VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI-590018

Mini Project Report on

"Quiz Android App"

Submitted in partial fulfillment of the requirements for the VI semester

Bachelor of Engineering in Computer Science

of Visvesvaraya Technological University, Belagavi

Submitted by:

S LOGESH	*1RN19CS119*
SS KOUSHIK	*1RN19CS120*

Under the Guidance of:
Prof. Medha G
Assistant Professor
Dept. of CSE

Department of Computer Science and Engineering
(NBA Accredited for academic years 2018-19, 2019-20, 2020-22)
RNS Institute of Technology
Channasandra, Dr.Vishnuvardhan Road, Bengaluru-560 098
2020-2021

RNS Institute of Technology

Channasandra, Dr.Vishnuvardhan Road, Bengaluru-560098

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

(NBA Accredited for academic years 2018-19, 2019-20, 2020-22)

CERTIFICATE

Certified that the mini project work entitled **“QUIZ ANDROID APP”**has been successfully carried out by **”S LOGESH** bearing USN **”1RN19CS119”** and **“SS KOUSHIK** bearing USN **”1RN19CS120”**, bonafide students of **”RNS Institute of Technology”** in partial fulfillment of the requirements for the 6th semester of **”Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University”**, Belagavi, during academic year 2021-2022. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the Mobile Application Development Laboratory requirements.

Signature of the Guide	Signature of the HoD	Signature of the Principal
Prof. Medha G	**Dr. Kiran P**	**Dr. M K Venkatesha**
Assistant Professor	Professor	Principal
Dept. of CSE	Dept. of CSE	

External Viva:

Name of the Examiners	Signature with Date
1.	
2.	

Acknowledgement

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped us in carrying out this project work. We would like to take this opportunity to thank them all.

We are grateful to Management and **Dr. M K Venkatesha**, Principal, RNSIT, Bangalore, for his support towards completing this mini project.

We would like to thank **Dr. Kiran P**, Professor & Head, Department of Computer Science & Engineering, RNSIT, Bangalore, for his valuable suggestions and expert advice.

We deeply express our sincere gratitude to our guide **Prof. Medha G**, Assistant Professor, Department of CSE, RNSIT, Bangalore, for her able guidance, regular source of encouragement and assistance throughout this project.

We would like to thank all the teaching and non-teaching staff of Department of Computer Science & Engineering, RNSIT, Bengaluru for their constant support and encouragement

Abstract

The project: “Quiz Android App” is a collection of number of different types of quizzes like technical, games, sports, etc. A user can access/play all of the quiz and can attempt any of the one. There will be limited number of questions and for each correct answer user will get a credit score. User can see answers as well as can ask a query related to it. There are many quiz applications available currently on internet. But there are few Which provide better understanding between users and the application like, providing proper answers, user query solving, uploading user questions as well as answer to it, etc. To develop a user friendly quiz application which will contain : Numbers of quiz , Answers to every question, Query solving regarding any question, Uploading of user question and answer, and to improve the knowledge level of users. To develop an application which will contain solution to the above problems. By this application the user will come to know about his/her level and can learn additional knowledge. Also by this application a user can expand his/her knowledge among the world.

Contents

List of Figures

Chapter 1

Introduction

1.1 About the Project

Quiz Android App is an android application that has general questions related to current afairs and computer. It has multiple choice questions with time limit and it also calculate scores of each correct answer. It is good for students of every age group it helps in increasing general knowledge about world ,Sports and computer etc. Don't need register simply give any user name and password it will saved automatically and you can login again with same user name and password don't have to worry about the past score. The application helps the user to increase his/her knowledge. Since Smartphone mobiles are being widely used by general population and students, the Quiz Contest application can provide on the Student's mobile.

1.2 Objective

The main objective of "Quiz Android App" is to facilitate a user friendly environment for all users and reduces the manual efort. In past days quiz is conducted manually but in further resolution of the technology we are able to generate the score and pose the queries automatically. The functional requirements include to create users that are going to participate in the quiz, automatic score and report generation and administrative tasks like add, delete, update for admin privilege users. In this application, all the permissions lies with the administrator i.e., specifying the details of the quiz with checking result will show to interviewee or not, addition of question and answers, marks for each question, Set timer for each quiz and generate report with score for each quiz.

1.3 Introduction to Java

Java is a programming language created by James Gosling from Sun Microsystems (Sun) in 1991. The first publicly available version of Java (Java 1.0) was released in 1995. Sun Microsystems was acquired by the Oracle Corporation in 2010. Over time new enhanced versions of Java have been released. The current version of Java is Java 1.7 which is also known as Java 7. From the Java programming language the Java platform evolved. The Java platform allows software developers to write program code in other languages than the Java programming language and still runs on the Java virtual machine. The Java platform is usually associated with the Java virtual machine and the Java core libraries.

1.3.1 Java Virtual machine

The Java virtual machine (JVM) is a software implementation of a computer that executes programs like a real machine. The Java virtual machine is written specifically for a specific operating system, e.g. for Linux a special implementation is required as well as for Windows.

1.3.2 Java Runtime Environment vs. Java Development Kit

A Java distribution comes typically in two flavours, the Java Runtime Environment (JRE) and the Java Development Kit (JDK).The Java runtime environment (JRE) consists of the JVM and the Java class libraries and contains the necessary functionality to start Java programs. The JDK contains in addition the development tools necessary to create Java programs. The JDK consists therefore of a Java compiler, the Java virtual machine, and the Java class libraries.

1.4 Android

Android is a software platform and operating system for mobile devices. Android is available as open source. It allows developers to write managed code in the Java language, controlling the device via Google-developed Java libraries. Android SDK was released by Open Handset Alliance in the month of November of the year 2007. Android was actually developed using the kernel of Linux 2.6 and the highlighting features of Android include the following:

- No fees for licensing, distribution and release approval
- GSM, 3G EDGE networks for telephony

- IPC message passing
- Background processes and applications
- Complete multimedia hardware control
- API's for location based services such as GPS.

