**Mobile Application Development**

**BSCS 6A**

**FINAL TERM PROJECT REPORT**



**Submitted By :**

Shahzaib Arshad **(Leader)**

**Submitted To :**

Mr. Shafique Ahmed

**Manage Me**

A self-Management App

**DEPARTMENT OF COMPUTER SCIENCES**

**BAHRIA UNIVERSITY, LAHORE CAMPUS**

Contents

[Marks Distribution 3](#_Toc61437016)

[Abstract 4](#_Toc61437017)

[1. Introduction 5](#_Toc61437018)

[2. Problem Statement 5](#_Toc61437019)

[3. Objectives 5](#_Toc61437020)

[4. Methodology 6](#_Toc61437021)

[5. Results 6](#_Toc61437022)

[6. Code 6](#_Toc61437023)

[A. Layouts and their Java Classes 6](#_Toc61437024)

[1. splash\_view.xml 6](#_Toc61437025)

[2. splashView.java 7](#_Toc61437026)

[3. activity\_welcome\_page.xml 7](#_Toc61437027)

[4. WelcomePage.java 8](#_Toc61437028)

[5. activity\_main.xml 9](#_Toc61437029)

[6. MainActivity.java 10](#_Toc61437030)

[7. task\_editor\_activity.xml 12](#_Toc61437031)

[8. taskEditorActivity.java 14](#_Toc61437032)

[9. activity\_math\_mission.xml 18](#_Toc61437033)

[10. math\_mission.java 19](#_Toc61437034)

[11. \_math\_mission\_setting.xml 20](#_Toc61437035)

[12. Math\_mission\_setting.java 23](#_Toc61437036)

[B. Supporting Class 25](#_Toc61437037)

[1. MathMission.java 25](#_Toc61437038)

[2. task.java 25](#_Toc61437039)

[3. AlertReciver,java 26](#_Toc61437040)

[4. NotificationHelper.java 26](#_Toc61437041)

[7. References 27](#_Toc61437042)

# Marks Distribution

**------------------------------ For Instructor Use Only ----------------------------------**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Project Grading Sheet** | | | | |
| **Sr. No.** | **Task** | **Max Marks** | **Obtained Marks** | **Comments(*if any*)** |
| 1. | Report | 05 |  |  |
| 2. | Implementation | 10 |  |  |
| 3. | Viva  (Oral/Task Based or Both) | 15 |  |  |
| **Total** | | **30** |  | **Signature:** |

**---------------------------------------------------------------------------**

# Abstract

The issue with time as well as task management has been since always. In a research conducted has concluded that people will forget 50% of information in 1 hour, 70% percent of information after 24 hour and about 90 percent of information after a week. It is noted that people getting late due to many reasons including as discussed due to STM (short term memory loose), being in subconsciousness and being lazy sometimes.

The motivation of application was to make such an application that can motivate as well as help people to complete their daily tasks. I believe this not only save people time, but it will save them from troubles.

The objective of this app was to make an app which is interactive, appealing, and easy to use for its user; despite of their age.

Procedures that were used during development are designing through paper sketching. Coding supported with testing and debugging after each module.

# Introduction

It has been noted that after some age people started being less effective with there reminding skills as well as it takes a lot of effort for them to remember each and everything. Human mind does not have that specific skill nor memory storage capabilities like any computing device. Human mind is for being preserve and utilizes for greater purposes, for innovations and for achieving what is not achieved for humanity. Manage me has purpose for being a part of people lives as they are making history. This application concise from simple task to detailed descriptions and reminders along with its easy-to-use design. Blue color is core scheme color for this application blue is about calmness, nature, and greatness.

# Problem Statement

* It will help its user for adding quick task and their details if they want to add further.
* It will be their daily task management application. Due to being small and effective.

# Objectives

* Make this app lock pleasing to use.
* Adding new task for user with quick task add option
* Providing option to add his task details in case our user wants to add.
* Providing Option to add his task for some specific day.
* Providing option to add task with alarm option on specific time.
  + - Notification providence option.

# Methodology

To achieve these objectives, I have used shared preferences and support shared preferences with classes. It was a problem to store all the data into a shared preference as it only updates a data. Shared preference is not like an array where you can store data in indexes. To store all the data and to get all data again into our shared preference, I complemented this project with help of GSON and **Type class**. Then for alarm I have used systems original SET\_ALARM user permissions. Notification for a task was a tough thing to do so.

# Results

Paste necessary screenshots of the project. ***[Assign number to each figure]***

# Code

Paste complete code of your project here.

