1 Introduction

Every year the University of East Anglia hosts an important event known as 'Open Day'where potential students and applicants visit the university to experience its schools and facilities first hand. As a group, we have been assigned the task of developing three stand-alone android applications that will provide useful functionality to support visitors of the UEA open day. All three applications will be developed using the 'Simple Android Application Framework'(also known as SAAF); as a group we will modify and update the SAAF with relevant functionality which can be used by each application. Throughout the application development stage, it has been unanimously agreed that we aim to take advantage of iterative/agile methodologies. Furthermore, Extreme Programming (also known as XP) and Scrum tactics will be deployed as an influence for continuous communication between all group members throughout the process. Along-side these methodologies, all the group members will use Trello and GitHub to organise, storing and allow easy collaboration between group members.

2 Android Application Identification

This section of the design document will describe each of the three android applications which the group aims to develop throughout the project. Each application has its own problem set that needs to be addressed, furthermore defining the application functionality at this stage will provide clarification during later development.

2.1 Room Finder Application

This application consists of a location finder and route planner based on the UEA/CMP campus map. Once the application has been initialised the user is able to enter a location within the campus, the user is then presented with instructions directing them to the desired location. The target audience for UEA open day will consist of a very broad range when considering applicants and their various family members. For these reasons there will two view types in order to compensate for how familiar each individual visitor is with smartphone applications. The first view type is a complex view which shows various information regarding the current location of the user, the desired location and further relevant information. The second view type is a simple view, displaying much less information. If the route is broken down into smaller stages, each stage can be shown individually and therefore the application becomes more user-friendly. Additionally, once the user has found the room/location or is within a set radius, the user will be prompted to confirm that they have been successfully directed to the specified destination, this could be confirmed by scanning a NFC tag found at the location.

2.2 Car Finder Application

For many visitors, it will be their first time parking on the UEA campus. This application will allow for visitors to set the location of their vehicle at the time of parking. Once the user would like to leave or return to their vehicle, the application will retrieve the previously entered location and provide information regarding the location of the vehicle. Once again there two view types, a simple type and a complex type, to ensure the application is user friendly; the simple view will also consist of a directional arrow within this application. The car finder application will also record and provide additional relevant information such as the time of parking.

2.3 Augmented Reality Building Scanner

Within the development of these applications, offering useful functionality to the users is main objective however, as these applications are aimed to be used at UEA open day as a group we decided that the final application should both offer functionality and impress/inspire applicants and their relatives who are

taking part in the UEA open day; augmented reality is a concept that is not often seen and therefore can be perceived as impressive. The application will use the image capture device on the users smart phone device, once the user taps the screen the application uses the GPS coordinates and direction of the user to determine which campus building the user has his/her device pointed towards. The user is then presented with information on top of the live image capture with relevant information/functionality.

3 Requirements Analysis

3.1 General Assumptions

Despite that there are three individual applications to develop, there is common ground which can be found between the requirements analysis of all three applications. Below are the general assumptions which apply to each application being developed within this project:

- Each application will be developed for the Android platform
- Each application will be based on the Simple Android Application Framework
- Each application is a stand-alone application with its own icon
- Each application must take into account multiple resolutions and form-factors
- Applications are free to targeted towards mobile platforms - phones, tablets or both.
- Each application will have access to the internet
- Each application will be developed using Android 4.2.2 (Jellybean)

3.2 Minimum Software Development Kit

In order to target as wide a range of end-users as possible, we have chosen to use Android version 4.2 (Jelly Bean), as it contains all the features we need and might want to use (NFC capability, GPS and Google Maps as standard, secure USB debugging, active download notifications, idle/lock display, etc.) as well as targeting an estimated 73.1 percent of all active Android handsets (data according to Google).

3.3 Target Software Development Kit

In order to ensure forwards compatibility as well as making sure we are not using deprecated code, we will be targeting the latest version of Android (Marshmallow). Despite only being released on a relatively few number of handsets, this will future-proof the applications, allowing them to be used in any number of future open days.

3.4 Application-Specific Assumptions

The next step is to identify assumptions that can be made for each application specifically. These assumptions may reiterate on general assumptions mentioned previously or be completely individual to a single android application.

3.4.1 Room Finder Application Assumptions

- The user's starting location is within the UEA campus
- The user's desired location is within the UEA campus
- The user has a smartphone device which allows NFC tag functionality

- The user has allowed for access to GPS coordinates
- The user has allowed for access NFC tag reader
- The user has knowledge/understanding of at least one of the supported languages
- The application language will be set to that of the device if supported

3.4.2 Car Finder Application Assumptions

- The user has parked their car within the UEA campus
- The user sets the car location whilst within a certain distance of the parked vehicle
- The user has allowed access to GPS coordinates
- The user has knowledge/understanding of at least one of the supported languages
- The application language will be set to that of the device if supported

3.4.3 Augmented Reality Building Scanner

- The user has a smartphone device which allows for image capture
- The user has allowed access to the GPS coordinates
- The user has allowed access to the image capture capacities of the device
- The user has knowledge/understanding of at least one of the supported languages
- The application language will be set to that of the device if supported.

3.5 Similar-System Analysis

Analysing systems which have a similar functionality or purpose may prove to be useful during the development of the android applications. The aim of this research is to identify key components and procedures of these similar systems, therefore potentially allowing the team address any issues or requirements generally gain a greater understanding of each application. Due to the fact that all of our desired applications have an overlap in functionality, the following systems that will be analysed are relevant to all three android applications mentioned previously.

3.5.1 App 1: Find My Car

The purpose of this application is to provide users a way to mark where they have parked their car (via GPS) and provide directions back to the car via google navigation.

Pros:

- Provides compass functionality to help users find their car if they do not have access to an internet connection
- Allows the user to take a picture of the parking spot to save alongside the GPS coordinates
- Allows the user to share their parking location with friends via email

Cons:

- Users must donate to remove advertisements
- The application has a dated UI design
- There is no way to deleted previously save parking locations

3.5.2 App 2: Get Point - GPS

The purpose of this application is to provide users a way to mark where they have parked their car (via GPS) and provide directions back to the car via google navigation.

Pros:

- Provides compass functionality to help users find their car if they do not have access to an internet connection
- Allows the user to take a picture of the parking spot to save alongside the GPS coordinates
- Allows the user to share their parking location with friends via email

Cons:

- Users must donate to remove advertisements
- The application has a dated UI design
- There is no way to deleted previously save parking locations

3.5.3 App 3: Google Maps

The purpose of this application is to provide users a way to search for and find directions to specific locations such as businesses, houses, landmarks etc. The application also provides a five star rating for the location if it?s a business (provided by users) and text reviews of the experiences users have had in these places.

Pros:

- Provides specialised routes depending on if the user is driving, biking, walking or using public transportation
- Provides public transportation information such as bus and train times
- Users are able to search for a location by address, business name, post code etc.
- Provides an estimated duration for the user's journey

Cons:

- Tougher accessibility for street view compared to the website –users must drop a pin on the map and then switch to street view, this is not as quick and intuitive as drag and drop
- Maps do not cover every country
- Gives a limited amount of detail in places (for example, individual UEA buildings are not marked on the map)

3.5.4 App 4: Metro AR Pro

The purpose of this application is to provide users a way to find the nearest train station within 5km by looking through the camera of their phone to see which direction it is in.

Pros:

- Works in multiple countries
- Provides the user with the distance to the station

- Simple, easy to use UI

Cons:

- Does not give detailed information such as train times
- Does not provide the user with directions to their chosen station or the estimated journey duration

3.5.5 App 5: Yelp Monocle

The purpose of this application is to provide users a way to leave and search for reviews of local businesses. Its 'Monocle'feature is an augmented reality application that consists of the user looking through their camera and reviews for local businesses appearing when the user points their camera at them.

Pros:

- Allows users to attach pictures of businesses to their reviews
- Extra information about businesses are given (such as opening hours, menus, etc.)

Cons:

- Hidden feature of the 'Yelp app'. Users must go into the 'More'section of the app to find Monocle
- Use of AR is more of a gimmick, it brings up information. The user could use the text box to search for information.

3.6 Use Cases

3.6.1 Room Finder Application

SEARCH FOR A LOCATION
Goal In Context: To allow the actor to search for a room at the university and show them directions towards it from their current location.
Preconditions: System is running and active, device is connected to the internet and is capable of GPS.
Success End Condition: The actor is shown the location of the room they selected on a map and is given directions towards it.
Fail End Condition: The actor cannot find their desired room or their current location cannot be found. In the case of the later, an error message is displayed.
Primary Actor: Application User.
Trigger: Actor presses the Find Room button on the GUI.
Priority: Top Priority
Performance: The actor?s target location should be displayed within 10 seconds.

Success Scenario
Step:
1: Actor presses Find Room button on main application activity
2: A window or activity is brought up with 3 drop down menus; the building (e.g SCI), the floor (e.g 01) and the room (e.g 17)
3: Actor presses the Submit button
4: Actor is presented with a map from their current location to the correct building along with directions on how to access the correct floor and room

Alternative Scenario
Step:
4a: The current location of the actor cannot be found

4b: An error message is displayed and the actor is just given the location of their desired building marked with no directions

ROOM FOUND CHECK

Goal In Context: To automatically notify the actor that they have reached their destination
Preconditions: System is running and active, the actor has a destination set and their current location can be obtained, device is connected to the internet and is capable of GPS
Success End Condition: The actor is given a notification that they?ve reached their destination building/room.
Fail End Condition: The actor does not reach their destination
Primary Actor: Application User
Trigger: Run repeatedly after a certain time interval after a destination is set
Priority: Top Priority
Performance: The actor should be notified that they are at or close to their destination within 5 seconds of entering the correct range

Success Scenario

Step:
1: Actor sets a destination
2: Application periodically compares actors current coordinates with the coordinates of the destination
3: If the user is within a certain radius of the destination, the user is alerted

Alternative Scenario

Step:
2a: The current location of the actor cannot be found
2b: The check is not performed until the actor?s current coordinates can be ascertained

GET ROOM INFORMATION

Goal In Context: To allow the actor to tap their phone against an NFC tag outside a room to bring up information on their phone about the room
Preconditions: System is running and active, user has a NFC capable device
Success End Condition: The actor is shown information about the room corresponding to the NFC tag tapped
Fail End Condition: The actor is not shown information about the room, an error message is displayed
Primary Actor: Application User
Trigger: Actor presses the Get Room Info button on the GUI and taps their device against a compatible NFC tag
Priority: Medium Priority
Performance: The actor should be given information about the room within 5 seconds

Success Scenario

Step:
1: Actor presses Get Room Info button on the GUI
2: A message is given to the user prompting them to tap their device against an NFC tag
3: Actor taps their device against a compatible NFC tag
4: The application displays text information about the room in question

Alternative Scenario

Step:
3a: The actor taps their phone against an incompatible NFC tag, an error message is displayed and the user is prompted to tap their phone again or go back to the main screen

FIND CURRENT LOCATION

Goal In Context: To allow the actor to find out which building they are currently located within
Preconditions: System is running and active, device is connected to the internet and is capable of GPS
Success End Condition: The actor is shown the name of the building they are currently in and its position

on a map
Fail End Condition: The actors current location cannot be found, an error message is displayed
Primary Actor: Application User
Trigger: Actor presses the 'Where am I'button on the GUI
Priority: Top Priority
Performance: The actors current location should be displayed within 10 seconds

Success Scenario
Step:
1: Actor presses Where am I button on main application activity
2: The actor?s location is retrieved via GPS
3: Application switches to a new activity with the actor?s current location on a map and any information about the building they are currently in (if they?re inside one)

Alternative Scenario
Step:
4a: The current location of the actor cannot be found
4b: An error message is displayed and the actor is returned to the first screen

CHANGE VIEW
Goal In Context: To allow the actor to swap between simple and complex view types
Preconditions: System is running and active, actor has an active internet connection and has also selected a room to navigate to
Success End Condition: The application switches from the simple view to the complex view or vice versa
Fail End Condition: The application does not switch between the two view types
Primary Actor: Application User
Trigger: Actor presses the Change View button on the GUI
Priority: Medium Priority
Performance: The application view type should be switched within 5 seconds

Success Scenario
Step:
1: Actor presses Change View button on the room finding GUI
2: The display of the route is switched to simple or complex view

3.6.2 Car Finder Application

SET CAR LOCATION
Goal In Context: The actor presses the button and his/her location is retrieved and stored representing where his/her vehicle has been parked
Preconditions: Application is live and internet access is available
Success End Condition: The application will have the information needed to track the users vehicle, allowing for the other application functionality to perform
Fail End Condition: The application failed to retrieve or store GPS location of the user?s vehicle
Primary Actor: Application User
Trigger: Get Location button is pressed by user
Priority: Top Priority
Performance: Retrieves information within 5 seconds of the user pressing Get location

Success Scenario
Step:
1: Application is open and running, user presses Get Location button
2: Message is displayed stating that the app is searching for GPS coordinates
3: GPS is retrieved successfully and a message is displayed making the user aware

Alternative Scenario

Step:

3a: GPS retrieval fails and a message is displayed to the user with relevant information regarding the issue that occurred

FIND CAR LOCATION

Goal In Context: The actor presses the find car button and information such as direction of the car is presented to the user

Preconditions: Application is live, internet access is available and the user has already chose to store GPS information

Success End Condition: The application will have provided the user with information needed to locate the car, e.g. direction and distance

Fail End Condition: The application failed to give correct information regarding the location of the car.

Primary Actor: Application User

Trigger: Find Car button is pressed by user

Priority: Top Priority

Performance: Provides information within 5 seconds of the user pressing Find car location

Success Scenario

Step:

1: Application is open and running, user presses Find Location button.

2: Message is displayed stating that the app is processing information.

