screens_support.html [Accessed April 13, 2017].
- Supporting Multiple Screens | Android Developers. Available at:
https://developer.android.com/guide/practices/screens_support.html [Accessed April 13, 2017].
- Bass, L., Clements, P. & Kazman, R., 2012. 4.2. Functionality | Understanding Quality Attributes in Software Architecture | InformIT. *Informit*. Available at:
<http://www.informit.com/articles/article.aspx?p=1959673&seqNum=2> [Accessed May 3, 2017].
- Coach, B., 2015. What is Feedback? | Definition of feedback in Communication. *Business Communication*. Available at: <http://bizcommunicationcoach.com/what-is-feedback-definition-of-feedback-in-communication/> [Accessed May 3, 2017].
- Connolly, T.M. & Begg, C.E., 2005. *Database Systems: A Practical Approach to Design, Implementation, and Management*, Pearson Education.
- Cress, K., 2017. 5 Types of Information Systems. *MindMeister*. Available at:
<https://www.mindmeister.com/37310006?title=5-types-of-information-systems> [Accessed April 12, 2017].
- Date, S., 2015. *An Illustrated Guide to Mobile Technology*, CreateSpace.
- Eastman, C.M., 1999. *Building Product Models: Computer Environments, Supporting Design and Construction*, CRC Press.
- Google, 2017. Best Practices. *Android Developers*. Available at:
<https://developer.android.com/guide/practices/index.html> [Accessed February 13, 2017].
- Hope, C., 2017. What is debugging? *Computer Hope*. Available at:
<https://www.computerhope.com/jargon/d/debuggin.htm> [Accessed May 3, 2017].
- Martin, P. & Tate, K., 2002. *Getting Started in Project Management*, John Wiley & Sons.
- Matz, K., 2013. *Designing Usable Apps: An agile approach to User Experience Design*, Winchelsea Press (Winchelsea Systems Ltd.).
- MY/STATIC/SELF, 2017. Closet+ / The Swiss Army Knife of iOS Style Assistants. *Closet+*. Available at: <http://closetapp.com> [Accessed April 6, 2017].
- Onsightapp, 2014. Android sales app - Onsight. *android-sales-app*. Available at:
<https://www.onsightapp.com/android-sales-app> [Accessed April 6, 2017].
- Qian, H., 2015. *Extending the Battery Life of Mobile Device by Computation Offloading*,

- Rob, P. & Coronel, C., 2004. *Database systems: design, implementation, and management*, Course Technology Ptr.
- Rouse, M., 2017a. What is Android OS? - Definition from WhatIs.com. *SearchEnterpriseLinux*. Available at: <http://searchenterpriselinux.techtarget.com/definition/Android> [Accessed May 3, 2017].
- Rouse, M., 2017b. What is mobile app? - Definition from WhatIs.com. *WhatIs.com*. Available at: <http://whatis.techtarget.com/definition/mobile-app> [Accessed May 3, 2017].
- Rouse, M., 2017c. What is prototype? - Definition from WhatIs.com. *SearchManufacturingERP*. Available at: <http://searchmanufacturingerp.techtarget.com/definition/prototype> [Accessed May 3, 2017].
- Sommerville, I., 2011. *Software Engineering*, Pearson Higher Ed.
- Sonmez, J., 2010. Back to Basics: What is an Interface? *Simple Programmer*. Available at: <https://simpleprogrammer.com/2010/11/02/back-to-basics-what-is-an-interface/> [Accessed May 3, 2017].
- Stair, R.M. & Reynolds, G.W., 2008. *Principles of Information Systems: A Managerial Approach*, tutorialspoint.com, 2017. Android Environment Setup. [www.tutorialspoint.com](http://www.tutorialspoint.com/android/android_environment_setup.htm). Available at: https://www.tutorialspoint.com/android/android_environment_setup.htm [Accessed April 28, 2017].
- Viswanathan, P., 2017. What Is a Mobile Device? *Lifewire*. Available at: <https://www.lifewire.com/what-is-a-mobile-device-2373355> [Accessed May 3, 2017].



System Logical and Physical design

eyeRS Development Team - ITSP200 (Deliverable 3)





Introduction

This deliverable presents the system logical and physical design document. This document will define what physical and logical design entails and transpire. It also includes the logical data model (ERD), the logical process model (DFD), the testing templates and the test plan. Additionally it includes the system interface designs along with the technologies that have been used in this project.

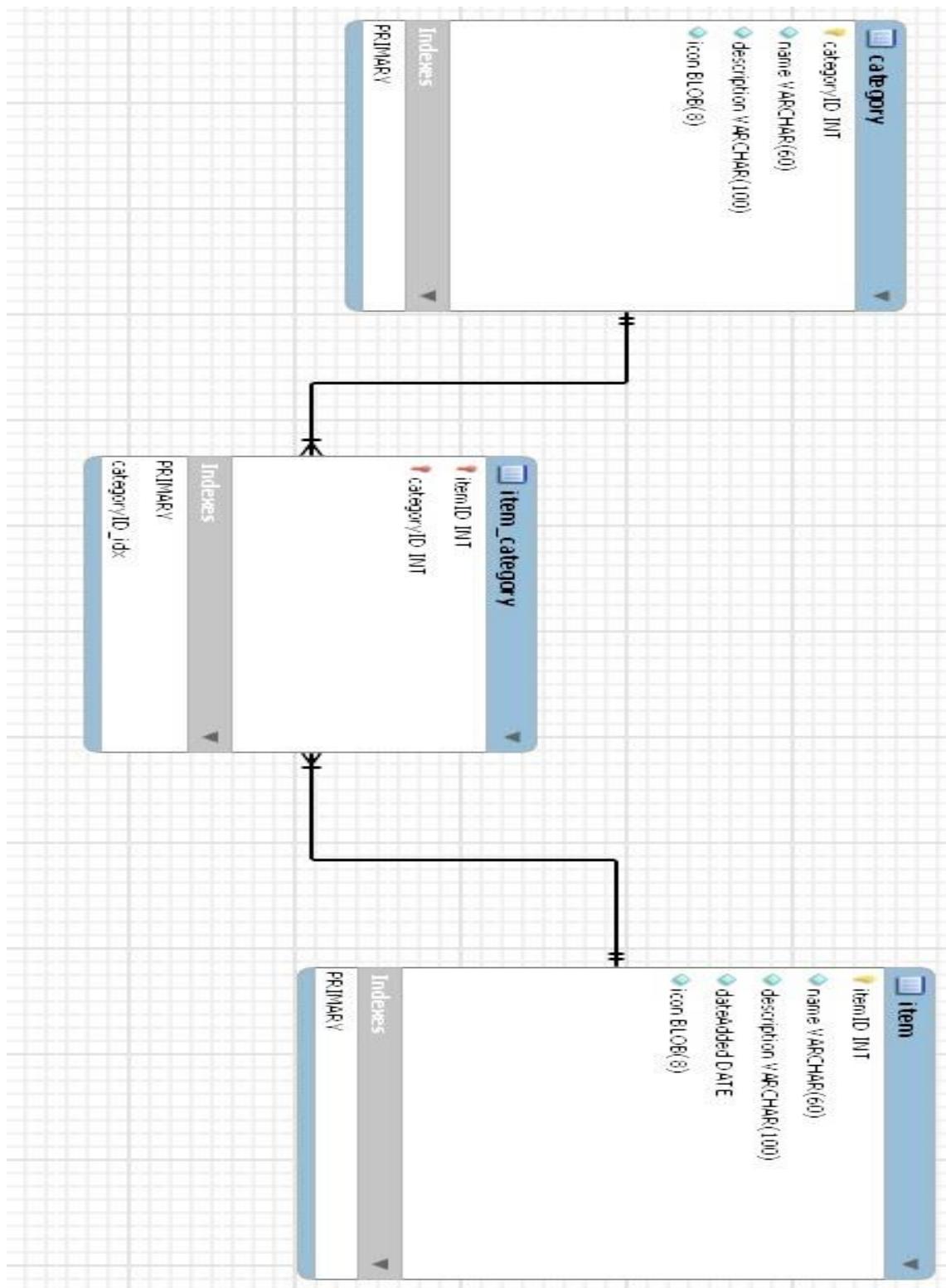
The purpose of the eyeRS app is to give individuals the power to manage and organise their personal and professional lives with less hassle. The eyeRS app achieves this control by allowing users to freely access and catalog all of their belongings and items that are uploaded to the app. Uploads can occur anywhere. Each belonging is saved by means of an image, title and a short description.

1. Information systems design

1.1. Logical design

Logical design of a system relates to the abstract presentation of the data flows within the system, it is mostly conducted via modelling by using an over-abstract model of the actual system (Satzinger, *et al.*, 2015). Logical design shows how the data that have been given to the system is processed (Satzinger, *et al.*, 2015). This design type is used to document information systems since the logical nature of the system is documented without specifying the detailed tasks of how, where and by whom the information that the system needs, are gathered. Logical designs can be represented by an Entity-Relationship diagram (Mahfuj, 2012).

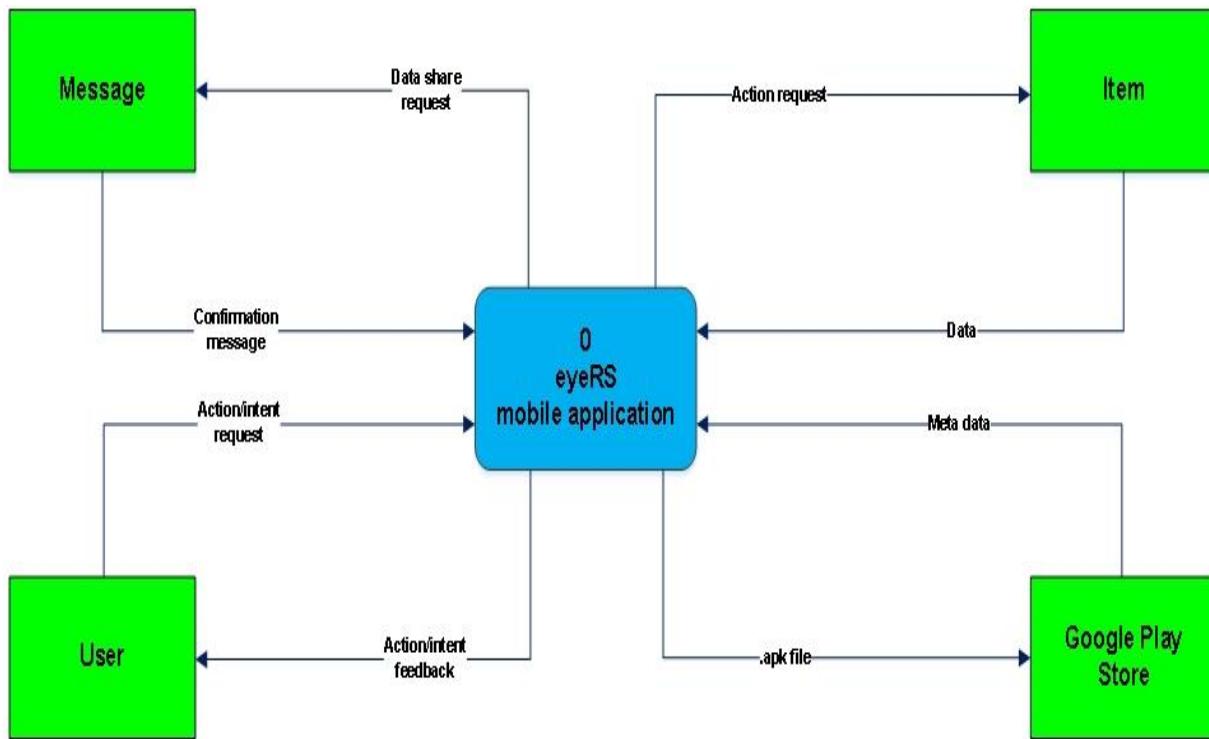
1.1.1. EyeRS Entity Relationship Diagram



1.1.2. Context & Logical Process Model

Context (High Level)

The diagram below indicates the context diagram (high-level DFD) for the eyeRS mobile application.



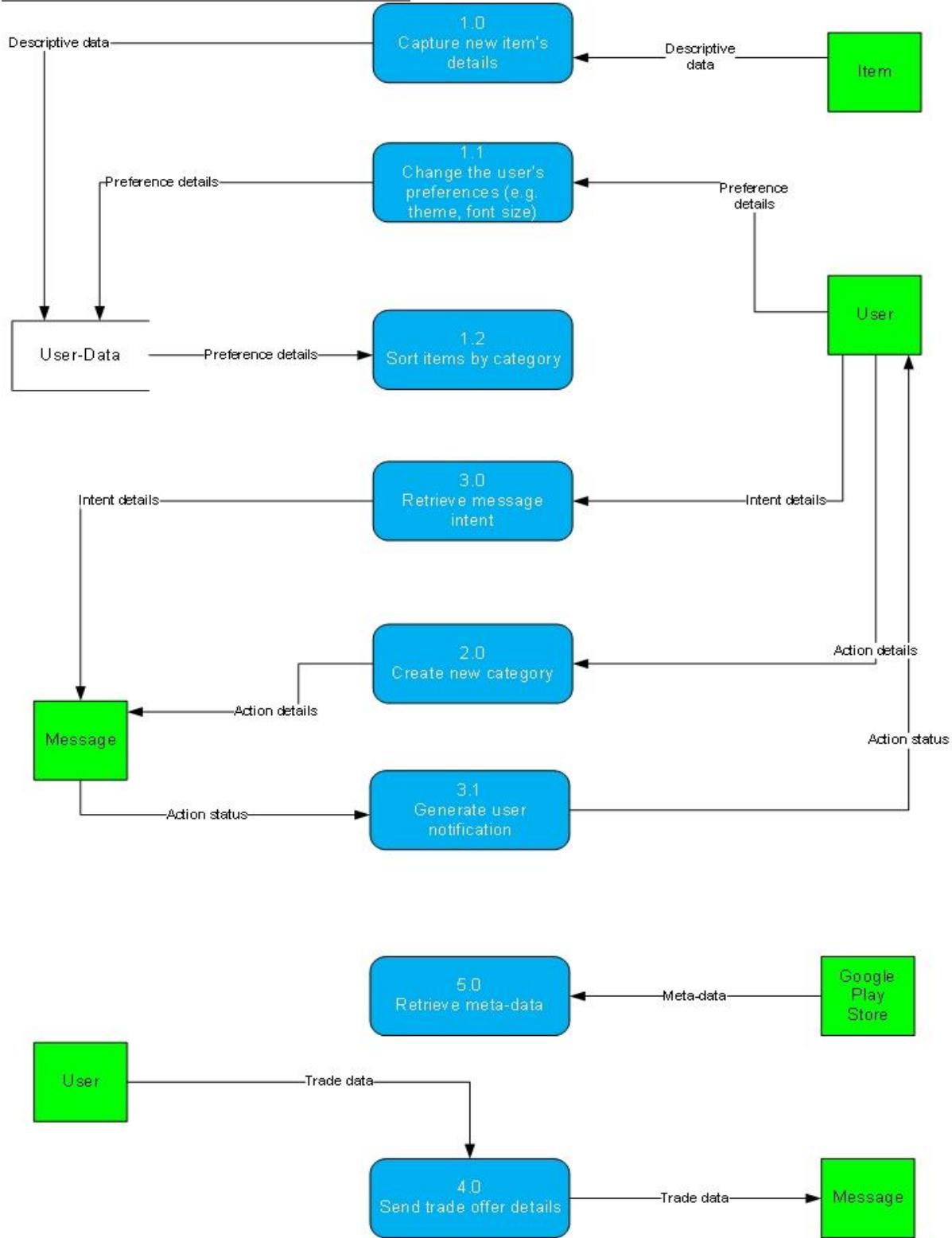
Processes

- 1.0 Capture the new item's details;
- 1.1 Get user preferences;
- 1.2 Sort items by category
- 2.0 Create new category;
- 3.0 Retrieve message intent;
- 3.1 Generate user notification;
- 4.0 Send trade offer details
- 5.0 Retrieve metadata

The table below is a summary of the data flows, processes and entities relating to the Data Flow Diagram.

Data Flow	From	To
Intent details	User entity	Retrieve message intent process
Descriptive data	Item entity	Capture new item's details process
Descriptive data	Retrieve item's details process	User-Data data store
Preference details	User entity	Implement the user's preferences process
Preference details	Implement the user's preferences process	User-Data data store
Intent details	Retrieve message intent	Message entity
Action status	Message entity	Generate user notification process
Action status	Generate user notification process	User entity
Meta-data	Google Play Store entity	Retrieve meta-data process
Action details	User entity	Create new category process
Preference details	User-Data data store	Sort items by category process
Trade data	User entity	Validate trade request process
Trade data	Validate trade request process	Message entity

Logical Process Model (Low level DFD)



1.2 Physical design

Physical design is the graphical presentation of the system (Dennis, et al., 2005). It shows the internal and external units. It also shows how the data is flowing within the system (Dennis, et al., 2005). The physical design relates to the input and output process of the system, how data

is authenticated, how it is processed and how it is displayed as output (Dennis, et al., 2005). Physical design can be broken down into three sub-tasks. These subtasks of the physical design are:

1. User Interface design;
2. Data design; and
3. Process design.

User interface design focus on how the user gives information to the system and how this data will be presented to the user (Shelly, et al., 2003). Data design is how the data is stored and represented within the system (Shelly, et al., 2003). Process design focuses on how data will move within the system, how and where the data is validated, secured or transformed as it is flowing within the system as well as out of the system (Mahfuj, 2012).

1.2.1 Investigation of technologies to be applied

The operating system that we will be using throughout project development, on the different stages of development from planning until implementation/maintenance will be Windows 7,8 or 10 operating systems.

There are other operating systems on the market, such as: Linux and Mac OS however, Windows will be the preferred operating system of use. On the Windows operating system we will run different technologies to enable successful development of the eyeRS app.

The following technologies are used throughout the development of the eyeRS project:

1. Justinmind;
2. Android studio;
3. Microsoft word;
4. Github;
5. Google Drive;
6. Java Runtime Environment (JRE);
7. Java Development Kit (JDK);
8. Android Software Development Kit Manager;
9. Android virtual Device Manager; and
10. SQLite.

The above mentioned technologies are briefly discussed below:

Justinmind is a system that allows users to create system and application prototypes, supporting many devices.(Justinmind, 2014) It uses a drag and drop feature and assign events to the objects (Justinmind, 2014). It will assist us in creating user interfaces and allow us to gain feedback from the client of specific design preferences.

Android studio is a development environment for the Android platform that we will use to develop, debug and deploy the eyeRS app (Android Developers, 2017). It provides the best tools for building high quality apps for any android device (Android Developers, 2017).

Microsoft word is a graphical word processing system that users can create documents with (WebAIM, 2016). We are using it for the project documentation . The documentation of the project is important for many reasons such as:

- 
1. A reference for other projects;
 2. For system upgrade or enhancement;
 3. Project troubleshooting and;
 4. Providing an understanding of the system concept behind its functionality (Shelly, et al., 2003).

Github is a repository hosting service (Finley, 2012). It will allow the team to collaborate on development tasks via its platform, in a secure and productive way (Finley, 2012). It allows project team members to work on the project tasks from anywhere regardless of location (Github, 2017).

Google Drive will provide access to files anywhere in a secure way, working as a cloud storage to backup images, videos and documents (Google, 2017).

It also enable the project team to utilize Google Docs, which is an online, word processing program to compile documents allow the team to work on the same document simultaneously (Google, 2017).

Java Runtime Environment is a software package that consists of the necessary requirements to execute a Java program (Oracle, 2017). The eyeRS team will make use of this technology (which comes as part of the Android Studio SDK) to create the app's functionality (Android Developers, 2017).

Java Development Kit is will be used to develop java programs as it also comprises of the Java Runtime Environment in the kit (Oracle, 2017).

Android Software Development Kit Manager is a set of development tools which are embedded in the Android Studio development environment which will be used to debug the eyeRS app (Android Developers, 2017).

Android virtual Device Manager is a program that will enable the development team to debug their apps on a virtual machine with specifics requirement such as: hardware, storage space, screen resolutions and allowing the development team to target various Application Programming Interface (API) levels (Android Developers, 2017).

SQLite is a database management system based on the structured query language.(Oracle, 2017). It is a technology used by the Android system to create databases to be used for storage by the application (Android Developers, 2017).

1.2.2 System testing

Testing types

Software testing is the ongoing process of ensuring that a program not only is function but also fulfils all set criteria for the program (IEEE, 1990). According to IEEE (1990) and Williams (2006) there are various primary types of programming tests that are should be performed on a program before it can be labeled as acceptable, these tests include:

1. Walkthroughs;
2. Trace table
3. Black box/data coverage; and
4. White box/Code coverage.

Bertolino (2001) suggested that software testing be conducted throughout the development process.

Walkthroughs are a fairly common and cost effective method of software testing. Walkthroughs are done by manually tracking a program's data and output on pen and paper. Walkthroughs are frequently used to facilitate discussion to what extent a program has met its design requirements. Trace tables are similar to walkthrough however they are focused on testing the logic of an algorithm rather than the program as a whole (ref). According to ref trace tables are good for highlighting small non obvious logic errors with an a program.

Black-box testing, also known data coverage, is when a subprogram verifies a set of known valid inputs are tested against possible input values (Williams, 2006). According Williams (2006) to black-box testing is used to verify that a program can process possible values and meets the proposed requirements.

White-box testing, also known as code coverage, is when the execution of statements, or groups of statements, are tested to ensure that they execute as expected (Williams, 2006). Williams (2006) stated White-box testing is used to ensure that all statements execute as expected.

Softwaretestingclass (2012) states that system testing tests the behaviour of the system based on risks, specifications and requirements which include functional and nonfunctional requirements that were established by the developers. A fully integrated system should be tested. The final testing done is the Systems Testing to verify that all specifications and requirements have been met.

Testing Templates

Test Case Number: 1 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Open the application.	Test Case Name: Open the app. Subsystem: Stating. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the eyeRS icon	A Welcome message appear with the login screen.			

Test Case Number: 2 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Test the registration for eyeRS.	Test Case Name: Register new user. Subsystem: Register. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click "Register" button	The system displays a screen with registration details			
2	Enter a username	The system displays the username in the text area			
3	Enter an email address	The system displays the email address in the text area			
4	Enter a pin "*****"	The system asks the user to verify the pin			
5	Re-enter the pin "*****"	The system verifies that the pins match			
6	Select a security question	The System opens a drop-down list			
7	Enter security response (Answer to security question).	The system accepts the input			
8	Click the "Register" button	The system registers the user with the database, a Registered notification appears and returns to the "Login" screen			



Test Case Number: 3 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Test the clear button on the register screen.	Test Case Name: Clear register new user information. Subsystem: Register. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the “Register” button	The system displays a screen with registration details			
2	Enter the registration information	Information display on the fields.			
3	Click the clear button	All the information in the fields will be erased. Same screen are displayed.			
4	Click the back button.	Login screen is displayed.			



Test Case Number: 4 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user the opportunity to retrieve a forgotten pin.	Test Case Name: Forgot Pin. Subsystem: Login. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the red “Forgot Pin” text	New screen appears that display security question and text field for the security question answer.			
2	Enter username	Information displays on the textfield.			
3	Create an new pin in the pin text field	Pin would appear as stars			
4	Verify the pin by retyping the pin in the text field	Pin would appear as stars			
5	Select the drop down box	A list of possible security questions will appear			
6	Enter security response	Text field is updated.			
8	Click the “Reset” button	New pin as well as new security question and security answer is created and saved in the database. Rese successful notification appear. User is returned to the Login screen			



Test Case Number: 5 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Test the clear button on the forgot pin screen.	Test Case Name: Clear reset pin information. Subsystem: Register. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the “Forgot Pin” text	The system displays a screen with the password reset details			
2	Enter the password reset information	Information display on the fields.			
3	Click the clear button	All the information in the fields will be erased. Same screen are displayed.			
4	Click the back button.	Login screen is displayed.			

Test Case Number: 6 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Log in to use the eyeRS application.	Test Case Name: Login. Subsystem: Main Screen. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the “Please enter your PIN” textfield	Keyboard appears.			
2	Enter pin	Pin appear as stars			
3	Click the “Login” button	If correct the system will open the main menu screen. If not the system will notify the user of the incorrect password, and the pin can be re-entered.			

Test Case Number: 7 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Displays side menu.	Test Case Name: View Side Menu. Subsystem: My Category. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the “More” icon in the main menu	A side menu with more options will appear.			

Test Case Number: 8 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Adds an item to the users database via the camera upload.	Test Case Name: Add Item. Subsystem: My Category. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on “Add new item” in the side menu	An Add Item screen will appear.			
2	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			
3	Click on Cancel	No photo were added.			
4	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			
5	Click on Take Photo	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Select the category drop box	A list of the categories fill appears.			
7	Select the appropriate category from the drop box list	The selected category will appear. (Note: Not yet saved).			



8	Click in the Name of item text field	Keyboard appears.			
9	Enter the title of the item in the item text field	Title will appear in the text field. (Note: Not yet saved).			
10	Click in the Item description text field	Keyboard appears.			
11	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
12	Click the “Add” button.	Item is added in the database. Notification of saved appears. User remains in the add item screen			
13	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture— picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
14	Enter the new item detail as before.	Item detail displays.			
15	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add item screen.			
16	Click on the back button.	Main menu appears.			



Test Case Number: 9 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Adds a category to the user's database via the camera upload.	Test Case Name: Add Category. Subsystem: Main menu. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.			
2	Click on "Add new category" in the side menu	An Add Category screen will appear.			
3	Click on the "Camera" icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Category Name text field	Keyboard appears.			
7	Enter the Name of the category in the category text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the category in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the "Add" button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.			
11	Click on the "Camera" icon	Notification appears asking for an upload of the image			



		via the camera or the gallery.			
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture— picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new category detail are before.	Category detail displays.			
14	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.			
15	Click on the back button.	Main menu appears.			



Test Case Number: 10 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: View images from the user catalogue.	Test Case Name: Slideshow. Subsystem: Main menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.			
2	Click on "Slideshow" in the side menu	An slideshow screen will appear and the images in the catalogue will appear in a slide show			
3	Click on the back button.	Main menu appears.			



Test Case Number: 11 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Shares a standard or custom user category and all content with other user's	Test Case Name: Share from side menu. Subsystem: Main menu. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the more icon on the main menu.	The side menu will appear.			
2	Click on "Share" in the side menu	A select category screen will appear.			
3	Click on appropriate category.	A Select Item screen will appear.			
4	Select an item to share.	A pop up will appear that shows all the supported sharing methods.			
5	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
6	Click the back button to return from the sharing app.	User is returned to the select item screen.			

*Note: Repeat for each sharing method.



Test Case Number: 12 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Searches for items by (name, category, details).	Test Case Name: Search Item to share via the side menu. Subsystem: Share. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon in the select item screen	Search text field will slide from the side and keyboard will appear.			
2	Search any item	Keyboard will disappear and search results will appear.			
3	Select the searched item	A pop up will appear that shows all the supported sharing methods.			
4	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
5	Click the back button to return from the sharing app.	User is returned to the select item screen.			

*Note: Repeat for each sharing method.

Test Case Number: 13 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Exits share screen.	Test Case Name: Exit Share. Subsystem: Share. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button.	The select category screen will appear.			
2	Click the back button.	Main menu will appear.			



Test Case Number: 14 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Trade category and all content with other user's.	Test Case Name: Trade via the side menu. Subsystem: Main menu. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the more icon on the main menu	Side menu will appear.			
2	Click on "Trade" in the side menu	A select category screen will appear.			
3	Click on appropriate category.	A Select Item screen will appear.			
4	Select an item to share.	A pop up will appear that shows all the supported sharing methods.			
5	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
6	Click the back button to return from the sharing app.	User is returned to the select item screen.			

Test Case Number: 15 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Searches for items by (name, category, details).	Test Case Name: Search Item to Trade via the side menu. Subsystem: Trade. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon in the select item screen	Search text field will slide from the side and keyboard will appear.			
2	Search any item	Keyboard will disappear and search results will appear.			
3	Select the searched item	A pop up will appear that shows all the supported sharing methods.			
4	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
5	Click the back button to return from the sharing app.	User is returned to the select item screen.			

Test Case Number: 16 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Exits Trade screen.	Test Case Name: Exit Trade via side menu. Subsystem: Trade. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button	The select category screen will appear.			
2	Click the back button	Main menu will appear.			

Test Case Number: 17 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Accessing help & tips for the app via the side menu.	Test Case Name: Help and Tips via Side menu. Subsystem: Side menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on "Help and Tips" in the side menu	A Help and Tips screen will appear.			
2	Click on the video.	A tutorial video will play.			
3	Click on the Send feedback button	User is taken to google play store to leave feedback at google play store.			
4	Click the back button to leave play store.	The help and tip screen appear.			
5	Click the back button in the help and tips screen.	User is taken back to the main menu.			

Test Case Number: 18 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Gives a brief history on the development team for eyeRS and the app via the side menu.	Test Case Name: About via the side menu. Subsystem: Side menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on "About" in the side menu.	A About screen will appear. Which contains the information about the app.			
2	Click the back button.	The settings screen appears.			



Test Case Number: 19 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Displaying of all possible sections for configuration of the app.	Test Case Name: Settings. Subsystem: Main menu. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.			
2	Click on "Settings" in the side menu	An settings screen will appear			

* Note Settings can be accessed by clicking on the more button on the right side of the screen. Repeat setting test for the settings accessed via the main menu.

Test Case Number: 20 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Security settings that are provided can now be configured.	Test Case Name: Security Settings. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Security	A security settings screen will appear.			
2	Click on the Enter your username text field	Keyboard appears			
3	Enter username	The username will appear in the text fiel. Note it is not yet saved.			
4	Click on the Enter your new PIN here text field	Keyboard appears			
5	Enter the new PIN	Pin display as dots in the text field. Not it is not yet saved.			
6	Select drop down box	A list of all possible security questions will appear.			



7	Select a security question from the list.	New security question will appear in the security question box.			
8	Click on the Security Response text field.	A keyboard appears.			
9	Enter the security response in the text field	The security question will appear in the text field. Note: Data is not yet saved.			
10	Click on the Clear button.	All the entered information are erased. PIN reset screen remain on the screen.			
11	Enter all the reset PIN information.	All text will be displayed.			
12	Click the reset button	Successful notification will appear that. User will be taken back to the settings screen.			



Test Case Number: 21 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Displays settings that can be configured.	Test Case Name: Display Settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on display	A display setting screen will appear.			
2	Click on the new colour block *To be repeated for each colour available	The colour of the app will change.			
3	Click on the select Font drop down box.	A list of all the available font will appear.			
4	Select a font in the list	The font of the app will change.			
5	Click on the select font size drop down box.	A list of all possible font sizes will appear.			
6	Click on the back button.	The Setting screen will appear.			

Test Case Number: 22 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: User profile can be configured. Upload a profile picture via the camera.	Test Case Name: Profile Settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Profile	A profile setting screen will appear.			
2	Click on the profile picture.	Notification appear asking to choose upload media source selection.			
3	Click on cancel	The request is canceled.			
4	Click on the profile picture.	Notification appear asking to choose upload media source selection.			
5	Click on camera	Camera opens-follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click on the Please enter your display username text field.	A keyboard appears.			
7	Enter username in the username text field	New user name will appear Note: Data is not yet saved.			
8	Click on the save changes button	Successful notification appears. Profile setting screen still appears.			

Test Case Number: 23 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: User profile can be configured. Upload a profile picture via file upload.	Test Case Name: Profile Settings add profile picture via a file upload. Subsystem: Settings. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the profile picture.	Notification appear asking to choose upload media source selection.			
2	Click on Choose from Library.	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
3	Click on the Please enter your display username text field.	A keyboard appears.			
4	Enter username in the username text field	New user name will appear Note: Data is not yet saved.			
5	Click on the save changes button	Successful notification appears. User is still at profile settings.			
6	Click the back button.	User is returned to the settings option screen.			

Test Case Number: 24 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: The sound for the app can be configured.	Test Case Name: Sound Settings. Subsystem: Settings. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on sound	A sound setting screen appears			
2	Click on the welcome message toggle	The toggle turns off and welcome message will not play next time the user access the app.			
3	Click on the touch sound toggle	The toggle turns of and the touch sound will not play.			
4	Click on the back button	The Setting Option screen will appear.			

Test Case Number: 25 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: The management of the categories in the app.	Test Case Name: Category Management. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Category Management settings	A category management screen will appear			

Test Case Number: 26 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Adding a new category while in settings.	Test Case Name: Add Category via Settings. Subsystem: Category Management. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.			
2	Click on “Add new category” in the side menu	An Add Category screen will appear.			
3	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Category Name text field	Keyboard appears.			
7	Enter the Name of the category in the category text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the category in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.			
11	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			



12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new category detail are before.	Category detail displays.			
14	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.			
15	Click on the back button.	Category management option screen appears.			



Test Case Number: 27 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Editing a category.	Test Case Name Edit category. Subsystem: Category Management. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click Edit category	List with current categories appear.			
2	Click on the category you wish to edit	Category information will appear.			
3	Click on the category icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Category Name text field	Keyboard appears.			
7	Enter the Name of the category in the category text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the category in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the “Save changes” button.	Category is edited in the database. Notification of saved action. User remains in add the Add category screen.			
11	Click on the category icon.	Notification appears asking for an upload of the image via the camera or the gallery.			



12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new category detail are before.	Category detail displays.			
14	Click the “Save changes” button.	Category is edited in the database. Notification of saved action. User remains in add the Add category screen.			
15	Click on the back button.	Category management option screen appears.			



Test Case Number: 28 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Removing a category from the catalogue.	Test Case Name: Delete category. Subsystem: Category Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Delete Category	A category selection screen appears.			
2	Select a category to delete.	Confirmation notification appears.			
3	Click No on the notification	Category selection screen appears.			
4	Select a category to delete	Confirmation notification appears.			
5	Select Yes on the notification	Category is deleted in the database. Successful notification appears.			
6	Click the back button	The category management option screen appears.			



Test Case Number: 29 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows user to sort the categories in multiple ways.	Test Case Name: Change Category Sorting. Subsystem: Category Management. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Change sorting	A sorting method screen appears.			
2	Click on the alphabetically toggle	Toggle turns off/on. Categories will be listed alphabetically. Can be viewed in the main menu.			
3	Click on the firstly added first toggle	Toggle turns on/off Categories is listed in a first added first view manner. Can be viewed in the main menu.			
4	Click on the Lastly added first toggle	Toggle turns on/off Categories is listed in a last added view first manner. Can be viewed in the main menu.			
5	Click on the back button.	The category management screen appears.			

Test Case Number: 30 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to return settings screen.	Test Case Name: Exit Category Management settings. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button.	The setting screen appears.			



Test Case Number: 31 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to manage their items in the catalogue.	Test Case Name: Item Management. Subsystem: Settings. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Item management	The item management screen appears.			

Test Case Number: 32 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to add a an item to the catalogue.	Test Case Name: Add item via Item Management. Subsystem: Item management. Design Date: Execution Date:
--	---

1	Click on “Add new item”	An Add Item screen will appear.			
2	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			
3	Click on Cancel	No photo were added.			
4	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			
5	Click on Take Photo	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Select the category drop box	A list of the categories fill appears.			
7	Select the appropriate category from the drop box list	The selected category will appear. (Note: Not yet saved).			
8	Click in the Name of item text field	Keyboard appears.			
9	Enter the title of the item in the item text field	Title will appear in the text field. (Note: Not yet saved).			
10	Click in the Item description text field	Keyboard appears.			
11	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
12	Click the “Add” button.	Item is added in the database. Notification of saved appears. User remains in the add item screen			



13	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
14	Enter the new item detail as before.	Item detail displays.			
15	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add item screen.			
16	Click on the back button.	Item management options appears.			

Test Case Number: 33 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to edit an item in the catalogue.	Test Case Name: Edit item via Item Management. Subsystem: Item Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click Edit Item	List with current items appear.			
2	Click on the item you wish to edit	Item information will appear.			
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Item Name text field	Keyboard appears.			
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.			
11	Click on the item icon.	Notification appears asking for an upload of the image via the camera or the gallery.			
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the			



		chosen picture— picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new item detail are before.	Item detail displays.			
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.			
15	Click on the back button.	Item management option screen appears.			



Test Case Number: 34 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to delete an item in the catalogue.	Test Case Name: Delete item via Item Management. Subsystem: Item Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Delete Item	A category selection screen appears			
2	Click on the category that contain the item that will be deleted.	A item selection screen will appear.			
3	Click on the item you want to delete.	Delete notification appear for confirmation.			
4	Click on No	Item is not deleted. User remains in the item selection screen			
5	Click on the item you want to delete.	Delete notification appear for confirmation.			
6	Click on Yes	Item will be deleted from the database and the item will no longer be visible in the item list. User remains in the select item screen.			



Test Case Number: 35 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to delete an item in the catalogue.	Test Case Name: Delete item via Item Management with search as assistance. Subsystem: Item Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon	Keyboard will appear and a text field will open.			
2	Search for an item	Search item will appear if it was found else it will display no result found			
3	Click on the item you want to delete.	Delete notification appear for confirmation.			
4	Click on No	Item is not deleted. User remains in the item selection screen			
5	Click on the item you want to delete.	Delete notification appear for confirmation.			
6	Click on Yes	Item will be deleted from the database and the item will no longer be visible in the item list. User remains in the select item screen.			
7	Click on the back button.	The item management option screen appears.			



Test Case Number: 36 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allowing user to sort an item in the catalogue.	Test Case Name: Sorting the items in the catalogue. Subsystem: Item Management. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Change sorting	A sorting method screen appears.			
2	Click on the alphabetically toggle	Toggle turns off/on. Items will be listed alphabetically.			
3	Click on the firstly added first toggle	Toggle turns on/off Items is listed in a first added first view manner.			
4	Click on the Lastly added first toggle	Toggle turns on/off Items is listed in a last added view first manner.			
5	Click on the back button.	The item management screen appears.			

Test Case Number: 37 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Returning to the main settings screen.	Test Case Name: Exiting Item Management. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the back button	The settings screen appears.			

Test Case Number: 38 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Accessing help & tips for the app via settings.	Test Case Name: Help and Tips via Settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on "Help and Tips"	A Help and Tips screen will appear.			
2	Click on the video.	A tutorial video will play.			
3	Click on the Send feedback button	User is taken to google play store to leave feedback at google play store.			
4	Click the back button to leave play store.	The help and tip screen appear.			
5	Click the back button in the help and tips screen.	User is taken back to the settings screen.			

Test Case Number: 39 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Gives a brief history on the development team for eyeRS and the app via settings.	Test Case Name: About via settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on "About"	A About screen will appear. Which contains the information about the app.			
2	Click the back button.	The settings screen appears.			



Test Case Number: 40 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Returning to the main menu to proceed with other functions that the app provide.	Test Case Name: Exit the settings. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button.	Main menu appears.			

Test Case Number: 41 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Searches for items in the catalog from the main screen.	Test Case Name: Search Item from the main screen. Subsystem: Main Menu. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon	Search text field will slide from the side and keyboard will appear.			
2	Search any item	Keyboard will disappear and search results will appear.			



Test Case Number: 42 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: View and edit items from the catalog directly from the search feature.	Test Case Name: View and Edit Item direct from the search feature. Subsystem: Item Detail. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the item you want to view or change.	Item detail screen display.			
2	Click Edit Item button	List with current items appear.			
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Item Name text field	Keyboard appears.			
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.			
11	Click on the item icon.	Notification appears asking for an upload of the image			



		via the camera or the gallery.			
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture— picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new item detail are before.	Item detail displays.			
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.			
15	Click on the back button.	Item detail screen appears.			



Test Case Number: 43 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item once in a searched item.	Test Case Name Share Item once in searched item. Subsystem: Item Detail. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the “Share” option in the pop up screen	Notification that asks you to choose a sharing method.			
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.			

*Note: Repeat for each sharing method.



Test Case Number: 44 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item once in a searched item.	Test Case Name Trade Item once in searched item. Subsystem: Item Detail. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the “Trade” option in the pop up screen	Notification that asks you to choose a sharing method.			
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.			

*Note: Repeat for each sharing method.



Test Case Number: 45 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category.	Test Case Name Delete Item once in searched item. Subsystem: Item Detail. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.			
3	Click the “No” button in the Delete conform notification	Delete confirm notification will disappear and the item detail screen will appear.			
4	Click on the more icon.	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
5	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.			
6	Click the “Yes” button in the Delete conform notification	Deleted notification appears. Item is deleted from the database. Main menu will appear			



Test Case Number: 46 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: View item from a category.	Test Case Name View items via main menu. Subsystem: Main menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on a category to view items.	Items in the category will display in a listview.			



Test Case Number: 47 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category.	Test Case Name Edit items via main menu. Subsystem: Main menu. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the item you want to view or change.	Item detail screen display.			
2	Click Edit Item button	List with current items appear.			
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Item Name text field	Keyboard appears.			
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.			
11	Click on the item icon.	Notification appears asking for an upload of the image via the camera or the gallery.			



12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new item detail are before.	Item detail displays.			
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.			
15	Click on the back button.	Item detail screen appears.			



Test Case Number: 48 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item via main menu.	Test Case Name Share Item via main menu. Subsystem: Item Detail. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the "Share" option in the pop up screen	Notification that asks you to choose a sharing method.			
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.			

*Note: Repeat for each sharing method.

Test Case Number: 49 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item from the main menu.	Test Case Name Trade Item from the main menu. Subsystem: Item Detail. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the "Trade" option in the pop up screen	Notification that asks you to choose a sharing method.			
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.			

*Note: Repeat for each sharing method.

Test Case Number: 50 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category from the main menu.	Test Case Name Delete Item from the main menu. Subsystem: Item Detail. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.			
3	Click the “No” button in the Delete conform notification	Delete confirm notification will disappear and the item detail screen will appear.			
4	Click on the more icon.	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
5	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.			
6	Click the “Yes” button in the Delete conform notification	Deleted notification appears. Item is deleted from the database. Item list appears.			

Test Case Number: 51 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Exit from viewing the items.	Test Case Name: Exit from viewing items Screen. Subsystem: My Category. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the back button	Main screen will appear.			

Test Case Number: 52 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Leaving the eyeRS app.	Test Case Name: Exit eyeRS. Subsystem: Main Menu -Side menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more button n the main menu.	A side menu appears.			
2	Click on "Exit"	A notification will appear.			
3	Click the NO	Main Menu will appear.			
4	Click on the more button n the main menu.	A side menu appears.			
5	Click on "Exit"	A notification will appear.			
6	Click the Yes	User will exit the eyeRS app.			

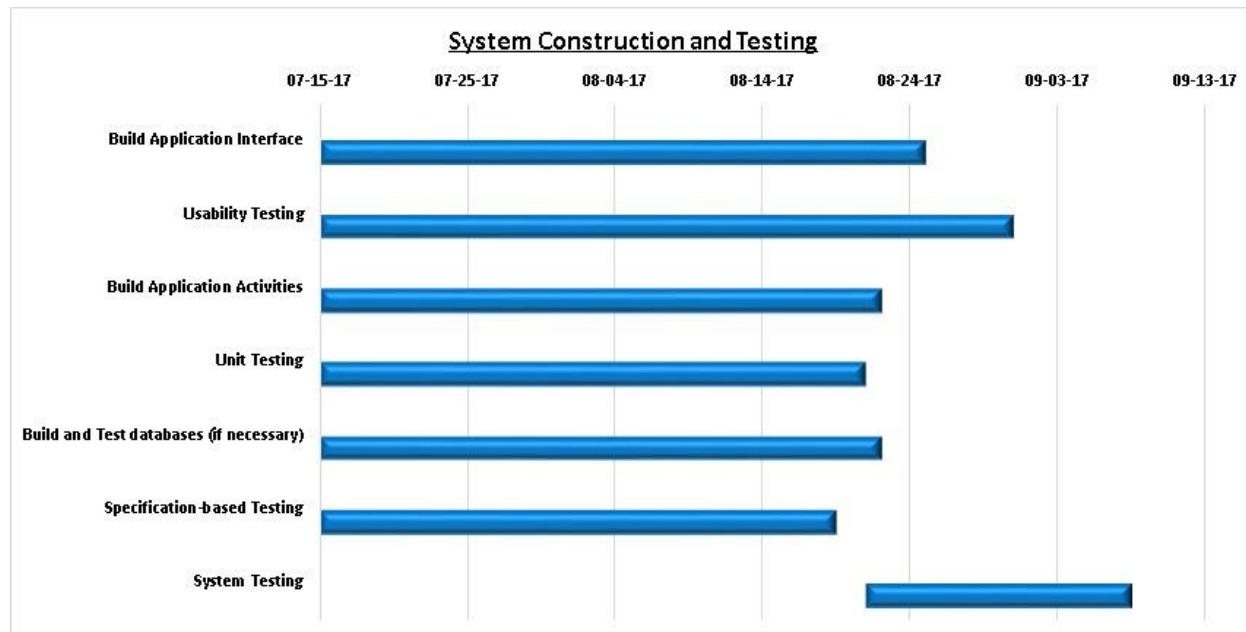
Test plan

Introduction

Our project is a mobile app, which is intended for operation on the Android OS platform. Hardware resources intended for testing will comprise of personal computers for compiling and emulation, together with Android smart-phones and tablets for testing on real devices. Software being used includes emulators, Javadocs, GitHub and the Android Developer console for additional resources. The team is currently using Android Studio which has various mobile and tablet device emulators that will utilize a computer's different Hyper-V features to run the emulators. All resources are available for each member of the eyeRS development team. Testing and debugging could take up to a total of approximately 48 hours.

Test Plan - eyeRS Mobile App			
Test type	Test date	Team members	Description
Unit (White box)	30/08/2017	Mr. Matthew Van Der Bijl, Mr. Nathan Shava	All the activities and/or fragments' code statements will be tested to ensure that they execute as expected.
Unit (Black box)	31/08/2017	Mr. Matthew Van Der Bijl, Mr. Nathan Shava	The app's activities and/or fragments will be tested to verify that each activity can process certain inputs and meet proposed requirements.
Integration	06/09/2017	Mrs. Ndai Makhurstane (Customer), Mr. Matthew Van Der Bijl, Mr. Nathan Shava, Ms. Andrea Cloete, Mr. Sajjaad Ishmail, Mr. Emilde Arsenio	Here the entire app's activities and/or fragments will be integrated to verify whether they interact all as expected.
System & Stress	07/09/2017	Mrs. Ndai Makhurstane (Customer), Mr. Matthew Van Der Bijl, Mr. Nathan Shava, Ms. Andrea Cloete, Mr. Sajjaad Ishmail, Mr. Emilde Arsenio	The performance of the system will be tested at this phase to find out whether the app's responsiveness, for example, is at an acceptable level.

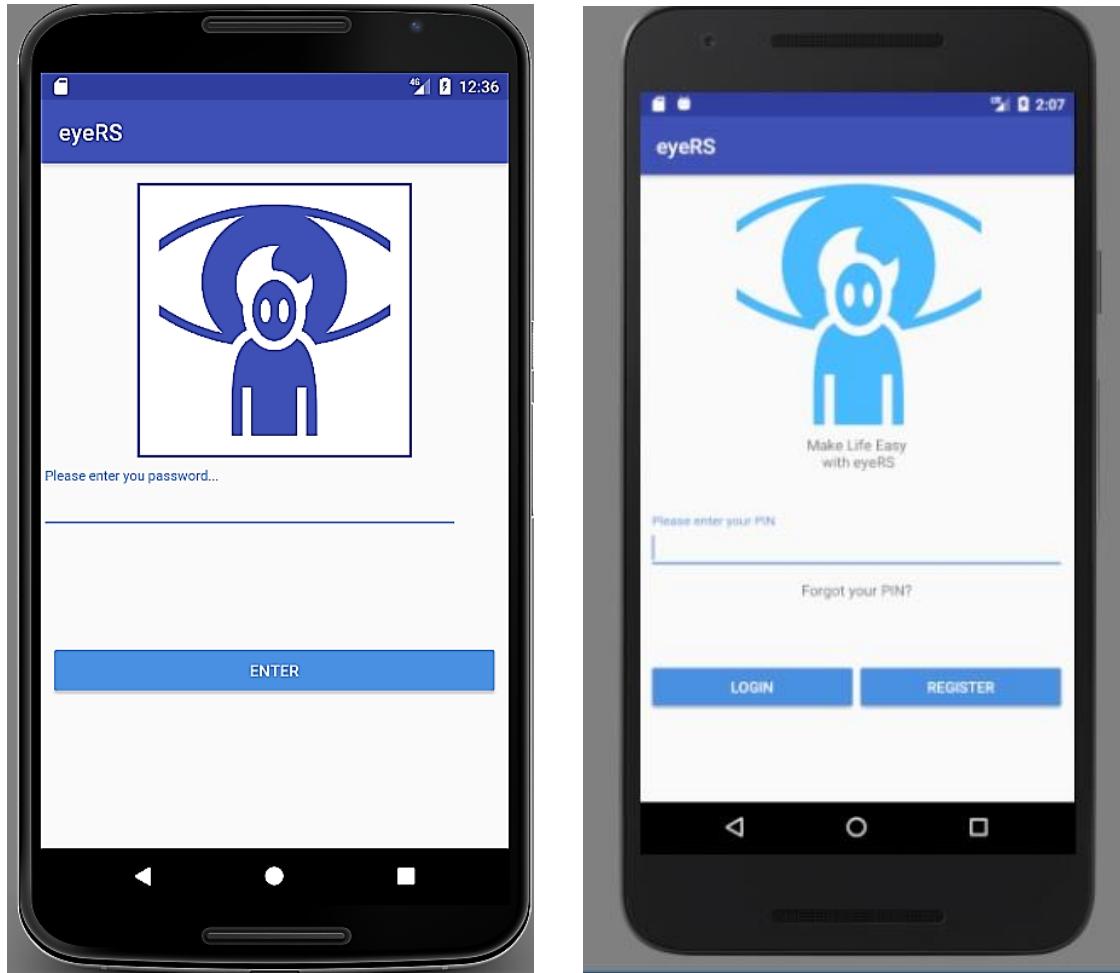
Entire App Walkthrough (Usability)	07/09/2017	Mrs. Ndai Makhurane (Customer), Mr. Matthew Van Der Bijl, Mr. Nathan Shava, Ms. Andrea Cloete, Mr. Sajjaad Ishmail, Mr. Emilde Arsenio	The entire app will be tested simulating the entire user experiences to debug for any other errors not detected during the other phases.
Acceptance	13/09/2017	Mrs. Ndai Makhurane (Customer), Mr. Matthew Van Der Bijl, Mr. Nathan Shava, Ms. Andrea Cloete, Mr. Sajjaad Ishmail, Mr. Emilde Arsenio	The proposed customer will test the app to verify if it has met all the stipulated requirements.
Review & Corrections	14/09/2017	Mr. Matthew Van der Bijl, Mr. Nathan Shava	Time allocated for corrections based on the review process.

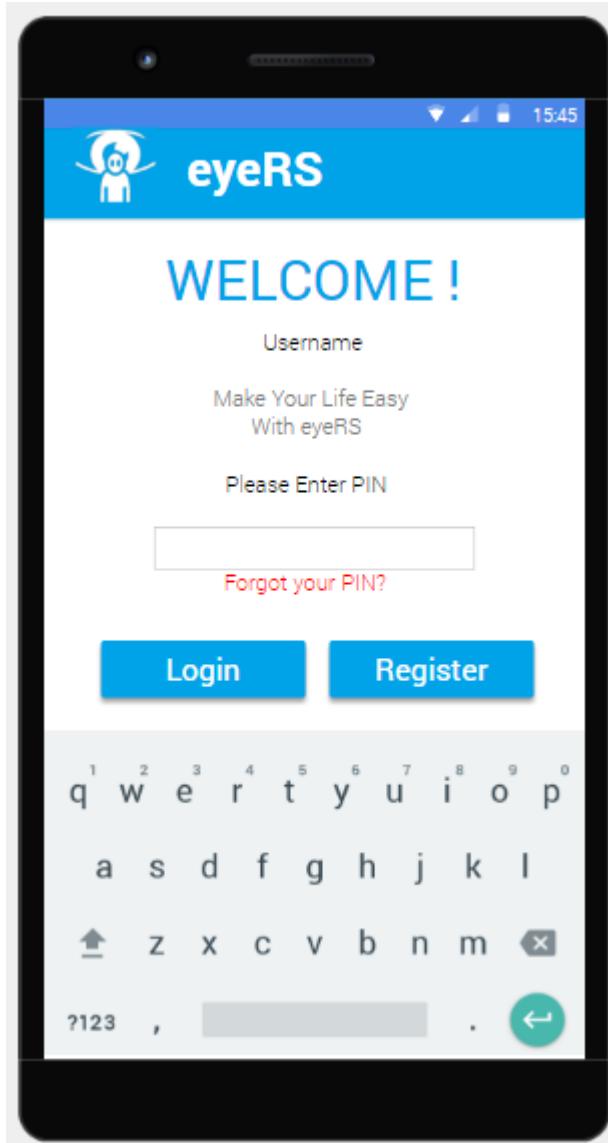


1.2.3 System interface design

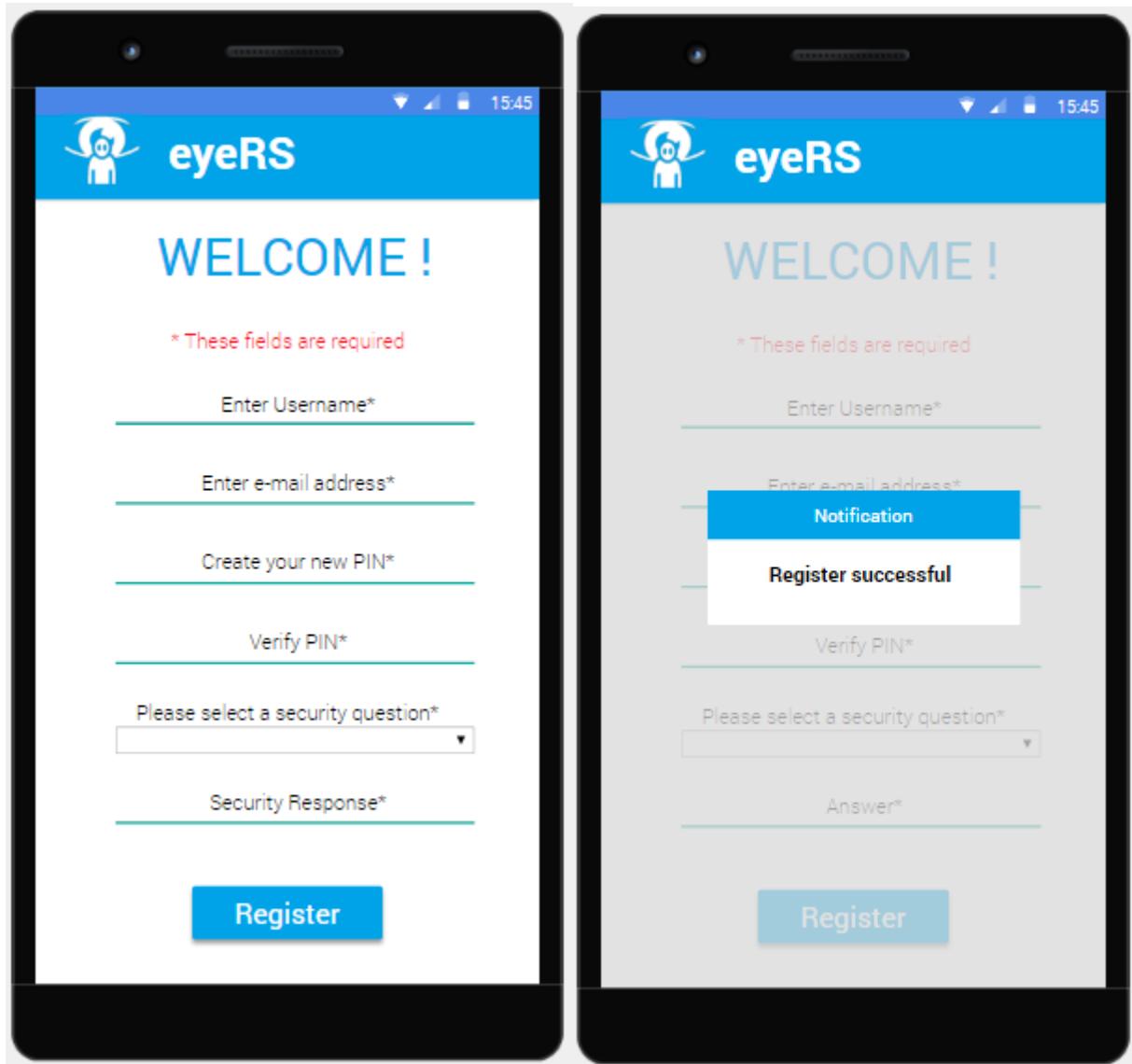
Login Screen

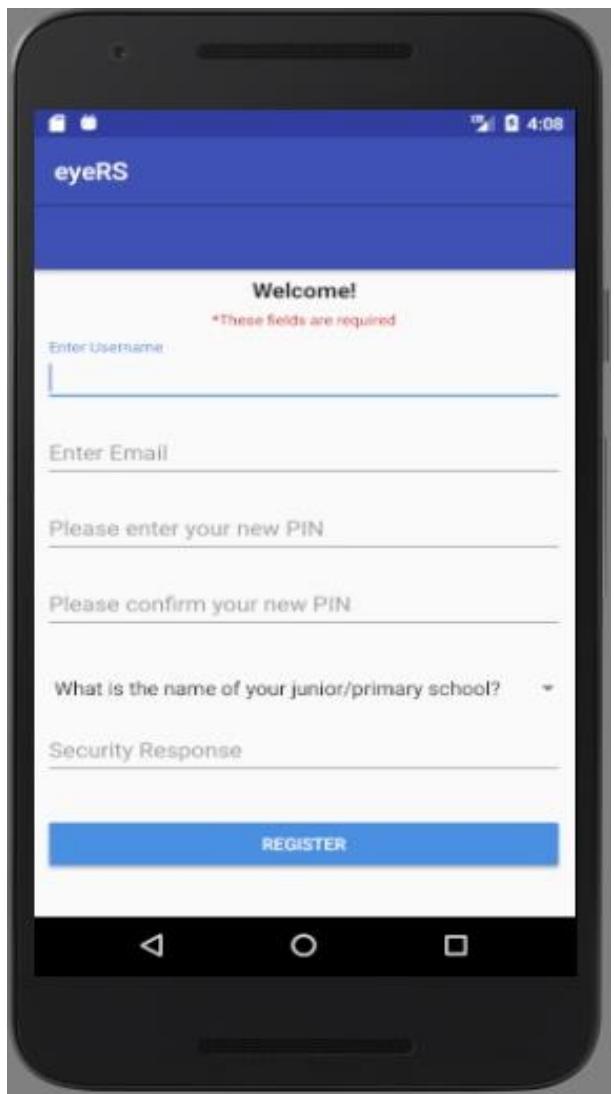
This screen will allow the user to gain access to the app by only supplying a PIN as there can only be one sole user of the app per device.



