

Project Report Document

Inventory App

Student: Keith Mullins
Supervisor: Nigel Whyte
Student ID: C00118202

Contents

1. Introduction	3
2. Description of submitted project	3
3. Application Layout - Android Application	4
Login	4
Registration	5
Main Menu	6
Inventory List	7
Add / Edit Item	8
Category List	9
Export	10
4. Application Layout - Website	11
Landing Page	11
Login	12
Password Reset	13
After Login	14
View Inventory	15
Item Details	16
5. Conformance to Specification and Design	17
Amazon API & XML Parsing	17
Description of learning	17
Technical achievements	17
Personal achievements	18
6. Review of project	19
Advice for Others	19
Acknowledgements	20

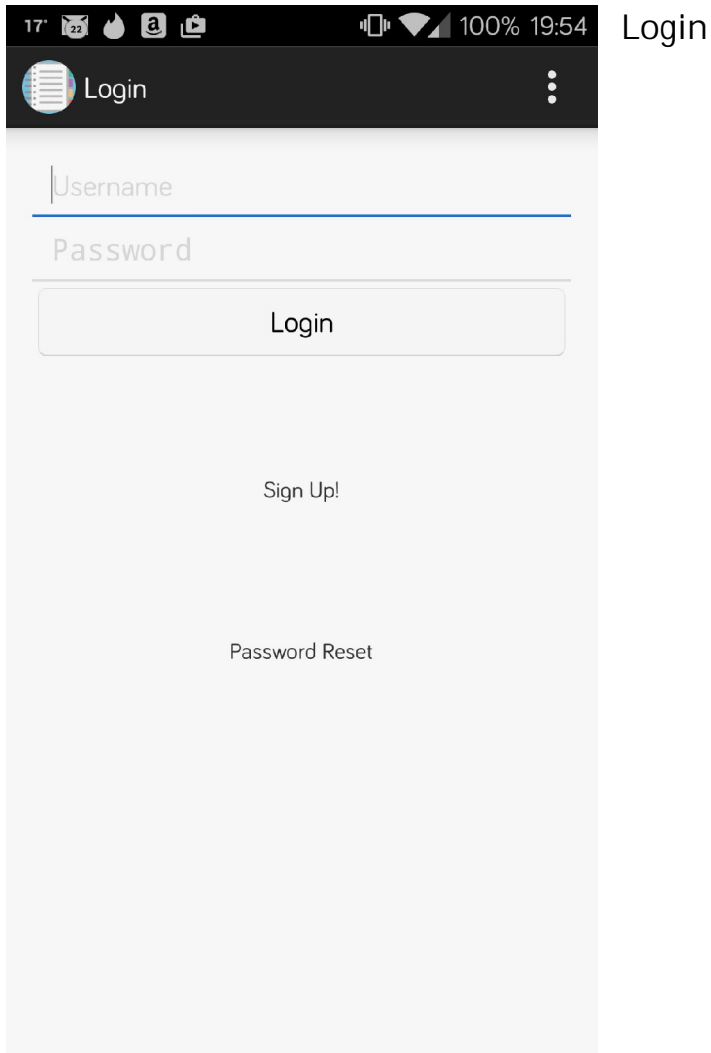
1. Introduction

The purpose of the project report is to take an overview of the final project and discuss the finished product, including what it does or does not do. Screen shots will also be provided to give an insight into what the app looks like. We will see if the project conforms to the original specification and design that was created. As well as the above, we will see what was learned throughout the app development process.

2. Description of submitted project

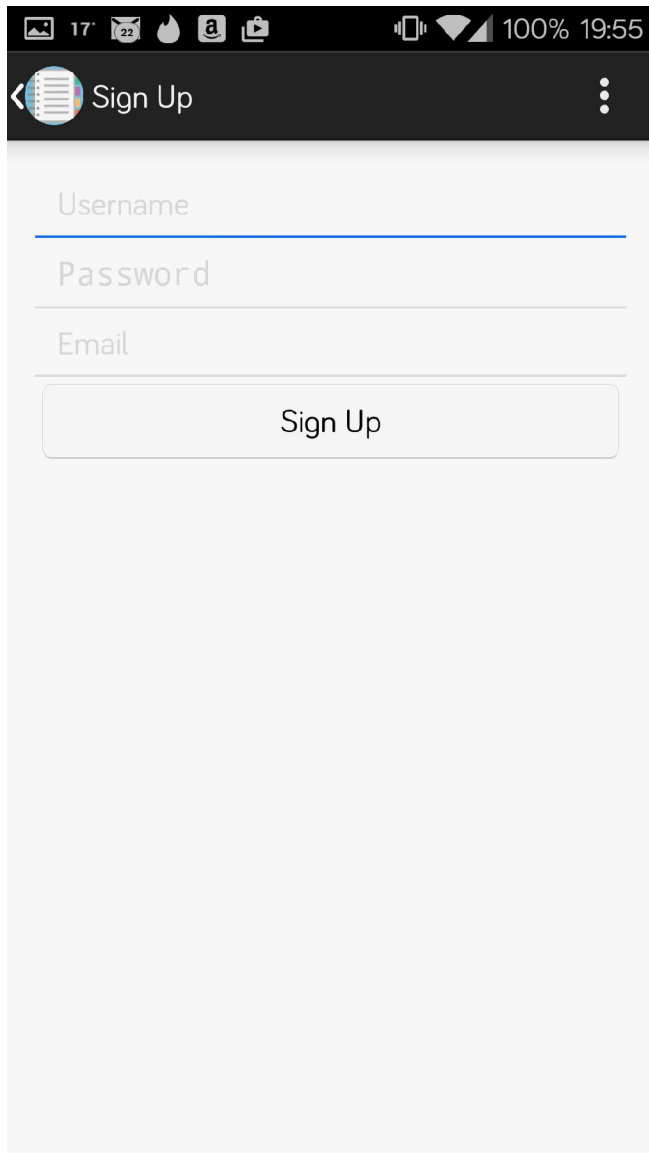
The project I chose required an app to be designed which would allow users of the app to create an inventory of their collection of books, cd's etc. This list of their collection allows them to keep a record in the case of an insurance problem so that they can verify what items they possess.