3: Information is displayed successfully leads user towards vehicle.

Alternative Scenario

Step:

3a: GPS information displayed is incorrect and the application attempts to try again.

CHANGE VIEW

Goal In Context: The actor presses the change view button, swapping between the simple and complex view types

Preconditions: Application is live, internet access is available and the user has already been given information on how to find the vehicle

Success End Condition: The application will change from simple to complex view or vice versa

Fail End Condition: The application failed to change between views and retain correct information

Primary Actor: Application User

Trigger: Change View button is pressed by user

Priority: Medium Priority

Performance: Changes information view within 5 seconds of the user pressing change view button

Successful Scenario

Step:

1: Application is open and running, user presses Change View button

2: Information display is changed to alternative view

CAR FOUND CHECK

Goal In Context: Once the GPS location matches or is close enough to the GPS location stored it prompts the user to confirm that they have reached their vehicle.

Preconditions: Application is live, internet access is available and the user has already stored GPS information and followed instructions on how to locate the vehicle

Success End Condition: The application will have completed and waiting for next entry of GPS location information

Fail End Condition: The application failed to direct the user to their vehicle

Primary Actor: Application User

Trigger: Current GPS location is close or matches vehicle GPS location

Priority: Medium Priority
Performance: Prompts user for confirmation within 5 seconds of the GPS locations being close

Success Scenario
Step:
1: Application is open and running, users location is close to the vehicle location
2: The user is prompted to confirm that they are near to the vehicle GPS location
3: The user confirms that they have found the location of the vehicle
Alternative Scenario
Step:
3a: The user has not found the location of the vehicle, informs the application

3.6.3 Augmented Reality Building Scanner

GET BUILDING INFORMATION
Goal In Context: To provide the user with the information found about the rooms of the location
Preconditions: Application is running and active, GPS is active and internet connection available, nearest location found, waiting for user to tap on the room info button.
Success End Condition: Information about the rooms in the location is shown to the user hovering over the camera feed
Fail End Condition: Information cannot be retrieved, user is shown an error message and directed back to the information screen
Primary Actor: Application User
Trigger: Actor tapping on the room info button.
Priority: Medium Priority
Performance: Information about the rooms of a given location should be retrieved within 5 seconds

Success Scenario
Step:
1: The application displays the information screen
2: The actor taps on room information button.
3: Application retrieves the information and displays it over the camera feed to give an A.R. feeling
Alternative Scenario
Step:
3a: Application fails to retrieve the information
4: User gets an error message and is directed back to the information screen
SCAN ENVIRONMENT
Goal In Context: To find the nearest location to the user and provide them with information about that location.
Preconditions: Application is running and active, GPS is active and internet connection available, user is using the scan menu, waiting for input from the user to begin.
Success End Condition: Nearest location has been found and user is directed to theinformation menu
Fail End Condition: Location cannot be found, error message is displayed and prompted to try again
Primary Actor: Application User
Trigger: Actor tapping on the screen.
Priority: High Priority
Performance: Nearest location should be found within 5 seconds

Success Scenario
Step:
1: The application displays a window that has the camera feed

2: The actor taps on the screen to scan
3: Application finds the nearest location and directs the user to the information page

Alternative Scenario

Step:
3a: Application fails to find the nearest location.
4: User gets an error message and is asked to try again

SEEK HELP

Goal In Context: To help the user understand how the application is used.
Preconditions: Application is running and active, waiting for user to tap on the help button.
Success End Condition: Help screen is displayed.
Fail End Condition: Help screen cannot be displayed, user is shown an error message and directed back to the information screen.
Primary Actor: Application User
Trigger: Actor tapping on the help button.
Priority: Medium Priority
Performance: Help screen should pop up and be ready in 5 seconds

Success Scenario

Step:
1: The application displays the information screen
2: The actor taps on the help button
3: Application directs the user to the help screen

Alternative Scenario

Step:
1: Application fails to open the help screen
2: User gets an error message and is directed back to the information screen

SHARE LOCATION

Goal In Context: To let the user share their location on social media
Preconditions: Application is running and active, GPS is active and internet connection available, nearest location found, waiting for user to tap on the share button
Success End Condition: Social media screen is displayed.
Fail End Condition: Social media screen cannot be opened, user is shown an error message and directed back to the information screen
Primary Actor: Application User
Trigger: Actor tapping on the room info button
Priority: Medium Priority
Performance: Social media screen should pop up and be ready in 5 seconds

Success Scenario

Step:
1: The application displays the information screen
2: The actor taps on room share button
3: Application directs the user to the social media screen

Alternative Scenario

Step:
3a: Application fails to open the social media screen.
4: User gets an error message and is directed back to the information screen

3.6.4 Use Case Diagrams

A use case diagram is a simplified way of representing and summarising the interactions between specified actors and the system. It details the steps taken when an actor wishes to perform a specific action in the system.

The use case diagram below depicts; The primary actor interacting with the system to set the geographical coordinates of a parked car. It is a crucial component of the applications car finding system as this gives the user a destination to travel to. The primary actor interacting with the system to receive a route to the coordinates they set as the location of a parked car. It is a crucial of the applications car finding system as it gives the user the location of their parked car as well as directions towards it. The periodical checking of the primary actors proximity to their parked car as well as the act of notifying them when theyre at or within close proximity of their parked car. This is a part of the applications car finding system as it lets the user know if they have reached their destination. The primary actor interacting with the application to switch between simple and complex views of the directions showing them to the car.

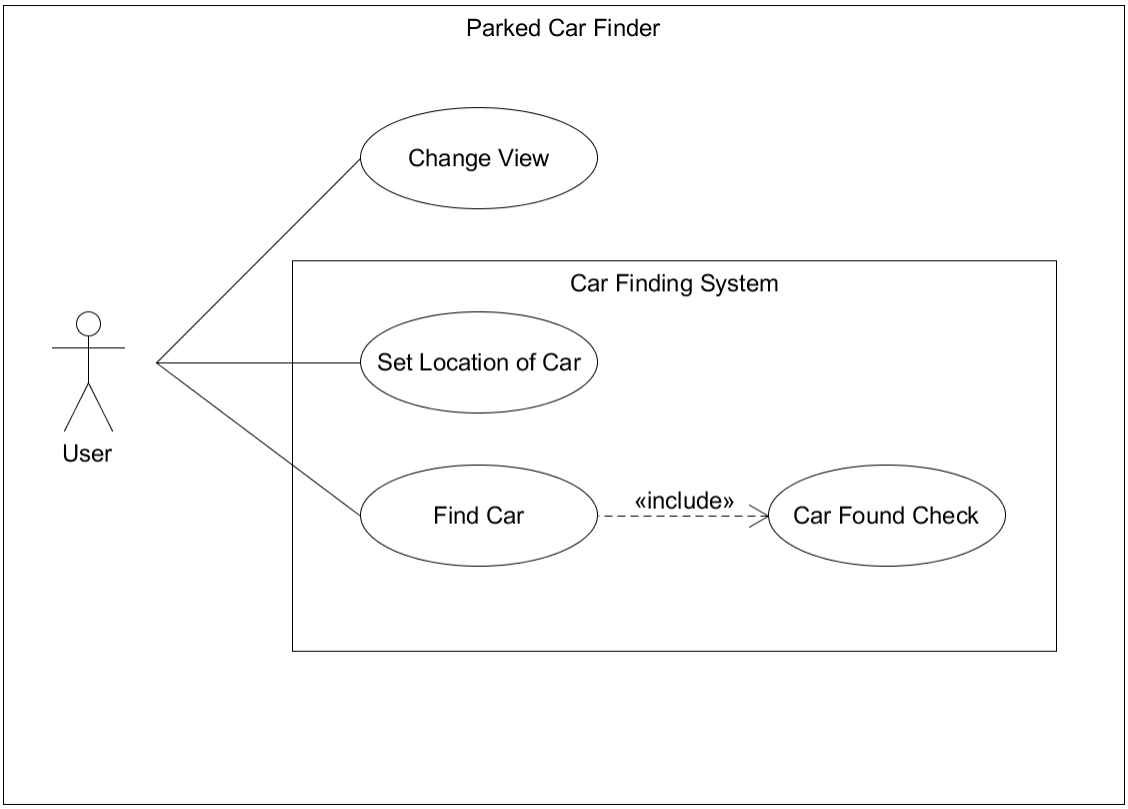


Figure 1: Car Find Application Use Case Diagram

The use case diagram below depicts; The primary actor interacting with the system to search for, locate and be given directions to a specified room on the UEA campus. It is a crucial part of the applications room finding system as it allows the actor to locate and select the correct destination for their journey. The primary actor interacting with the system to gain information about the room that they are currently inside or close to. The primary actor receiving information about a room that they are outside by tapping their NFC-capable device against an NFC tag by the rooms entrance. The primary actor interacting with the application to switch between simple and complex views of the directions showing them to their specified destination room. The periodical checking of the primary actors proximity to their destination room as well as the act of notifying them when theyre at or within close proximity of the room. This is a part of the applications room finding system as it lets the user know if they have reached their destination.

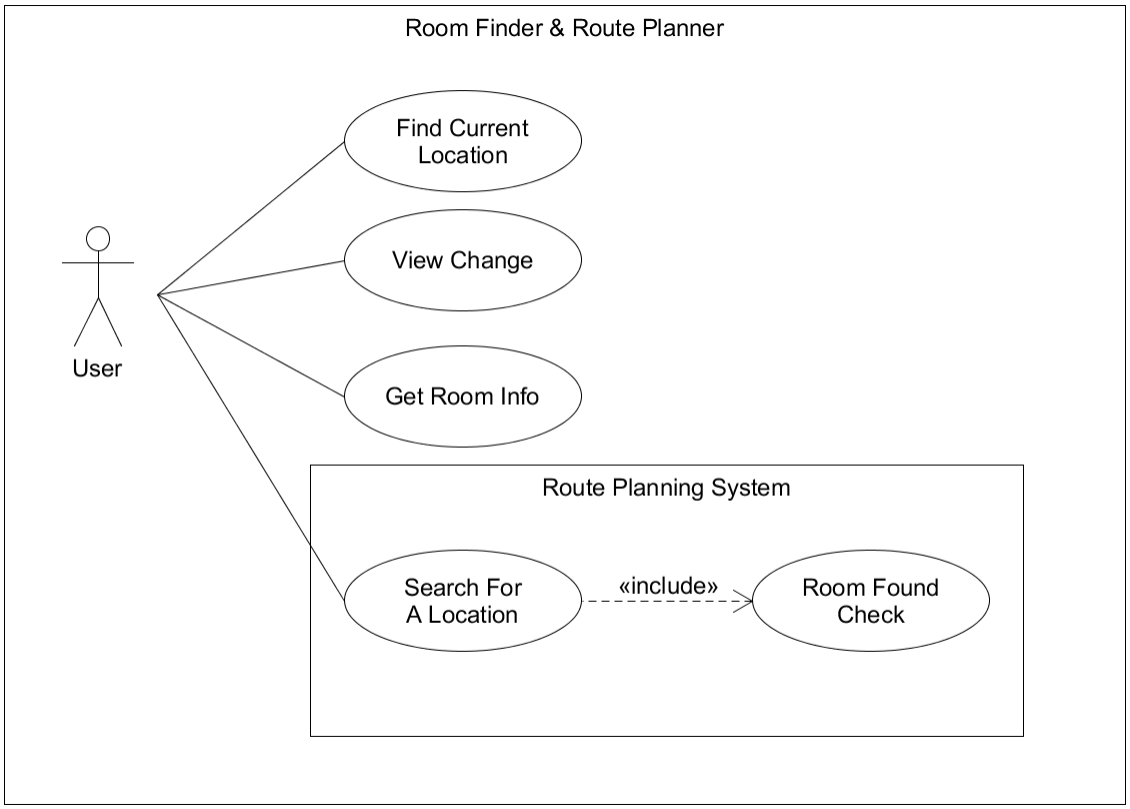


Figure 2: Car Find Application Use Case Diagram

The use case diagram below depicts; The primary actor interacting with the system by locating the nearest location to them while presenting them with a GUI laid over their camera feed. This is a crucial part of the scanning system as it determines which building the actor is to be given information about. The primary actor receiving information from the system about the building that the environment scan has determined that they are pointing their camera at. This is a crucial part of the applications scanning system as it gives the actor the information that they are seeking. The primary actor interacting with the system to share the location they are currently viewing on different social networking platforms such as Facebook, Twitter, etc. This is a part of the scanning system as users must first scan a building before they receive the option to share its information on social media. The primary actor interacting with the system to ask for help on using the application. The actor is given information and instructions on how to use the application.