1.4.1 Architecture of Android OS

The skeleton of Android framework and its constituents are shown in the following igure:

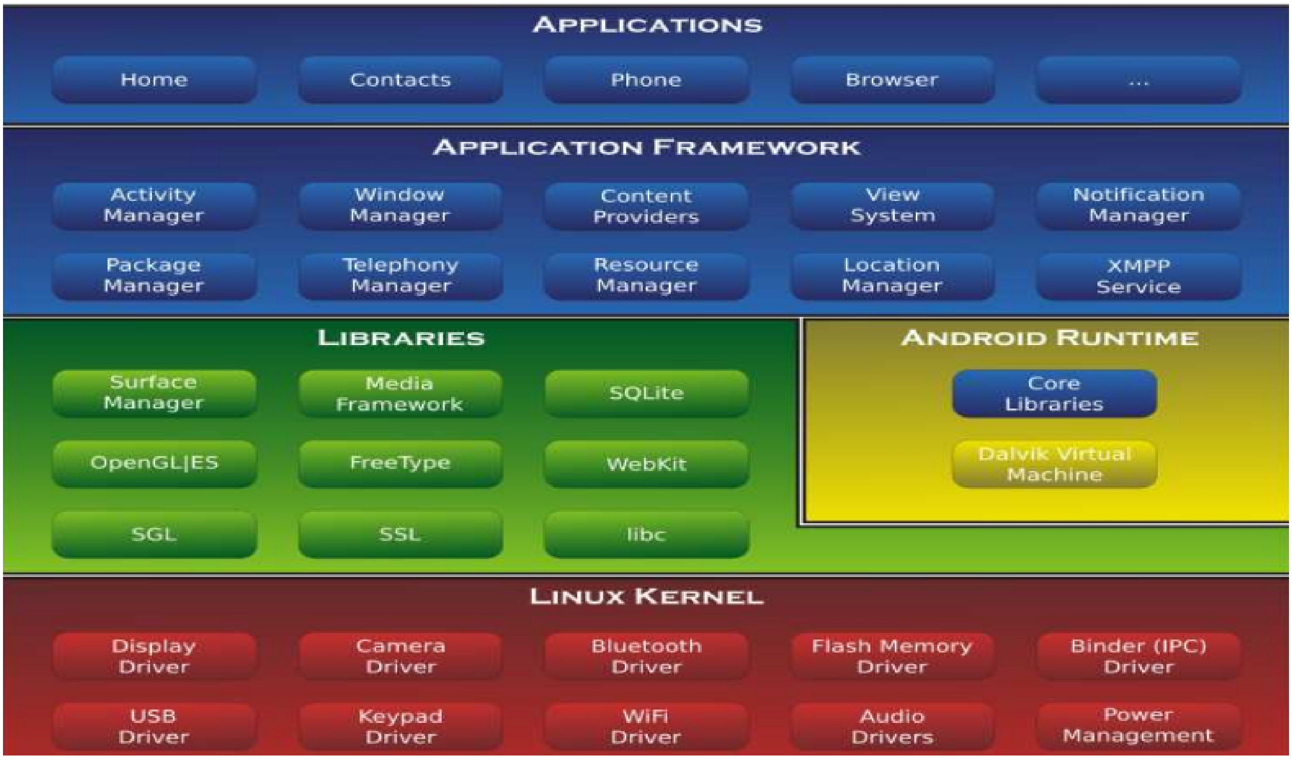


Figure 1.1: Architecture of Android OS

Applications Layer

Android ships with a set of core applications including an email client, SMS program, calendar, maps, browser, contacts and others. All applications are built using the Java. Each of the applications aims at performing a speciic task that it is actually intended to do.

Application Framework Layer

The next layer is the application framework. This includes the programs that manage the phone's basic functions like resource allocation, telephone applications, switching between processes or programs and keeping track of the phone's physical location. Application developers have full access to Android's application framework. This allows them to take advantage of Android's processing capabilities and support features when building an Android application. We can think of the application framework as a set of basic tools with which a developer can build much more complex tools.

Libraries Layer

The next layer contains the native libraries of Android. These shared libraries are all written in C or C++, compiled for the particular hardware architecture used by the phone and preinstalled by the phone vendor.

Android Runtime Layer

Android Runtime layer includes Dalvik Virtual Machine (DVM) and a set of core java libraries. Every Android app gets its own instance of DVM. Dalvik has been written so that a device can run multiple virtual machines eiciently and it executes iles with .dex (Dalvik Executable Format) extension optimized for minimum memory.

1.4.2 Components of Android

The basic components of an Android application include Activity, Broadcast Receiver, Service, and Content Provider. Each of the above, which when used for any application, has to be declared in the AndroidManifest.xml. The user interface of the component is determined by the Views. For the communication among these basic components we use Intents and Intent ilters which play crucial role during app development.

Activity

An Activity is, fundamentally, an object that has a lifecycle. An Activity is a chunk of code that does some work, as necessary. The work can include displaying a UI to the user, though it doesn't have to as some Activities never display UIs. Typically, we designate one of our application's Activities as the entry point to our application.

Broadcast Receiver

Broadcast Receiver is yet another type of component that can receive and respond to any of the broadcast announcements.

Service

A Service is a body of code that runs in the background. It can run in its own process, or in the context of another application's process, depending on its needs. Other components "bind" to a Service and invoke methods on it via remote procedure calls. An example of a Service is a media player; even when the user quits the mediaselection UI, she probably still intends for her music to keep playing. A Service keeps the music going even when the UI has completed.

Content Provider

Content Provider is a data storehouse that provides access to data on the device; the classic example is the Content Provider that is used to access the user's list of contacts. Our application can access data that other applications have exposed via a Content Provider, and we can also define our own Content Providers to expose data of our own.

1.5 Feasibility Study

The feasibility study is an evaluation and analysis of the potential of a proposed project which is based on extensive investigation and research to support the process of decision making. Depending on the results of the initial investigation the survey is now expanded to a more detailed feasibility study.

1.6 Existing System

User has to Login first by giving any user name and password it will be added automatically in database so that user can login again in future by same user name and password.

Then Welcome(user profile) will be appear user has to choose any of three option 'Start Game', 'High Score', 'Logout'. After choosing start game continue page will be appear to continue game. After Clicking on continue multiple choice questions will come automatically one by one with a specific time limit.

User has to choose any of four existing options and then user has to hit the submit button and each right answer will automatically count the score.