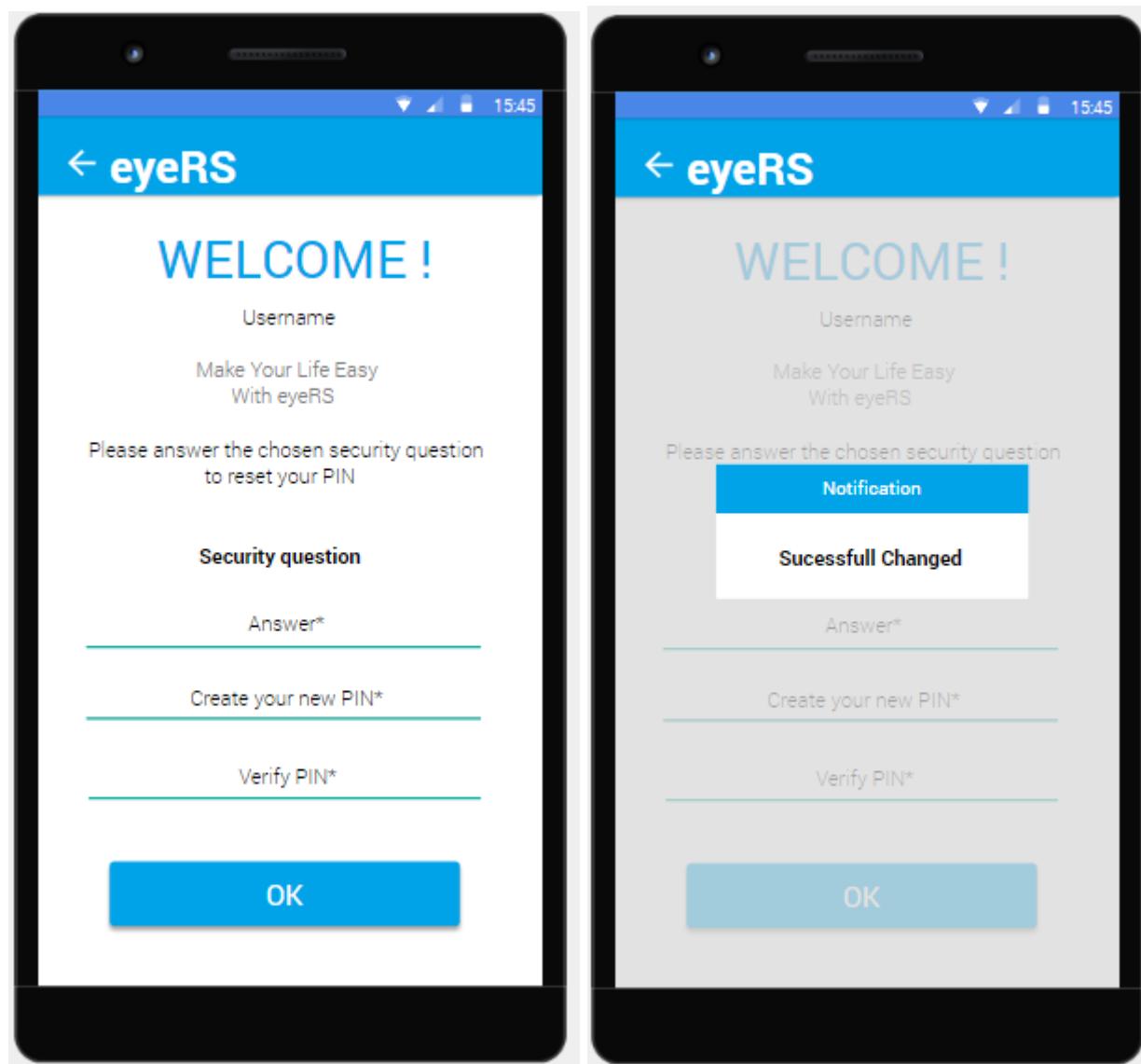


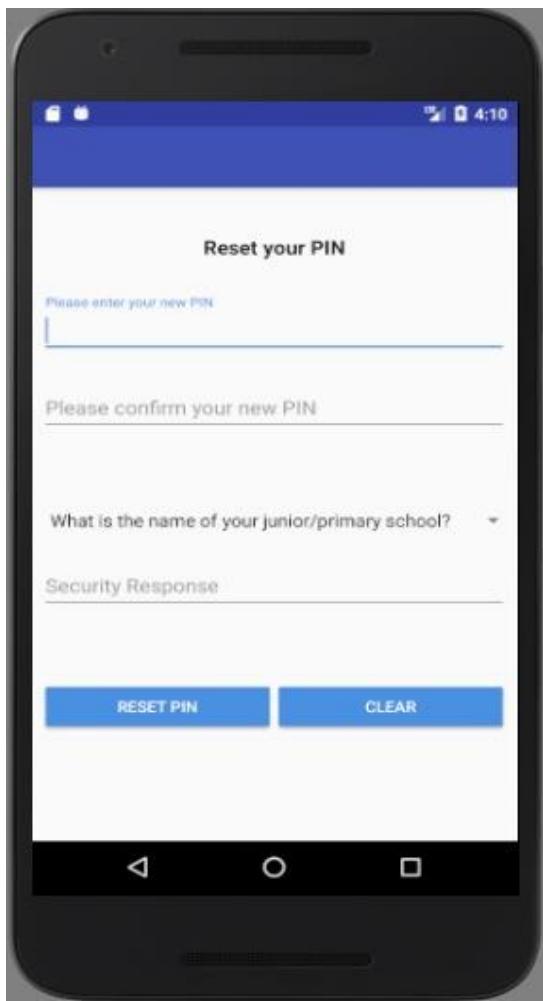
Register Screen





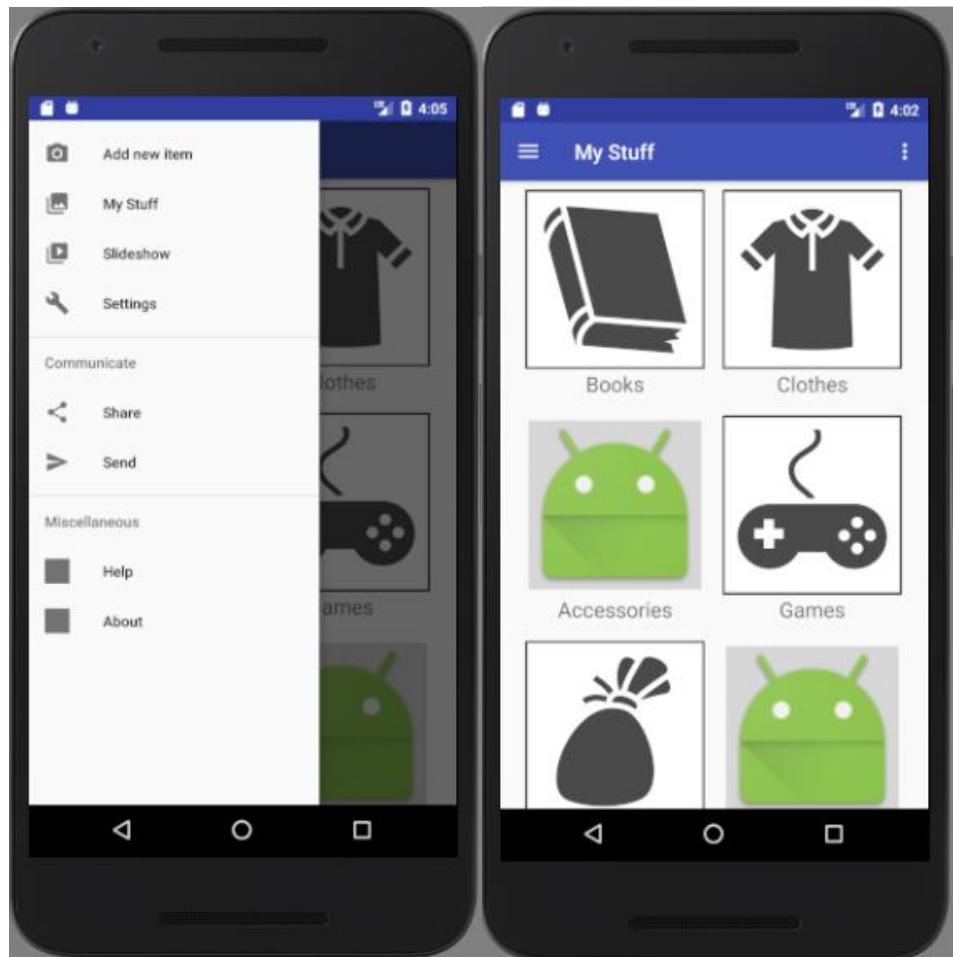
Forgot PIN Screen



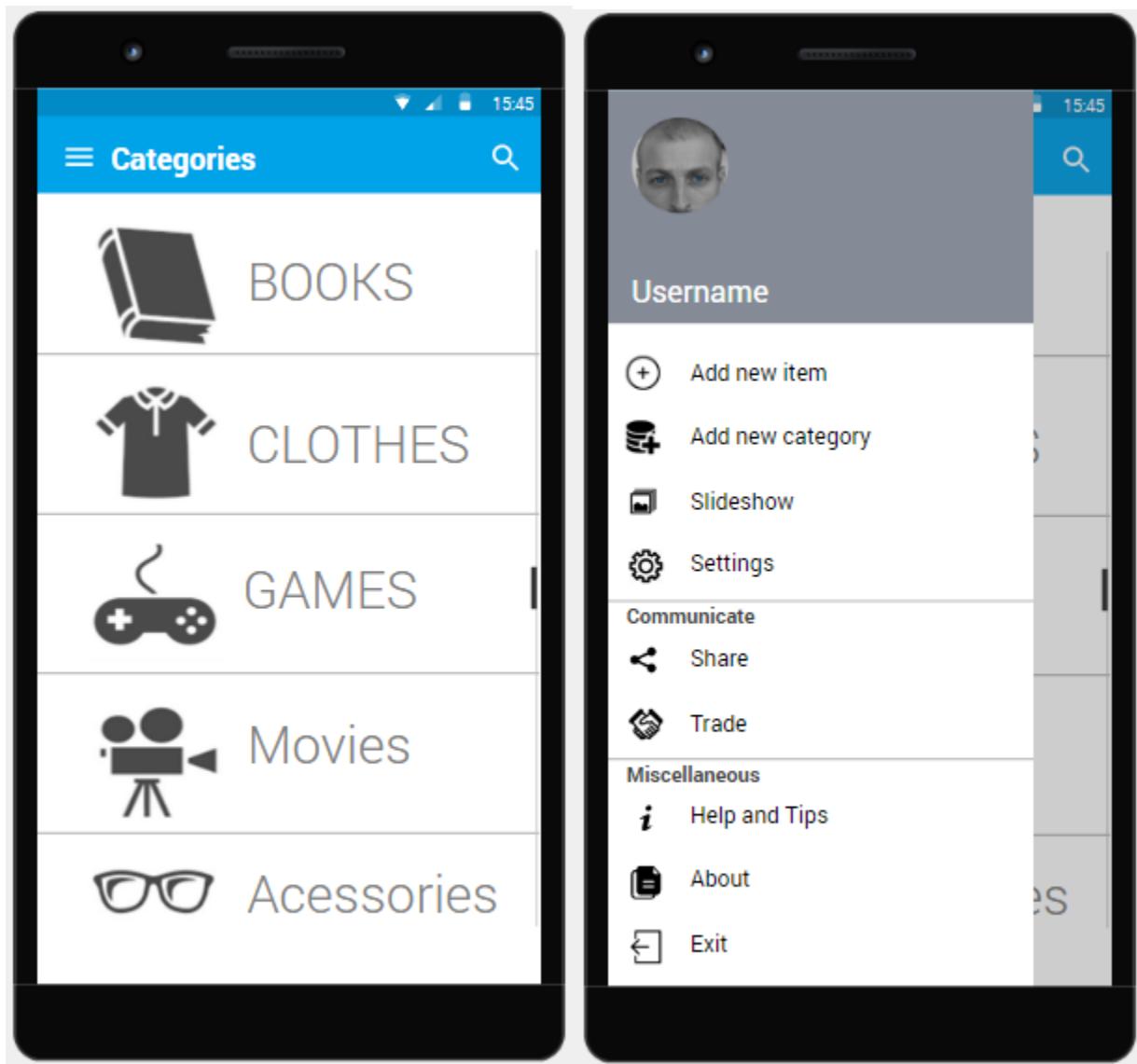


Main Menu Screen

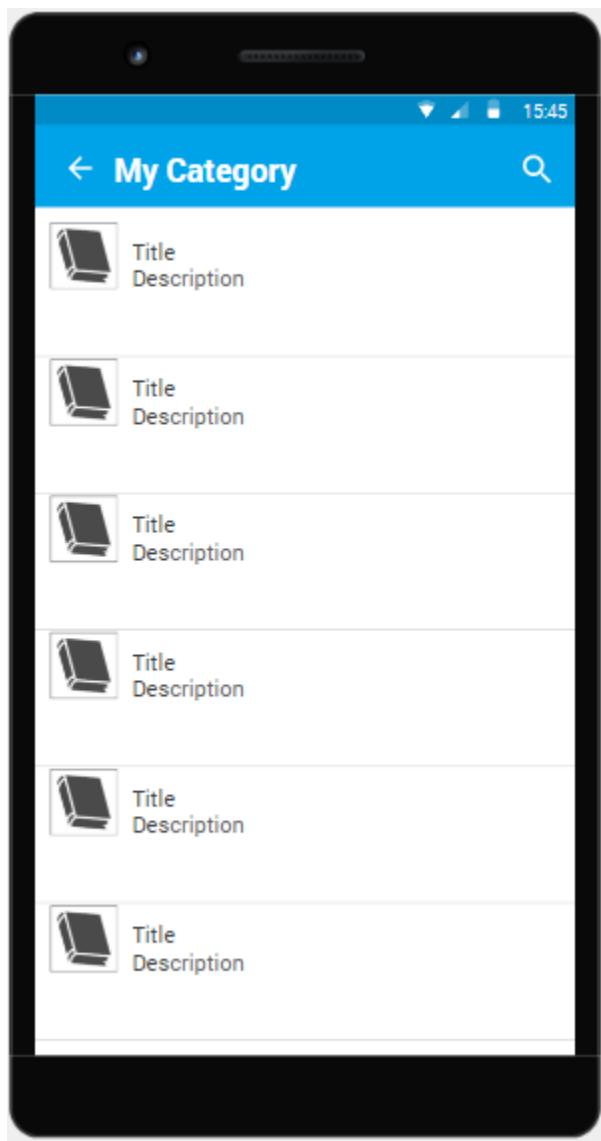
This screen will consist of a Navigation drawer in which the user can access by swiping right from the left edge of the screen or by clicking on the drawer button found on the top left corner of the screen on the action bar.



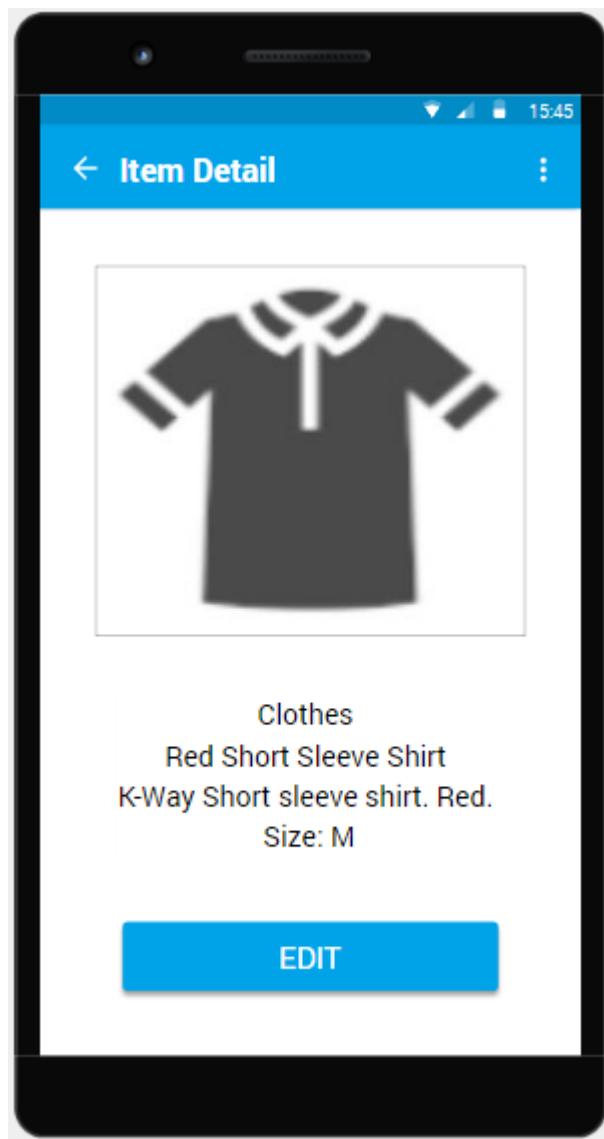
The My Stuff screen also features a graphical UI consisting of clickable icons representing the app's catalog categories.