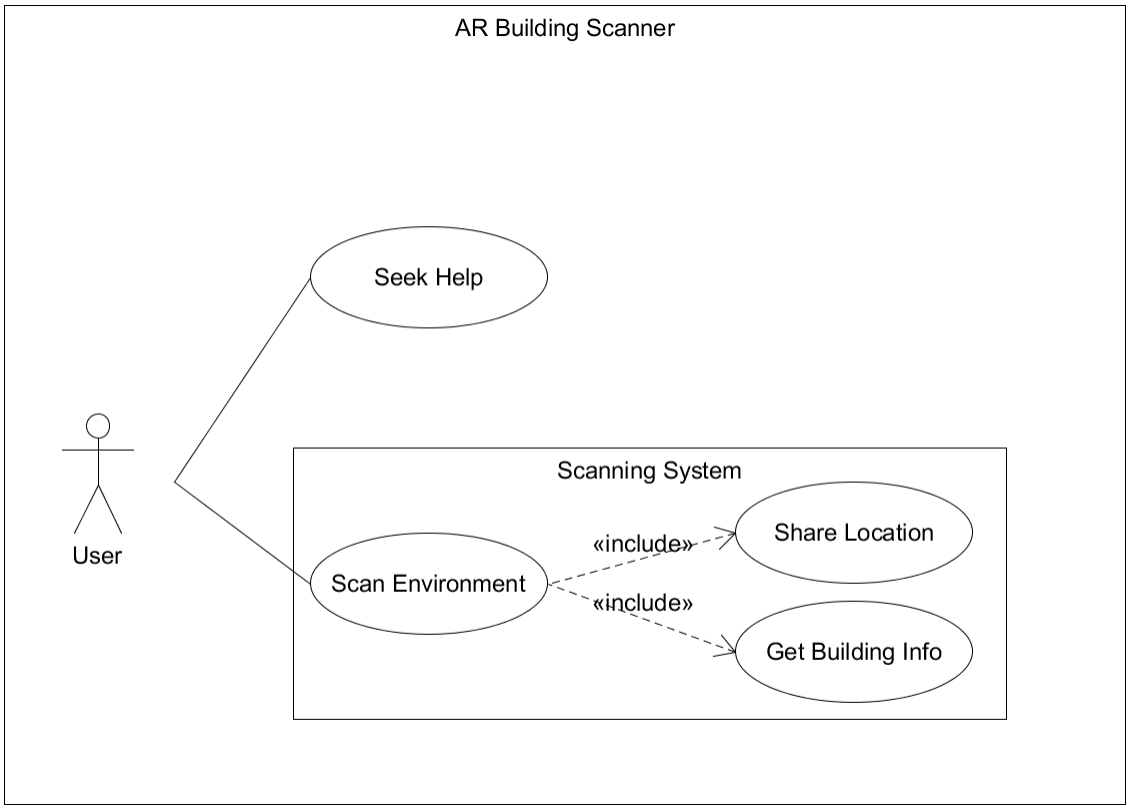


Figure 3: Car Find Application Use Case Diagram

4 Framework Analysis

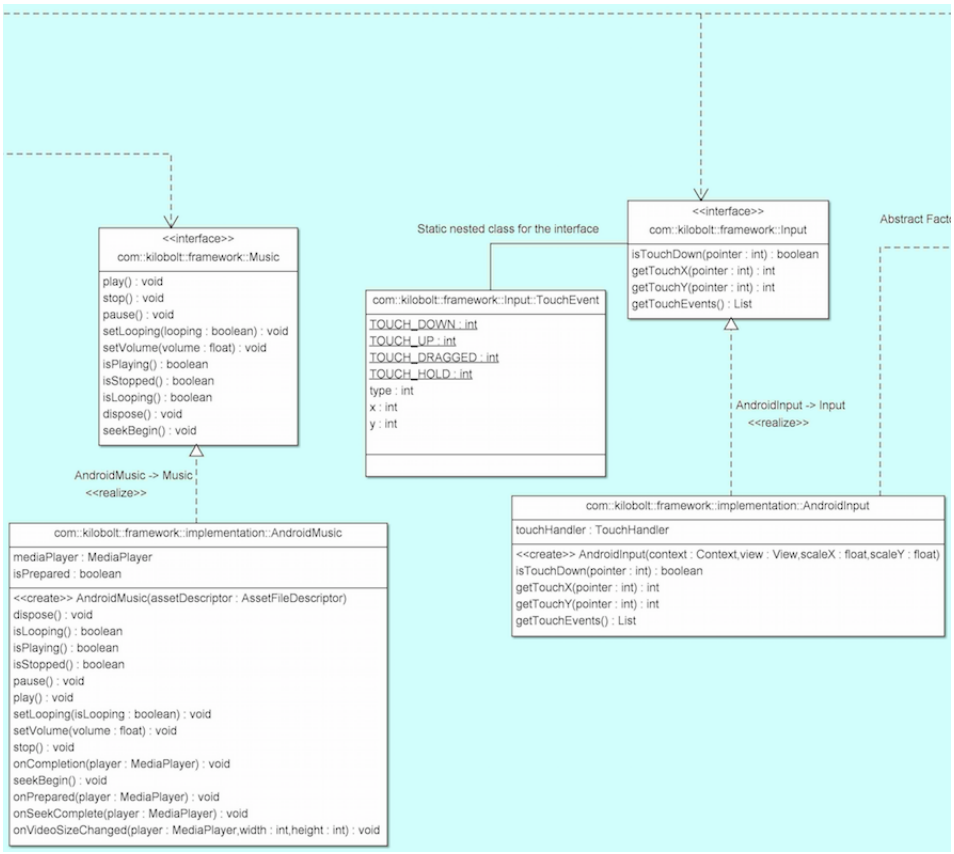
As previously mentioned in the introduction of the document, within this project there is a focus on reuse and refactoring of the 'Simple Android Application Framework'in hope of developing common code which provides generic functionality shared by all three of the applications.

4.1 SAAF Java Documentation

For the documentation of the framework, having a thorough understanding of the structure and how it interacts when running was crucial. We specifically spent time analysing the RobotGame example provided to us in labs, as well as constructing a series of class diagrams to map the relationship between the classes before starting the documentation, giving us an insight into the function of each class, as well as the thought process of the original programmer.

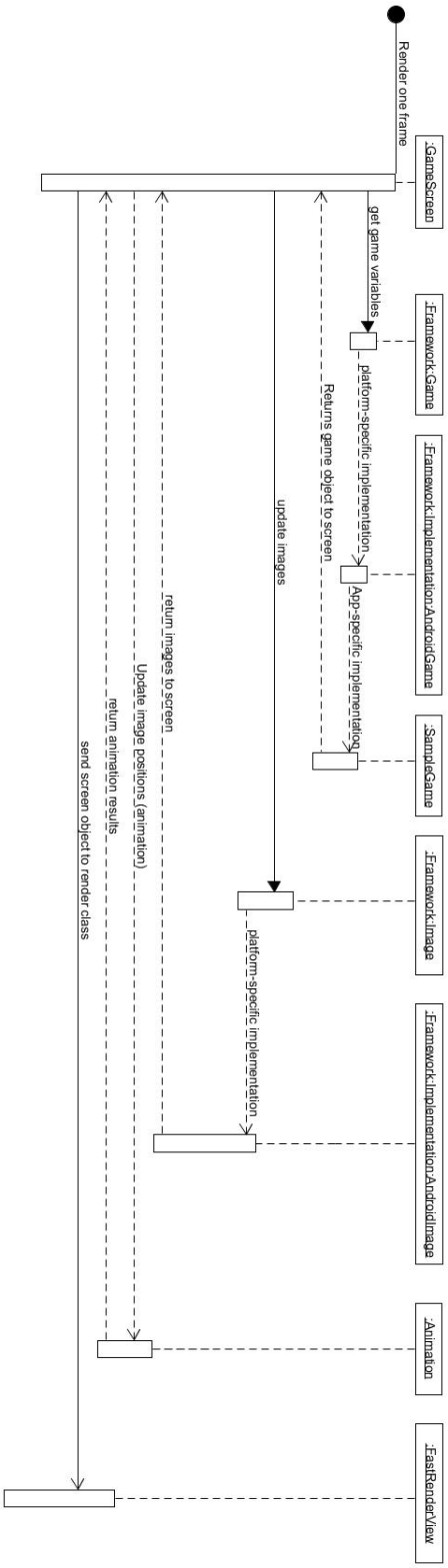
4.2 Original SAAF Class Diagram

Creating a class diagram was the first step to better understanding the Simple Android Application Framework. Whilst much of the meaning can be obtained by looking at the code, understanding how the classes interact and implement each specific interface helped us gain further insight into the inner workings of the code, especially where the input-based classes are concerned, where design patterns such as the Abstract Factory are used in order to implement multiple types of touch simultaneously.

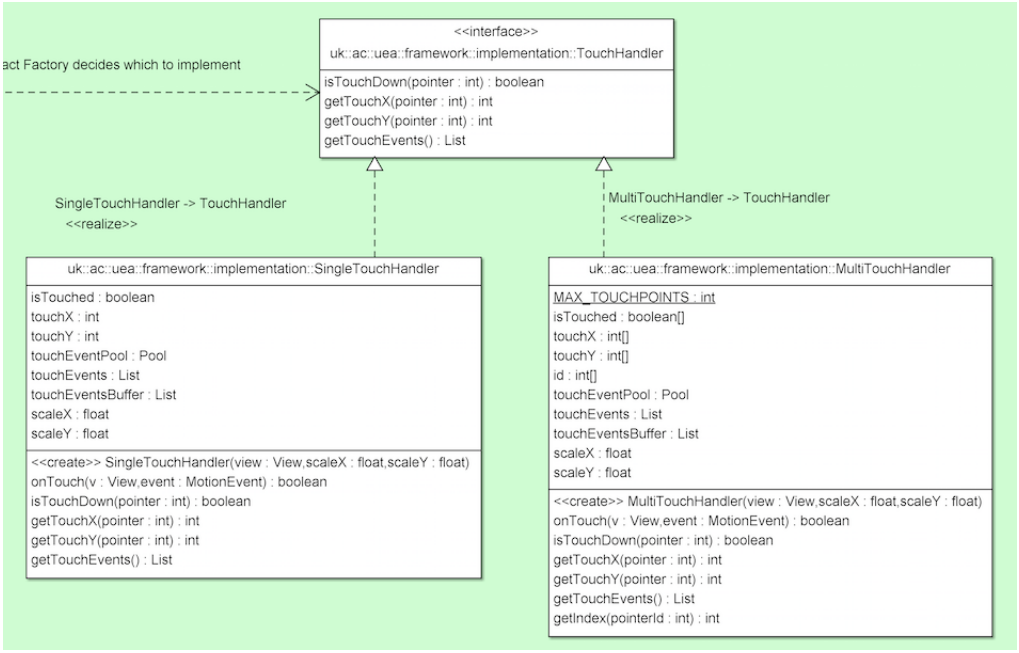


The figure above is an example and only displays a small section of the Original SAAF Class Diagram. The full class diagram has been placed in the containing folder of this document.

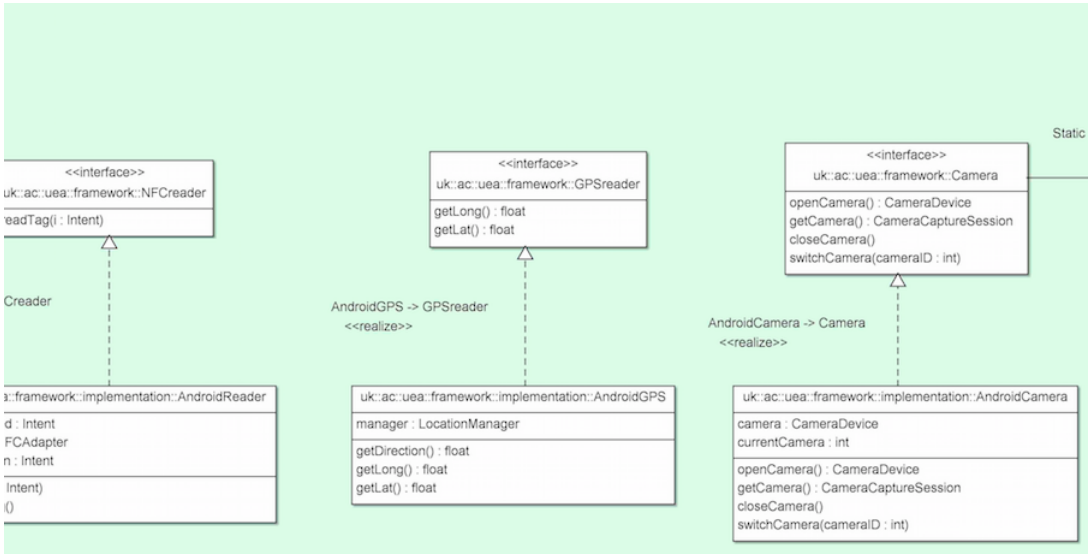
4.3 Robot Game Sequence Diagram



4.4 Revised SAAF Class Diagram



4.5 In-Progress SAAF Class Diagram



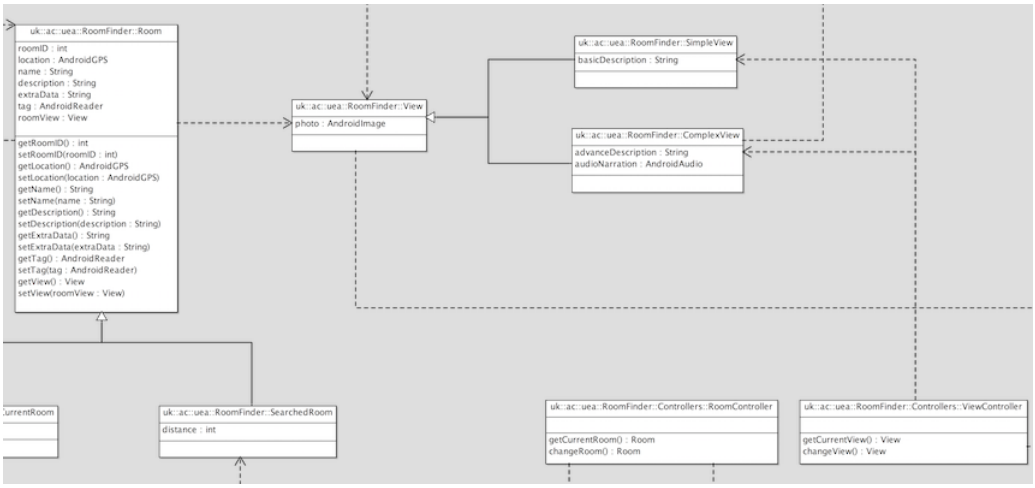
5 Application Design

5.1 Class Diagrams

For the constructions of the class diagrams, we made use of a mapping tool (ArgoUML) to give us a better understanding of the existing framework relations, giving us a clear idea of the existing functions and how they could be adapted and otherwise worked with to construct a suitable set of apps.