At the end of the game it will show the score of the user

All these findings demands new application which will reduce the manual work do everything automatically. Also the existing systems have some major drawbacks which motivate us to develop new system. Those drawbacks are as follows:

- Quest won't get update automatically need to update manually.
- Existing systems has not type of ields in like questions on History,Game,Sports,Geography etc.
- There is possibility of hanging down the existing systems.

1.6.1 Proposed System

Unlike the websites where you need to make account for every quiz you want to play, using this application based on android, you just have to login by user name and password and then you will get access to all quizzes from it. The proposed system is an application for the Android platform mobiles will help in improving the knowledge and accuracy. Android is a Linux-based operating system designed primarily for touch screen mobile devices such as smart phones and tablet computers, developed by Google in conjunction with the Open Handset Alliance. The proposed system will be developed for Android mobiles only because the market share of Android is more than other operating systems.

Chapter 2

Requirement Analysis

2.1 Hardware Requirements

The hardware requirements are very minimal and the program can be run on most of the machines

Processor : Qualcomm Snapdragon processor

Processor Speed : 1.4 GHz

RAM : 2 GB

Storage Space : 10 GB

Display Resolution : 1024*768

I/O Elements : Camera, Speaker, Microphone, GPS

Network : 5 Mbps

2.2 Software Requirement

Operating System : Android / iOS

2.3 Functional Requirements

2.3.1 Major Entities

i The System should assist people with visual impairment in analyzing their surroundings by providing voice assistance

ii The System should detect objects in the user's surroundings. It can inform the user about the obstacles in his pathway which helps him/her to navigate from one place to another saving him from accidents

2.3.2 Flutter Framework architecture

Dart is a client-optimized programming language for apps on multiple platforms. It is developed by Google and is used to build mobile, desktop, server, and web applications. Dart is an object-oriented, class-based, garbage-collected language with C-style syntax. Dart can compile to either native code or JavaScript. It supports interfaces, mixins, abstract classes, reified generics, and type inference. Whether you're creating a mobile app, web app, command-line script, or server-side app, there's a Dart solution for that. Flexible compiler technology lets you run Dart code in different ways, depending on your target platform and goals: $x = a + b$

Dart Native

For programs targeting devices (mobile, desktop, server, and more), Dart Native includes both a Dart VM with JIT (Just-In-Time) compilation and an AOT (Ahead-Of-Time) compiler for producing machine code.

Dart Web

For programs targeting the web, Dart Web includes both a development time compiler (dartdevc) and a production time compiler (dart2js).

Chapter 3

System Design

3.1 System Architecture

The project consists of the following parts as shown in figure 5.1

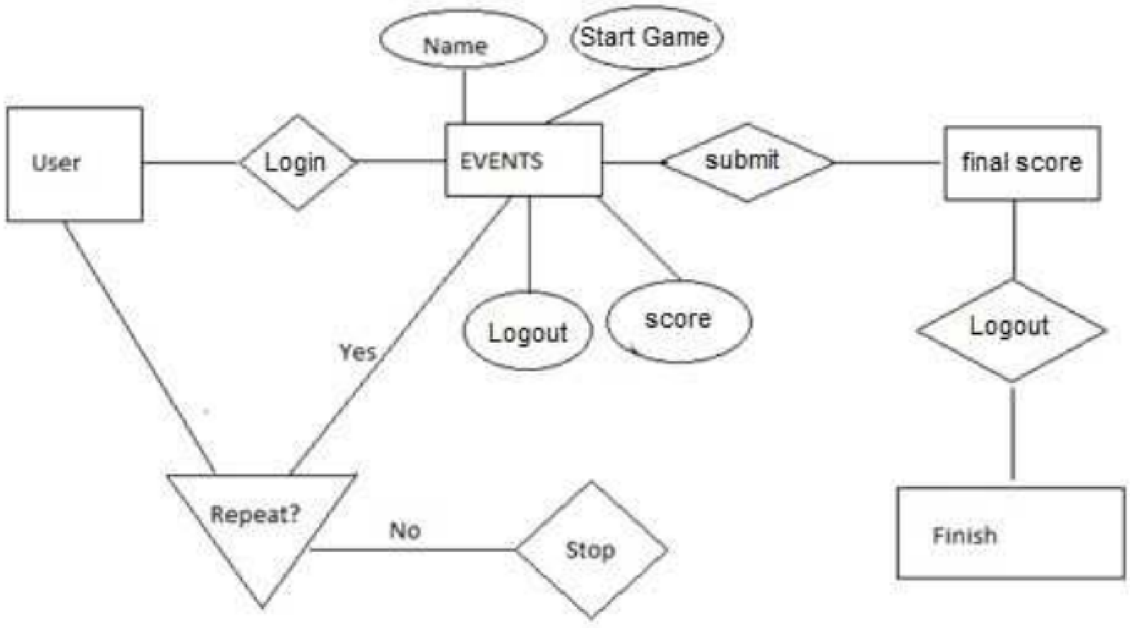


Figure 3.1: Quiz App

3.2 Components

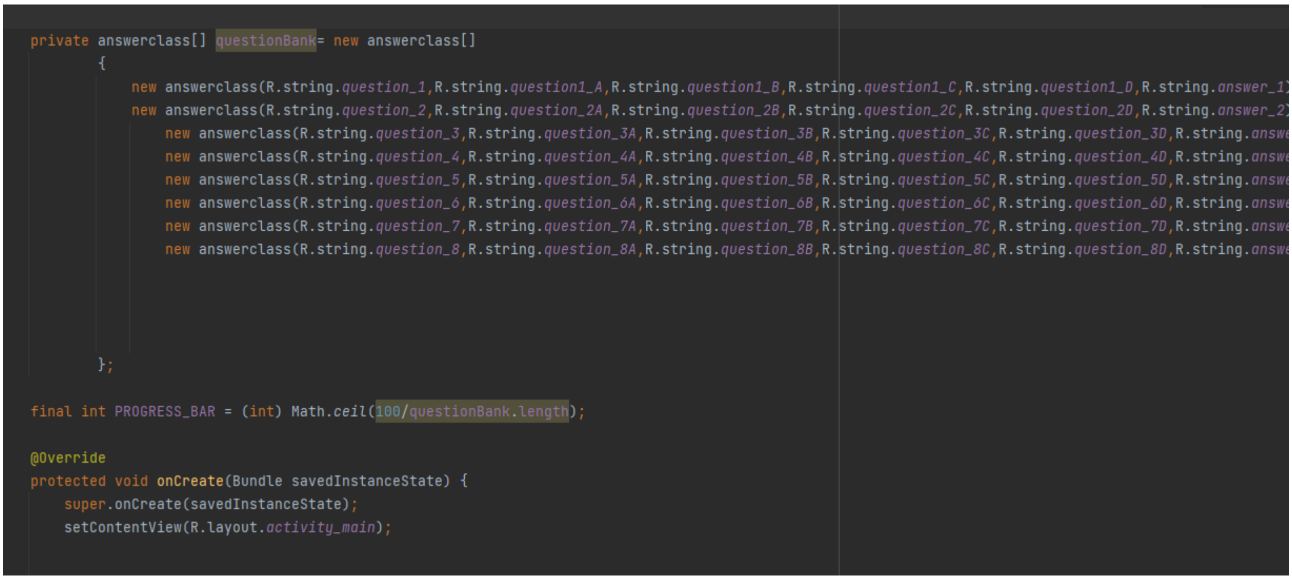
- No fees for licensing, distribution and release approval
- Linear Layout (Vertical)

