

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

GitHub Username: Jarraro

Uni Events

Description

University students usually miss important events because they don't read every announcement.

This app allows students to stay informed about new events and gives the chance to their university to easily reach them and push them notifications based on their major or interest.

Intended User

This app targets university students.

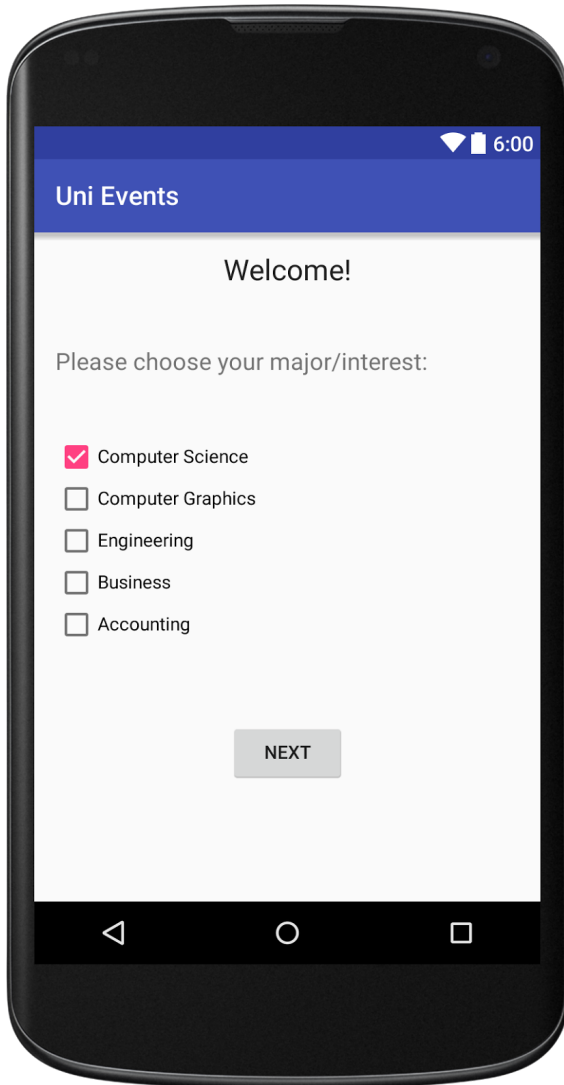
Features

- Downloads events from server.
- Saves student's information and events.

- Receives notifications via GCM.
- Shares events through intents.

