SURESH GYAN VIHAR UNIVERSITY
Accredited by NAAC with 'A' Grade

Department of Computer Engineering & Information Technology

Lab Name- Advance Android Lab

Quiz_Game_Application

Project Report On Quiz Game Application

Submitted to:

Name: Mr. Bharat Bhushan

Designation: Assistant Professor

Submitted by:

Name: MD Faizan

SID No: 73735

Akshay Kumar Baitha
SID_No.:- 75223

Acknowledgement

In performing our group project, we had to take the help and guideline of some respected persons, who deserve our greatest gratitude. The completion of this project gives us much Pleasure. We would like to show our gratitude **Mr. Bharat Bhushan sir** for giving us a good guideline for project throughout numerous consultations and thankful every group members. We would also like to expand our deepest gratitude to all those who have directly and indirectly guided us in writing this.

Introduction:-

This is a simply and beautify android quiz application with SQLite database. The android quiz applications test your knowledge and understanding in different areas. After each quiz, the user will see the final score and you have the ability to go through the quiz result analysis to see questions that you fail and the correct answers.

Overview:

In today's world, Smart phones have changed our lives and have become an indispensable part of our lives because of its specialty to simplify our routine work and thereby saving our time. A Smartphone with an Android OS offers excellent functionality to the users offering a distinct experience. Android is a Linux based operating system and it was bought by Google in 2007.There are tons of application available and one of the prime reason for this vast number is android being an open source. On the other hand, android based device like mobile, tab are very user friendly. A survey has done by "Light Castle Partners" research wing which indicates that though other operating system mobile users exist but the majority users are goes with android operating system.In this context, Project application is developed based on android platform. The name of application is define as 'My Awesome Quiz'

Purpose:

This document provides a base to all the functionalities which should be carried out by the application, how that works the outputs available to the end user.

Motivation:

Currently most of the Examination like organizational recruitment, University class test are paper based, which costs time and resources. Questionnaire is developed, printed, and then collect data, entry, editing, cleaning, which time consuming and costly. Proposed application is the starting for avoid those circumstances which are been currently faced by any organization.

System Requirements:

1. Smartphone with Android OS version 4.4 (Kitkat) or higher
2. Minimum 512 MB of RAM
3. A processor with speeds above 1.2 GHz (any make)
4. 16 MB of storage for the app and extra for the data stored, the size of the app increases as the number of entries are increased
5. Android API version 19
6. Permission to install applications over USB and installation from unknown sources from 'Developer Options'

Applications and Technology Used:

Android Application Development is possible with a couple of software and development kits to support the software and execution, they are as follows,

1. ***Android Studio:*** Android studio is the official Integrated Development Environment (IDE) for designing, coding, debugging and executing applications for Google's Smartphone operating system,

Android. It has all keywords inbuilt for ease in back end programming and also design statements, for ease in designing as well. The version of Android Studio used to design the application is 1.4.1.to 3.1 as we are working on studios 3.1.

2. ***Java Development Kit (JDK):*** Since, Android applications require Java programming for its backend programming; it needs a JAVA environment to support its functions, executions and syntax.

3. Xml: The front end design of the application involves xml statements for the Relative layouts, Radio buttons, Radio Group, buttons, text boxes and text views.

Description:

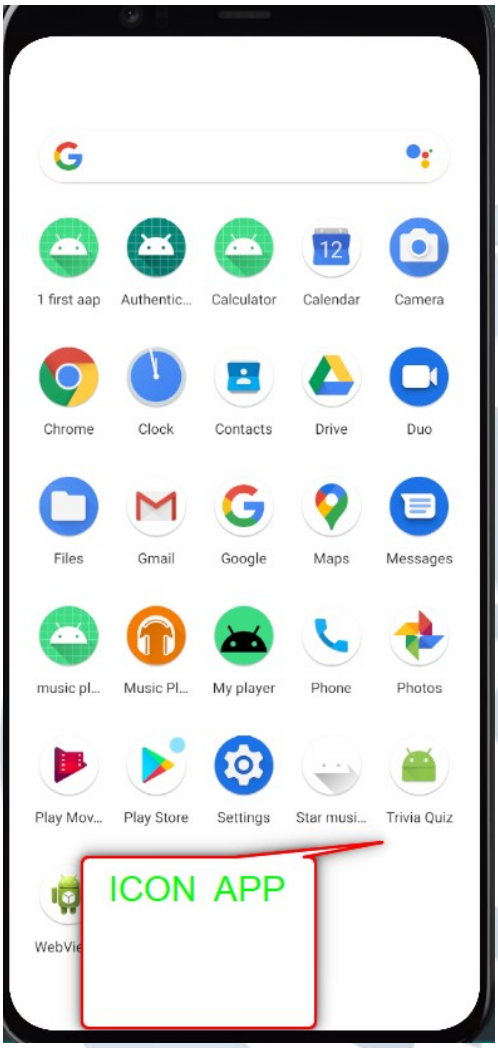
Using "Android Quiz App" source code package you can create amazing android quiz app. "Android Quiz App" source code package is built with latest android studio and is easy to use and configure. With little or no coding knowledge you can easily create quiz app with the help of this source code package.