- Linear Layout (Horizontal)
- TextView(Medium)
- EditText
- Button
- CheckBox
- ListView
- Spinner
- ImageView
- AlertDialog

Chapter 4

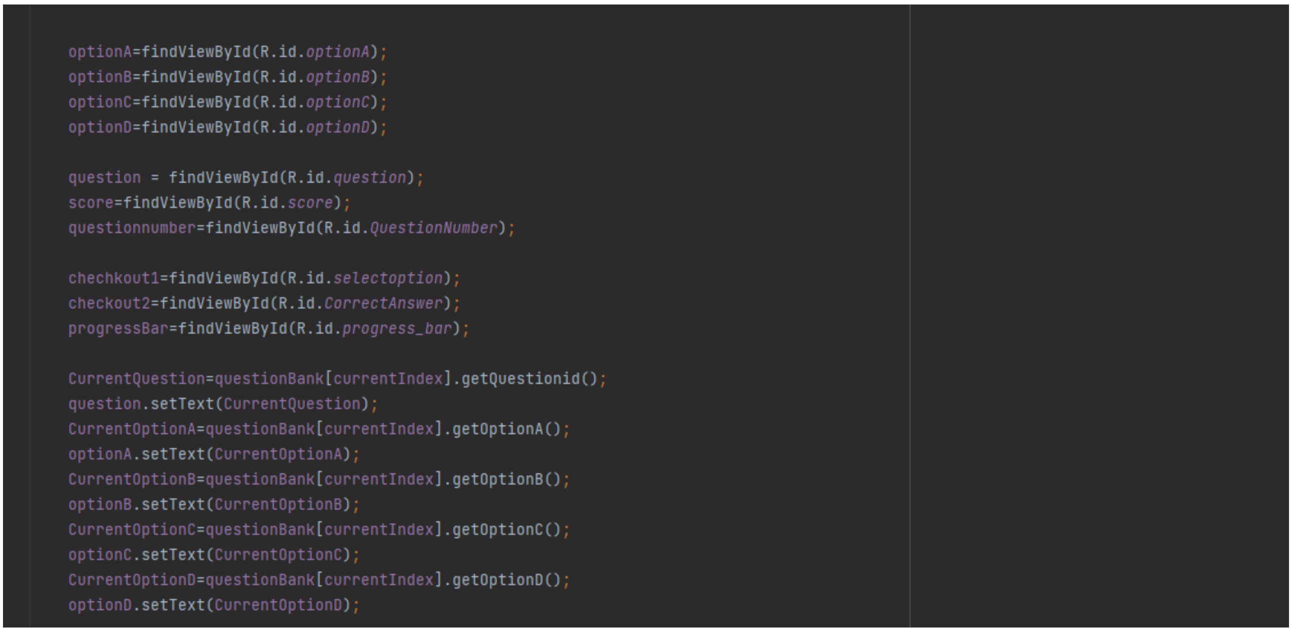
Implementation

4.1 Code



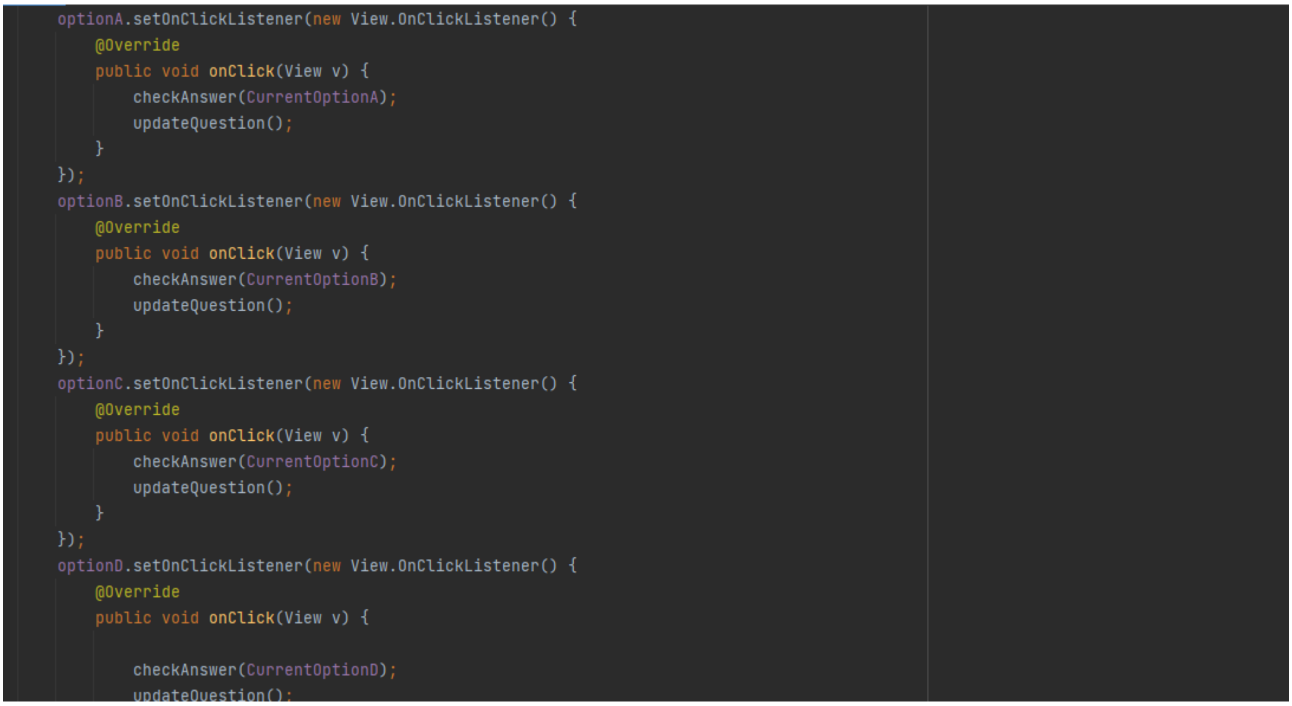
```
private answerclass[] questionBank= new answerclass[]
        {
            new answerclass(R.string.question_1,R.string.question1_A,R.string.question1_B,R.string.question1_C,R.string.question1_D,R.string.answer_1)
            new answerclass(R.string.question_2,R.string.question_2A,R.string.question_2B,R.string.question_2C,R.string.question_2D,R.string.answer_2)
                new answerclass(R.string.question_3,R.string.question_3A,R.string.question_3B,R.string.question_3C,R.string.question_3D,R.string.answe
                new answerclass(R.string.question_4,R.string.question_4A,R.string.question_4B,R.string.question_4C,R.string.question_4D,R.string.answe
                new answerclass(R.string.question_5,R.string.question_5A,R.string.question_5B,R.string.question_5C,R.string.question_5D,R.string.answe
