VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Belagavi-590014

A Mobile Application Development Mini-Project Report
On
"QUICK MATH APPLICATION"

A Mini-project report submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in **Computer Science and Engineering** *of Visvesvaraya Technological University, Belgaum.*

Submitted by:

AMOGH S BHARADWAJ (1DT19CS013)
VISHWAJIT H (1DT19CS051)
HARSH V CHALLA (1DT19CS054)

Under the guidance of:
Prof. KEERTHANA SHANKAR
Asst. Prof. Dept of CSE

Prof. CHAITRA Y R
Asst. Prof. Dept of CSE

DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT

Opp. Art of Living, Udayapura, Kanakapura Road, Bangalore-560 082
Affiliated to Visvesvaraya Technological University, Belagavi and Approved by AICTE, New Delhi) CE, CSE, ECE, EEE, ISE, ME Courses Accredited by NBA, New Delhi, NAAC A+
2021-2022

DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT

Opp. Art of Living, Udayapura, Kanakapura Road, Bangalore-560 082
Affiliated to Visvesvaraya Technological University, Belagavi and Approved by AICTE, New Delhi) CE, CSE, ECE, EEE, ISE, ME Courses Accredited by NBA, New Delhi, NAAC A+
2021-2022

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the Mini-Project on Mobile App Development(MAD) entitled “**QUICK MATHS APPLICATION**” has been successfully carried out by **AMOGH S BHARADWAJ(1DT19CS013)** , **VISHWAJIT H(1DT19CS051) and HARSH V CHALLA(1DT19CS054)**, a bonafide students of **Dayananda Sagar Academy of Technology and Management** in partial fulfilment of the requirements for the award of degree in **Bachelor of Engineering** in **Computer Science and Engineering** of the **Visvesvaraya Technological University, Belgaum** during academic year 2021-22. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements in respect of project work for the said degree.

Signature of project guide
Prof. Keerthana Shankar
Asst. Professor Dept. of CSE

Signature of the project guide
Prof. Chaitra Y R
Asst. Professor Dept. of CSE

Examiners:
1.
2.

Dr. C Nandini
Vice Principal & HOD, Dept. of CSE

ACKNOWLEDGEMENT

It gives us immense pleasure to present before you our project titled **“QUICK MATHS APPLICATION”.** The joy and satisfaction that accompany the successful completion of any task would be incomplete without the mention of those who made it possible. We are glad to express our gratitude towards our prestigious institution **DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT** for providing us with utmost knowledge, encouragement and the maximum facilities in undertaking this project.

We wish to express a sincere thanks to our respected **Principal, Dr. Ravishankar M** for all their support.

We express our deepest gratitude and special thanks to **Dr. C Nandini, Vice Principal & H.O.D, Dept. Of Computer Science Engineering**, for all her guidance and encouragement.

We sincerely acknowledge the guidance and constant encouragement of our mini-project guides, **Prof. Keerthana Shankar ,Asst. Prof. Dept of CSE** and **Prof. Chaitra Y R,Asst. Prof. Dept of CSE.**

AMOGH S BHARADWAJ(1DT19CS013)

VISHWAJIT H(1DT19CS051)

HARSH V CHALLA(1DT19CS054)

ABSTRACT

The growth of Android applications around the world is extraordinary. People turn towards technology for making their life more innovative and find solutions to their daily problems. In recent years, the emergence of smart phones has changed the definition of mobile phones. Phone is no longer just a communication tool, but also an essential part of the people's communication and daily life. Various applications added unlimited fun for people's lives.

So, our aim is to develop an android application to **'Quick Maths Application'** which deals with enhancing one's fundamental mathematical skills. Many students have sat in the tutoring centre working through problem after problem on their math homework wondering why they cannot get correct answer. It turns out that much of the people tend to forget the fundamental as they go in advance with complex operation. So, Maths Quiz Application helps in this scenario by brushing up and honing the fundamental mathematical operation such as BODMAS.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Android

Android is open-source code mobile phone operating system that comes out by Google. Quick Maths in this project is application software based on Google Android. Android is a mobile operating system based on a modified version of the Linux kernel and other open-source software, designed primarily for touch screen mobile devices such as smart phones and tablets. Android is developed by a consortium of developers known as the Open Handset Alliance, with the main contributor and commercial marketer being Google. Initially developed by Android Inc., which Google bought in 2005, Android was unveiled in 2007, with the first commercial Android device launched in September 2008. The current stable version is Android 10, released on September 3, 2019.