## Layouts and their Java Classes

### splash\_view.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*=".splashView">  
  
 <ImageView  
 *android:id*="@+id/imageView"  
 *android:layout\_width*="288dp"  
 *android:layout\_height*="157dp"  
 *android:layout\_marginTop*="165dp"  
 *android:layout\_marginBottom*="268dp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:srcCompat*="@drawable/ic\_manage\_me" />  
  
 <TextView  
 *android:id*="@+id/textView"  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:fontFamily*="@font/open\_sans"  
 *android:text*="Manage Me"  
 *android:textColor*="@color/schema\_heading"  
 *android:textSize*="30sp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.5"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.60" />  
</androidx.constraintlayout.widget.ConstraintLayout>

### splashView.java

*package* com.example.manageme;  
  
*import* androidx.appcompat.app.AppCompatActivity;  
  
*import* android.content.Intent;  
*import* android.content.*SharedPreferences*;  
*import* android.os.Bundle;  
*import* android.os.Handler;  
  
*public class* splashView *extends* AppCompatActivity {  
  
 *SharedPreferences* sharedPreferences;  
 *boolean* firstTime;  
 @Override  
 *protected void* onCreate(Bundle savedInstanceState) {  
 *super*.onCreate(savedInstanceState);  
 setContentView(R.layout.*splash\_view*);  
  
 initializer();  
 *final* Handler handler = *new* Handler();  
 handler.postDelayed(*new* Runnable() {  
 @Override  
 *public void* run() {  
 *//Write whatever to want to do after delay specified (1 sec)  
 if*(firstTime){  
 Intent intent = *new* Intent(splashView.*this*, WelcomePage.*class*);  
 firstTime=*false*;  
 sharedPreferences.edit().putBoolean("first\_time",firstTime).apply();  
 startActivity(intent);  
 finish();  
 }*else* {  
 Intent intent = *new* Intent(splashView.*this*, MainActivity.*class*);  
 startActivity(intent);  
 finish();  
 }  
 }  
 }, 2000);  
 }  
 *private void* initializer() {  
 sharedPreferences=getSharedPreferences("com.example.manage",*MODE\_PRIVATE*);  
 firstTime=sharedPreferences.getBoolean("first\_time",*true*);  
  
 }  
}

### activity\_welcome\_page.xml

<?*xml version*="1.0" *encoding*="utf-8"?>  
<LinearLayout *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:id*="@+id/mainActivity"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="match\_parent"  
 *android:orientation*="vertical"  
 *android:paddingLeft*="16dp"  
 *android:paddingTop*="32dp"  
 *android:paddingRight*="16dp"  
 *android:paddingBottom*="16dp"  
 *tools:context*=".MainActivity">  
  
 <LinearLayout  
  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="wrap\_content"  
 *android:background*="@drawable/custom\_edit\_text\_light"  
 *android:orientation*="horizontal"  
 *android:paddingStart*="20dp"  
 *android:paddingTop*="5dp"  
 *android:paddingEnd*="20dp"  
 *android:paddingBottom*="10dp">  
  
 <EditText  
 *android:id*="@+id/addDirectTaskEditText"  
 *android:layout\_width*="0dp"  
 *android:layout\_height*="wrap\_content"  
 *android:textColorHint*="@color/secondary\_text\_light"  
 *android:textColor*="@color/schema\_heading"  
  
 *android:layout\_weight*="0.8"  
 *android:backgroundTint*="@color/schema\_heading"  
  
 *android:fontFamily*="@font/open\_sans"  
 *android:hint*="@string/add\_title\_of\_task"  
 *android:textSize*="24sp"  
 *android:textStyle*="bold" />  
  
 <Button  
 *android:id*="@+id/addButton"  
 *android:layout\_width*="42dp"  
 *android:layout\_height*="42dp"  
 *android:background*="@drawable/ic\_baseline\_add\_24" />  
  
 </LinearLayout>  
  
 <ListView  
 *android:id*="@+id/taskListView"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="match\_parent"  
 *android:layout\_marginTop*="16dp"  
 *android:layout\_marginBottom*="16dp"  
 *android:background*="@drawable/custom\_list"  
 *android:scrollbarSize*="4dp"  
 *android:touchscreenBlocksFocus*="?android:attr/windowSwipeToDismiss"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.0"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.0" />  
  
  
</LinearLayout>

### WelcomePage.java

*package* com.example.manageme;  
  
*import* androidx.appcompat.app.AppCompatActivity;  
  
*import* android.content.Intent;  
*import* android.media.Image;  
*import* android.os.Bundle;  
*import* android.view.View;  
*import* android.widget.Button;  
*import* android.widget.ImageView;  
  