Main Features:

- Beautiful UI
- Multilingual support
- Easy to Customize/Re-skin
- Support all Screen size
- Set Unlimited Questions

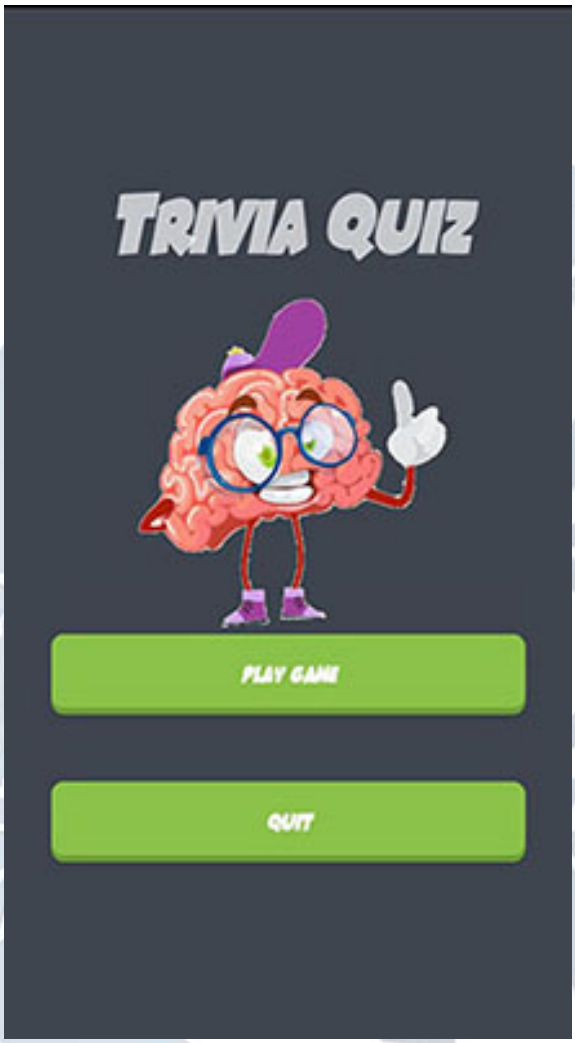
This Quiz app package is easy to customize and use. Make most from this Quiz app source code package. Best code with lowest price on this marketplace.

Icon of the Application:



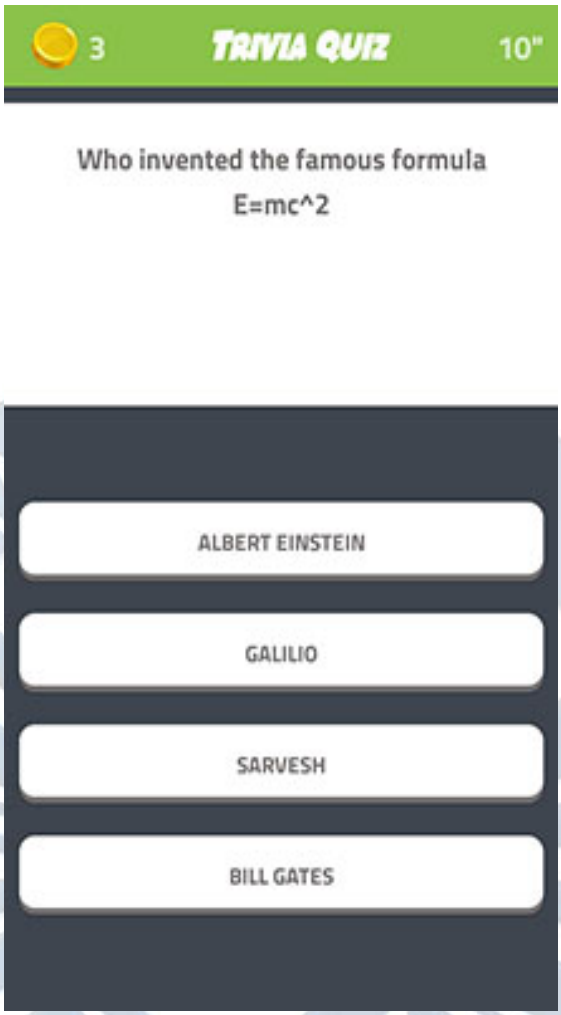
This is the icon of the project which will display on the app drawer of our android phone. After clicking this icon the app will run. This icon is created by using Adobe Photoshop.

