UL Pathfinder Manual

Group Members: Aoife Power 15140679

Daniel Keeley 15189082

App API Level: minSdkVersion 17  
 targetSdkVersion 25

Step by step guide:

1. Please enable location services for the app
2. Load up UL Pathfinder
3. You will be prompted to enter an ID, here are some you can use to test (15189082, 15172899, 15140679) you can go to <http://www.timetable.ul.ie/tt1.asp>

and verify for yourself it is correct.

1. Click Submit
2. You will be shown a colour coded timetable with your module code, time and building location with room number.
3. You can click on any of your modules and this will initiate the MapActivity giving you GPS directions from your current location to your class.
4. Enjoy never missing a lecture again.

Description:

UL Pathfinder is a timetable android application with built in google maps GPS for directional aid. Our aim with this application was to make a timetable application that any student could use to view their timetable and if unsure of class location, use the same application to get directions to the building. While many timetable apps do exist, as well as a building code translator, none of the available applications have the built-in map feature UL Pathfinder has.

Application Features:  
Timetable: The user will be greeted with a basic input page on initial run of the application. The user will be prompted to input a UL ID number that the application will then search for on <http://www.timetable.ul.ie/tt1.asp> to get the students timetable. Once searched for, the app will then bring up the users timetable for them to view.

Maps Activity: If the user clicks on a class on the timetable, the application will get the building code from that class, and search for it in the hard-coded building codes within the application. When it matches one of the hard-coded building codes, it will then divert to a new map view, showing the users location and the buildings hard coded longitude and latitude.

Possible additional features:

The application can be developed further as we have several ideas for additional features that are possible to improve the application.   
  
The use of notifications i.e. setting application alarms for classes using the Google map feature to give the user the minimum required time to get to class.

Editable Timetable i.e. in the case of labs and tutorials, while lectures would be static, the user would be given the option to move labs and tutorials to a more preferred time matching the timetable of the module.

Search for Module Timetable: The user could be given the additional option to search for a module code rather than a student timetable.

Contribution of Group:

Aoife Power (15140679): I worked on and developed the xml of the application, as well as the MainActivity, the displayTimetable, the CheckWifi and the getTimetable java files.   
I created the main activity where the user is prompted for a University of Limerick ID. When a valid UL ID is entered that matches the input pattern, it then checks if the user has internet access using CheckWifi. Only if both the ID is valid and the user has internet access will the application will then use JSoup to

search for the UL ID on the timetable.ul.ie website. The returned data is placed in a Document object in our SQLlite database, and parsed so the data is easily retrievable. The displayTimetable java file is then called to search through the data and format it in a simpler and easier to read format. Which is then applied to the appropriate segments of the timetable that matches the day and time of the class and gives each module code a unique colour to make the classes distinguishable between each other.

Daniel Keeley (15189082): I worked on all the location services in the app, including GetGPSLocation, displayTimetable and MainActivity . When learning locations I first learnt how to get current location, then I learnt how to update it zoom in on it etc. Then I implemented markers on current locations and some test locations. After that I finally used Polylines to draw a line between these two markers. All the building coordinates were hard coded from google maps and when the building code contains the code for a coordinate it is displayed.