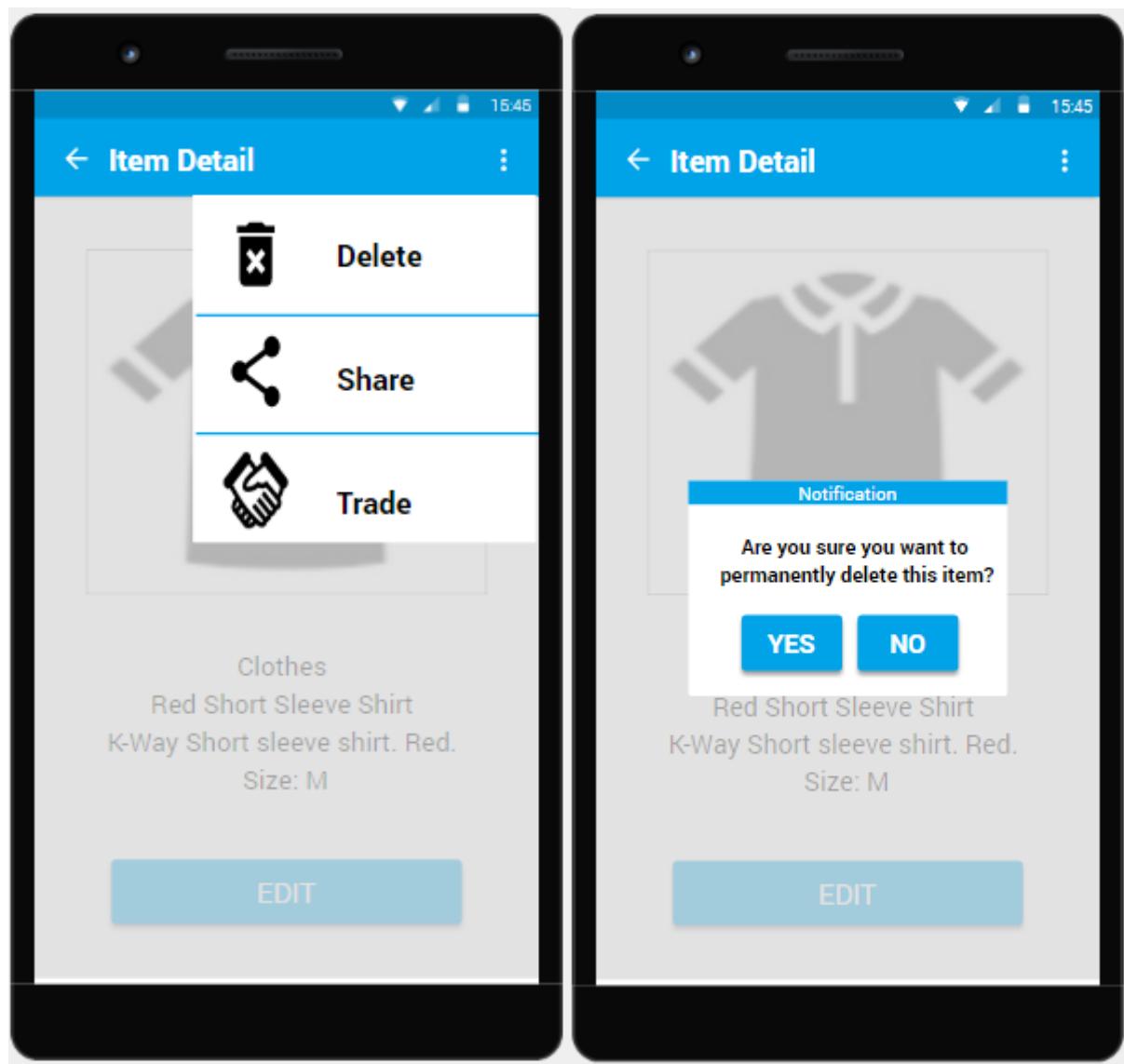
View Item Screen



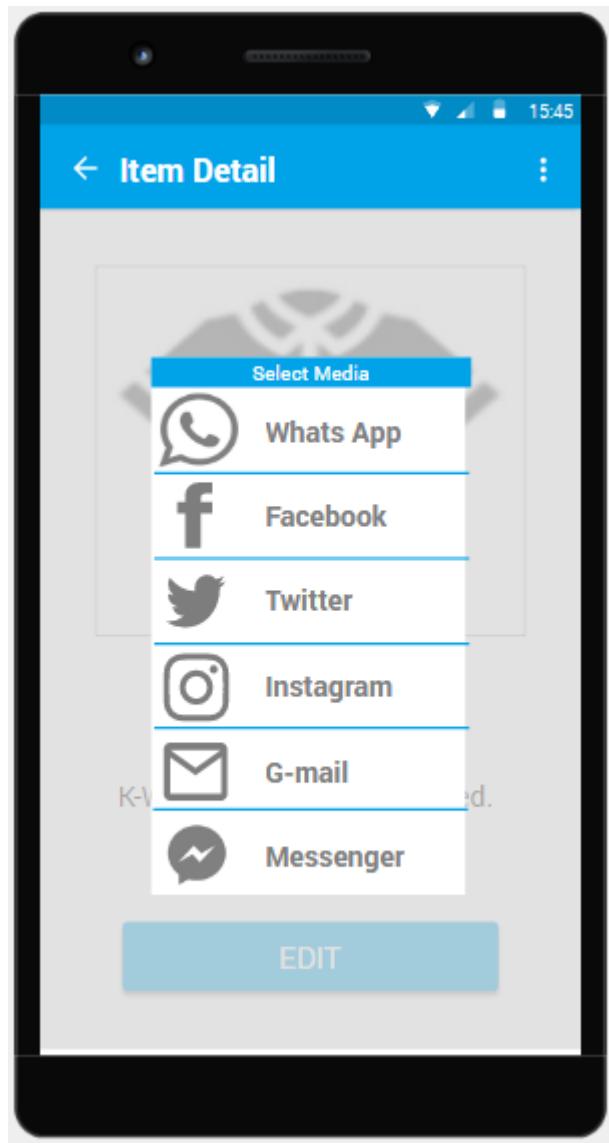
Item Detail Screen



Item Delete Screen

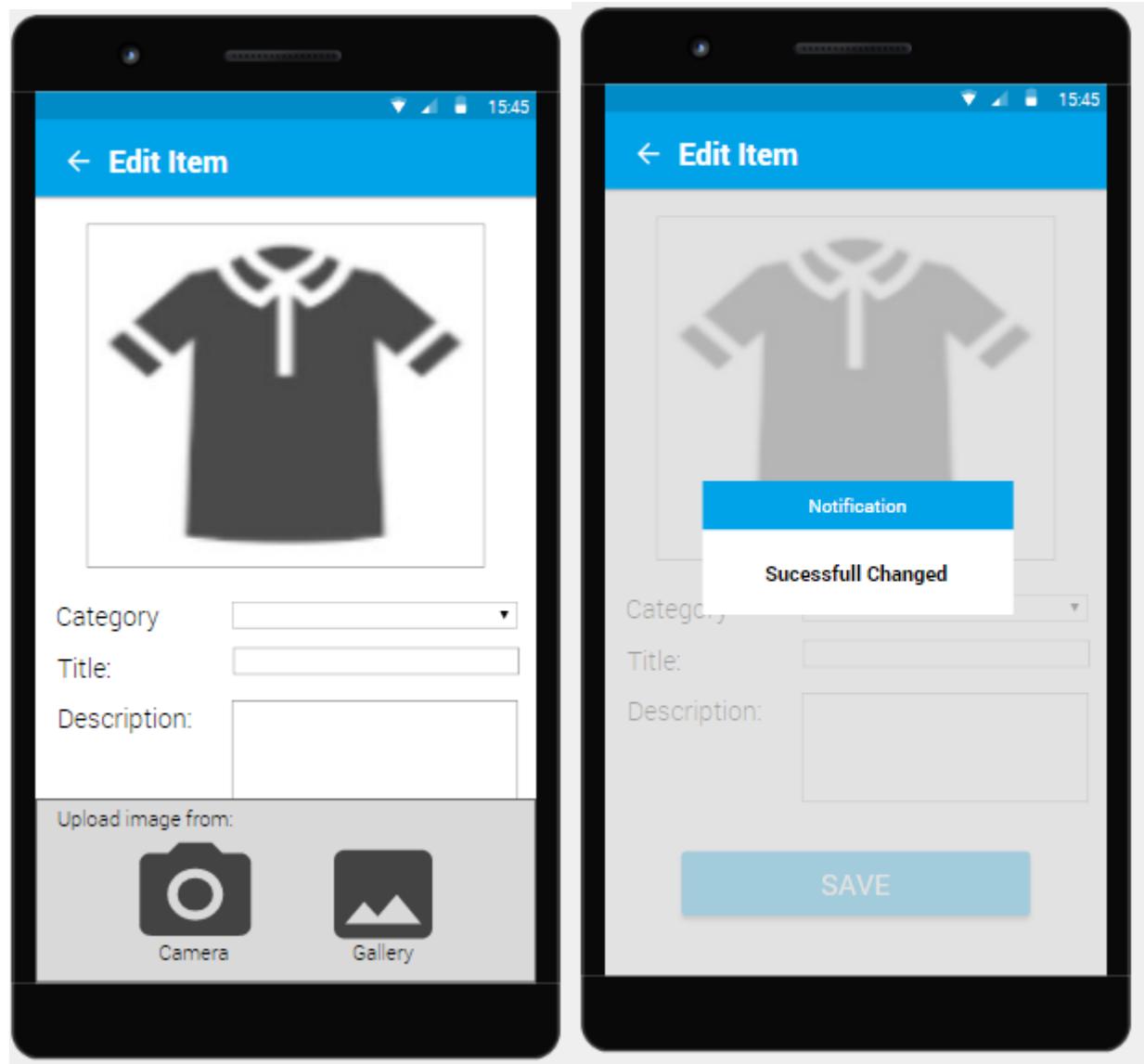


Item Share and Trade Screen

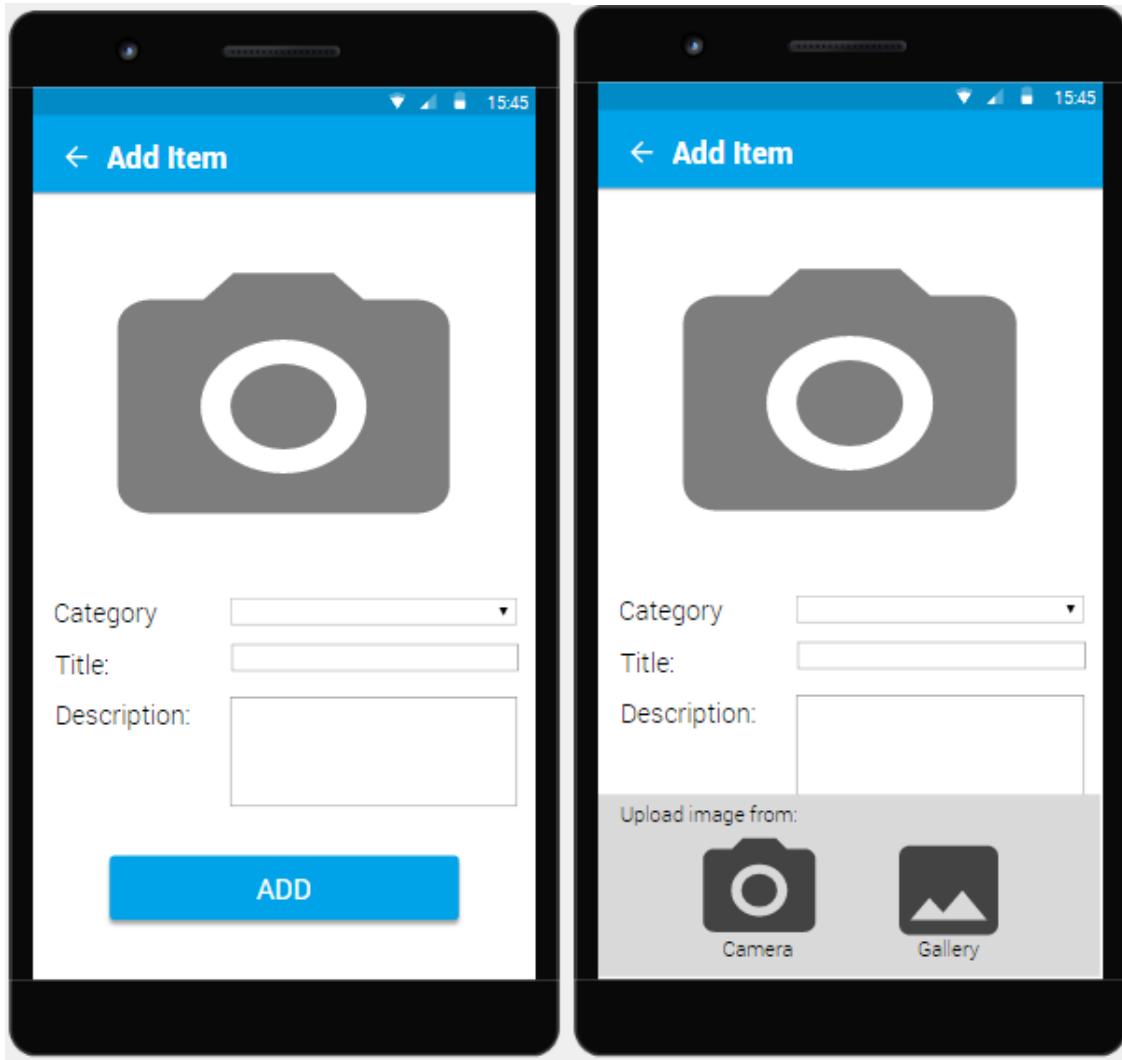


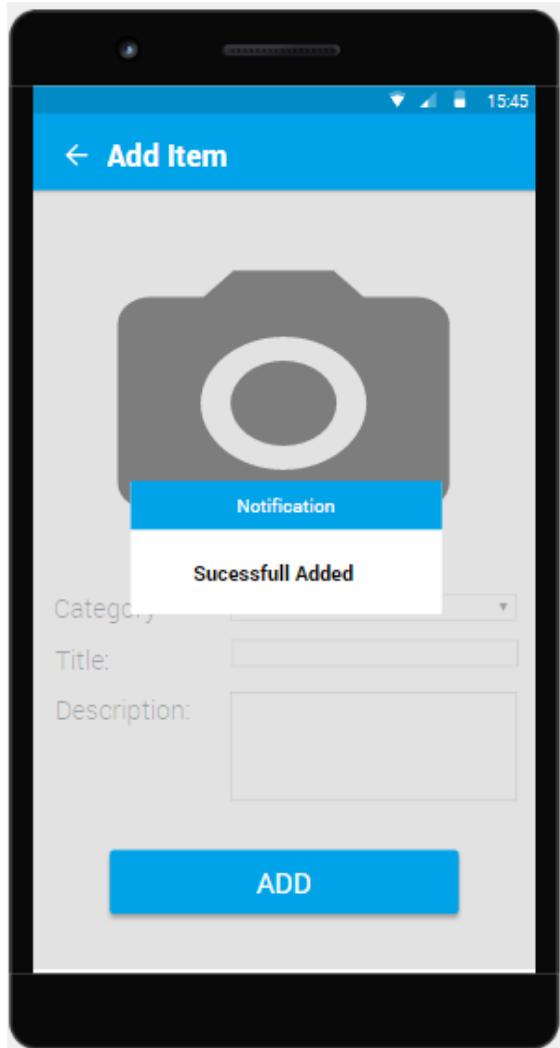
Edit Item Screen



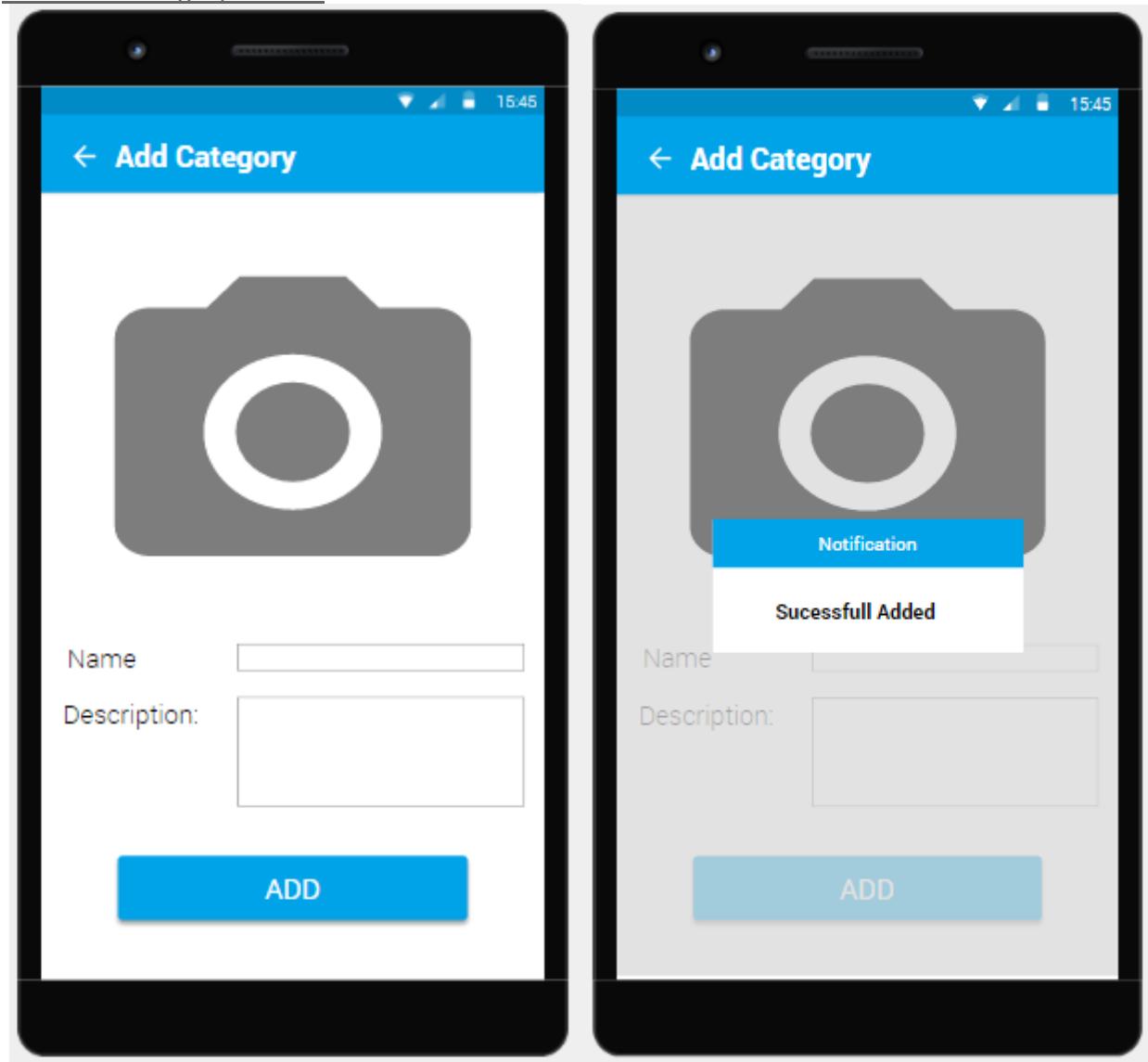


Add Item Screen





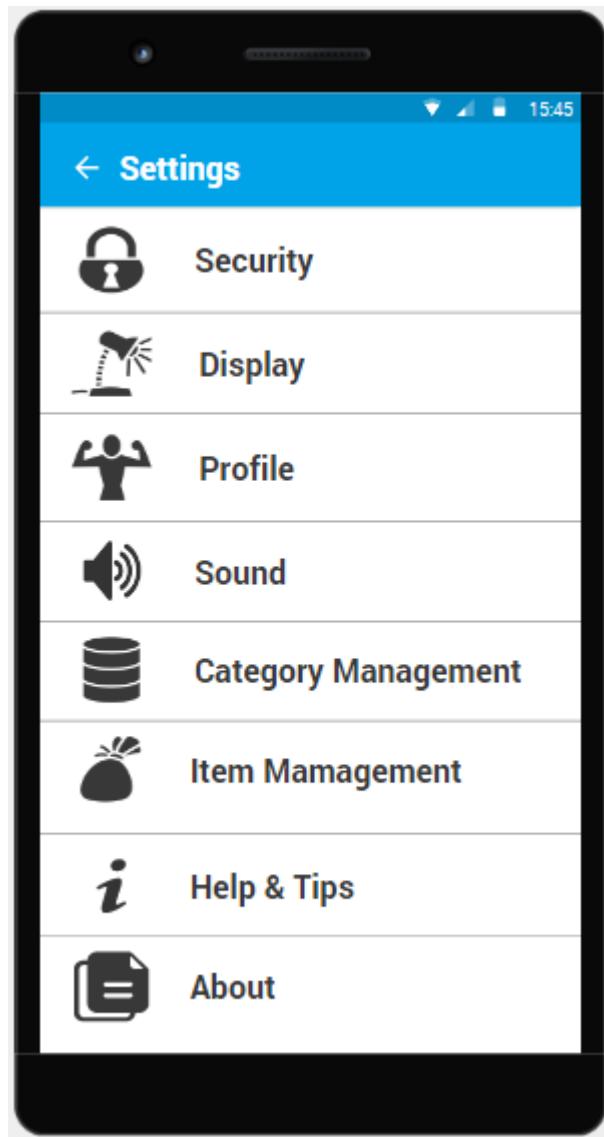
Add New Category Screen



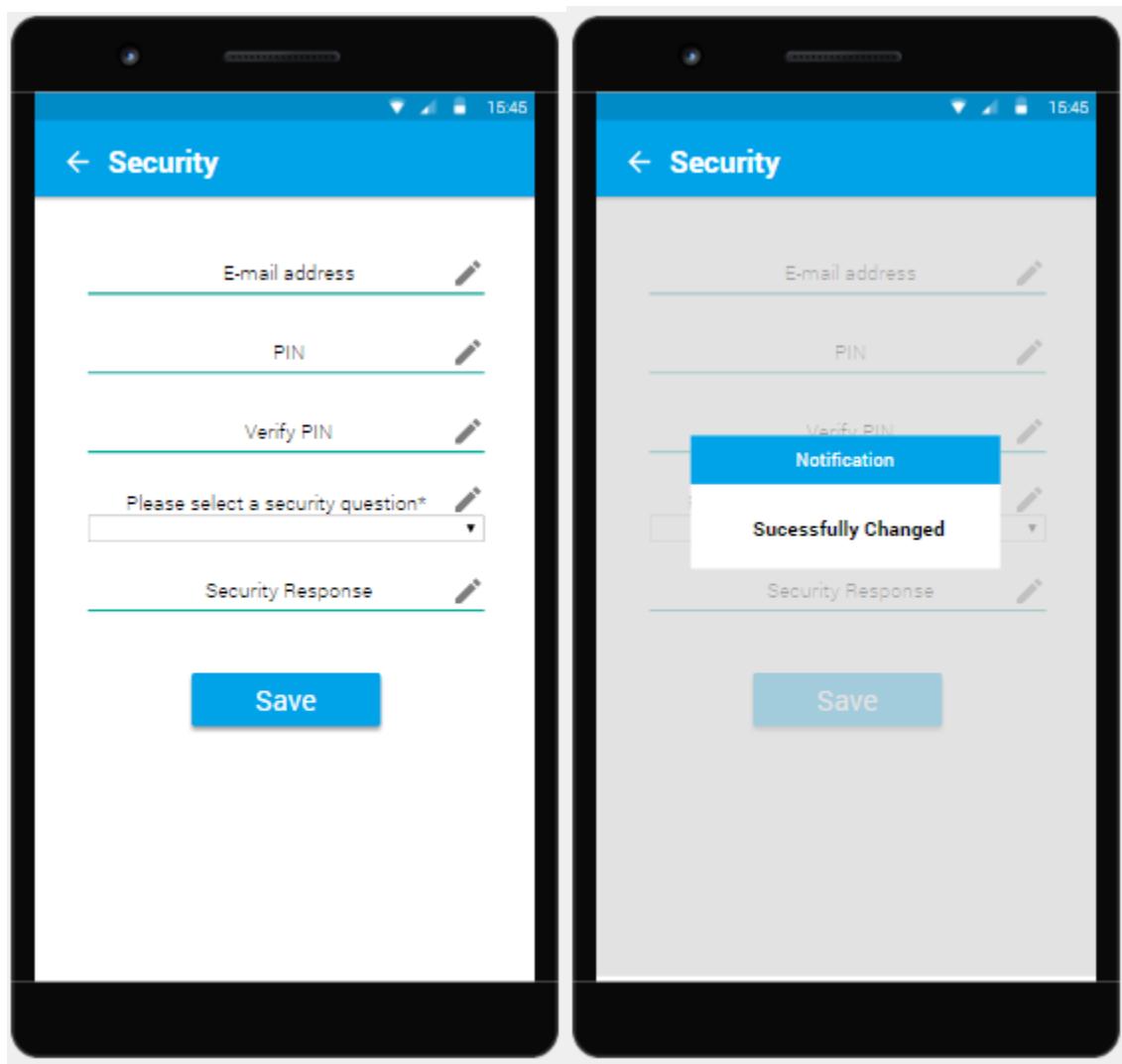
Slideshow Screen



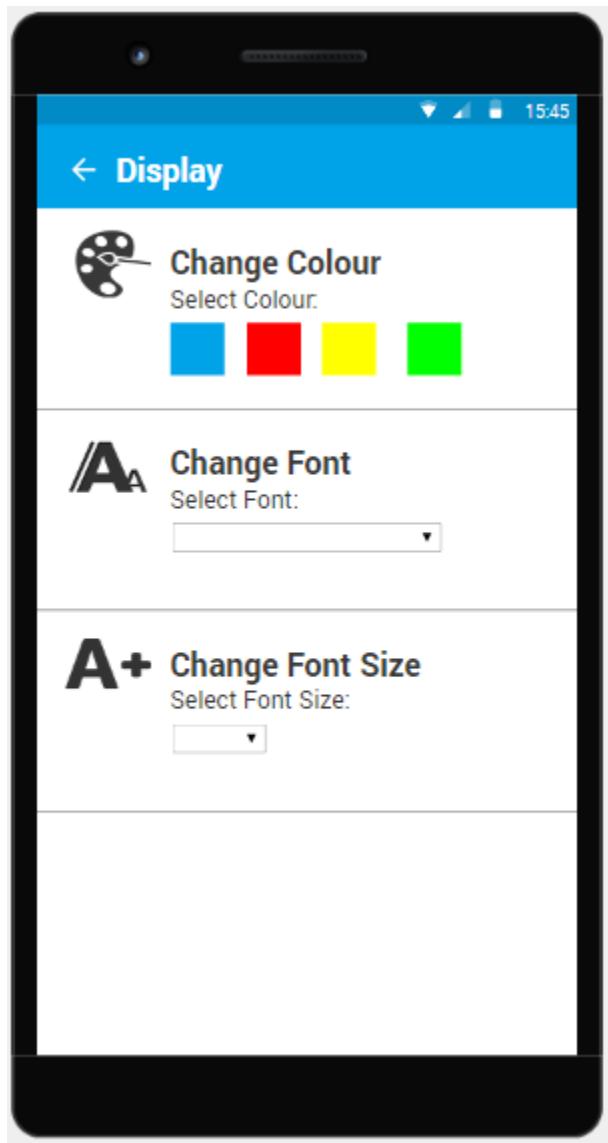
Settings Screen



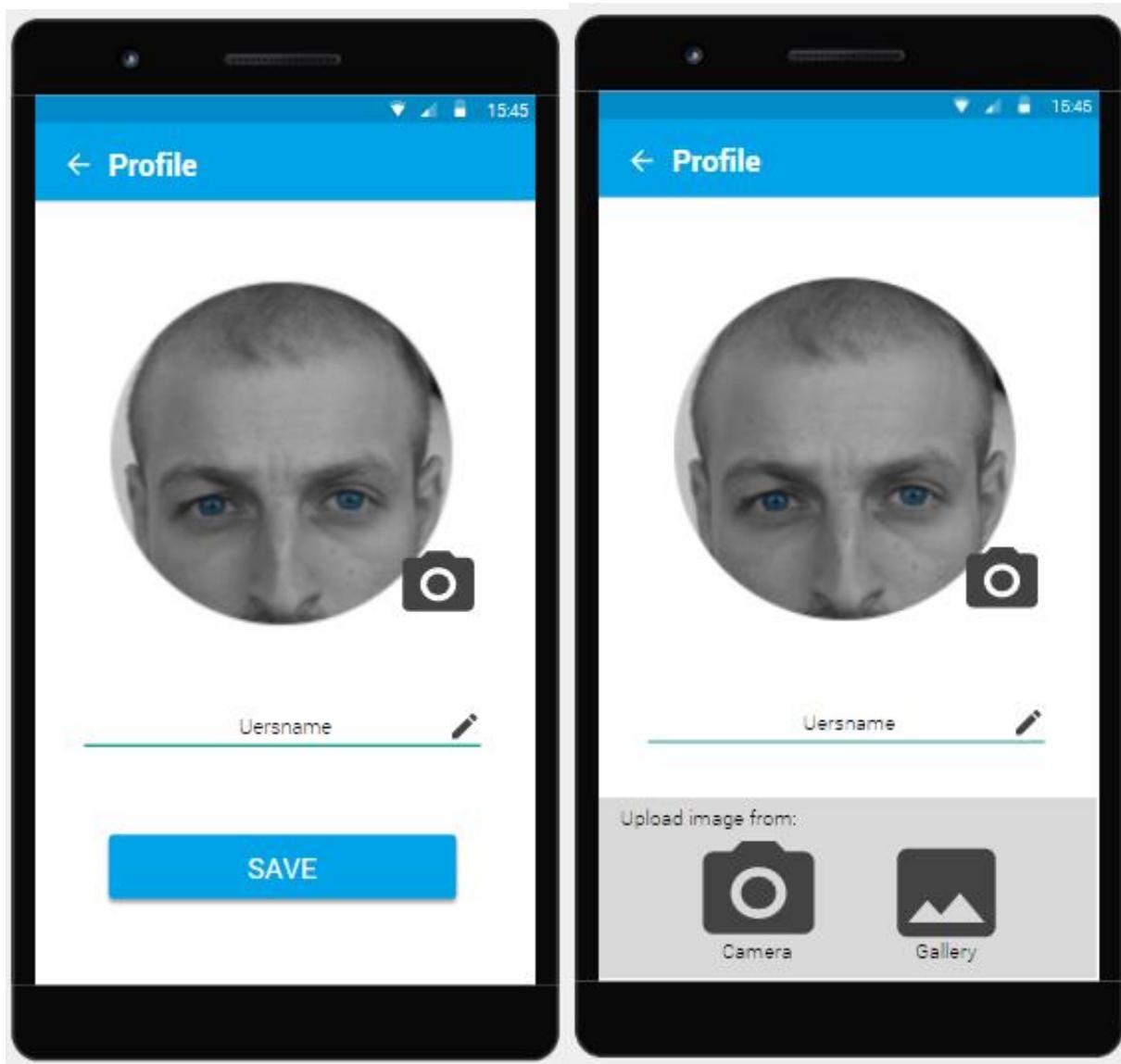
Settings > Security Screen



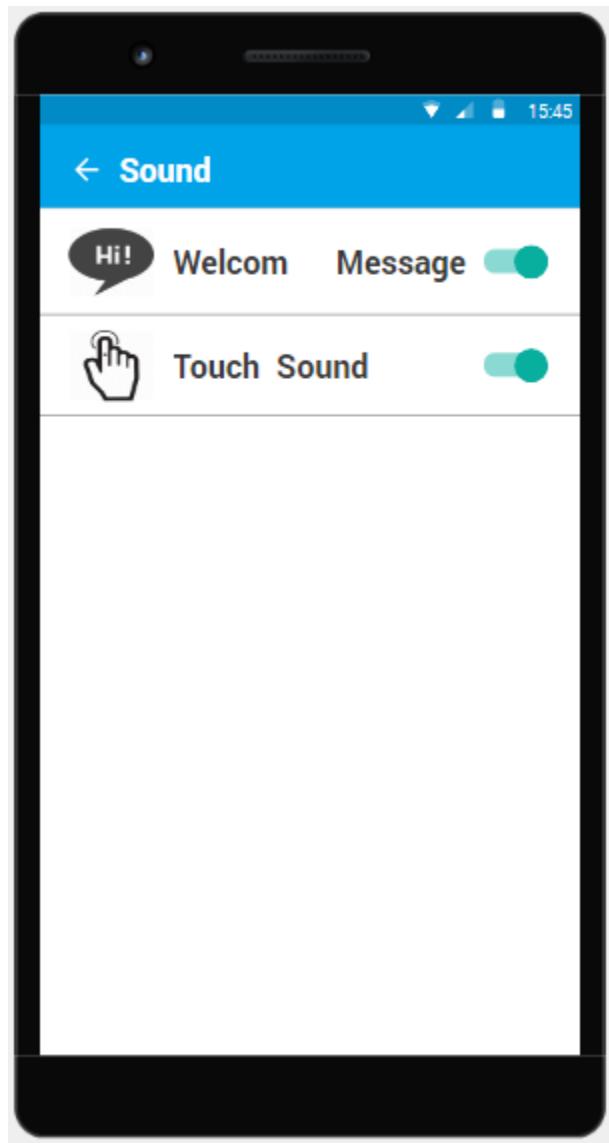
Settings > Display Screen



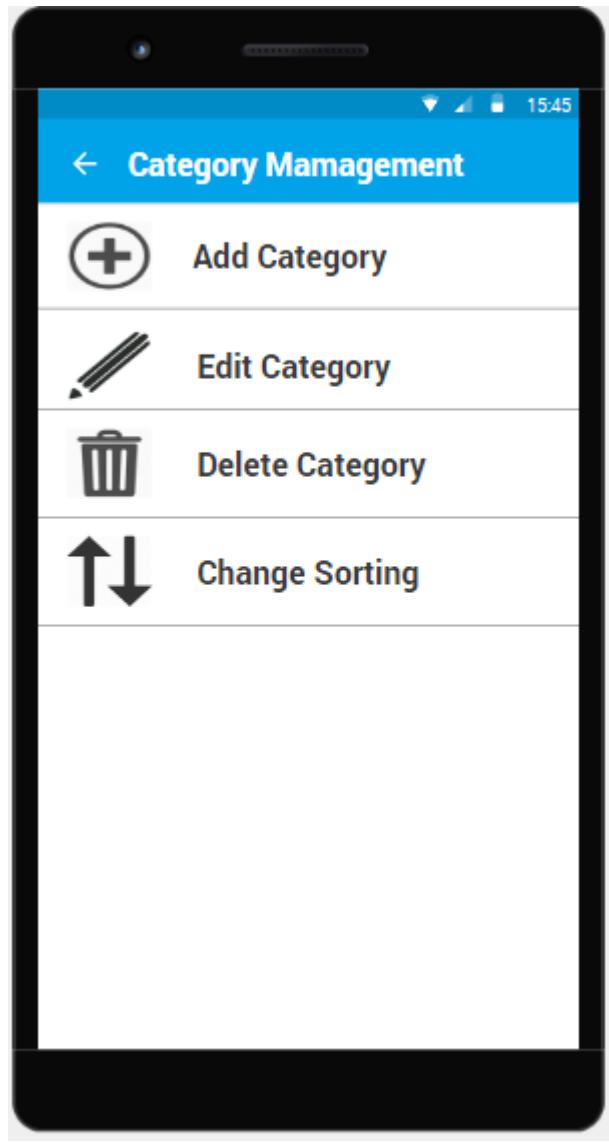
Settings > Profile Screen



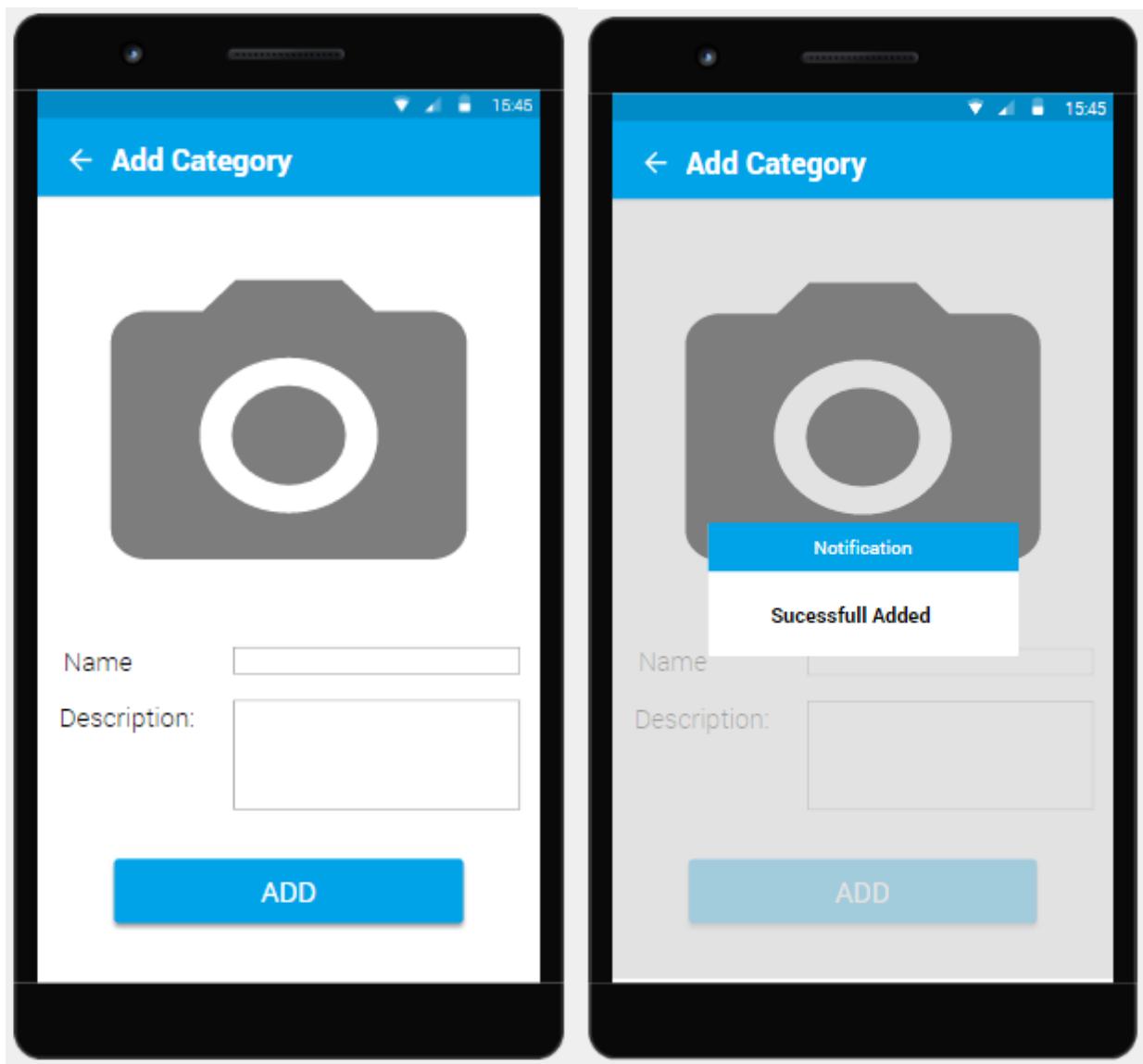
Settings > Sound Screen



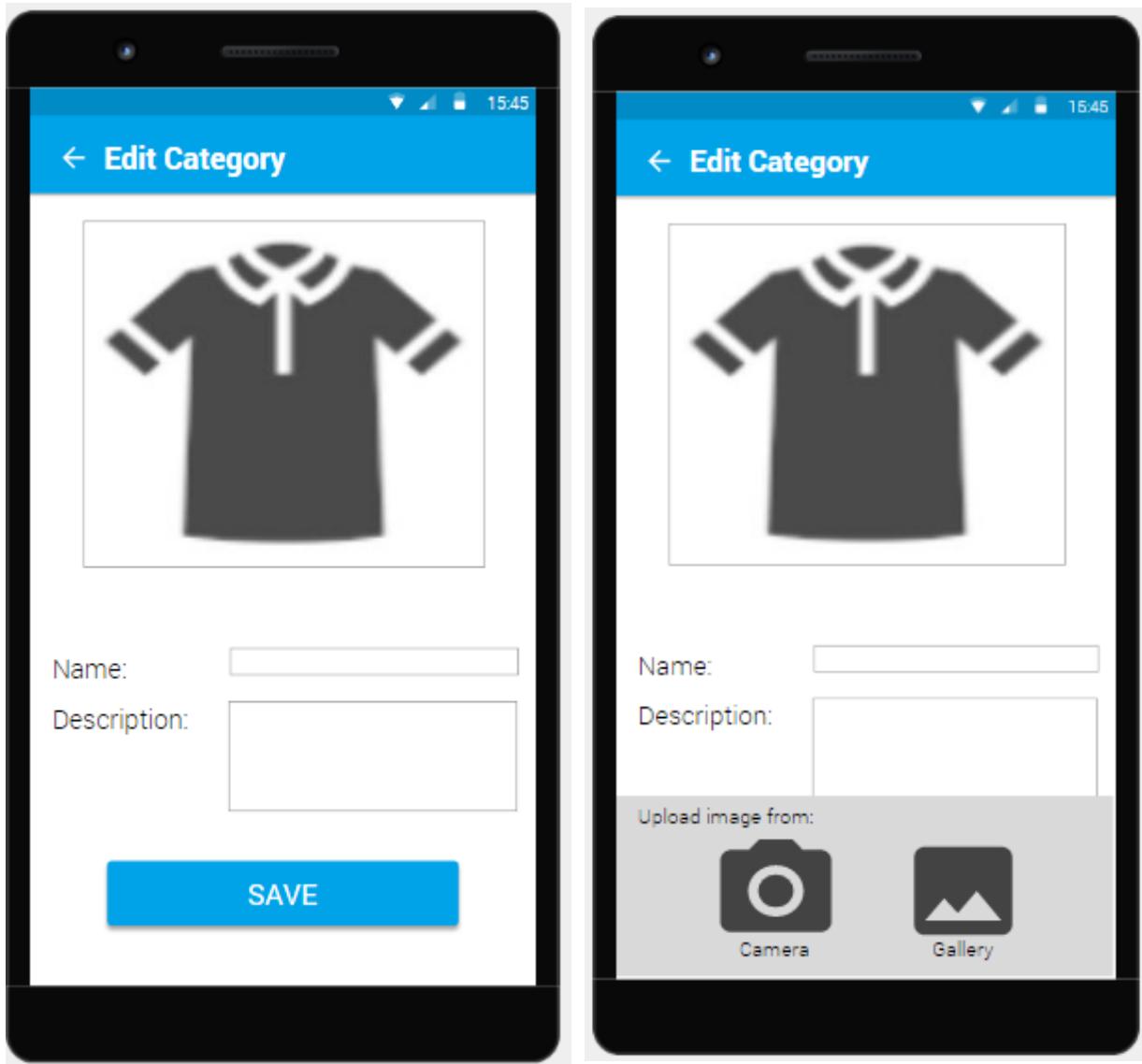
Setting > Category Management Screen

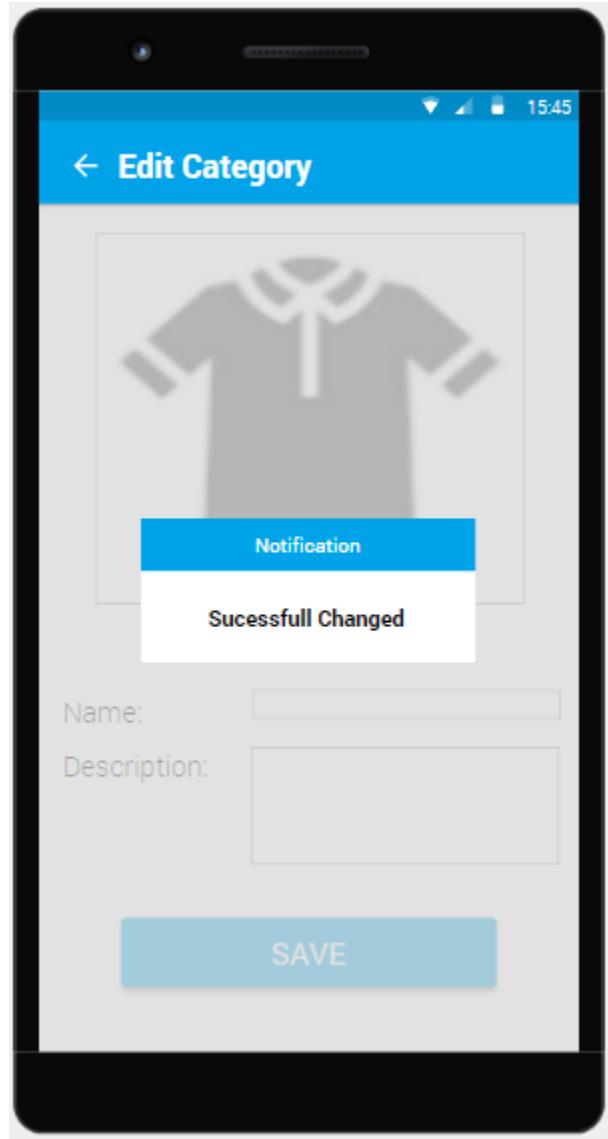


Setting > Category Management > Add Category Screen

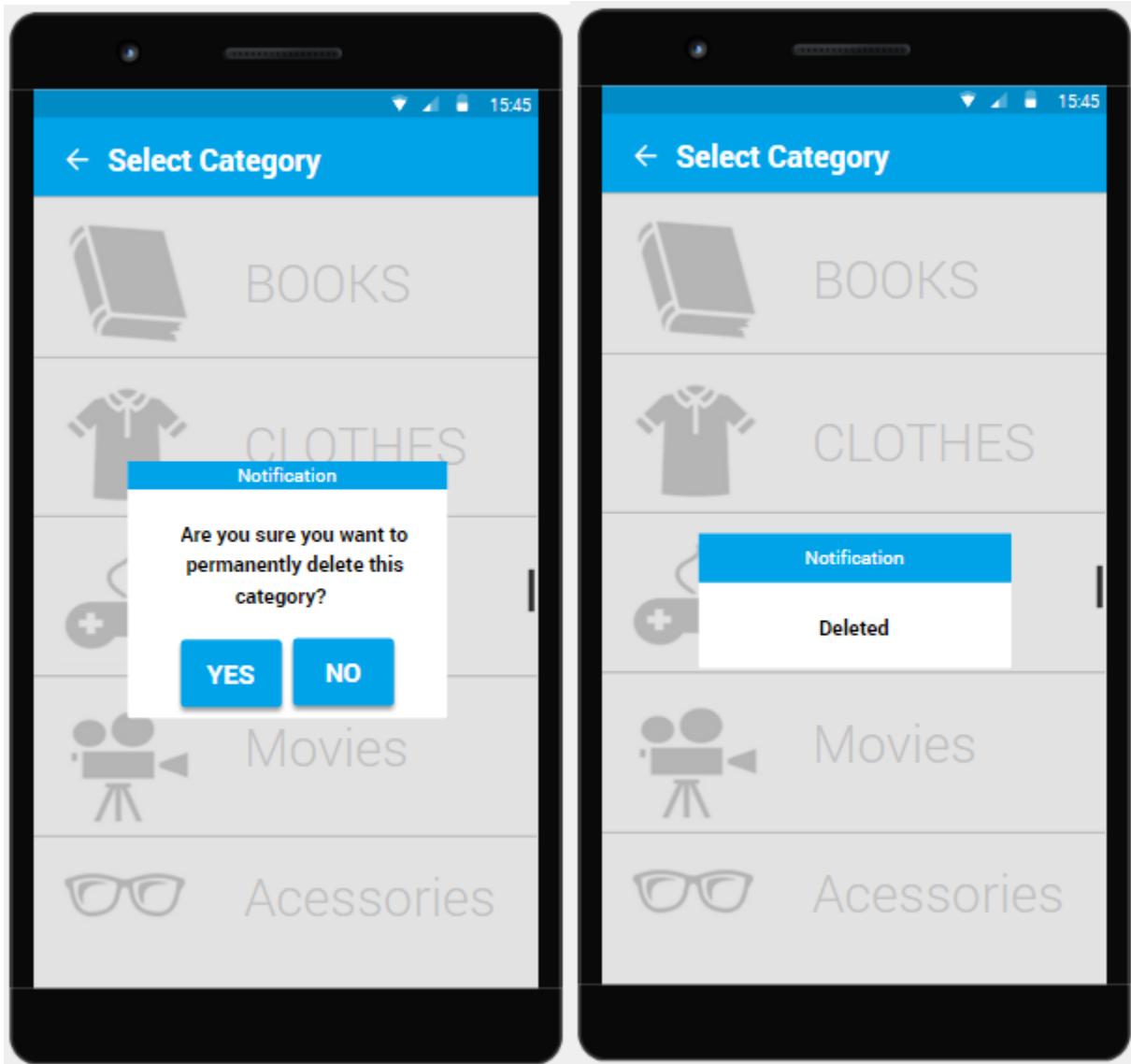


Setting > Category Management > Edit Category Screen

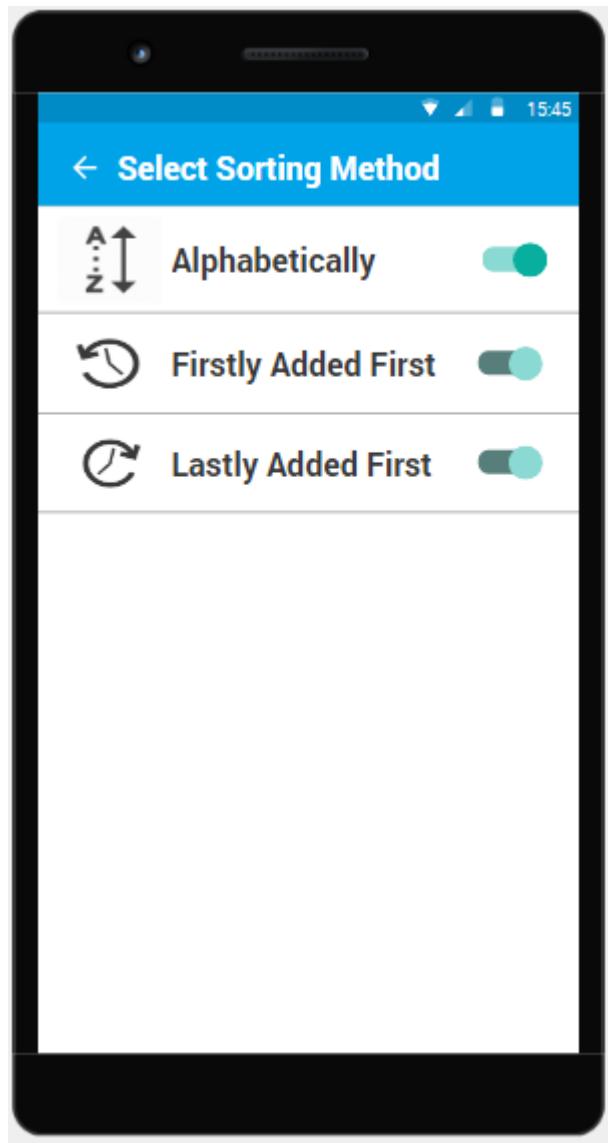




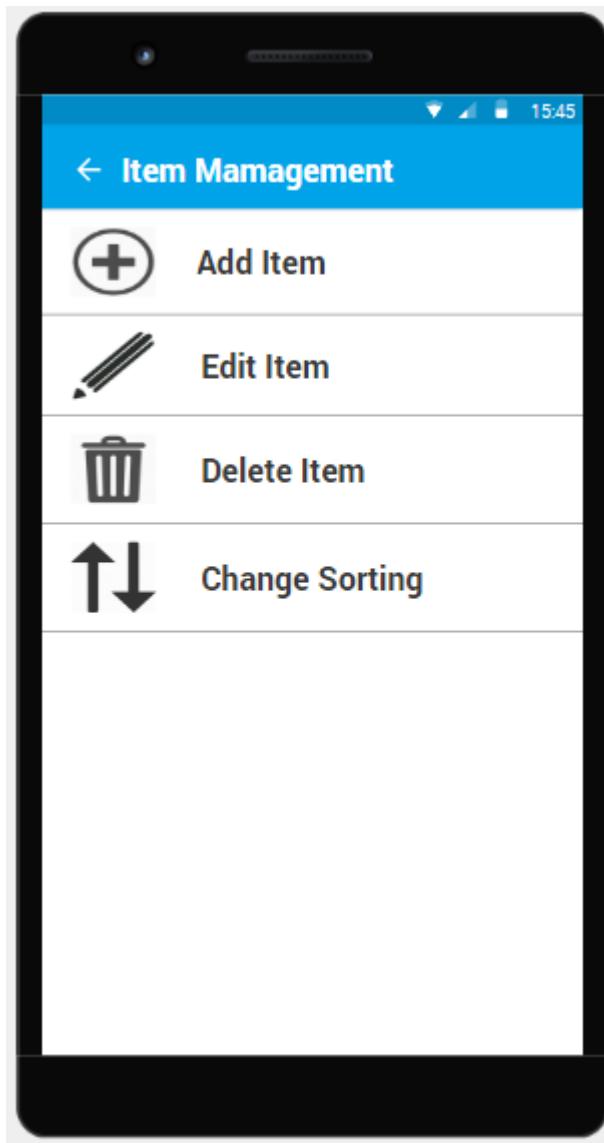
Setting > Category Management > Delete Category Screen



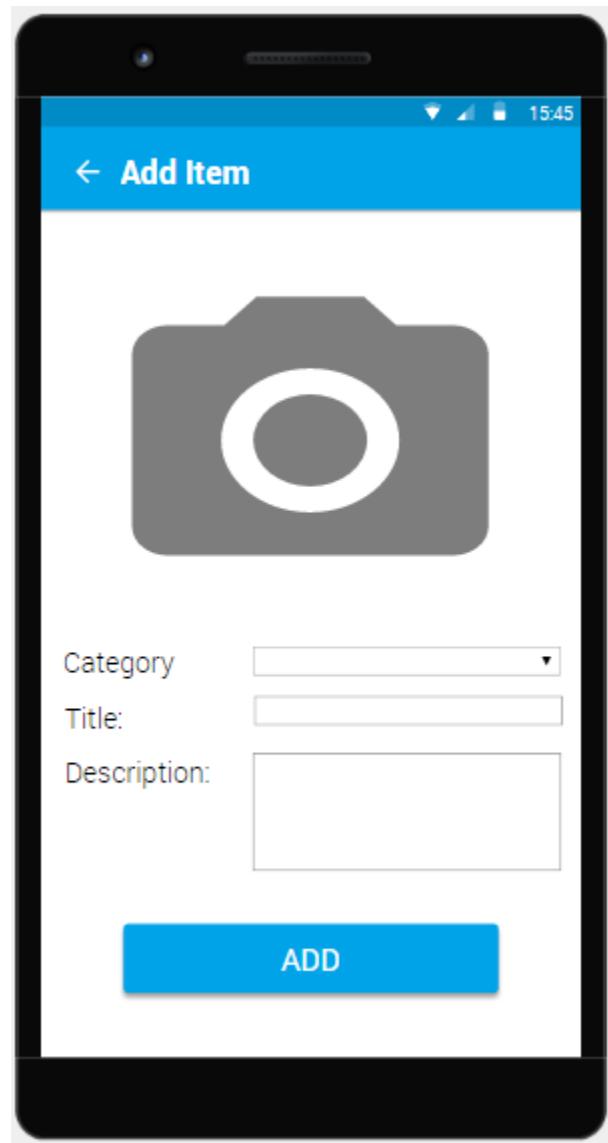
Setting > Category Management > Change Sorting Screen

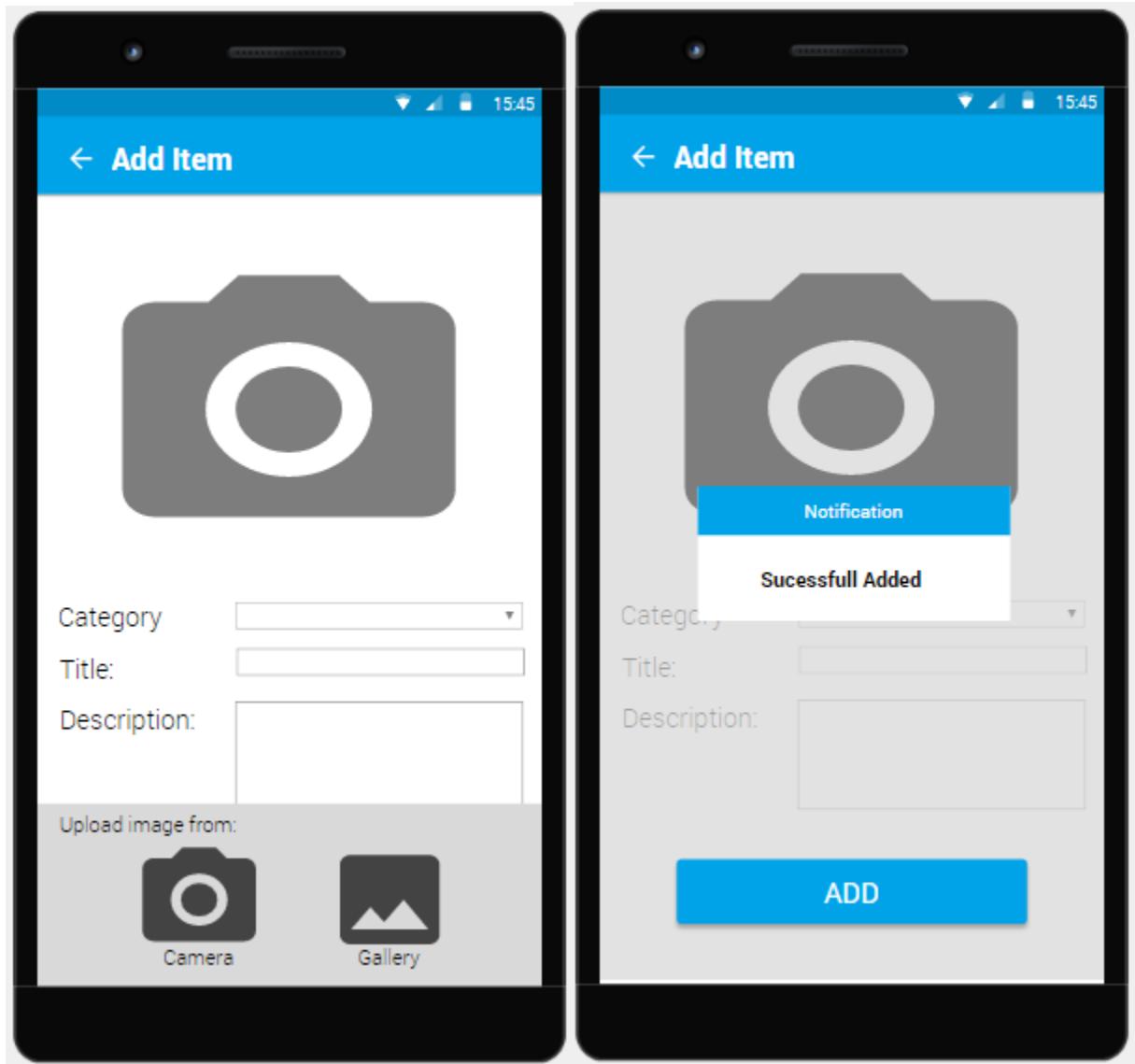


Setting >Item Management



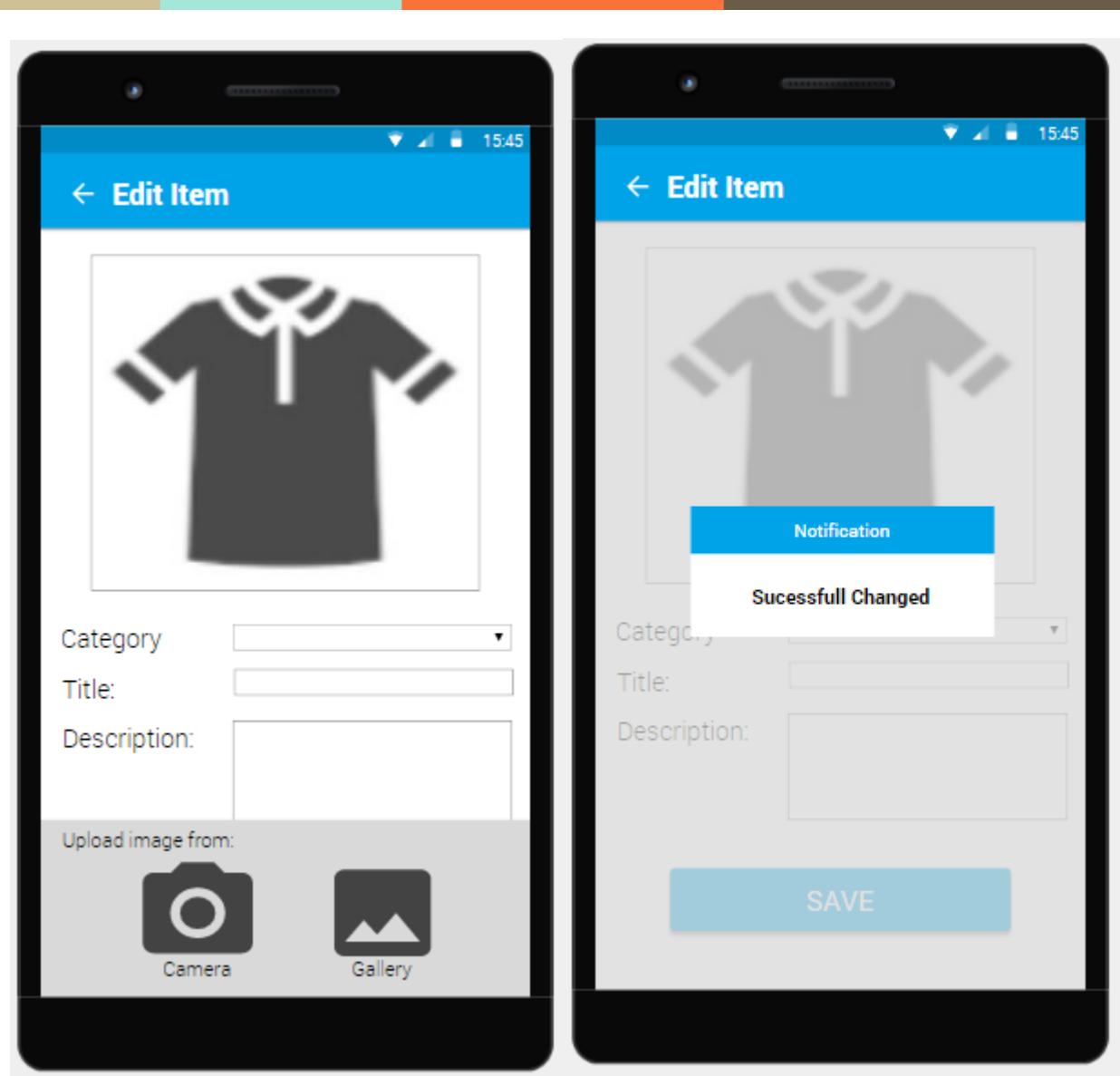
Setting >Item Management > Add Item Screen



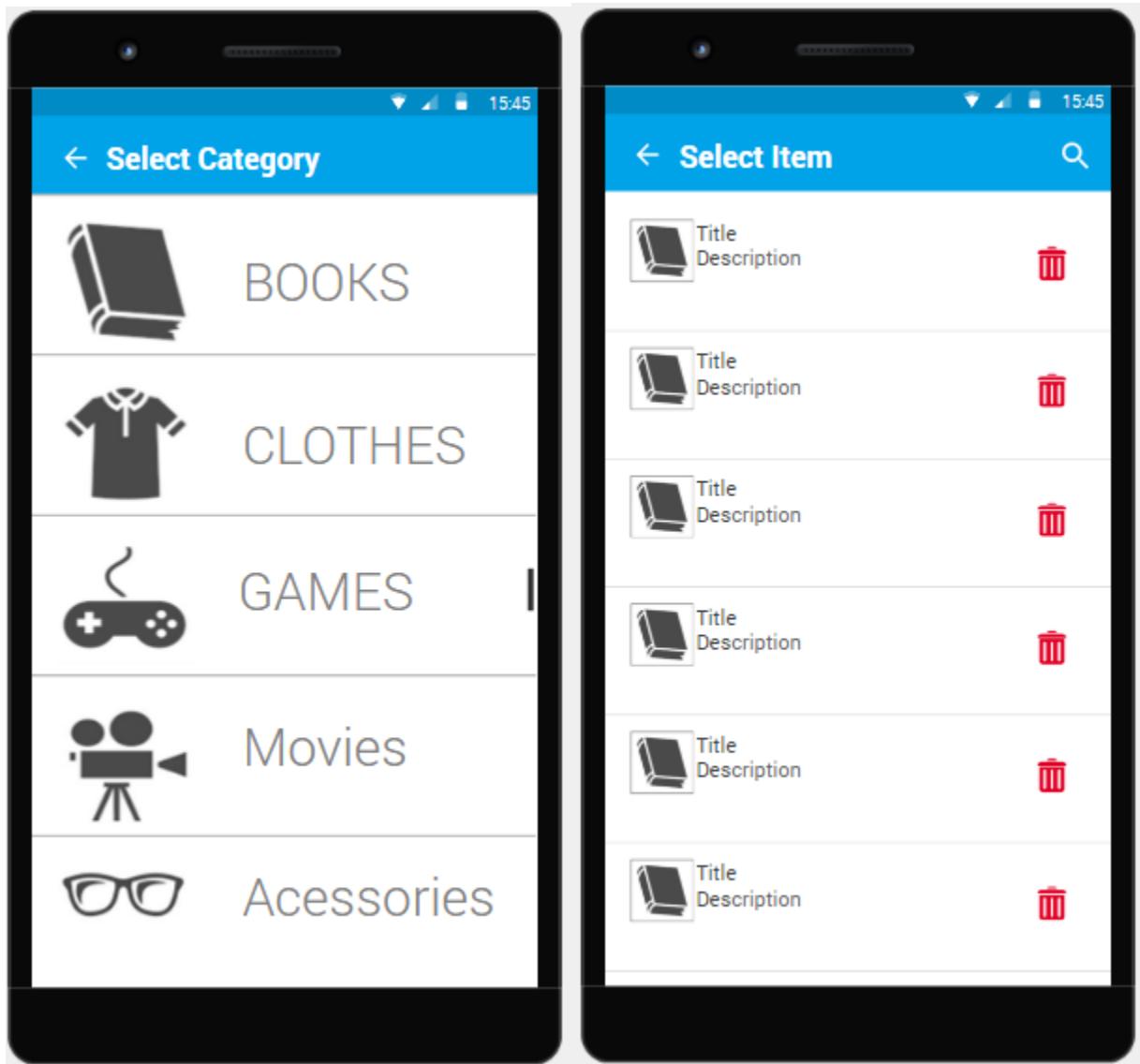


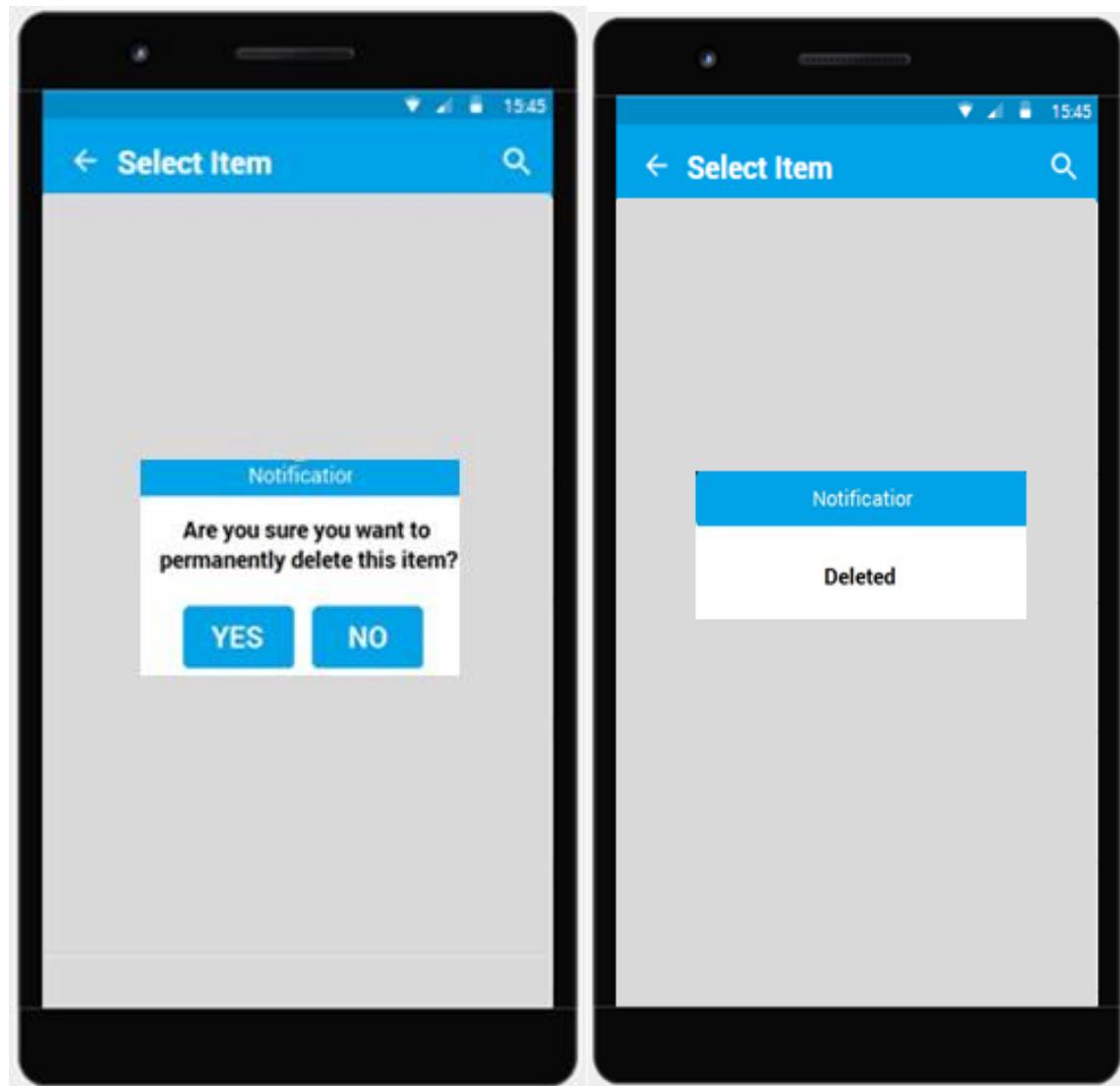
Setting >Item Management > Edit Item Screen



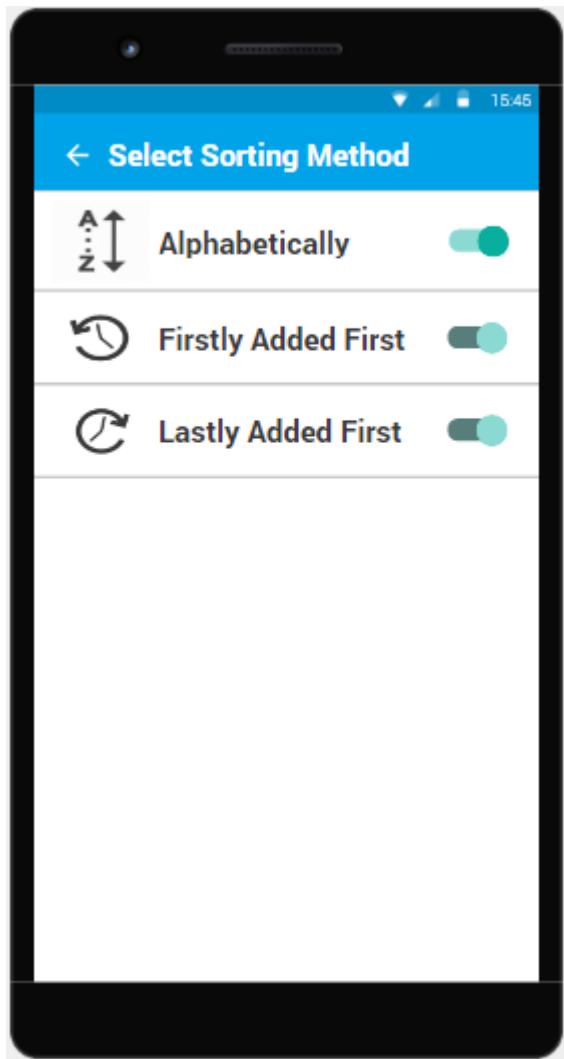


Setting >Item Management > Delete Item Screen

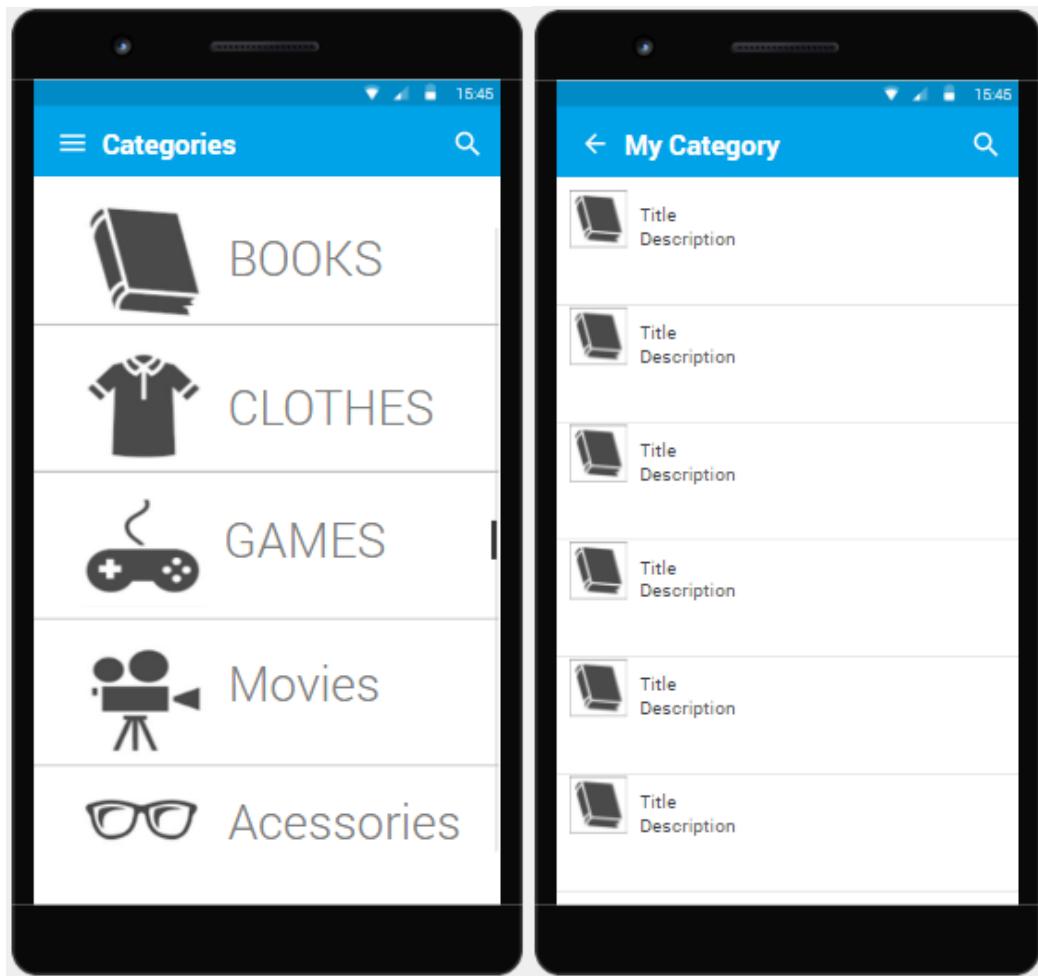


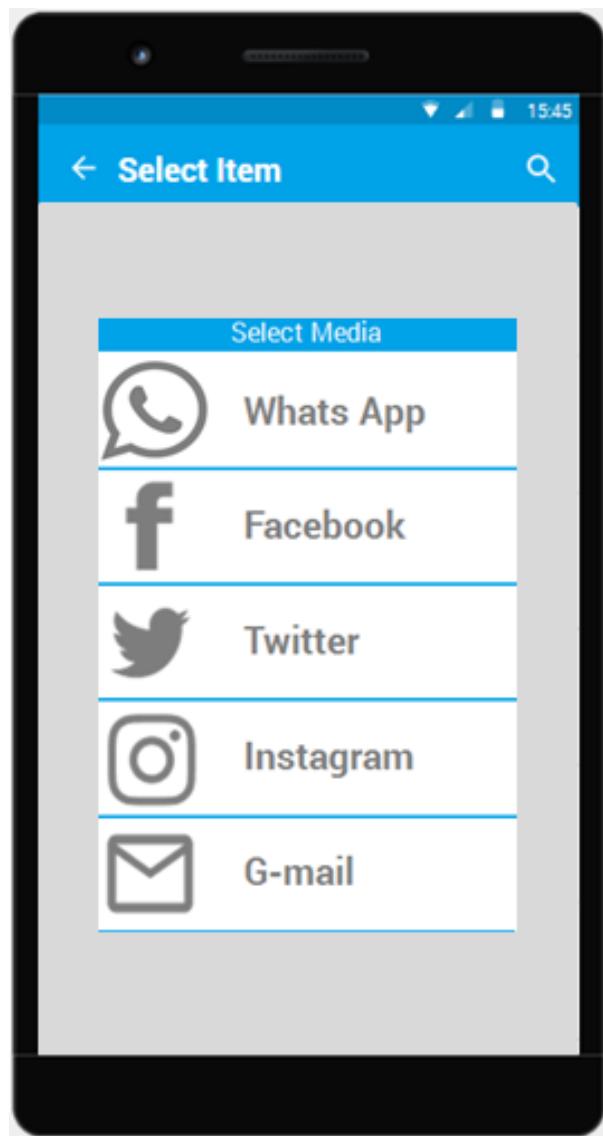


Setting >Item Management > Change Sorting Screen

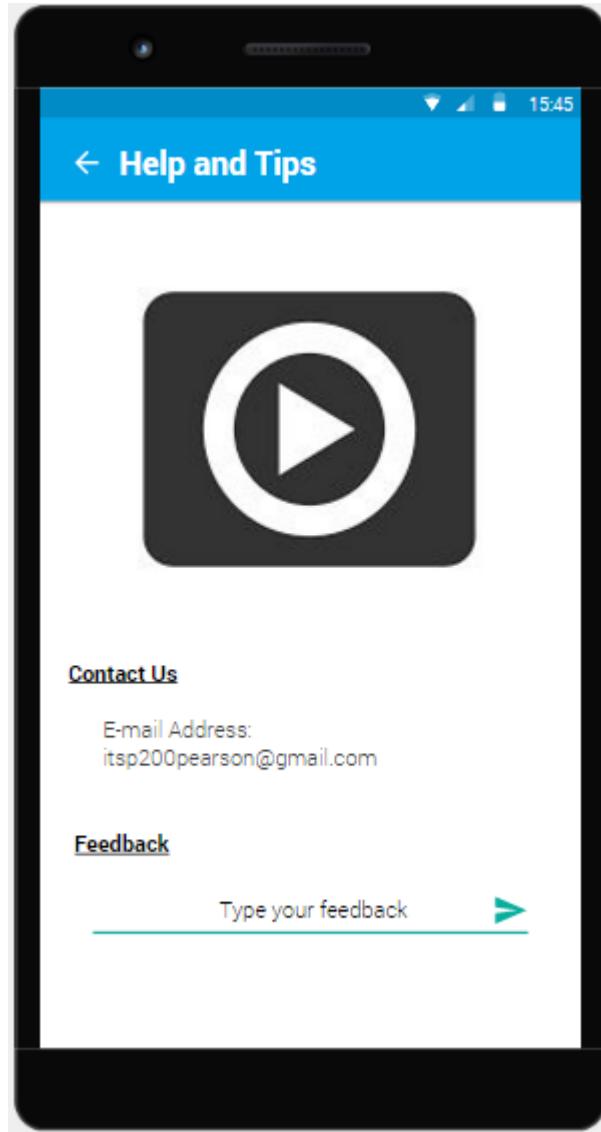


Share and Trade Screen





Help and Tips Screen



About Screen





2. Customer sign-off

Customer name and surname	Customer signature
Group leader name and surname	Group leader signature

Appendices

Appendix A

Nielsen's usability heuristics are a set of high-level design principles that are used to evaluate a user-interface to determine whether the interfaces conforms to well established and tested design principles (Preece et al. 2015). A revised version of Nielsen's heuristics are listed below:

- 1) The system should always provide the user with feedback;
- 2) The systems should match the user perception of the real world;
- 3) The system should provide the user with a clear exit to an unwanted state;
- 4) The systems should maintain the same standard and a high a level of consistency throughout;
- 5) The system should seek to eliminate error-prone conditions;
- 6) The system should minimise the load on the user's memory by making options and potential actions more visible;
- 7) The system must be efficient to use;
- 8) The system should make use of a minimalist aesthetic;
- 9) The system should aid users to recognise potential errors and assist the user in accounting for them; and
- 10) The system should provide the user with help documentation (Preece et al. 2015).

These heuristics are intended to be used by designers to compare or evaluate their designs and change their design accordingly. Each iteration of the design should seek to use these guidelines to solve usability problems (Stair & Reynolds 2015).

Bibliography

- Android Developers, Create and Manage Virtual Devices | Android Studio. Available at:
<https://developer.android.com/studio/run/managing-avds.html>
[Accessed August 3, 2017].
- Android Developers, Download Android Studio and SDK Tools | Android Studio. Available at:
https://developer.android.com/studio/index.html?gclid=CjwKEAjw9_jJBRCXycSarr3csWcSJABthk071PgsXQmXkW2jaenCzmxk5faXdS3nKbo99zzQizAWmhoC6N_w_wcB
[Accessed June 14, 2017].
- IEEE, 1990. *IEEE Standard Glossary of Software Engineering Terminology*, Inst of Engineering, Electrical & Electronic Standards.
- Oracle, Java SE Development Kit 8 - Downloads. Available at:
<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
[Accessed August 3, 2017].
- Oracle, Java SE Runtime Environment 8 - Downloads. Available at:
<http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>
[Accessed August 3, 2017].
- Dennis, A., Wixom, B. H. & Tegarden, D., 2005. *Systems Analysis and Design with UML Version 2.0*. 2nd ed. New Jersey: John Wiley & Sons, Inc.
- Satzinger, J. W., Jackson, R. B. & Burd, S. D., 2015. *Systems Analysis and Design in a Changing World*. 7th (E-BOOK) ed. Massachusetts: Cengage Learning.
- Shelly, G. B., Cashman, T. J. & Rosenblatt, H. J., 2003. *Systems Analysis and Design*. 5th ed. Boston: Thomson Course Technology.
- Bertolino, A., 2001. Chapter 5: Software Testing. *University of Houston-Clear Lake*. Available at: http://sce.uhcl.edu/helm/SWEBOK_IEEE/data/swebok_chapter_05.pdf
[Accessed August 7, 2017].
- Connolly, T.M. & Begg, C.E., 2005. *Database Systems: A Practical Approach to Design, Implementation, and Management*, Pearson Education.
- Finley, K., 2012. What Exactly Is GitHub Anyway? *TechCrunch*. Available at: <http://social.techcrunch.com/2012/07/14/what-exactly-is-github-anyway/>
[Accessed June 14, 2017].
- Justinmind, Prototyping tool for web and mobile apps - Justinmind. Available at: <https://www.justinmind.com>
[Accessed June 14, 2017].
- Mahfuj, M.M., 2012. Logical design vs physical design. Available at: <https://www.slideshare.net/MahediMahfujAnik/logical-design-vs-physical-design>
[Accessed May 17, 2017].
- Preece, J., Rogers, Y. & Sharp, H., 2015. *Interaction Design: Beyond Human-Computer*

Interaction, John Wiley & Sons.

softwaretestingclass, 2012. System Testing: What? Why? & How? *Software Testing Class*. Available at: <http://www.softwaretestingclass.com/system-testing-what-why-how/> [Accessed June 15, 2017].

Stair, R. & Reynolds, G., 2015. *Principles of Information Systems*, Cengage Learning.

Williams, L., 2006. Testing Overview and Black-Box Testing Techniques. *North Carolina State University*. Available at: <http://agile.csc.ncsu.edu/SEMaterials/BlackBox.pdf> [Accessed August 7, 2017].

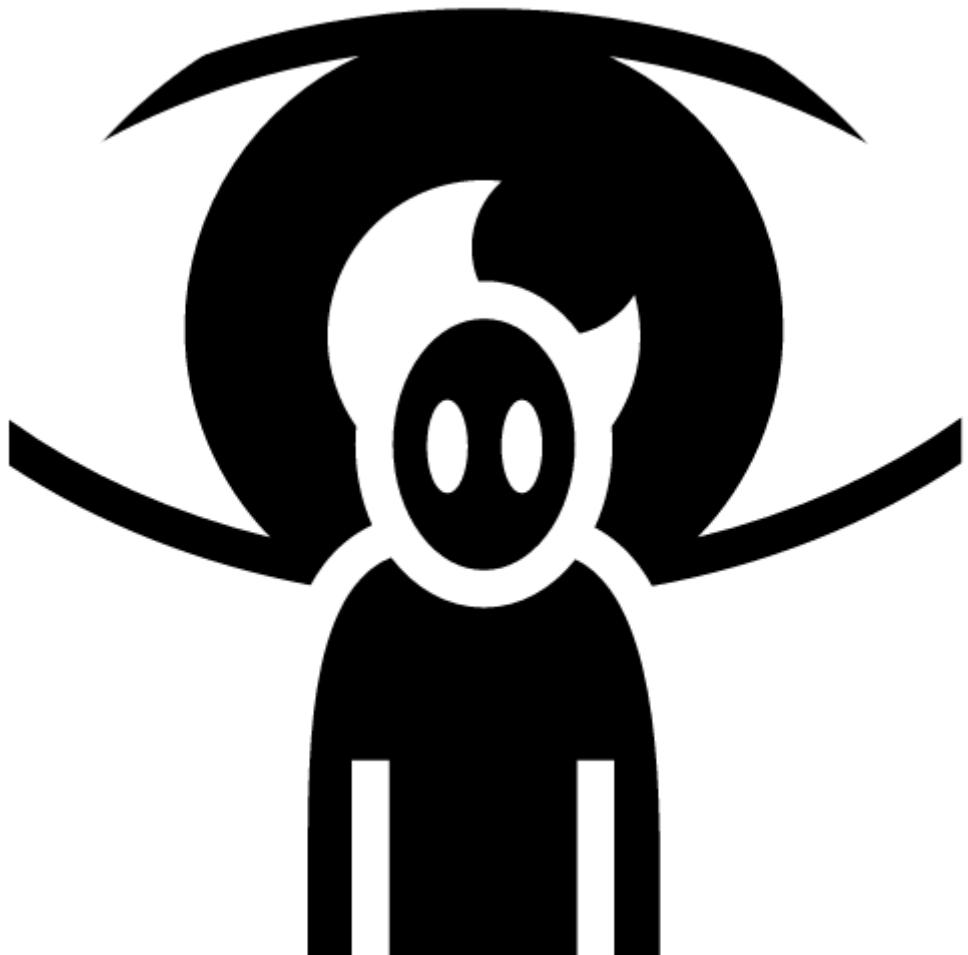
WebAIM, 2016. Microsoft Word: Creating Accessible Documents

<http://webaim.org/techniques/word/>
[Accessed August 9, 2017].



System Logical and Physical design

eyeRS Development Team - ITSP200 (Deliverable 3)



Introduction

This deliverable presents the system logical and physical design document. This document will define what physical and logical design entails and transpire. It also includes the logical data model (ERD), the logical process model (DFD), the testing templates and the test plan. Additionally, it includes the system interface designs along with the technologies that have been used in this project.

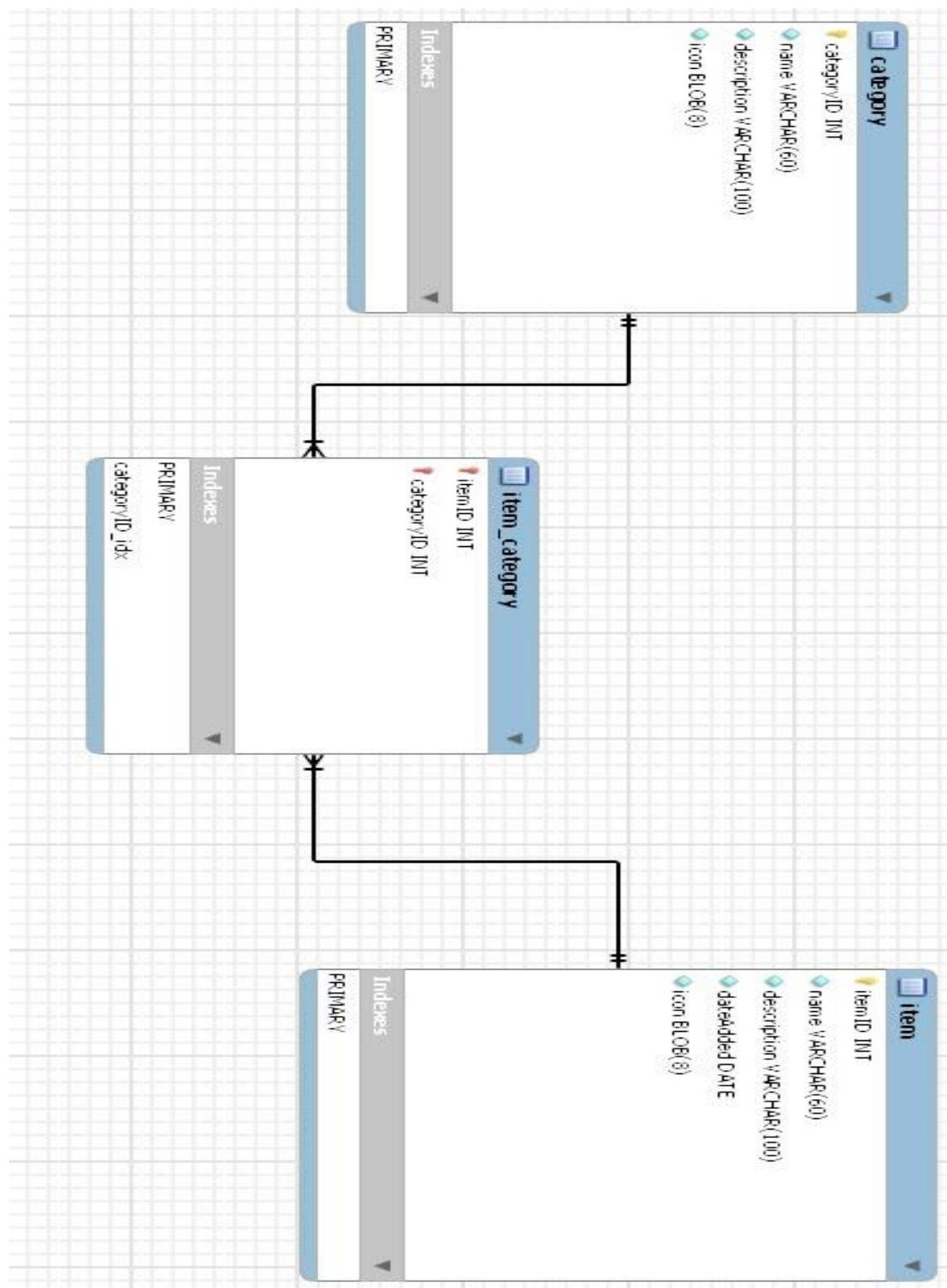
The purpose of the eyeRS app is to give individuals the power to manage and organise their personal and professional lives with less hassle. The eyeRS app achieves this control by allowing users to freely access and catalog all of their belongings and items that are uploaded to the app. Uploads can occur anywhere. Each belonging is saved by means of an image, title and a short description.

