

# Comp 319 Programming Assignment #1

## QuizApp

Instructor: Seyhan Ucar

Deadline: 23 July 2017, 23:59

### 1 Objectives and Definition

The objective of this assignment is to make a simple quiz application, namely QuizApp, on Android platform. To implement this assignment you need to cover Android Studio, Activity, Multiple Activities, Intent, States of Activity and usage of Handler. QuizApp is a simple game that users can test their knowledge on different questions. Questions are randomly selected from the application embedded question pool. Each question has answer pool where only one of the answer is correct.

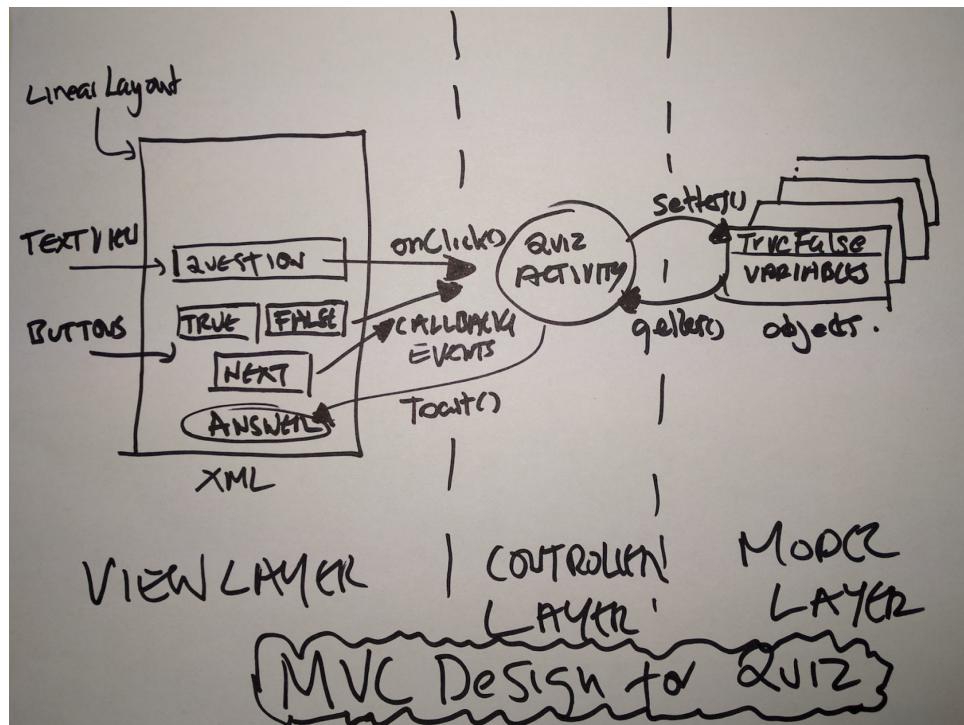


Figure 1: MVC Design of Initial Quiz Application

The initial Android application where a single activity with a true and false button is shown in the lecture (source codes are attached to this assignment on Blackboard). Fig. 2

demonstrates the single activity of QuizApp. On the other hand, Fig. 1 represents the current Model-View-Controller (MVC) structure of the application. However, this code is poor in terms of object oriented design and supporting for multiple activities. In this assignment, you are responsible for correcting this application and improving the capabilities of the application based on defined requirements.

