COP290 - Assignment 0: Design Document

Aditi(2014CS10205), Ayush Bhardwaj(2014CS10091), Nikhil Gupta(2014CS5140462)

January 24, 2016

1 Overall Design

Our overall design of the app would consist of three main activities. The first one would be the splash screen, which would be displayed for a fixed amount of time. The next activity is the main page which is the actual interface between the user and the server. The third screen would be the confirmation page, which appears if the data is posted successfully. Other than the activities, a separate volley module, and a checker class has been included.
As per the requirements of the assignment,

- Different classes have been defined in separate files to maintain modularity in code.
- Volley has been used to interact with the server.
- Different layouts [6] have been made for tablet and mobile phone view for all the three activities.
- Various resources, such as strings, images and styles, have been included in separate files.
- **Doxygen** has been implemented to create an HTML documentation of the application structure.

2 User Interface

2.1 Splash Screen

The files activity_splash.xml and content_splash.xml define the layout of the splash screen. It consists of a big image that contains the logo and name of the application. The images for phone view and tab view have been included in drawable folder. An animation [1] has been applied on the app logo to make it grow and shrink. Below is the screen shot of the splash screen in tablet view. The layout for tab view is in layout-sw590dp folder.(This is picked when smallest width exceeds 590 dp)

2.2 Design Of Main Activity

- The layout of main page has been defined by activity_main.xml and content_main.xml files in the layout folder.
- For tab view, the layout files are in layout-sw590dp folder. The application automatically picks the layout file in this folder, as soon as the smallest width available to the app exceeds 590dp.
- The page consists of a total of five data fields, including team name and name & entry numbers of team members to be filled in by the user to register his team.
- The '+' Button [5] has been included which on clicking shows the data fields for the name and entry number of the third member of the team.
- The '-' Button has been included which on clicking hides the data fields for the name and entry number of the third member of the team. It is visible only when the '+' button is clicked.

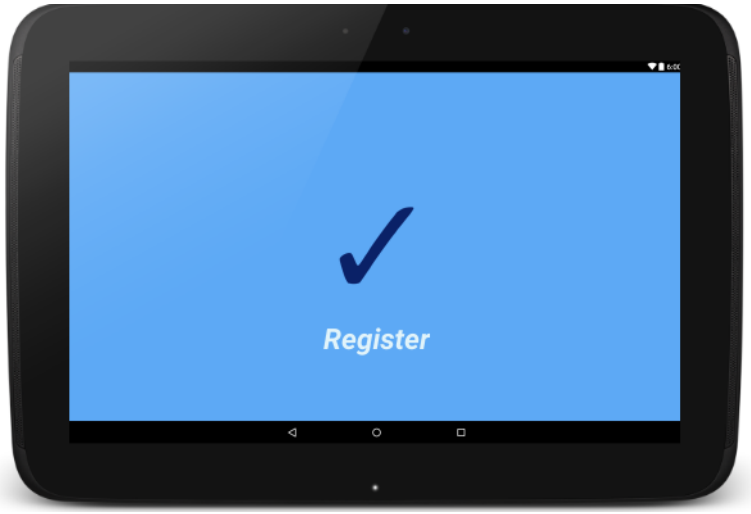

Figure 1: Splash Screen on Nexus 10(Landscape)

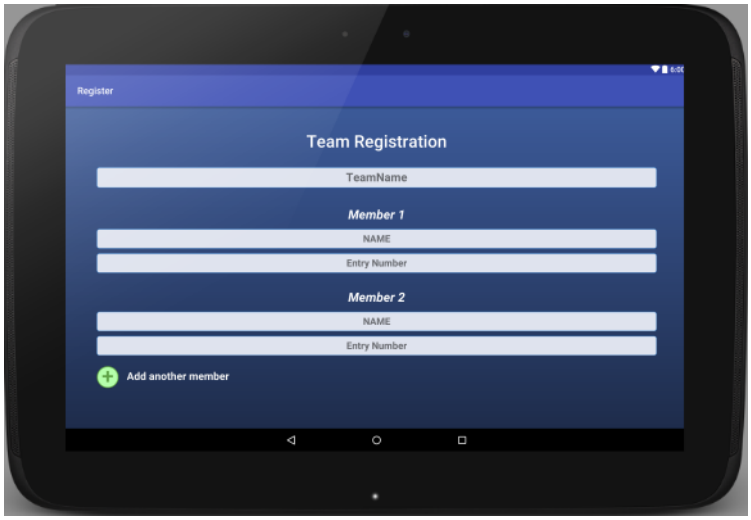

Figure 2: Main Screen on Nexus 10(Landscape)

- The 'Register' Button handles the data entered on being clicked.
- A text block for displaying the server response and the appropriate message has been placed above the register button.

The following are the screen shots of the main page on different orientations as well as screen sizes [2]:

2.3 Final Screen

It appears when the correct data has been successfully posted on the server.It displays the details of the team entered by the user. It also includes a 'Register another team' button, that takes the app back to the main page, so that the user can register another team if needed. Below is the screen shot of the final screen.