*public class* WelcomePage *extends* AppCompatActivity {  
  
Button getStartedButtonWelcomePage;  
 @Override  
 *protected void* onCreate(Bundle savedInstanceState) {  
 *super*.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_welcome\_page*);  
 getStartedButtonWelcomePage=findViewById(R.id.*getStartedButtonWelcomePage*);  
  
 }  
  
 *public void* getStarted(View view) {  
 Intent intent = *new* Intent(WelcomePage.*this*, MainActivity.*class*);  
 startActivity(intent);  
 finish();  
 }  
}

### activity\_main.xml

<?*xml version*="1.0" *encoding*="utf-8"?>  
<LinearLayout *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:orientation*="vertical"  
 *android:paddingLeft*="16dp"  
 *android:paddingTop*="32dp"  
 *android:paddingRight*="16dp"  
 *android:paddingBottom*="16dp"  
 *tools:context*=".MainActivity">  
  
 <LinearLayout  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="wrap\_content"  
 *android:background*="@drawable/custom\_edit\_text\_light"  
 *android:orientation*="horizontal"  
 *android:paddingStart*="20dp"  
 *android:paddingTop*="5dp"  
 *android:paddingEnd*="20dp"  
 *android:paddingBottom*="10dp">  
  
 <EditText  
 *android:id*="@+id/addDirectTaskEditText"  
 *android:layout\_width*="0dp"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_weight*="0.8"  
 *android:backgroundTint*="@color/schema\_heading"  
 *android:fontFamily*="@font/open\_sans"  
 *android:hint*="@string/add\_title\_of\_task"  
 *android:textSize*="24sp"  
 *android:textStyle*="bold" />  
  
 <Button  
 *android:id*="@+id/addButton"  
 *android:layout\_width*="42dp"  
 *android:layout\_height*="42dp"  
 *android:background*="@drawable/ic\_baseline\_add\_24" />  
  
 </LinearLayout>  
  
 <ListView  
 *android:id*="@+id/taskListView"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="match\_parent"  
 *android:layout\_marginTop*="16dp"  
 *android:layout\_marginBottom*="16dp"  
 *android:scrollbarSize*="4dp"  
 *android:touchscreenBlocksFocus*="?android:attr/windowSwipeToDismiss"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.0"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.0" />  
  
  
</LinearLayout>

### MainActivity.java

*package* com.example.manageme;  
  
*import* androidx.appcompat.app.AlertDialog;  
*import* androidx.appcompat.app.AppCompatActivity;  
  
*import* android.content.Context;  
*import* android.content.*DialogInterface*;  
*import* android.content.Intent;  
*import* android.content.*SharedPreferences*;  
*import* android.os.Bundle;  
*import* android.view.View;  
*import* android.widget.AdapterView;  
*import* android.widget.ArrayAdapter;  
*import* android.widget.Button;  
*import* android.widget.EditText;  
*import* android.widget.ListView;  
*import* com.google.gson.Gson;  
*import* com.google.gson.reflect.TypeToken;  
*import* java.lang.reflect.*Type*;  
*import* java.util.ArrayList;  
  
*public class* MainActivity<firstTime> *extends* AppCompatActivity {  
  
 ListView taskListView;  
 EditText addDirectTaskEditText;  
 *//hello* Button addButton;  
 *static* ArrayList<String> *taskItemArrayList*;  
 *static* ArrayAdapter<String> *taskItemArrayAdapter*;  
 *SharedPreferences* sharedPreferences;  
 ArrayList<task> arrayList;  
 Gson gson;  
  
 @Override  
 *protected void* onCreate(Bundle savedInstanceState) {  
 *super*.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
  
  
 initializer(); *// initialize* String description = sharedPreferences.getString("identity\_task", *null*);  
 *Type* type = *new* TypeToken<ArrayList<task>>() {}.getType(); *// storing type with type token* arrayList = gson.fromJson(description, type); *// json format to java object  
  
 if*(arrayList==*null*){  
 arrayList=*new* ArrayList<>();  
 }  
 *else* {  
 *for*(*int* i=0; i<arrayList.size();i++){  
 *taskItemArrayList*.add(arrayList.get(i).getTaskTitle());  
 }  
*// taskItemArrayAdapter = new ArrayAdapter<>(this, android.R.layout.simple\_list\_item\_1, taskItemArrayList);  
// taskListView.setAdapter(taskItemArrayAdapter);* }  
  
*// HashSet<String> set= (HashSet<String>) sharedPreferences.getStringSet("task",null);  
  