                new answerclass(R.string.question_6,R.string.question_6A,R.string.question_6B,R.string.question_6C,R.string.question_6D,R.string.answe
                new answerclass(R.string.question_7,R.string.question_7A,R.string.question_7B,R.string.question_7C,R.string.question_7D,R.string.answe
                new answerclass(R.string.question_8,R.string.question_8A,R.string.question_8B,R.string.question_8C,R.string.question_8D,R.string.answe




        };

final int PROGRESS_BAR = (int) Math.ceil(100/questionBank.length);

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

Figure 4.1: Main activity



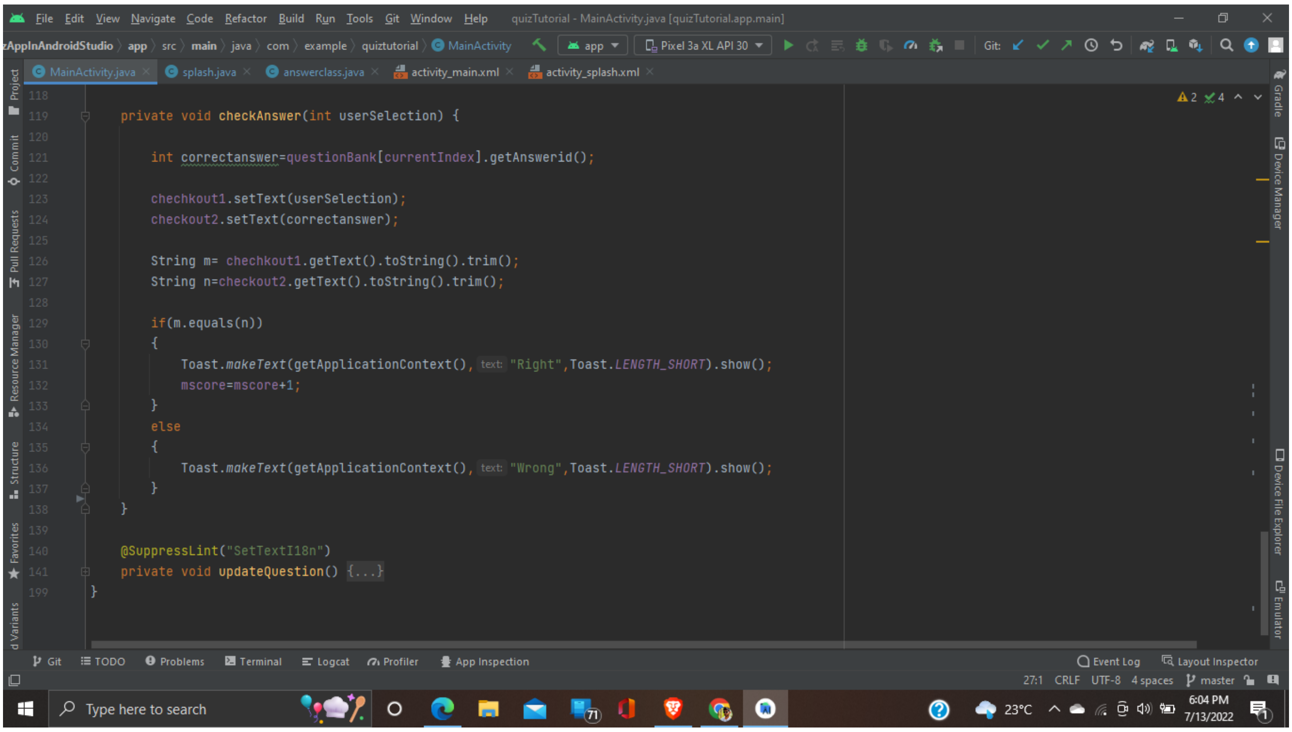
```
optionA=findViewById(R.id.optionA);
optionB=findViewById(R.id.optionB);
optionC=findViewById(R.id.optionC);
optionD=findViewById(R.id.optionD);

question = findViewById(R.id.question);
score=findViewById(R.id.score);
questionnumber=findViewById(R.id.QuestionNumber);

chechkout1=findViewById(R.id.selectoption);
checkout2=findViewById(R.id.CorrectAnswer);
progressBar=findViewById(R.id.progress_bar);