3. Application Layout - Android Application



This is the first screen that the user is presented with. It will allow them to login to the app provided that they have created an account. If it is their first time using the app and therefore have no account, they can go to the registration page via the sign up link. Various types of error checking is carried out to ensure that the user enters information that can be accepted i.e. no blank fields/incorrect password/username doesn't exist etc.

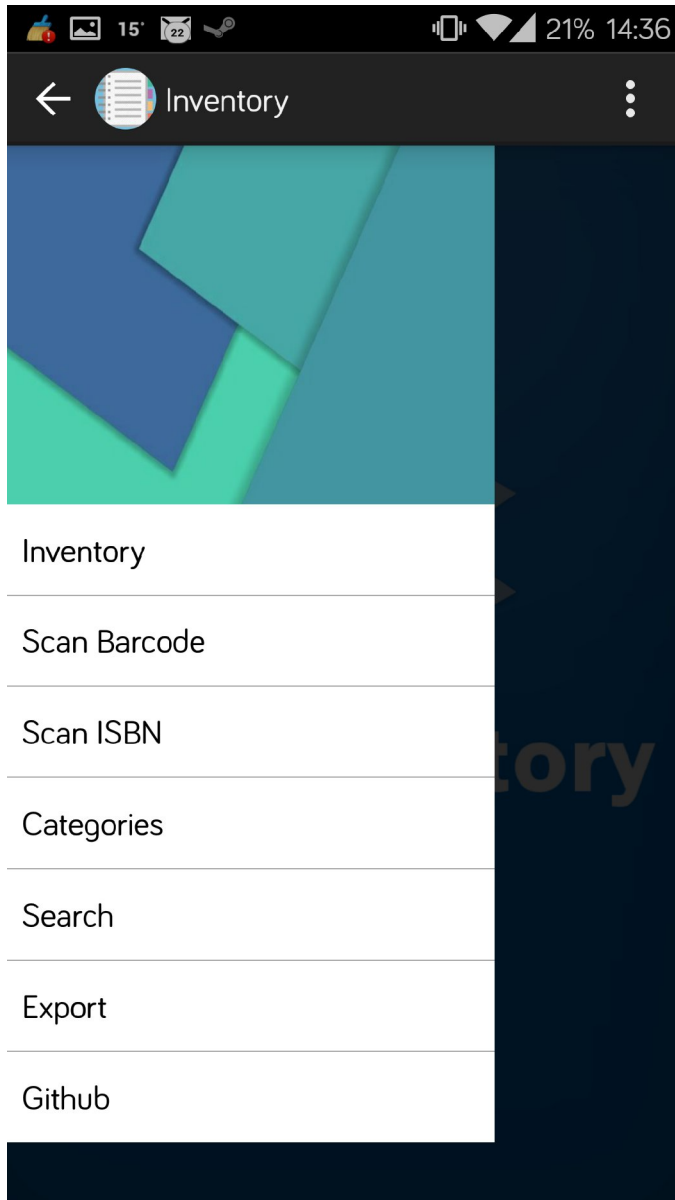
Registration



The screenshot shows a mobile application interface for registration. At the top, there is a dark navigation bar with a back arrow on the left, a menu icon, and the text "Sign Up". Below this bar, the screen is light gray and contains three input fields stacked vertically, labeled "Username", "Password", and "Email". Each field has a thin blue underline. At the bottom of the form is a rounded rectangular button with the text "Sign Up". The top of the screen shows a status bar with various icons and the time "19:55".

