Mobile Client Report

Android studio was used to develop the android application. Android studio is an IDE designed and developed specifically for android app development. Also with a use of a gradle based build system you have the option to preview a layout on multiple screen configurations while editing. With the use of extensible Mark-up Language (XML) it aided the development of the screen layouts. This was beneficial as it gave the ability to meet the usability and functional requirements of the application. To display the screens an android emulator was used, and an android phone was used to make sure the application was running successfully as well as provide a visual feel on how the app would feel.

Nodejs was used to connect the android app to the server. Node.js is an open-source and cross-platform JavaScript runtime environment. To connect to the server numerous methods were used, one being the Android volley library. This library was used because it is an HTTP library that makes networking for Android apps easier and most importantly faster. Android Volley allowed us to do automatic scheduling of network requests and helped support for request prioritisation. Furthermore using Android Volley we :

* Sent a simple request using the default actions of Volley
* Set up RequestQueue
* Made a standard request to send a request using one of Volley's out-of-the-box request types (raw strings, images, and JSON)

Design and Implementation

The User Experience was very important in the design of this application, so this aspect of the application was carefully considered. The navigation around the application was simple enough for the user to understand. It was decided that the application should follow a consistent design throughout, this meant determining a suitable colour scheme which will be used in all the pages of the application. A navigation menu is used for simplicity reasons to save space on the pages of the application. In the application there are 3 main sections, Home page, File options Page, Help page.

Home Page

The home page consists of two main parts including a list which represents all the files and an upload button. To be able to achieve a working list view, we needed to design what each entry of the list would look like, and to design a custom adapter to fill the list view with the data received from the server. The code for the list uses the Android Volley library to send a GET request to the server to request for a JSON response. This response was then separated into 2 Array Lists, chosen due to the simplicity of adding and removing elements. The contents of each of these Array Lists were then copied to 2 String arrays, which were passed to a custom adapter. This custom adapter takes the data passed and assigns it to the correct parts of the list view. Lastly, an onclick listener was used for each entry of the list view to enable the user to access the file options for the file selected.

The next main part was the Upload button, an onclick listener was used so when the user clicked the button a menu of all available file locations would pop up in the application. Android has a library which we were able to use which enabled quick and easy navigation and selection of files in numerous locations such as camera, recent, download and others. When a file is selected the file name is passed to the upload function…(finish)

File Options page

When a file has been selected from the list view on the home page, a new fragment is called which will display the name of the file and two buttons remove and download.