// Log.i("Set: ", String.valueOf(set));  
// taskItemArrayList=new ArrayList<>(set);  
// taskItemArrayAdapter = new ArrayAdapter<>(this, android.R.layout.simple\_list\_item\_1, taskItemArrayList);  
// taskListView.setAdapter(taskItemArrayAdapter);* addButton.setOnClickListener(*new* View.OnClickListener() {  
 @Override  
 *public void* onClick(View v){  
  
  
 *// check if edit text is not empty  
 taskItemArrayList*.add( addDirectTaskEditText.getText().toString());  
 *taskItemArrayAdapter*.notifyDataSetChanged();  
  
 task myNewTask=*new* task(addDirectTaskEditText.getText().toString(),"");  
 arrayList.add(myNewTask);  
  
 String updated=gson.toJson(arrayList);  
 sharedPreferences.edit().putString("identity\_task",updated).apply();  
  
*// HashSet<String> set = new HashSet<>(MainActivity.taskItemArrayList);  
// sharedPreferences.edit().putStringSet("task", set).apply();  
// Log.i("Shared Preference", String.valueOf(sharedPreferences.getStringSet("task", null)));* }  
 });  
  
 taskListView.setOnItemClickListener(*new* AdapterView.OnItemClickListener() {  
 @Override  
 *public void* onItemClick(AdapterView<?> parent, View view, *int* position, *long* id) {  
 Intent intent = *new* Intent(MainActivity.*this*, taskEditorActivity.*class*);  
 intent.putExtra("dataPosition", position);  
 startActivity(intent);  
 }  
 });  
  
 taskListView.setOnItemLongClickListener(*new* AdapterView.OnItemLongClickListener() {  
 @Override  
 *public boolean* onItemLongClick(AdapterView<?> parent, View view, *int* position, *long* id) {  
  
 *int* itemToDelete = position;  
 *new* AlertDialog.Builder(MainActivity.*this*)  
 .setIcon(android.R.drawable.*ic\_delete*)  
 .setTitle("Are you sure?")  
 .setMessage("You really want to delete this note?")  
 .setPositiveButton("Yes", *new DialogInterface*.OnClickListener() {  
 @Override  
 *public void* onClick(*DialogInterface* dialog, *int* which) {  
 arrayList.remove(itemToDelete);  
 String updated=gson.toJson(arrayList);  
 sharedPreferences.edit().putString("identity\_task",updated).apply();  
 }  
 })  
 .setNegativeButton("No", *null*).show();  
  
  
 *return true*;  
 }  
 });  
 }  
  
  
 *private void* initializer() {  
 taskListView = findViewById(R.id.*taskListView*);  
 addButton = findViewById(R.id.*addButton*);  
 *taskItemArrayList* = *new* ArrayList<>();  
  
 taskListView = (ListView) findViewById(R.id.*taskListView*);  
 addDirectTaskEditText = findViewById(R.id.*addDirectTaskEditText*);  
 sharedPreferences = getApplicationContext().getSharedPreferences("com.example.manage", Context.*MODE\_PRIVATE*);  
  
 *taskItemArrayAdapter* = *new* ArrayAdapter<>(*this*, android.R.layout.*simple\_list\_item\_1*, *taskItemArrayList*);  
 taskListView.setAdapter(*taskItemArrayAdapter*);  
  
 gson = *new* Gson();  
 }  
}

### task\_editor\_activity.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*=".taskEditorActivity">  
  
 <LinearLayout  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="match\_parent"  
 *android:orientation*="vertical"  
 *android:paddingLeft*="16dp"  
 *android:paddingTop*="32dp"  
 *android:paddingRight*="16dp"  
 *android:paddingBottom*="16dp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent">  
  
 <LinearLayout  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="wrap\_content"  
 *android:background*="@drawable/custom\_edit\_text\_light"  
 *android:orientation*="horizontal"  
 *android:paddingStart*="20dp"  
 *android:paddingTop*="5dp"  
 *android:paddingEnd*="20dp"  
 *android:paddingBottom*="10dp">  
  
 <EditText  
 *android:id*="@+id/titleTextTaskEditor"  
 *style*="@style/Theme.ManageMe"  
 *android:layout\_width*="0dp"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_weight*="0.8"  
 *android:autofillHints*="@string/auto\_fill\_string"  
 *android:backgroundTint*="@color/schema\_heading"  
 *android:ems*="10"  
 *android:fontFamily*="@font/open\_sans"  
 *android:hint*="@string/add\_title\_of\_task"  
 *android:inputType*="textPersonName"  
 *android:textColor*="@color/schema\_heading"  
 *android:textSize*="24sp"  
 *android:textStyle*="bold" />  
  