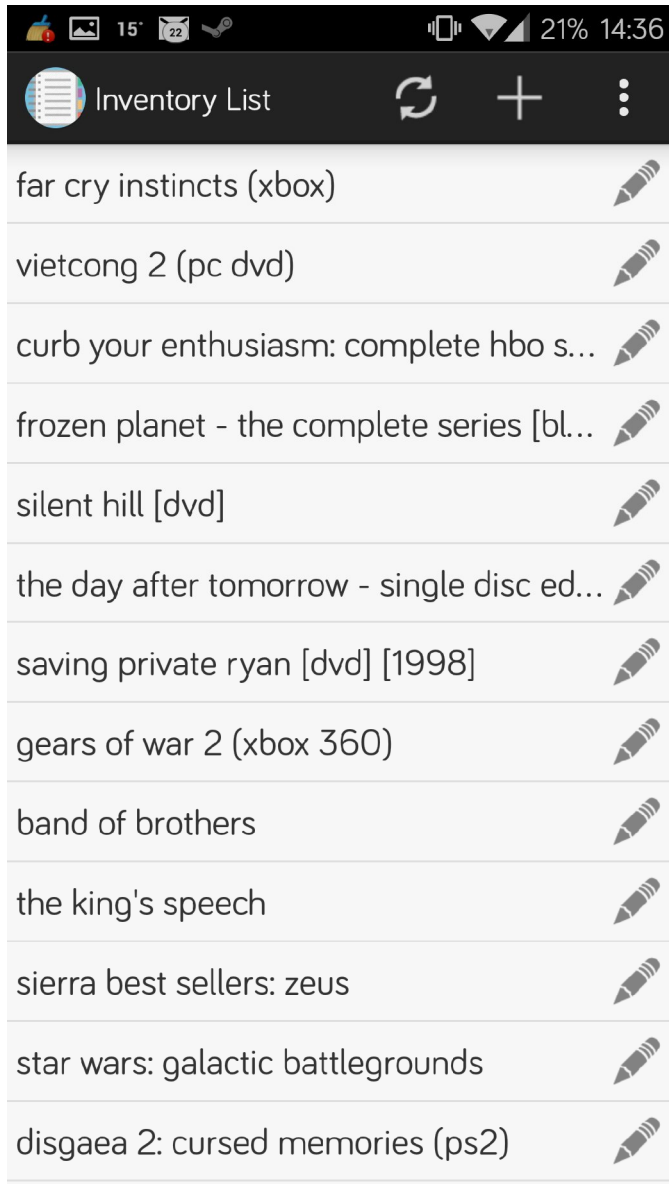
This is the screen that will allow new users to create their account in order to use the app. The user must enter a username, password and their e-mail address which is to be used for password resets.. Error checking includes checking for blank fields as well as ensuring that the username and email has not already been used. The account is created on the cloud once the user clicks sign up.

Main Menu



The main menu is based on the Material drawer that is used in a lot of high end apps such as Google Email, reddit is fun, boards.ie, Steam, tinder, etc. It is the go to design at the moment in android applications as it provides a clean and easy to navigate interface by hiding the menu items off screen until they are needed. From here the user can launch any of the options and gain immediate access to each of them.

Inventory List



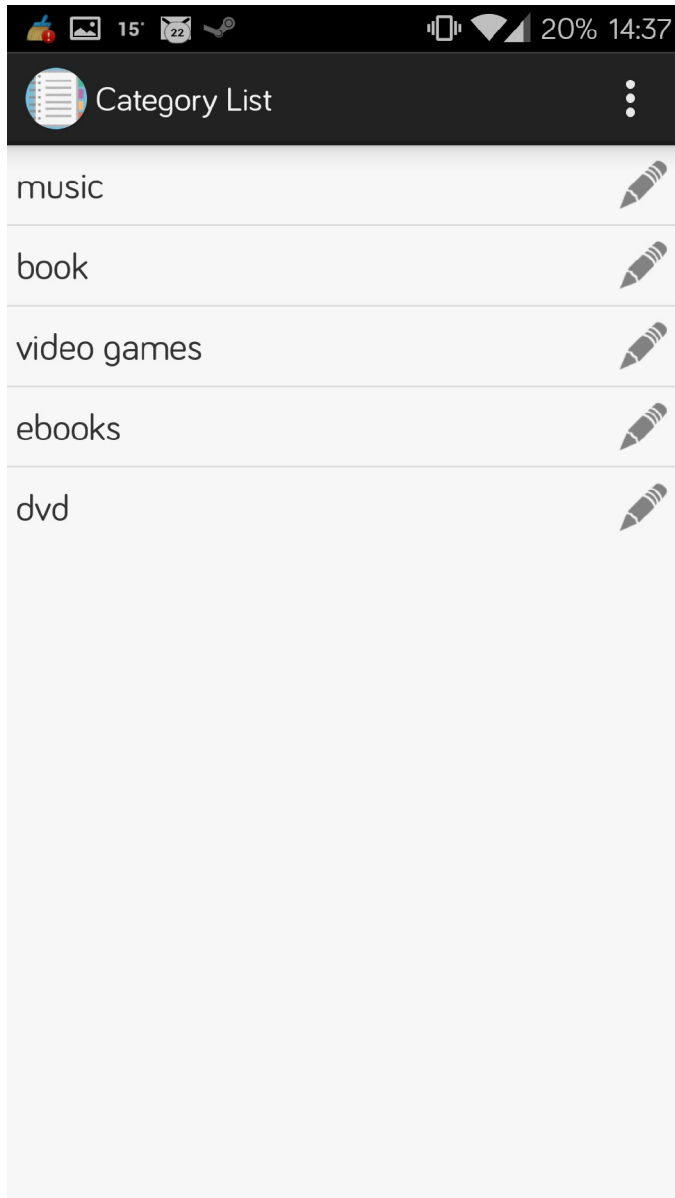
The inventory list is the list of items that are stored in the cloud server. All the titles are pulled down and stored in a listview in the application. Clicking on an item takes you to the add / edit page where the specific details of the item are loaded and can be viewed. You can also refresh the list of items using the swirling arrows and add an item by clicking on the “plus” sign on the action bar at the top. To delete an item you can long press on an item upon which a context menu will appear displaying the option the delete.