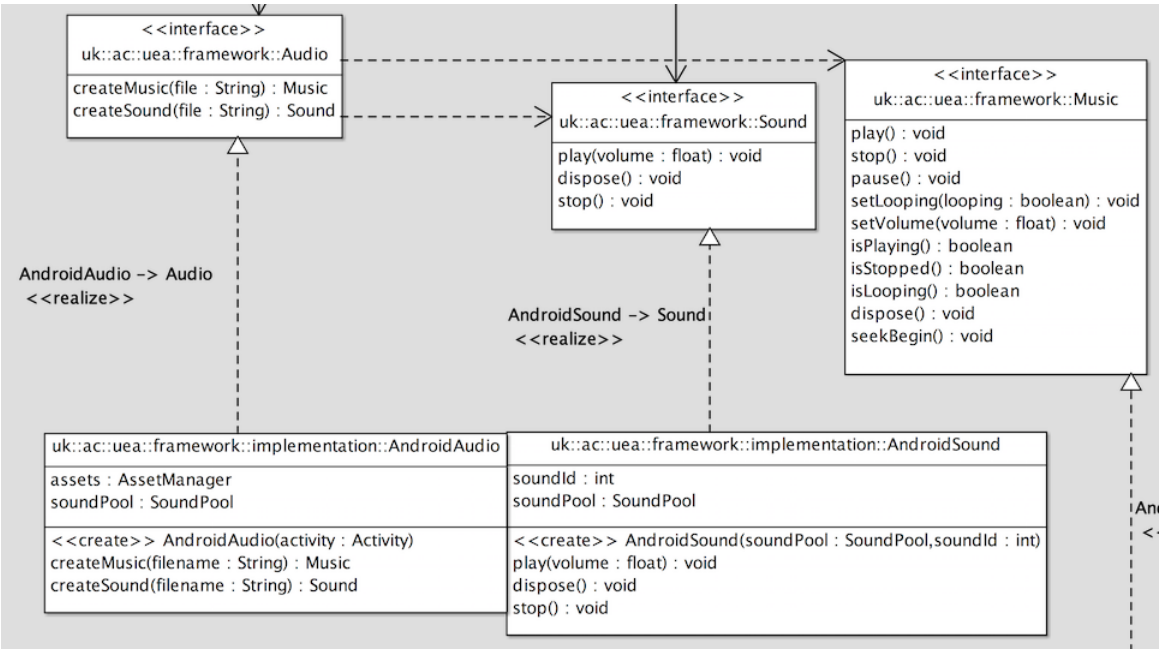
The three figures found above only display a section of each class diagram. The complete class diagrams can be found within the containing folder or by following the relevant dropbox link. In addition these class diagrams have been emailed to the marker of this project.

5.1.1 Room Finder Application



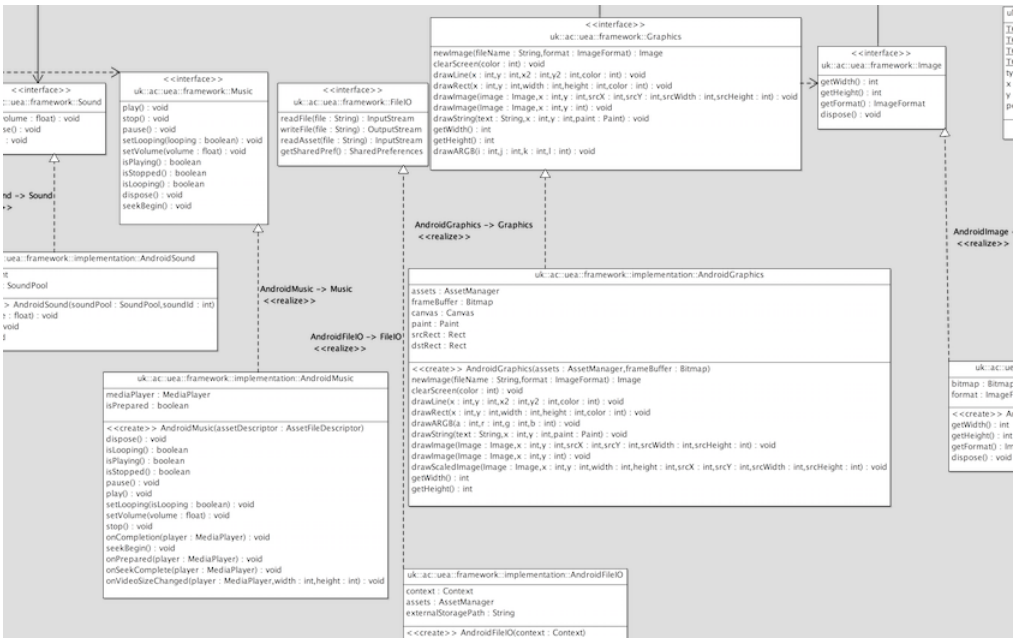
Link: `https://www.dropbox.com/s/onghfioe2t1ancl/Room%20Finder%20Class%20Diagram.png?dl=0`

5.1.2 Car Finder Application



Link: `https://www.dropbox.com/s/qye31vry0bgk0dr/Car%20Finder%20Class%20Diagram.png?dl=0`

5.1.3 Augmented Reality Building Scanner

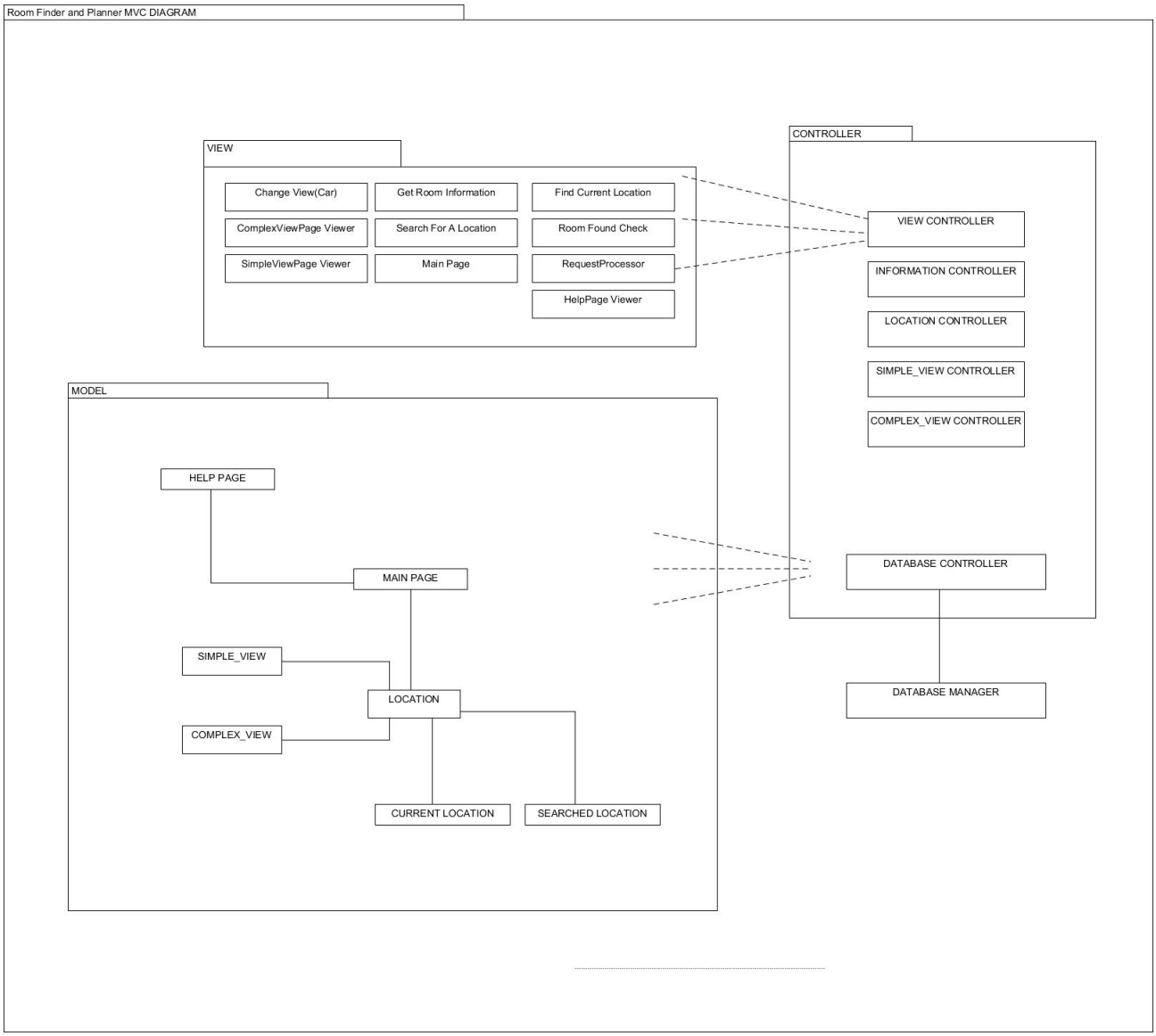


Link: `https://www.dropbox.com/s/8b42idqmmfssd7z/AR%20Building%20Scanner%20Diagram.png?dl=0`