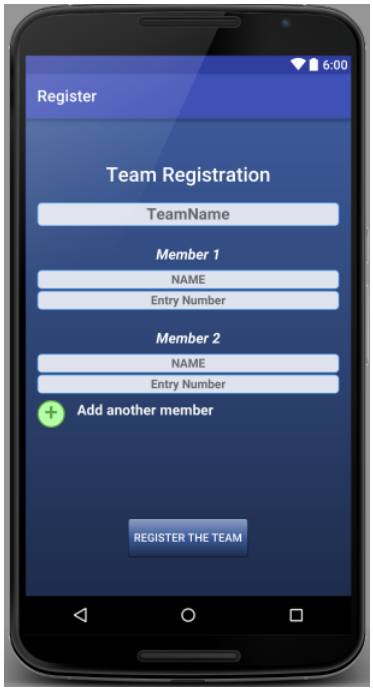

Figure 3: Main Screen on Nexus 6(Portrait)

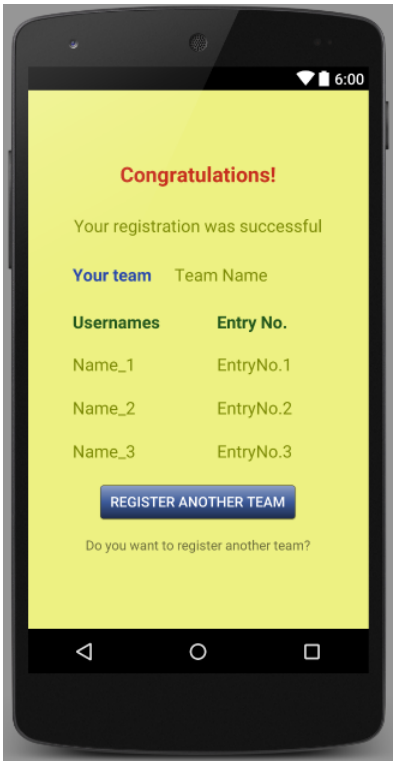

Figure 4: Final Screen on Nexus 5(Portrait)

3 Implementation Details

3.1 Splash Class

This class contains the various auto-generated methods to initialize an activity.
A handler has been added to execute the method Run() of a runnable after the fixed time specified by a private variable, splash_time.
The Run() method creates a new Intent to start the main activity.

3.2 AppGlobal Class

This class [3] has been made to make an object of the application, so as to create a global request queue for the volley push requests. The same queue is used to add requests each time the user clicks the Register button on Main Page.

3.3 Check_constraints Class

This class has been made separately, to perform all the required checks on the validity of the data entered by the user in the main page of the application. It includes the following methods:

- **Boolean NAME** : It checks for the validity of Name being entered. Name containing only alphabets are allowed.
- **Boolean EntryNo** : It checks for the validity of Entry Number being entered.Only those Entry Numbers in IIT Delhi format are valid.
- **Boolean Diff_EntryNo** : It checks if all the Entry Numbers being entered are different else returns false.

3.4 Main Activity Class

The main activity class handles the dynamics and events of the main page. The following methods have been included other than the standard methods:

- **Add** : This method handles the OnClick event of the '+' button on the main page. It changes the visibility of the name and entry number fields of the third member and the '-' button to Visible and its own visibility to Invisible.
- **sub** : This method handles the OnClick event of the '-' button. It changes the visibility of the fields for details of the third member back to invisible.'+' button appears again.
- **SendData** : This is method to handle the OnClick event of the 'Register' button. It first uses an instance of the Check_constraints class explained above, to check if the data entered is valid. Then, it initializes a new StringRequest object, a class in the module VOLLEY to send the data [4] to the server. The request is added to a new Request Queue. OnResponse and Error Listners are also handled via this StringRequest instance.
- **fadein** : This method is used to implement an animation on certain text fields to slowly bring them into view.
- **fadeout** : This method is used to implement an animation on certain text fields to slowly fade them out of view.
- **pop** : This method is used to implement an animation on the error messages that appear on the screen.

3.5 FinalScreen Class

Along with the auto generated protected method onCreate it contains the following methods:

- **Register** : It handles the onClick event of the Ŕegister Another team buttonánd directs back to the Main Screen.
- **grow** : It starts the animation of the 'Congratulations' text. It is called in the onCreate method.

3.6 Interaction Amongst Classes

- **Splash Screen to MainPage** : An instance of the Intent Class is instantiated to start the MainActivity layout after a few seconds delay.
- **Check_constraints Object in MainPage**: Instance of Check_constraints checks the validity of all the entered inputs one by one. Only if all the inputs are valid, StringRequest object is created and added to the request queue. Else an error message is displayed.
- **Data to Final Screen** : Final Screen is only displayed when a successful response comes from the server. Flow of information from Main Screen to final screen takes place via Intent class object. This information is used to print the Team Details on the final screen.
- **Final Screen to MainScreen** : An instance of the Intent Class is instantiated to start the MainActivity layout upon clicking the Register Another Team button.

4 Error scenarios

- **Team Name** The first check is on the team name. If it is an empty string then an error is displayed.
- **Member names** The second check is on the names of the team members. If they are empty or contain any number, then a corresponding error is displayed.
- **Entry Numbers** The third check is on the entry numbers. If any two are same or, they do not adhere to the format : 20nnxxnnnnn(n denotes number and x denotes alphabet), then a suitable error message is displayed.

5 Future endeavours

- Keeping a record of all the team names that have been registered since the installation of the app, to perform local check of validity of data before sending it to server.
- Adding How-To-Use instructions on Splash screen to enhance first time experience of users.

6 Source Code

The source code of the project is maintained in the following repository:
https://github.com/aditi741997/Assgt0-Final.git

References

[1] Adding animations. http://www.tutorialspoint.com/android/android_animations.htm.

[2] Adding figures in latex. https://www.latex-tutorial.com/tutorials/beginners/latex-figures//.

[3] Android app object. https://www.mobomo.com/2011/05/how-to-use-application-object-of-android/.

[4] Android volley tutorial. https://www.simplifiedcoding.net/android-volley-post-request-tutorial/.

[5] Custom buttons and gradients. http://angrytools.com/android/button/.

[6] Supporting tab and phone view. http://developer.android.com/training/multiscreen/screensizes.html.