1. Information systems design

1.1. Logical design

Logical design of a system relates to the abstract presentation of the data flows within the system, it is mostly conducted via modelling by using an over-abstract model of the actual system (Satzinger, *et al.*, 2015). Logical design shows how the data that have been given to the system is processed (Satzinger, *et al.*, 2015). This design type is used to document information systems since the logical nature of the system is documented without specifying the detailed tasks of how, where and by whom the information that the system needs, are gathered. Logical designs can be represented by an Entity-Relationship diagram (Mahfuj, 2012).

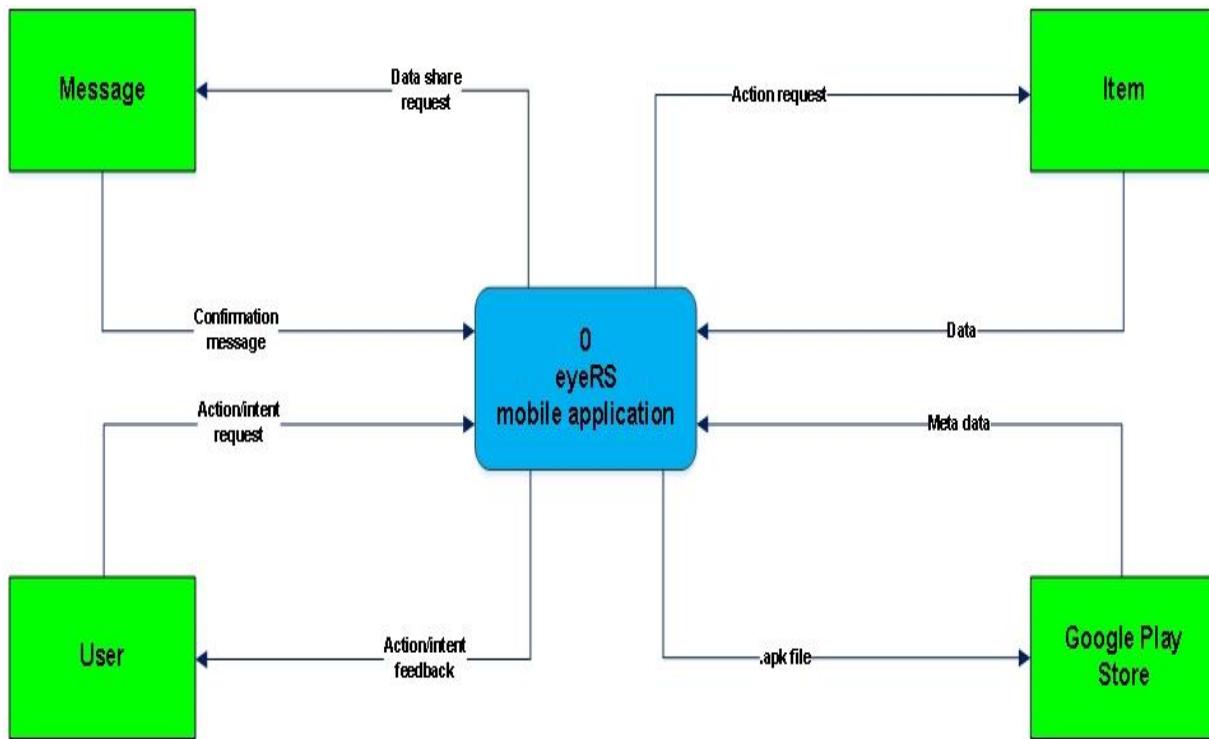
1.1.1. EyeRS Entity Relationship Diagram



1.1.2. Context & Logical Process Model

Context (High Level)

The diagram below indicates the context diagram (high-level DFD) for the eyeRS mobile application.



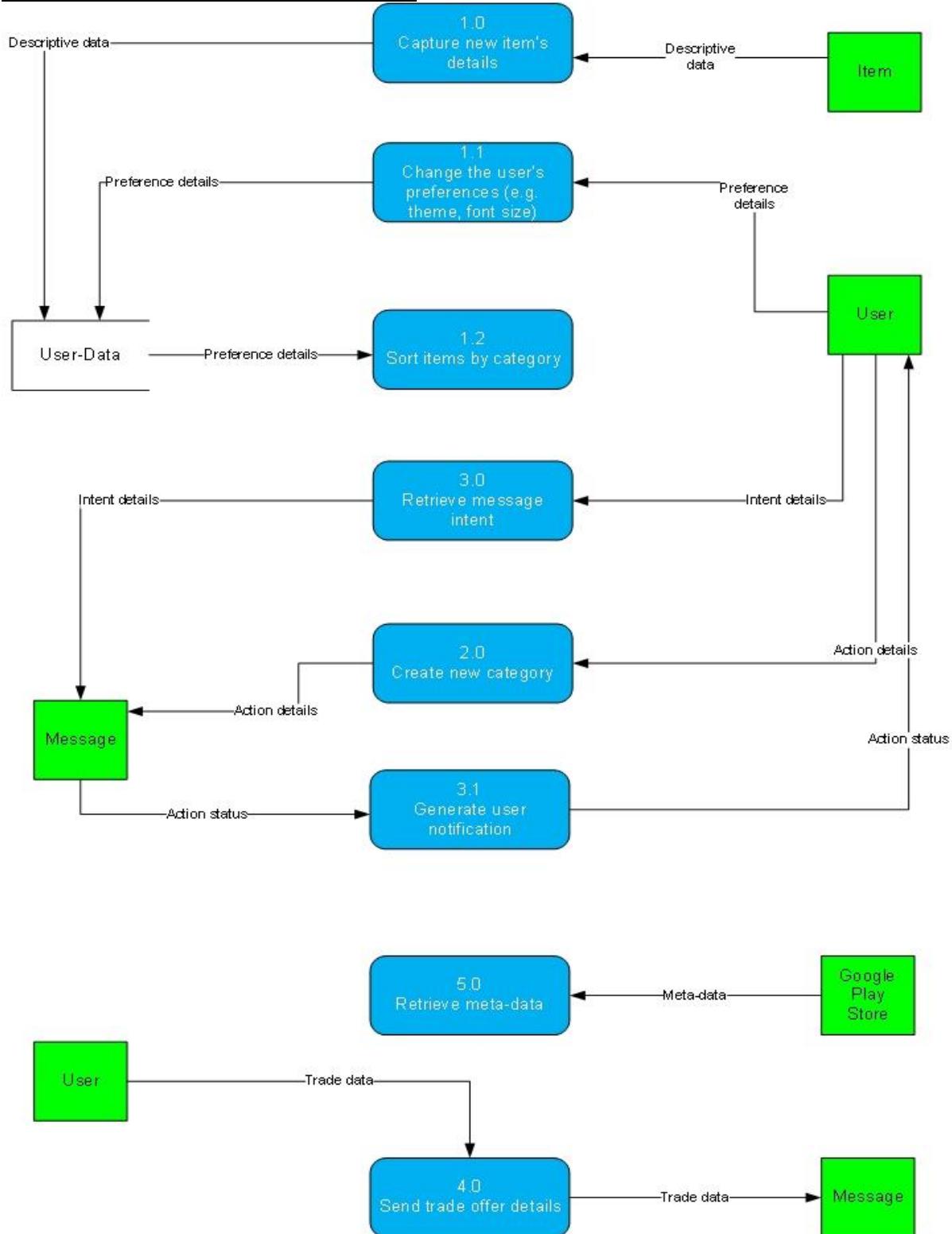
Processes

- 1.0 Capture the new item's details;
- 1.1 Get user preferences;
- 1.2 Sort items by category
- 2.0 Create new category;
- 3.0 Retrieve message intent;
- 3.1 Generate user notification;
- 4.0 Send trade offer details
- 5.0 Retrieve metadata

The table below is a summary of the data flows, processes and entities relating to the Data Flow Diagram.

Data Flow	From	To
Intent details	User entity	Retrieve message intent process
Descriptive data	Item entity	Capture new item's details process
Descriptive data	Retrieve item's details process	User-Data data store
Preference details	User entity	Implement the user's preferences process
Preference details	Implement the user's preferences process	User-Data data store
Intent details	Retrieve message intent	Message entity
Action status	Message entity	Generate user notification process
Action status	Generate user notification process	User entity
Meta-data	Google Play Store entity	Retrieve meta-data process
Action details	User entity	Create new category process
Preference details	User-Data data store	Sort items by category process
Trade data	User entity	Validate trade request process
Trade data	Validate trade request process	Message entity

Logical Process Model (Low level DFD)



1.2 Physical design

Physical design is the graphical presentation of the system (Dennis, et al., 2005). It shows the internal and external units. It also shows how the data is flowing within the system (Dennis, et al., 2005). The physical design relates to the input and output process of the system, how data

is authenticated, how it is processed and how it is displayed as output (Dennis, et al., 2005). Physical design can be broken down into three sub-tasks. These subtasks of the physical design are:

4. User Interface design;
5. Data design; and
6. Process design.

User interface design focus on how the user gives information to the system and how this data will be presented to the user (Shelly, et al., 2003). Data design is how the data is stored and represented within the system (Shelly, et al., 2003). Process design focuses on how data will move within the system, how and where the data is validated, secured or transformed as it is flowing within the system as well as out of the system (Mahfuj, 2012).

1.2.1 Investigation of technologies to be applied

The operating system that we will be using throughout project development, on the different stages of development from planning until implementation/maintenance will be Windows 7,8 or 10 operating systems.

There are other operating systems on the market, such as: Linux and Mac OS however, Windows will be the preferred operating system of use. On the Windows operating system we will run different technologies to enable successful development of the eyeRS app.

The following technologies are used throughout the development of the eyeRS project:

11. Justinmind;
12. Android studio;
13. Microsoft word;
14. Github;
15. Google Drive;
16. Java Runtime Environment (JRE);
17. Java Development Kit (JDK);
18. Android Software Development Kit Manager;
19. Android virtual Device Manager; and
20. SQLite.

The above mentioned technologies are briefly discussed below:

Justinmind is a system that allows users to create system and application prototypes, supporting many devices.(Justinmind, 2014) It uses a drag and drop feature and assign events to the objects (Justinmind, 2014). It will assist us in creating user interfaces and allow us to gain feedback from the client of specific design preferences.

Android studio is a development environment for the Android platform that we will use to develop, debug and deploy the eyeRS app (Android Developers, 2017). It provides the best tools for building high quality apps for any android device (Android Developers, 2017).

Microsoft word is a graphical word processing system that users can create documents with (WebAIM, 2016). We are using it for the project documentation . The documentation of the project is important for many reasons such as:

- 
5. A reference for other projects;
 6. For system upgrade or enhancement;
 7. Project troubleshooting and;
 8. Providing an understanding of the system concept behind its functionality (Shelly, et al., 2003).

Github is a repository hosting service (Finley, 2012). It will allow the team to collaborate on development tasks via its platform, in a secure and productive way (Finley, 2012). It allows project team members to work on the project tasks from anywhere regardless of location (Github, 2017).

Google Drive will provide access to files anywhere in a secure way, working as a cloud storage to backup images, videos and documents (Google, 2017).

It also enable the project team to utilize Google Docs, which is an online, word processing program to compile documents allow the team to work on the same document simultaneously (Google, 2017).

Java Runtime Environment is a software package that consists of the necessary requirements to execute a Java program (Oracle, 2017). The eyeRS team will make use of this technology (which comes as part of the Android Studio SDK) to create the app's functionality (Android Developers, 2017).

Java Development Kit is will be used to develop java programs as it also comprises of the Java Runtime Environment in the kit (Oracle, 2017).

Android Software Development Kit Manager is a set of development tools which are embedded in the Android Studio development environment which will be used to debug the eyeRS app (Android Developers, 2017).

Android virtual Device Manager is a program that will enable the development team to debug their apps on a virtual machine with specifics requirement such as: hardware, storage space, screen resolutions and allowing the development team to target various Application Programming Interface (API) levels (Android Developers, 2017).

SQLite is a database management system based on the structured query language.(Oracle, 2017). It is a technology used by the Android system to create databases to be used for storage by the application (Android Developers, 2017).

1.2.2 System testing

Testing types

Software testing is the ongoing process of ensuring that a program not only is function but also fulfils all set criteria for the program (IEEE, 1990). According to IEEE (1990) and Williams (2006) there are various primary types of programming tests that are should be performed on a program before it can be labeled as acceptable, these tests include:

5. Walkthroughs;
6. Trace table
7. Black box/data coverage; and
8. White box/Code coverage.

Bertolino (2001) suggested that software testing be conducted throughout the development process.

Walkthroughs are a fairly common and cost effective method of software testing. Walkthroughs are done by manually tracking a program's data and output on pen and paper. Walkthroughs are frequently used to facilitate discussion to what extent a program has met its design requirements. Trace tables are similar to walkthrough however they are focused on testing the logic of an algorithm rather than the program as a whole (ref). According to ref trace tables are good for highlighting small non obvious logic errors with an a program.

Black-box testing, also known data coverage, is when a subprogram verifies a set of known valid inputs are tested against possible input values (Williams, 2006). According Williams (2006) to black-box testing is used to verify that a program can process possible values and meets the proposed requirements.

White-box testing, also known as code coverage, is when the execution of statements, or groups of statements, are tested to ensure that they execute as expected (Williams, 2006). Williams (2006) stated White-box testing is used to ensure that all statements execute as expected.

Softwaretestingclass (2012) states that system testing tests the behaviour of the system based on risks, specifications and requirements which include functional and nonfunctional requirements that were established by the developers. A fully integrated system should be tested. The final testing done is the Systems Testing to verify that all specifications and requirements have been met.

Testing templates

Test Case Number: 1 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Open the application.	Test Case Name: Open the app. Subsystem: Stating. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the eyeRS icon	A Welcome message appear with the login screen.			

Test Case Number: 2 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Test the registration for eyeRS.	Test Case Name: Register new user. Subsystem: Register. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click “Register” button	The system displays a screen with registration details			
2	Enter a username	The system displays the username in the text area			
3	Enter an email address	The system displays the email address in the text area			
4	Enter a pin “*****”	The system asks the user to verify the pin			
5	Re-enter the pin “*****”	The system verifies that the pins match			
6	Select a security question	The System opens a drop-down list			
7	Enter security response (Answer to security question).	The system accepts the input			
8	Click the “Register” button	The system registers the user with the database, a Registered notification appears and returns to the “Login” screen			

Test Case Number: 3 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Test the clear button on the register screen.	Test Case Name: Clear register new user information. Subsystem: Register. Design Date: Execution Date:
--	---

1	Click the "Register" button	The system displays a screen with registration details			
2	Enter the registration information	Information display on the fields.			
3	Click the clear button	All the information in the fields will be erased. Same screen are displayed.			
4	Click the back button.	Login screen is displayed.			



Test Case Number: 4 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user the opportunity to retrieve a forgotten pin.	Test Case Name: Forgot Pin. Subsystem: Login. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the red “Forgot Pin” text	New screen appears that display security question and text field for the security question answer.			
2	Enter username	Information displays on the textfield.			
3	Create an new pin in the pin text field	Pin would appear as stars			
4	Verify the pin by retyping the pin in the text field	Pin would appear as stars			
5	Select the drop down box	A list of possible security questions will appear			
6	Enter security response	Text field is updated.			
8	Click the “Reset” button	New pin as well as new security question and security answer is created and saved in the database. Rese successful notification appear. User is returned to the Login screen			



Test Case Number: 5 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Test the clear button on the forgot pin screen.	Test Case Name: Clear reset pin information. Subsystem: Register. Design Date: Execution Date:
--	---

1	Click the “Forgot Pin” text	The system displays a screen with the password reset details			
2	Enter the password reset information	Information display on the fields.			
3	Click the clear button	All the information in the fields will be erased. Same screen are displayed.			
4	Click the back button.	Login screen is displayed.			

Test Case Number: 6 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Log in to use the eyeRS application.	Test Case Name: Login. Subsystem: Main Screen. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the “Please enter your PIN” textfield	Keyboard appears.			
2	Enter pin	Pin appear as stars			
3	Click the “Login” button	If correct the system will open the main menu screen. If not the system will notify the user of the incorrect password, and the pin can be re-entered.			



Test Case Number: 7 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Displays side menu.	Test Case Name: View Side Menu. Subsystem: My Category. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the “More” icon in the main menu	A side menu with more options will appear.			

Test Case Number: 8 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Adds an item to the users database via the camera upload.	Test Case Name: Add Item. Subsystem: My Category. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on “Add new item” in the side menu	An Add Item screen will appear.			
2	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			
3	Click on Cancel	No photo were added.			
4	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			
5	Click on Take Photo	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Select the category drop box	A list of the categories fill appears.			
7	Select the appropriate category from the drop box list	The selected category will appear. (Note: Not yet saved).			



8	Click in the Name of item text field	Keyboard appears.			
9	Enter the title of the item in the item text field	Title will appear in the text field. (Note: Not yet saved).			
10	Click in the Item description text field	Keyboard appears.			
11	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
12	Click the “Add” button.	Item is added in the database. Notification of saved appears. User remains in the add item screen			
13	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture— picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
14	Enter the new item detail as before.	Item detail displays.			
15	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add item screen.			
16	Click on the back button.	Main menu appears.			

Test Case Number: 9 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Adds a category to the user's database via the camera upload.	Test Case Name: Add Category. Subsystem: Main menu. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.			
2	Click on "Add new category" in the side menu	An Add Category screen will appear.			
3	Click on the "Camera" icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Category Name text field	Keyboard appears.			
7	Enter the Name of the category in the category text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the category in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the "Add" button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.			
11	Click on the "Camera" icon	Notification appears asking for an upload of the image via the camera or the gallery.			



12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new category detail are before.	Category detail displays.			
14	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.			
15	Click on the back button.	Main menu appears.			

Test Case Number: 10 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: View images from the user catalogue.	Test Case Name: Slideshow. Subsystem: Main menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.			
2	Click on “Slideshow” in the side menu	An slideshow screen will appear and the images in the catalogue will appear in a slide show			
3	Click on the back button.	Main menu appears.			



Test Case Number: 11 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Shares a standard or custom user category and all content with other user's	Test Case Name: Share from side menu. Subsystem: Main menu. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the more icon on the main menu.	The side menu will appear.			
2	Click on "Share" in the side menu	A select category screen will appear.			
3	Click on appropriate category.	A Select Item screen will appear.			
4	Select an item to share.	A pop up will appear that shows all the supported sharing methods.			
5	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
6	Click the back button to return from the sharing app.	User is returned to the select item screen.			

***Note: Repeat for each sharing method.**

Test Case Number: 12 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Searches for items by (name, category, details).	Test Case Name: Search Item to share via the side menu. Subsystem: Share. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon in the select item screen	Search text field will slide from the side and keyboard will appear.			
2	Search any item	Keyboard will disappear and search results will appear.			
3	Select the searched item	A pop up will appear that shows all the supported sharing methods.			
4	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
5	Click the back button to return from the sharing app.	User is returned to the select item screen.			

*Note: Repeat for each sharing method.

Test Case Number: 13 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Exits share screen.	Test Case Name: Exit Share. Subsystem: Share. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button.	The select category screen will appear.			
2	Click the back button.	Main menu will appear.			



Test Case Number: 14 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Trade category and all content with other user's.	Test Case Name: Trade via the side menu. Subsystem: Main menu. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the more icon on the main menu	Side menu will appear.			
2	Click on "Trade" in the side menu	A select category screen will appear.			
3	Click on appropriate category.	A Select Item screen will appear.			
4	Select an item to share.	A pop up will appear that shows all the supported sharing methods.			
5	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
6	Click the back button to return from the sharing app.	User is returned to the select item screen.			



Test Case Number: 15 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Searches for items by (name, category, details).	Test Case Name: Search Item to Trade via the side menu. Subsystem: Trade. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon in the select item screen	Search text field will slide from the side and keyboard will appear.			
2	Search any item	Keyboard will disappear and search results will appear.			
3	Select the searched item	A pop up will appear that shows all the supported sharing methods.			
4	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
5	Click the back button to return from the sharing app.	User is returned to the select item screen.			

Test Case Number: 16 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Exits Trade screen.	Test Case Name: Exit Trade via side menu. Subsystem: Trade. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button	The select category screen will appear.			
2	Click the back button	Main menu will appear.			



Test Case Number: 17 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Accessing help & tips for the app via the side menu.	Test Case Name: Help and Tips via Side menu. Subsystem: Side menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on “Help and Tips” in the side menu	A Help and Tips screen will appear.			
2	Click on the video.	A tutorial video will play.			
3	Click on the Send feedback button	User is taken to google play store to leave feedback at google play store.			
4	Click the back button to leave play store.	The help and tip screen appear.			
5	Click the back button in the help and tips screen.	User is taken back to the main menu.			

Test Case Number: 18 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Gives a brief history on the development team for eyeRS and the app via the side menu.	Test Case Name: About via the side menu. Subsystem: Side menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on “About” in the side menu.	A About screen will appear. Which contains the information about the app.			
2	Click the back button.	The settings screen appears.			



Test Case Number: 19 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Displaying of all possible sections for configuration of the app.	Test Case Name: Settings. Subsystem: Main menu. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.			
2	Click on "Settings" in the side menu	An settings screen will appear			

* Note Settings can be accessed by clicking on the more button on the right side of the screen.
Repeat setting test for the settings accessed via the main menu.



Test Case Number: 20 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Security settings that are provided can now be configured.	Test Case Name: Security Settings. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Security	A security settings screen will appear.			
2	Click on the Enter your username text field	Keyboard appears			
3	Enter username	The username will appear in the text fiel. Note it is not yet saved.			
4	Click on the Enter your new PIN here text field	Keyboard appears			
5	Enter the new PIN	Pin display as dots in the text field. Not it is not yet saved.			
6	Select drop down box	A list of all possible security questions will appear.			
7	Select a security question from the list.	New security question will appear in the security question box.			
8	Click on the Security Response text field.	A keyboard appears.			
9	Enter the security response in the text field	The security question will appear in the text field. Note: Data is not yet saved.			
10	Click on the Clear button.	All the entered information are erased. PIN reset screen remain on the screen.			
11	Enter all the reset PIN information.	All text will be displayed.			
12	Click the reset button	Successful notification will appear that. User will			



		be taken back to the settings screen.			
--	--	---------------------------------------	--	--	--

Test Case Number: 21 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Displays settings that can be configured.	Test Case Name: Display Settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on display	A display setting screen will appear.			
2	Click on the new colour block *To be repeated for each colour available	The colour of the app will change.			
3	Click on the select Font drop down box.	A list of all the available font will appear.			
4	Select a font in the list	The font of the app will change.			
5	Click on the select font size drop down box.	A list of all possible font sizes will appear.			
6	Click on the back button.	The Setting screen will appear.			



Test Case Number: 22 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: User profile can be configured. Upload a profile picture via the camera.	Test Case Name: Profile Settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Profile	A profile setting screen will appear.			
2	Click on the profile picture.	Notification appear asking to choose upload media source selection.			
3	Click on cancel	The request is canceled.			
4	Click on the profile picture.	Notification appear asking to choose upload media source selection.			
5	Click on camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click on the Please enter your display username text field.	A keyboard appears.			
7	Enter username in the username text field	New user name will appear Note: Data is not yet saved.			
8	Click on the save changes button	Successful notification appears. Profile setting screen still appears.			

Test Case Number: 23 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: User profile can be configured. Upload a profile picture via file upload.	Test Case Name: Profile Settings add profile picture via a file upload. Subsystem: Settings. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the profile picture.	Notification appear asking to choose upload media source selection.			
2	Click on Choose from Library.	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
3	Click on the Please enter your display username text field.	A keyboard appears.			
4	Enter username in the username text field	New user name will appear Note: Data is not yet saved.			
5	Click on the save changes button	Successful notification appears. User is still at profile settings.			
6	Click the back button.	User is returned to the settings option screen.			



Test Case Number: 24 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: The sound for the app can be configured.	Test Case Name: Sound Settings. Subsystem: Settings. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on sound	A sound setting screen appears			
2	Click on the welcome message toggle	The toggle turns off and welcome message will not play next time the user access the app.			
3	Click on the touch sound toggle	The toggle turns of and the touch sound will not play.			
4	Click on the back button	The Setting Option screen will appear.			

Test Case Number: 25 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: The management of the categories in the app.	Test Case Name: Category Management. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Category Management settings	A category management screen will appear			

Test Case Number: 26 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Adding a new category while in settings.	Test Case Name: Add Category via Settings. Subsystem: Category Management. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.			
2	Click on "Add new category" in the side menu	An Add Category screen will appear.			
3	Click on the "Camera" icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Category Name text field	Keyboard appears.			
7	Enter the Name of the category in the category text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the category in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the "Add" button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.			
11	Click on the "Camera" icon	Notification appears asking for an upload of the image via the camera or the gallery.			



12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new category detail are before.	Category detail displays.			
14	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.			
15	Click on the back button.	Category management option screen appears.			

Test Case Number: 27 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Editing a category.	Test Case Name Edit category. Subsystem: Category Management. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click Edit category	List with current categories appear.			
2	Click on the category you wish to edit	Category information will appear.			
3	Click on the category icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Category Name text field	Keyboard appears.			
7	Enter the Name of the category in the category text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the category in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the “Save changes” button.	Category is edited in the database. Notification of saved action. User remains in add the Add category screen.			
11	Click on the category icon.	Notification appears asking for an upload of the image via the camera or the gallery.			
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the			



		chosen picture— picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new category detail are before.	Category detail displays.			
14	Click the “Save changes” button.	Category is edited in the database. Notification of saved action. User remains in add the Add category screen.			
15	Click on the back button.	Category management option screen appears.			



Test Case Number: 28 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Removing a category from the catalogue.	Test Case Name: Delete category. Subsystem: Category Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Delete Category	A category selection screen appears.			
2	Select a category to delete.	Confirmation notification appears.			
3	Click No on the notification	Category selection screen appears.			
4	Select a category to delete	Confirmation notification appears.			
5	Select Yes on the notification	Category is deleted in the database. Successful notification appears.			
6	Click the back button	The category management option screen appears.			

Test Case Number: 29 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows user to sort the categories in multiple ways.	Test Case Name: Change Category Sorting. Subsystem: Category Management. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Change sorting	A sorting method screen appears.			
2	Click on the alphabetically toggle	Toggle turns off/on. Categories will be listed alphabetically. Can be viewed in the main menu.			
3	Click on the firstly added first toggle	Toggle turns on/off Categories is listed in a first added first view manner. Can be viewed in the main menu.			
4	Click on the Lastly added first toggle	Toggle turns on/off Categories is listed in a last added view first manner. Can be viewed in the main menu.			
5	Click on the back button.	The category management screen appears.			



Test Case Number: 30 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to return settings screen.	Test Case Name: Exit Category Management settings. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button.	The setting screen appears.			



Test Case Number: 31 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to manage their items in the catalogue.	Test Case Name: Item Management. Subsystem: Settings. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Item management	The item management screen appears.			

Test Case Number: 32 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to add a an item to the catalogue.	Test Case Name: Add item via Item Management. Subsystem: Item management. Design Date: Execution Date:
--	---

1	Click on “Add new item”	An Add Item screen will appear.			
2	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			
3	Click on Cancel	No photo were added.			
4	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.			
5	Click on Take Photo	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Select the category drop box	A list of the categories fill appears.			
7	Select the appropriate category from the drop box list	The selected category will appear. (Note: Not yet saved).			
8	Click in the Name of item text field	Keyboard appears.			
9	Enter the title of the item in the item text field	Title will appear in the text field. (Note: Not yet saved).			
10	Click in the Item description text field	Keyboard appears.			
11	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
12	Click the “Add” button.	Item is added in the database.			



		Notification of saved appears. User remains in the add item screen			
13	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture— picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
14	Enter the new item detail as before.	Item detail displays.			
15	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add item screen.			
16	Click on the back button.	Item management options appears.			



Test Case Number: 33 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to edit an item in the catalogue.	Test Case Name: Edit item via Item Management. Subsystem: Item Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click Edit Item	List with current items appear.			
2	Click on the item you wish to edit	Item information will appear.			
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Item Name text field	Keyboard appears.			
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.			
11	Click on the item icon.	Notification appears asking for an upload of the image via the camera or the gallery.			



12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new item detail are before.	Item detail displays.			
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.			
15	Click on the back button.	Item management option screen appears.			

Test Case Number: 34 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to delete an item in the catalogue.	Test Case Name: Delete item via Item Management. Subsystem: Item Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Delete Item	A category selection screen appears			
2	Click on the category that contain the item that will be deleted.	A item selection screen will appear.			
3	Click on the item you want to delete.	Delete notification appear for confirmation.			
4	Click on No	Item is not deleted. User remains in the item selection screen			
5	Click on the item you want to delete.	Delete notification appear for confirmation.			
6	Click on Yes	Item will be deleted from the database and the item will no longer be visible in the item list. User remains in the select item screen.			



Test Case Number: 35 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to delete an item in the catalogue.	Test Case Name: Delete item via Item Management with search as assistance. Subsystem: Item Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon	Keyboard will appear and a text field will open.			
2	Search for an item	Search item will appear if it was found else it will display no result found			
3	Click on the item you want to delete.	Delete notification appear for confirmation.			
4	Click on No	Item is not deleted. User remains in the item selection screen			
5	Click on the item you want to delete.	Delete notification appear for confirmation.			
6	Click on Yes	Item will be deleted from the database and the item will no longer be visible in the item list. User remains in the select item screen.			
7	Click on the back button.	The item management option screen appears.			



Test Case Number: 36 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allowing user to sort an item in the catalogue.	Test Case Name: Sorting the items in the catalogue. Subsystem: Item Management. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Change sorting	A sorting method screen appears.			
2	Click on the alphabetically toggle	Toggle turns off/on. Items will be listed alphabetically.			
3	Click on the firstly added first toggle	Toggle turns on/off Items is listed in a first added first view manner.			
4	Click on the Lastly added first toggle	Toggle turns on/off Items is listed in a last added view first manner.			
5	Click on the back button.	The item management screen appears.			

Test Case Number: 37 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Returning to the main settings screen.	Test Case Name: Exiting Item Management. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the back button	The settings screen appears.			



Test Case Number: 38 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Accessing help & tips for the app via settings.	Test Case Name: Help and Tips via Settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on “Help and Tips”	A Help and Tips screen will appear.			
2	Click on the video.	A tutorial video will play.			
3	Click on the Send feedback button	User is taken to google play store to leave feedback at google play store.			
4	Click the back button to leave play store.	The help and tip screen appear.			
5	Click the back button in the help and tips screen.	User is taken back to the settings screen.			

Test Case Number: 39 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Gives a brief history on the development team for eyeRS and the app via settings.	Test Case Name: About via settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on “About”	A About screen will appear. Which contains the information about the app.			
2	Click the back button.	The settings screen appears.			



Test Case Number: 40 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Returning to the main menu to proceed with other functions that the app provide.	Test Case Name: Exit the settings. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button.	Main menu appears.			

Test Case Number: 41 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Searches for items in the catalog from the main screen.	Test Case Name: Search Item from the main screen. Subsystem: Main Menu. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon	Search text field will slide from the side and keyboard will appear.			
2	Search any item	Keyboard will disappear and search results will appear.			

Test Case Number: 42 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: View and edit items from the catalog directly from the search feature.	Test Case Name: View and Edit Item direct from the search feature. Subsystem: Item Detail. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the item you want to view or change.	Item detail screen display.			
2	Click Edit Item button	List with current items appear.			
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Item Name text field	Keyboard appears.			
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.			
11	Click on the item icon.	Notification appears asking for an upload of the image via the camera or the gallery.			



12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new item detail are before.	Item detail displays.			
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.			
15	Click on the back button.	Item detail screen appears.			



Test Case Number: 43 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item once in a searched item.	Test Case Name Share Item once in searched item. Subsystem: Item Detail. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the “Share” option in the pop up screen	Notification that asks you to choose a sharing method.			
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.			

*Note: Repeat for each sharing method.



Test Case Number: 44 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item once in a searched item.	Test Case Name Trade Item once in searched item. Subsystem: Item Detail. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the “Trade” option in the pop up screen	Notification that asks you to choose a sharing method.			
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.			

*Note: Repeat for each sharing method.

Test Case Number: 45 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category.	Test Case Name Delete Item once in searched item. Subsystem: Item Detail. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.			
3	Click the “No” button in the Delete conform notification	Delete confirm notification will disappear and the item detail screen will appear.			
4	Click on the more icon.	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
5	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.			
6	Click the “Yes” button in the Delete conform notification	Deleted notification appears. Item is deleted from the database. Main menu will appear			



Test Case Number: 46 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: View item from a category.	Test Case Name View items via main menu. Subsystem: Main menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on a category to view items.	Items in the category will display in a listview.			



Test Case Number: 47 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category.	Test Case Name Edit items via main menu. Subsystem: Main menu. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the item you want to view or change.	Item detail screen display.			
2	Click Edit Item button	List with current items appear.			
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.			
4	Click on Cancel	No photo were added.			
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
6	Click in the Item Name text field	Keyboard appears.			
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).			
8	Click in the Description text field	Keyboard appears.			
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).			
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.			
11	Click on the item icon.	Notification appears asking for an upload of the image via the camera or the gallery.			

12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture— picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).			
13	Enter the new item detail are before.	Item detail displays.			
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.			
15	Click on the back button.	Item detail screen appears.			

Test Case Number: 48 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item via main menu.	Test Case Name Share Item via main menu. Subsystem: Item Detail. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the “Share” option in the pop up screen	Notification that asks you to choose a sharing method.			
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.			

*Note: Repeat for each sharing method.

Test Case Number: 49 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item from the main menu.	Test Case Name Trade Item from the main menu. Subsystem: Item Detail. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the "Trade" option in the pop up screen	Notification that asks you to choose a sharing method.			
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.			
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.			

*Note: Repeat for each sharing method.

Test Case Number: 50 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category from the main menu.	Test Case Name Delete Item from the main menu. Subsystem: Item Detail. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
2	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.			
3	Click the “No” button in the Delete conform notification	Delete confirm notification will disappear and the item detail screen will appear.			
4	Click on the more icon.	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.			
5	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.			
6	Click the “Yes” button in the Delete conform notification	Deleted notification appears. Item is deleted from the database. Item list appears.			



Test Case Number: 51 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Exit from viewing the items.	Test Case Name: Exit from viewing items Screen. Subsystem: My Category. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the back button	Main screen will appear.			

Test Case Number: 52 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Leaving the eyeRS app.	Test Case Name: Exit eyeRS. Subsystem: Main Menu -Side menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more button n the main menu.	A side menu appears.			
2	Click on "Exit"	A notification will appear.			
3	Click the NO	Main Menu will appear.			
4	Click on the more button n the main menu.	A side menu appears.			
5	Click on "Exit"	A notification will appear.			
6	Click the Yes	User will exit the eyeRS app.			

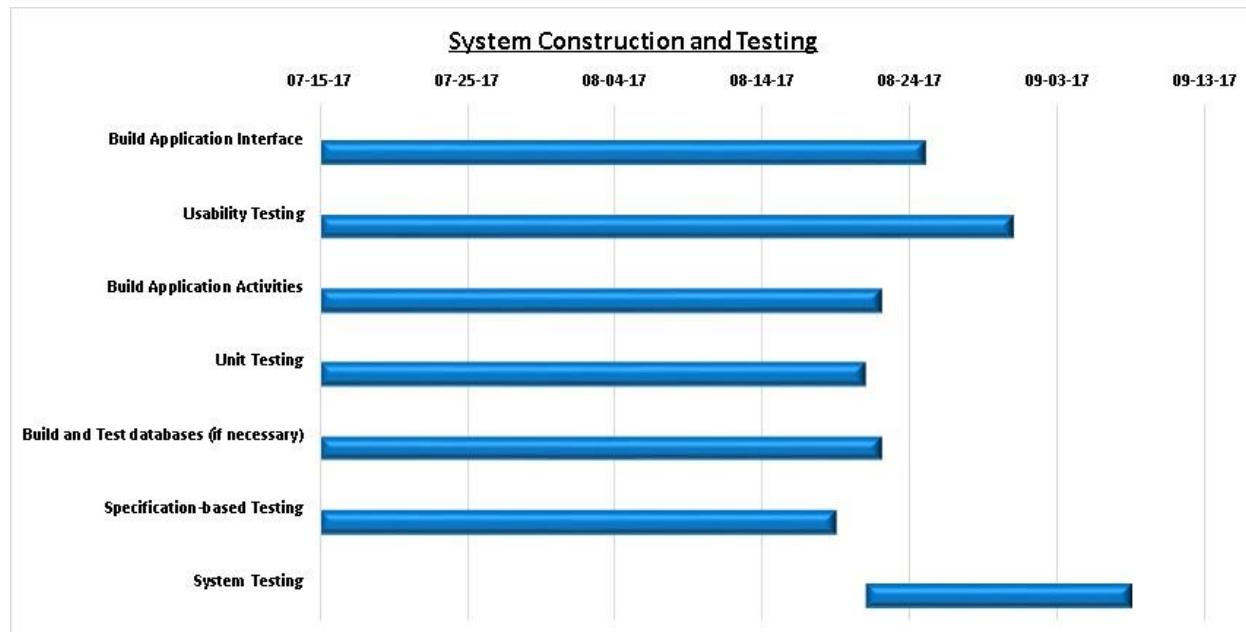
Test plan

Introduction

Our project is a mobile app, which is intended for operation on the Android OS platform. Hardware resources intended for testing will comprise of personal computers for compiling and emulation, together with Android smart-phones and tablets for testing on real devices. Software being used includes emulators, Javadocs, GitHub and the Android Developer console for additional resources. The team is currently using Android Studio which has various mobile and tablet device emulators that will utilize a computer's different Hyper-V features to run the emulators. All resources are available for each member of the eyeRS development team. Testing and debugging could take up to a total of approximately 48 hours.

Test Plan - eyeRS Mobile App			
Test type	Test date	Team members	Description
Unit (White box)	30/08/2017	Mr. Matthew Van Der Bijl, Mr. Nathan Shava	All the activities and/or fragments' code statements will be tested to ensure that they execute as expected.
Unit (Black box)	31/08/2017	Mr. Matthew Van Der Bijl, Mr. Nathan Shava	The app's activities and/or fragments will be tested to verify that each activity can process certain inputs and meet proposed requirements.
Integration	06/09/2017	Mrs. Ndai Makhurane (Customer), Mr. Matthew Van Der Bijl, Mr. Nathan Shava, Ms. Andrea Cloete, Mr. Sajjaad Ishmail, Mr. Emilde Arsenio	Here the entire app's activities and/or fragments will be integrated to verify whether they interact all as expected.
System & Stress	07/09/2017	Mrs. Ndai Makhurane (Customer), Mr. Matthew Van Der Bijl, Mr. Nathan Shava, Ms. Andrea Cloete, Mr. Sajjaad Ishmail, Mr. Emilde Arsenio	The performance of the system will be tested at this phase to find out whether the app's responsiveness, for example, is at an acceptable level.

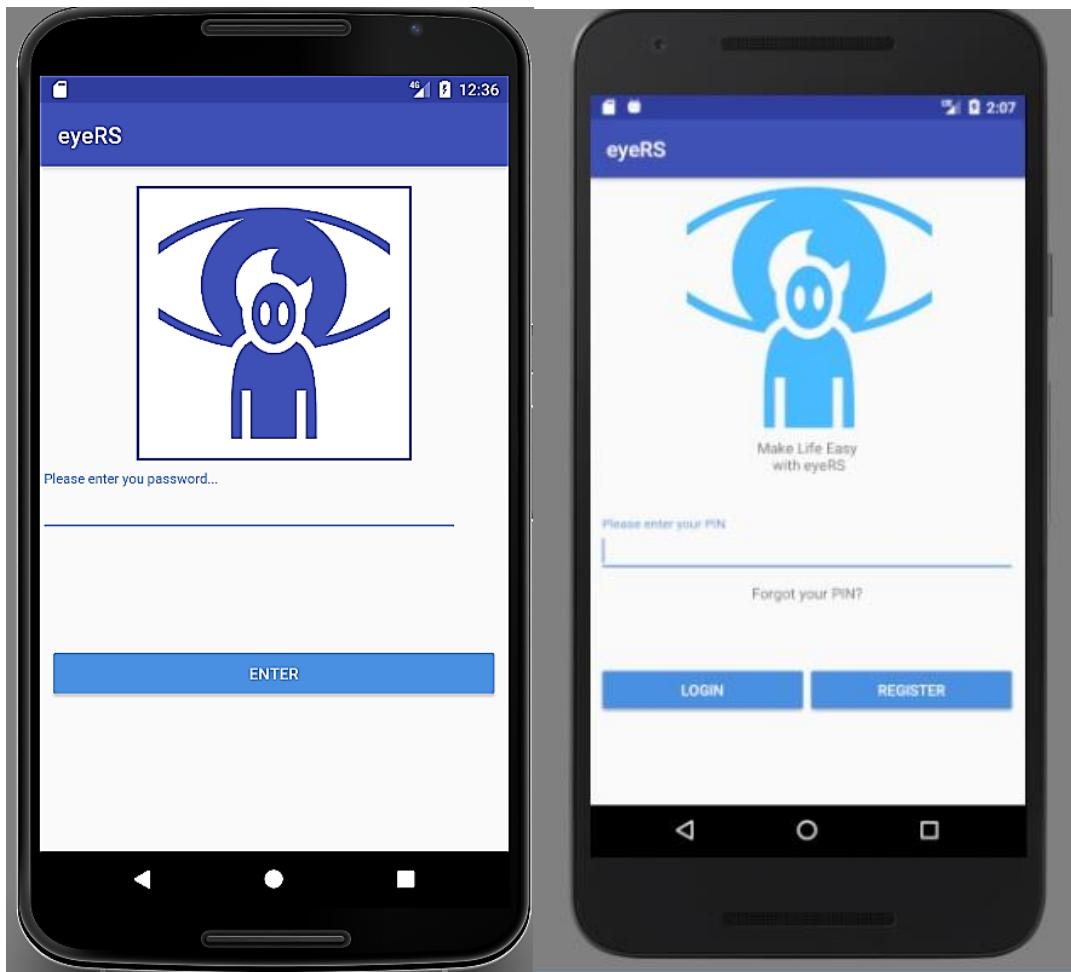
Entire App Walkthrough (Usability)	07/09/2017	Mrs. Ndai Makhurane (Customer), Mr. Matthew Van Der Bijl, Mr. Nathan Shava, Ms. Andrea Cloete, Mr. Sajjaad Ishmail, Mr. Emilde Arsenio	The entire app will be tested simulating the entire user experiences to debug for any other errors not detected during the other phases.
Acceptance	13/09/2017	Mrs. Ndai Makhurane (Customer), Mr. Matthew Van Der Bijl, Mr. Nathan Shava, Ms. Andrea Cloete, Mr. Sajjaad Ishmail, Mr. Emilde Arsenio	The proposed customer will test the app to verify if it has met all the stipulated requirements.
Review & Corrections	14/09/2017	Mr. Matthew Van Der Bijl, Mr. Nathan Shava	Time allocated for corrections based on the review process.

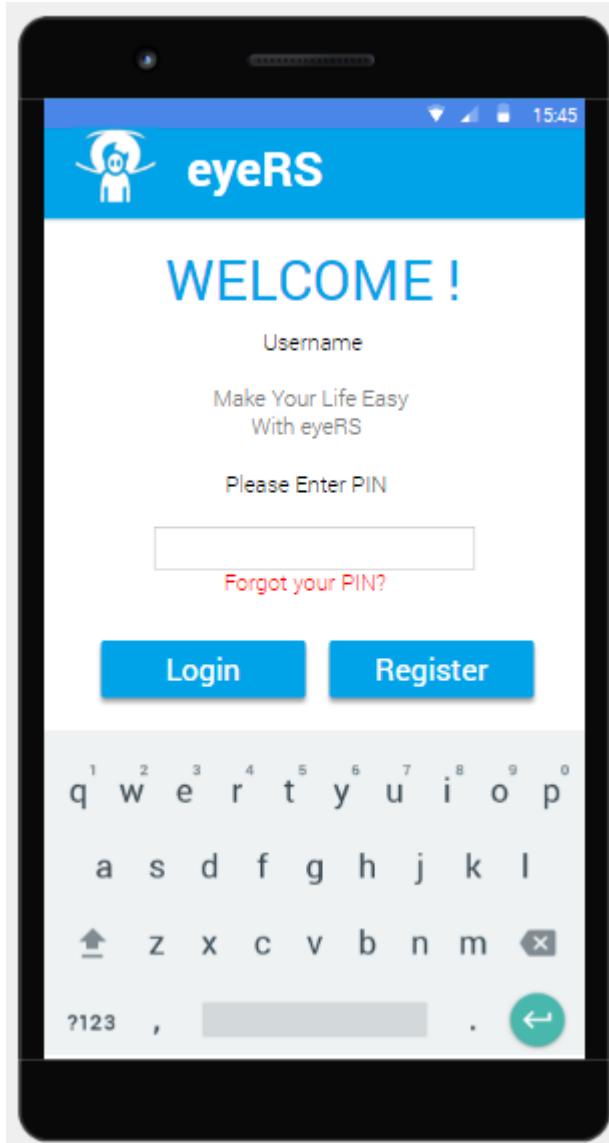


1.2.3 System interface design

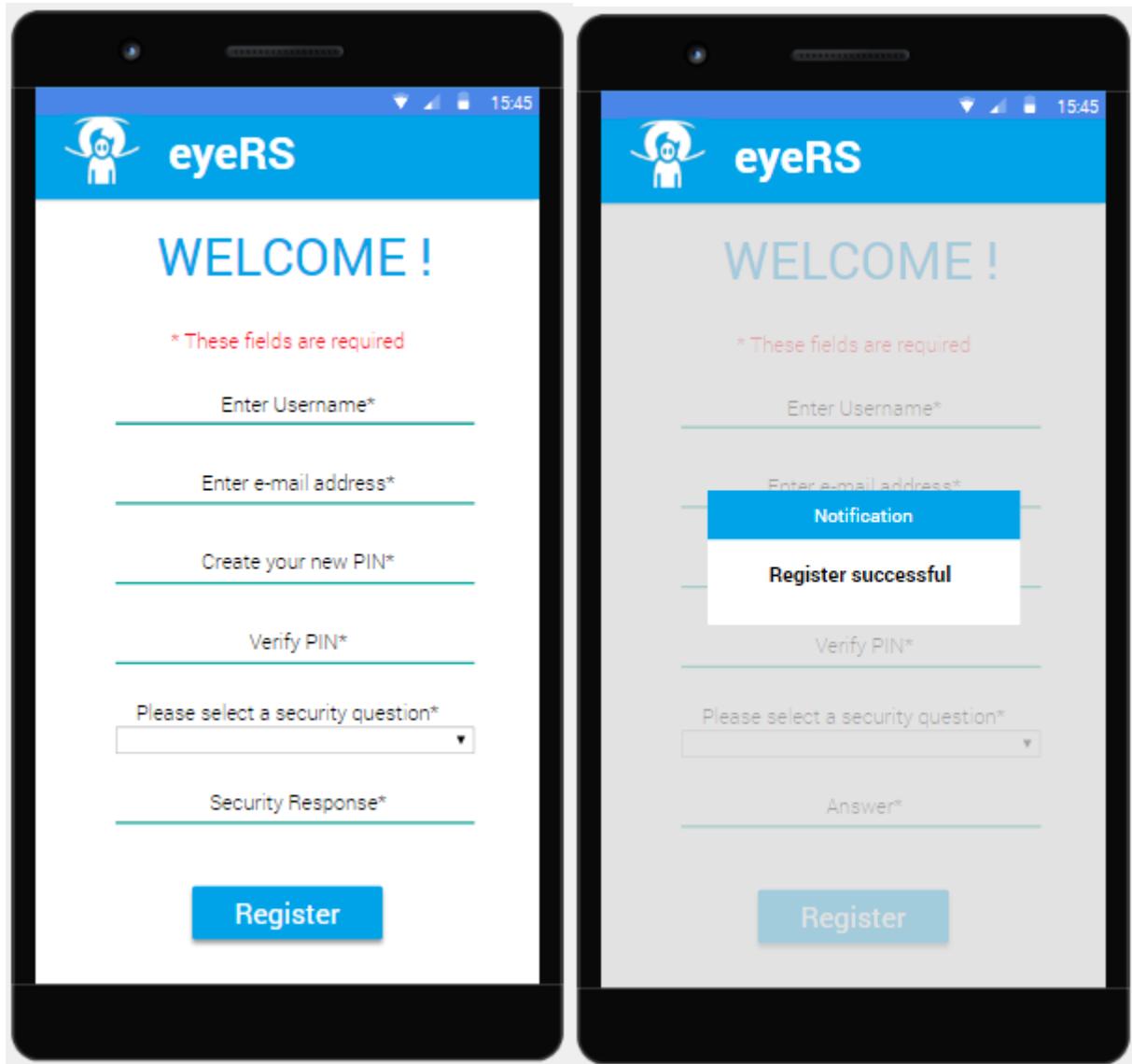
Login Screen

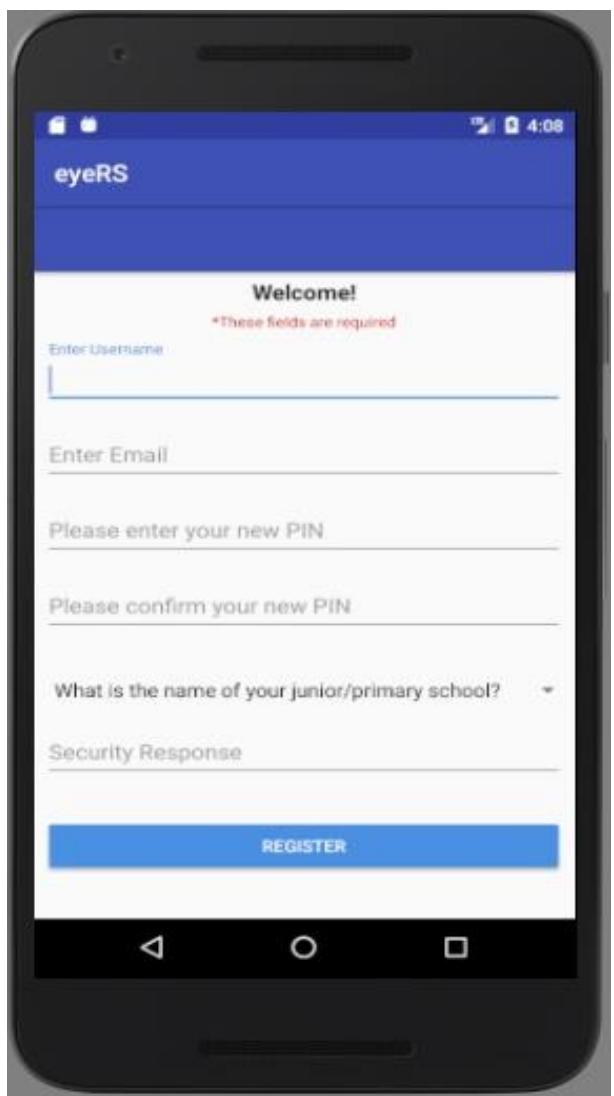
This screen will allow the user to gain access to the app by only supplying a PIN as there can only be one sole user of the app per device.