Android Architecture: -

Android operating system is a stack of software components which is roughly divided into five sections and four main layers as shown below in the architecture diagram. GLUT gives you the ability to create a window, handle input and render to the screen without being Operating System dependent.

Libraries

On top of Linux kernel there is a set of libraries including open -source Web browser engine WebKit, well known library libc, SQLite database which is a useful repository for storage and sharing of application data, libraries to play and record audio and video, SSL libraries responsible for Internet security etc

Android Runtime

This is the third section of the architecture and available on the second layer from the bottom. This section provides a key component called Dalvik Virtual Machine which is a kind of Java Virtual Machine specially designed and optimized for Android.

The Dalvik VM makes use of Linux core features like memory management and multithreading, which is intrinsic in the Java language. The Dalvik VM enables every Android application to run in its own process, with its own instance of the Dalvik virtual machine.

The Android runtime also provides a set of core libraries which enable Android application developers to write Android applications using standard Java programming language.

Application Framework

The Application Framework layer provides many higher-level services to applications in the form of Java classes. Application developers are allowed to make use of these services in their applications.

Applications

You will find all the Android application at the top layer. You will write your application to be installed on this layer only. Examples of such applications are Contacts Books, Browser, and Games etc.

Android UI

An Android application user interface is everything that the user can see and interact with.

Installation steps of the developing environment

- Step 1: install the Java virtual machine JDK version -7
- Step 2: install the Android SDK: first download the Android SDK
- Download address: http://developer-android-com/sdk/index-html
- Input SDK tools path in the SDK location: D: \ android \ software \ android SDK–Windows and click OK
- The Android environment is set up successfully.

1.2Android Studio

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

- A flexible Gradle-based build system
- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Apply Changes to push code and resource changes to your running app without restarting your app
- Code templates and GitHub integration to help you build common app features and import sample code
- Extensive testing tools and frameworks
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and NDK support
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine

Each project in Android Studio contains one or more modules with source code files and resource files. Types of modules include:

- Android app modules
- Library modules
- Google App Engine modules

By default, Android Studio displays your project files in the Android project view, This view is organized by modules to provide quick access to your project's key source files. All the build files are visible at the top level under Gradle Scripts and each app module contains the following folders:

- manifests: Contains the AndroidManifest.xml file.
- java: Contains the Java source code files, including JUnit test code.
- res: Contains all non-code resources, such as XML layouts, UI strings, and bitmap images.

The Android project structure on disk differs from this flattened representation. To see the actual file structure of the project, select Project from the Project dropdown.

You can also customize the view of the project files to focus on specific aspects of your app development. For example, selecting the Problems view of your project displays links to the source files containing any recognized coding and syntax errors, such as a missing XML element closing tag in a layout file

1. The toolbar lets you carry out a wide range of actions, including running your app and launching Android tools.

2. The navigation bar helps you navigate through your project and open files for editing. It provides a more compact view of the structure visible in the Project window.

3. The editor window is where you create and modify code. Depending on the current file type, the editor can change. For example, when viewing a layout file, the editor displays the Layout Editor.

4. The tool window bar runs around the outside of the IDE window and contains the buttons that allow you to expand or collapse individual tool windows.

5. The tool windows give you access to specific tasks like project management, search, version control, and more. You can expand them and collapse them.

6. The status bar displays the status of your project and the IDE itself, as well as any warnings or messages.

You can organize the main window to give yourself more screen space by hiding or moving toolbars and tool windows. You can also use keyboard shortcuts to access most IDE features.

At any time, you can search across your source code, databases, actions, elements of the user interface, and so on, by double-pressing the Shift key, or clicking the magnifying glass in the upper right-hand corner of the Android Studio window. This can be very useful if, for example, you are trying to locate a particular IDE action that you have forgotten how to trigger.