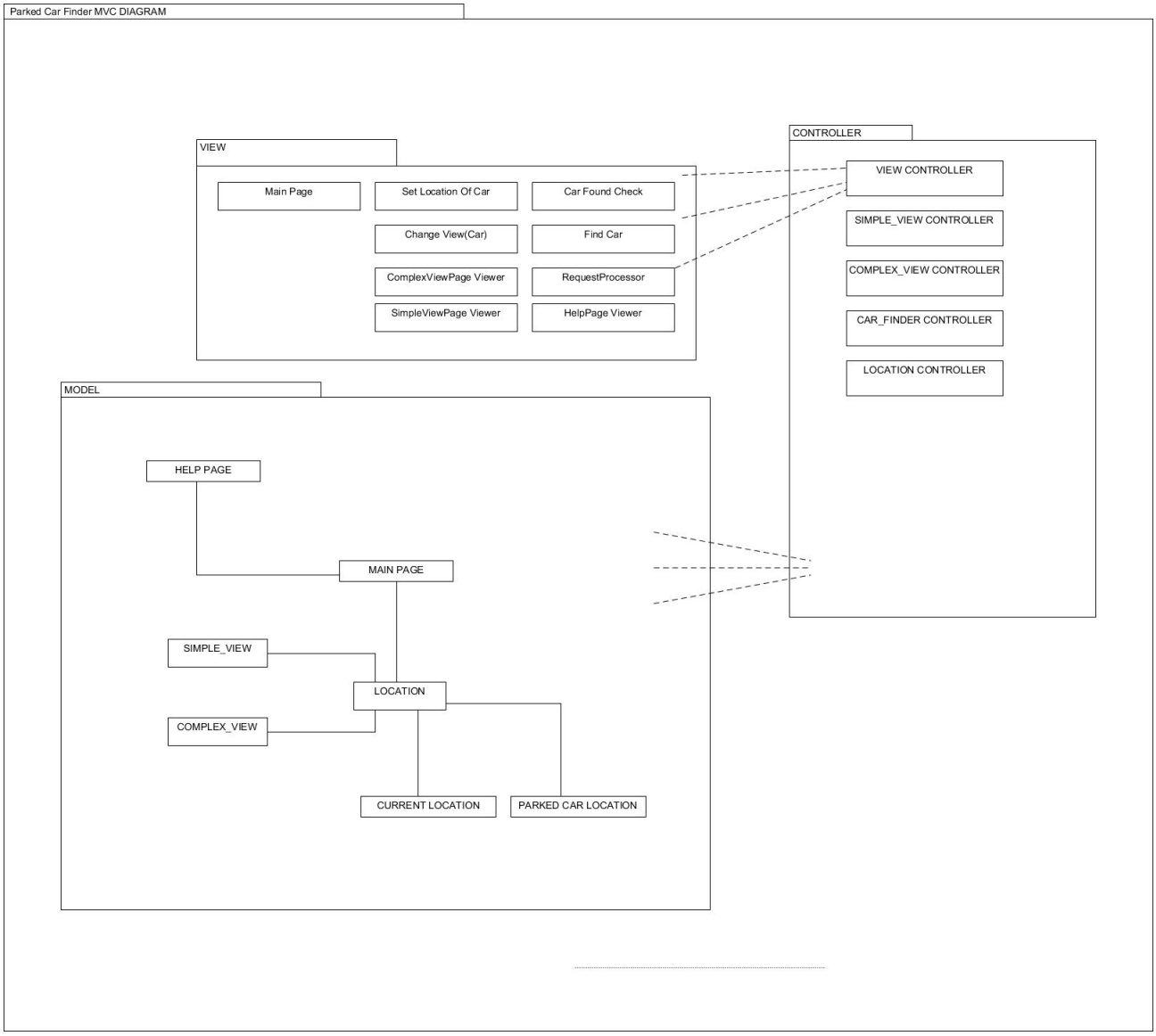
5.2 Architectural Diagram

MVC Diagrams

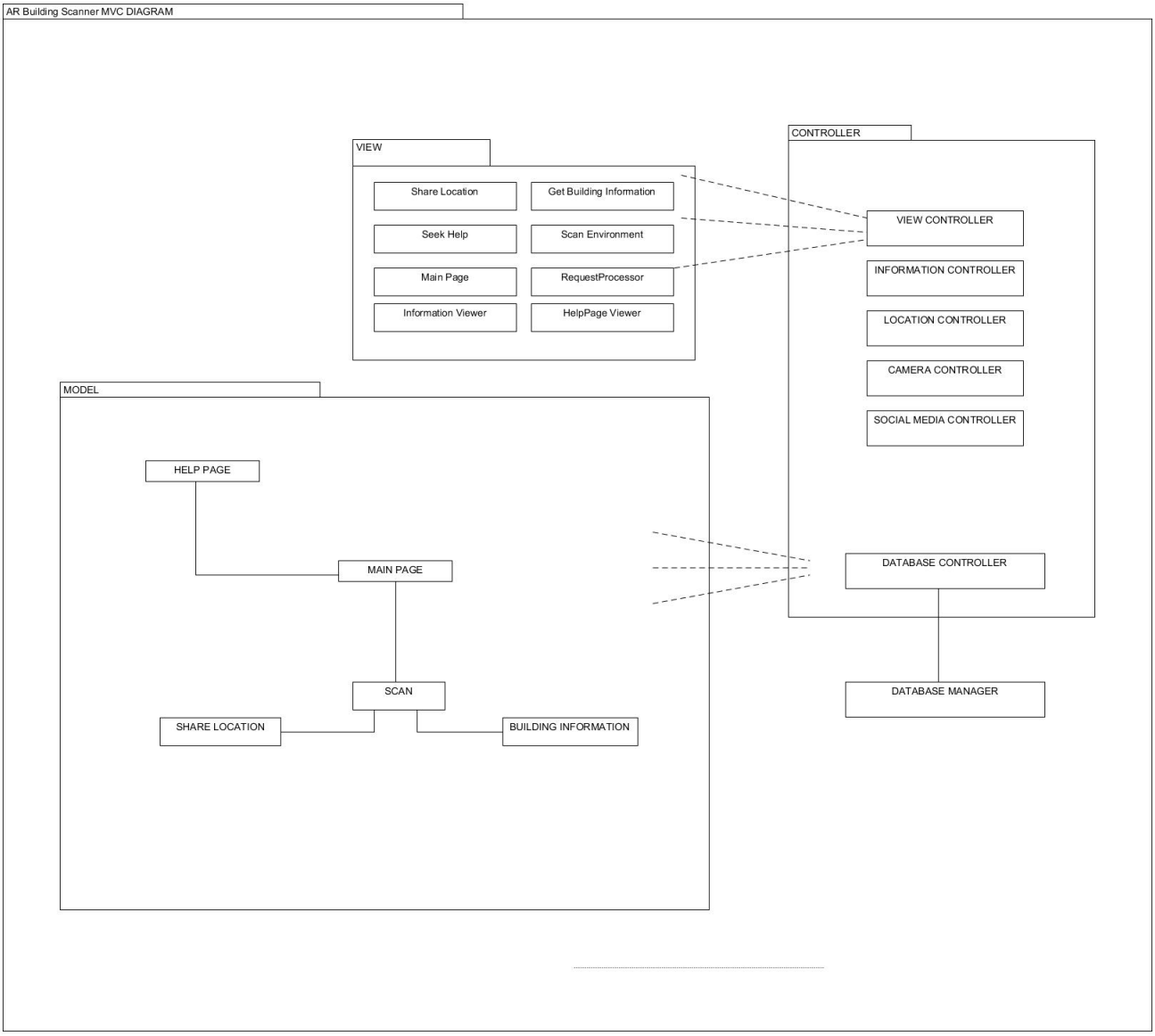
5.2.1 Room Finder Application



5.2.2 Car Finder Application

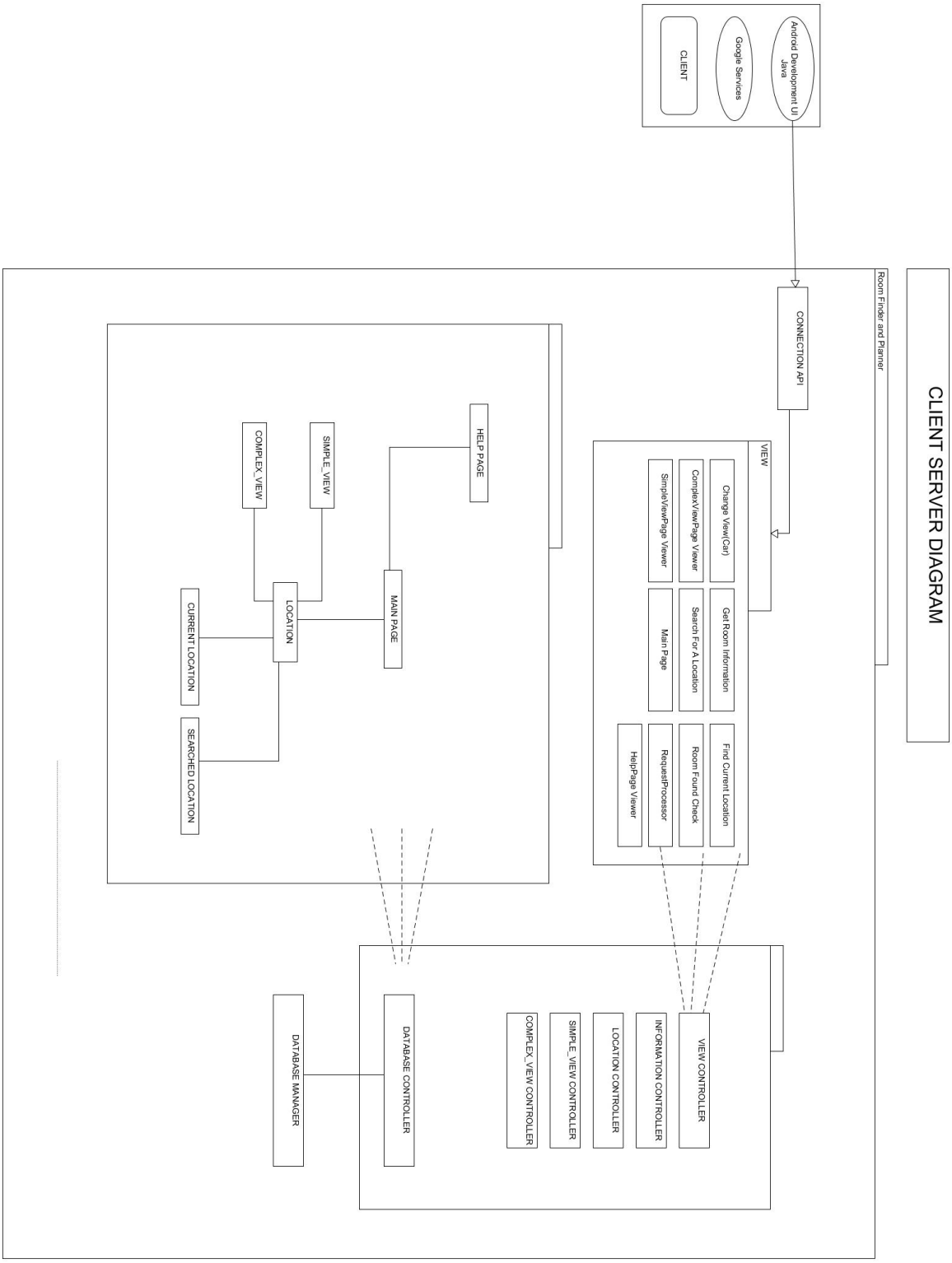


5.2.3 Augmented Reality Building Scanner

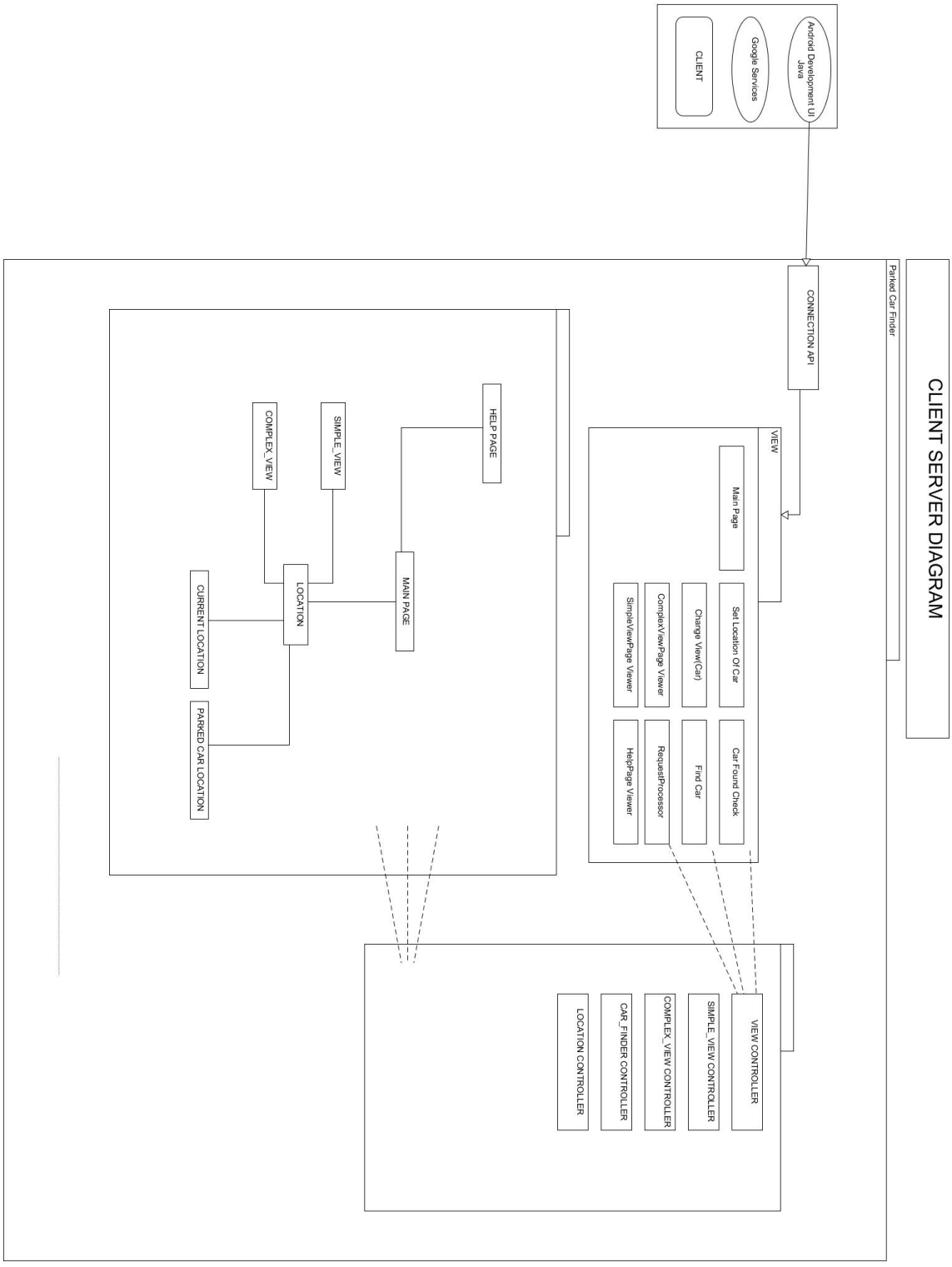


Client-Server Diagrams

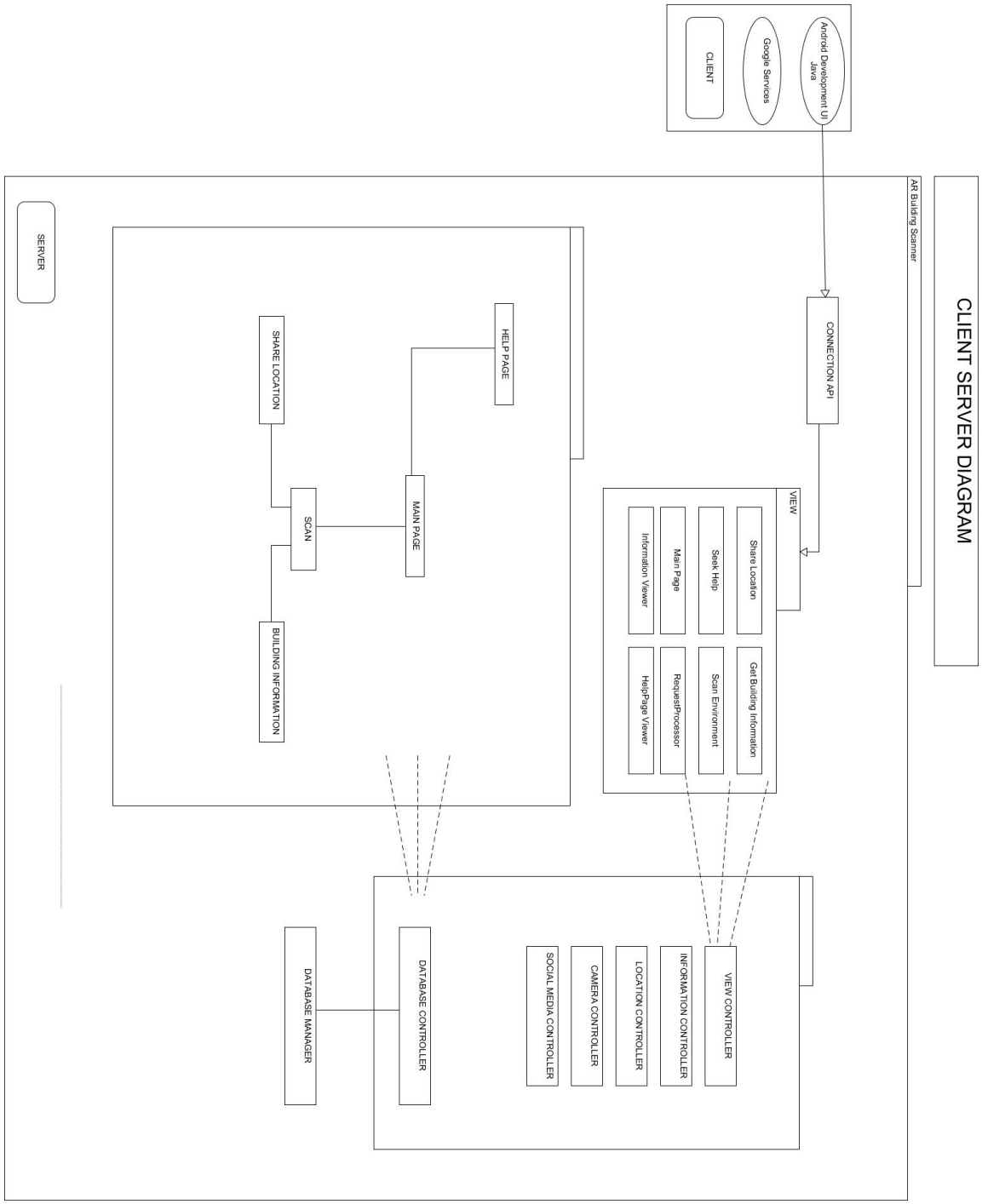
5.2.4 Room Finder Application



5.2.5 Car Finder Application



5.2.6 Augmented Reality Building Scanner

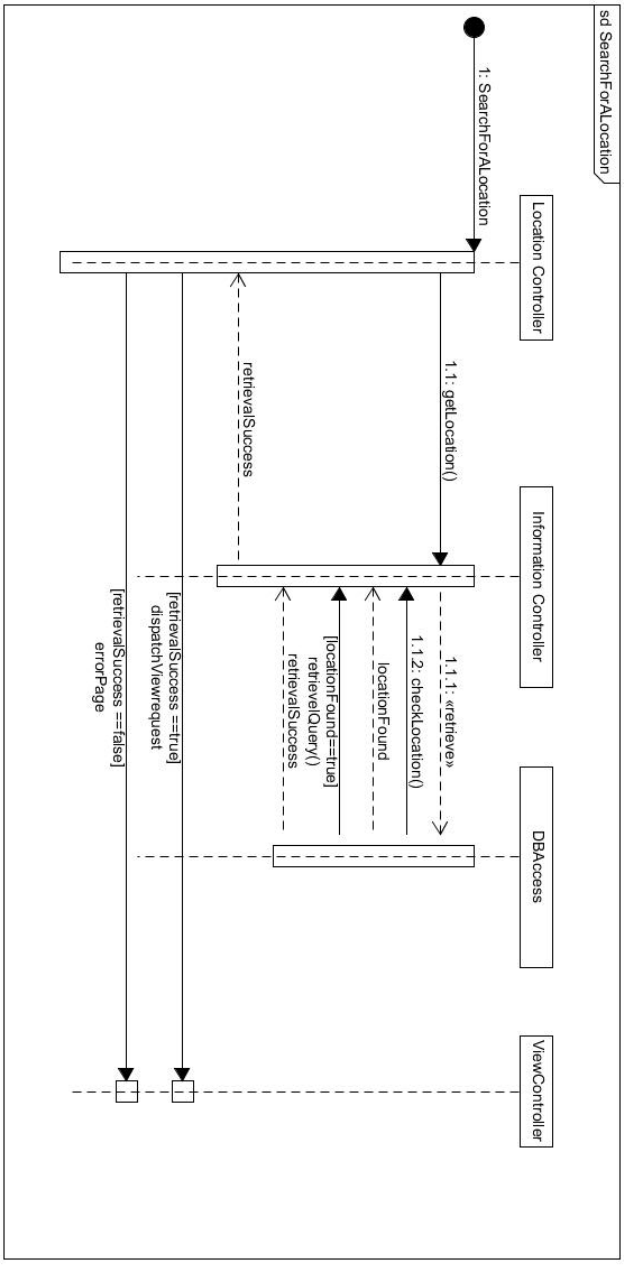


5.3 Sequence Diagram

Sequence diagrams represent the runtime behaviour of a certain feature in a system by showing the interactions between classes.

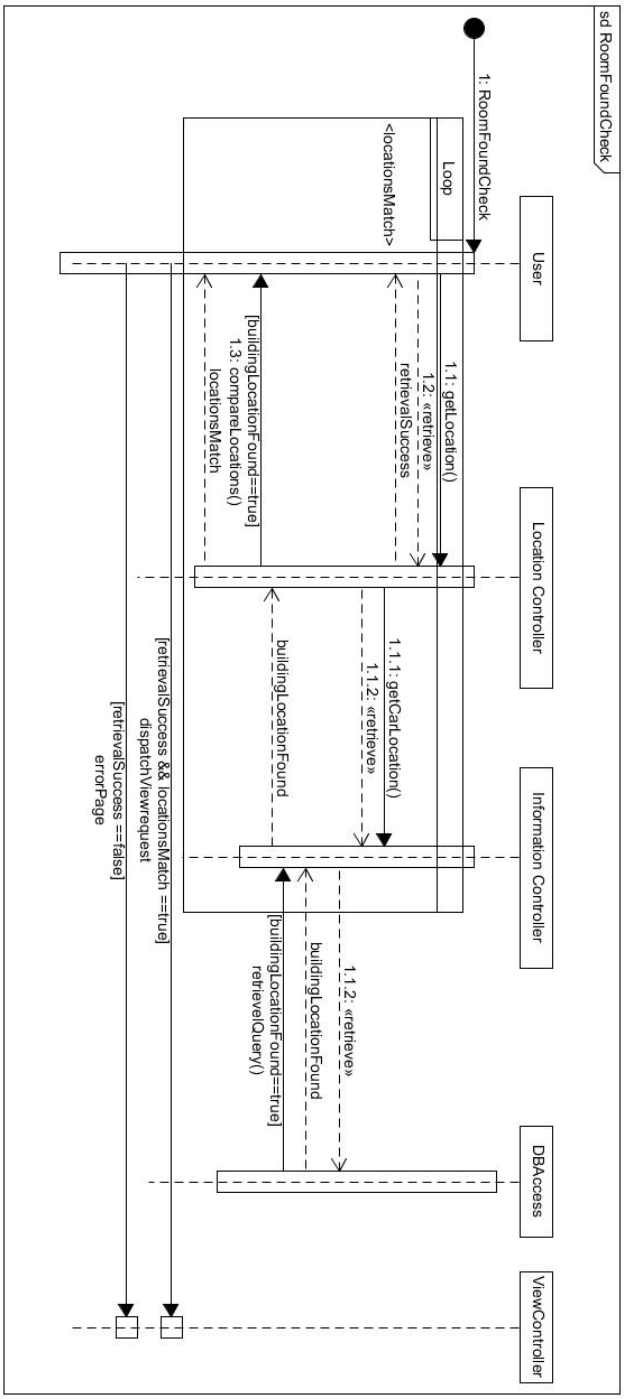
5.3.1 Room Finder Application

Search For Location