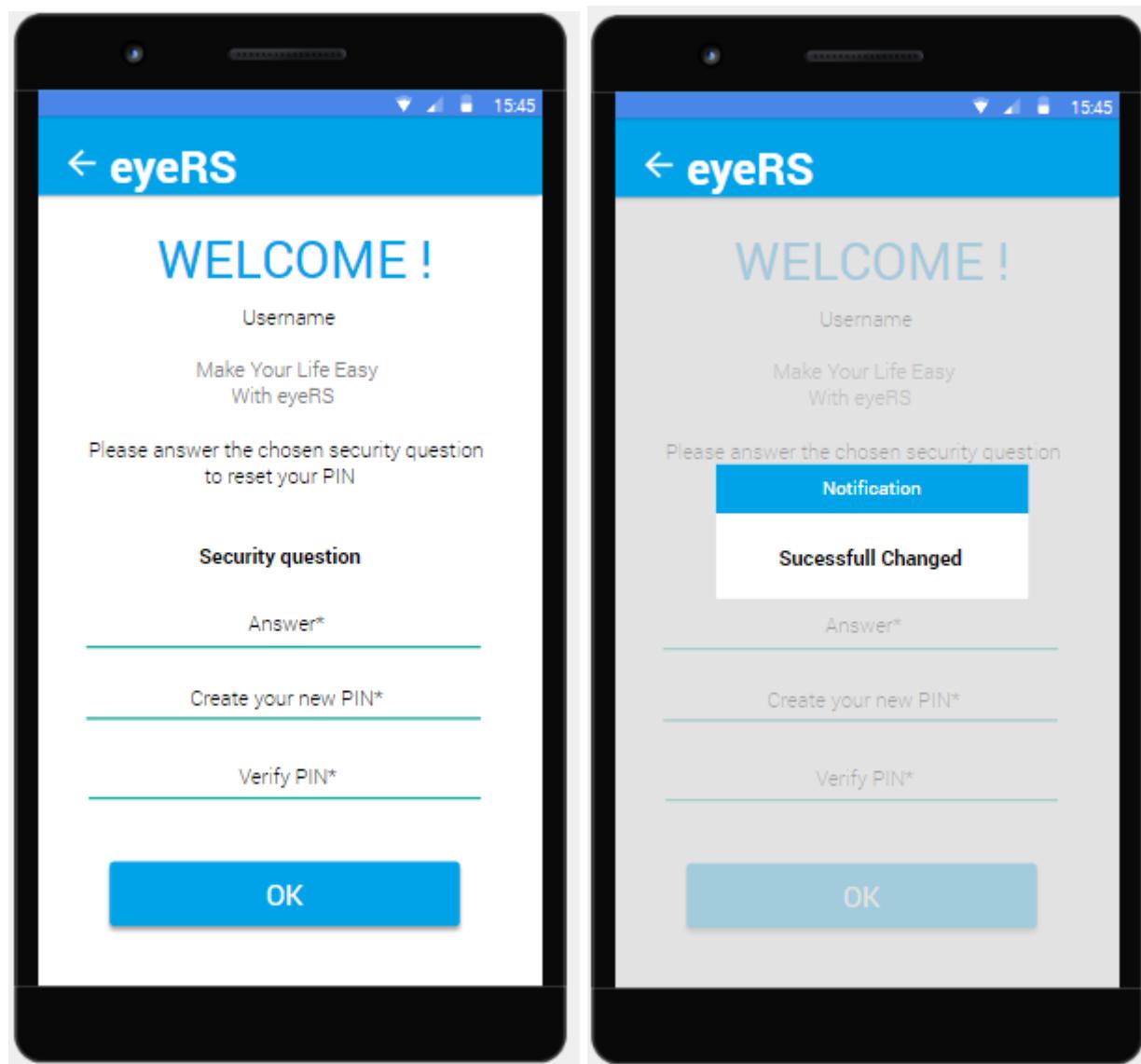


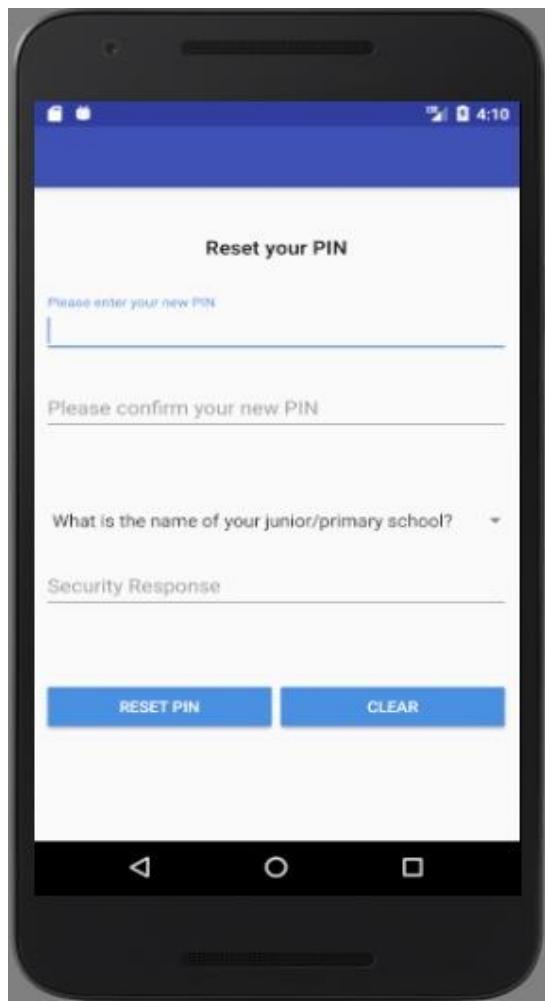
Register Screen





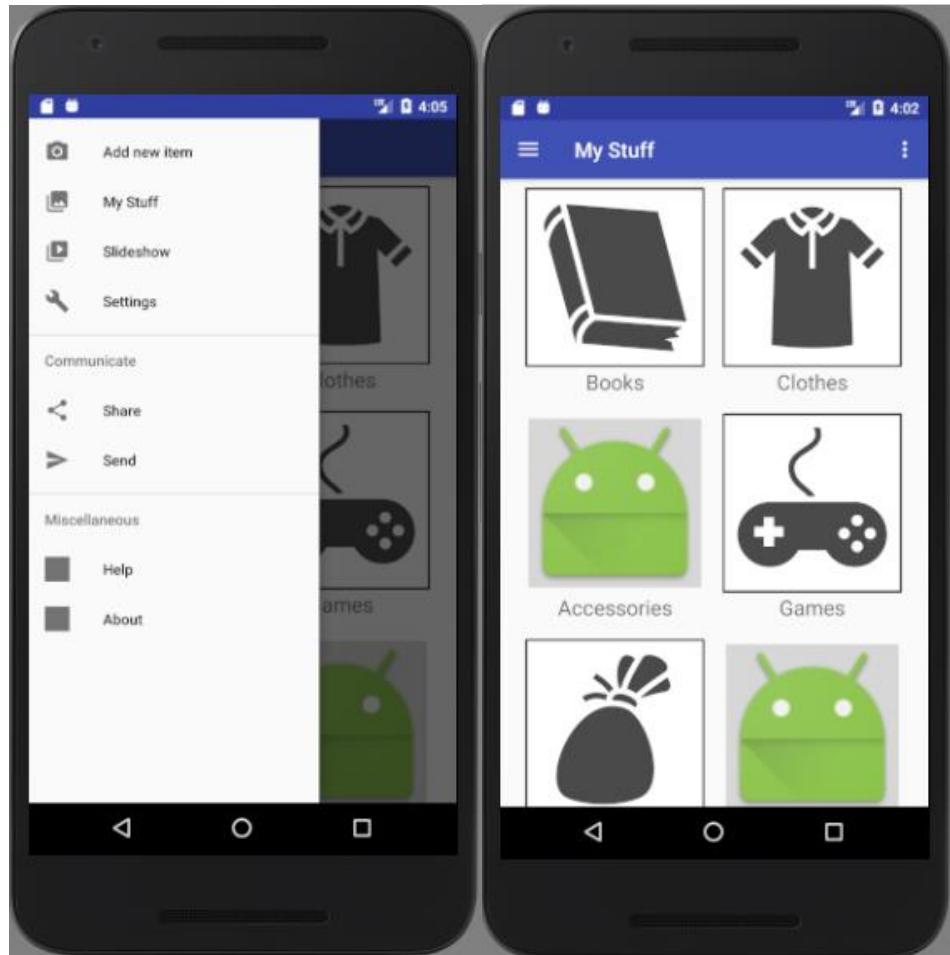
Forgot PIN Screen



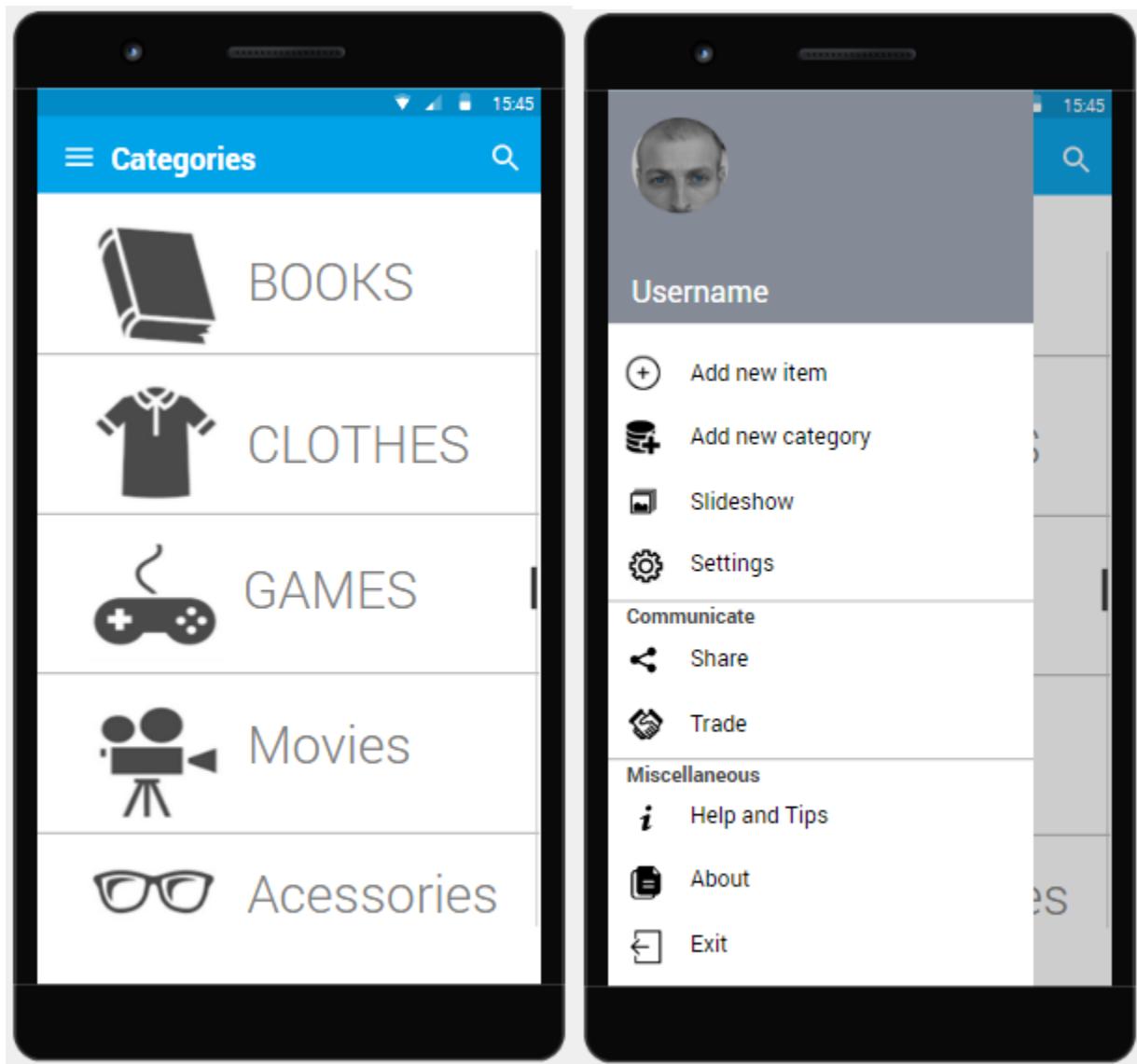


Main Menu Screen

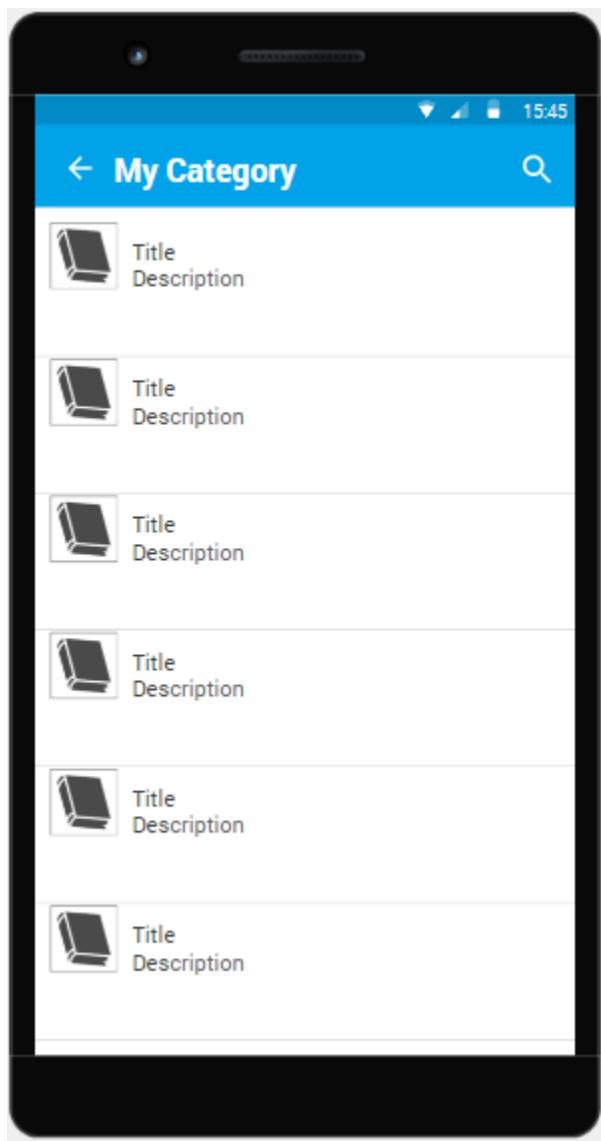
This screen will consist of a Navigation drawer in which the user can access by swiping right from the left edge of the screen or by clicking on the drawer button found on the top left corner of the screen on the action bar.



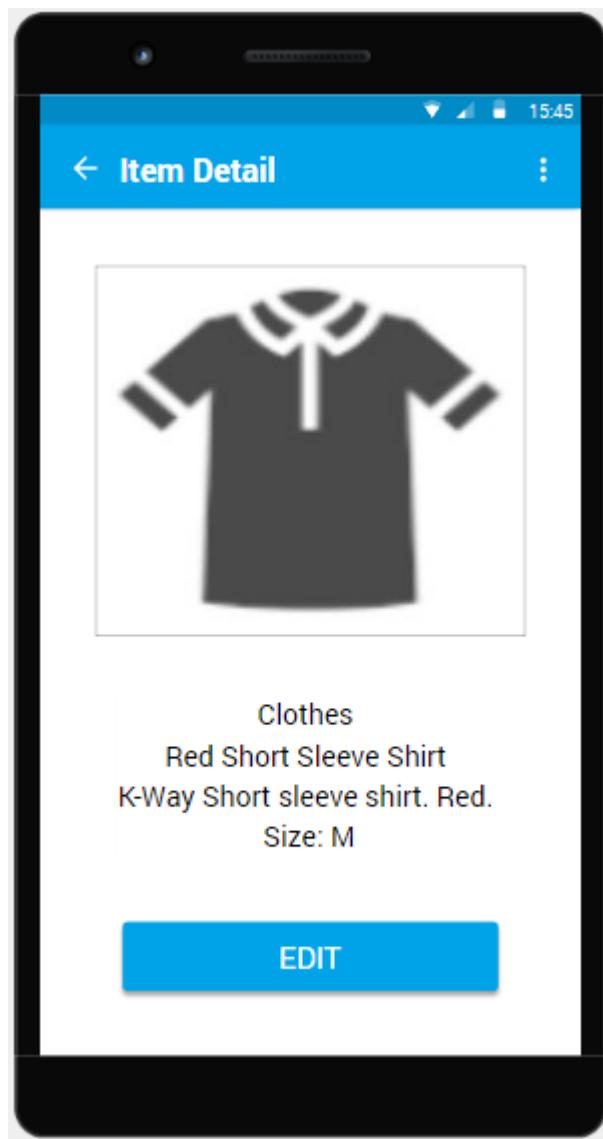
The My Stuff screen also features a graphical UI consisting of clickable icons representing the app's catalog categories.