Tool windows

Instead of using preset perspectives, Android Studio follows your context and automatically brings up relevant tool windows as you work. By default, the most commonly used tool windows are pinned to the tool window bar at the edges of the application window.

- To expand or collapse a tool window, click the tool's name in the tool window bar. You can also drag, pin, unpin, attach, and detach tool windows.
- To return to the current default tool window layout, click Window > Restore Default Layout or customize your default layout by clicking Window > Store Current Layout as Default.
- To show or hide the entire tool window bar, click the window icon in the bottom left hand corner of the Android Studio window.
- To locate a specific tool window, hover over the window icon and select the tool window from the menu.

You can also use keyboard shortcuts to open tool windows. Table 1 lists the shortcuts for the most common windows.

If you want to hide all toolbars, tool windows, and editor tabs, click View > Enter Distraction Free Mode. This enables Distraction Free Mode. To exit Distraction Free Mode, click View > Exit Distraction Free Mode.

You can use Speed Search to search and filter within most tool windows in Android Studio. To use Speed Search, select the tool window and then type your search query.

CHAPTER 2

REQUIREMENTS

The requirement specification is a comprehensive description of the software and the hardware requirements required to run the project successfully

2.1 Hardware Requirements

- Display: Monitor.
- Input: Keyboard/mouse.
- Memory: 4GB
- Processor: intel/AMD processor.
- RAM: 8GB.

2.2 Software Requirements

- Operating system: WINDOWS 10
- Language used: Xml and Java.
- Software: Android Studio.

2.3 Requirement Analysis of System

The feasibility analysis: This section verified that it is feasible doing maths quickly on the Android system from the aspects of economic, technical and social feasibility

Economic feasibility: Android Quick Maths Application can be designed as long as a computer has the Android development and the application development of Android is free. In addition, this customization can be made as addition to test centre which conducts mathematical test. The information that which functions are necessary form all the consumers, which functions are needed for some people, and which features are seldom to use is easy to understand. And a lot of research is eliminated, thus saved the spending. Therefore, the whole process of development doesn't need to spend any money that is economic feasibility.

Technical feasibility: To design a Quick Maths Application which meets the basic requirements, a deep understand of JAVA language, the Android system architecture, application of framework and other technical knowledge are needed. (Framework is the core of the application, and rules that all the programmers participating in the development must abide by).

CHAPTER 3

IMPLEMENTATION

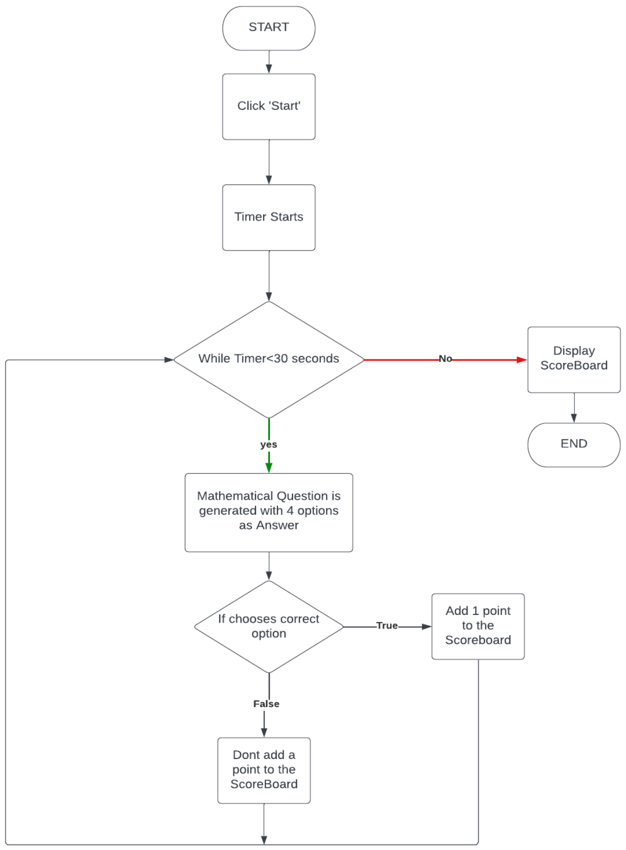
3.1 Creating the Application

The application's idea was conceived based on the idea of app-based math practice system. It was then decided to build the app based on quizzing the users with a rapid-fire quiz model. Other features that were to be decided to be added was a timer along with a score counter.