 <Button  
 *android:id*="@+id/calenderButtonTaskEditor"  
 *android:layout\_width*="34dp"  
 *android:layout\_height*="34dp"  
 *android:background*="@drawable/ic\_baseline\_calendar\_today\_24" />  
  
 </LinearLayout>  
  
 <EditText  
 *android:id*="@+id/descriptionTaskEditor"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="wrap\_content"  
 *android:autofillHints*="@string/auto\_fill\_string"  
 *android:backgroundTint*="@color/schema\_heading"  
 *android:ems*="10"  
 *android:fontFamily*="@font/open\_sans\_light"  
 *android:hint*="@string/add\_description"  
 *android:inputType*="textPersonName"  
 *android:paddingTop*="32dp"  
 *android:textColor*="@color/primary\_text\_light"  
 *android:textSize*="20sp" />  
  
 <LinearLayout  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="wrap\_content"  
 *android:orientation*="horizontal">  
  
 </LinearLayout>  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
  
 *android:id*="@+id/relativeLayout"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="wrap\_content"  
 *android:orientation*="horizontal"  
 *android:paddingTop*="32dp">  
  
 <Button  
 *android:id*="@+id/timerButtonTaskEditor"  
 *android:layout\_width*="40dp"  
 *android:layout\_height*="40dp"  
 *android:layout\_marginStart*="16dp"  
 *android:layout\_marginBottom*="16dp"  
 *android:background*="@drawable/ic\_baseline\_alarm\_24"  
 *android:fontFamily*="@font/open\_sans"  
  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.5" />  
  
 <Button  
 *android:id*="@+id/priorityButtonTaskEditor"  
 *android:layout\_width*="40dp"  
 *android:layout\_height*="40dp"  
 *android:layout\_marginStart*="64dp"  
 *android:layout\_marginBottom*="16dp"  
 *android:background*="@drawable/ic\_baseline\_priority\_low\_24"  
 *android:enabled*="false"  
 *android:fontFamily*="@font/open\_sans"  
 *android:onClick*="setPriority"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.5" />  
  
 <Button  
 *android:id*="@+id/mathMissionButtontaskEditor"  
 *android:layout\_width*="40dp"  
 *android:layout\_height*="40dp"  
 *android:layout\_marginStart*="112dp"  
 *android:layout\_marginBottom*="16dp"  
 *android:background*="@drawable/ic\_baseline\_math\_mission\_24"  
 *android:enabled*="false"  
 *android:onClick*="setMathMission"  
 *android:padding*="0dp"  
  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.5" />  
  
 <Button  
 *android:id*="@+id/doneBtnNotesEditor"  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_marginEnd*="8dp"  
 *android:layout\_marginBottom*="16dp"  
 *android:background*="@drawable/custom\_button"  
 *android:insetLeft*="0dp"  
 *android:onClick*="doneUpdate"  
 *android:text*="@string/done"  
 *android:textColor*="@color/white"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="1"  
 *app:layout\_constraintStart\_toEndOf*="@+id/timerButtonTaskEditor"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.5" />  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
 </LinearLayout>  
</androidx.constraintlayout.widget.ConstraintLayout>

### taskEditorActivity.java

*package* com.example.manageme;  
  
*import* androidx.appcompat.app.AppCompatActivity;  
  
*import* android.app.AlarmManager;  
*import* android.app.DatePickerDialog;  
*import* android.app.PendingIntent;  
*import* android.app.TimePickerDialog;  
*import* android.content.Context;  
*import* android.content.Intent;  
*import* android.content.*SharedPreferences*;  
*import* android.os.Bundle;  
*import* android.provider.AlarmClock;  
*import* android.text.*Editable*;  
*import* android.text.*TextWatcher*;  
*import* android.util.Log;  
*import* android.view.View;  
*import* android.widget.Button;  
*import* android.widget.DatePicker;  
*import* android.widget.TextView;  
*import* android.widget.TimePicker;  
*import* android.widget.Toast;  
  
*import* com.google.gson.Gson;  
*import* com.google.gson.reflect.TypeToken;  
  
*import* java.lang.reflect.*Type*;  
*import* java.util.ArrayList;  
*import* java.util.Calendar;  
*import* java.util.HashSet;  
  
*public class* taskEditorActivity *extends* AppCompatActivity {  
 *int* dataPosition;  
  
 TextView titleTextTaskEditor;  
 TextView descriptionTextTextEditor;  
 *//Buttons* Button calenderButton;  
 Button timerButtonTaskEditor;  
  
 Intent intent;  
 *public static* ArrayList<task> *arrayList*;  
 Gson gson;  
  
 *//date set* Calendar calendar;  
 DatePickerDialog datePickerDialog;  
 TimePickerDialog timePickerDialog;  
  