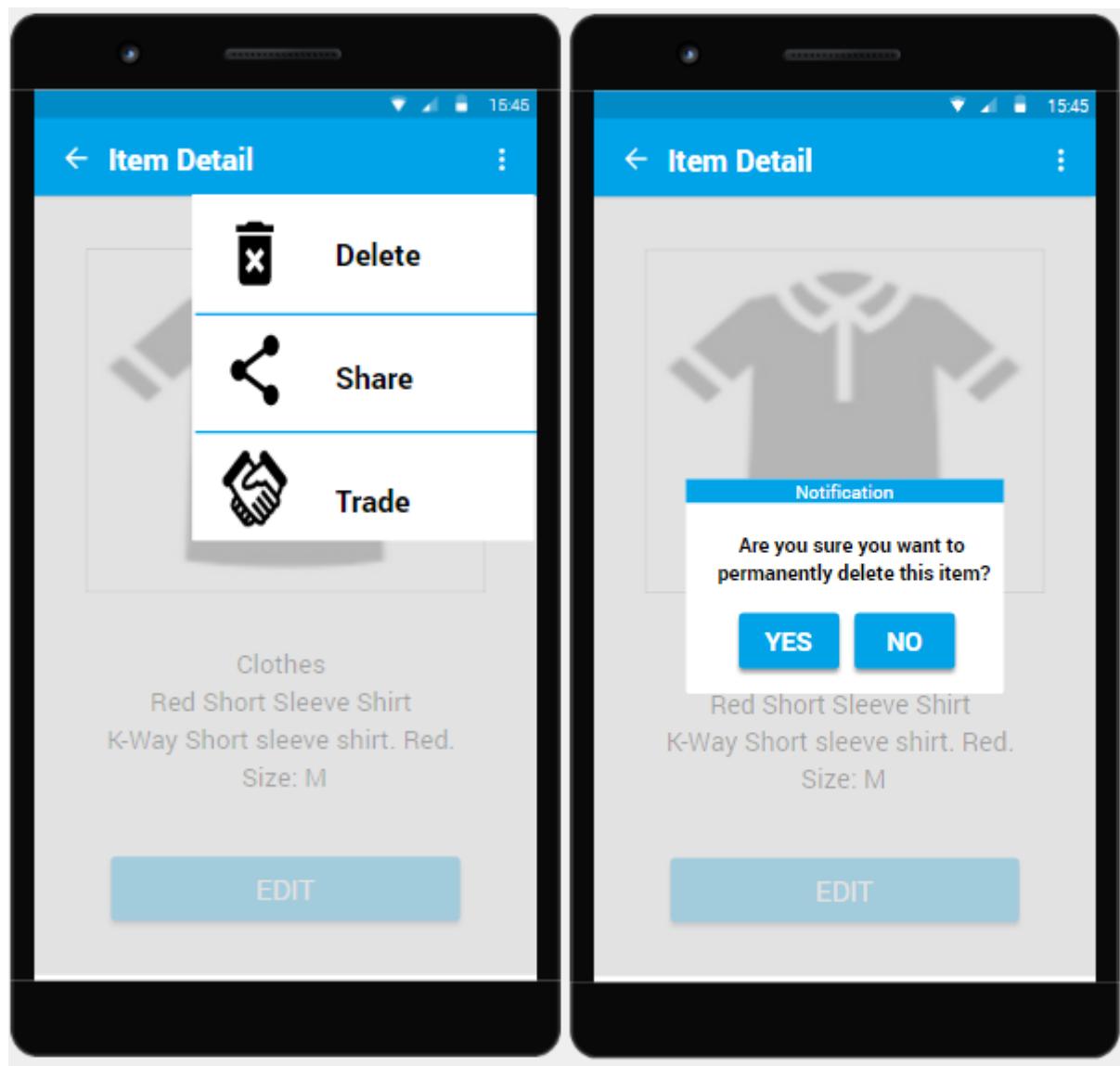
View Item Screen



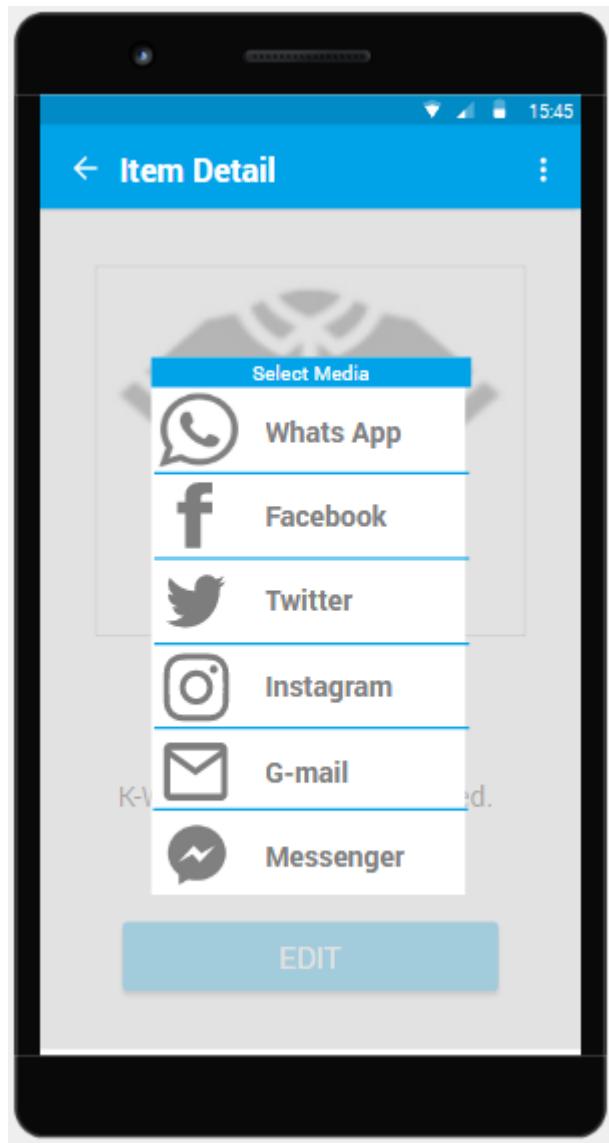
Item Detail Screen



Item Delete Screen

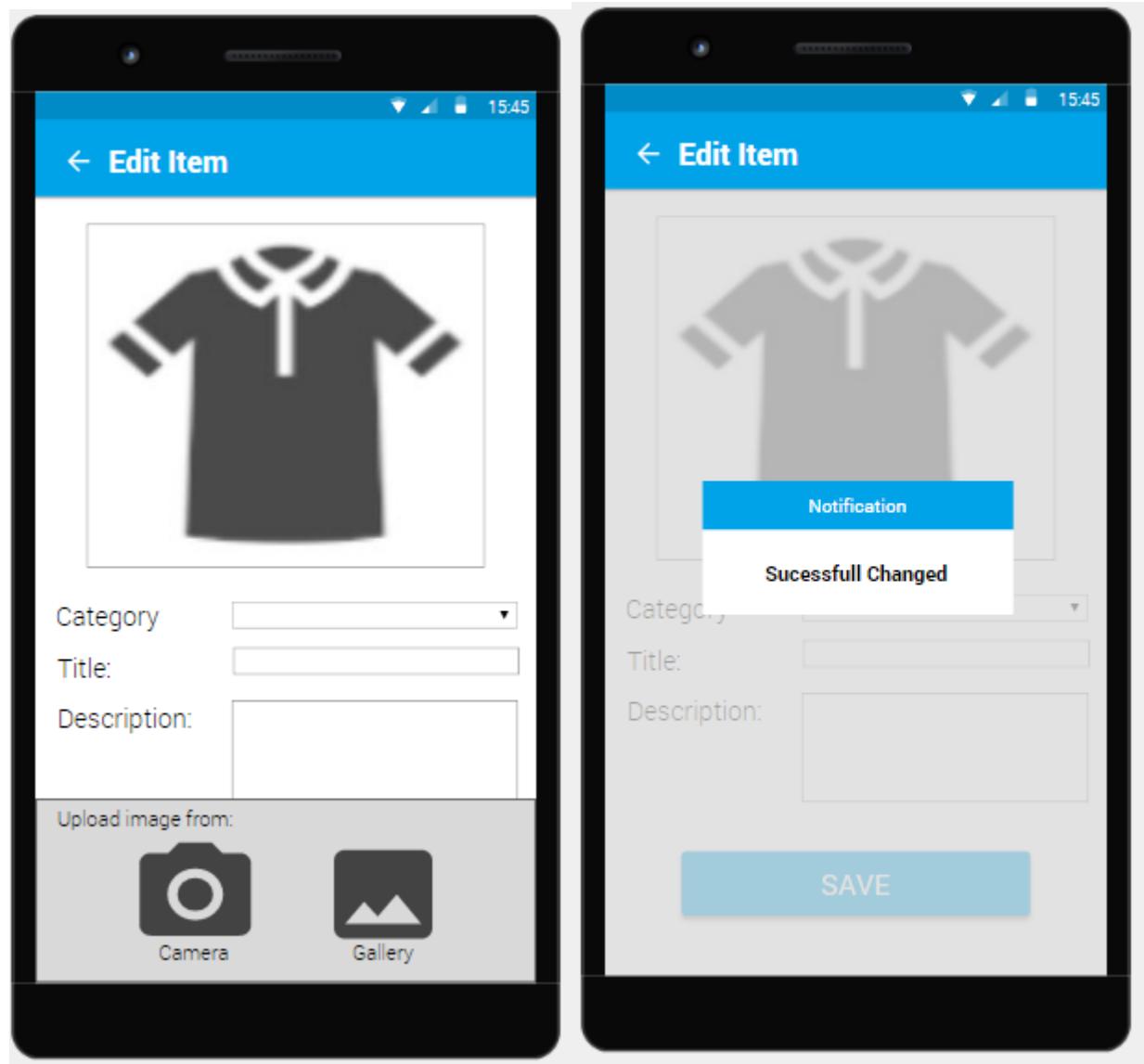


Item Share and Trade Screen

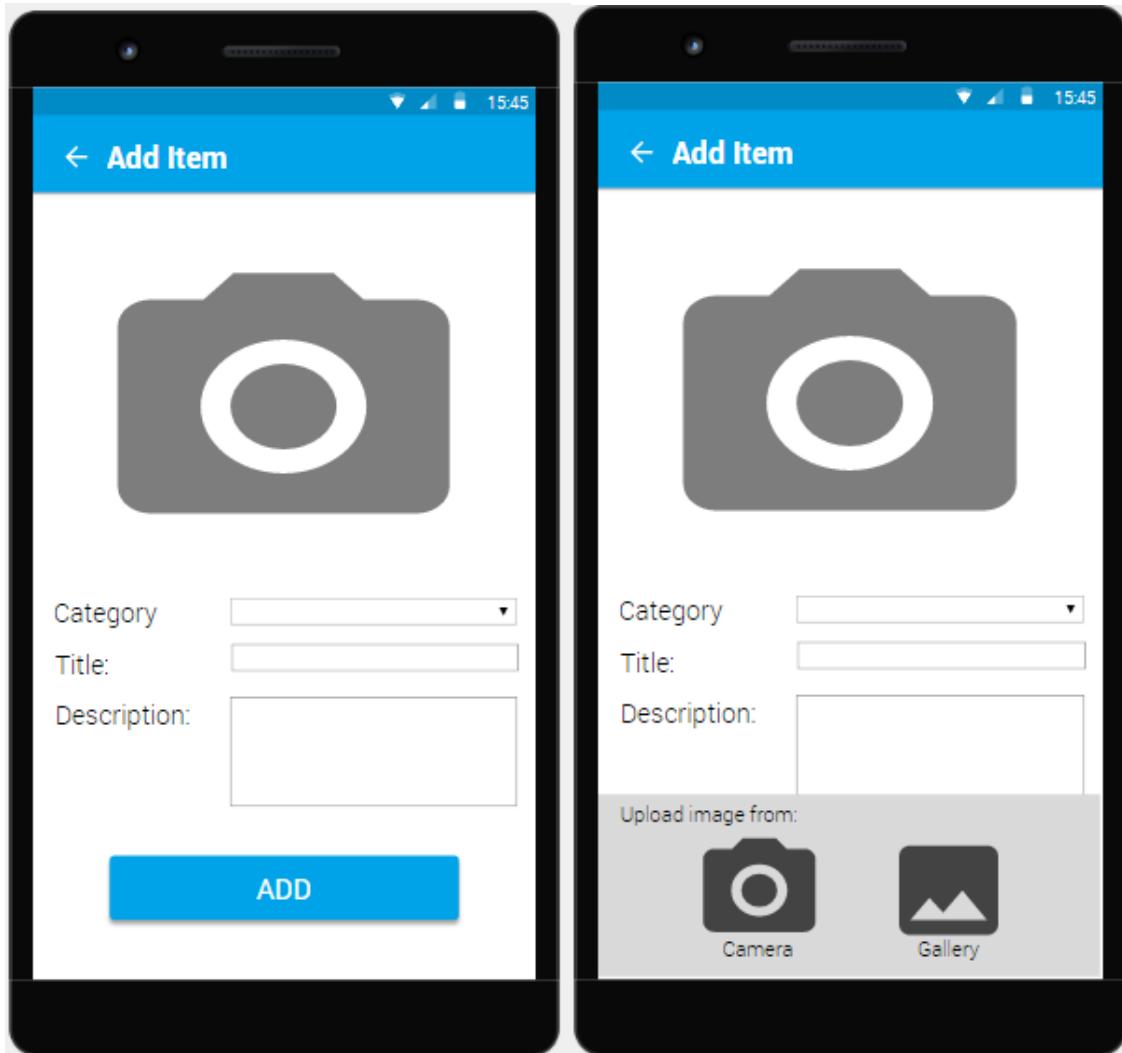


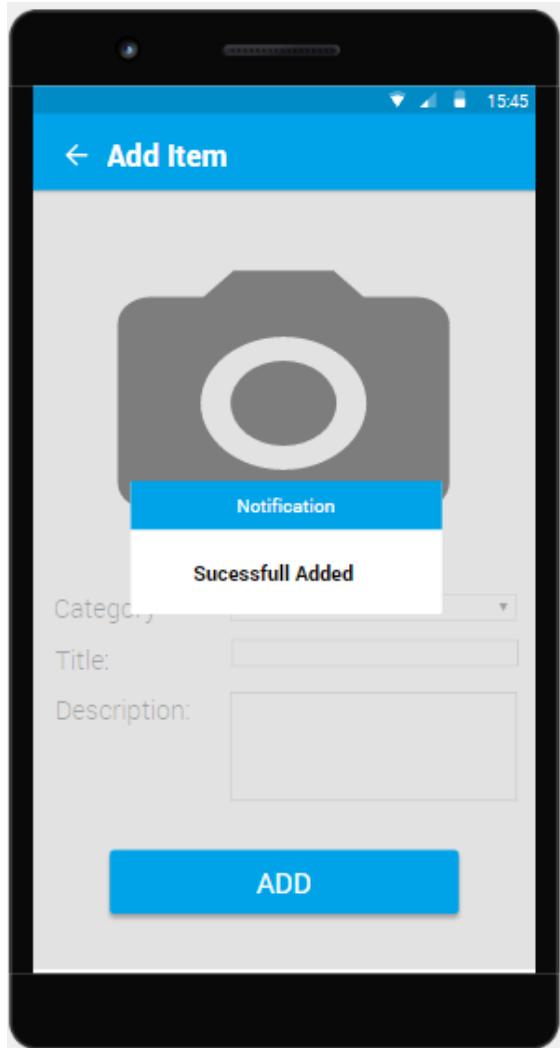
Edit Item Screen



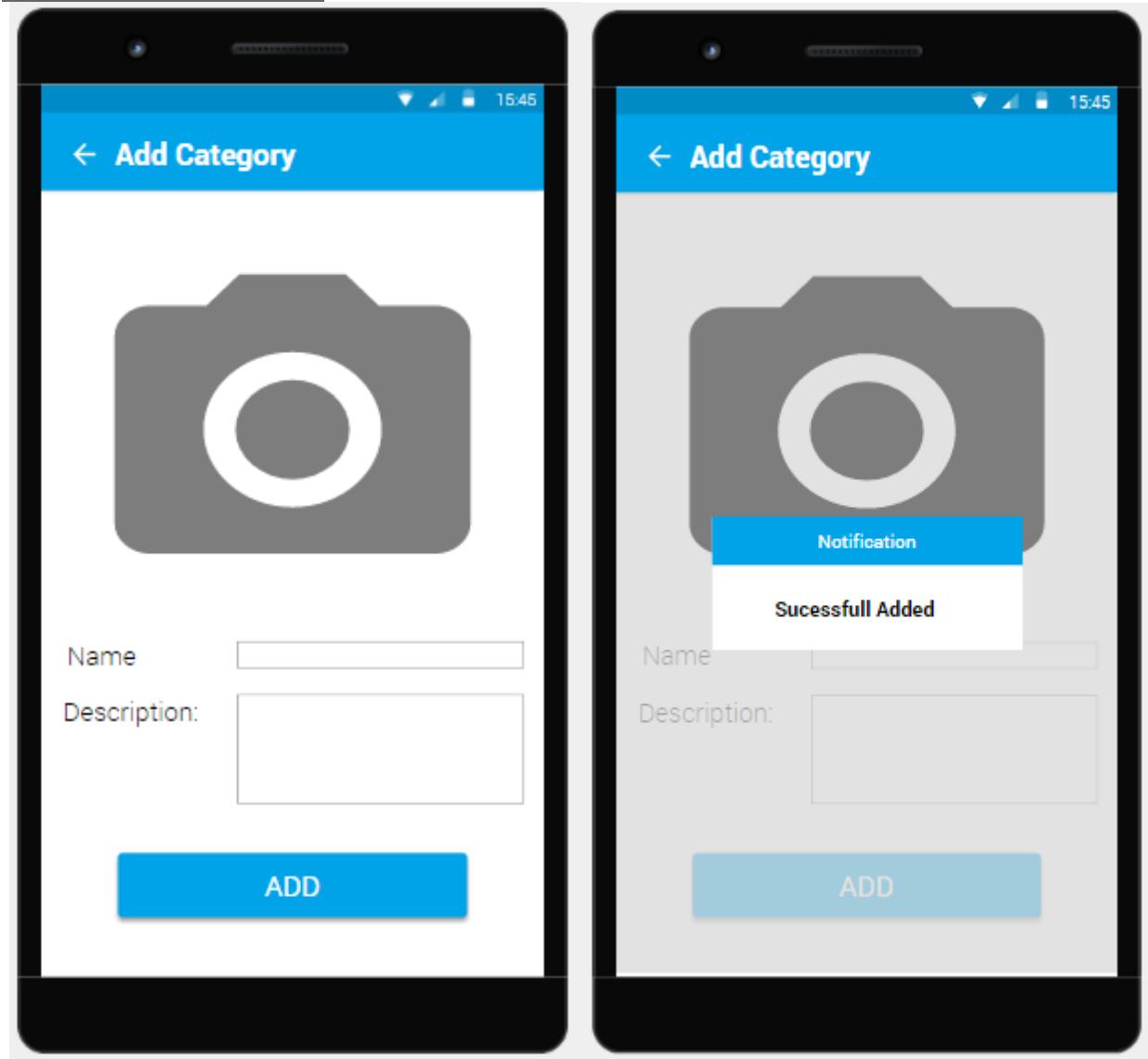


Add Item Screen





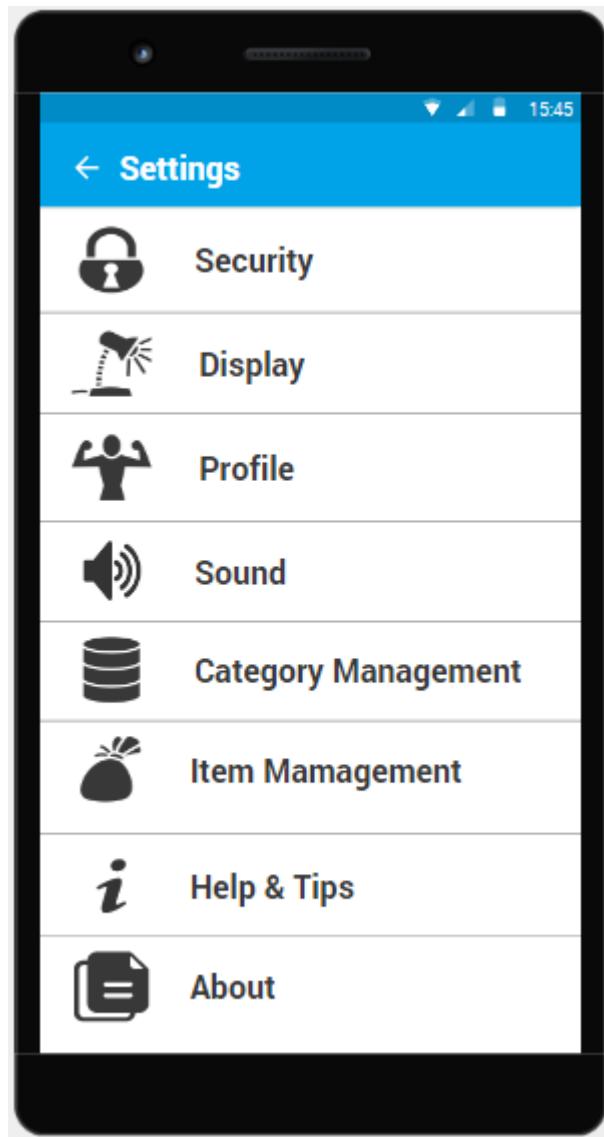
Add New Category Screen



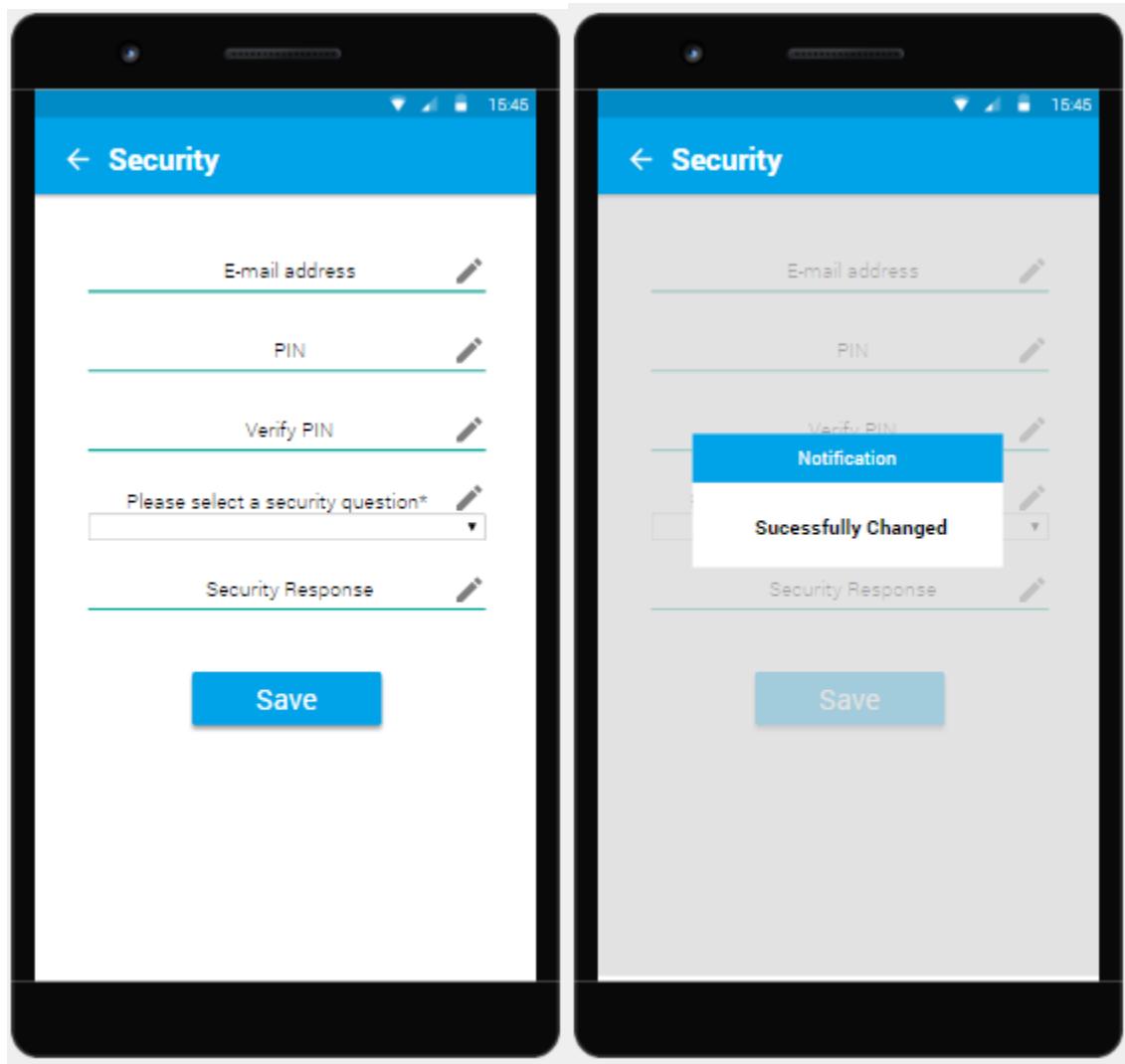
Slideshow Screen



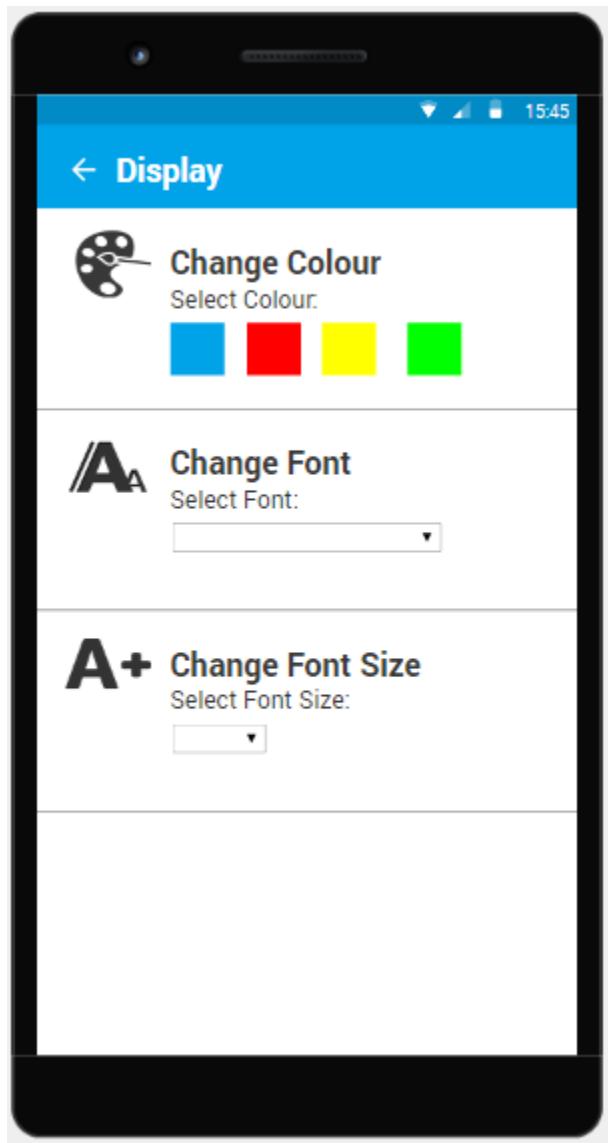
Settings Screen



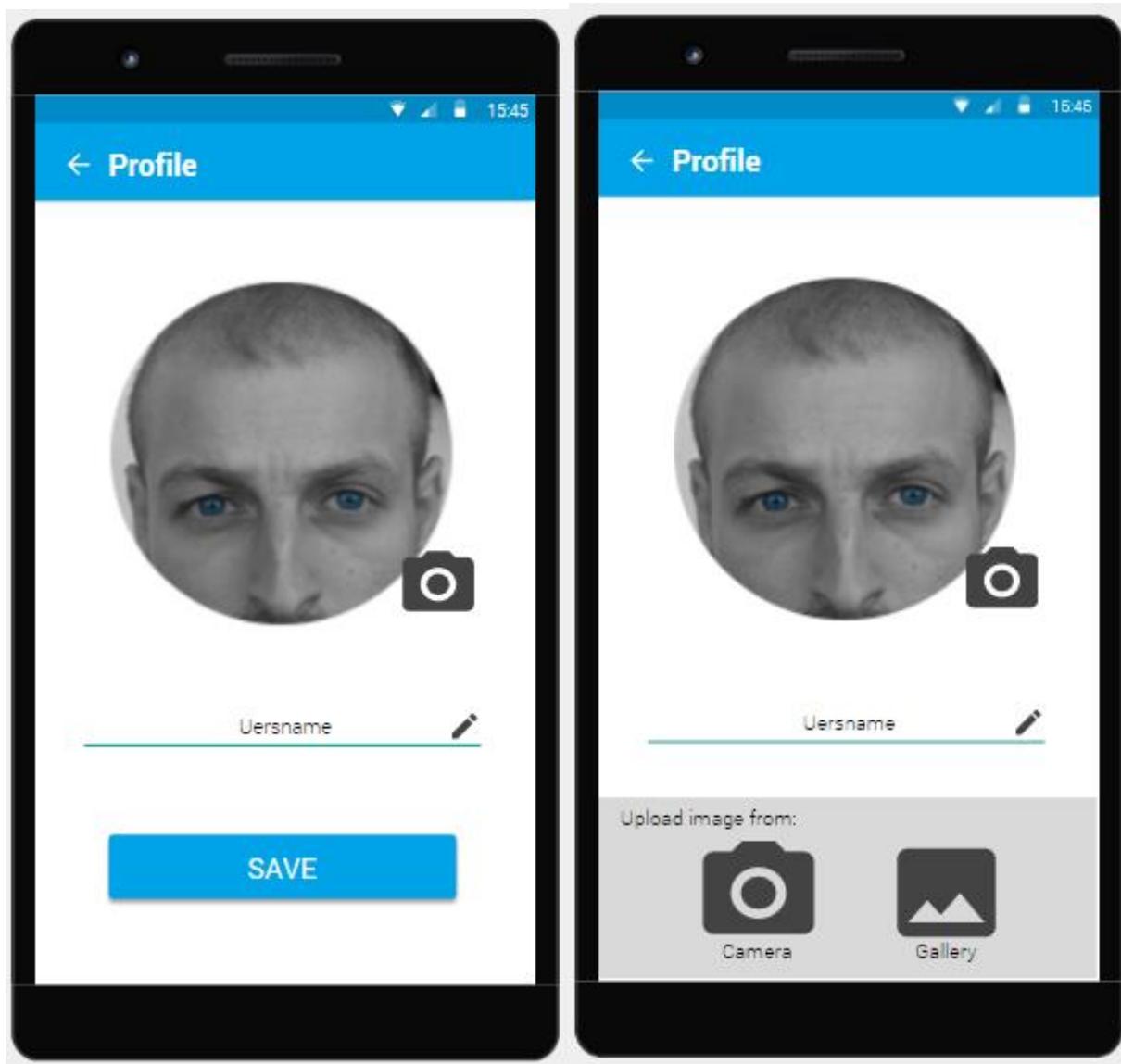
Settings > Security Screen



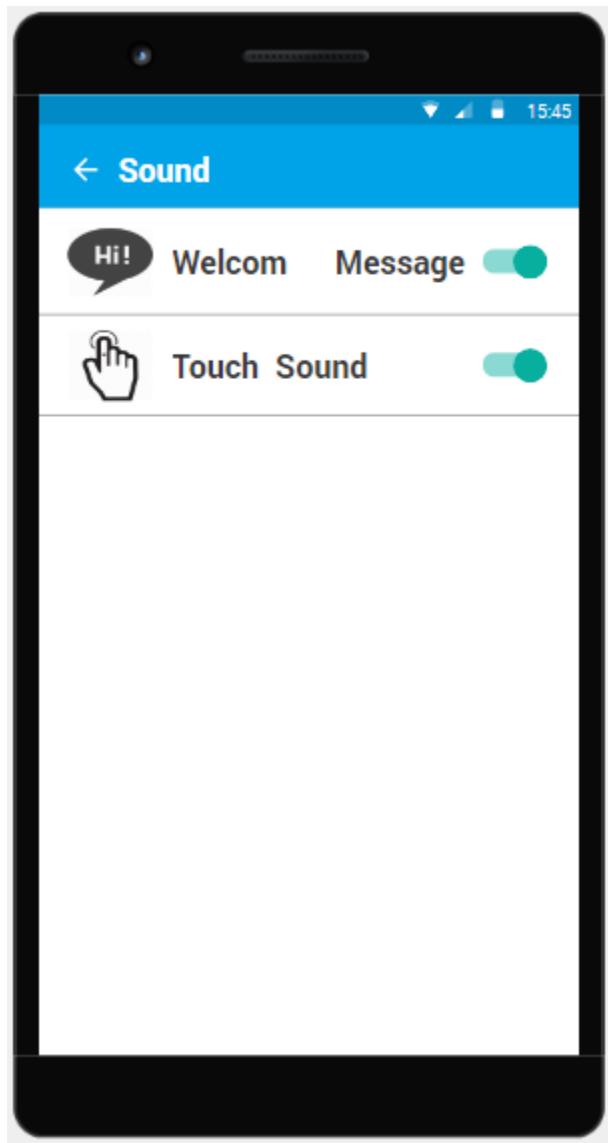
Settings > Display Screen



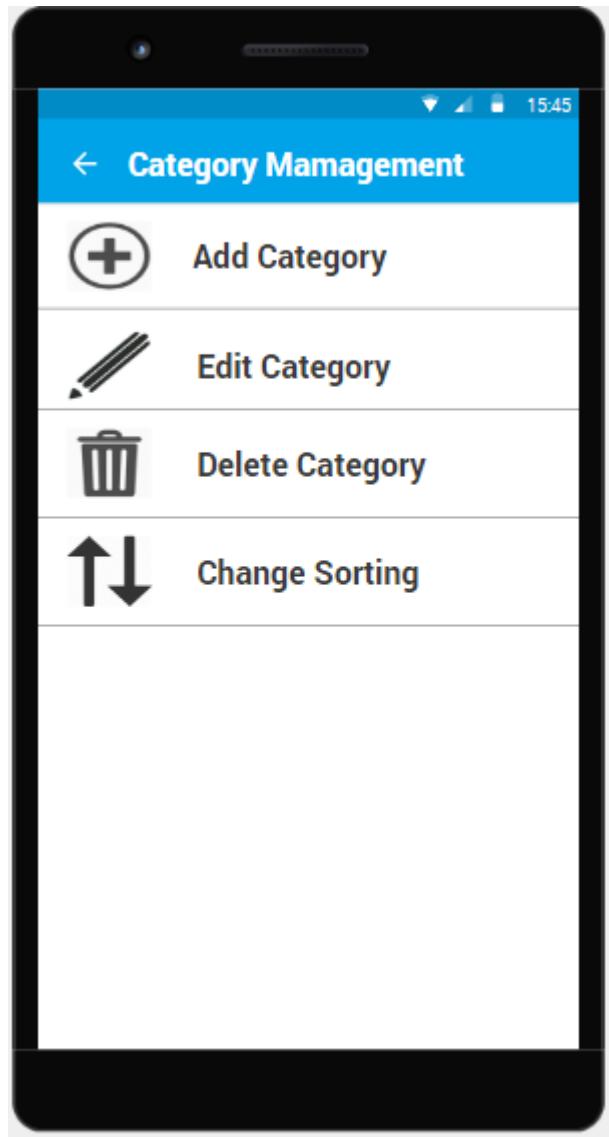
Settings > Profile Screen



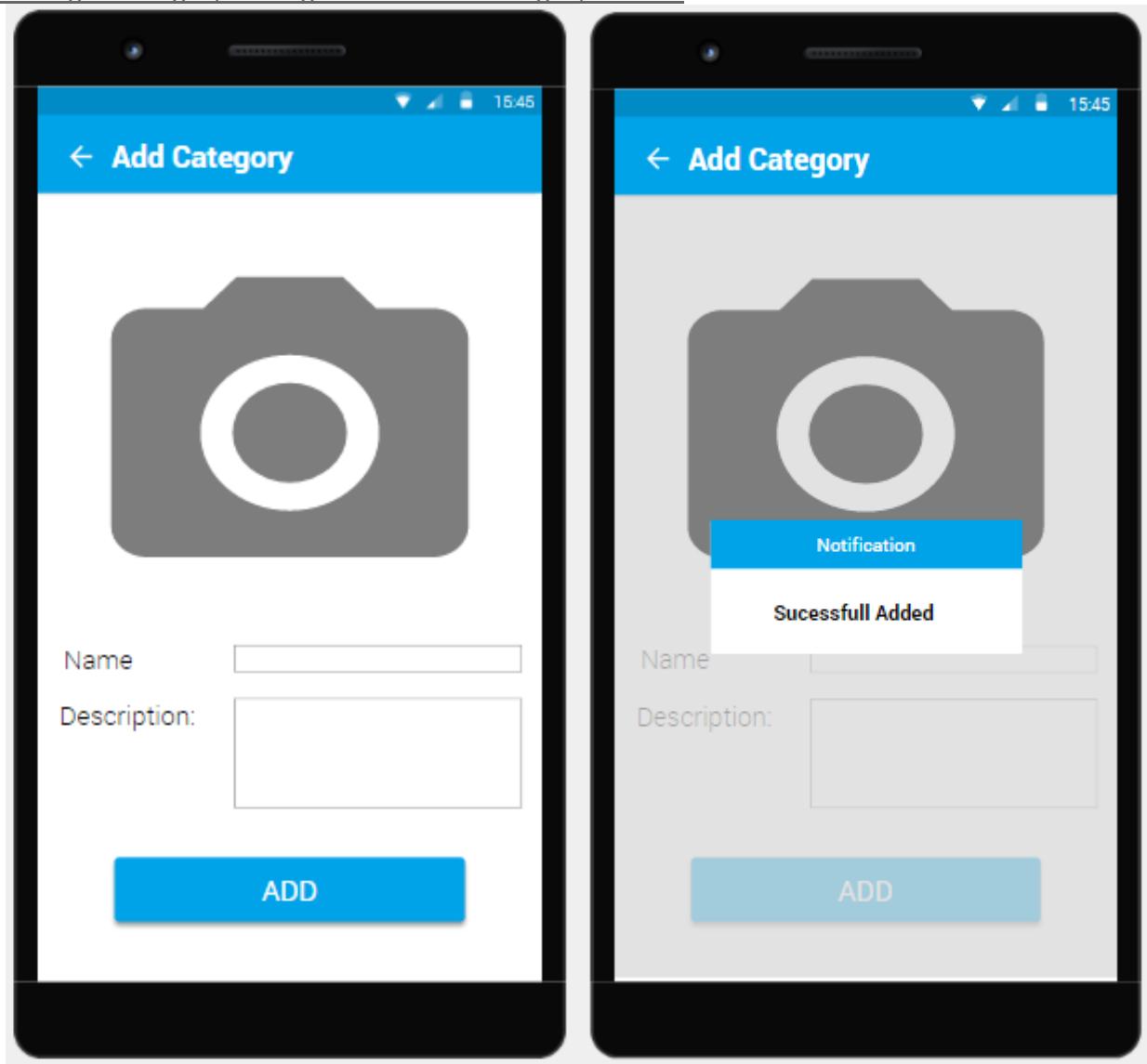
Settings > Sound Screen



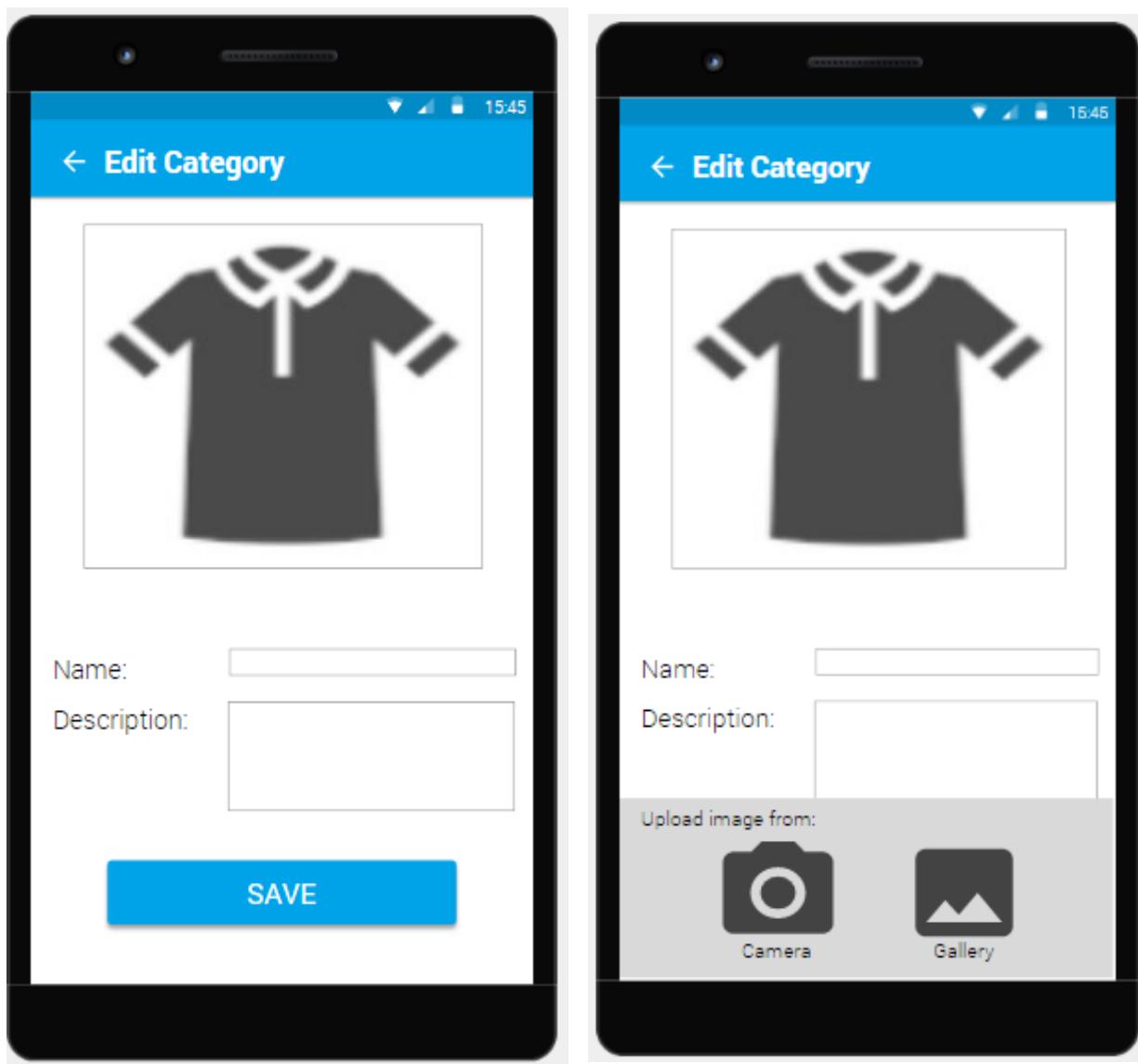
Setting > Category Management Screen

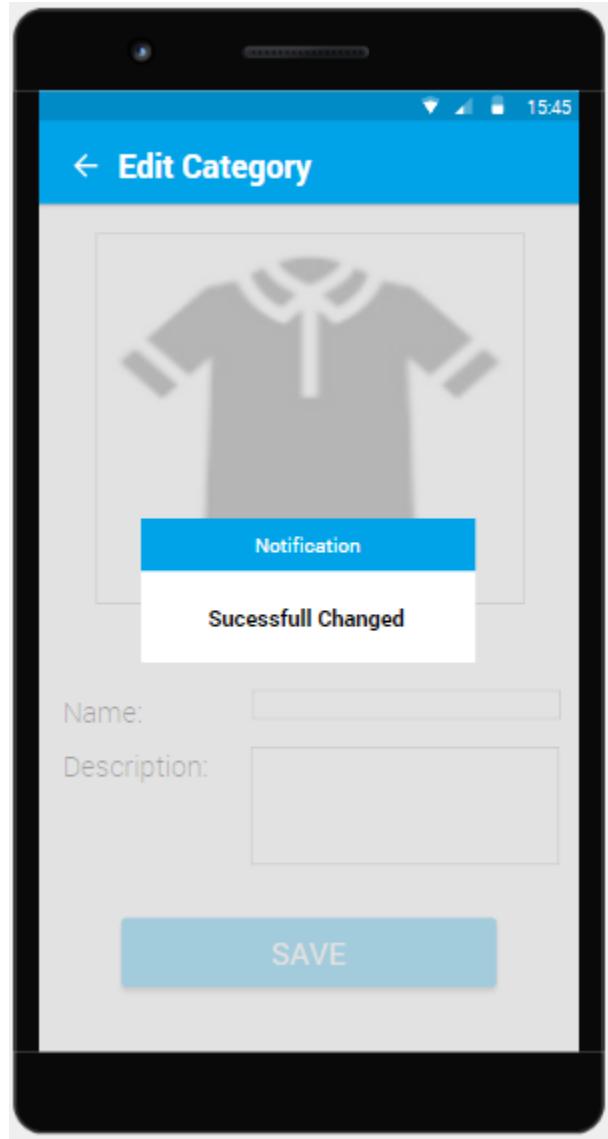


Setting > Category Management > Add Category Screen

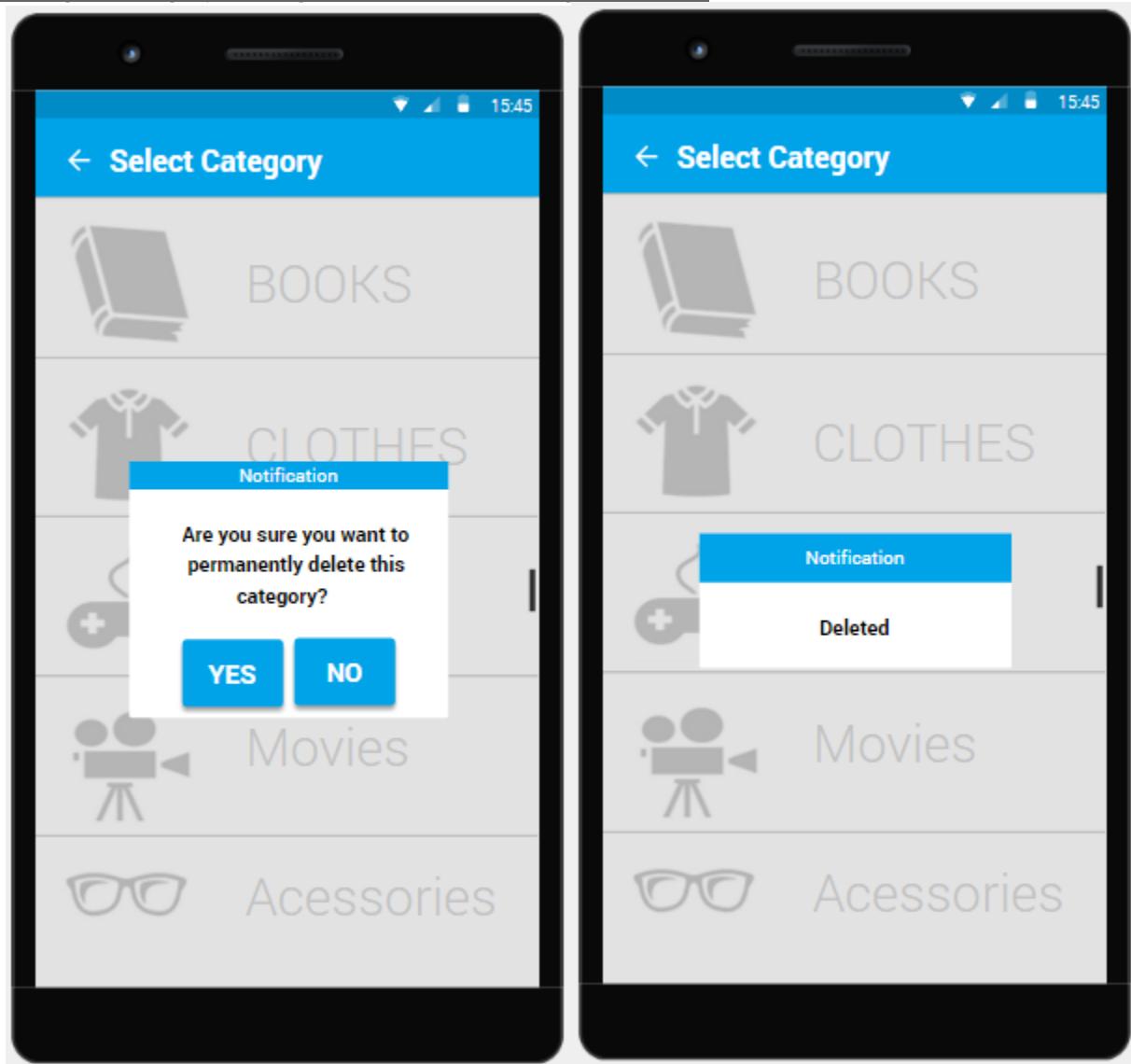


Setting > Category Management > Edit Category Screen

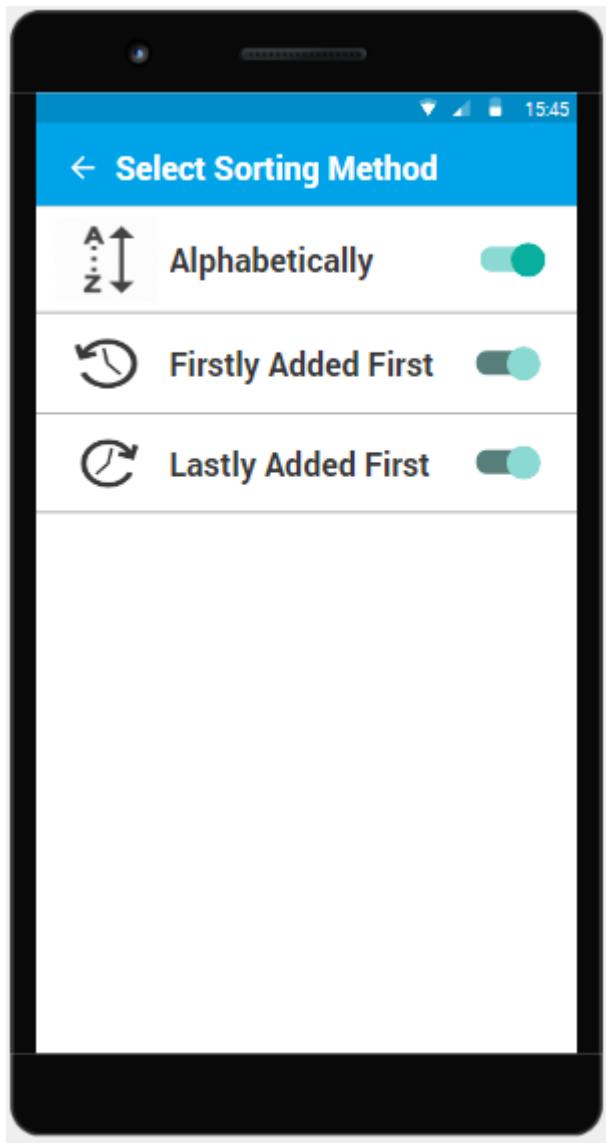




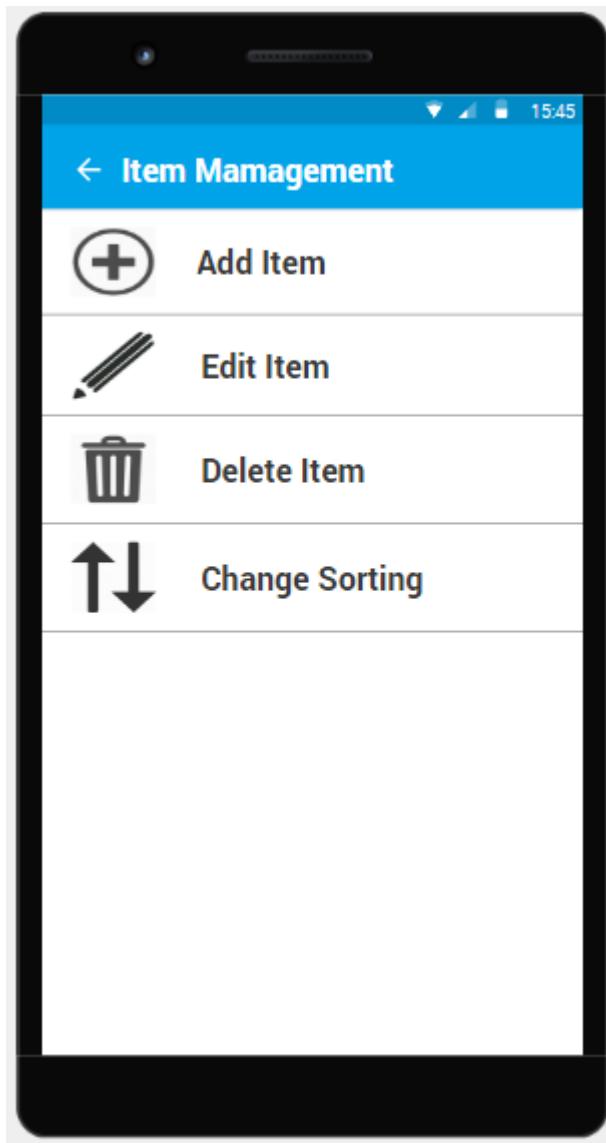
Setting > Category Management > Delete Category Screen



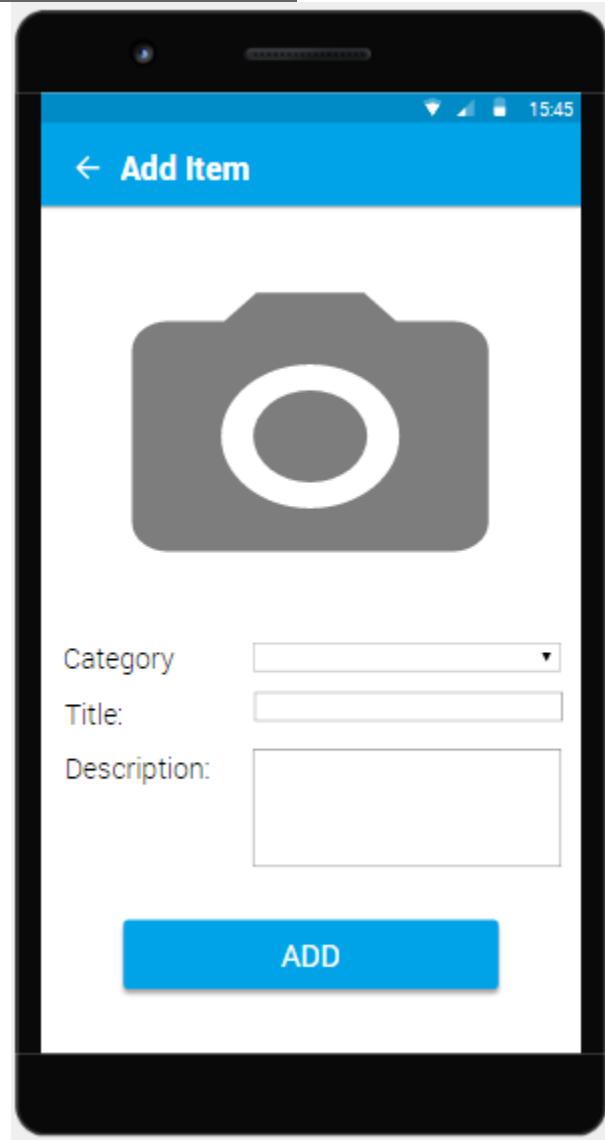
Setting > Category Management > Change Sorting Screen

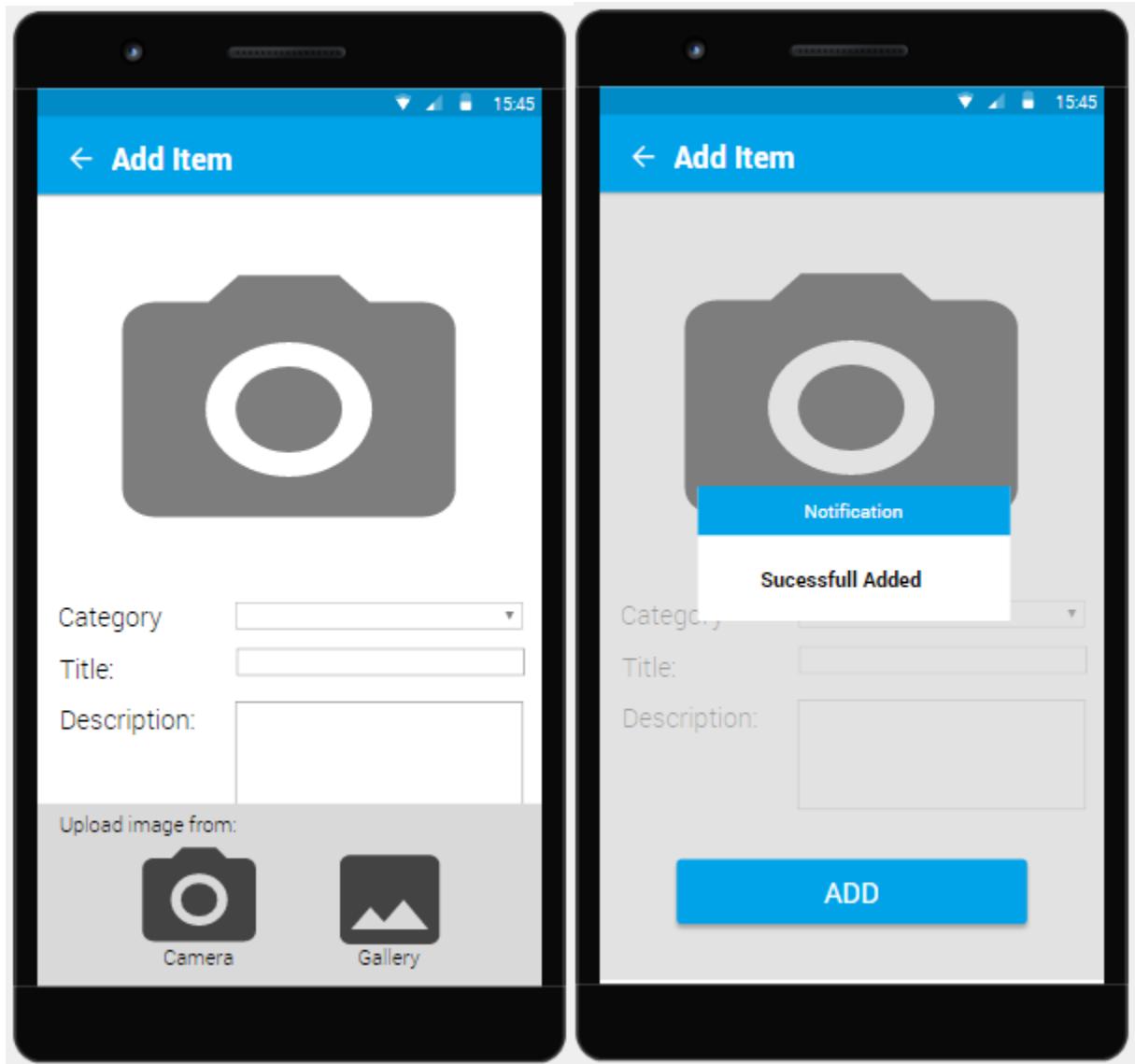


Setting >Item Management



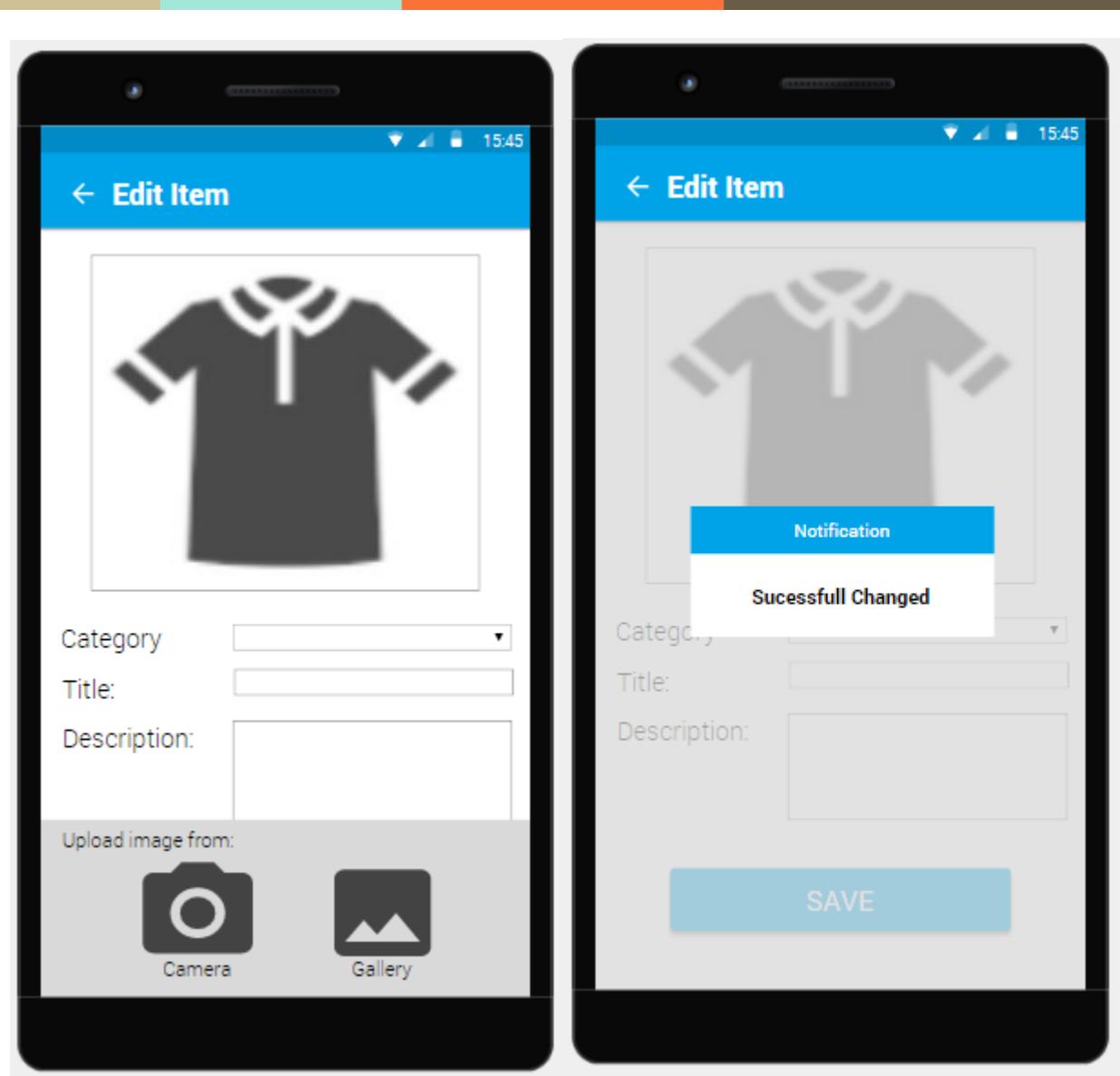
Setting >Item Management > Add Item Screen



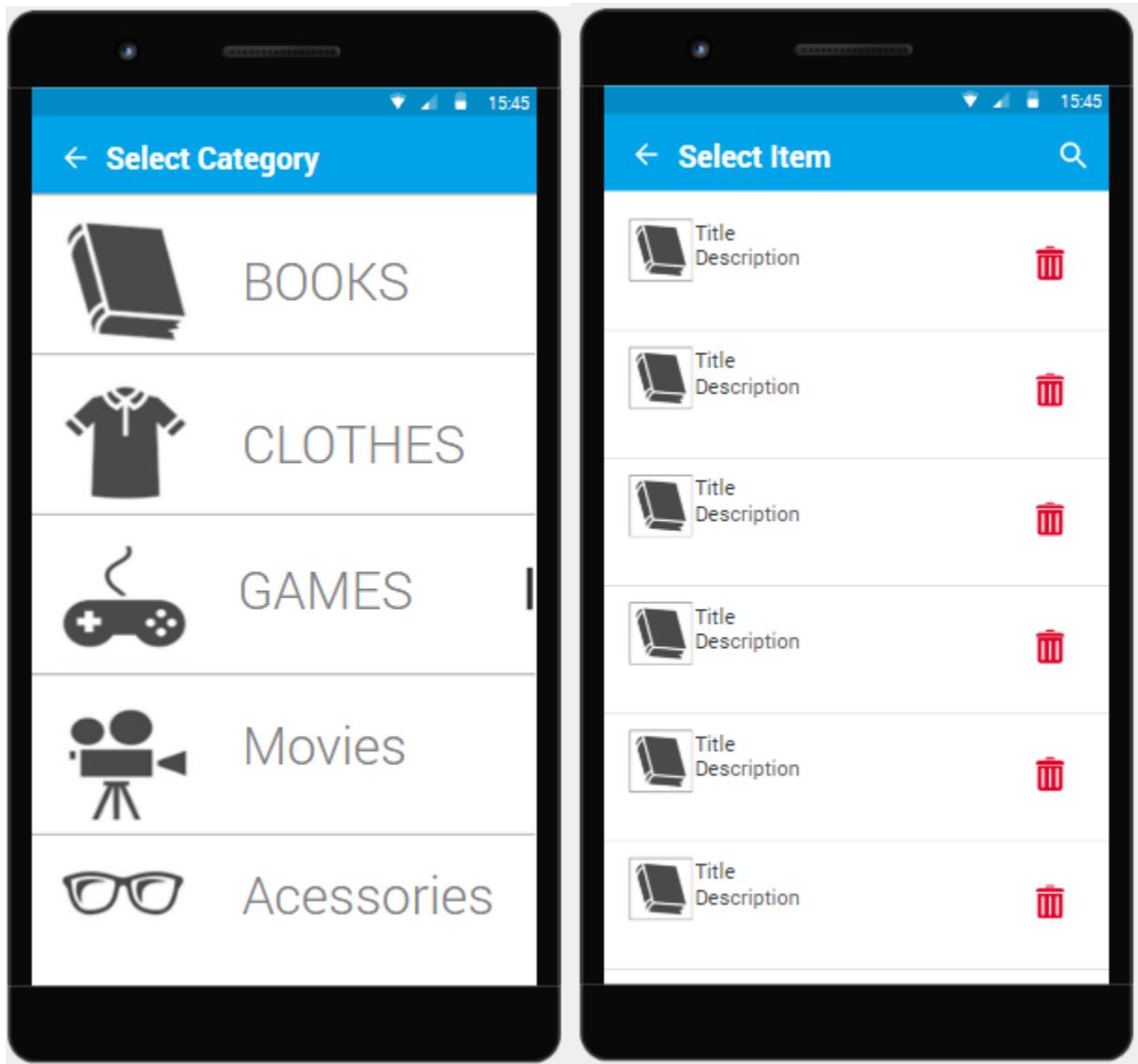


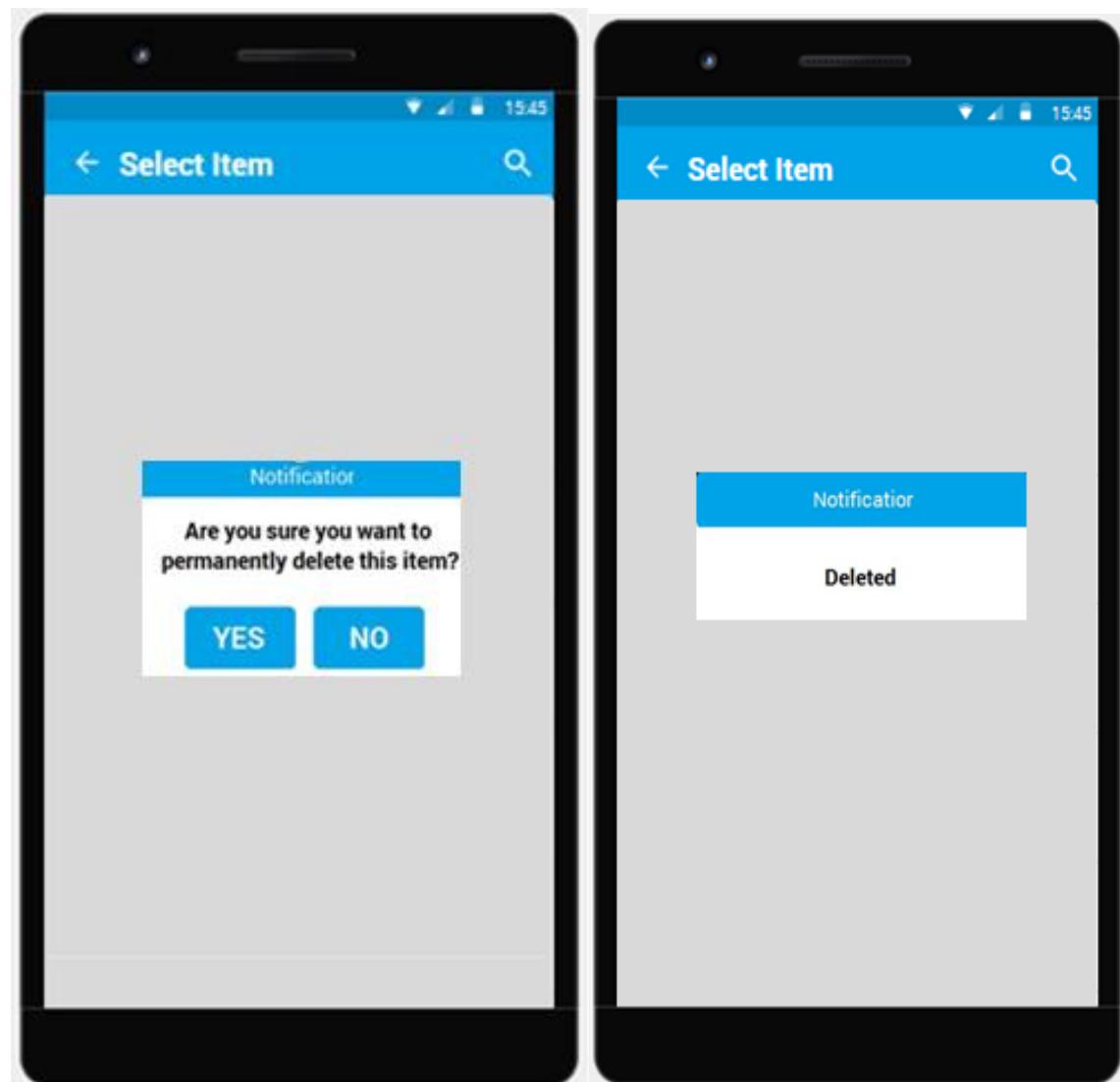
Setting >Item Management > Edit Item Screen



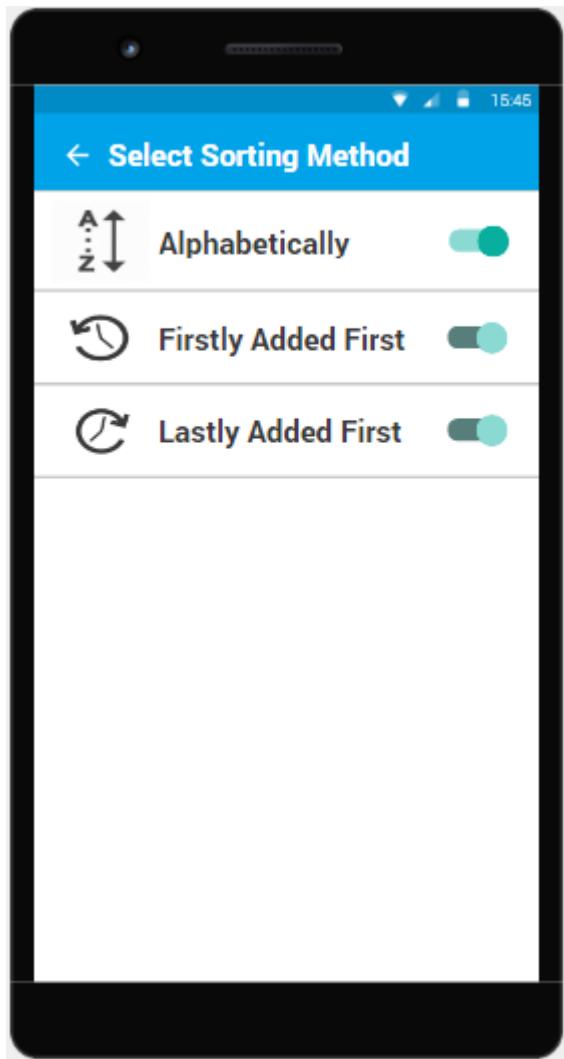


Setting >Item Management > Delete Item Screen

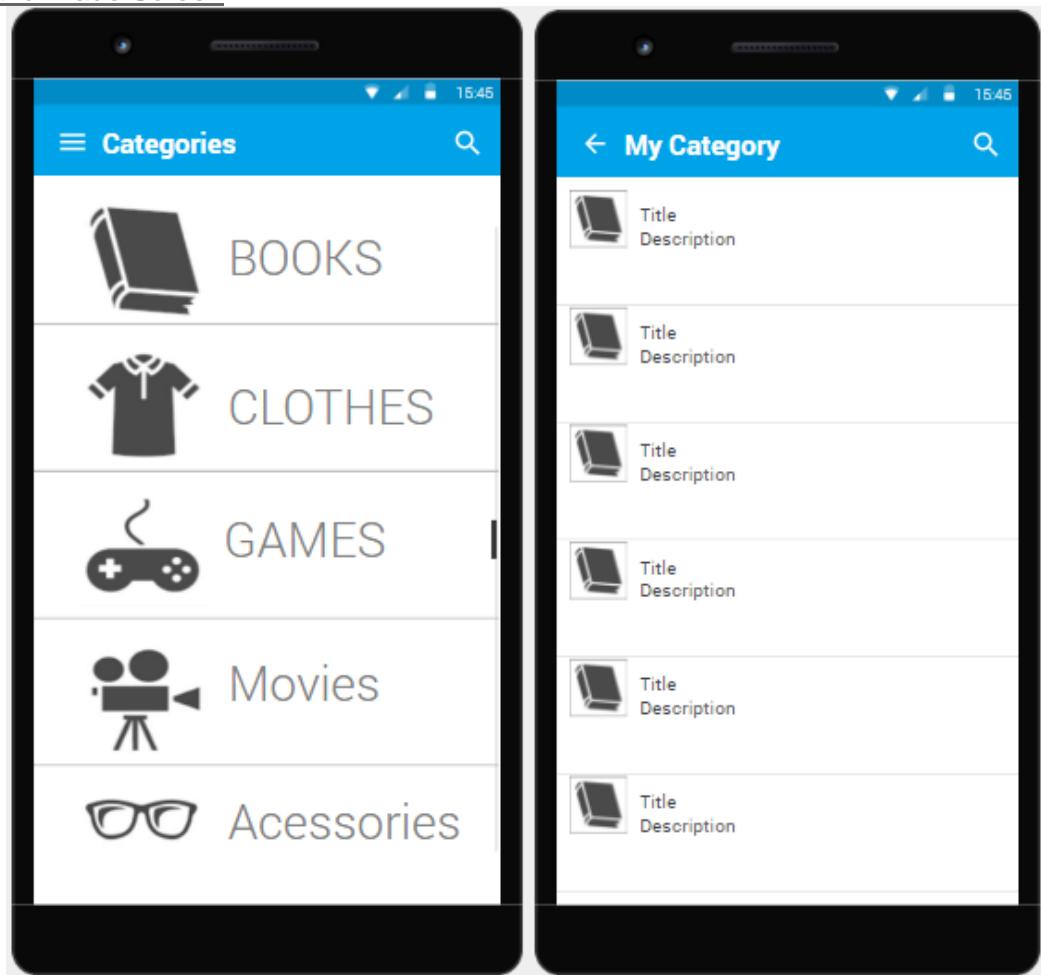


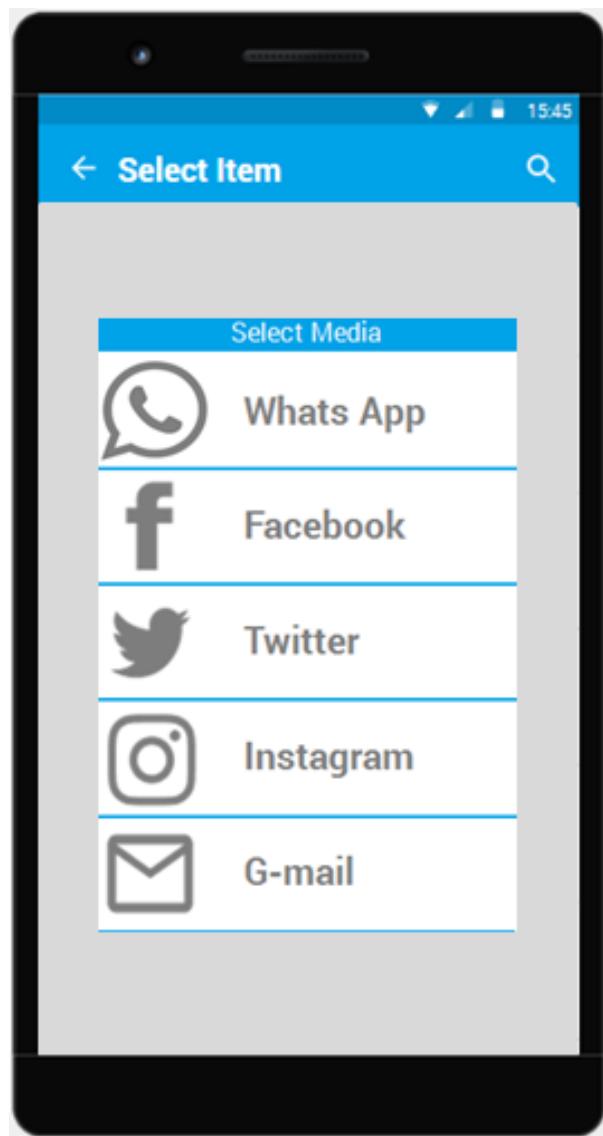


Setting >Item Management > Change Sorting Screen

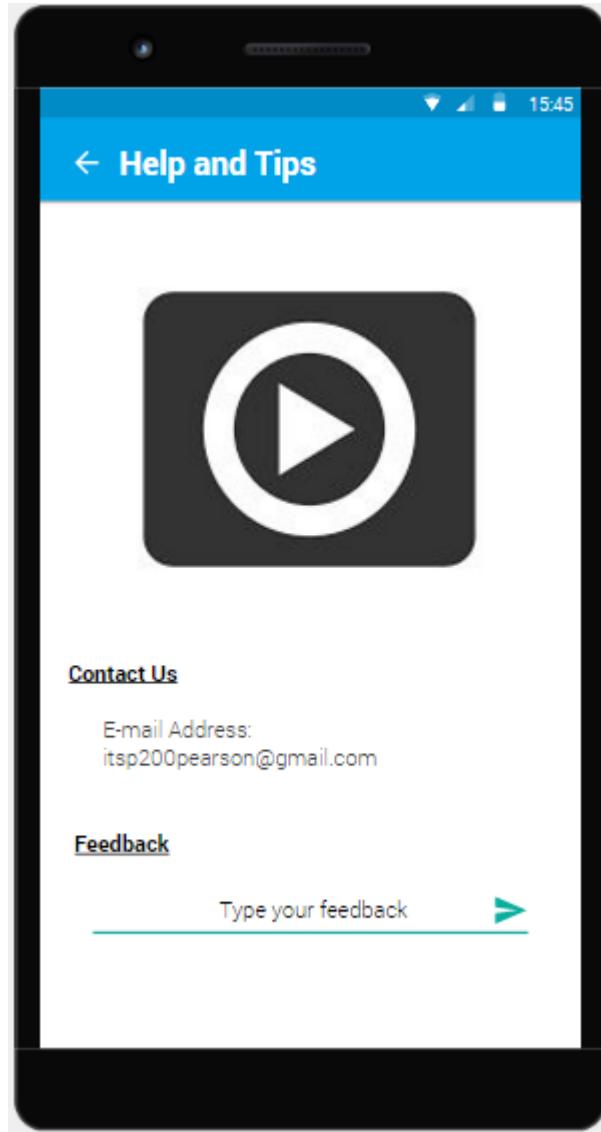


Share and Trade Screen

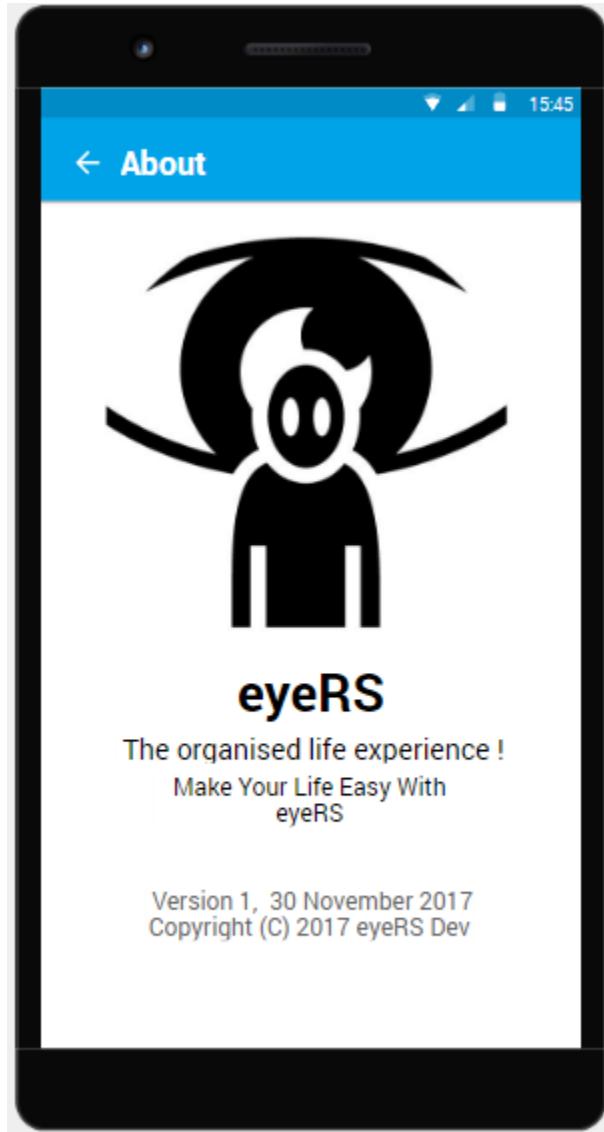




Help and Tips Screen



About Screen





2. Customer sign-off

Customer name and surname	Customer signature
Group leader name and surname	Group leader signature

Appendices

Appendix A

Nielsen's usability heuristics are a set of high-level design principles that are used to evaluate a user-interface to determine whether the interfaces conforms to well established and tested design principles (Preece et al. 2015). A revised version of Nielsen's heuristics are listed below:

- 11) The system should always provide the user with feedback;
- 12) The systems should match the user perception of the real world;
- 13) The system should provide the user with a clear exit to an unwanted state;
- 14) The systems should maintain the same standard and a high a level of consistency throughout;
- 15) The system should seek to eliminate error-prone conditions;
- 16) The system should minimise the load on the user's memory by making options and potential actions more visible;
- 17) The system must be efficient to use;
- 18) The system should make use of a minimalist aesthetic;
- 19) The system should aid users to recognise potential errors and assist the user in accounting for them; and
- 20) The system should provide the user with help documentation (Preece et al. 2015).

These heuristics are intended to be used by designers to compare or evaluate their designs and change their design accordingly. Each iteration of the design should seek to use these guidelines to solve usability problems (Stair & Reynolds 2015).

Bibliography

- Android Developers, Create and Manage Virtual Devices | Android Studio. Available at:
<https://developer.android.com/studio/run/managing-avds.html>
[Accessed August 3, 2017].
- Android Developers, Download Android Studio and SDK Tools | Android Studio. Available at:
https://developer.android.com/studio/index.html?gclid=CjwKEAjw9_jJBRCXycSarr3csWcSJABthk071PgsXQmXkW2jaenCzmxk5faXdS3nKbo99zzQizAWmhoC6N_w_wcB
[Accessed June 14, 2017].
- IEEE, 1990. *IEEE Standard Glossary of Software Engineering Terminology*, Inst of Engineering, Electrical & Electronic Standards.
- Oracle, Java SE Development Kit 8 - Downloads. Available at:
<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
[Accessed August 3, 2017].
- Oracle, Java SE Runtime Environment 8 - Downloads. Available at:
<http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>
[Accessed August 3, 2017].
- Dennis, A., Wixom, B. H. & Tegarden, D., 2005. *Systems Analysis and Design with UML Version 2.0*. 2nd ed. New Jersey: John Wiley & Sons, Inc.
- Satzinger, J. W., Jackson, R. B. & Burd, S. D., 2015. *Systems Analysis and Design in a Changing World*. 7th (E-BOOK) ed. Massachusetts: Cengage Learning.
- Shelly, G. B., Cashman, T. J. & Rosenblatt, H. J., 2003. *Systems Analysis and Design*. 5th ed. Boston: Thomson Course Technology.
- Bertolino, A., 2001. Chapter 5: Software Testing. *University of Houston-Clear Lake*. Available at: http://sce.uhcl.edu/helm/SWEBOK_IEEE/data/swebok_chapter_05.pdf
[Accessed August 7, 2017].
- Connolly, T.M. & Begg, C.E., 2005. *Database Systems: A Practical Approach to Design, Implementation, and Management*, Pearson Education.
- Finley, K., 2012. What Exactly Is GitHub Anyway? *TechCrunch*. Available at: <http://social.techcrunch.com/2012/07/14/what-exactly-is-github-anyway/>
[Accessed June 14, 2017].
- Justinmind, Prototyping tool for web and mobile apps - Justinmind. Available at: <https://www.justinmind.com>
[Accessed June 14, 2017].
- Mahfuj, M.M., 2012. Logical design vs physical design. Available at: <https://www.slideshare.net/MahediMahfujAnik/logical-design-vs-physical-design>
[Accessed May 17, 2017].

Preece, J., Rogers, Y. & Sharp, H., 2015. *Interaction Design: Beyond Human-Computer Interaction*, John Wiley & Sons.

softwaretestingclass, 2012. System Testing: What? Why? & How? *Software Testing Class*. Available at: <http://www.softwaretestingclass.com/system-testing-what-why-how/> [Accessed June 15, 2017].

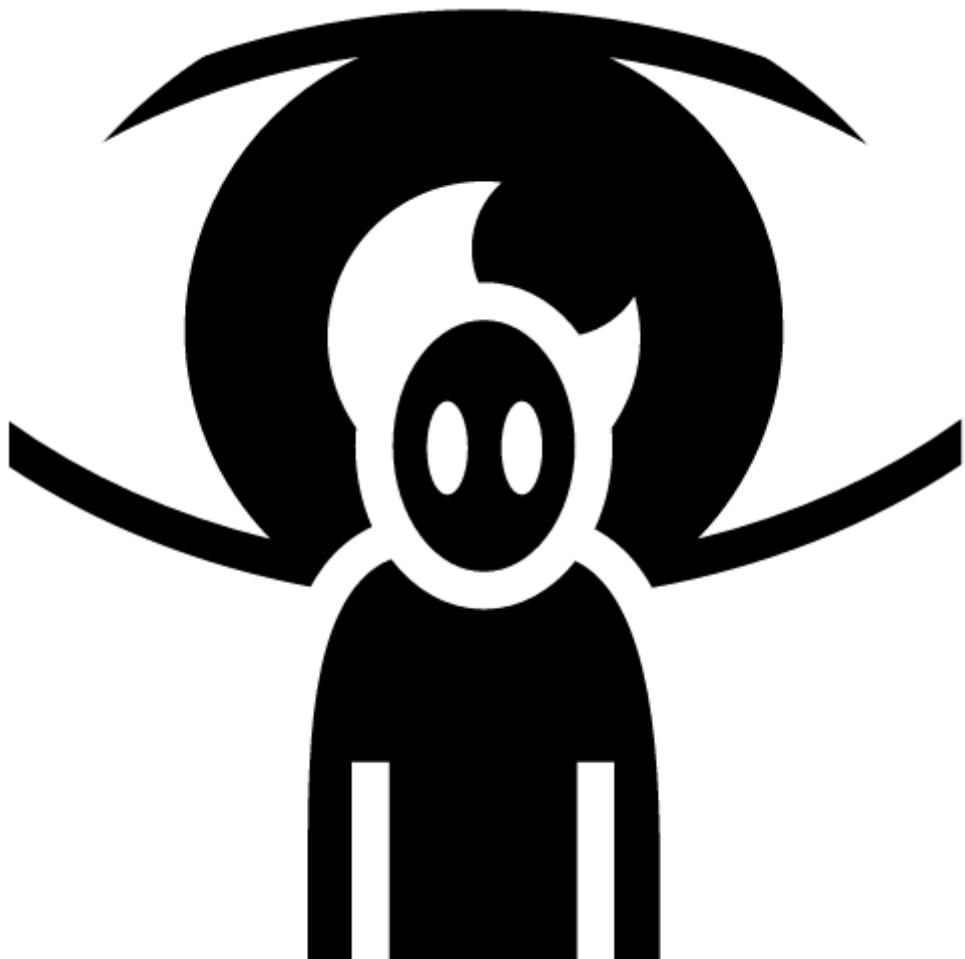
Stair, R. & Reynolds, G., 2015. *Principles of Information Systems*, Cengage Learning.

Williams, L., 2006. Testing Overview and Black-Box Testing Techniques. *North Carolina State University*. Available at: <http://agile.csc.ncsu.edu/SEMaterials/BlackBox.pdf> [Accessed August 7, 2017].

WebAIM, 2016. Microsoft Word: Creating Accessible Documents
<http://webaim.org/techniques/word/>
[Accessed August 9, 2017].

Information System Testing

eyeRS Development Team - ITSP200 (Deliverable 4)





Introduction

This deliverable presents the test results of the eyeRS application that were documented after testing was done with the user. The test were performed by the team members namely Nathan Shava, Matthew Van der Bijl, Emilde Arsenio, Andrea Cloete, Sajjaad Ishmail and the customer of the eyeRS application namely Ndai Mapaso

1. System test case results

Test Case Number: 1 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Open the application.	Test Case Name: Open the app. Subsystem: Stating. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the eyeRS icon	A Welcome message appear with the login screen.	A Welcome message appear with the login screen.	P	Successful.

Test Case Number: 2 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Test the registration for eyeRS.	Test Case Name: Register new user. Subsystem: Register. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click "Register" button	The system displays a screen with registration details	The system displays a screen with registration details	P	Successful.
2	Enter a username	The system displays the username in the text area	The system displays the username in the text area	P	Successful.
3	Enter an email address	The system displays the email address in the text area	The system displays the email address in the text area	P	Successful.
4	Enter a pin "*****"	The system asks the user to verify the pin	The system asks the user to verify the pin	P	Successful.
5	Re-enter the pin "*****"	The system verifies that the pins match	The system verifies that the pins match	P	Successful.
6	Select a security question	The System opens a drop-down list	The System opens a drop-down list	P	Successful.
7	Enter security response (Answer to	The system accepts the input	The system accepts the input	P	Successful.



	security question).				
8	Click the “Register” button	<i>The system registers the user with the database, a Registered notification appears and returns to the “Login” screen</i>	<i>App stopped working. User was prompted to restart the app</i>	F	<i>Failed. App stopped working.</i>

Test Case Number: 3 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Test the clear button on the register screen.	Test Case Name: Clear register new user information. Subsystem: Register. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the “Register” button	The system displays a screen with registration details	The system displays a screen with registration details	P	Successful.
2	Enter the registration information	Information display on the fields.	Information display on the fields.	P	Successful.
3	Click the clear button	All the information in the fields will be erased. Same screen are displayed.	All the information in the fields will be erased. Same screen are displayed.	P	Successful.
4	Click the back button.	Login screen is displayed.	Login screen is displayed.	P	Successful.

Test Case Number: 4 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user the opportunity to retrieve a forgotten pin.	Test Case Name: Forgot Pin. Subsystem: Login. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the red “Forgot Pin” text	New screen appears that display security question and text field for the security question answer.	New screen appears that display security question and text field for the security question answer.	P	<i>Successful</i>
2	Enter username	Information displays on the textfield.	Information displays on the textfield.	P	<i>Successful</i>
3	Create an new pin in the pin text field	Pin would appear as stars	Pin would appear as stars	P	<i>Successful</i>
4	Verify the pin by retying the pin in the text field	Pin would appear as stars	Pin would appear as stars	P	<i>Successful</i>
5	Select the drop down box	A list of possible security questions will appear	A list of possible security questions will appear	P	<i>Successful</i>
6	Enter security response	Text field is updated.	Text field is updated.	P	<i>Successful</i>
8	Click the “Reset” button	New pin as well as new security question and security answer is created and saved in the database. Rese successful notification appear. User is returned to the Login screen	Unable to add item message appeared and user were taken to the main screen of the app.	F	<i>Unable to add item message appeared and user were taken to the main screen of the app.</i>

Test Case Number: 5 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Test the clear button on the forgot pin screen.	Test Case Name: Clear reset pin information. Subsystem: Register. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the "Forgot Pin" text	The system displays a screen with the password reset details	The system displays a screen with the password reset details	P	Successful.
2	Enter the password reset information	Information display on the fields.	Information display on the fields.	P	Successful.
3	Click the clear button	All the information in the fields will be erased. Same screen are displayed.	All the information in the fields will be erased. Same screen are displayed.	P	Successful.
4	Click the back button.	Login screen is displayed.	Login screen is displayed	P	Successful.



Test Case Number: 6 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Log in to use the eyeRS application.	Test Case Name: Login. Subsystem: Main Screen. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the “Please enter your PIN” textfield	Keyboard appears.	<i>Keyboard appears.</i>	P	<i>Successful.</i>
2	Enter pin	Pin appear in text field	<i>Pin appear in text field</i>	P	<i>Successful.</i>
3	Click the “Login” button	If correct the system will open the main menu screen. If not the system will notify the user of the incorrect password, and the pin can be re-entered.	<i>Pin was entered correctly thus the user were taken to the main screen.</i>	P	<i>Successful.</i>

Test Case Number: 7 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Displays side menu.	Test Case Name: View Side Menu. Subsystem: My Category. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the “More” icon in the main menu	A side menu with more options will appear.	<i>A side menu with more options will appear.</i>	P	<i>Successful.</i>

Test Case Number: 8 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Adds an item to the users database via the camera upload.	Test Case Name: Add Item. Subsystem: My Category. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on “Add new item” in the side menu	An Add Item screen will appear.	<i>An Add Item screen will appear.</i>	P	Successful.
2	Click on the “Image” .	Notification appears asking for an upload of the image via the camera or the gallery.	<i>Notification appears asking for an upload of the image via the camera or the gallery.</i>	P	Successful.
3	Click on Cancel	No photo were added.	<i>No photo were added.</i>	P	Successful.
4	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.	<i>Notification appears asking for an upload of the image via the camera or the gallery.</i>	P	Successful.
5	Click on Take Photo	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).	<i>Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app.</i>	P	Successful.
6	Select the category drop box	A list of the categories fill appears.	<i>A list of the categories fill appears.</i>	P	Successful.
7	Select the appropriate category from the drop box list	The selected category will appear. (Note: Not yet saved).	<i>The selected category will appear. (Note: Not yet saved).</i>	P	Successful.
8	Click in the Name of item text field	Keyboard appears.	<i>Keyboard appears.</i>	P	Successful.
9	Enter the title of the item in the item text field	Title will appear in the text field. (Note: Not yet saved).	<i>Title will appear in the text field. (Note: Not yet saved).</i>	P	Successful.

10	Click in the Item description text field	Keyboard appears.	<i>Keyboard appears.</i>	P	Successful.
11	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).	<i>Description will appear in the description text field.</i>	P	Successful.
12	Click the “Add” button.	Item is added in the database. Notification of saved appears. User remains in the add item screen	<i>A notification box appeared to prompt the user to choose a method to add a photo.</i>	F	<i>A notification box appeared to prompt the user to choose a method to add a photo.</i>
13	Click on image, select choose from from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).	<i>Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).</i>	P	Successful.
14	Enter the new item detail as before.	Item detail displays.	<i>Item detail displays.</i>	P	Successful.
15	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add item screen.	<i>A notification box appeared to prompt the user to choose a method to add a photo</i>	F	<i>A notification box appeared to prompt the user to choose a method to add a photo</i>
16	Click on the back button.	Main menu appears.	<i>The main menu appears.</i>	P	Successful.



Test Case Number: 9 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Adds a category to the user's database via the camera upload.	Test Case Name: Add Category. Subsystem: Main menu. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.	<i>Side Menu appears.</i>	P	<i>Successful.</i>
2	Click on "Add new category" in the side menu	An Add Category screen will appear.	<i>An Add Category screen will appear.</i>	P	<i>Successful.</i>
3	Click on the image	Notification appears asking for an upload of the image via the camera or the gallery.	<i>Nothing happened</i>	F	<i>Not yet functional.</i>
4	Click on Cancel	No photo were added.	<i>Not able to click on cancel, there is no such option.</i>	F	<i>Not yet functional.</i>
5	Upload Image via camera	Camera opens-follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).	<i>Not able to make the selecting. There is no such selection option.</i>	F	<i>Not yet functional.</i>
6	Click in the Category Name text field	Keyboard appears.	<i>Keyboard appears.</i>	P	<i>Successful.</i>
7	Enter the Name of the category in the category text field	Title will appear in the text field. (Note: Not yet saved).	<i>Title will appear in the text field. (Note: Not yet saved).</i>	P	<i>Successful.</i>
8	Click in the Description text field	Keyboard appears.	<i>Keyboard appears.</i>	P	<i>Successful.</i>
9	Enter the description of the category in the description text field	Description will appear in the description text field. (Note Change is not yet saved).	<i>Description will appear in the description text field. (Note Change is not yet saved).</i>	P	<i>Successful.</i>
10	Click the "Add" button.	Item is added in the database.	<i>Item is added in the database.</i>	P	<i>Successful.</i>



		Notification of saved action. User remains in add the Add category screen.	<i>Notification of saved action. User remains in add the Add category screen.</i>		
11	Click on the image	Notification appears asking for an upload of the image via the camera or the gallery.	<i>Nothing happened</i>	F	<i>Not yet functional</i>
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).	<i>No such option to choose.</i>		<i>Not yet functional.</i>
13	Enter the new category detail displays before.	Category detail displays.	<i>Category detail displays.</i>	P	<i>Successful.</i>
14	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.	<i>Item is added in the database. Notification of saved action. User remains in add the Add category screen.</i>	P	<i>Successful.</i>
15	Click on the back button.	Main menu appears.	<i>Main menu appears.</i>	P	<i>Successful.</i>



Test Case Number: 10 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: View images from the user catalogue.	Test Case Name: Slideshow. Subsystem: Main menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.	<i>Side Menu appears.</i>	P	Successful.
2	Click on "Slideshow" in the side menu	An slideshow screen will appear and the images in the catalogue will appear in a slide show	<i>An slideshow screen will appear and the images in the catalogue will appear in a slide show</i>	P	Successful.
3	Click on the back button.	Main menu appears.	<i>Main menu appears.</i>	P	Successful.