3.2 About the Application

The app is made to prompt basic mathematical problems based on BODMAS, with 4 options, As soon as user clicks on start the timer of 30 seconds starts and the user needs to select one of options and if the correct answer is choosing the user gets a point . At end of 30 seconds user can view his/her score.

3.3 Flow of the Application


START
Click 'Start'
Timer Starts
While Timer<30 seconds
No
Display ScoreBoard
END
yes
Mathematical Question is generated with 4 options as Answer
If chooses correct option
True
Add 1 point to the Scoreboard
False
Dont add a point to the ScoreBoard

3.1: flow diagram

3.4 Source code

Frontend (XML):

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
  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"
   tools:context=".MainActivity">

   <androidx.gridlayout.widget.GridLayout
      android:id="@+id/gridLayout"
      android:layout_width="362dp"
      android:layout_height="256dp"
      android:layout_marginTop="32dp"
      app:columnCount="2"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toBottomOf="@+id/problemTextView"
      app:rowCount="2">

      <Button
         android:id="@+id/button"
         android:layout_width="151dp"
         android:layout_height="118dp"
         android:background="@android:color/holo_blue_light"
         android:onClick="checkAnswer"
         android:tag="1"
```

```
    android:textSize="30sp"
    app:layout_columnWeight="1"
    app:layout_gravity="fill"
    app:layout_rowWeight="1" />

<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@android:color/holo_green_dark"
    android:onClick="checkAnswer"
    android:tag="2"
    android:textSize="30sp"
    app:layout_columnWeight="1"
    app:layout_gravity="fill"
    app:layout_rowWeight="1" />

<Button
    android:id="@+id/button3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@android:color/holo_orange_light"
    android:onClick="checkAnswer"
    android:tag="3"
    android:textSize="30sp"
    app:layout_columnWeight="1"
    app:layout_gravity="fill"
    app:layout_rowWeight="1" />

<Button
```

```
        android:id="@+id/button4"
        android:layout_width="152dp"
        android:layout_height="125dp"
        android:background="@android:color/holo_purple"
        android:onClick="checkAnswer"
        android:tag="4"
        android:textSize="30sp"
        app:layout_columnWeight="1"
        app:layout_gravity="fill"
        app:layout_rowWeight="1" />
</androidx.gridlayout.widget.GridLayout>

<TextView
    android:id="@+id/timerTextView"
    android:layout_width="109dp"
    android:layout_height="63dp"
    android:background="#4CAF50"
    android:gravity="center"
    android:text="30s"
    android:textSize="36sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.092"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.76" />

<TextView
    android:id="@+id/problemTextView"
```

```
        android:layout_width="355dp"
        android:layout_height="59dp"
        android:layout_marginTop="20dp"
        android:background="#FFC107"
        android:gravity="center"
        android:textSize="24sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/pointTextView"
        android:layout_width="109dp"
        android:layout_height="63dp"
        android:background="#E91E63"
        android:gravity="center"
        android:textSize="30sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.92"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.76" />

    <TextView
        android:id="@+id/answerTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="44dp"
        android:textSize="36sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/gridLayout" />

