Project Prime : Mobile Client

# Implementation:

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 grade 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 Android volley was used. This library was 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

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, Edit Page, Help page.

1. Android Studio
   1. Java
   2. XML
   3. NodeJS
   4. Android volley!!!!
2. How?
   1. File listing
      1. Use listview and get file info ( filename and author/upload date)
      2. Print these info into the list
      3. Using volley and JSONArray request to get the json from server
         1. Iteratively go through each of the json objects to get the information (last upload and file name).
         2. Put the information in separate array lists
         3. Print into the listview
         4. Needs request queue to ensure the request is done
      4. On click open that file in file editing
   2. File Editing
      1. Parse Files onto app as strings
      2. Write strings into files on the server
      3. Use file scanner to write contents of file into a text box
         1. Make text box editable
         2. Can save the text box contents into the previous file
3. Success Criteria
   1. Ensure all files from server are accessible
      1. Maybe only able to edit text documents?
   2. Ensure ability to delete files from mobile
   3. All changes update the server files
   4. Any collisions must be dealt with
      1. Decide which takes priority?
      2. If there’s a clash discard the mobile?
      3. Upload as a different title?

# Design:

1. Main objective: Productivity
   1. 4 Main parts of the application
      1. Home page
      2. Edit Page
      3. Help Page
         1. Provide Instructions for user, clear up any possible issues
      4. Navigation Bar
         1. Just have links to the previous three pages mentioned
         2. Use navigation tray design
   2. Limit of 2 pages for the main feature
      1. Home page – Display all the files
      2. Edit page – Edit the files selected from the home page
   3. Why?
      1. Simplicity
      2. Efficiency
      3. Aim of the app is to work with files – over complicated pages would be a hinderance
   4. Colour Scheme
      1. Sticking with default android colour scheme
      2. Only sticking to a single colour scheme – don’t want an overly complex design
2. Home Page
   1. Consist of listview
      1. First line: File name
      2. Second Line: Author
      3. Click on list entry to edit
3. Edit page – Main challenge
   1. Text box where file will be parsed
   2. Changes made in the text box
   3. Press save to use file writer to write into the file the file witch changes
      1. Use reader to write the data onto app
      2. Use writer to write data on app page into the file again