Add / Edit Item

The screenshot shows the 'Add / Edit Item' screen. At the top, there is a dark header with a hamburger menu icon, a list icon, the text 'Add / Edit', and a three-dot menu icon. Below the header, there are five text input fields: 'Title' (placeholder 'Item Title'), 'Description' (placeholder 'Item Description'), 'Category' (placeholder 'Item Category'), 'Author/Artist' (placeholder 'Author / Artist'), and 'Release/Publish Date' (placeholder 'Release / Publish Date'). At the bottom, there is a large empty rectangular box for an image, a camera icon button, an 'OCR' button, a 'Save' button, and a 'Cancel' button. The status bar at the top shows the time as 14:37 and 21% battery.

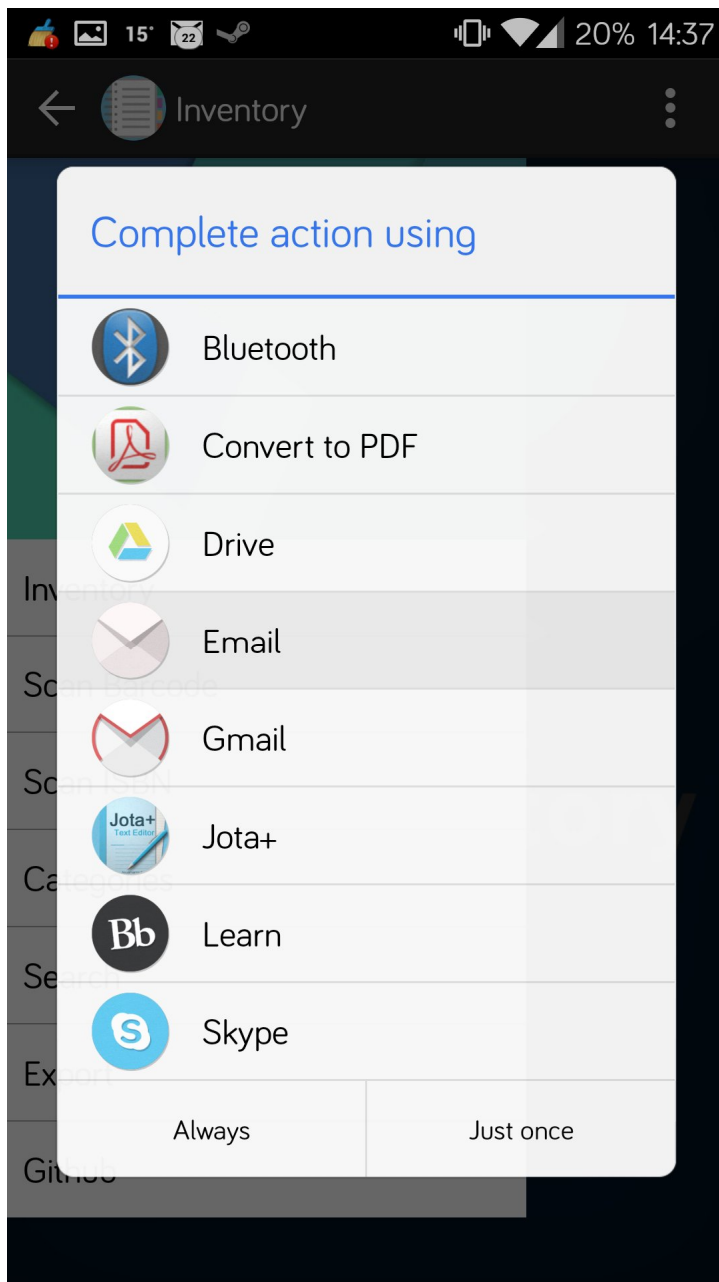
The add / edit screen holds the different values of the item that are pulled from the cloud server. This screen allows the user to edit the details of the item and then save them back up to the cloud. A picture of the item will be displayed in the imagebox near the bottom of the screen. The button with the camera icon allows the user to take a picture of the item which will overwrite the image of the item if it has been generated by Amazon using the barcode scanner. The OCR button will allow the user to capture the text on the back of the item, in the case of a book it could be the title, instead of manually typing it into the app. If the add / edit screen is launched from using the barcode scanner or the isbn scanner it will automatically generate the fields by doing a call to amazon based on the barcode or isbn and pull the relevant fields from an XML file and display them in the application.