 *int* Year;  
 *int* Month;  
 *int* DayOfMonth;  
 *int* HoursOfDay;  
 *int* Min;  
 *boolean* Is24HourView;  
  
  
 *SharedPreferences* sharedPreferences;  
  
 @Override  
 *protected void* onCreate(Bundle savedInstanceState) {  
 *// priority + title task + details + time(layout shift)  
 super*.onCreate(savedInstanceState);  
 setContentView(R.layout.*task\_editor\_activity*);  
  
  
 initializer();  
  
 *//initialize Gson* gson = *new* Gson();  
  
 String details = sharedPreferences.getString("identity\_task", *null*);  
 *Type* type = *new* TypeToken<ArrayList<task>>() {  
 }.getType();  
 *arrayList* = gson.fromJson(details, type);  
  
  
 *if* (*arrayList* == *null*) { *// base case  
 arrayList* = *new* ArrayList<>();  
 } *else* {  
 *//* titleTextTaskEditor.setText(*arrayList*.get(dataPosition).getTaskTitle());  
 descriptionTextTextEditor.setText(*arrayList*.get(dataPosition).getTaskDescription());  
 calendar = *arrayList*.get(dataPosition).getDue();  
 updateCalender();  
 Is24HourView = *false*; *// time picker dialogue* }  
  
  
 *if* (dataPosition != -1) {  
 titleTextTaskEditor.setText(MainActivity.*taskItemArrayList*.get(dataPosition));  
 } *else* {  
 MainActivity.*taskItemArrayList*.add("");  
 dataPosition = MainActivity.*taskItemArrayList*.size() - 1;  
 }  
  
 titleTextTaskEditor.addTextChangedListener(*new* TextWatcher() {  
 @Override  
 *public void* beforeTextChanged(*CharSequence* s, *int* start, *int* count, *int* after) {  
  
 }  
  
 @Override  
 *public void* onTextChanged(*CharSequence* s, *int* start, *int* before, *int* count) {  
 MainActivity.*taskItemArrayList*.set(dataPosition, String.*valueOf*(s));  
 MainActivity.*taskItemArrayAdapter*.notifyDataSetChanged();  
 HashSet<String> set = *new* HashSet<>(MainActivity.*taskItemArrayList*);  
 sharedPreferences.edit().putStringSet("task", set).apply();  
 }  
  
 @Override  
 *public void* afterTextChanged(*Editable* s) {  
  
 }  
 });  
  
  
 calenderButton.setOnClickListener(*new* View.OnClickListener() {  
 @Override  
 *public void* onClick(View v) {  
 datePickerDialog = *new* DatePickerDialog(taskEditorActivity.*this*,  
 *new* DatePickerDialog.OnDateSetListener() {  
  
 @Override  
 *public void* onDateSet(DatePicker view, *int* year, *int* month, *int* dayOfMonth) {  
*// calendar=Calendar.getInstance();* calendar.set(year, month, dayOfMonth);  
 Log.*i*("Calendar value Calendar", String.*valueOf*(calendar.getTime()));  
 }  
 }, Year, Month, DayOfMonth);  
 datePickerDialog.show();  
 }  
 });  
  
 *// TIME SET BUTTON* timerButtonTaskEditor.setOnClickListener(*new* View.OnClickListener() {  
 @Override  
 *public void* onClick(View v) {  
 timePickerDialog = *new* TimePickerDialog(taskEditorActivity.*this*,  
 *new* TimePickerDialog.OnTimeSetListener() {  
 @Override  
 *public void* onTimeSet(TimePicker view, *int* hourOfDay, *int* minute) {  
 calendar = Calendar.*getInstance*();  
 calendar.set(Year, Month, DayOfMonth, hourOfDay, minute);  
 *if*(hourOfDay>=0 && hourOfDay<12){  
 calendar.set(Calendar.*AM*,1);  
 } *else* {  
 *if*(hourOfDay == 12){  
 calendar.set(Calendar.*PM*,1);  
 calendar.set(Calendar.*AM*,0);  
  
 } *else*{  
 calendar.set(Calendar.*PM*,1);  
 calendar.set(Calendar.*AM*,0);  
 }  
 }  
 Log.*i*("calendar", String.*valueOf*("AM\_PM: "+ calendar.get(Calendar.*AM*) + calendar.get(Calendar.*PM*)));  
 }  
  
 }, HoursOfDay, Min, Is24HourView);  
 timePickerDialog.show();  
  
 }  
 });  
  