Test Case Number: 11 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Shares a standard or custom user category and all content with other user's	Test Case Name: Share from side menu. Subsystem: Main menu. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the more icon on the main menu.	The side menu will appear.	<i>Side menu appears.</i>	P	<i>Successful</i>
2	Click on "Share" in the side menu	A select category screen will appear.	<i>Main menu appears.</i>	F	<i>Busy working on it.</i>
3	Click on appropriate category.	A Select Item screen will appear.	<i>Not functional</i>	F	<i>Busy working on it.</i>
4	Select an item to share.	A pop up will appear that shows all the supported sharing methods.	<i>Not functional</i>	F	<i>Busy working on it.</i>
5	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.	<i>Not functional</i>	F	<i>Busy working on it.</i>
6	Click the back button to return from the sharing app.	User is returned to the select item screen.	<i>Not functional</i>	F	<i>Busy working on it.</i>

* Note: Repeat for each sharing method.

Test Case Number: 12 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Searches for items by (name, category, details).	Test Case Name: Search Item to share via the side menu. Subsystem: Share. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon in the select item screen	Search text field will slide from the side and keyboard will appear.	<i>Slide from the side and keyboard appears.</i>	P	Successful
2	Search any item	Keyboard will disappear and search results will appear.	<i>Not functional</i>	F	<i>Busy working on it.</i>
3	Select the searched item	A pop up will appear that shows all the supported sharing methods.	<i>Not functional</i>	F	<i>Busy working on it.</i>
4	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.	<i>Not functional</i>	F	<i>Busy working on it.</i>
5	Click the back button to return from the sharing app.	User is returned to the select item screen.	<i>Not functional</i>	F	<i>Busy working on it.</i>

*Note: Repeat for each sharing method.



Test Case Number: 13 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Exits share screen.	Test Case Name: Exit Share. Subsystem: Share. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button.	The select category screen will appear.	<i>Not functional</i>	F	<i>Busy working on it.</i>
2	Click the back button.	Main menu will appear.	<i>Not functional</i>	F	<i>Busy working on it.</i>

Test Case Number: 14 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Trade category and all content with other user's.	Test Case Name: Trade via the side menu. Subsystem: Main menu. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the more icon on the main menu	Side menu will appear.	<i>Side menu appears</i>	P	<i>Successful</i>
2	Click on "Trade" in the side menu	A select category screen will appear.	<i>Main menu appears</i>	F	<i>Busy working on it.</i>
3	Click on appropriate category.	A Select Item screen will appear.	<i>Not functional</i>	F	<i>Busy working on it.</i>
4	Select an item to share.	A pop up will appear that shows all the supported sharing methods.	<i>Not functional</i>	F	<i>Busy working on it.</i>
5	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.	<i>Not functional</i>	F	<i>Busy working on it.</i>
6	Click the back button to return from the sharing app.	User is returned to the select item screen.	<i>Not functional</i>	F	<i>Busy working on it.</i>

Test Case Number: 15 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Searches for items by (name, category, details).	Test Case Name: Search Item to Trade via the side menu. Subsystem: Trade. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon in the select item screen	Search text field will slide from the side and keyboard will appear.	<i>Text field appears and keyboard appears.</i>	P	Successful
2	Search any item	Keyboard will disappear and search results will appear.	<i>Main menu appears</i>	F	<i>Busy working on it.</i>
3	Select the searched item	A pop up will appear that shows all the supported sharing methods.	<i>Not functional</i>	F	<i>Busy working on it.</i>
4	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.	<i>Not functional</i>	F	<i>Busy working on it.</i>
5	Click the back button to return from the sharing app.	User is returned to the select item screen.	<i>Not function</i>	F	<i>Busy working on it.</i>



Test Case Number: 16 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Exits Trade screen.	Test Case Name: Exit Trade via side menu. Subsystem: Trade. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button	The select category screen will appear.	<i>Not functional</i>	F	<i>Busy working on it.</i>
2	Click the back button	Main menu will appear.	<i>Not functional</i>	F	<i>Busy working on it.</i>

Test Case Number: 17 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Accessing help & tips for the app via the side menu.	Test Case Name: Help and Tips via Side menu. Subsystem: Side menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on "Help and Tips" in the side menu	A Help and Tips screen will appear.	<i>Help and Tips screen appear.</i>	P	<i>Successful</i>
2	Click on the video.	A tutorial video will play.	<i>Not functional</i>	F	<i>Busy working on it.</i>
3	Click on the Send feedback button	User is taken to google play store to leave feedback at google play store.	<i>Options appears to where to send.</i>	F	<i>Busy working on it.</i>
4	Click the back button to leave play store.	The help and tip screen appear.	<i>Not functional</i>	F	<i>Busy working on it.</i>
5	Click the back button in the help and tips screen.	User is taken back to the main menu.	<i>Main menu appears.</i>		<i>Successful</i>



Test Case Number: 18 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Gives a brief history on the development team for eyeRS and the app via the side menu.	Test Case Name: About via the side menu. Subsystem: Side menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on “About” in the side menu.	A About screen will appear. Which contains the information about the app.	<i>About screen appears</i>	P	<i>Successful</i>
2	Click the back button.	The settings screen appears.	<i>Settings screen appears</i>	P	<i>Successful</i>

Test Case Number: 19 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Displaying of all possible sections for configuration of the app.	Test Case Name: Settings. Subsystem: Main menu. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.	<i>Side menu appears</i>	P	<i>Successful</i>
2	Click on “Settings” in the side menu	An settings screen will appear	<i>Settings screen appears</i>	P	<i>Successful</i>

Test Case Number: 20 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Security settings that are provided can now be configured.	Test Case Name: Security Settings. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Security	A security settings screen will appear.	<i>Security settings appears</i>	P	Successful
2	Click on the Enter your username text field	Keyboard appears	<i>Keyboard slides up</i>	P	Successful
3	Enter username	The username will appear in the text field. Note it is not yet saved.	<i>The username appears</i>	P	Successful
4	Click on the Enter your new PIN here text field	Keyboard appears	<i>Keyboard slides up</i>	P	Successful
5	Enter the new PIN	Pin display as dots in the text field. Not it is not yet saved.	<i>Pin display as dots</i>	P	Successful
6	Select drop down box	A list of all possible security questions will appear.	<i>List of security questions appears</i>	P	Successful
7	Select a security question from the list.	New security question will appear in the security question box.	<i>New security question appears</i>	P	Successful
8	Click on the Security Response text field.	A keyboard appears.	<i>Keyboard slides up</i>	P	Successful
9	Enter the security response in the text field	The security response will appear in the text field. Note: Data is not yet saved.	<i>The security response appears</i>	P	Successful
10	Click on the Clear button.	All the entered information are erased. PIN reset screen remain on the screen.	<i>All information is erased from fields</i>	P	Successful
11	Enter all the reset PIN information.	All text will be displayed.	<i>All text is displayed</i>	P	Successful



12	Click the reset button	Successful notification will appear that. User will be taken back to the settings screen.	<i>Could not save information. Back to the main menu.</i>	F	Busy working on it.
----	------------------------	---	---	---	---------------------

Test Case Number: 21 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Displays settings that can be configured.	Test Case Name: Display Settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on display	A display setting screen will appear.	<i>Display setting appears.</i>	P	Successful
2	Click on the new colour block *To be repeated for each colour available	The colour of the app will change.	<i>The colour remains the same.</i>	F	Unsuccessful
3	Click on the select Font drop down box.	A list of all the available font will appear.	<i>A list of all available fonts does not appear.</i>	F	Unsuccessful
4	Select a font in the list	The font of the app will change.	<i>No response.</i>	F	Unsuccessful
5	Click on the select font size drop down box.	A list of all possible font sizes will appear.	<i>A list of all possible font sizes does not appear.</i>	F	Unsuccessful
6	Click on the back button.	The Setting screen will appear.	<i>The settings screens appears.</i>	P	Successful



Test Case Number: 22 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: User profile can be configured. Upload a profile picture via the camera.	Test Case Name: Profile Settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Profile	A profile setting screen will appear.	<i>The profile screen appears.</i>	P	<i>Successful</i>
2	Click on the profile picture.	Notification appear asking to choose upload media source selection.	<i>No notification appears.</i>	F	<i>Unsuccessful</i>
3	Click on cancel	The request is canceled.	<i>N/A</i>	F	<i>Unsuccessful</i>
4	Click on camera	Camera opens-follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).	<i>Camera does not open.</i>	F	<i>Unsuccessful</i>
5	Click on the Please enter your display username text field.	A keyboard appears.	<i>A keyboard appears.</i>	P	<i>Successful</i>
6	Enter username in the username text field	New user name will appear Note: Data is not yet saved.	<i>A typed username appears on screen.</i>	P	<i>Successful</i>
7	Click on the save changes button	Successful notification appears. Profile setting screen still appears.	<i>Notification does not appear.</i>	F	<i>Unsuccessful</i>
8	Click the back button.	User is returned to the settings option screen.	<i>User is returned to settings screen</i>	P	<i>Unsuccessful</i>

Test Case Number: 24 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: The sound for the app can be configured.	Test Case Name: Sound Settings. Subsystem: Settings. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on sound	A sound setting screen appears	Sound setting screen appears.	P	Successful
2	Click on the welcome message toggle	The toggle turns off and welcome message will not play next time the user access the app.	"Welcome message disabled!" appears on screen.	P	<i>A message is displayed after changing the toggle.</i>
3	Click on the touch sound toggle	The toggle turns off and the touch sound will not play.	"Touch sounds disabled!" appears on screen	P	<i>A message is displayed after changing the toggle.</i>
4	Click on the back button	The Setting Option screen will appear.	The Setting Option screen will appears.	P	Successful

Test Case Number: 25 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: The management of the categories in the app.	Test Case Name: Category Management. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Category Management settings	A category management screen will appear	The category management screen appears.	P	Successful



Test Case Number: 26 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Adding a new category while in settings.	Test Case Name: Add Category via Settings. Subsystem: Category Management. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon	Side Menu appears.	N/A	F	<i>There is no side menu.</i>
2	Click on "Add new category" in the side menu	An Add Category screen will appear.	<i>The add category screen appears.</i>	P	Successful
3	Click on the "Camera" icon	Notification appears asking for an upload of the image via the camera or the gallery.	<i>No notification appears.</i>	F	Unsuccessful
4	Click on Cancel	No photo were added.	N/a	F	Unsuccessful
5	Upload Image via camera	Camera opens-follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).	<i>Camera does not open.</i>	F	Unsuccessful
6	Click in the Category Name text field	Keyboard appears.	<i>A keyboard appears.</i>	P	Successful
7	Enter the Name of the category in the category text field	Title will appear in the text field. (Note: Not yet saved).	<i>The title appears in the text field.</i>	P	Successful
8	Click in the Description text field	Keyboard appears.	<i>A keyboard appears.</i>	P	Successful
9	Enter the description of the category in the description text field	Description will appear in the description text field. (Note Change is not yet saved).	<i>The description appears in the text field.</i>	P	Successful

10	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.	<i>Your new category has been created successfully appears on screen.</i>
11	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.	<i>No notification appears.</i>
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).	N/A
13	Enter the new category detail as before.	Category details display.	<i>Details are displayed.</i>
14	Click the “Add” button.	Item is added in the database. Notification of saved action. User remains in add the Add category screen.	<i>“Your new category has been created successfully appears on screen.</i>
15	Click on the back button.	Category management option screen appears.	<i>Category management screen appears.</i>



Test Case Number: 27 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Editing a category.	Test Case Name Edit category. Subsystem: Category Management. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click Edit category	List with current categories appear.	No response.	F	<i>Not yet working.</i>
2	Click on the category you wish to edit	Category information will appear.	N/A	F	<i>Unsuccessful</i>
3	Click on the category icon	Notification appears asking for an upload of the image via the camera or the gallery.	N/A	F	<i>Unsuccessful</i>
4	Click on Cancel	No photo were added.	N/A	F	<i>Unsuccessful</i>
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).	N/A	F	<i>Unsuccessful</i>
6	Click in the Category Name text field	Keyboard appears.	N/A	F	<i>Unsuccessful</i>
7	Enter the Name of the category in the category text field	Title will appear in the text field. (Note: Not yet saved).	N/A	F	<i>Unsuccessful</i>
8	Click in the Description text field	Keyboard appears.	N/A	F	<i>Unsuccessful</i>
9	Enter the description of the category in the description text field	Description will appear in the description text field. (Note Change is not yet saved).	N/A	F	<i>Unsuccessful</i>
10	Click the “Save changes” button.	Category is edited in the database. Notification of saved action. User remains	N/A	F	<i>Unsuccessful</i>



		in add the Add category screen.			
11	Click on the category icon.	Notification appears asking for an upload of the image via the camera or the gallery.	N/A	F	<i>Unsuccessful</i>
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).	N/A	F	<i>Unsuccessful</i>
13	Enter the new category detail are before.	Category detail displays.	N/A	F	<i>Unsuccessful</i>
14	Click the “Save changes” button.	Category is edited in the database. Notification of saved action. User remains in add the Add category screen.	N/A	F	<i>Unsuccessful</i>
15	Click on the back button.	Category management option screen appears.	N/A	F	<i>Unsuccessful</i>



Test Case Number: 28 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Removing a category from the catalogue.	Test Case Name: Delete category. Subsystem: Category Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Delete Category	A category selection screen appears.	No response.	F	<i>Not yet working.</i>
2	Select a category to delete.	Confirmation notification appears.	N/A	F	<i>Unsuccessful</i>
3	Click No on the notification	Category selection screen appears.	N/A	F	<i>Unsuccessful</i>
4	Select a category to delete	Confirmation notification appears.	N/A	F	<i>Unsuccessful</i>
5	Select Yes on the notification	Category is deleted in the database. Successful notification appears.	N/A	F	<i>Unsuccessful</i>
6	Click the back button	The category management option screen appears.	N/A	F	<i>Unsuccessful</i>



Test Case Number: 29 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows user to sort the categories in multiple ways.	Test Case Name: Change Category Sorting. Subsystem: Category Management. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Change sorting	A sorting method screen appears.	<i>Sorting method screen appears.</i>	P	Successful
2	Click on the alphabetically toggle	Toggle turns off/on. Categories will be listed alphabetically. Can be viewed in the main menu.	<i>"All categories have been sorted automatically / All categories will be displayed as normal" appears on screen.</i>	P	Successful
3	Click on the firstly added first toggle	Toggle turns on/off Categories is listed in a first added first view manner. Can be viewed in the main menu.	<i>"All categories have been sorted according to their order of entry / All categories will be displayed as normal" appears on screen.</i>	P	Successful
4	Click on the Lastly added first toggle	Toggle turns on/off Categories is listed in a last added view first manner. Can be viewed in the main menu.	N/A	F	<i>There is only recently added toggle.</i>
5	Click on the back button.	The category management screen appears.	<i>The category management screen appears.</i>	P	Successful



Test Case Number: 30 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to return settings screen.	Test Case Name: Exit Category Management settings. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button.	The setting screen appears.	<i>The setting screen appears.</i>	P	Successful

Test Case Number: 31 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to manage their items in the catalogue.	Test Case Name: Item Management. Subsystem: Settings. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Item management	The item management screen appears.	As expected	P	<i>The user could navigate to the screen as expected</i>

Test Case Number: 32 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to add a an item to the catalogue.	Test Case Name: Add item via Item Management. Subsystem: Item management. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on “Add new item”	An Add Item screen will appear.	As expected	P	<i>The user could navigate to the screen as expected</i>
2	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.	As expected	P	<i>Pop up displayed prompting user to select their preferred option</i>
3	Click on Cancel	No photo were added.	As expected	P	<i>Pop up closed</i>
4	Click on the “Camera” icon	Notification appears asking for an upload of the image via the camera or the gallery.	As expected	P	<i>Pop up displayed prompting user to select their preferred option</i>
5	Click on Take Photo	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).	As expected	P	<i>Use of built-in camera to capture image</i>
6	Select the category drop box	A list of the categories fill appears.	As expected	P	<i>Spinner containing categories where the item will be stored</i>
7	Select the appropriate category from the drop box list	The selected category will appear. (Note: Not yet saved).	As expected	P	<i>Selected category remains selected</i>

					<i>until it has been saved</i>
8	Click in the Name of item text field	Keyboard appears.	<i>As expected</i>	P	<i>In-built keyboard pops up automatically</i>
9	Enter the title of the item in the item text field	Title will appear in the text field. (Note: Not yet saved).	<i>As expected</i>	P	<i>The user was able to insert text using the in-built keyboard</i>
10	Click in the Item description text field	Keyboard appears.	<i>As expected</i>	P	<i>In-built keyboard pops up automatically</i>
11	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).	<i>As expected</i>	P	<i>The user was able to insert text using the in-built keyboard</i>
12	Click the “Add” button.	Item is added in the database. Notification of saved appears. User remains in the add item screen	<i>Unsuccessful</i>	F	<i>Pop up appears informing user item has been added however no reference could be located to the item</i>
13	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).	<i>As expected</i>	P	<i>In-built gallery appears prompting user to select image to upload</i>
14	Enter the new item detail as before.	Item detail displays.	<i>As expected</i>	P	<i>Text input successfully entered into text fields using in-built keyboard.</i>

15	Click the "Add" button.	Item is added in the database. Notification of saved action. User remains in add the Add item screen.	<i>Unsuccessful</i>	<i>F</i>	<i>Pop up appears informing user item has been added however no reference could be located to the item</i>
16	Click on the back button.	Item management options appears.	<i>As expected</i>	<i>P</i>	<i>The user was able to navigate back to Item Management screen</i>

Test Case Number: 33 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to edit an item in the catalogue.	Test Case Name: Edit item via Item Management. Subsystem: Item Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click Edit Item	List with current items appear.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
2	Click on the item you wish to edit	Item information will appear.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
4	Click on Cancel	No photo were added.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image	<i>Unsuccessful</i>	F	<i>Because items were not successfully</i>

		of the item in the eyeRS app. (Note: Not yet saved).			<i>added into the db this operation could not be tested</i>
6	Click in the Item Name text field	Keyboard appears.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
8	Click in the Description text field	Keyboard appears.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
11	Click on the item icon.	Notification appears asking for an upload	<i>Unsuccessful</i>	F	<i>Because items were</i>



		of the image via the camera or the gallery.			<i>not successfully added into the db this operation could not be tested</i>
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
13	Enter the new item detail are before.	Item detail displays.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
15	Click on the back button.	Item management option screen appears.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>



Test Case Number: 34 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to delete an item in the catalogue.	Test Case Name: Delete item via Item Management. Subsystem: Item Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Delete Item	A category selection screen appears	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
2	Click on the category that contain the item that will be deleted.	A item selection screen will appear.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
3	Click on the item you want to delete.	Delete notification appear for confirmation.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
4	Click on No	Item is not deleted. User remains in the item selection screen	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
5	Click on the item you want to delete.	Delete notification appear for confirmation.	<i>Unsuccessful</i>	F	<i>Because items were not successfully</i>



					<i>added into the db this operation could not be tested</i>
6	Click on Yes	Item will be deleted from the database and the item will no longer be visible in the item list. User remains in the select item screen.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>

Test Case Number: 35 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allows the user to delete an item in the catalogue.	Test Case Name: Delete item via Item Management with search as assistance. Subsystem: Item Management. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon	Keyboard will appear and a text field will open.	As expected	P	<i>In-built keyboard appears when user clicks on the search bar</i>
2	Search for an item	Search item will appear if it was found else it will display no result found	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
3	Click on the item you want to delete.	Delete notification appear for confirmation.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
4	Click on No	Item is not deleted. User remains in the item selection screen	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
5	Click on the item you want to delete.	Delete notification appear for confirmation.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation</i>



					<i>could not be tested</i>
6	Click on Yes	Item will be deleted from the database and the item will no longer be visible in the item list. User remains in the select item screen.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
7	Click on the back button.	The item management option screen appears.	<i>Unsuccessful</i>	F	<i>Because items were not successfully added into the db this operation could not be tested</i>



Test Case Number: 36 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Allowing user to sort an item in the catalogue.	Test Case Name: Sorting the items in the catalogue. Subsystem: Item Management. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on Change sorting	A sorting method screen appears.	As expected	P	<i>The user could navigate to the item sorting screen successfully</i>
2	Click on the alphabetically toggle	Toggle turns off/on. Items will be listed alphabetically.	Unsuccessful	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
3	Click on the recently added first toggle	Toggle turns on/off. Items is listed according to the most recently added item first	Unsuccessful	F	<i>Because items were not successfully added into the db this operation could not be tested</i>
4	Click on the back button.	The item management screen appears.	As expected	P	<i>The user was able to navigate back to Item Management screen</i>



Test Case Number: 37 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Returning to the main settings screen.	Test Case Name: Exiting Item Management. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the back button	The settings screen appears.	<i>As expected</i>	P	<i>The user was able to navigate back to Settings screen</i>

Test Case Number: 38 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Accessing help & tips for the app via settings.	Test Case Name: Help and Tips via Settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on "Help and Tips"	A Help and Tips screen will appear.	As expected	P	<i>The user could navigate to the Help & Tips screen successfully</i>
2	Click on the video.	A tutorial video will play.	Unsuccessful	F	<i>The video could not be run at the time due to its unavailability</i>
3	Click on the Send feedback button	User is taken to google play store to leave feedback at google play store.	As expected	P	<i>The user was redirected to the application's Google Play Store portal where feedbacks & app ratings could be made</i>
4	Click the back button to leave play store.	The help and tip screen appear.	As expected	P	<i>The user navigated back to the Help & Tips screen</i>
5	Click the back button in the help and tips screen.	User is taken back to the settings screen.	As expected	P	<i>The user navigated back to the Settings screen</i>

Test Case Number: 39 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Gives a brief history on the development team for eyeRS and the app via settings.	Test Case Name: About via settings. Subsystem: Settings. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on "About"	About screen will appear. Which contains the information about the app.	As expected	P	<i>The user is navigated to the About screen</i>
2	Click the back button.	The settings screen appears.	As expected	P	<i>The user is navigated back to the Settings screen</i>

Test Case Number: 40 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Returning to the main menu to proceed with other functions that the app provide.	Test Case Name: Exit the settings. Subsystem: Settings. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click the back button.	Main menu appears.	As expected	P	<i>The user is navigated back to the Main Home screen.</i>



Test Case Number: 41 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Searches for items in the catalog from the main screen.	Test Case Name: Search Item from the main screen. Subsystem: Main Menu. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the search icon	Search text field will slide from the side and keyboard will appear.	<i>As expected</i>	P	<i>Search bar and in-built keyboard appear enabling user to input text to search for an item.</i>
2	Search any item	Keyboard will disappear and search results will appear.	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>

Test Case Number: 42 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: View and edit items from the catalog directly from the search feature.	Test Case Name: View and Edit Item direct from the search feature. Subsystem: Item Detail. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the item you want to view or change.	Item detail screen display.	Unsuccessful	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
2	Click Edit Item button	List with current items appear.	Unsuccessful	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.	Unsuccessful	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
4	Click on Cancel	No photo were added.	Unsuccessful	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
5	Upload Image via camera	Camera opens-follow steps used to take a photo – photo appears as the image of the item in	Unsuccessful	F	<i>Because no items had been successfully added this</i>



		the eyeRS app. (Note: Not yet saved).			<i>operation could not be tested at the time.</i>
6	Click in the Item Name text field	Keyboard appears.	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
8	Click in the Description text field	Keyboard appears.	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
11	Click on the item icon.	Notification appears asking for an upload of the image via the	<i>Unsuccessful</i>	F	<i>Because no items had been</i>

		camera or the gallery.			<i>successfully added this operation could not be tested at the time.</i>
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. (Note: Not yet saved).	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
13	Enter the new item detail are before.	Item detail displays.	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>
15	Click on the back button.	Item detail screen appears.	<i>Unsuccessful</i>	F	<i>Because no items had been successfully added this operation could not be tested at the time.</i>



Test Case Number: 43 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item once in a searched item.	Test Case Name Share Item once in searched item. Subsystem: Item Detail. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.	<i>No were to click</i>	F	<i>Unsuccessful</i>
2	Click in the "Share" option in the pop up screen	Notification that asks you to choose a sharing method.	<i>No notification, since nothing could have been clicked</i>	F	<i>Unsuccessful</i>
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.	<i>No sharing method to select</i>	F	<i>Unsuccessful</i>
4	Click the back button to return from the sharing screen.	User is returned to the Item detail screen appears.	<i>Was never in the sharing screen</i>	F	<i>Unsuccessful</i>

*Note: Repeat for each sharing method.



Test Case Number: 44 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item once in a searched item.	Test Case Name Trade Item once in searched item. Subsystem: Item Detail. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.	<i>No such pace to click</i>	F	<i>Unsuccessful</i>
2	Click in the “Trade” option in the pop up screen	Notification that asks you to choose a sharing method.	<i>No were to click since no such pop up.</i>	F	
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.	<i>No method to select.</i>	F	<i>Unsuccessful</i>
4	Click the back button to return from the trading screen.	User is returned to the Item detail screen appears.	<i>Was never in the trading screen</i>	F	<i>Unsuccessful</i>

*Note: Repeat for each sharing method.



Test Case Number: 45 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category.	Test Case Name Delete Item once in searched item. Subsystem: Item Detail. Design Date: Execution Date:
---	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.	<i>No such place to click.</i>	F	<i>Unsuccessful</i>
2	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.	<i>No delete button</i>	F	<i>Unsuccessful</i>
3	Click the “No” button in the Delete conform notification	Delete confirm notification will disappear and the item detail screen will appear.	<i>No delete button no conformation.</i>	F	<i>Unsuccessful</i>
4	Click on the more icon.	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.	<i>No such pop up.</i>	F	<i>Unsuccessful</i>
5	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.	<i>No such option, since no pop up.</i>	F	<i>Unsuccessful</i>
6	Click the “Yes” button in the Delete conform notification	Deleted notification appears. Item is deleted from the database. Main menu will appear	<i>No such option since no button.</i>	F	<i>Unsuccessful</i>



Test Case Number: 46 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: View item from a category.	Test Case Name View items via main menu. Subsystem: Main menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on a category to view items.	Items in the category will display in a listview.	<i>No items to view</i>	F	<i>Unsuccessful</i>



Test Case Number: 47 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category.	Test Case Name Edit items via main menu. Subsystem: Main menu. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the item you want to view or change.	Item detail screen display.	No items in database	F	Unsuccessful
2	Click Edit Item button	List with current items appear.	No items in database	F	Unsuccessful
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.	No such screen	F	Unsuccessful
4	Click on Cancel	No photo were added.	No such button since no screen.	F	Unsuccessful
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).	No screen, no upload.	F	Unsuccessful
6	Click in the Item Name text field	Keyboard appears.	No screen, no text fields	F	Unsuccessful
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).	No text field.	F	Unsuccessful
8	Click in the Description text field	Keyboard appears.	No text field.	F	Unsuccessful
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).	No screen , no description.	F	Unsuccessful
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.	No screen, no save button.	F	Unsuccessful



11	Click on the item icon.	Notification appears asking for an upload of the image via the camera or the gallery.	<i>No screen, no icon.</i>	<i>F</i>	<i>Unsuccessful</i>
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. <i>(Note: Not yet saved).</i>	<i>No screen, no choice.</i>	<i>F</i>	<i>Unsuccessful</i>
13	Enter the new item detail are before.	Item detail displays.	<i>No screen, no text fields.</i>	<i>F</i>	<i>Unsuccessful</i>
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.	<i>No screen, no save button.</i>	<i>F</i>	<i>Unsuccessful</i>
15	Click on the back button.	Item detail screen appears.	<i>Nothing to exit from.</i>	<i>F</i>	<i>Unsuccessful</i>



Test Case Number: 48 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item via main menu.	Test Case Name Share Item via main menu. Subsystem: Item Detail. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.	No such place	F	Unsuccessful
2	Click in the “Share” option in the pop up screen	Notification that asks you to choose a sharing method.	No such button	F	Unsuccessful
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.	No such option	F	Unsuccessful
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.	No such place to exit from	F	Unsuccessful

*Note: Repeat for each sharing method.



Test Case Number: 47 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category.	Test Case Name Edit items via main menu. Subsystem: Main menu. Design Date: Execution Date:
---	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the item you want to view or change.	Item detail screen display.	No items in database	F	Unsuccessful
2	Click Edit Item button	List with current items appear.	No items in database	F	Unsuccessful
3	Click on the item icon	Notification appears asking for an upload of the image via the camera or the gallery.	No such screen	F	Unsuccessful
4	Click on Cancel	No photo were added.	No such button since no screen.	F	Unsuccessful
5	Upload Image via camera	Camera opens- follow steps used to take a photo – photo appears as the image of the item in the eyeRS app. (Note: Not yet saved).	No screen, no upload.	F	Unsuccessful
6	Click in the Item Name text field	Keyboard appears.	No screen, no text fields	F	Unsuccessful
7	Enter the Name of the item in the item name text field	Title will appear in the text field. (Note: Not yet saved).	No text field.	F	Unsuccessful
8	Click in the Description text field	Keyboard appears.	No text field.	F	Unsuccessful
9	Enter the description of the item in the description text field	Description will appear in the description text field. (Note Change is not yet saved).	No screen , no description.	F	Unsuccessful
10	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add item screen.	No screen, no save button.	F	Unsuccessful



11	Click on the item icon.	Notification appears asking for an upload of the image via the camera or the gallery.	<i>No screen, no icon.</i>	<i>F</i>	<i>Unsuccessful</i>
12	Click on Choose from Library	Files open. Follow steps as needed to navigate to the chosen picture—picture appears as the image of the item in the eyeRS app. <i>(Note: Not yet saved).</i>	<i>No screen, no choice.</i>	<i>F</i>	<i>Unsuccessful</i>
13	Enter the new item detail are before.	Item detail displays.	<i>No screen, no text fields.</i>	<i>F</i>	<i>Unsuccessful</i>
14	Click the “Save changes” button.	Item is edited in the database. Notification of saved action. User remains in add the Add Item screen.	<i>No screen, no save button.</i>	<i>F</i>	<i>Unsuccessful</i>
15	Click on the back button.	Item detail screen appears.	<i>Nothing to exit from.</i>	<i>F</i>	<i>Unsuccessful</i>

Test Case Number: 49 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Share item from the main menu.	Test Case Name Trade Item from the main menu. Subsystem: Item Detail. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.	No such place to click.	F	<i>Unsuccessful</i>
2	Click in the “Trade” option in the pop up screen	Notification that asks you to choose a sharing method.	No such option available.	F	<i>Unsuccessful</i>
3	Select the appropriate sharing method	Share method will open – follow steps as normal for the chosen method.	No methods to select.	F	<i>Unsuccessful</i>
4	Click the back button to return from the sharing app.	User is returned to the Item detail screen appears.	No screen to exit from.	F	<i>Unsuccessful</i>

***Note: Repeat for each sharing method.**

Test Case Number: 50 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Deletes item from a category from the main menu.	Test Case Name Delete Item from the main menu. Subsystem: Item Detail. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more icon in the item detail screen	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.	<i>No such place to click</i>	F	<i>Unsuccessful</i>
2	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.	<i>No pop up</i>	F	<i>Unsuccessful</i>
3	Click the “No” button in the Delete conform notification	Delete confirm notification will disappear and the item detail screen will appear.	<i>No conformation since no pop up.</i>	F	<i>Unsuccessful</i>
4	Click on the more icon.	A pop up with more options (Delete, Share and Trade) will appear in the top right corner of the screen.	<i>No such button.</i>	F	<i>Unsuccessful</i>
5	Click in the “Delete” option in the pop up screen	Notification that asks for confirmation appears.	<i>No such option.</i>	F	<i>Unsuccessful</i>
6	Click the “Yes” button in the Delete conform notification	Deleted notification appears. Item is deleted from the database. Item list appears.	<i>No such button</i>	F	<i>Unsuccessful</i>

Test Case Number: 51 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Exit from viewing the items.	Test Case Name: Exit from viewing items Screen. Subsystem: My Category. Design Date: Execution Date:
--	---

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the back button	Main screen will appear.	Was never in the view item screen.	F	<i>Unsuccessful</i>

Test Case Number: 52 System: eyeRS Designed by: eyeRS Development Team Executed by: Description: Leaving the eyeRS app.	Test Case Name: Exit eyeRS. Subsystem: Main Menu -Side menu. Design Date: Execution Date:
--	--

Test Number	Action	Expected Response	Actual Response	P/F	Comments
1	Click on the more button n the main menu.	A side menu appears.	A side menu appears.	P	Successful.
2	Click on "Exit"	A notification will appear.	Nothing happens	F	<i>Unsuccessful</i>
3	Click the NO	Main Menu will appear.	No such button	F	<i>Unsuccessful</i>
4	Click on the more button n the main menu.	A side menu appears.	Already in side menu	F	<i>Unsuccessful</i>
5	Click on "Exit"	A notification will appear.	Nothing happens	F	<i>Unsuccessful</i>
6	Click the Yes	User will exit the eyeRS app.	No button clicked so nothing happens, still in the app.	F	<i>Unsuccessful</i>



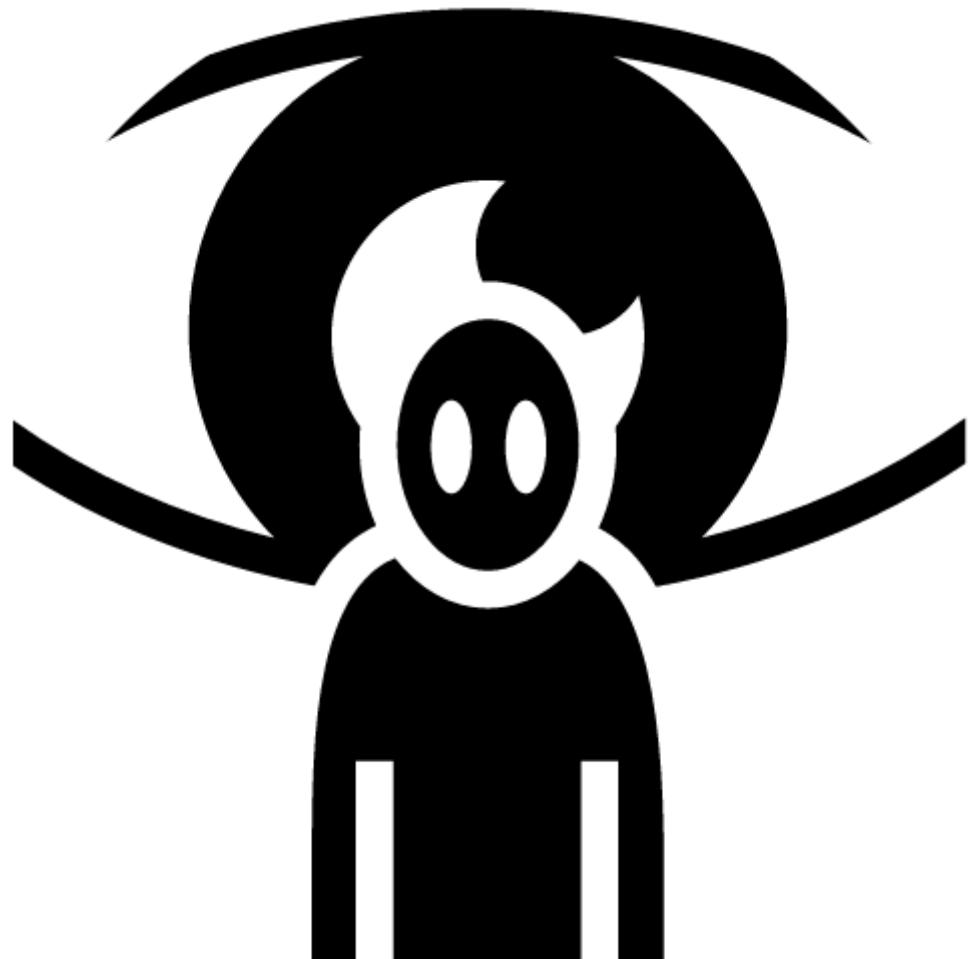
2. Customer sign-off

Customer name and surname	Customer signature
Group leader name and surname	Group leader signature



User Manual

eyeRS Development Team - ITSP200 (Deliverable 5)





Introduction

The proposed eyeRS mobile application (eyeRS app) is designed to allow users to catalogue their personal effects and belongings while creating a platform to facilitate a trade or simply just share their items.

The eyeRS app provides individuals with the power to manage and organise their personal and professional lives with less workforce. The eyeRS app achieves this control by allowing users to freely access and catalogue all of their belongings and items that are uploaded to the app. Users will be able to make use of the application on the go as it is a mobile application for android users. Each item is saved to a database, and allows the user to specify details for each item by means of photo identification and a short description.

The app allows users to be connected with friends, family as well as other users who have shown interest in items owned by other user. The connection is achieved by allowing users to share and trade items on their catalogues. The app also allows a user to be connected with their personal belongings on the go so they may view items stored in their catalogue.

The user manual for the eyeRS application is a guide to help users understand how to navigate within the app as well as how to make full use of the eyeRS app. The document will tell the user to have assistance from the first step which is installing the app down to the last step which is leaving the app once finished.

The development team of the eyeRS app are Nathan Shava, Matthew Van der Bijl, Emilde Arsenio, Andrea Cloete and Sajaad Ishmail. Should the user require further assistance with the app they can contact the dev team for assistance via the eyeRS development group at the following email address: itsp200pearson@gmail.com.

Getting Started

Now that we know what the eyeRS app is as well as who to contact if a user requires further assistance we can go ahead and show the user how to use the app as it is meant to be operated. The following instructions will guide the user on the functionality of the app.

User Characteristics

The table below indicate user characteristics and preferences for the use of the eyeRS app.

Characteristic	Preference
Age	3+
Gender	Male/Female
Educational level (Level of experience)	A basic knowledge of operating mobile devices.
Language	English
Computer skills	Novice; Technologically literate.
Domain-related knowledge and skills	Interior design.
Physical environment	Home, office, on the go.
Social environment	Family members, friends or business..

Hardware & Software requirements

The table below is a list of the minimum mobile specifications for the customer and/or other users to run the mobile application:

Hardware	Software
Chipset: ARM-based or better.	OS Codename: Android JellyBean or better.
Memory: <ul style="list-style-type: none">● 750 MB RAM; and;● 2 GB Flash External.	OS Version: 4.3.x or higher.
Storage Type: <ul style="list-style-type: none">● Internal;● Mini SD; or● Micro SD.	OS API Level: 18 or higher
Primary Display: <ul style="list-style-type: none">● QVGA TFT LCD or larger; and;● 16-bit/16M colours.	Browser: <ul style="list-style-type: none">● HTML; and;● Adobe Flash Lite.
Navigation Keys: <ul style="list-style-type: none">● 5-way navigation;● 5 application keys;● Power;● Camera; and● Volume controls.	Messaging: <ul style="list-style-type: none">● SMS;● MMS;● Email; and;● Push Email.
Camera: 2 MP CMOS or better	
Sound: <ul style="list-style-type: none">● Loudspeaker; and;● 3.5mm audio jack.	
USB: Standard microUSB v2.0 or better.	
Bluetooth: v1.2.	
Network:	

- (2G Bands) GSM 850 or better;
- (3G Bands) HSDPA 900 or better;
- 4G Bands LTE band 1(2100) or better;
- GPRS;
- EDGE;
- GPS; and;
- WLAN (Wi-Fi 802.11 b/g/n).

How to install eyeRS

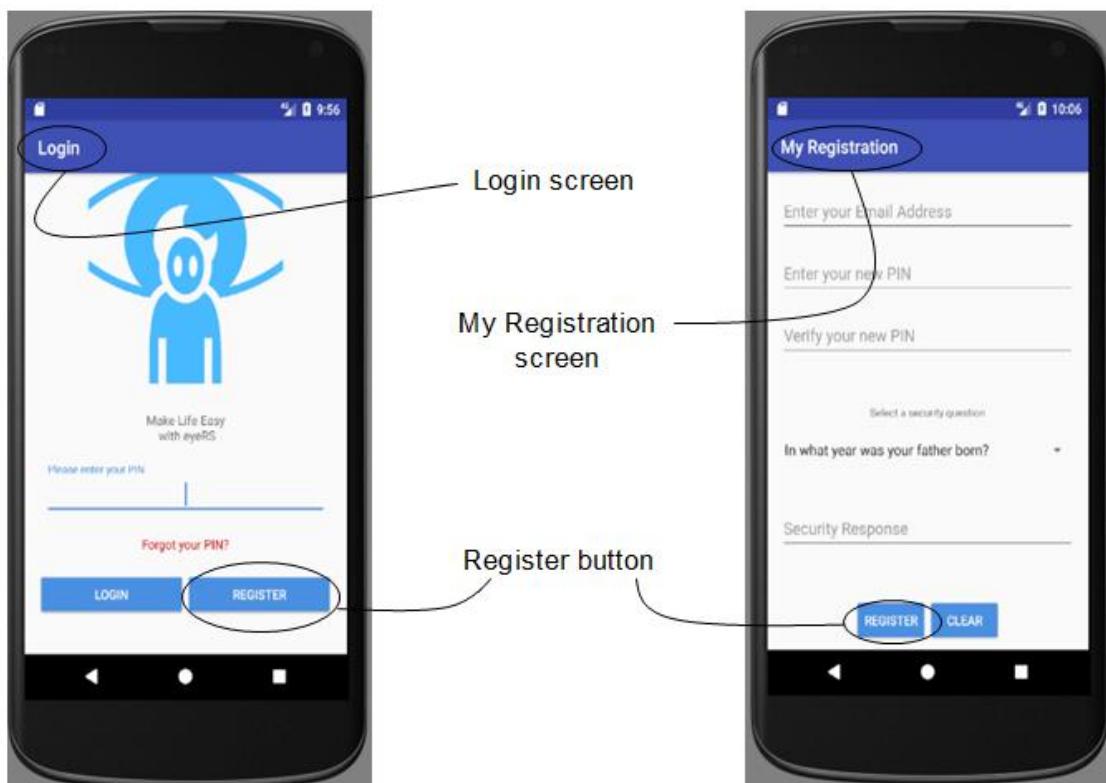
A user can install the eyeRS app from the Android play store.



1. Opening the **Play Store**.
2. Enter “eyeRS” into the search bar;
3. Download the application, the eyeRS app will install on your mobile device; and;
4. The eyeRS app can be found in the applications menu, or on the home screen of the mobile device.

How to... Register

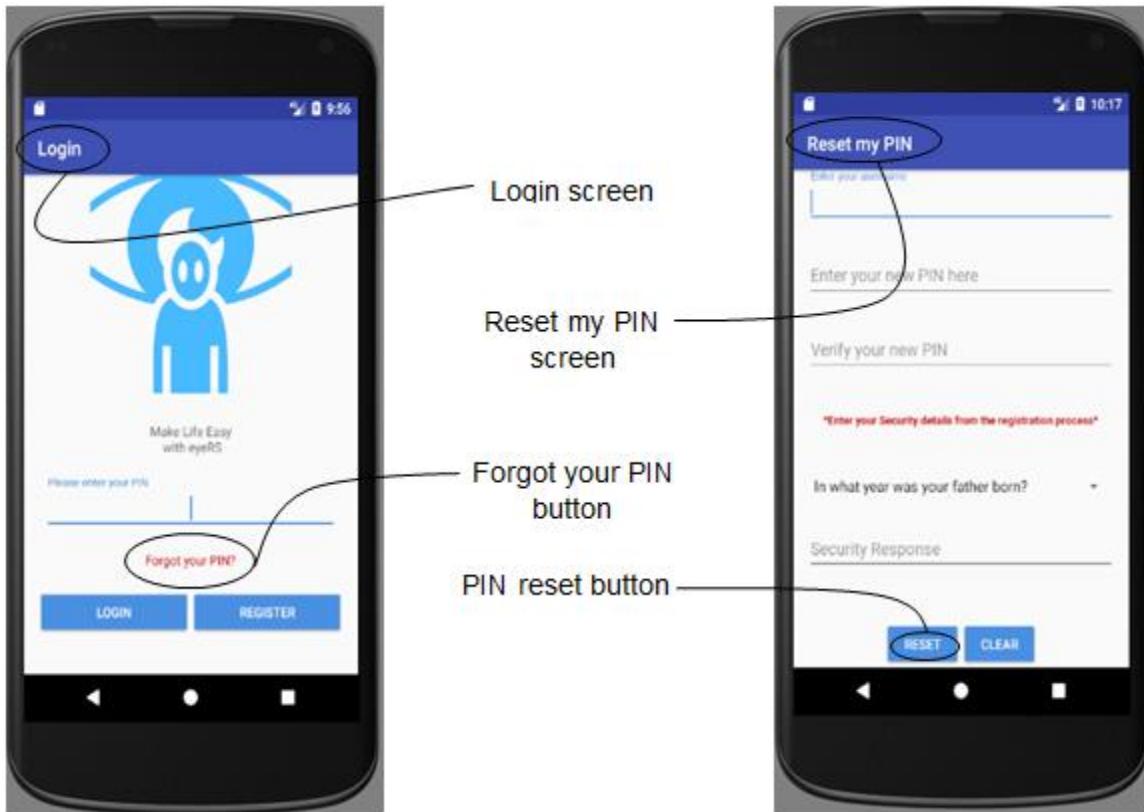
1. Press on the ‘REGISTER’ button on the Login screen . The Registration should screen appear;
2. In the registration screen fill in all the required text fields; then;
3. Press the ‘REGISTER’ button . You will be notified about the state of the register process.



Reset a forgotten PIN

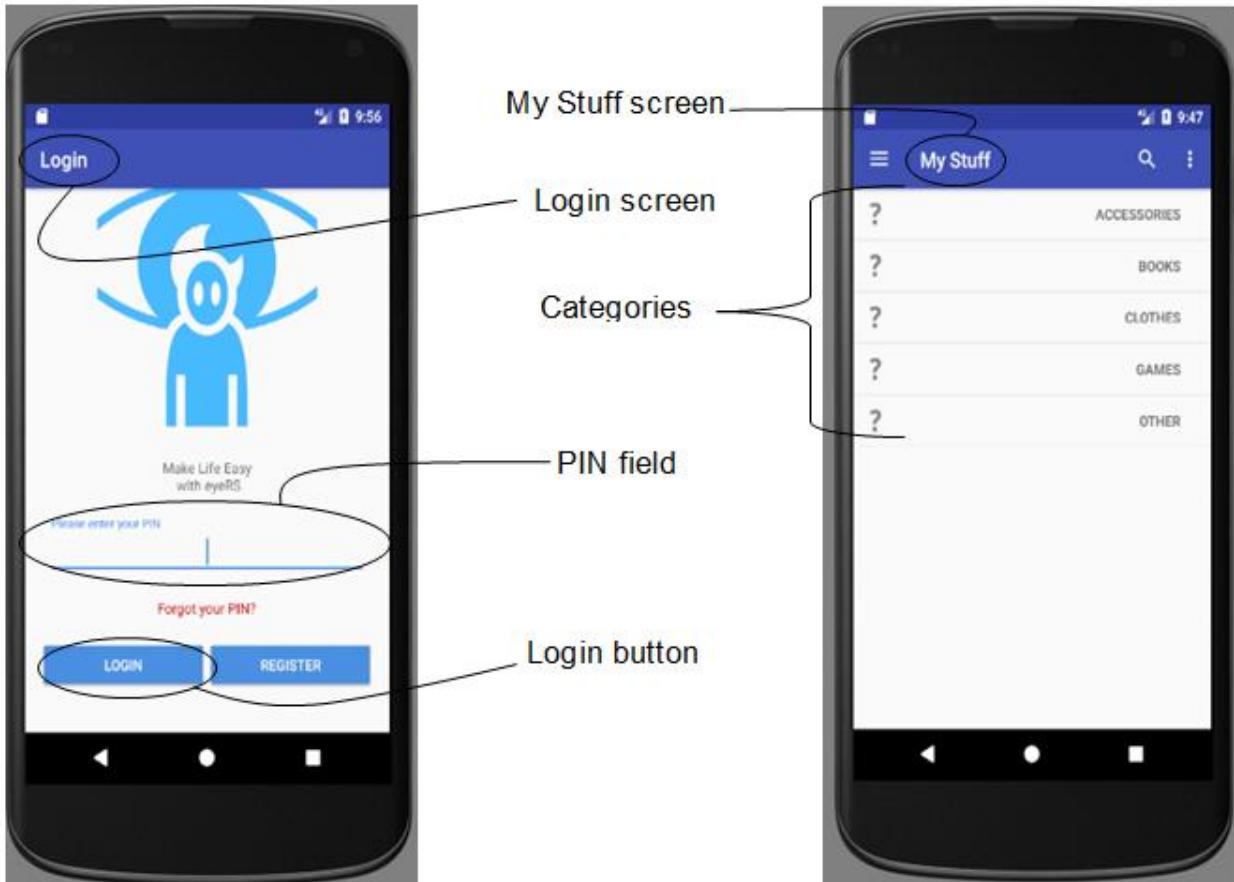
1. Press on the ‘Forgot PIN’ text in the Login screen . The Reset my PIN should screen will appear;
2. Enter all the required fields; then;

3. Press the 'RESET' button . You will be informed about state of the process.



Login

1. In the Login screen enter your PIN in the Please enter your PIN text field; then;
2. Press the 'LOGIN' button  . The My Stuff screen will appear.



Add Item

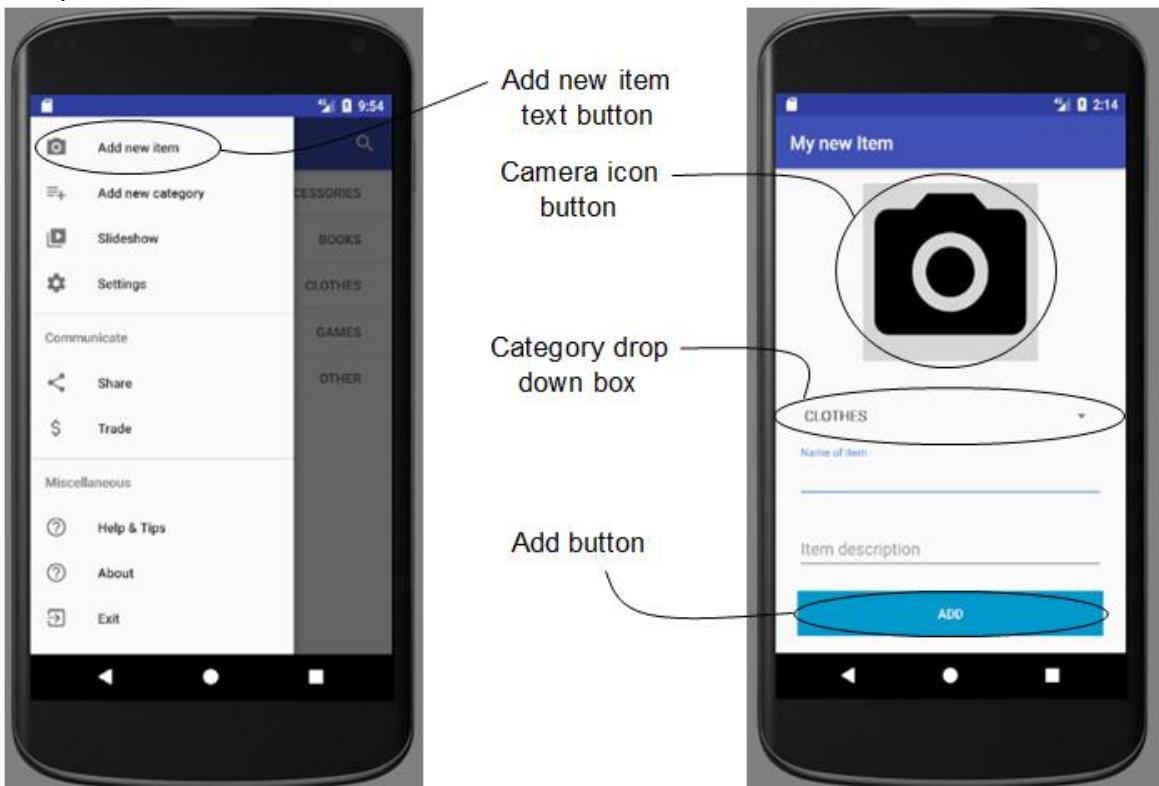
1. Press the more icon on the My Stuff screen ;

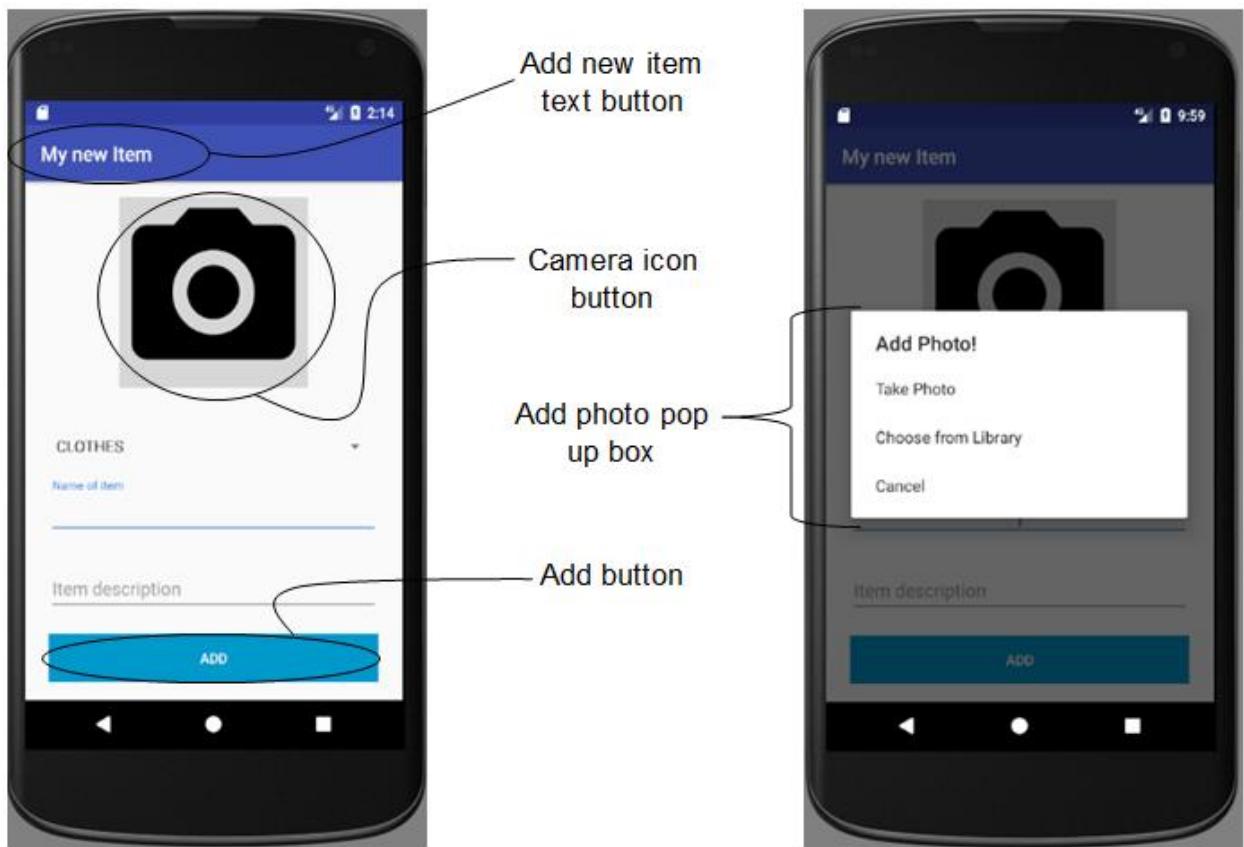
2. In the side menu press the 'Add new item' text .
screen will appear;
3. Add image of item;

- a. In the My new Item screen, press on the camera icon  . The Add Photo pop up appears.
- b. In the Add Photo pop up select the 'Take Photo' text  . The camera on the device will open.
- c. Take a photo. The photo will appear as the item image in the eyeRS app.
OR
- a. In the Add Photo pop up Select 'Choose from Library'  . The library on the device will open.
- b. Navigate through the library to the desired image.
- c. Select the desired image. The selected image will appear as the item image icon in the eyeRS app.

