# COP290: Moodle App

Abhijeet Anand (2014CS10202)
Kapil Kumar (2014CS50736)
Sourav Das (2014CS10258)
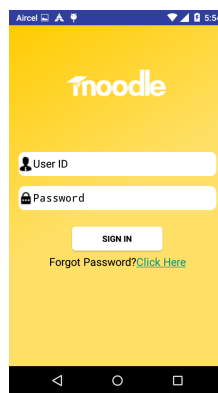
September 16, 2018

# 1   User Interface
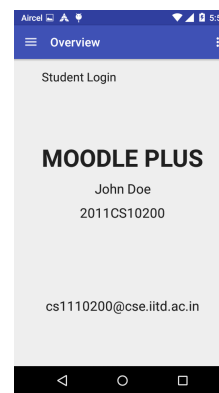
- **Screens visible to the user:**



Splash Screen



Login Screen



Home Page

– **Splash Screen:** The app starts with a splash screen which displays the logo of Moodle App . The next activity can be opened after 4 seconds. It also displays a welcome toast greeting the user.

– **Login Page:** The app starts with the introduction page which is basically a splash screen. It displays the logo of our Moodle app.It automatically closes in a few moments and starts the LoginActivity Class. The LoginActivity page provides an interface for users to enter their credentials to log in to the app. The two text fields represent the user-id and password to be entered. Text-watcher is implemented on both the fields. Initially both fields contain placeholders of name and password and their respective symbols. When we start typing into the fields the Text-watcher comes into play and the placeholder fly off the area to make space for typing. Upon pressing the login button it checks if any of the text fields are empty. If any one of them is empty then it reports an error to supply valid username and password. If the credentials are valid then a Volley request is generated and the details are sent to the server for verification. The response is a JSON object containing information of success of the login attempt. If the login was successful then the boolean value indicates it and the other data items represents the various details of the userlike user name, email, password, id etc. This set of information is sent to the activity containing the home page.The Home Page opens.

– **Home Page:** This Page allows us to enter the details of first team member, then validates the data entered and proceeds to second member details page if data is valid.
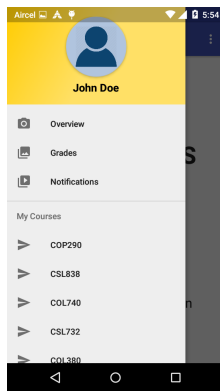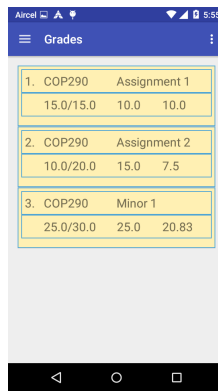
Navigation Menu          Grades Page          Notifications Page

- **Navigation Menu:** This screen is the central navigation window and provides access to all the menu items. The top fragment contains a field to place the image of the respective user. Below it the name of the user is displayed for recognition. Below it is the link to navigate to the various parts of the app and expand its functionalities. The first group contains options to navigate to Overview, Grades and Notifications. They lead to their respective pages. Another group contains a list of courses the person is in involved in, be it a student or a Professor. The individual courses provide access to the details of that particular course. It consists of the grades of the course, the assignments related to it etc. In the bottom a logout link is provided to log out of the app. Here the cookie will reset its value until a new person successfully logs into the app.

- **Grades Page:** In the navigation bar we have a menu item called "Grades" which, on click, displays the overall grades till now in all the courses. The Grades fragment displays the course Name, Assignment number, Total Marks, Weights of a particular course. This task has been implemented by sending a request to "domain/default/grades.json" API which sends a JSON Objects consisting all the grades details of a particular assignment.

- **Notifications Page:** Notification menu item in the Navigation Menu, on click, displays all the notifications related to that particular user, which can be of any one of the course he is reg-

3

istered like it will display a new notification if someone posts a new thread. It also displays the date and time of posting of that thread along with the name of the user who posted it. This task has been implemented by sending a request to "domain/default/notifications.json" API which sends a JSON Object consisting of all the details of notifications.
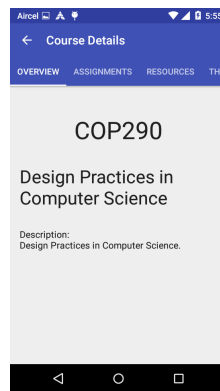


Course Details Page

– **Course Details Page:** On clicking on any particular course displayed in the navigation window, i.e. the courses user is registered for the particular semester, it starts a new activity consisting of various details of that particular course. It consists of 5 tabs/pages:

* **Overview** It displays general details about that particular course as received from the API.

* **Assignments**: This tab displays the current assignments which are due or displays "No Assignments. Yay!!" if there are no upcoming assignments.

* **Resources**: It allows the user to upload a resource/file to his/her account. But we don't have any API for uploading so it is just a static page.

* **Threads**: This tab allows a user to post a new thread related course and also allows the end user to view all the threads posted by other users related to that course.

∗ **Grades**: This is the last tab related to a particular course whose functionality is similar to the "Grades" but the only difference is that it displays grades of only that particular course.

- **Animations, buttons (enabled/disabled under what conditions)**

  – Animations begin with the creation of the app. The splash screen provides a smooth transition to the Login Page. Animations are activated whenever a new activity is started, The login button is disbled unless a legitimate entry is placed in the placeholders.The navigation bar is animated and it sweeps out whenever the user swipes around the corner and drags it outwards. The navigation window also pops out when the button on the menu bar is clicked. Animations are also activated when the user presses the back button.

  – The login button also implements a TextWatcher and shows a transition of the placeholder with an animation.

- **Actions performed when a user enters information, presses a button/icon etc.**

  – When the user clicks on the name or entry number field, then the placeholder text floats upward and the entered text is displayed below it.

  – When the user presses "Log In" button on any page, our app first validates the data. It proceeds to next page only if the data entered is valid otherwise it displays a red coloured error message below the wrong entered data.

## 2   Implementation Details

- **Organization of user information:** Organisation of user information The app starts with a splash screen showing front page of the app. The next activity is the login activity containing fields to enter information into it. The information is verified and the app proceeds according

to whether the credentials are correct or not. The next page is the home page containing details and implementation of the navigation bar.

- Methods to verify the user information:

  – Validate Details: In validation process we have generated a valid pattern for the Entry Numbers using java regular expression and matching the pattern of this with the user.

The code for the project is being maintained in this repository: *https://gitHacker@bitbucket.org/ app.git.*

# 3 References

[1] http://developer.android.com/training/volley/index.html
http://stackoverflow.com/questions/16680701/using-cookies-with-android-volley-library
http://developer.android.com
http://stackoverflow.com/
http://stackexchange.com/
https://www.sharelatex.com/
https://en.wikipedia.org/wiki/
http://youtube.com