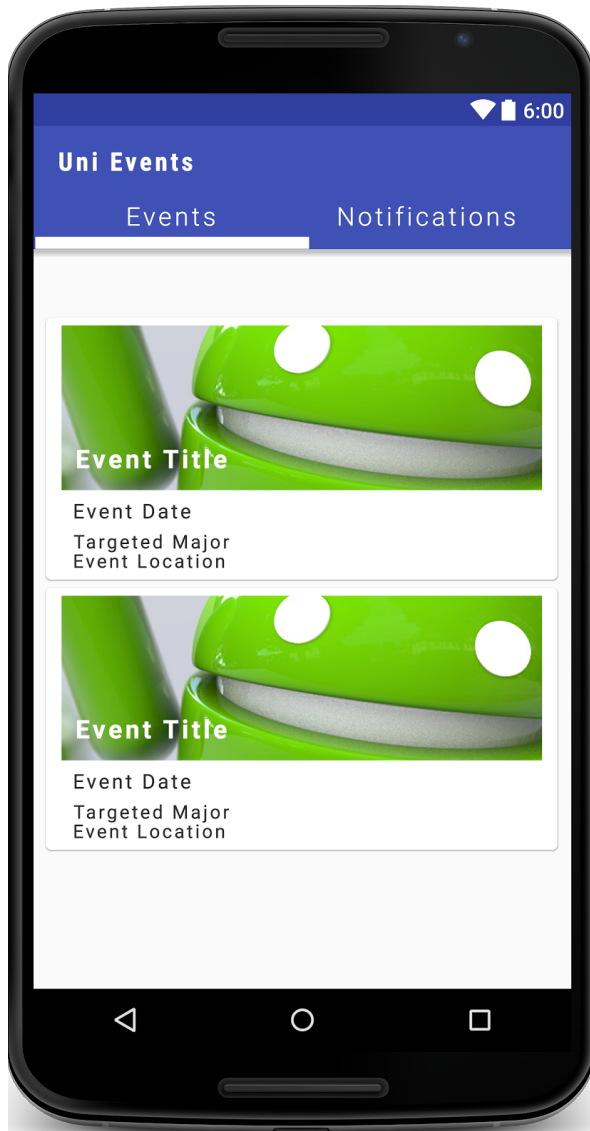
User Interface Mocks

Screen 1: Welcome Activity



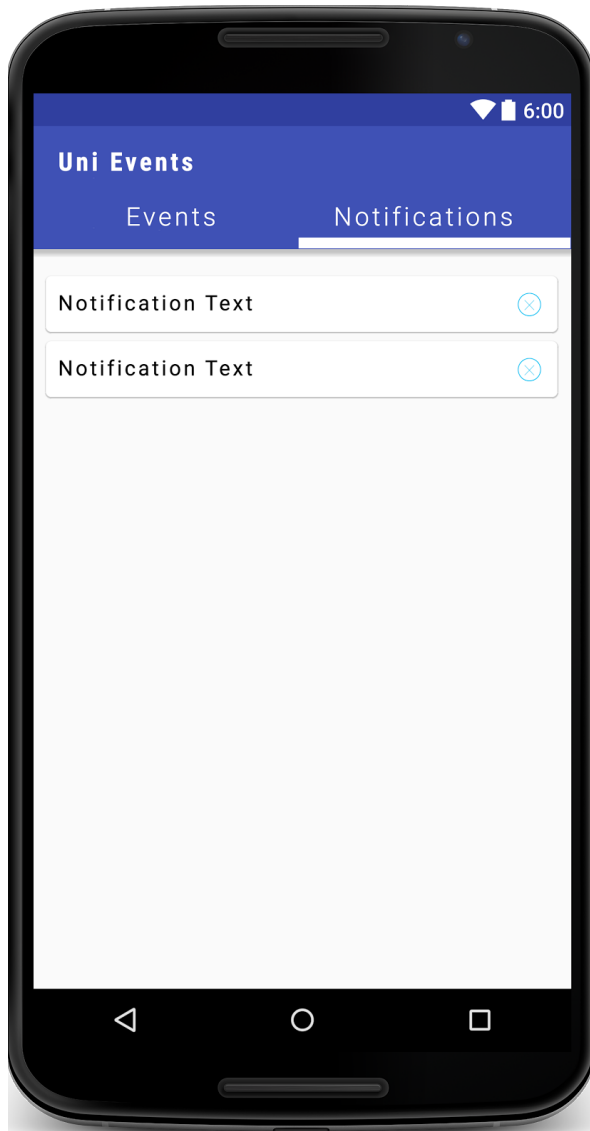
This screen appears only once during the first use of the app.
Student Assigns their major/interest.

Screen 2: MainActivity-EventsFragment



This is the events fragment/tab, each list item has an image, title, date, Targeted major and event location.

Screen 3: MainActivity-NotificationsFragment



This is the Notifications fragment/tab, each list item has an notification text and a dismiss button to delete the notification from database.

Screen 4: DetailActivity



The detail activity shows the event details and an option to add the event to Google calendar, and share event to friends.

Screen 5: Tablet view - MasterDetail layout



For tablets, the Activity contains a Master/Detail layout

Key Considerations

How will your app handle data persistence?

App saves settings and student's interest/major in a SharedPreferences file. And saves events and notifications to the database.

Describe any corner cases in the UX.

App provides a widget that shows future events only.

App uses SwipeRefreshLayout in the Events Activity "MainActivity".

Describe any libraries you'll be using and share your reasoning for including them.

Glide - for loading images

Next Steps: Required Tasks

Task 1: Project Setup

- Create two modules, one for app and a Google Cloud Endpoints module
- Configure app dependencies.
- Add needed permissions to the manifest file.

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
 - MainActivity contains two sections/tabs, one to show events and another to show previous GCM notifications.
 - Event section/tab shows a SwipeRefreshLayout with list/grid containing CustomViews that briefly describe each event.
 - GCM section/tab shows a list of notifications and gives the user the option to dismiss any notification.
 - MainActivity contains an AdMob Banner Ad.
- Build UI for DetailActivity
 - DetailActivity shows event title, date and time, location and image if exists.
 - The user should be able to easily share event or add it to Google Calendar.
- Build UI for widget showing future events.

Task 3: Configure and deploy GCE

- Create needed classes in the GCE module to send GCM regId to server.
- Create needed classes to load events from server.
- Deploy the module module.

Task 4: Loading data from server and saving it

- Create a database and all related classes.
- Load events from GCE module via an AsyncTask.

- Parse and save the returned JSON file into database.

Task 5: Handle messages received via GCM

- User can enable/disable notifications.
- Save received notifications to database.
- Set the PendingIntent for the notification.

Task 6: Implement the app widget

- Create and configure the widgetService and test it.

Task 7: Tests

- Project contains connected tests to verify that the AsyncTask is loading events from the server.
-