 }  
  
  
 *private void* setNotification(Calendar calendar) {  
 AlarmManager alarmManager = (AlarmManager) getSystemService(Context.*ALARM\_SERVICE*);  
 Intent intent = *new* Intent(*this*, AlertReceiver.*class*);  
 PendingIntent pendingIntent = PendingIntent.*getBroadcast*(*this*, 1, intent, 0);  
 intent.putExtra("identity\_task", calendar);  
 alarmManager.setExact(AlarmManager.*RTC\_WAKEUP*, calendar.getTimeInMillis(), pendingIntent);  
  
 }  
  
 *private void* cancelAlarm() {  
 AlarmManager alarmManager = (AlarmManager) getSystemService(Context.*ALARM\_SERVICE*);  
 Intent intent = *new* Intent(*this*, AlertReceiver.*class*);  
 PendingIntent pendingIntent = PendingIntent.*getBroadcast*(*this*, 1, intent, 0);  
 alarmManager.cancel(pendingIntent);  
 }  
  
  
 *public void* doneUpdate(View view) {  
  
  
 setNotification(calendar);  
 *arrayList*.get(dataPosition).setTaskTitle(titleTextTaskEditor.getText().toString());  
 *arrayList*.get(dataPosition).setTaskDescription(descriptionTextTextEditor.getText().toString());  
 *arrayList*.get(dataPosition).setDue(calendar);  
  
 String update = gson.toJson(*arrayList*);  
 sharedPreferences.edit().putString("identity\_task", update).apply();  
  
 *//storing values of date so it will show accordingly when user is on same page after setting his date. NOTE: NOT ON LOAD CASE* updateCalender();  
 newAlarm(calendar);  
 Toast.*makeText*(getApplicationContext(), "Task Updated Successfully", Toast.*LENGTH\_SHORT*).show();  
 }  
  
 *private void* newAlarm(Calendar calendar) {  
 *int* hour=calendar.get(Calendar.*HOUR*);  
 *int* min=calendar.get(Calendar.*MINUTE*);  
 *int* day=calendar.get(Calendar.*DAY\_OF\_YEAR*);  
  
 Intent intent= *new* Intent(AlarmClock.*ACTION\_SET\_ALARM*);  
 intent.putExtra(AlarmClock.*EXTRA\_HOUR*,hour);  
 intent.putExtra(AlarmClock.*EXTRA\_MINUTES*,min);  
 intent.putExtra(AlarmClock.*EXTRA\_DAYS*,day);  
  
  
  
 Calendar c = Calendar.*getInstance*();  
  
  
 *if*(calendar.get(Calendar.*AM*)== 1){  
 intent.putExtra(AlarmClock.*EXTRA\_IS\_PM*, *false*);  
 Log.*i*("calendar","First is run");  
  
 }*else*{  
 intent.putExtra(AlarmClock.*EXTRA\_IS\_PM*, *true*);  
 Log.*i*("calendar","Sec is run");  
 }  
  
 startActivity(intent);  
  
 }  
  
 *public void* updateCalender() {  
*// calendar=Calendar.getInstance();  
 // Log.i("Calender: ", String.valueOf(calendar.getTime()));* Year = calendar.get(Calendar.*YEAR*);  
 Month = calendar.get(Calendar.*MONTH*);  
 DayOfMonth = calendar.get(Calendar.*DAY\_OF\_MONTH*);  
 HoursOfDay = calendar.get(Calendar.*HOUR\_OF\_DAY*);  
 Min = calendar.get(Calendar.*MINUTE*);  
  
 *// CALENDER SET BUTTON* }  
  
 *private void* initializer() {  
 titleTextTaskEditor = findViewById(R.id.*titleTextTaskEditor*);  
 descriptionTextTextEditor = findViewById(R.id.*descriptionTaskEditor*);  
  
 calendar = Calendar.*getInstance*();  
  
 intent = getIntent();  
 dataPosition = intent.getIntExtra("dataPosition", -1);  
 sharedPreferences = getSharedPreferences("com.example.manage", *MODE\_PRIVATE*);  
  
 calenderButton = findViewById(R.id.*calenderButtonTaskEditor*);  
 timerButtonTaskEditor = findViewById(R.id.*timerButtonTaskEditor*);  
  
 Year = 0;  
 Month = 0;  
 DayOfMonth = 0;  
 HoursOfDay = 0;  
 Min = 0;  
  
 updateCalender();  
  
 Is24HourView = *false*;  
 Log.*i*("Line 142: ", String.*valueOf*(calendar.getTime()));  
 calendar.set(Year, Month, DayOfMonth, HoursOfDay, Min);  
 Log.*i*("Line 144: ", String.*valueOf*(calendar.getTime()));  
  
