

**Institute of Vocational Education**  
**Department of Information and Communications Technology**  
**HDSE (IT114105)**  
**ITP4501 Programming Techniques for Mobile Systems**  
**Summer Semester 2020-2021**  
**Assignment**

---

**Submission Guidelines**

- This is an individual assignment.
  - The submission deadline of the assignment to is **11:55pm, 11 July 2021 (Sunday)**.
  - You need to submit all program sources (in a single zip file) to the Moodle website <http://moodle.vtc.edu.hk> assignment dropbox before the deadline. You are advised to upload your work at a time reasonably earlier than the cut-off date and time. Moodle allows multiple submissions, however, only the latest copy will be retained. You will receive **NO MARKS for LATE SUBMISSION**.
  - You are also required to give a demonstration. 40% of total marks will be deducted if demonstration is not done.
- 

## **1 Aims and Objectives**

- To gain experience in mobile application UI and program design.
- To gain practical skill of Android application development.
- To understand the constraints and limitation of mobile application and the ways to overcome them.
- To obtain knowledge on connecting the mobile device to the internet services and building a multi-tier distributed system.

## **2 Introduction**

In this assignment, you are required to develop an Android Application to play a Mathematic Game by downloading the questions from a server and then request player to solve them. This app will also record the result and corresponding time required to complete a game and use charts to show the history records.

## **3 Functional Requirements**

Listed below are the basic requirements of your application. You need to refer to the Local Database section for the database schema.

1. An activity which contain a button “Start”. When players touch this button, your app will download a mathematic question from an API server and then ask players to answer it.
2. Once players answer a question, your app will show players' answer is correct or wrong and then ask players to continue or quit the game.

3. If players want to continue the game, your app will download a question and ask players to answer it again.
4. If players choose to quit the game, your app will show the time spent on all the questions and the average time for one question.
5. A database which contains two tables. One table (**QuestionsLog**) stores the questions and your corresponding answers. Another table (**GamesLog**) stores the date and time you play a game and corresponding duration to complete a game. (You are required to build and initialise the database on Android mobile phone.)
6. Two activities (one for **QuestionsLog**, one for **GamesLog**) properly show the data stored in the local database.
7. Display a pie chart to show the results on how many answers are correct and how many answers are wrong in the **GamesLog** table, i.e., two sections in the pie chart. On each section of your pie chart, you are also required to show the number of correct answers and wrong answer correspondingly. You need to handle the rotation of mobile phone.

Note: You are encouraged to design and implement extra features. 10% of the total mark will be allocated on such additional functions. Refer to section 7 Marking Guidelines for more details.

#### 4 Local Database

The database scheme described here is an extremely simple one. Many fields are intended not to be included in order to reduce the complexity of this assignment. You are free to add columns and tables to the database to fit for your own needs.

**QuestionsLog** (questionID, question, answer, yourAnswer)

**GamesLog** (gameID, playDate, playTime, duration, correctCount, wrongCount)

#### 5 Questions JSON Server

You can obtain a question from the api server:

<https://2vtyxazuaa.execute-api.us-east-1.amazonaws.com/default/ITP4501AssignmentAPI>

and the data returned is in JSON format.

The sample JSON string returned is shown below:

```
{"question": "6 - 4", "answer": 2}
```

#### 6 Additional Constraints

- The UI of the mobile application must be produced with Android widgets such as **TextView**, **CheckBox**, and **Spinner** etc. Web-based UI is **NOT** allowed.
- The statistical charts must be produced using Android built-in graphics API such as **drawRect()** and **drawText()**. Using any other external drawing packages or libraries is **NOT** allowed.

## 7 Marking Guidelines

You project will be assessed according to the items below.

- Database initialisation
- Level of completion
- Correctness
- UI design
- Program design and implementation
- Program style and comments

10% of marks will be allocated to extra features not described in section 3. Each student can develop at most 3 additional functions such as animation effect or sound effect on the Android device or any other relevant and useful functions.

40% of total marks **will be deducted** if demonstration is not done.

**END**