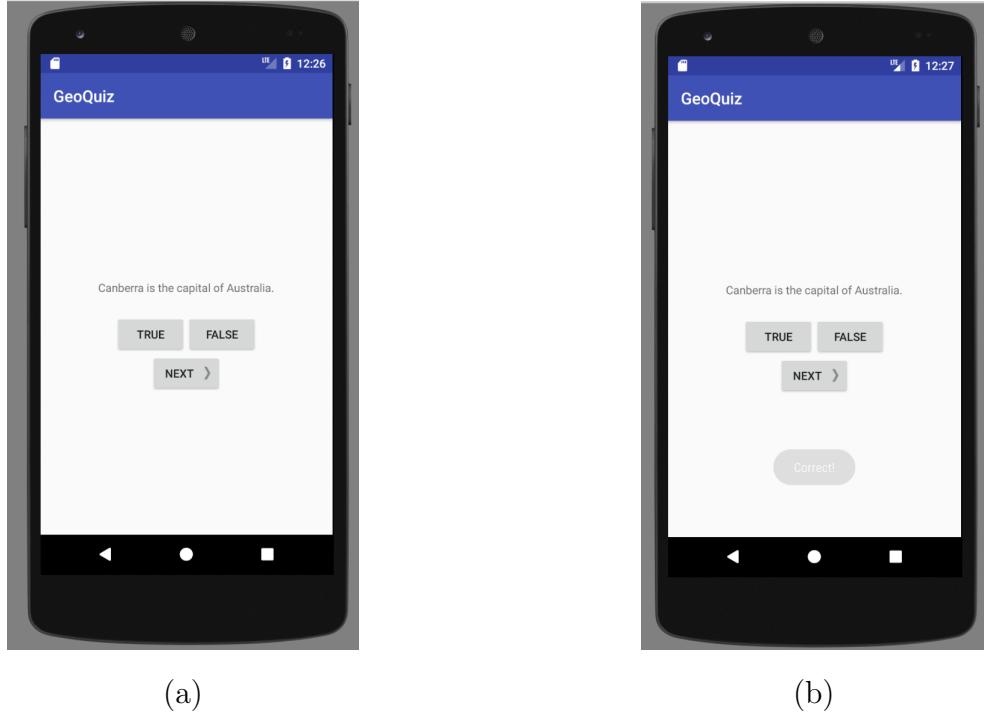


Figure 2: QuizApp Single Activity

As a part of your work, you need to change this single activity into multiple activities where multiple activities use Intent to communicate. Moreover, you need to draw the MVC of your design as a part of the assignment. You can refer to the initial MVC design of the Quiz application in Fig. 1.

The Quiz App has different steps of execution that are as follows. When the application is started by the user, the user will enter his/her nickname. In the nickname entrance screen, you need to put your logo (any image as long as it is suitable) as image view to represent your application. This nickname represents user identity for one session usage of the app. After nickname entrance, the user presses a button, namely Start, and the game starts to represent a question with 4 possible answers. There are two timers where one of the timers counts up and hold the total time that user completes the quiz. On the other hand, another timer will be used for each question where each question has a static predefined amount of time limit and the user is required to answer the question within this time (this timer is visible to the user). If the question is not answered within the time then application passes to next question. If the user answers the question then s/he is not allowed to answer again or change the answer by returning back. When the user returns back to the question, the application shows the question with user's answer. If the given answer is correct then user earns 100 points. For each wrong answer, the user loses 20 points. When all of the questions

are answered, quiz app shows the total time that is elapsed while answering the questions and total points earned from the quiz.

Application has the following properties;

- You need to submit your MVC design of your app with the assignment. Each student needs to be aware of this MVC design. You can draw the MVC model by your hands.
- Application starts with an initial screen and user needs to enter a nickname. This screen needs to have your application logo as image view.
- After nickname entrance, users will be prompted a screen where a start button is placed in the center of the screen. The user can start to use the quiz app by pushing this start button.
- When the quiz app is started by pushing a start button, application holds a timer that counts up for the total time calculation.
- Application has a question pool which has at least 10 questions and students are required to find these questions and reasonable answers.
- Each question has 4 possible answers and students are required to find these questions and answers. When a question is represented by the application a timer will count down to show the remaining time for this question.
- When the time is over or user gives an answer to this question, the application passes to next question. If the given answer is correct then user earns 100 points. For each wrong answer or unanswered question, user loses 20 points
- When the user pushes home button or user gets a call, the application is suspended. When the user opens the application, the application needs to start from the point that was on before suspended. Test your application before coming to demo.
- When the user rotates the device, you need to save the current state to make Android remember the previous state that the QuizApp was. For this part of the assignment, do not concern about the responsive layouts.
- The game is over when all questions are shown to the user. When the game is over, the user is informed with the total point that he/she gains and the total points that elapsed for this session.
- Apart from these, animations, played music or custom user interface design are considered to be bonus points by your TAs. The bonus points will be determined later.

## 2 Requirements

The application needs to be designed in object oriented manner. User, Question, and may be other objects have to be separate classes. Students have to be aware of the underlying class model and the relation of classes. The application has to be designed in multiple

activities and needs to satisfy the defined objectives and properties in Section 1. Each of the objectives and properties is tested before submission. Students need to aware of the relationship between activities that are demonstrated on the lecture.

## **3 Submission**

Students need to submit their source code, the screenshot of the working demo and MVC design to Blackboard website. The missing or corrupted files, nonworking demos are not accepted. Students are expected to be aware of underlying MVC model, explain the code details or show working demo when it is requested.

## **4 Demo and Grading**

In grading, each group or student needs to demonstrate the working copy of the code with the submitted files. Each defined objectives and properties is evaluated separately. Students will be asked a different kind of questions related to the application, class model and application structure. Group members can take different points based on the evaluation. The doodle poll for demo session will be announced by TAs.