CurrentQuestion=questionBank[currentIndex].getQuestionid();
question.setText(CurrentQuestion);
CurrentOptionA=questionBank[currentIndex].getOptionA();
optionA.setText(CurrentOptionA);
CurrentOptionB=questionBank[currentIndex].getOptionB();
optionB.setText(CurrentOptionB);
CurrentOptionC=questionBank[currentIndex].getOptionC();
optionC.setText(CurrentOptionC);
CurrentOptionD=questionBank[currentIndex].getOptionD();
optionD.setText(CurrentOptionD);
```

Figure 4.2: Main activity



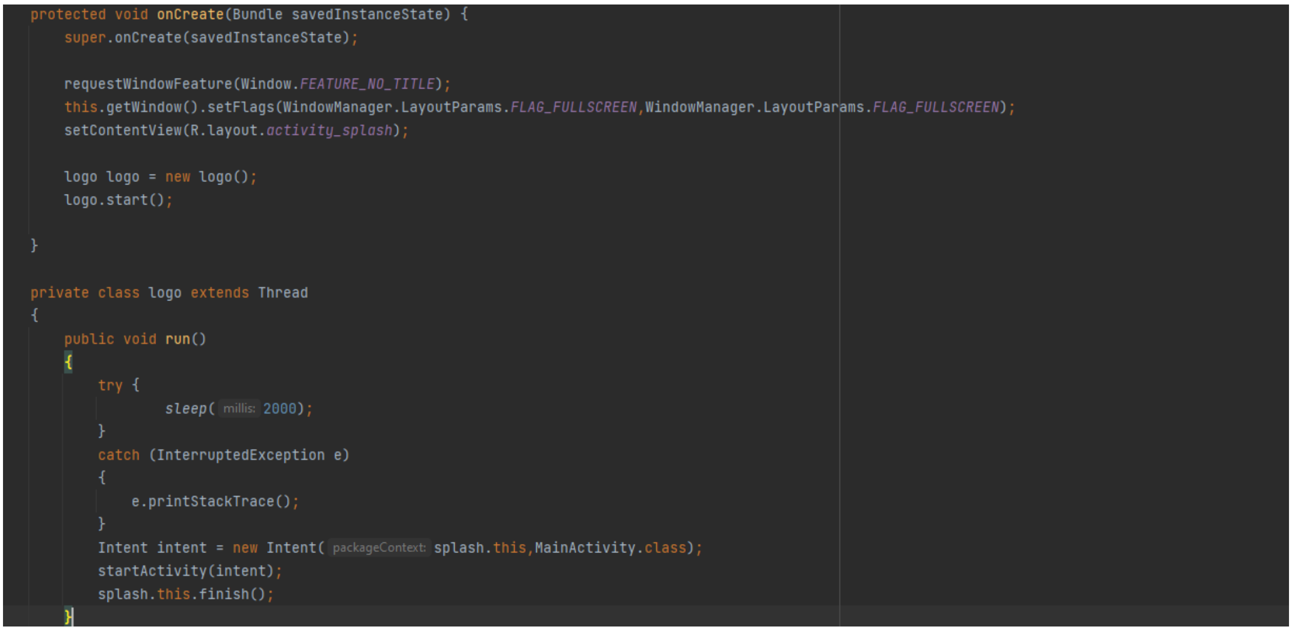
```
optionA.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        checkAnswer(CurrentOptionA);
        updateQuestion();
    }
});
optionB.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        checkAnswer(CurrentOptionB);
        updateQuestion();
    }
});
optionC.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        checkAnswer(CurrentOptionC);
        updateQuestion();
    }
});
optionD.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        checkAnswer(CurrentOptionD);
        updateQuestion();
```

Figure 4.3: Main activity


File Edit View Navigate Code Refactor Build Run Tools Git Window Help
quizTutorial - MainActivity.java [quizTutorial.app.main]
zAppInAndroidStudio 〉 app 〉 src 〉 main 〉 java 〉 com 〉 example 〉 quiztutorial 〉 MainActivity
app
Pixel 3a XL API 30
MainActivity.java
splash.java
answerclass.java
activity_main.xml
activity_splash.xml
Git
TODO
Problems
Terminal
Logcat
Profiler
App Inspection
Event Log
Layout Inspector
27:1 CRLF UTF-8 4 spaces master
Type here to search
23°C
6:04 PM
7/13/2022

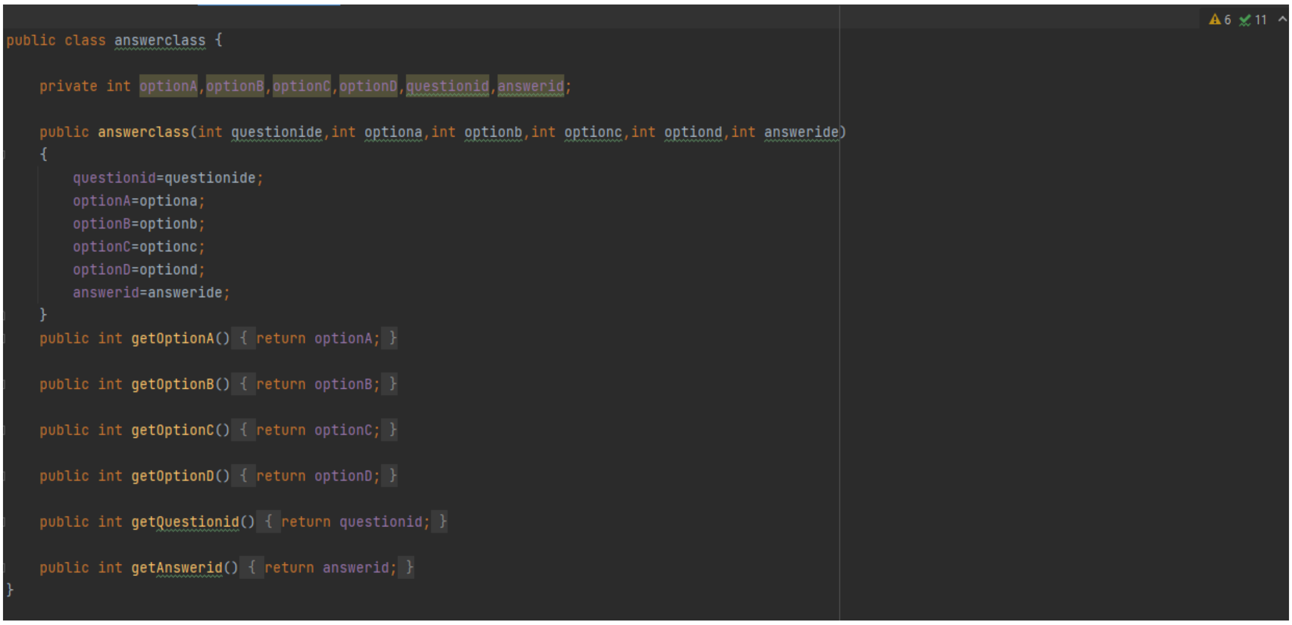
```
    private void checkAnswer(int userSelection) {

        int correctanswer=questionBank[currentIndex].getAnswerid();

        chechkout1.setText(userSelection);
        checkout2.setText(correctanswer);