This sequence diagram represents the Search For A Location function; User inputs a location, the location is checked against the database and then saved in SharedPreferences to be used later.

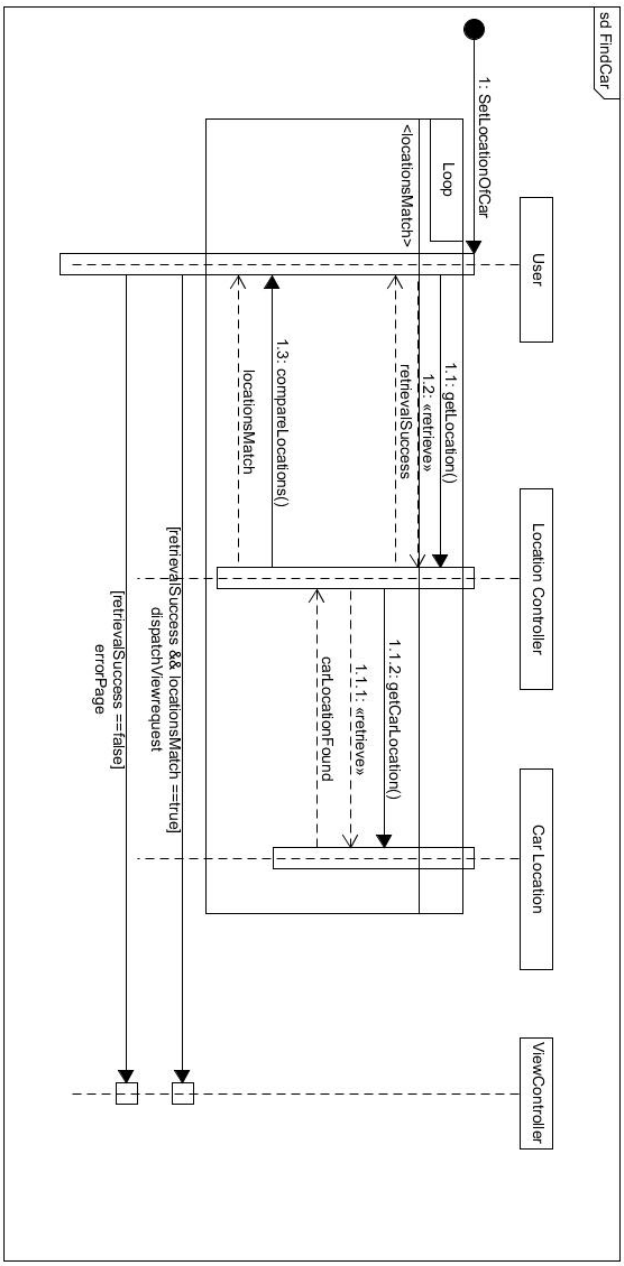
Room Found Check



This sequence diagram represents the Room Found Check function; User inputs a location, users location is saved and checked against the searched location until the room is found, user is prompted when the locations match.

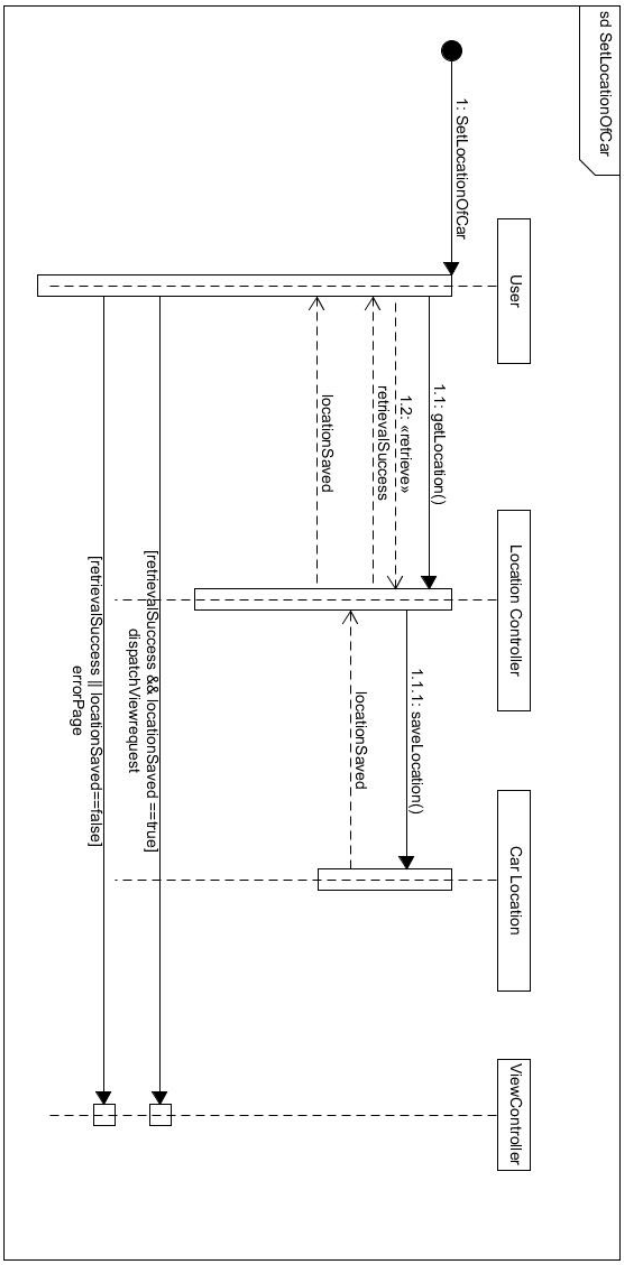
5.3.2 Car Finder Application

Find Car



This sequence diagram represents the Find Car function; Car location is retrieved, users location is saved and checked against the car location until the car is found, user is prompted when the locations match.

