**Mobile Application Development Assignment 2 Report**

*Student Name: Minh (Adam) Vu*

*ID: 19718114*

*Lecturer: Sajib Mistry*

1. Overview

The assignment 2 for Mobile Application Development unit for semester 2 of 2021 has tasked us to design an online tester to assess students from grade 1 to 3 on their math skills. With a system implemented with internal database to keep track of student’s detail and a self-signed certified server (for testing purposes) to provide a sequence of unique questions each time a student starts a test, the application provides a platform for students to do math tests online and have their results recorded.

1. Application Criteria:
2. Student Registration:

* The main activity of the application which shows the navigation panel to direct user to student registration by clicking the REGISTER button.
* Here, user can input first name and last name, both of which allows more than one word to correspond with people having distinctive names.
* User can add up to 10 phone numbers by clicking on ADD MORE button to have extra input brackets comes into view. Clicking DELETE button next to the entry will erase the view of the input bracket and its phone number content. An “inputType” flag is restricted to phone digits only when enter phone number.
* Similar code features are applied with emails with any “inputType” is acceptable.
* Profile picture of the student needs to be added with the student registration.
  1. *Importing information from Contacts:*
* User can choose to prefill the registration student’s input by selecting a contact from the device’s contact list and retrieving detail from the device’s storage. Permission to read contact need to be added to the manifest file to get detail.
  1. *Getting and saving the photo of the student*
* User can directly take photo from camera or pick from the device’s gallery or enter a search phrase to search photos online from <https://pixabay.com/> website as the student’s profile picture.
* Intent is used to request external app access to retrieve the photo. Getting photo from external storage needs read external storage permission added to manifest file.
* Input a search phrase would start an asynchronous task initially to download all the related image URLs to the search value and then, in the fragment, another asynchronous task started to download all the image’s bitmaps and present it in a grid layout with 3 photos each row up to max of 50 images. This grid layout view is presented as scrollable list through the recycle view.
* To send GET request from the app to the pixabay web server for retrieving URLs and images, an “APP key” from the website is given to my account on pixabay instead of using the lecturer’s key to prevent congestion of downloading images.
  1. *Edit or delete student information*
* Saved information on the student record is editable and could be removed as a whole student entry by selecting on one of the student entries on the navigation panel’s list of students.
* Editing a student info or deleting a student will make changes automatically to the student database.

1. View all the registered students

2.1) *View student’s Test History*

* Student’s test history can be viewed by selecting View Student History button on the main activity’s navigation panel. Then, History Activity is started and requires user to select a student entry to show his entire test records. The selected student entry is highlighted with cyan color to indicate the current student test history. Then list of test record which is scrollable to view only is shown underneath after the user clicks on FIND button.
* A student test record can’t be edit or removed unless the student entry who take that test is removed, which makes all the test records associated with that student name removed.

1. Math Test

* By selecting “TAKE TEST” button from the navigation panel, user is directed to “SelectTestActivity” which asks user to select a student entry before doing the test. Selecting a student entry on the scrollable list will highlight that entry with cyan color like in viewing the test history activity.
* Clicking on the BEGIN TEST button will direct user to the test screen where the test content is shown.
* QuestionBankServer is edit with variable “IGNORE\_URL” to false to check if the GET request contains the appropriate key values. If the values aren’t correct or parts of the key aren’t attached, the connection won’t be established.
* When the Test Activity is being created, an asynchronous task starts to send 10 GET request from the app to the question bank server to retrieve the question’s contents in JSON format.
* If the connection is established, each time a question is downloaded the async task will export the value to onProgressUpdate(). Here, JSON values are returned and then, are parsed to appropriate type. Then, the parsed values are stored in ArrayList or array[] of variables to access later when the user navigates through test’s question.
* With this design, only one downloading asynchronous task is needed initially to get all the questions before the student moves to that question. Hence, the design saves waiting time for student navigating through the test.
* Questions, and remaining time to solve are set to the TextView as String and integer.
* A custom MyCountDownTimer class, extending CountDownTimer standard class is implemented to keep track of remaining time and update the progress bar by every second. The countdown only starts when the question has been successfully loaded to view.
* A fragment is implemented to shown only the answer options to the current question. The fragment is dynamically set based on user input or when the timer runs out through replacing with a new fragment with updated parameters to dynamically inflate the view of fragment like a recycle view. This way ensures that different viewed number of options presents unique layout view.
* Currently viewed options (if not for short answer ones) can only be 4 or 3 or 2 at a time and an algorithm in decideLayout() is called whenever a question is downloaded successfully to decide the sequence of layouts.
* For short answer question when the size of options retrieved is 0, user needs to enter the result in an EditText view and clicks on Enter button to save the result, otherwise the input value won’t be saved, and the user won’t be awarded any mark for that question.
* Total score is updated and shown with the name of the student who currently takes the test.
* Move To Next Question button is unclickable if the next question has been downloaded yet or finishing the test won’t be possible if any of the question hasn’t been downloaded successfully.
* If the user can finish the test, then the score, date of test taken, duration of the test will be saved to the test’s internal app database and can hence, be viewed in test history later.

1. View test result and sending information

* User can only view the test result in History Activity screen. The list of the test records can be sorted from highest score to lowest and vice versa. This is done by insertion sort algorithm on the score of the test for the list of student’s tests.
* An email containing the entire test records of a student can be sent to all the student’s registered emails through (Intent.ACTION\_SEND). The email address will be validated before sending. Application sets the type of external app that is allowed to send the email to a specific type to restrict range of apps.

1. **REFERENCE LIST**

<https://developer.android.com/reference/android/provider/ContactsContract.CommonDataKinds.StructuredName>

<https://developer.android.com/reference/android/provider/ContactsContract.Data#CONTENT_URI>

<https://stackoverflow.com/questions/4366746/problems-with-contactscontract-commondatakinds-structuredname>

<https://stackoverflow.com/questions/40587168/simple-android-grid-example-using-recyclerview-with-gridlayoutmanager-like-the>

<https://stackoverflow.com/questions/38352148/get-image-from-the-gallery-and-show-in-imageview>

<https://stackoverflow.com/questions/6147884/onactivityresult-is-not-being-called-in-fragment>

<https://stackoverflow.com/questions/23778479/downloads-multiple-images-in-the-same-time-and-update-the-list-view>

<https://stackoverflow.com/questions/26865787/get-bitmap-from-imageview-in-android-l>

<https://medium.com/@aminullahtaj/asynctask-in-recyclerview-c82ff3ea5012>

<https://www.willowtreeapps.com/craft/android-fundamentals-working-with-the-recyclerview-adapter-and-viewholder-pattern>

<https://stackoverflow.com/questions/9061410/android-sqlite-query-with-multiple-where>

<https://stackoverflow.com/questions/44498524/android-adding-a-border-and-rounding-to-recyclerview-items>

<https://stackoverflow.com/questions/15683032/android-convert-base64-encoded-string-into-image-view>

<https://www.journaldev.com/9896/android-countdowntimer-example>

<https://stackoverflow.com/questions/10078867/how-to-initialize-all-the-elements-of-an-array-to-any-specific-value-in-java>

<https://stackoverflow.com/questions/27194044/how-to-properly-highlight-selected-item-on-recyclerview>

<https://stackabuse.com/sorting-algorithms-in-java/>

<https://www.journaldev.com/17899/java-simpledateformat-java-date-format>

<https://stackoverflow.com/questions/12947620/email-address-validation-in-android-on-edittext>

<https://www.javatpoint.com/how-to-send-email-in-android-using-intent>