        String m= chechkout1.getText().toString().trim();
        String n=checkout2.getText().toString().trim();

        if(m.equals(n))
        {
            Toast.makeText(getApplicationContext(), text: "Right",Toast.LENGTH_SHORT).show();
            mscore=mscore+1;
        }
        else
        {
            Toast.makeText(getApplicationContext(), text: "Wrong",Toast.LENGTH_SHORT).show();
        }
    }

    @SuppressLint("SetTextI18n")
    private void updateQuestion() {...}
}
```

Figure 4.4: Main activity



```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    requestWindowFeature(Window.FEATURE_NO_TITLE);
    this.getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,WindowManager.LayoutParams.FLAG_FULLSCREEN);
    setContentView(R.layout.activity_splash);

    logo logo = new logo();
    logo.start();

}

private class logo extends Thread
{
    public void run()
    {
        try {
                sleep( millis: 2000);
        }
        catch (InterruptedException e)
        {
            e.printStackTrace();
        }
        Intent intent = new Intent( packageContext: splash.this,MainActivity.class);
        startActivity(intent);
        splash.this.finish();
    }
```

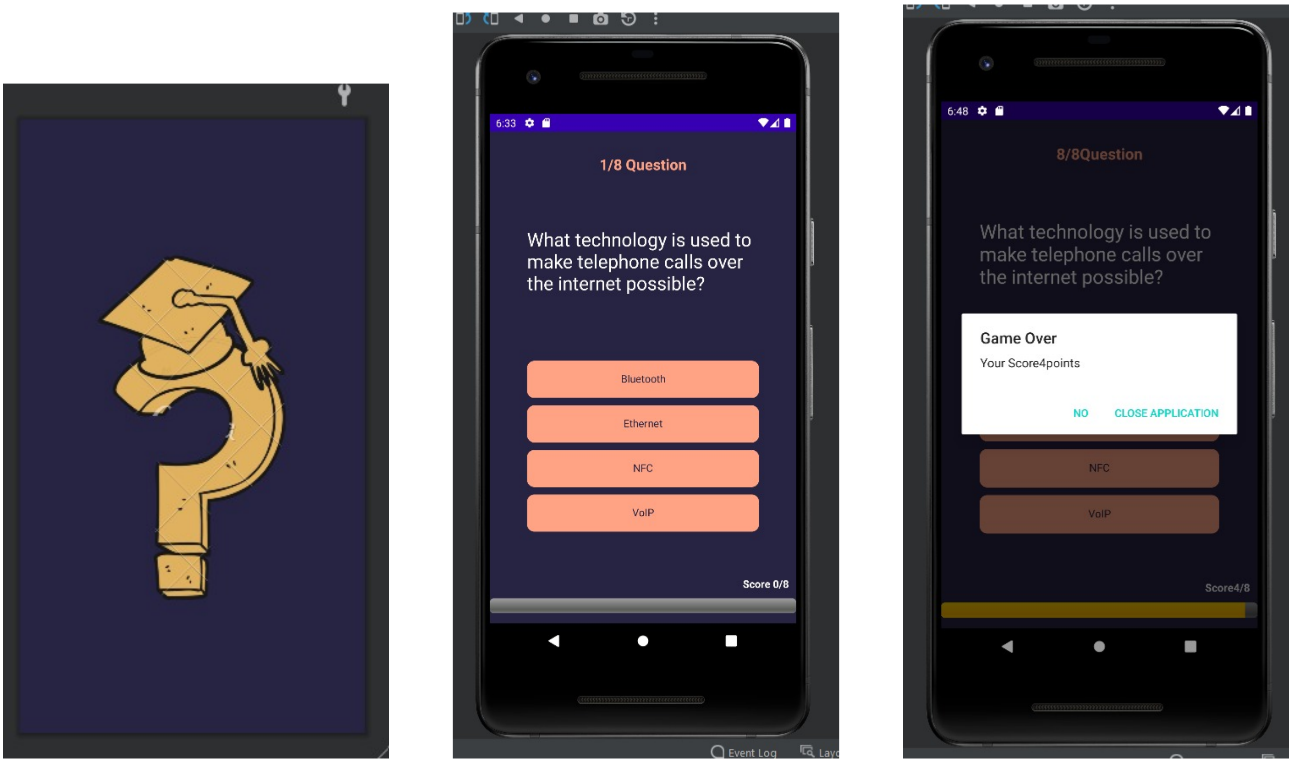
Figure 4.5: Architecture of Android OS



```
public class answerclass {

    private int optionA,optionB,optionC,optionD,questionid,answerid;

    public answerclass(int questionide,int optiona,int optionb,int optionc,int optiond,int answeride)
    {
        questionid=questionide;
        optionA=optiona;
        optionB=optionb;
        optionC=optionc;
        optionD=optiond;
        answerid=answeride;
    }
    public int getOptionA() { return optionA; }

