# Requirements and design

From the start of development the team had a general idea of how the application was to look as well as the general flow of the various pages. However as the application development team was new to Android programming and design this project was very much a learning experience. Many of the design plans and software structures changed as the team became more proficient with Android and experimented with different layouts and interfaces.

It was decided early on that as this was a university project and not an application to be released that the team would target only Androids newest software “Lollipop”. This decision was made as the team wanted experience that would be more relative to the software landscape at the end of the course.

## Interface Changes

During development many changers have been made when compared to the initial planning and cognitive walkthrough, for example:

* The application colour pallet was changed slightly to a lighter blue as the original caused problems with contrast and readability when deployed to a phone
* The home page (displaying a list of current adverts as well as buttons to create new ones) was changed to a “Dashboard” with several icons linking to the main parts of the application as well as the ability to search all adverts from the task bar. This change was made after deciding the original design was too cluttered and awkward to use.
* While the flow of the advert creation process was largely the same several changes have been made to reduce the number of steps involved and to de-clutter the interface.
* Unlike the website that was developed the user’s different bids, such as incoming and outgoing, are separated into different tabs that are reachable by swiping left and right. This was one of the many subtle changes made in the design to accommodate the limited screen space available to a mobile application.
* The inclusion of a menu bar that is consistently accessible from any page to aid user navigation

These interface changes and others where made as a result of experimentation and user testing, and while there are several areas that could still be improved upon if given more time, the application flows well and looks professional in most places.

## Problems and Issues – design

While the android team have created high quality work in the past they were new to android, this meant having to research and experiment in order to implement almost every feature that has been added, and while Android is based on Java there are many differences and quirks. These problems meant that development proceeded much more slowly than was anticipated and a lot of effort was expended fixing bugs and problems that didn’t have a clear source.

As the development team had no experience in mobile development many early interface designs where awkward to use and felt cluttered. As development progressed the interface design improved markedly, as well as the team’s ability to come up with new designs quickly and with fewer iterations.

Another problem encountered by the design team was a lack of Model View Controller support within android, making many of the early system designs incorrect. Android requires much more to be done within the “Activity” class that deals with the GUI than the team was used to and this has led to some messy code being included in the release. However with the experience gained on this project the team feels confident that they have learnt enough to improve upon this drastically, if given enough time or on a new project.

## Missing Features

While most of the major features have been included in the application there are several that have not made the final release. Most of these features have been dropped due to time restraints, however several where not included by design.

### Not included due to time restraints:

* Receive notifications of bids and transactions.
* View a different members details.
* Option to delete an advert.
* View the users past adverts.
* View bids on an advert via the advert details page.
* Updates to bids and transactions on the dashboard
* Ability to contact admins from inside the application

### Not included by design:

* A list of all active members
* The ability to search for a member
* The ability to edit an advert that has been posted
* Filter adverts based on category etc. (This has been replaced by the search feature)

Many of the features not included due to time restraints would be easily implemented if the project where to continue, unfortunately due to the difficulties with android and the lack of a 5th member they have been dropped for the release.

## Conclusion?

While the Android application has gone through several iterations and is missing several minor features, it is very close to the team’s initial idea and completes all of the required functionality, looks professional, consistent, and is relatively easy to use.

Overall the team is pleased with the outcome of the application and agree that given more time to polish and develop it, that it could be used in the real world.

# Software Engineering

The Android team has gained much experience by undertaking this project, improving their skills in not only Android itself, but also in Java and increasing their skill and understanding with regards to multi-threaded programmes.

Android applications are required to do many processes on separate threads in order to prevent locking up the main thread: if an application blocks the main thread for more than a few seconds it is immediately terminated. Because of this the team has gained much experience working with threads and debugging multi-threaded code, as well as a much better understanding of thread safety and how best to use threads to speed up applications.

The team is mostly pleased with the software side of the application: it runs well and is mostly clear and easy to understand, however if the project was to be undertaken again it is believed that the overall software structure and design could be greatly improved using the experience gained whilst undertaking the project.

## Noteworthy Features

Almost the entire application is multi-threaded, running different threads for encryption, networking and the GUI itself. The application is thread safe and no corruption can occur whilst any particular thread is running.

As the team has learnt from previous projects it is important to design the class structure well to contain any changes that need to be made. Toward this end the application has been split into packages as much as was feasible, and the data model and classes have been separated from the GUI as much as possible, making development and maintenance much easier, as well as improving code navigation.

For several sections of the application the team used the same XML layout form in order to increase the consistency of the layout. The whole application uses the same menu and action bar layout throughout and many of the list layouts have been repeated.

For the list of bids and transactions the team opted to use Fragments as this allows multiple displays to be included in a single activity, allowing the user to quickly swipe between lists. This implementation of a fragment allows the encapsulation of several types of bids, meaning fewer resources are used and the usability is improved as navigation between bids does not involve starting new activities.

## Coding Practice

Several methodologies have been employed in the development of the Android application including the “Do not Repeat Yourself” principle (DRY) where possible and the separation of classes into packages to not only increase productivity during development, but also aid in maintenance of the code in the future. The code has also been extensively commented by the team in order to help readability.

Standard Java naming conventions have been used throughout the application, aiding development and making it easier for other developers to understand how the programme works, however some of the variable names are not ideal and leave much to be desired. This was mainly caused because many of the features required experimentation and research to implement and sometimes code was not correctly cleaned up after it was fixed.

While it was initially planned to use more abstractions and interfaces to improve the structure of the application the team has failed to use these extensively due to a combination of being new to Android, the structure of Android itself, and the experimental nature of the development process.

## Problems and Issues – code

Several problems with Android development had to be overcome during this project, requiring many hours of research and slowing development down significantly. One of the worst problems the team came across was the amount of deprecated code included in Android and the lack of up to date guides provided by Google; many of the official guidelines where out of date or incorrect at best, non-existent at worst.

During the development many software bugs where overcome, some simple programming mistakes, some caused by the works of Android itself some of the more notable ones include:

* The JSON library required to send information to the API encoding a string different in Android and Java: in some cases Android was adding “[]” around the string even though the same library was being used as in Java. This caused the API to return a failed message and was fixed by using string manipulation to remove the brackets after the object had been encoded.
* On the Bids and Transaction interfaces the application was crashing when one of the team was using it but not when another was. This bug was caused by the way the first member was holding the phone; an item in the list was being pressed as the menu was being swiped. This bug was fixed by delaying the generation of an on click listener until after the swipe had been completed.

It is believed that the application is stable at release, however there are a few problems still to be worked out, such as the application logging the user out if the user switches to another application and back again, this also sometimes causes an infinite log in screen that requires the application to be removed from the application history of the device and restarted.

## Conclusion?

While most of the planned features have been included and the application is mostly stable, several more advanced features are missing and some bugs are yet to be resolved. The team feels that given a little more time that the application could be made to be completely stable and any missing or additional features could be quickly integrated.