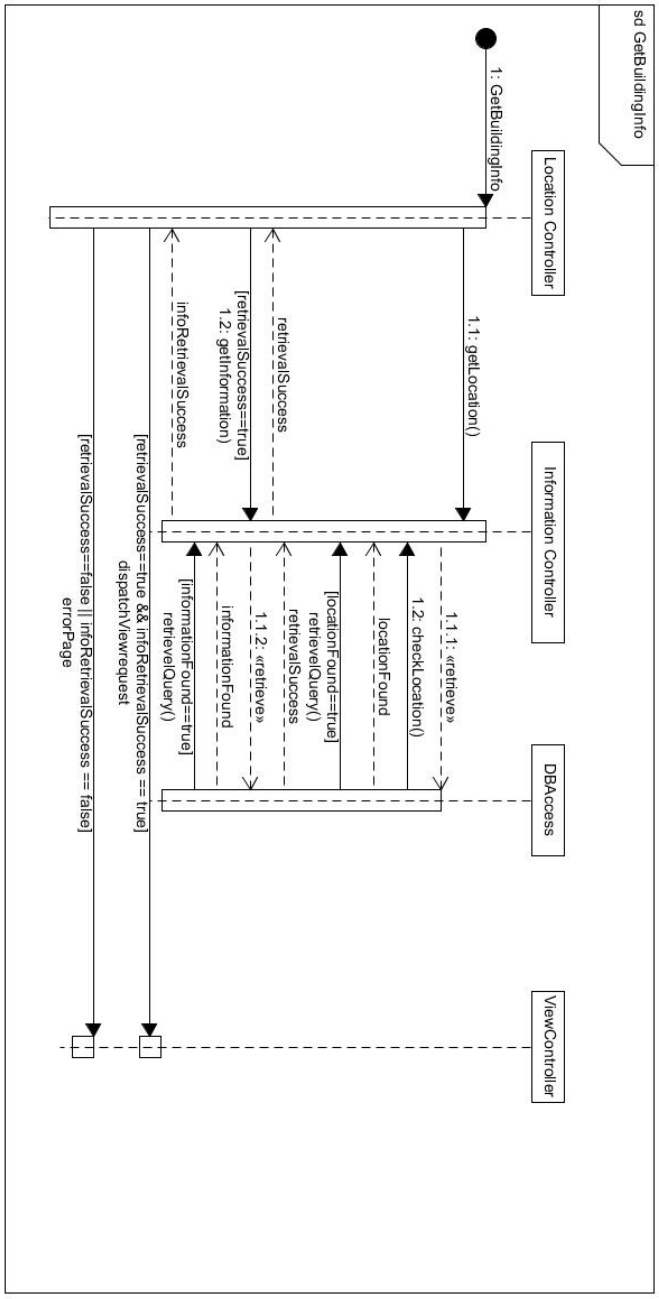
Set Car Location



This sequence diagram represents the Set Location Of Car function; The users location is checked and saved as Car Location to SharedPreferences to be used later.

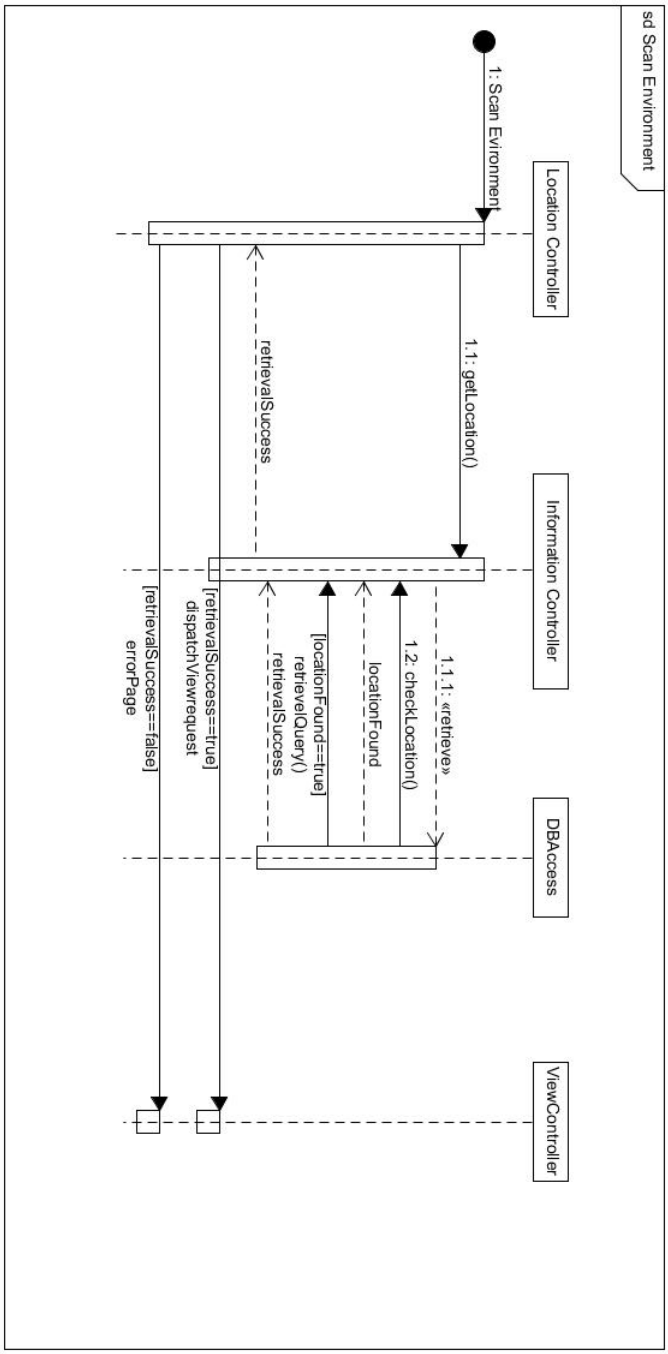
5.3.3 Augmented Reality Building Scanner

Get Building Info



This sequence diagram represents the Get Building Info function; The users location is checked and database is accessed to find the closest location, Information about the found location is also retrieved then shown on the camera screen to be viewed by the user.

Scan Environment



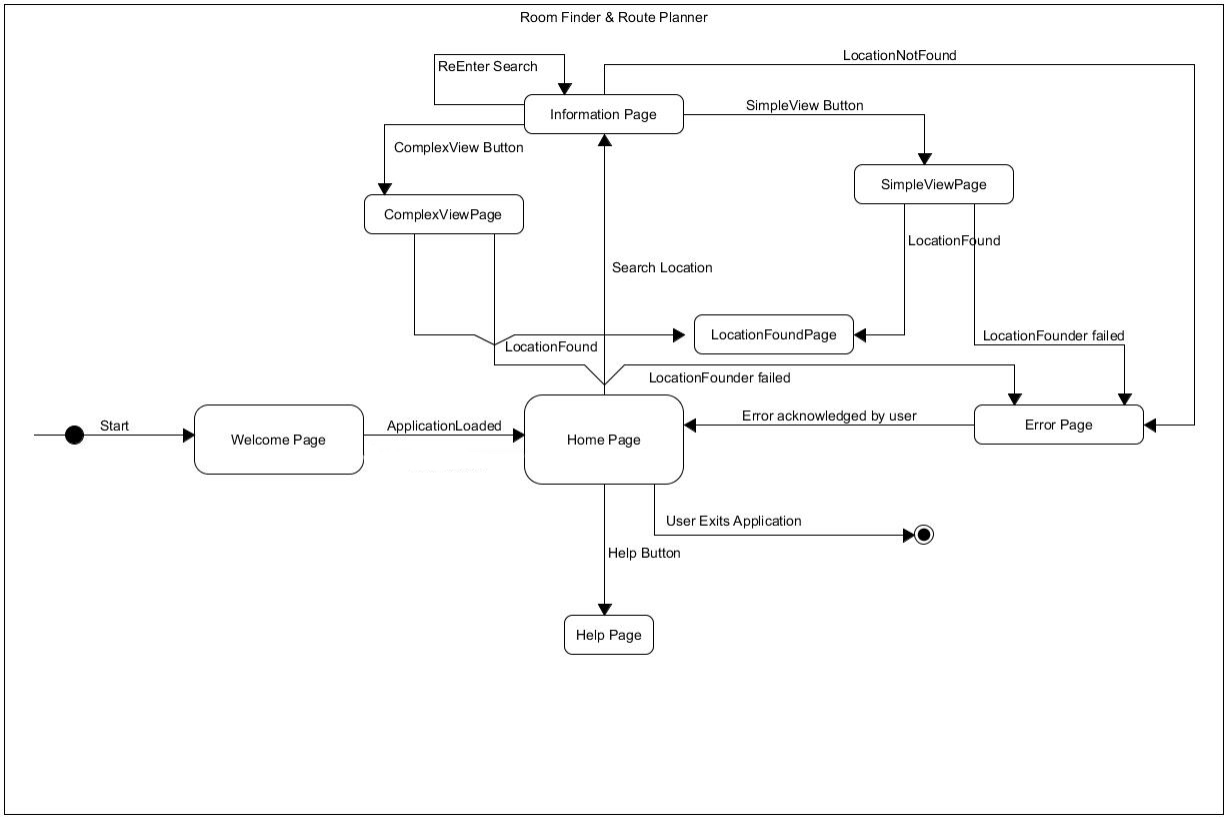
This sequence diagram represents the Scan Environment function; The users location is checked and database is accessed to find the closest location. Location is saved in SharedPreferences to be used later.

5.4 State Diagram

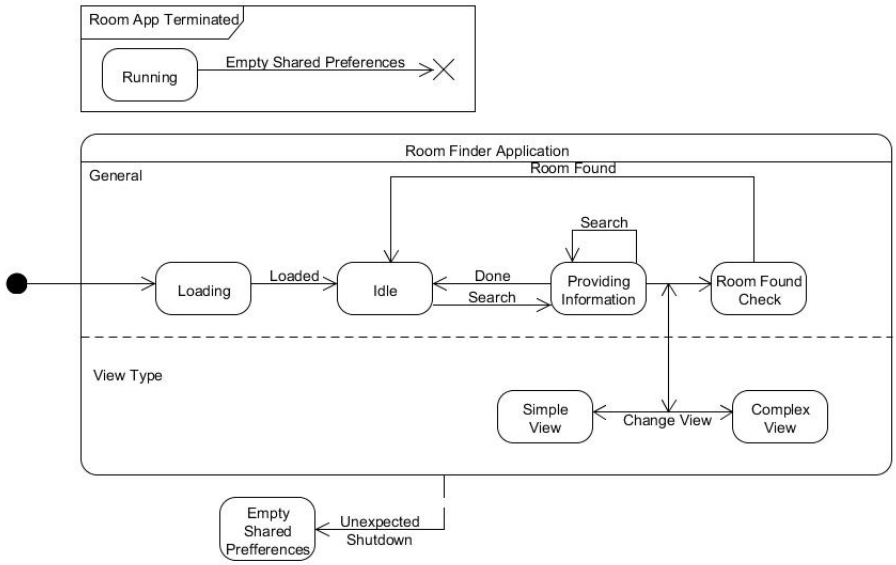
The following section will consist of state diagrams; state diagrams provide a visual representation of objects which have both behaviour and states. These diagrams visualise the various states of an object/component within our applications in addition to the the transition process between states. These diagrams are used to reflect any abstract behaviour within the applications as well as an indication of restrictions on specific actions.