Category List



The category list simply lists the categories of items available so that the user can quickly see which items belong to which category.

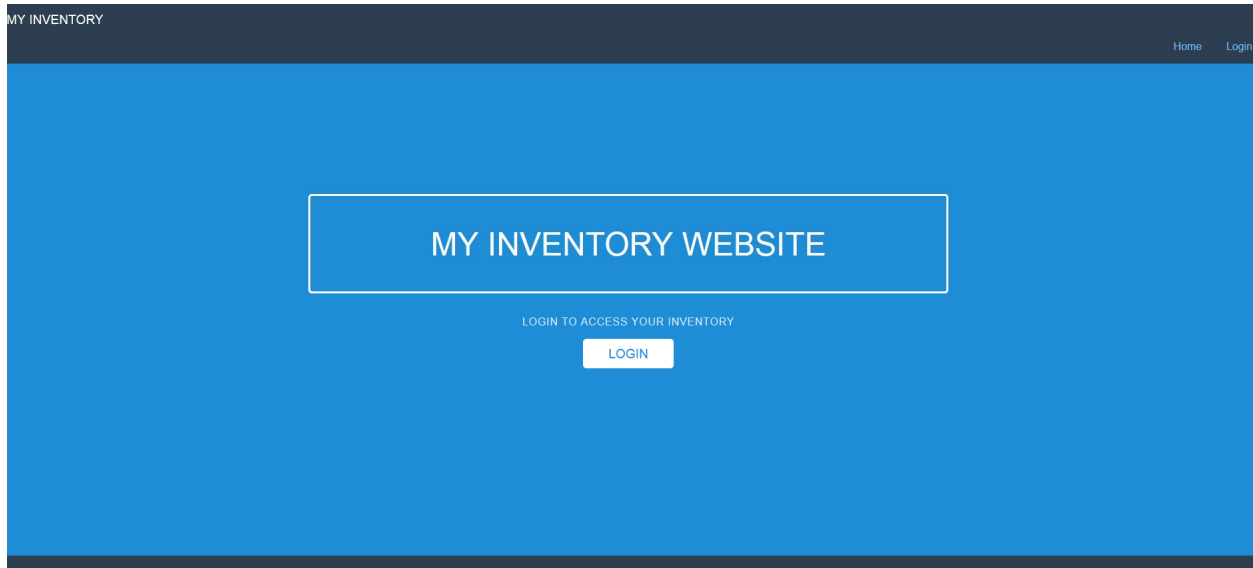
Export



When the user presses the export button a CSV (comma separated value) file is generated containing all the items details in a clean format that can be viewed on their computer in an Excel or similar viewer. This will allow them to keep a backup on their own computer. The file can be attached to an email or uploaded to their respective hosting account e.g. google drive