Processes and Validation:



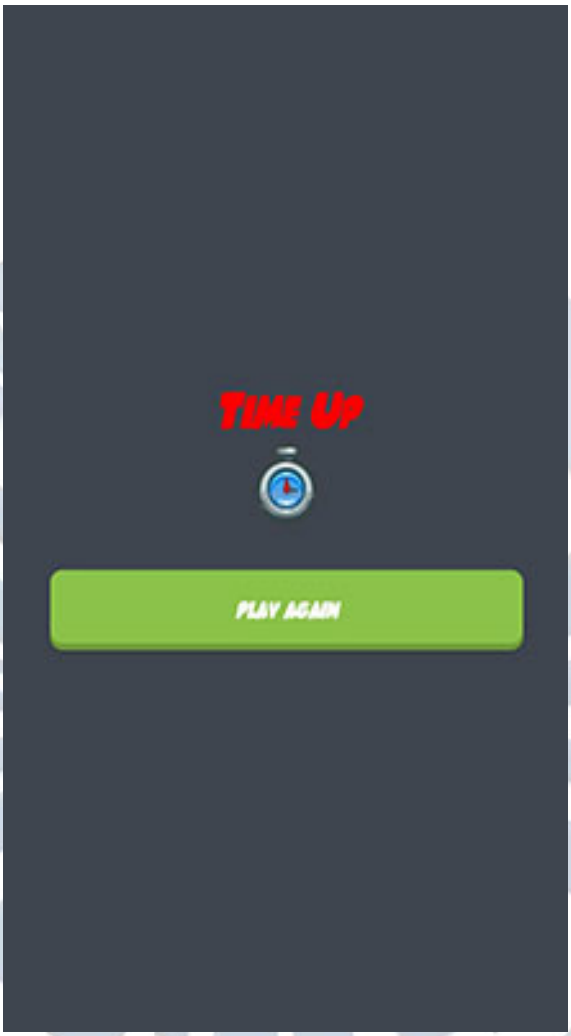
Main Screen:

This image is the starting screen where the high score is showing and where I can start the quiz. There are use three elements two text box and one button. This is the main screen on quiz app.



This image is for wrong answer. The one which is the correct answer shows up as green. The other two is on red. Red color indicates the wrong answer.
There is also a timer up at the top right. If the timer go out you will loose.
There are three Radio button and one normal Button. Radio button use for options and normal button is use for finished the task.

This image is for correct answer. This image has next button and then it will press and go for the next question and it will be continued.



This image is that the timers ran out and user didn't choose any answer so user looses. When the time will over then timer shown red color.

Quiz DFD:


2
TRIVIA QUIZ
17"
Michael Jackson is a distinguished person in the field of ?
CORRECT
NEXT
JOUNALISM
SPORTS
ACTING

Creating Layout

- **Go to xml layout default as "activity_mail.xml"**
- **One can either choose Design or Text for designing the app I choose Design.**
- **After that one should go to text and Delete the Constant Layout to Relative Layout.**

Example: <RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@color/colorBackground"
android:padding="16dp"
tools:context="com.codinginflow.myawesomequiz.StartingScreenActivity">

- It will look something like this,

- *Definition of Relative* ***Layout - RelativeLayout is a view group that displays child views in relative positions. The position of each view can be specified as relative to sibling elements (such as to the left-of or below another view) or in positions relative to the parent RelativeLayout area (such as aligned to the bottom, left or center).***

- **If one choose design then go to the design option then at the left side there is palette where one can create the button and TextView, to customize the button at the rightmost side there are the customize option, changing background, changing text color, changing the text and all.**
- **If one chooses for Text rather than design the codes are as follows:**

Code Explanation

At first we have to define all the buttons and then we have define the controls which performs start quiz, next and finish the task and four TextViews which are defines show the higher score, timing, points and Questions and Radio Buttons are use show the options.

After defining the variables we need to assign the ID's of the Buttons and the TextViews and Radio Button

ID

To do that there is a separate function written known as "setupUIViews", then taking the first variable "start quiz" and typecast with widget "Button" and the we use findViewByID so that we associate with the XML layout. In XML layout the ID of button one is called rb1.

E.g. given below:

```
rb1 = findViewById(R.id.radio_button1);
```

``````OnCreate:
``````

After defining all the Buttons and TextViews we need call the function "Oncreate". "Oncreate" is the first function called when the activity runs, so that why we need to setup all UI views "setupUIViews". That's why we first set up the UI views. After all this we need to set up OnClickListener on all of the buttons. "Android Studios" will autocomplete the maximum codes. Then we need to setText on the Info screen not on the result.