    public int getOptionB() { return optionB; }

    public int getOptionC() { return optionC; }

    public int getOptionD() { return optionD; }

    public int getQuestionid() { return questionid; }

    public int getAnswerid() { return answerid; }
}
```

Figure 4.6: Architecture of Android OS

4.2 Result



What technology is used to make telephone calls over the internet possible?

What technology is used to make telephone calls over the internet possible?

Game Over

Your Score4points

(a) Cover Page (b) Questions (c) Score

Figure 4.7: Quiz App

Chapter 5

Testing

5.1 Objective

The objective our test plan is to ind and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal. We will also test the user friendliness of our app .The application will be used as an important tool, but we would like to ensure that it could be run on a variety of platforms with little impact on performance or usability Table **??** gives details of validation.

5.2 Process Overview

The following represents the overall low of the testing process:

Identify the requirements to be tested. All test cases shall be derived using the current Program Speciication.

Identify which particular test(s) will be used to test each module.

Review the test data and test cases to ensure that the unit has been thoroughly veriied and that the test data and test cases are adequate to verify proper operation of the unit.

Identify the expected results for each test.

Document the test case coniguration, test data, and expected results.

Perform the test(s).

Document the test data, test cases, and test coniguration used during the testing process. This information shall be submitted via the Unit/System Test Report (STR).

Successful unit testing is required before the unit is eligible for component integration/system testing.

Unsuccessful testing requires a Bug Report Form to be generated. This document shall describe the test case, the problem encountered, it's possible cause, and the sequence of events that led to the problem. It shall be used as a basis for later technical analysis.

Test documents and reports shall be submitted. Any speciications to be reviewed, revised, or updated shall be handled immediately.

5.3 Testing Process

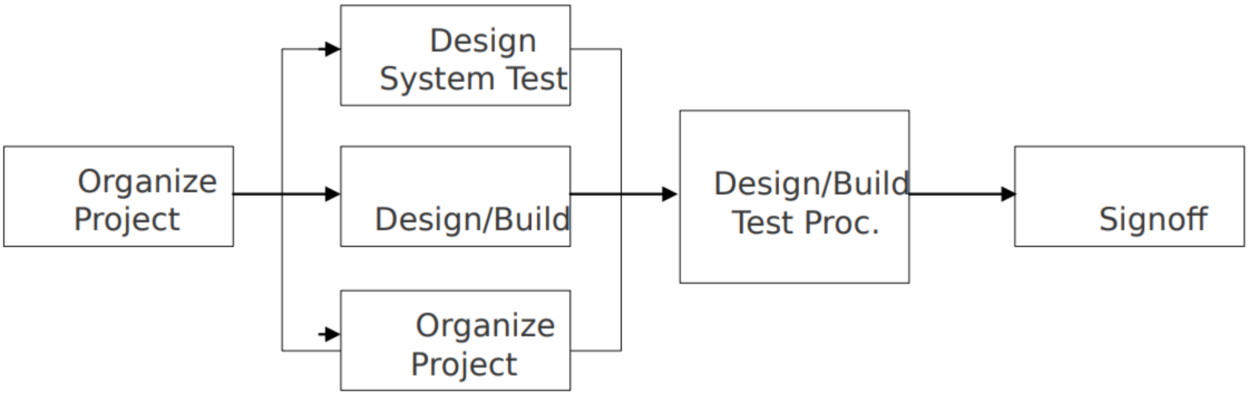


Figure 5.1: Testing Process

The diagram above outlines the Test Process approach that will be followed.

Organize Project

t involves creating a System Test Plan, Schedule Test Approach, and assigning responsibilities.

Design/Build System Test

involves identifying Test Cycles, Test Cases, Entrance Exit Criteria, Expected Results, etc. In general, test conditions/expected results will be identiied by the Test Team in conjunction with the Development Team. The Test Team will then identify Test Cases and the Data required. The Test conditions are derived from the Program Speciications Document.

Design/Build Test Procedures

includes setting up procedures such as Error Management systems and Status reporting.

Build Test Environment

includes requesting/building hardware, software and data setups.

Execute System Tests

identiied in the Design/Build Test Procedures will be executed. All results will be documented and Bug Report Forms illed out and given to the Development Team as necessary.

Signof

happens when all pre-deined exit criteria have been achieved.

Chapter 6

Conclusion And Future Enhancement

6.1 Conclusion

Currently there are many medication reminder systems which are operable manually. Due to increased manual work, the available system 51 becomes more time consuming. So in the given work, an attempt has been made to implement fully automatic medication reminder system. It eases the user's task of recalling when to take the medicine by reminding them of the particular medicine at the correct time thereby reducing the much prevalent manual work.

6.2 Further Enhancement

We have put an endless efort on this application also provided room for future expansion of this application.

Interface can be enriched later.

Selection of alarm tone are in progress.

Prescription Parameter (PP) contains the necessary information of a medicine including its name M, the dosage size g and the amount n to be taken each time (as multiples of the dosage size) and its therapy duration T during which the patient should take the medicine. A picture of the medicine is also incorporated to assist patients identifying the medicine. This parameter can be added.

Dosage Parameter (DP) states the minimum and maximum dose sizes [dmin, dmax] and the minimum and maximum separations [nsmin, nsmax] between two consecutive doses. It also speciies the supply rate (B, R), which the maximum dosage B is taken within any time interval R and the demand rate (L, P) and which the minimum dosage L is taken in any time interval p. DP parameter should be implemented in future days

References

[1] Abrar Ahmed Chhipa, et al., *Adaptive Neuro-fuzzy Inference System Based Maximum Power Tracking Controller for Variable Speed WECS*, 2021 *Energies*, Vol. 14, No. 19, pp.6275. https://doi.org/10.3390/en14196275

[2] Abrar Ahmed Chhipa, et al., *MPPT optimisation techniques and power electronics for renewable energy systems: wind and solar energy systems*, 2022 *Int. J. Swarm Intelligence (IJSI)*, Vol. 7, No. 2. https://doi.org/10.1504/IJSI.2021.10041290

[3] Abrar Ahmed Chhipa and Vinod Kumar, *DC-Microgrid Voltage Regulation using Dual Active Bridge based SVR,2021 IEEE 7th International Conference on Electrical Energy Systems (ICEES)*, 2021, pp. 490-495, doi: 10.1109/ICEES51510.2021.9383696.

[4] Abrar Ahmed Chhipa and Vinod Kumar, *Grid-Connected PV System Power Forecasting Using Nonlinear Autoregressive Exogenous Model*, *The 2nd Electric Power and Renewable Energy Conference (EPREC-2021)*, 28-30 May, 2021. (In Process)

[5] @online Raspberry pi, `https://www.raspberrypi.org/` Online; accessed 10-June-2019

[6] HU, Yun Chao, et al., *Mobile edge computing?A key technology towards 5G*, ETSI white paper, 2015, vol. 11, no 11, p. 1-16.