5.4.1 Room Finder Application

Application State Map

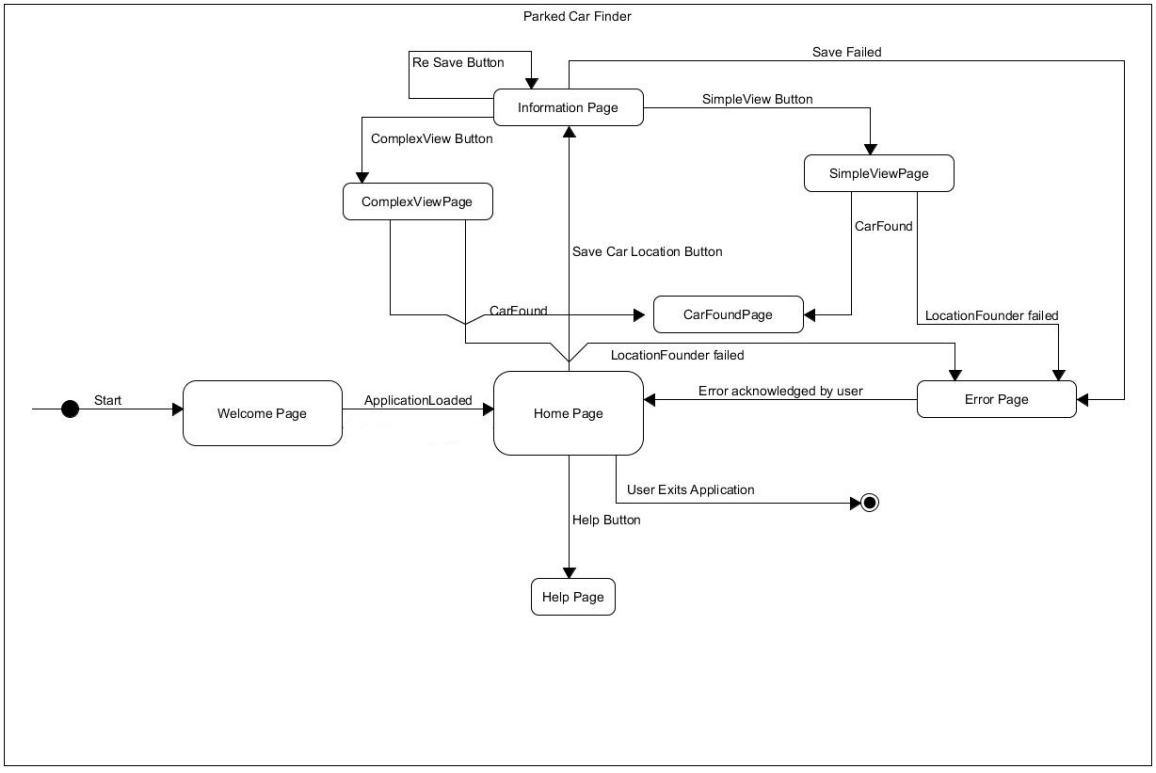


Active Application State

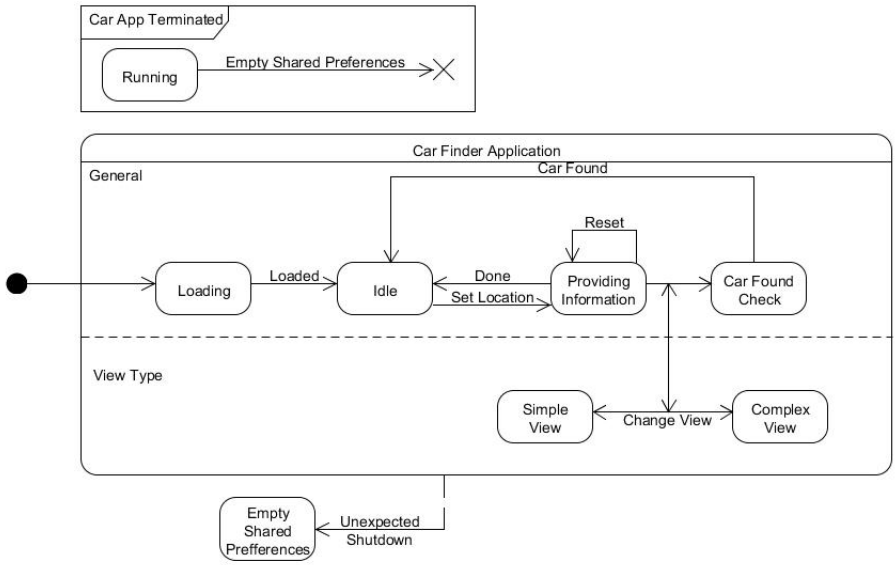


5.4.2 Car Finder Application

Application State Map

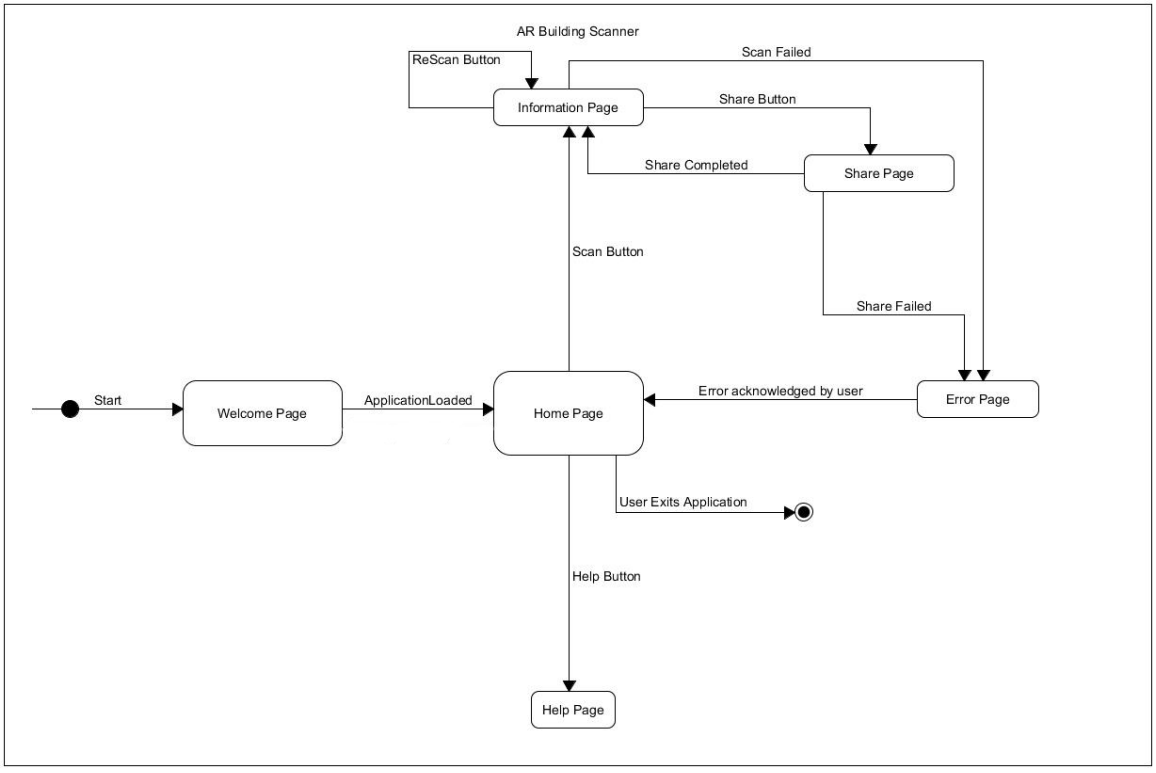


Active Application State

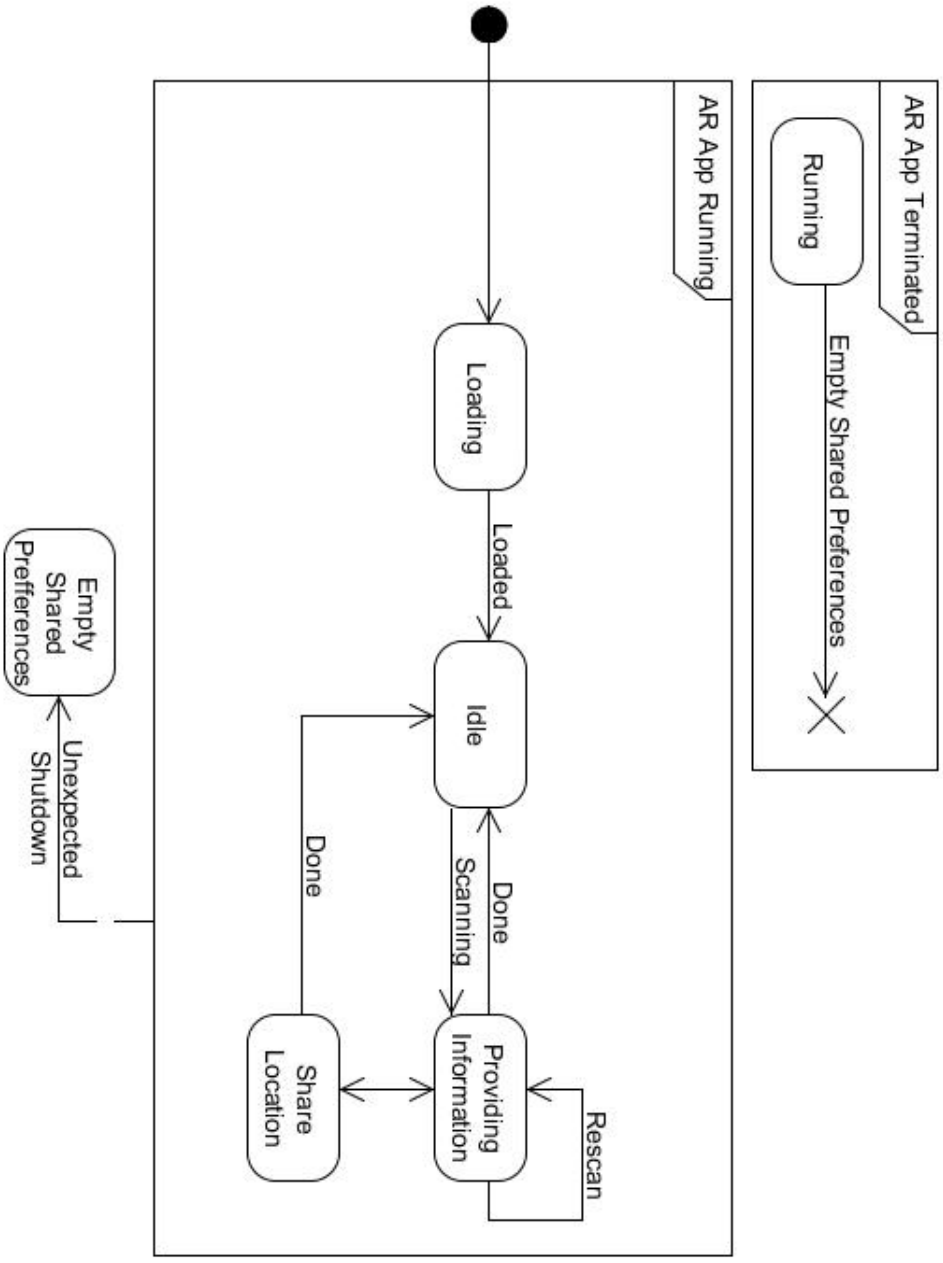


5.4.3 Augmented Reality Building Scanner

Application State Map

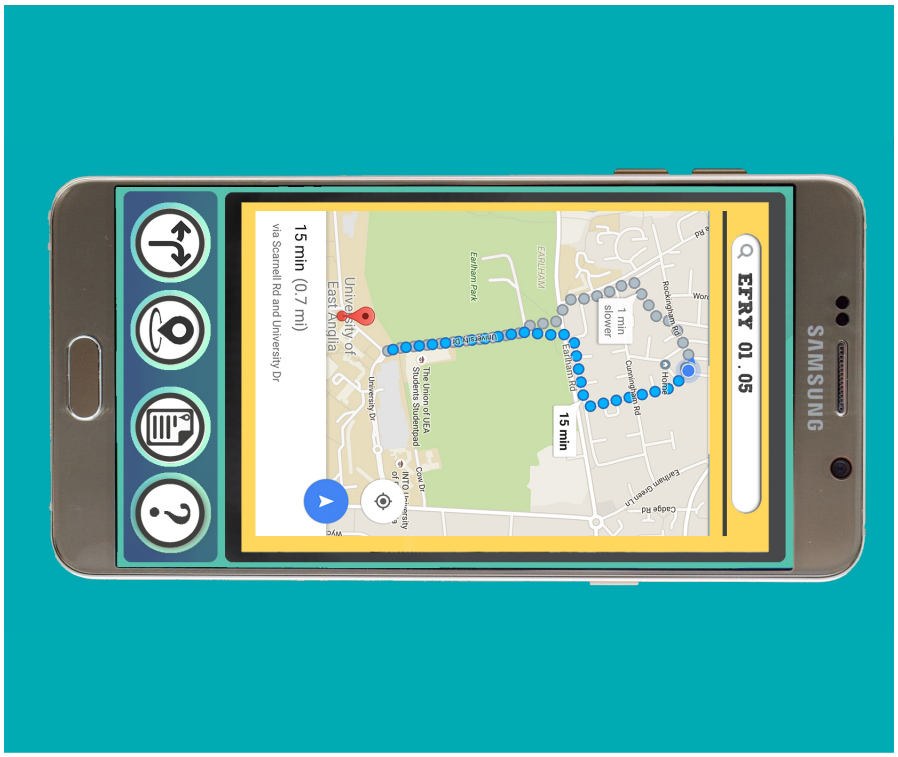


Active Application State

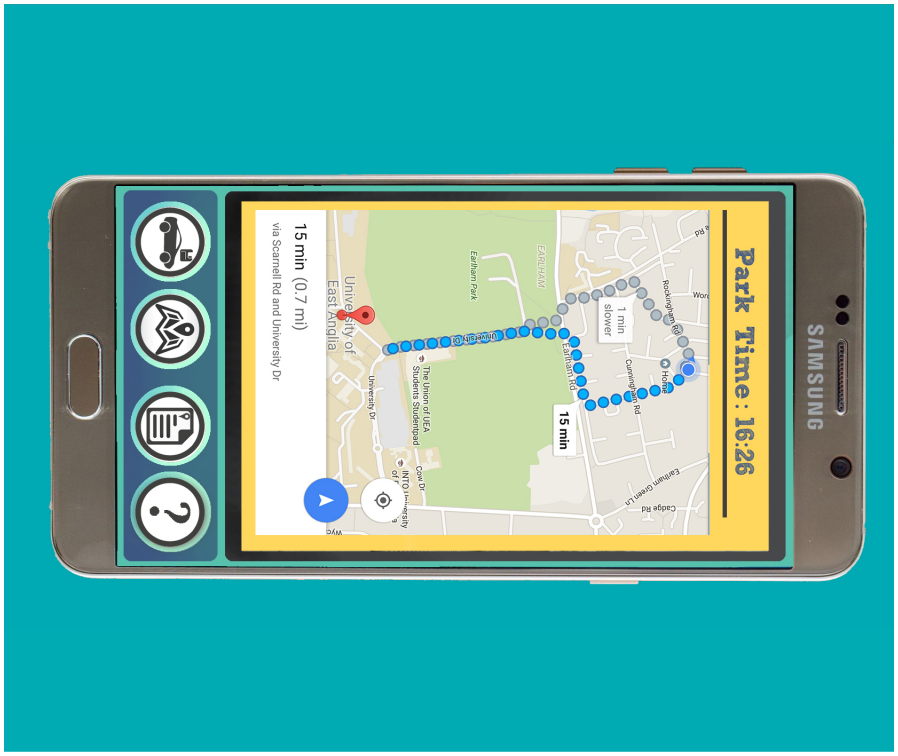


6 Concept Designs

6.1 Room Finder App



6.2 Car Finder App



6.3 AR Building Scanner App

7 Conclusion

In conclusion, after analysing the research that we gathered we believe that our diagrams are an accurate representation of applications and will aid the group greatly during the future implementation phase. Furthermore, there is a high level of consistency that can be found throughout the design document; similarities can be found within relevant diagrams which indicates a high level of code re-use. Lastly, as a group we will this document has concluded all aspects of the design phase well and leads neatly into the implementation phase.

8 Glossary

Location: An area of importance within the limits of the university.

Social Media Screen: A window that contains the social media buttons to share the Location.(twitter, facebook, swarm, tumblr, etc.)

Help Screen: A window that shows information that helps the user understand how the application works.

UEA Open Day: The University of East Anglia encourages potential undergraduates to visit the campus and get a feel of student life at the university. Open days allow for this opportunity as well as viewing both academic and leisure facilities.

Concept Design: Allows for a visual representation of how the applications will potentially be viewed by the user.

Information Screen: The screen that pops up after scanning an area on the Augmented Reality Building Scanner application. The screen has two windows, one for the camera feed and one for the buttons in the bottom which are:

- Scan Environment
- Room Info Button
- Share Button
- Help Button

9 Trello

Link: `https://trello.com/b/xM1LbGDu/software-engineering-2-open-day-app-development`

10 Student Numbers

- Joshua Lee - 100057593
- Beren Kusmenoglu - 100047218
- Caine Cridland - 100071987
- Jack Clements - 100053671