    <Button
        android:id="@+id/button5"
        android:layout_width="109dp"
        android:layout_height="63dp"
        android:layout_marginTop="48dp"
        android:background="#F44336"
        android:text="Start"
        android:textSize="18sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.741" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Backend(Java)

```
package com.example.quickmathapp;
import androidx.appcompat.app.AppCompatActivity;
import android.graphics.Color;
import android.os.Bundle;
import android.os.CountDownTimer;
import android.view.View;
import android.widget.Button;
import android.widget.GridLayout;
import android.widget.TextView;
import org.w3c.dom.Text;
import java.util.Random;

public class MainActivity extends AppCompatActivity {
   TextView timerTextView, problemTextView, pointTextView, answerTextView;
   Button qtn1, qtn2, qtn3, qtn4, startButton;
   androidx.gridlayout.widget.GridLayout mygridLayout;
   Double answer;
   String question;
   int answerPosition;
   int numberOfCorrectAnswers = 0;
   int numberOfQuestionsGenerated = 0;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      qtn1 = (Button) findViewById(R.id.button);
      qtn2 = (Button) findViewById(R.id.button2);
      qtn3 = (Button) findViewById(R.id.button3);
      qtn4 = (Button) findViewById(R.id.button4);
      startButton = (Button) findViewById(R.id.button5);

      timerTextView = (TextView) findViewById(R.id.timerTextView);
      problemTextView = (TextView) findViewById(R.id.problemTextView);
      pointTextView = (TextView) findViewById(R.id.pointTextView);
      answerTextView = (TextView) findViewById(R.id.answerTextView);
```

```
    mygridLayout = findViewById(R.id.gridLayout);
    mygridLayout = findViewById(R.id.gridLayout);
    for (int i = 0; i < mygridLayout.getChildCount(); i++) {
        mygridLayout.getChildAt(i).setEnabled(false);
    }
    pointTextView.setText("0/0");

    startButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            mygridLayout.setEnabled(true);
            answerTextView.setText("");
            generateRandomNumber();
            for (int i = 0; i < mygridLayout.getChildCount(); i++) {
                mygridLayout.getChildAt(i).setEnabled(true);
            }
            startTimer();
        }
    });



}

public void checkAnswer(View view) {

    if (view.getTag().toString().equals(Integer.toString(answerPosition))) {
        numberOfCorrectAnswers++;
        answerTextView.setText("Correct");
        answerTextView.setBackgroundColor(Color.GREEN);
    } else {
        answerTextView.setText("Wrong");
        answerTextView.setBackgroundColor(Color.RED);
    }
    pointTextView.setText(Integer.toString(numberOfCorrectAnswers) + "/" +
Integer.toString(numberOfQuestionsGenerated));
    generateRandomNumber();
}


private void generateRandomNumber() {
    numberOfQuestionsGenerated++;
    Random random = new Random();
    double i = random.nextInt(15);
    double j = random.nextInt(15);
    double k = random.nextInt(15);
    double z = 2;
    int position = random.nextInt(4);
    int type = random.nextInt(4);
    position++;


    answerPosition = position;


    if(type== 0) {
        answer = ((i + j) - k) / z;
        question = ("(" + "(" + Integer.toString((int)i) + " + " + Integer.toString((int)j) +
")" + "-" + Integer.toString((int)k) + ")" + "/" + Integer.toString((int)z) + "= ?");
    }
    else if(type== 1) {
        answer = i*(j+(k/z)-i);
         question = Integer.toString((int)i) +"("  + Integer.toString((int)j) + " + " +
Integer.toString((int)k) + "/" + Integer.toString((int)z) + "-" + Integer.toString((int)i)+
")" + "= ?";
```

```
    }
    else if(type== 2) {
        answer = (i + j) * (k-z);
        question = ( "(" + Integer.toString((int)i) + " + " + Integer.toString((int)j) + ")"
+ "*" + "(" + Integer.toString((int)k) + " - " + Integer.toString((int)z) + ")" + "= ?");
    }
    else if(type==3) {
        answer = ((i + j) * k) / z;
         question = ("(" + "(" + Integer.toString((int)i) + " + " + Integer.toString((int)j)
+ ")" + "*" + Integer.toString((int)k) + ")" + "/" + Integer.toString((int)z) + "= ?");
    }
    String q1 ="", q2 ="", q3 = "", q4 = "";

    if (position == 1) {
        q1 = Double.toString(answer);
        q2 = Double.toString((i*j));
        q3 = Double.toString(j+1);
        q4 = Double.toString(k+2);
    } else if (position == 2) {
        q1 = Double.toString(i+1);
        q2 = Double.toString(answer);
        q3 = Double.toString((i*j));
        q4 = Double.toString(k+2);
    } else if (position == 3) {
        q1 = Double.toString(j+1);
        q2 = Double.toString(i+2);
        q3 = Double.toString(answer);
        q4 = Double.toString((j*k));
    } else if (position == 4) {
        q1 = Double.toString(j+1);
        q2 = Double.toString(i+2);
```

```
            q3 = Double.toString((k*i));
            q4 = Double.toString(answer);
        }
        qtn1.setText(q1);
        qtn2.setText(q2);
        qtn3.setText(q3);
        qtn4.setText(q4);
        problemTextView.setText(question);
    }

    private void startTimer() {
        new CountDownTimer(30000,1000){

            @Override
            public void onTick(long millisUntilFinished) {
                timerTextView.setText(String.valueOf(millisUntilFinished/1000)+"s");
                startButton.setEnabled(false);
            }

            @Override
            public void onFinish() {
                timerTextView.setText("30s");
                answerTextView.setText("Your Score : " +
String.valueOf(numberOfCorrectAnswers));
                problemTextView.setText("0");
                pointTextView.setText("0");
                startButton.setEnabled(true);
                numberOfCorrectAnswers = 0;
                numberOfQuestionsGenerated = 0;
                for (int i = 0; i < mygridLayout.getChildCount(); i++) {
                    mygridLayout.getChildAt(i).setEnabled(false);
                }
            }
        }.start();
    }
}
```