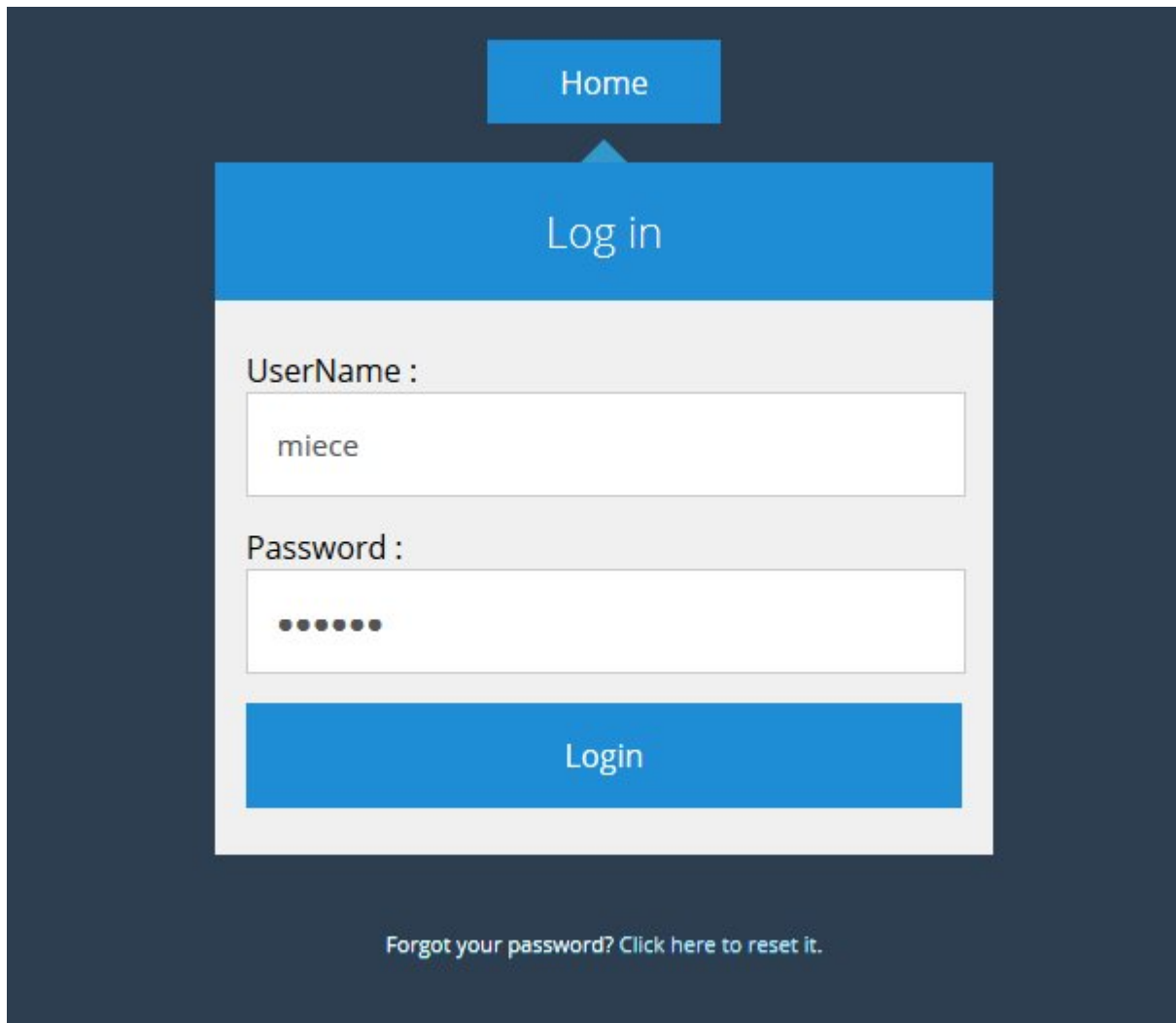
4. Application Layout - Website

Landing Page



This screen is the landing page for the website which will allow the user to login to the website to view their collection.

Login

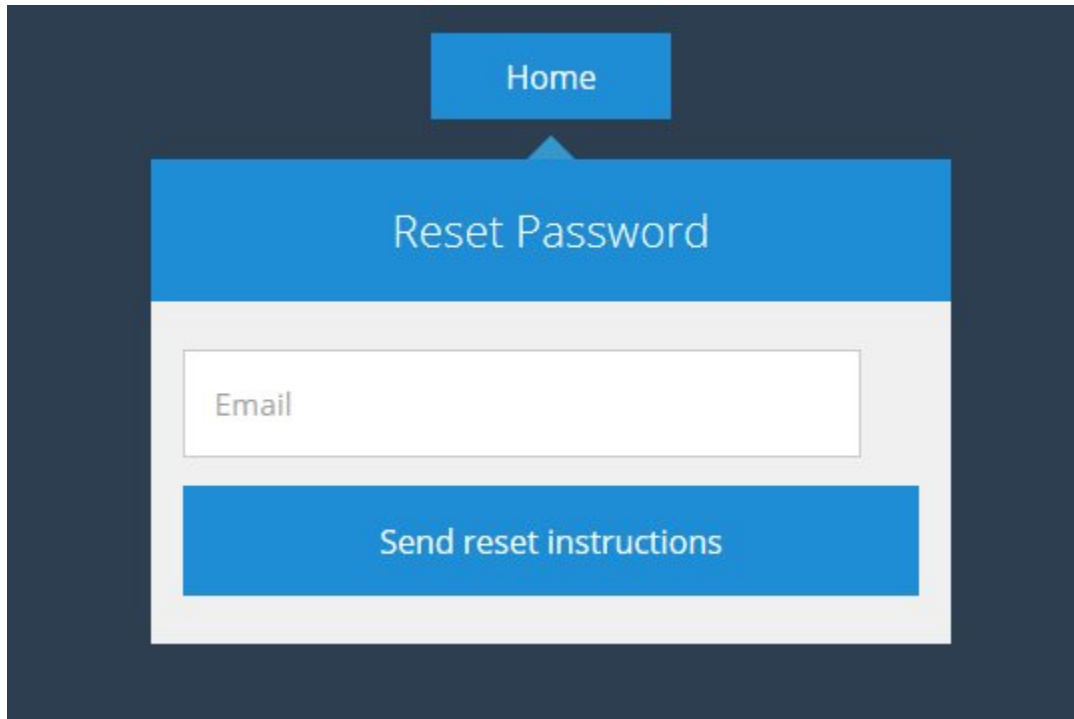


The image shows a login interface on a dark blue background. At the top center is a blue button labeled "Home". Below it is a white box with a blue header containing the text "Log in". Inside this box, there are two input fields: the first is labeled "UserName :" and contains the text "miece"; the second is labeled "Password :" and contains seven black dots. Below the password field is a blue button labeled "Login". At the bottom of the white box, there is a link that says "Forgot your password? Click here to reset it."

This is the login page the user sees when they try to login.