E.g.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_quiz);
```

Radio Button and Radio Group:

By now you must be very much familiar with EditText and TextView and various layouts. Let's move on and learn more about other views like ***RadioButton and RadioGroup.***

In this tutorial we are going to design a form where user will have to select one of the options using radio button. There will be some more suggestions and options that the user will have to select using Radio Button. Then we will display all the options selected by the user using a Toast on the display screen.

E.g

```
private void showNextQuestion() {
    rb1.setTextColor(textColorDefaultRb);
    rb2.setTextColor(textColorDefaultRb);
    rb3.setTextColor(textColorDefaultRb); rbGroup.clearCheck();

    if (questionCounter < questionCountTotal) {
        currentQuestion = questionList.get(questionCounter);
```

```
        textViewQuestion.setText(currentQuestion.getQuestion());
        rb1.setText(currentQuestion.getOption1());
        rb2.setText(currentQuestion.getOption2());
        rb3.setText(currentQuestion.getOption3());

private void showSolution() { rb1.setTextColor(Color.RED);
    rb2.setTextColor(Color.RED);
    rb3.setTextColor(Color.RED);

    switch (currentQuestion.getAnswerNr()) {
        case 1:
            rb1.setTextColor(Color.GREEN);
            textViewQuestion.setText("Answer 1 is correct");
            break;
        case 2:
            rb2.setTextColor(Color.GREEN);
            textViewQuestion.setText("Answer 2 is correct");
            break;
        case 3:
            rb3.setTextColor(Color.GREEN);
            textViewQuestion.setText("Answer 3 is correct");
            break;
```

Time settings:

I am making an Android quiz application and I would like for it to have a timer.. 30 seconds. When the timer finishes it moves to a new question. I have added this code for a countdown and you see when it finishes it goes to a method generateQuestion() and start the timeing goes to method startCountDown. Now it generates a new question but the answers are mixed up. It shows answer from some other questions. E.g.

```
  timeLeftInMillis = COUNTDOWN_IN_MILLIS; startCountDown();
    } else {
        finishQuiz();
    }
```

Database entry:

This android quiz application tests your knowledge and understanding in different areas. After each quiz, the user will see the final score and you have the ability to go through the quiz result analysis to see questions that you fail and the correct answers. All the quiz data is stored in SQLite database and fill the data goes to method fillQuestionsTable().

E.g.

```
@Override
  public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
     db.execSQL("DROP TABLE IF EXISTS " + QuestionsTable.TABLE_NAME); onCreate(db);
  }

  private void fillQuestionsTable() {
     Question q1 = new Question("A is correct", "A", "B", "C", 1);
     addQuestion(q1);
     Question q2 = new Question("B is correct", "A", "B", "C", 2);
     addQuestion(q2);
     Question q3 = new Question("C is correct", "A", "B", "C", 3);
     addQuestion(q3);
     Question q4 = new Question("A is correct again", "A", "B", "C", 1);
     addQuestion(q4);
     Question q5 = new Question("B is correct again", "A", "B", "C", 2);
     addQuestion(q5);
  }
```

SQLiteDatabase:

SQLite is an in-process library that implements a self-contained, zero-configuration, serverless, transactional SQL database engine. The source code for SQLite exists in the public domain and is free for both private and commercial purposes. SQLiteDatabase has methods to create, delete, execute SQL commands, and perform other common database management tasks.

E.g. *@Override*

```
  public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
     db.execSQL("DROP TABLE IF EXISTS " + QuestionsTable.TABLE_NAME); onCreate(db);
```

fillQuestionTable:

This is a method.It is used store the questions and show the questions continuously at run time.

E.g.

```
private void fillQuestionsTable() {
    Question q1 = new Question("A is correct", "A", "B", "C", 1);
    addQuestion(q1);
    Question q2 = new Question("B is correct", "A", "B", "C", 2);
    addQuestion(q2);
```

QuizContract:

QuizContract is the Method that is used in implement questions options, and answer and it store the data.

E.g. private QuizContract() {

```
}

public static class QuestionsTable implements BaseColumns {
    public static final String TABLE_NAME = "quiz_questions"; public
    static final String COLUMN_QUESTION = "question"; public static
    final String COLUMN_OPTION1 = "option1"; public static final
    String COLUMN_OPTION2 = "option2"; public static final String
    COLUMN_OPTION3 = "option3";
    public static final String COLUMN_ANSWER_NR = "answer_nr";
}
```

Conclusion and future Scope

- **Educational (General knowledge).**
- **Added comments above code so it will help you understand code much better**
- **It has database to store questions**
- **Correct answer +1 coin (Coins at top left corner)**
- **Timmer of 20 sec for each question (Timer at top right corner)**
- **Random questions at each play (It will peek random ques and will show it to the user)**
- **Good and Extensible UI.**
- **Can be played offline**
- **Compatibility with Android API-14 and above**
- **minSdkVersion 14**
- **targetSdkVersion 25**

----------X----------

THANK YOU