CHAPTER 4

SNAPSHOTS

4.1 Start Screen

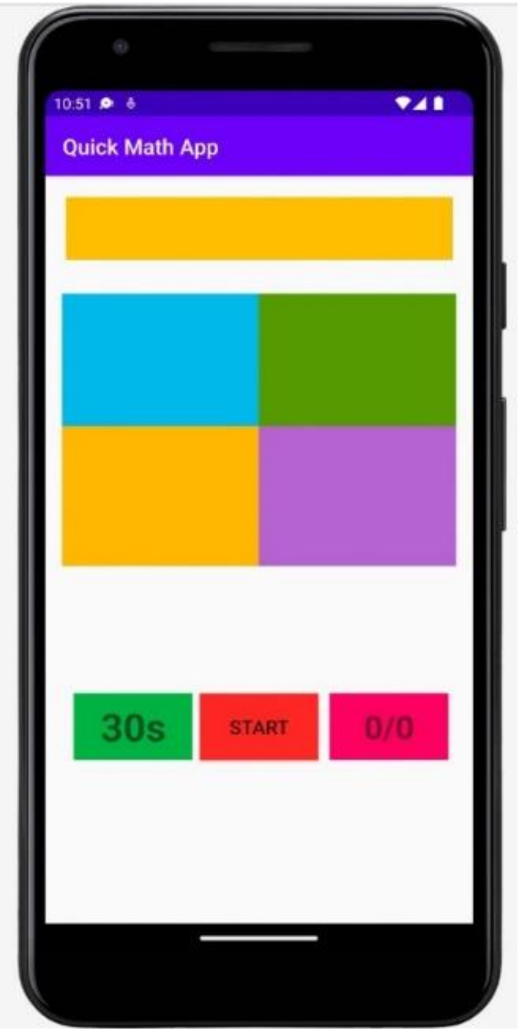
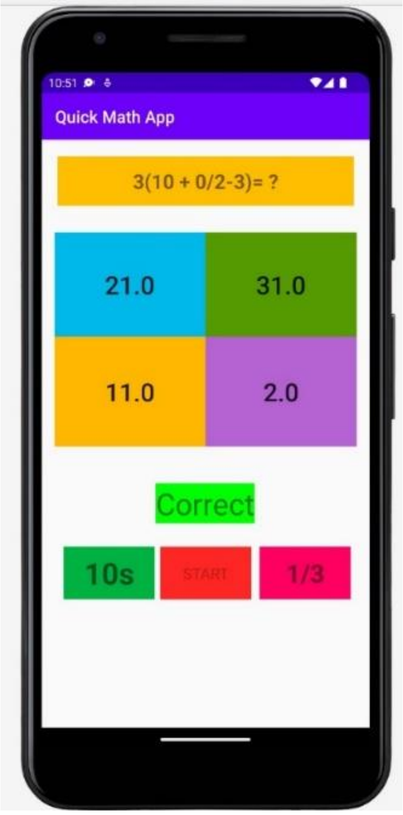