The website is part of the application in that a user cannot create an account on it but must do so in the android app to be able to login here. A non registered user with the app will not be allowed to login.

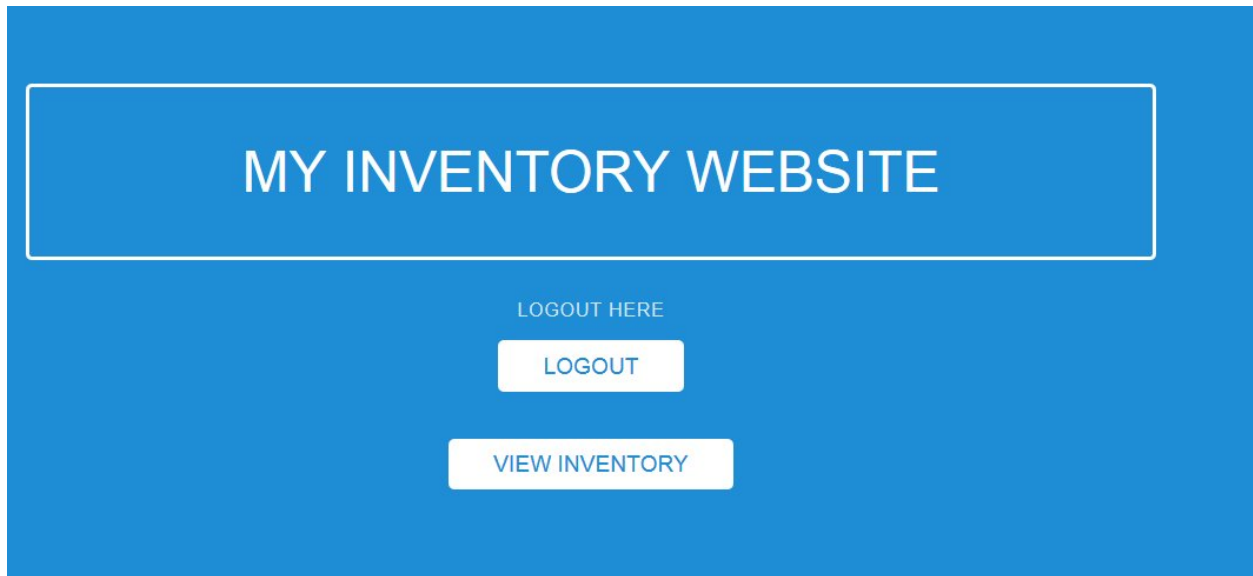
Password Reset



The image shows a user interface for a password reset process. At the top, there is a blue button labeled "Home". Below it is a large blue header with the text "Reset Password". Underneath the header is a light gray rectangular area containing a white input field with the placeholder text "Email". Below the input field is a blue button with the text "Send reset instructions".

This is the password reset for the website. It works in similar fashion to the android app in that it sends an email to the user allowing them to reset their password.

After Login



After the user has logged into the website they will they be granted access to their inventory collection through the appearance of the view inventory button.

View Inventory

- **band of brothers**
- **chocolate starfish and the hot dog flavored water [special uk edition with bonus cd]**
- **curb your enthusiasm: complete hbo season 3 [dvd] [2005]**
- **digital fortress**
- **disgaea 2: cursed memories (ps2)**
- **dragon age inquisition (xbox one)**
- **far cry instincts (xbox)**
- **frozen planet - the complete series [blu-ray]**
- **gears of war 2 (xbox 360)**
- **saving private ryan [dvd] [1998]**
- **sierra best sellers: zeus**
- **silent hill [dvd]**
- **star wars: galactic battlegrounds**
- **the day after tomorrow - single disc edition [2004] [dvd]**

1 | [2 Next »](#)

The view inventory screen of the website will display a paginated list of items (14 per page) Clicking on an item will bring the user to the view item details page.

Item Details

Item Details

Title :

band of brothers

Description :

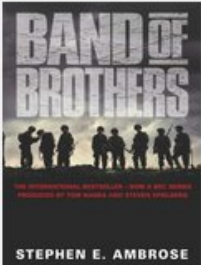
they fought on utah beach, in arnhem, bastogne, the bulge; they spearheaded the rhine offensive and took possession of hitler's eagle's nest in berchtesgaden. easy company, 506th airborne

Category :

ebooks

Author / Artist :

ebooks



Update

Delete

[Go back to inventory.](#)

This screen shows the details on the item and allows the user to edit them or if they choose, they can delete the item using the delete button. A alert box will inform the user if they are sure they want to delete the item before doing so.

5. Conformance to Specification and Design

The final design conforms to what I have already thought out in previous documents.

The original GUI that was displayed in the functional specification was only a prototype and not very interesting to use, however I have gone about the final design very similarly and not strayed too far from the original design.

The material drawer design that I chose to use for the menu is a very popular theme at the moment with many prominent apps using said design.

Listviews are used as a way of neatly displaying data to the user such as the item list and categories list etc. The screens remain uncluttered which also makes for easier viewing on mobile devices. It was decided based on use of other inventory apps that a light colour scheme should be used in order to improve readability; however the dark blue scheme used in the web app didn't pose any problems in this regard.