4. Fill in all the required text fields.
5. Press the 'Add' button .

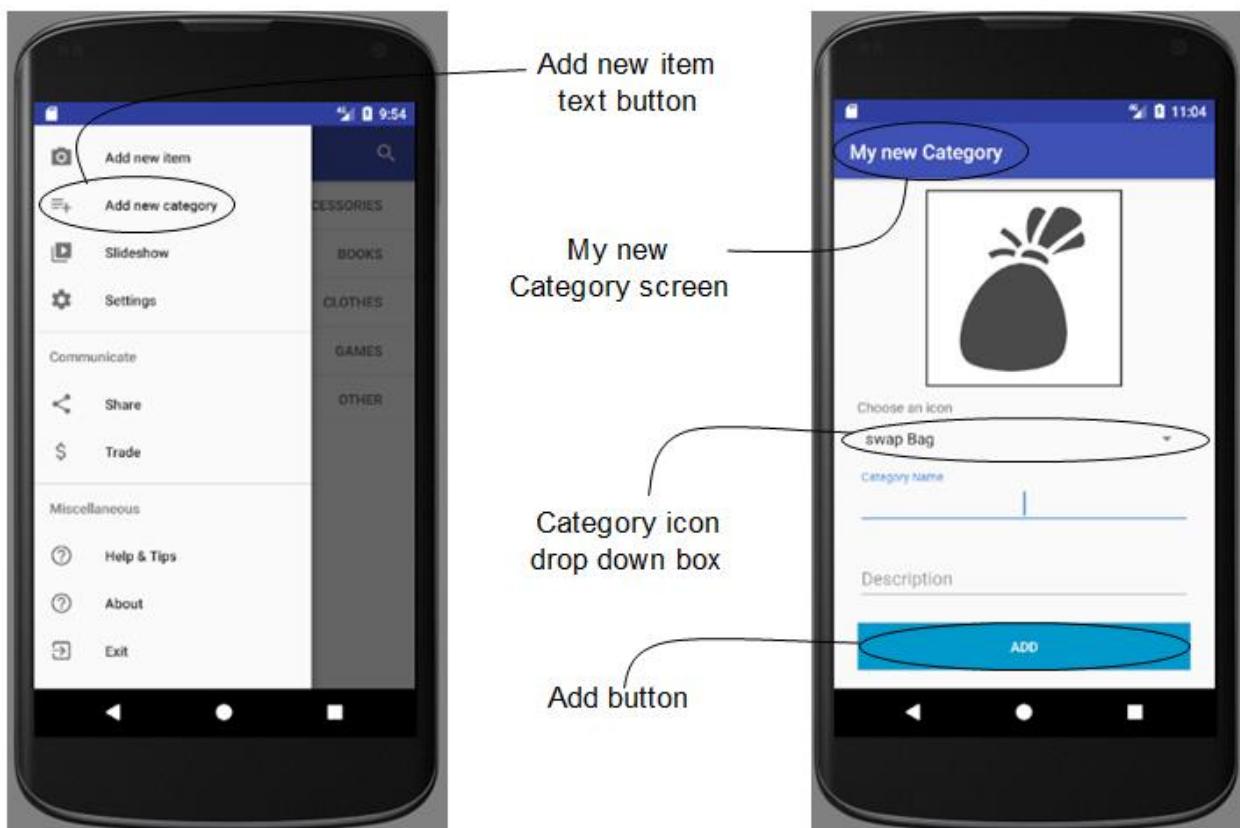
To exit press the back button .





Add a category

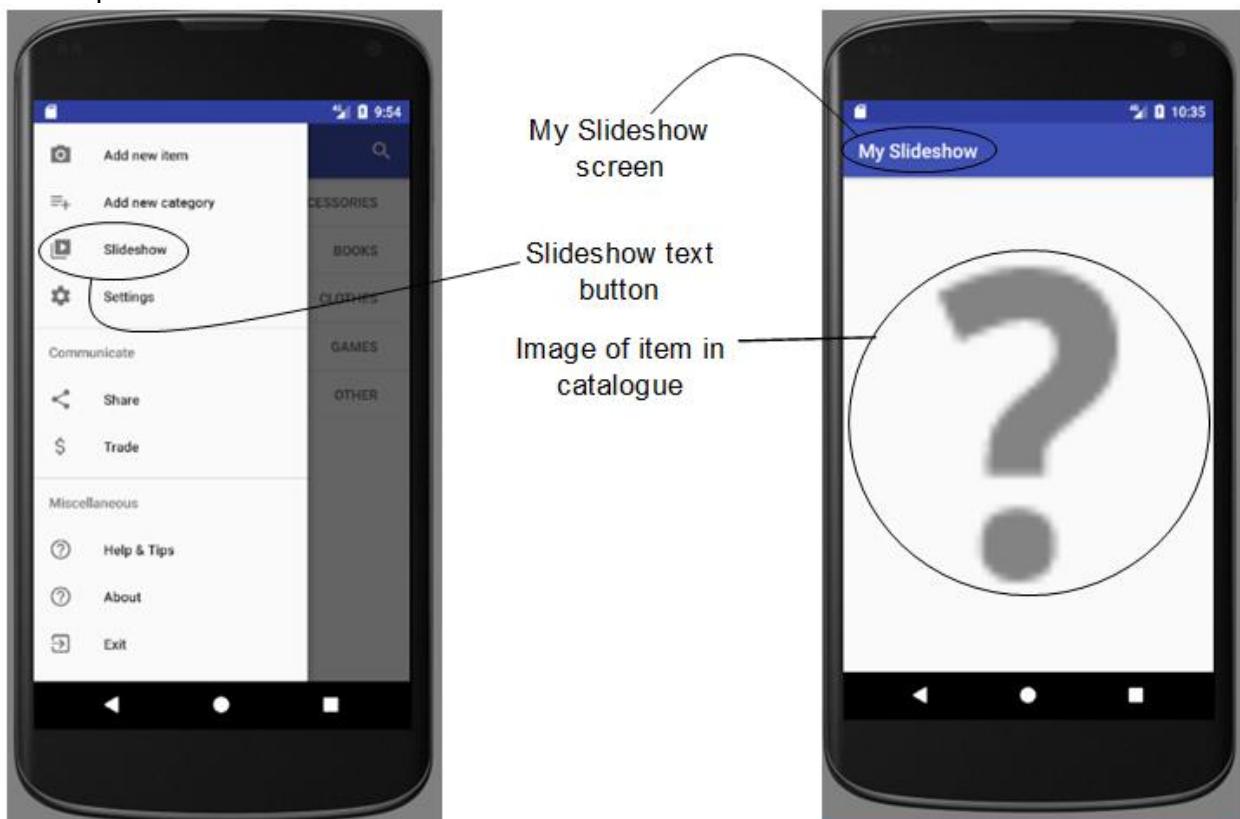
1. Press the more icon  . Side menu appears;
2. Press the 'Add new Category' text  . The My new Category screen appears;
3. Choose a category icon from the Choose an icon drop down box;
4. Press Category Name text field Keyboard appears;
5. Type the desired name of the new category;
6. Press on the Description text field. Keyboard appears;
7. Type the desired description on the ew category; then;
8. Press the 'ADD' button .



View the slideshow

1. Open the side menu .
2. In the side menu click on the **Slideshow** text.
 - The slideshow will start.

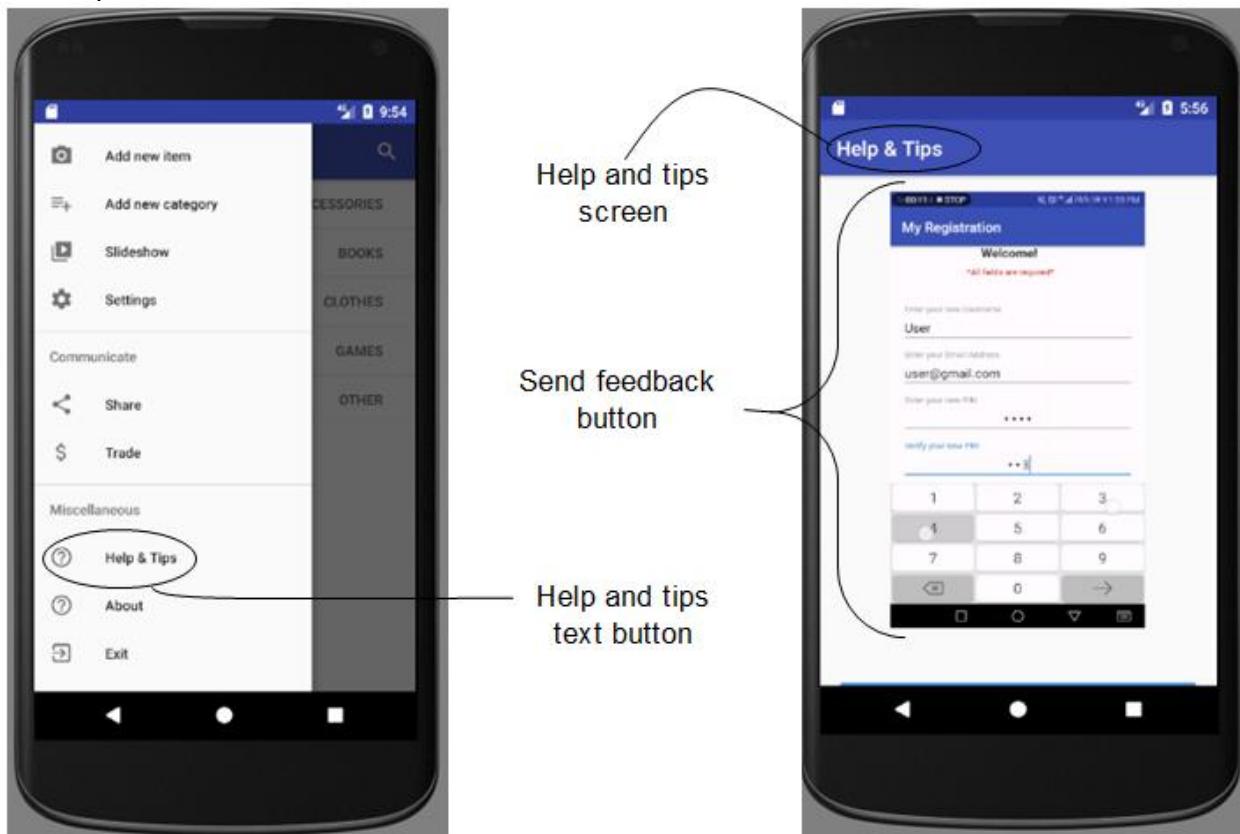
To exit press the back button .



Access help and tips

1. Press on the more icon in the My Stuff screen 
 2. In the side menu click on the 'Help & Tips' text 
 3. Press on the video to play the tips available; then;
 4. Press the 'Send Feedback' button in the Help and Tips screen 
- You will be taken to the Android play store where you can leave your feedback or ask for additional assistance.

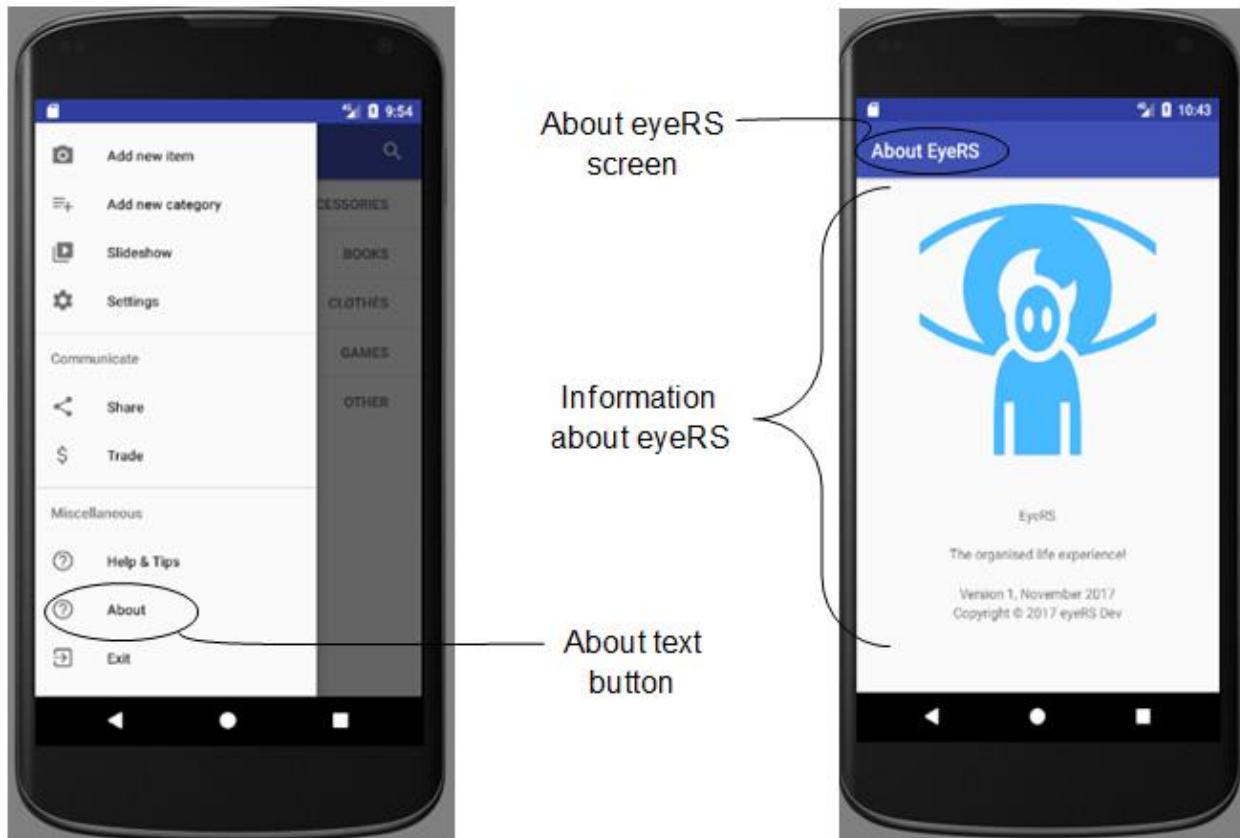
To exit press the back button .



Access information about the app.

1. Press the more icon in the My Stuff screen ; then;
2. On the side menu press the **About** text  . Information about the eyeS app can be viewed.

To exit press the back button .



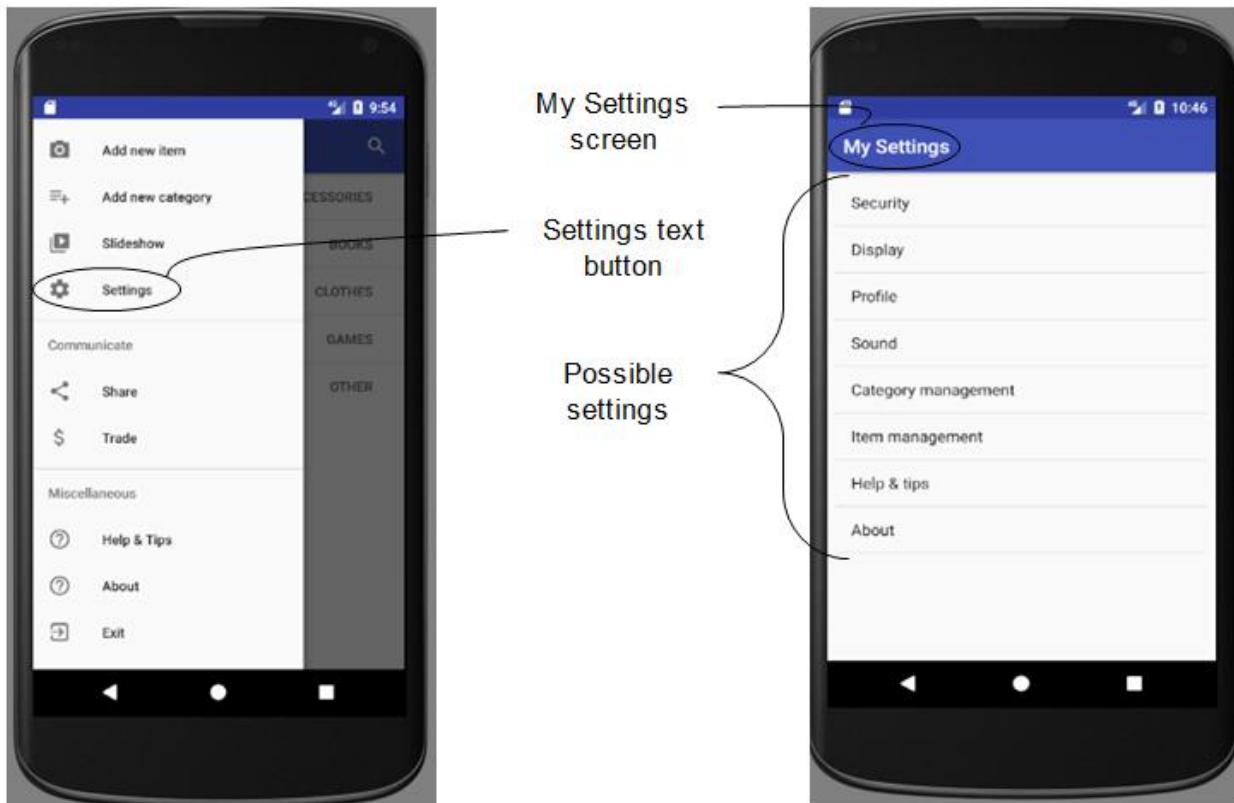
Access the settings

1. Press on the more icon on the top right corner of the My Stuff screen ; then;
2. Press the 'Settings' button in the side menu .

OR

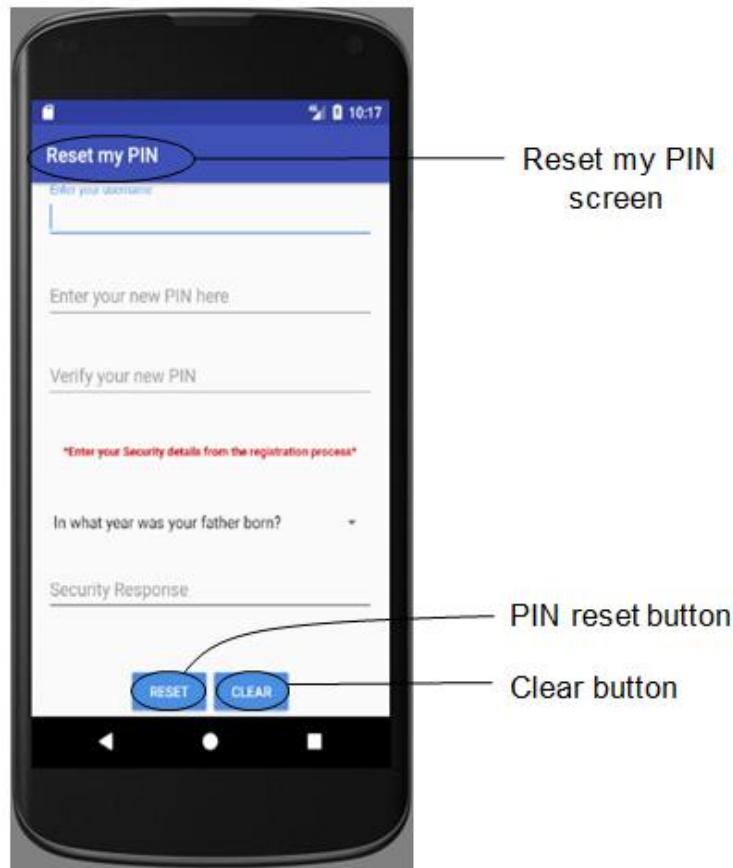
1. Press the more icon on the top left corner of the My Stuff screen ; then;
2. Press the 'Settings' text in the pop up .

To exit press the back button .



Reset your PIN

1. In the Settings screen press the 'Security' text  . The reset pin screen will appear where you can reset your pin;
2. In the Reset my PIN screen fill in all the required fields; then;
3. Press the **RESET** button 



To exit press the back button .

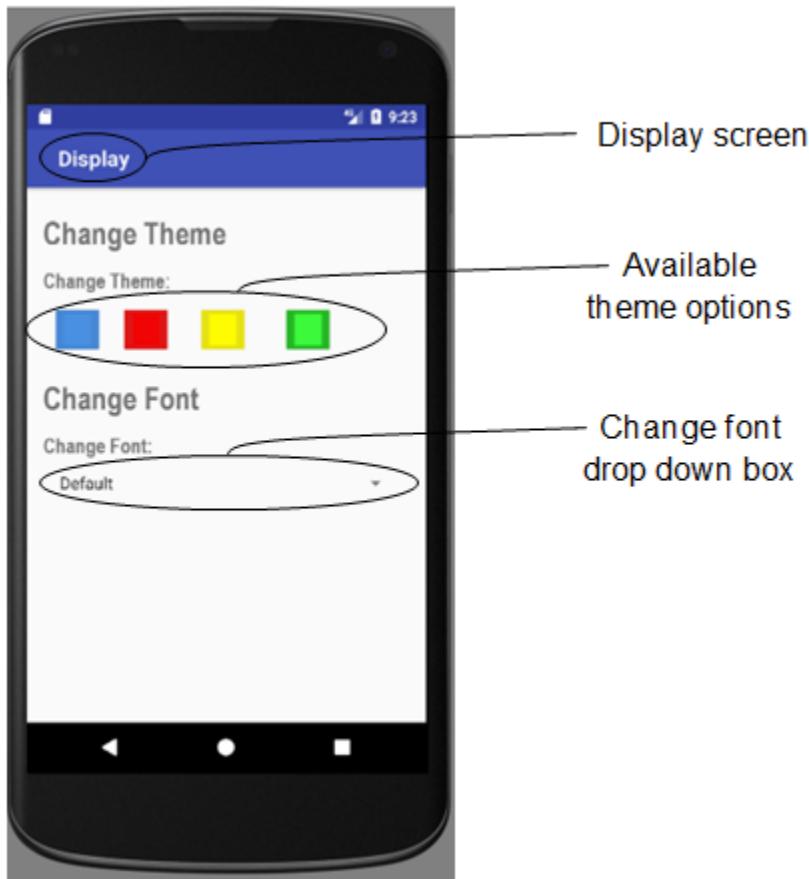
Change Display Options

Change Theme

1. In the My Settings screen press the **Display** text  . A Display setting screen will appear where different display settings can be accessed; then;
2. In the Display setting screen, under the Change Theme select the appropriate color     . The theme of the app will adjust according to the chosen color.

Change Font

1. In the Display setting screen, under Change Font press on the drop down box  . A list of supported fonts will appear; then;
2. Select the appropriate font from the drop down list. The font will be implemented on the

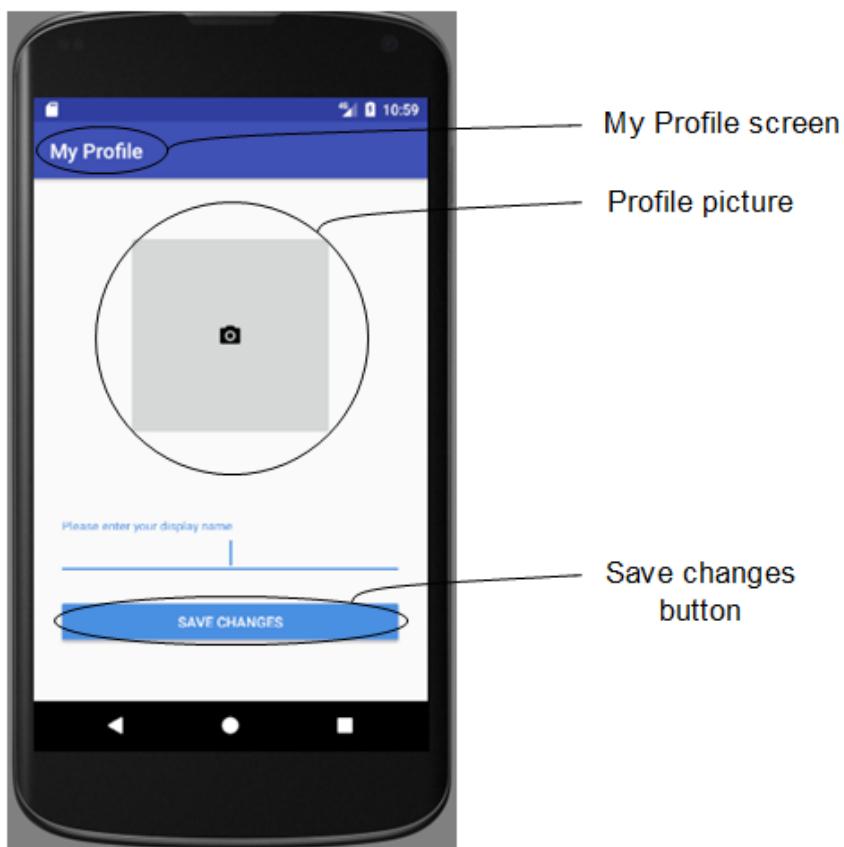


app.

Change profile details

1. In the My Setting screen press on the **Profile** text in the settings screen **Profile**.
 2. Add or change profile picture.
 - a. In the My Profile screen click on the profile picture.
 - b. Select **Choose from Library**. The library on the device will open.
 - c. Navigate through the library to the desired image.
 - d. Click on the desired image. The selected image will appear as the profile picture in the eyeRS app.
- OR
- a. In the My Profile screen click on the profile picture.
 - b. Select **Take Photo**. The camera on the device will open.
 - c. Take a photo. The photo will appear as the profile picture in the eyeRS app.
3. In the My Profile screen click on the **Please Enter your display name** text field.
 - A keyboard will appear.
 4. Enter your desired display name.
 5. Press Click on the **SAVE CHANGES** button **SAVE CHANGES**.
 - Profile picture is now saved

To exit press the back button .



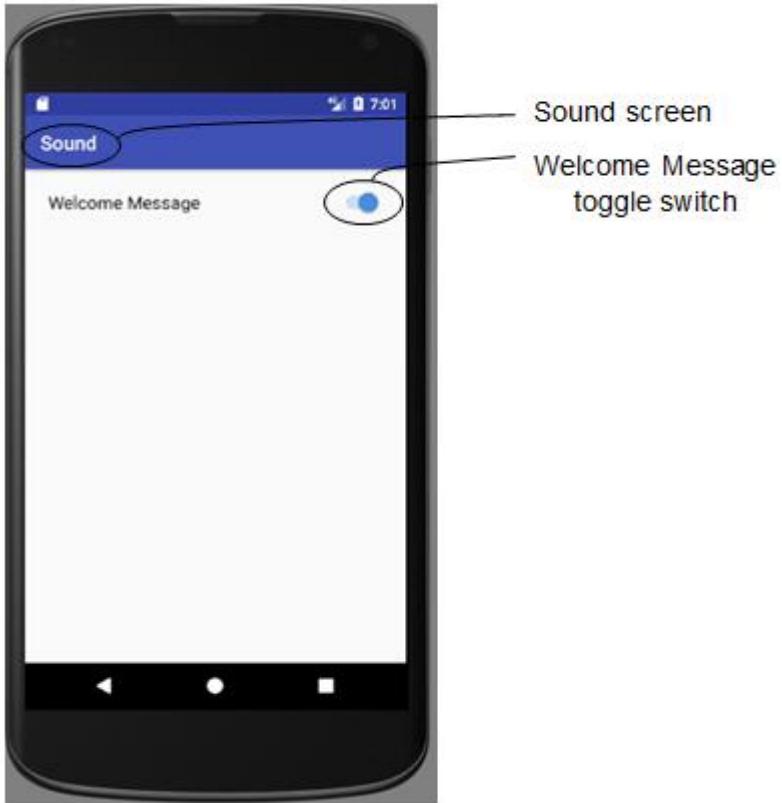
Configure sound settings

Configure the welcome message.

1. In the setting screen press the **Sound** text **Sound**.
 - A Sound setting screen will appear.
2. In the Sound setting screen press on the **Welcome Message** toggle switch **Welcome Message** .

- Toggle switch will appear blue if the welcome message is activated and grey if deactivated.

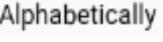
To exit press the back button .



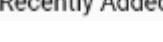
Sort the categories

1. In the My Stuff screen press the more icon .
2. In the side menu press **Settings**  .
3. In the My Setting screen press **Category management**   . The Category Management screen will appear.
4. In the Category Management screen, press the **Change Sorting**   text. A Select Sorting Preference screen will appear.

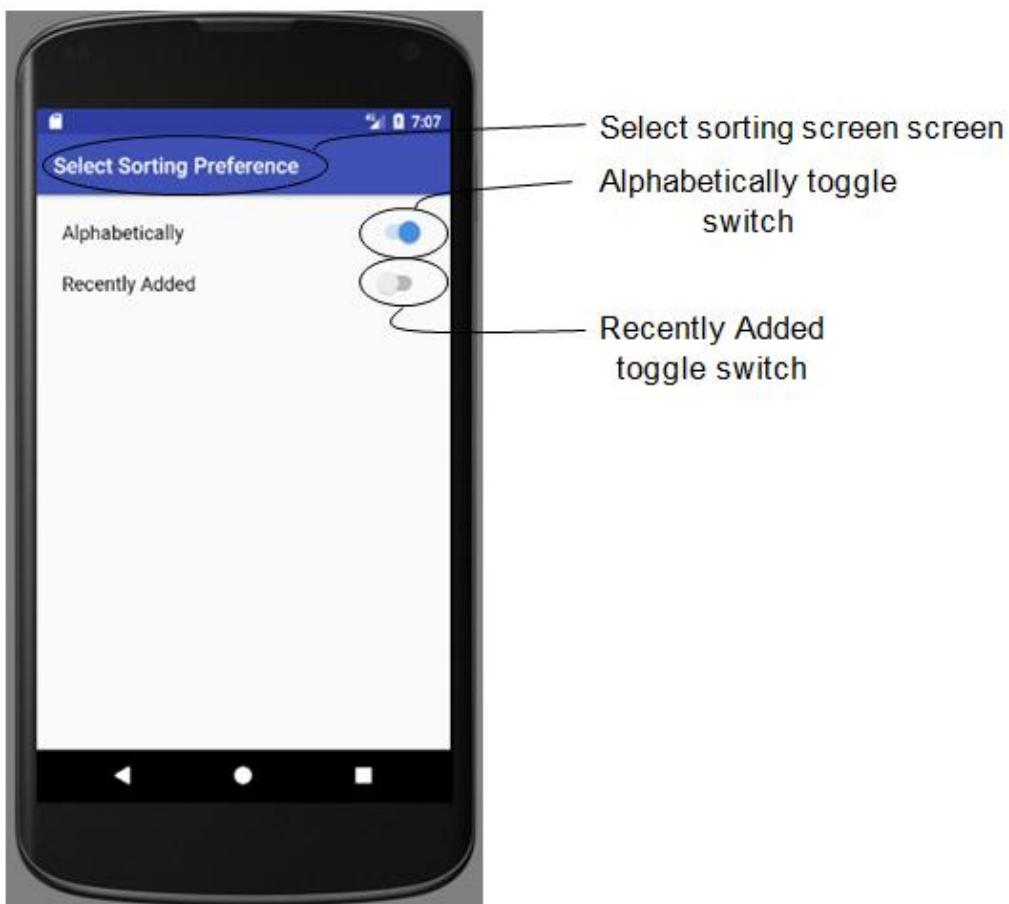
Sort categories alphabetically.

- a. In the Select Sorting Preference screen press on the **Alphabetically** toggle switch  
 - Toggle switch will appear blue if the touch sound is activated and grey if deactivated.

Sort categories by recently added.

- b. In the Select Sorting Preference screen press on the **Recently Added** toggle switch  

- Toggle switch will appear blue if the touch sound is activated and grey if deactivated.



deactivated. To exit press the back button .

Sort the items

1. In the My Stuff screen press the more icon 
2. In the side menu press **Settings**  Settings
3. In the My Setting screen press **Item management**  Item management
4. In the Item Management screen press **Change Sorting**  Change Sorting

Sort alphabetically.

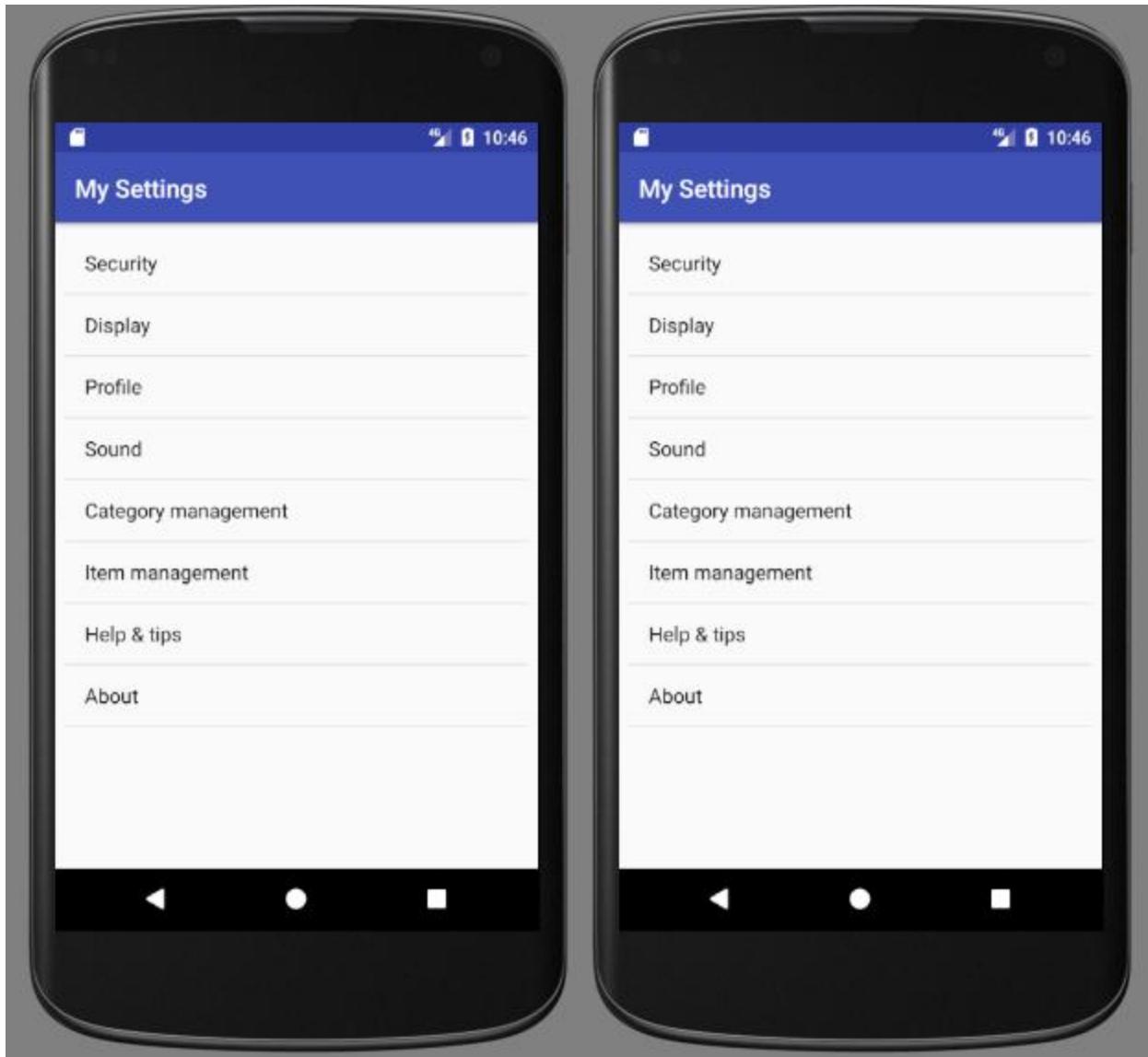
- a. In the Select Sorting Preference screen press the Alphabetically toggle switch  Alphabetically

- Toggle switch will appear blue if the items is sorted alphabetically and grey if sorted in a default manner.

Sort items by recently added.

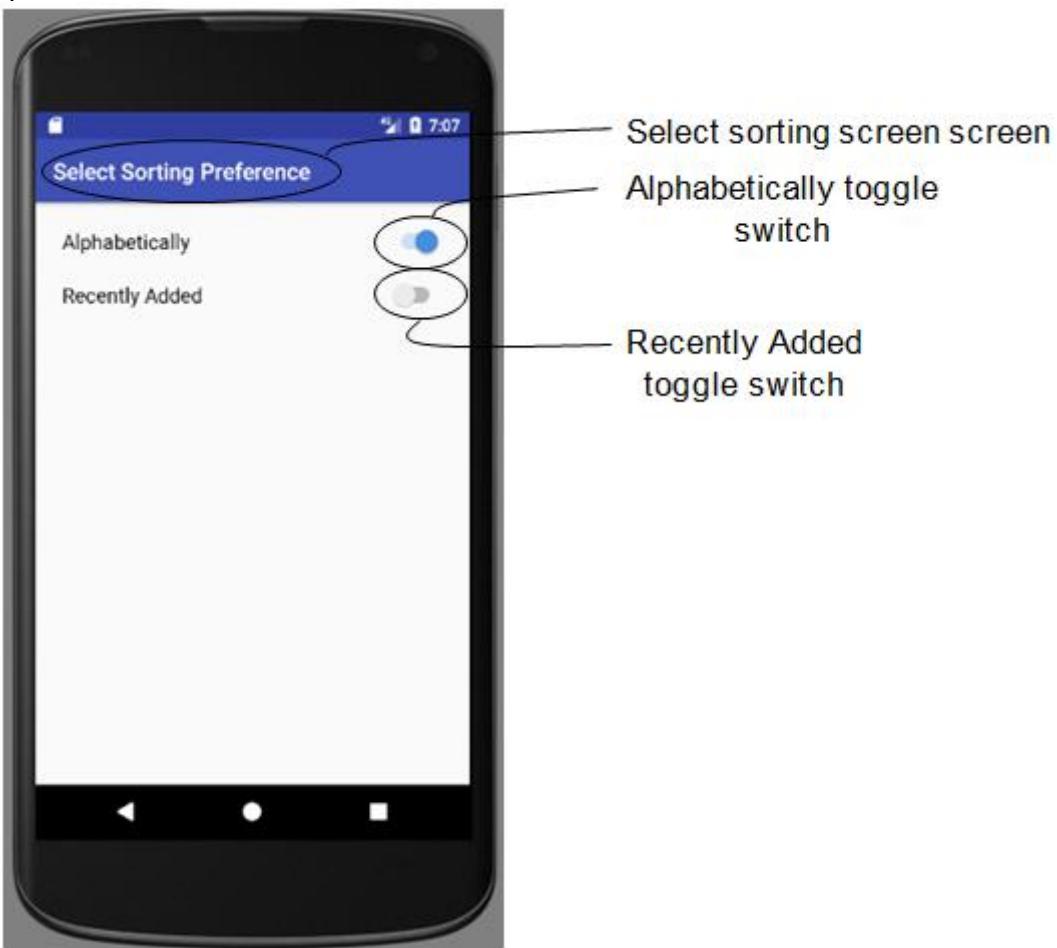
- b. In the Select Sorting Preference screen press the Recently Added toggle switch  Recently Added

- Toggle switch will appear blue if the items is sorted according to recently added items and grey if sorted in a default manner.



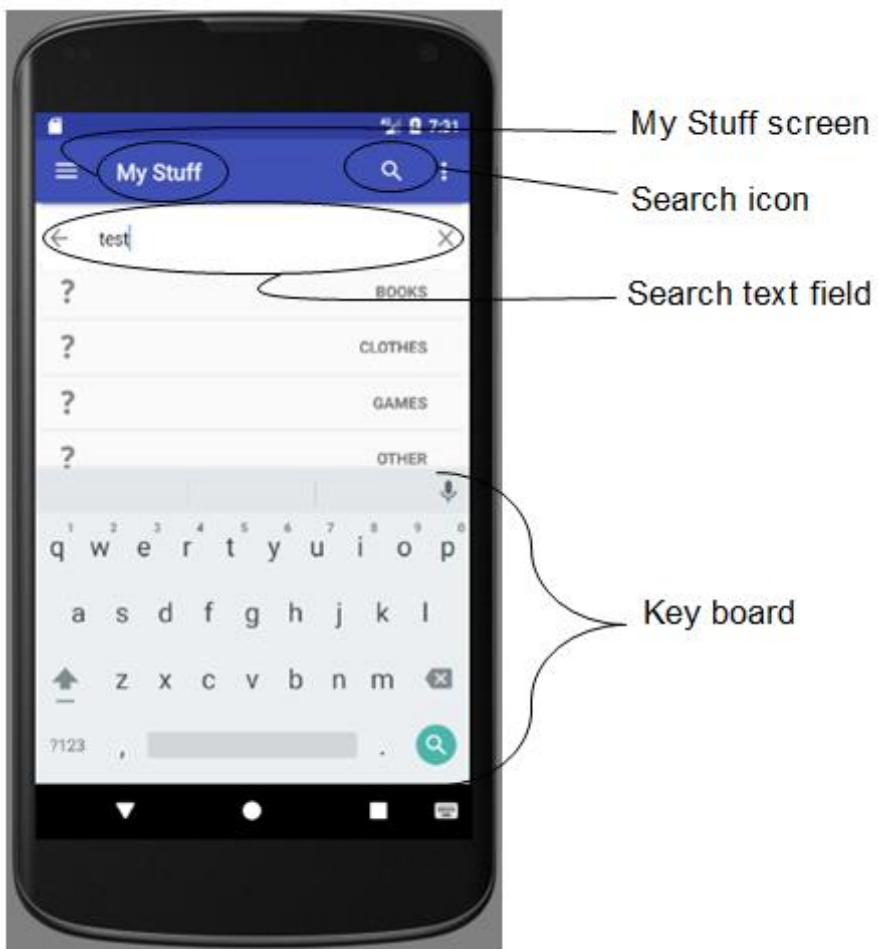


To exit press the back button .

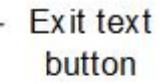


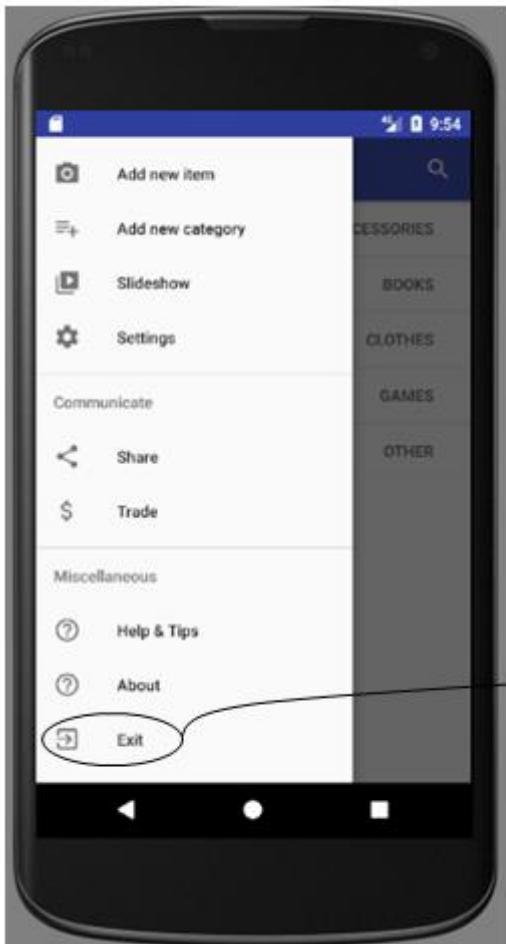
Search

1. In the My Stuff screen press the search icon .
2. Type the search detail as you desire in the search field.
3. Press the search icon on the keyboard .



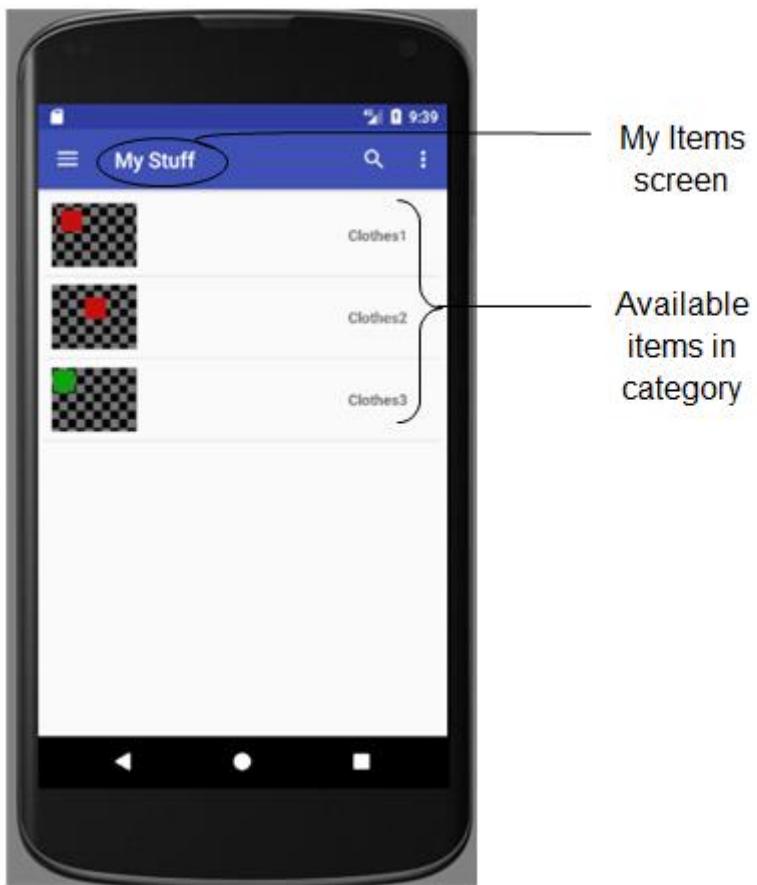
Exit

1. In the My Stuff screen press the more icon .
2. In the side menu press **Exit**  



View items in a category

1. In the My Stuff screen press on a category icon.
 - All the items in the category appears.





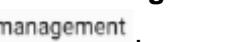
Edit a category

1. Open the side menu  and select **Settings**.
2. Select **Category management** 
3. Select **Edit Category** 
4. Select **Take Photo**
 - Click the drop down menu
 - Select an icon
5. Enter a category name.
6. Enter a description.
7. Press the **SAVE CHANGES** button .

Delete a category

1. Open the side menu  and select **Settings**.
2. Select **Category management** 
3. Select **Delete Category** 
 - A list of the categories are displayed.
 - Select a category to be deleted.
4. Press the **Delete** button.

Edit an item

1. Open the side menu  and select **Settings**.
2. Select **Item management** 
3. Select **Edit Item** 
4. Press the camera icon .
5. Select an option from the popup.
 - Select **Take Photo**
 - Select **Choose from library**
 - Select cancel (To cancel the upload)
6. Enter an item name.
7. Enter a description for the item.
8. Press the **SAVE CHANGES** button .

Delete an item

1. Open the side menu  and select **Settings**.
2. Select **Item management** .
3. Select **Delete Item**.
 - A list of the items are displayed.
 - Select an item to be deleted.
4. Press the **Delete** button.

Share

1. Open the side menu  and select **Settings**.
2. Select **Share** .
3. Select an item or category to share.
4. Select the application through which the share will be used. (Example. WhatsApp )
5. Follow each application's protocol for sharing links.

*Note: Share provide less information about the item than the trade function.

Trade

1. Open the side menu  and select **Settings**.
2. Select **Trade** .
3. Select an item or category to Trade.
4. Select the application through which the trade will be used. (Example. WhatsApp )
5. Follow each application's protocol for sharing links.

*Note: Trade provide more information about the item than the share function.

Glossary

Acronyms and abbreviations

The table below explains some of the acronyms and abbreviation that were mentioned throughout the user manual.

Acronyms and Abbreviations	
Acronyms/Abbreviation	Expanded
API	Application Programming Interface
App	Application
ARM	Advanced RISC (Reduced Instruction Set Computer) Machines
HTML	Hyper Text Markup Language
OS	Operating System
PIN	Personal Identification Number
RAM	Random Access Memory
USB	Universal Serial Bus



Evaluation Report

eyeRS Development Team - ITSP200 (Deliverable 5)



1. Evaluation Report

1.1 Introduction

As the development of the eyeRS app has come to an end, we have constructed this evaluation report as a conclusion to the development process. The evaluation report will contain information on the final system scope. The client outlined objectives and requirements for the app. The system specifications will be assessed to determine whether the client's requirements were met or not.

Stating the outcome of the project and whether the project was a success or failure and giving reasons for failure or successful outcomes. The report will contain an overall performance report for the team. There will be detailed descriptions of what worked and can be used in the future for improvement, and descriptions of what caused tension or failure that was experienced during the project by the team members. Knowledge can be gained from the good and bad experiences, and this could lead to growth, whether it is experienced by the team as a whole or specific individuals.

Time management will be discussed in the report to state whether the team managed the project schedule effectively. Whether or not the schedule was managed well, it should be explained so that there can be an improvement for future projects that are to be undertaken. Each member of the team will give a short description on the lessons learnt within the team after the formation at the beginning of the project lifecycle.

A short conclusion will be given to describe the group's performance.

The proposed eyeRS mobile application (eyeRS app) is designed to allow users to catalog their personal effects and beloved belongings while creating a platform to facilitate the trade of their items in the future.



1.2 Final System and Customer Requirements

The eyeRS mobile application consists of various inventories for the user's items. A user is able to upload these items onto the application via the built in camera or a local file source such as the gallery. The concept of a 'bid or buy' system is implemented for users who wish to trade any unwanted items with other people. All items that are selected for trading are available to those users who would have received the intents. Contact information is made available for interested parties. Personal inventories can be shared with other users via social media WhatsApp or by means such as Bluetooth connectivity.

Included in the system requirements is information users need to know about their software and hardware components in order to run the eyeRS app successfully. The eyeRS app will be a conventional app when it comes to requirements, in which users will need to have any device that runs an Android OS of JellyBean/better. There are many apps on the market, each one with its identifiers, look and functionality. eyeRS will help users organise their belongings (virtually) into categories, or to trade and share items by using a single app.

1.3 Group Dynamics and Collaboration

As we know group dynamics have to do with the group behavior patterns and the attitude of the group. This concerns the interactions and forces between the members of the group. The information with regards to how the group was formed, the roles and structure of the group, and the functioning of the group is important in this regard to ensure success.

As a group we went through different stages. The first stage we went through was the formation of the group. This stage took place at the beginning of the year. This was the first time we met as a group. At this stage we had some ice breaker conversations to get to know each other as individuals as well as to get comfortable with each other. As it is normal for members to meet for the first time each of us acted independently, behaved at our best and focused on ourselves individually. After we got used to each other we started to set goals and tasks to achieve as a team.

As a new group there was a need to have the group leader provide us with standards required in order to ensure that each deliverable was met to the scope and quality of the client which was exercised throughout the entire project's lifecycle.

The second phase that we went through was the storming phase. This phase allowed us to grow by raising questions about the group and the goals of the group as it also allowed each of us to find our roles in the first step of the system development process. In this stage the group leader would intervene when disagreements arose on when goals and system specifications were presented. This improved our motivation and allowed us to become stronger leading us to work more effectively as a team.

The third phase went through is the norming phase. This stage commenced once all our issues were resolved which lead to greater trust and cooperation among team members. At this phase we became aware of the competition of other groups which then, as a result, made us realise that our common ideas and interests kept us working efficiently together. In this phase each individual took responsibility and had the ambition to work towards completing the project

After the norming stage we went through the performing stage. In this stage we successfully established roles which lead to us to accomplish synergy. At this point we are knowledgeable and comfortable allowing each team member to work atomically to achieve the goals and handle decisions without having to depend on the guidance for every step.



1.4 Time Management

The overall time management of the project was handled well beyond satisfactory standards. Each deliverable was completed well in advance so as to avoid late submissions which could have compromised the project's success.

Tools including WhatsApp, for group communication purposes; Google Docs, for all word processing and documentation requirements; Google Drive, for sharing of resources and files; and Github, for version control and collaboration of the system development, enabled the team members to work efficiently around the clock from different locations to ensure that the tasks could be carried out and completed according to the required specifications.

An agile approach to the development and testing of the system would have helped the team members to carry out and complete the required tasks well in advance. As a result of utilizing the waterfall development approach, key tasks such as system testing and integration had to be carried out with a few team members as there was a need to allocate roles for other tasks, so as to ensure that the remaining project specifications would be met on time. This was also a result of minimal planning carried out for completing key tasks that required more time and the entire team's workforce.

1.5 Lessons Learnt

I have learned a lot from working in a group which consists of each member bringing to the team a different skill set. The biggest thing that I will be able to take away from this project would be the knowledge I gained from learning to use the android studio application which I know to be valuable for my growth as a developer. I can use this knowledge to further build a good foundation. One area that could use a little more work would have to be communication. The communication was great at the start, up until a point during the course of the project. Where the communication would start slacking towards the end as the pressure built up. All in all it was a good experience due to a great team that have assisting in learning the new skills that had to be learned in order to complete the project. - Sajjaad Ishmail

I discovered that in any task you come across, careful planning can be the deciding factor between success or failure if it is not carried out and observed well. Having different perspectives and opinions of other team members also showed me that a simple or dull idea could have a great outcome as each individual would have a different angle and approach to solving the same problem. Communication and motivation are also key ingredients in any group setting and if they weren't utilized effectively then surely the project would have been a failure from the very beginning. Time is also a rare and valuable commodity. Especially in a group setting where it is so easy to relax as one could have the perspective that other team members are managing the project schedule, however, it is everyone's responsibility to keep track of the remaining time to complete each task to ensure that each goal is completed on time. Above all, effective planning, group collaboration and improved development skills using various tools is definitely something I can take away from this experience. - Nathan Shava

From the start of the development of the eyeRS, I knew that there were going to face multiple obstacles to overcome. The most valuable lesson that I learnt from the obstacles that we encountered through the development process was time management. I learnt how important it is to stick to the time line that was laid out for the development of the system. Along with the time management skills that I gained I learned that groups can make better decisions than individual since I saw different perspectives, constructive deliberation, questioning, and critical analysis which resulted in better solutions and performance. This lead to me learning how to work with other students with different backgrounds and experience. - Andrea Cloete

EyeRS project was a very good experience and helped me in different knowledge fields. Work with XML, enhance my knowledge in Java, and skills to work in a group. It was my first time to work with xml language, and I learned how to work with it and understand it a good part that helped me to contribute to the development of the app. I had previous knowledge on java. The project taught me about different areas of coding based on the app requirements. Working in a group is not easy but the members in this group made this experience manageable and enjoyable. I also learn communication skills and time management within this group.

The overall learning experience was great and I did not only learn things mention above, but also dealing with the customers and changes in the process of the development of the eyeRS app. - Emilde Arsenio

From the start of project all group members were willing and eager to work. Throughout the development of the project all group member chip in where they could and did so to the best of their abilities. As we developed the project we needed to learn how to use new software, namely the Android SDK, and develop soft skills. Over the past eight months my understanding of the development of Android apps, and software in general, has grown immensely. I have a new profound appreciation for the amount of documentation and other work that needs to happen



behind the scenes when developing a system as team. In conclusion, I feel that the development of the EyeRS app has ultimately been a success. Even if the system does not fulfil our expectations the lessons learned while developing the app are invaluable.

- Matthew Van der Bijl

1.6 Conclusion

Overall, the project was a success as the project deliverables were all completed on time.

The group chose to assign a strategic leader at the beginning of each deliverable, who would act as a 'scrum master', to ensure that the required tasks (for each deliverable) were communicated to other team members and carried out in an ordered manner, while taking into consideration the time and roles required by each member to complete the deliverable.

With regards to work input, the group performed well as each deliverable was completed to a standard that satisfied the client while surpassing their expectation.

With regards to time management, the group was above satisfactory for all deliverables which were submitted well before their respective deadlines.

The group displayed good communication skills, ensuring that each deliverables' scope and quality expected was well outlined and understood by each team member.

There was also constant encouragement and motivation which boosted the moral of the group as a whole when facing challenging tasks which drove team members to complete the tasks despite the odds.

With regards to the quality/scope of the work, the group produced a fully integrated and working app to the user's satisfaction which was a display of the exceptional work put in by each individual so as to ensure that the project was a success.