Figure 4.1 Start Screen

4.2 Quiz Start



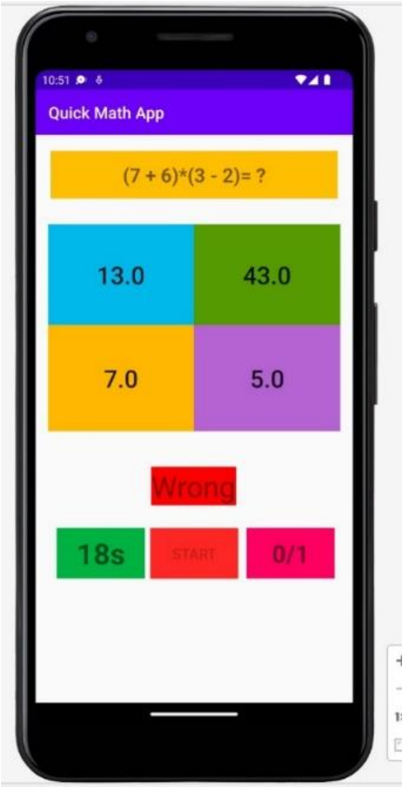


Figure 4.2 Quiz page in which scores are added to the scoreboard when the correction option is chosen.

4.3 Scoreboard

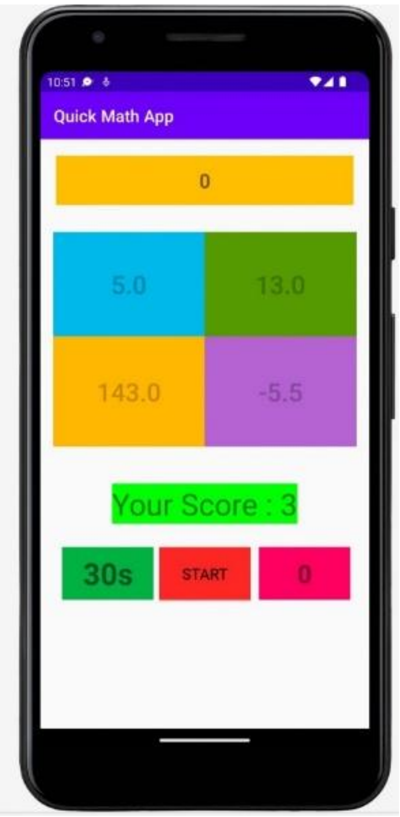


Figure 4.3 Scoreboard is displayed once the timer is reached to 30 seconds

CONCLUSION AND FUTURE WORK

The Android app is a useful way in order for one to sharpen skills in term of fast mental math, which can be proved beneficial in various occasions such as in everyday life or even during competitive examinations

Advantages

•Provides different types of question formats.

•All generated problems are random

•Mobile-based

•Any number of users can use it

•Provides functionality to assess performance in terms of scored points

Future Enhancement

The app can be enhanced by including more functionality like adding more variety of question formats, presenting hints to difficult questions, providing an cumulative report of the correct answers to all questions in the end

BIBLIOGRAPHY

WEBSITE REFERENCES:

- **https://developer.android.com/guide**
- **https://stackoverflow.com/**
- **https://www.w3schools.com/**

BOOK REFERENCES:

- “Professional Android 4 Application Development” by Reto Meier
- “Java: The Complete Reference” by Herbert Schildt

PERSONAL DETAILS

NAME: AMOGH S BHARADWAJ
USN: 1DT19CS013
SEMESTER: 5TH SEM, 'A' SECTION
COLLEGE: DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT
CONTACT INFORMATION:
EMAIL-ID: amoghsai1665@gmail.com
PHONE NO: 9108130261

NAME: VISHWAJIT H
USN: 1DT19CS051
SEMESTER: 5TH SEM, 'A' SECTION
COLLEGE: DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT
CONTACT INFORMATION:
EMAIL-ID: reddyvishwajit2255@gmail.com
PHONE NO: 95915 72556

NAME: HARSH V CHALLA
USN: 1DT19CS054
SEMESTER: 5TH SEM, 'A' SECTION
COLLEGE: DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT
CONTACT INFORMATION:
EMAIL-ID: hrzc204@gmail.com
PHONE NO: 9916997810