Amazon API & XML Parsing

When a user scans using the barcode scanner in the app a call is made to the Amazon Product Advertising API . Originally the Product Advertising API provides access to Amazon's product selection and discovery functionality so that developers can advertise Amazon products to monetize a website, but instead of using it for that I used it to pull details of items that I can store within the cloud and display them in my app. The data is sent back from amazon in an XML file which then had to be parsed using one of the java XML parsers such as TagSoup or XSoup. Each element houses specific nodes in the XML file e.g. the element <ItemAttributes> can hold the name, description, price nodes etc. of an item. Then each node has its attribute i.e. the string associated with the node. Pulling this attribute from a specific node i was able to get the title of items, price, etc. which is then displayed in my app and saved to the cloud.

Description of learning

Technical achievements

Prior to the beginning of this project I had no previous knowledge about android apps and the various frameworks that are available for developing web apps, with only some knowledge of some of the various scripting languages such as PHP and Javascript due to my time working with those languages during my work experience

I learned how an application can manipulate all of the underlying hardware features that the Android device has at its disposal.

I learned a great deal about how to parse XML files and how to process and manipulate data to get what I want out of the file.

I also began looking at JavaScript / PHP which is a vital part of creating any web app. Javascript can be a difficult language to learn as it sometimes does a lot of things that you don't expect it to do and the syntax is quite difficult to understand. I had thought about working with Json and Ajax as they can allow for web apps to behave much more dynamically, such that allow it to pass data to the page and display it without needing to refresh the browser.

Personal achievements

I feel that i have achieved quite a lot of work and that the app is quite advanced state and would be ready for release on the app store as a fully working item. However I still feel it could do with a bit more touching up, i.e. in the UI side of things, but in the end I am not an interface designer so that side of things has let me down.

It was a great opportunity to get exposure to all of these technologies.

I also would have never been comfortable working with Array Lists or Hash Maps but because I had to use them in my project I would not have a problem working with them again.

6. Review of project

Overall I'm quite happy with how the project turned out. At the beginning I had trouble trying to learn how Android programming works such as the use of activities and the passing of data and the whole structure of an android app. It felt a bit daunting as I wasn't really getting the grasp of it, but I continued to look up different resources and different tutorials that helped explain how the different parts work together and eventually I was able to use my programming experience to apply to what I needed to do into the application. Working with a cloud provider for the back end proved to be no problem as Parse provides excellent documentation when it comes to explaining how to connect to their API , using their SDK and saving data to the cloud.

Although I had been reading up on JavaScript and JSON I had some difficulty understanding how to use them so I decided not to use them as part of my project for the webapp side. I stuck with PHP, as the cloud provider Parse have a very good SDK and documentation and also the fact that I spent 6 months work experience working on PHP helped me to make that decision.

The functionality that I got working in the android app does work fine on mobile devices as well as the webapp on the PC, however the visual side of things could be improved in both.

If I were to start the project all over again the first thing I would I do differently is really study how Android works and learn the UI side of an app first. Although it is quite easy to change small features of the UI such as adding buttons, a full redesign such as using a material drawer does include some major code changes and did delay the project for about 2 - 3 weeks.

In the end I believe I did choose the right technologies for developing my app. Android is a massive market and will continue to grow in the coming years.

Advice for Others

I would advise them to spend a lot of time researching the technologies that they are going to use because all of the technologies that I researched thoroughly were excellent. They really worked well together as I put effort into making sure that they would work before spending days trying to get them to work when they might not and waste my time.

The one technology that I had the most trouble with was the OCR and the material drawer design. The reason for this is because I had never even heard of it while I was researching OCR engines. I discovered it late into the development of the project.

The reason I decided to use it was that I could see its potential of how easy it would be to use. If I had have spent the time to researching the OCR engine I feel that I could have prevented around 2-3 weeks of delays during the development of the project. The material drawer took up a lot of time as I spent quite a while trying to integrate different versions into my app of the drawer menu and many of them didn't want to work.

Acknowledgements

I would like to thank my supervisor Nigel Whyte for his help and guidance during the development of the Inventory App.

Robert Theis (<https://github.com/rmtheis>)

for the use of his modified tess-two OCR engine and optical character recognition on Android project.

ZXing (<https://github.com/zxing/zxing/>) (<http://zxing.org/w/decode.jspx>)

for the use of their open source barcode scanner without which most of my project would not be possible

Stackoverflow (<http://stackoverflow.com/>)

for the multiple answers to the infinite number of Java/Android/PHP.. etc. questions that I had.

Parse (<http://www.parse.com>)

for providing the cloud backend for my application with an easy to use interface and SDK & documents

Amazon (www.amazon.co.uk)

for being able to use their API to access product attributes.

The Android Arsenal (<https://android-arsenal.com/>)

for providing free design elements for Android

And all the various other websites and tutorials that I have forgotten to mention that have shown me how to become a better programmer.