EuroTrip

Android project for Linnaeus University

Authors: Karl Mösenbacher
Andree Höög

Table Of Contents

General description

The application EutroTrip is a quiz game where the inspiration has been taken from the Swedish television program “På spåret”. The game consists of 5 “trips” and on each trip the player has fifteen minutes to find out the correct answer on where we are going in the world. At start the player is given a weak description which is worth 10 points on which city we are looking for. On every third minute the description is getting easier and the points given is then reduced by 2 and the game view, which consists of a map is then zoomed closer to the location until points given is reduced to 0 when the time is out. The player is then, if he or she presses the button “next trip”, given a new “trip” with other descriptions and a new city that we are looking for. If the player knows the answer on any description and answers correctly, he or she is then given the points that the description is worth. The map is then zoomed to the start city and end city of current “trip” and the route is then displayed on the map. If he or she answers wrong no points will then be given and the player will be given another “trip” instead. When the player reaches 5 “trips” he or she will be asked to save he or she progress. If the player doesn’t know the answer we are looking for , the player is not able to press next question but he or she may in that case answer wrong or wait until time has expired. The player will only be able to “move” the map whenever he or she has answered correctly.

Features

If a player wants to start a new game, he or she can then press the new game button in the main menu and the game activity will be displayed. A trip is then randomly selected from firebase database where all trips and descriptions are stored. If same “trip” comes twice during gameplay a new “trip” will be selected from the database. Player results will be saved to SQLite database and will be stored on the device. The player has the ability to close a game and resume it whenever he or she wants. If the player minimizes the game a notification will appear on top of the screen with the current time left on the “trip”. The player has then the ability to press the notification to return back to the game and continue playing.
Every description on each trip will be shown on top of the screen in a small field where the player can scroll it and if the player thinks the text view is to small he or she can then press the text view and a dialog will be shown with the whole description instead.

In the settings menu the player is given several features. If the player wants to do some settings about the game, he or she can then press the settings button in the game menu and the settings activity will then be displayed. In settings menu he or she can choose if he or she wants menu music, vibration when a button is pressed or sound effects when answered correctly, right or when a game is finished.

If the player wants to se past results the user can then press the top-score button in the main menu and the top-score activity will be displayed. A list of past results will be shown and if the player is “long pressing” a row in the list a dialog will appear. In the dialog the player will be given several options, one if he or she wants to share pasts results to others

via Facebook, SMS or similar social media programs, two if he or she wants to view a detailed scoreboard on each point they gained on every “trip”. The third option is if the user wants to clear the table on all results.

Problems during project

During the project the main question was how to store all descriptions and trips so it could be easily available during gameplay. Because every trips consists of large files of text we didn’t want to store it on the device, and we also wanted a database that we could update with more trips in the future, instead we wanted to download every current trip during gameplay. The solution we found was a database called firebase which has a web interface where you easily can manage and change the database. The database is then connected to a java class called “Trip” in the android project. By doing this we then have the possibility to easily download every current “trip” during gameplay and get access to all questions etc. on a specific “trip”.
All “trips” tables have their own URL for downloading coordinates that are shown when answered correctly.
All KML files are stored in a dropbox account with a customized URL for direct download to the application.

During the project we also faced some kind of performance issue in the application that we believe came from all the threads that we are using and sometimes bad implemented notification and service. We also think that we have too much processing on the main thread in the application.

Problems left unsolved and possible solutions

One issue that´s not solved is that we are now only playing music in the game menu. From start the meaning was to play music in the game menu and the top score activity but due lack of time we focused having music that is paused and resumed when moved away and returned to the activity. One solution we had in mind is that creating a service that will handle the music and in some way code the application so it would know which activity that was current.
The threads that we are using in the application we want to implement better so that the program can operate more effectively. The solution of this matter is to get a better understanding on how threads works and where they should be implemented for best result.
In the game play activity we want to clean unnecessary code that works inefficient and create more smaller classes so that not all heavy work runs in only one activity.

One tiny problem that we want to solve is that we want to translate the game into English so many more users can play the game, but above all understanding all descriptions and puns in the game.

Possible future work, additional features

For future work we would like to add chromecast support so that the user can play the game on his or her TV and use his or her device as input. On the television all the questions etc. will be displayed.
For keeping the game active and fun as long as possible we would also like a better and easier way to implement more questions to the firebase database so that the game lasts longer.
We would also want questions with pictures and also music so that we have a great mix of questions for the players.
And last we would also want the application to be able in landscape mode.

Instructions on how to compile/run/use your application

To run our application, you only have to follow these steps:

1. Start the application by pressing the icon
2. Start a new game by pressing “new game” button.
   2.1 To get first question press play
   2.2 To answer the question, write your answer in the text field in the lower part of the screen.
   2.3 For submitting your answer press the submit button in the lower right corner.
   2.4 Press the next button for next question
   2.5 If you want to view the trip or the explanation and don’t want to scroll the text for each trip question, then press the text view for a pop-up window.
   2.6 Repeat 2.1-2.5 until game is finished.

3. View top score by pressing the top score button in the game menu
   3.1 Press the menu music button for music
   3.2 Press the vibration button for vibrations in game
   3.3 Press the sound effects button for sound effects in game
4. To view top score press the top score button in the game menu
   4.1 For more options long press one row, a pop-up dialog will appear.
   4.2 For sharing your results press the share result button in the pop-up window
   4.3 For detailed list, press the view detailed top score button.
   4.4 If you want to delete the whole table, press delete table button.

To use our our application, the user must have an android smartphone with at least 26 mb of free space and run on at least SDK version 21. The device must be connected to internet for downloading data. In order to display the google map the user has to have a valid API key and Google Play service installed on the device. We recommend that the user have a pretty high-end smartphone with the latest updates in order to play smoothly. This game wont run properly in the emulator.

References to external resources (documents, books, projects, source codes) used in the project

To implement the firebase database, we had to read the documentation that firebase offered on their homepage. The documentation can be found on the following URL:

- https://www.firebase.com/docs/android/

Other functions in the game like the KML parser has been taken from assignment 3.

To implement the SQL database, we have taken code from assignment 3 and the phone task.

Other functions like Google Maps, Services, notification and threads we took help from android developer's webpage.

- www.developer.android.com
- http://developer.android.com/intl/zh-tw/reference/android/app/Service.html
- http://developer.android.com/intl/zh-tw/reference/java/lang/Thread.html
- http://developer.android.com/intl/zh-tw/reference/android/app/Notification.Builder.html

With inspiration from different kind of forum posts from stackoverflow.

- http://stackoverflow.com

Use Case

Following figure shows a very simple use case.