 }  
  
 *public void* setOnetimeTimer(Context context){  
 AlarmManager am=(AlarmManager)context.getSystemService(Context.*ALARM\_SERVICE*);  
 Intent intent = *new* Intent(context, AlertReceiver.*class*);  
 intent.putExtra("Shahzaib", Boolean.*TRUE*);  
 PendingIntent pi = PendingIntent.*getBroadcast*(context, 0, intent, 0);  
 am.set(AlarmManager.*RTC\_WAKEUP*, System.*currentTimeMillis*(), pi);  
 }  
  
 *public void* setPriority(View view) {  
 }  
  
 *public void* setMathMission(View view) {  
 Intent intent = *new* Intent(*this*, math\_mission\_setting.*class*);  
 intent.putExtra("position", dataPosition);  
 startActivity(intent);  
 }  
}

}

### activity\_math\_mission.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*=".math\_mission">  
  
 <TextView  
 *android:id*="@+id/queryTextView"  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:text*="@string/query"  
 *android:textSize*="60sp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.5"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.325"  
 *tools:ignore*="MissingConstraints" />  
  
  
 <TextView  
 *android:id*="@+id/timerTextView"  
 *android:layout\_width*="100sp"  
 *android:layout\_height*="100sp"  
 *android:background*="@drawable/custom\_circle\_button"  
 *android:paddingStart*="23sp"  
 *android:paddingTop*="25dp"  
 *android:text*="@string/timerText"  
 *android:textColor*="@color/primary\_text\_dark"  
 *android:textSize*="35sp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.1"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.1"  
 *tools:ignore*="RtlSymmetry" />  
  
 <TextView  
 *android:id*="@+id/scoreTextView"  
 *android:layout\_width*="100dp"  
 *android:layout\_height*="100dp"  
 *android:background*="@drawable/custom\_circle\_button"  
 *android:paddingStart*="23dp"  
 *android:paddingTop*="25dp"  
 *android:text*="@string/scoreText"  
 *android:textColor*="@color/primary\_text\_dark"  
 *android:textSize*="35sp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.9"  
 *app:layout\_constraintStart\_toEndOf*="@+id/timerTextView"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.1" />  
  
 <EditText  
 *android:id*="@+id/answerEditText"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_marginStart*="32dp"  
 *android:layout\_marginTop*="32dp"  
 *android:layout\_marginEnd*="32dp"  
 *android:layout\_marginBottom*="32dp"  
 *android:hint*="Enter Correct Solution"  
 *android:gravity*="center"  
 *android:inputType*="number"  
 *android:textSize*="24sp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.5"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.5" />  
  
  
</androidx.constraintlayout.widget.ConstraintLayout>

### math\_mission.java

*package* com.example.manageme;  
  
*import* androidx.appcompat.app.AppCompatActivity;  
  
*import* android.os.Bundle;  
*import* android.os.CountDownTimer;  
*import* android.view.View;  
*import* android.widget.Button;  
*import* android.widget.EditText;  
*import* android.widget.TextView;  
*import* android.widget.Toast;  
  
*import* java.util.ArrayList;  
*import* java.util.Random;  
  
*public class* math\_mission *extends* AppCompatActivity {  
  
 *// Variable deceleration* TextView timerTextView,scoreTextView, queryTextView;  
 EditText answerEditText;  
 ArrayList<String> symbolArray=*new* ArrayList<String>();  
 Random randomQueryVariable;  
  
 CountDownTimer countDownTimer;  
  
 *int* score;  
  
 *boolean* stopGame;  
  
 *public void* start(View view){  
 *// text views initialized* queryTextView.setVisibility(View.*VISIBLE*);  
 newQuestion(view);  
 }  
  
 *public void* timer(View view){  
 countDownTimer=*new* CountDownTimer(30000,1000) {  
 @Override  
 *public void* onTick(*long* millisUntilFinished) {  
 timerTextView.setText(millisUntilFinished / 1000 +"s");  
 }  
 @Override  
 *public void* onFinish() {  
 scoreTextView.setText(Integer.*toString*(score) +"/"+Integer.*toString*(3));  
 newQuestion(view);  
 }  
 };  
 }  
  
  
 *public void* newQuestion(View view){  
 timer(view);  
 countDownTimer.start();  
  
 *int* first= randomQueryVariable.nextInt(21);  
 *int* second= randomQueryVariable.nextInt(21);  
 *int* indexForFunction= randomQueryVariable.nextInt(4);  
  
 queryTextView.setText(first+" "+symbolArray.get(indexForFunction)+" "+second);  
 }  
  
  
  
 @Override  
 *protected void* onCreate(Bundle savedInstanceState) {  
 *super*.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_math\_mission*);  
 initializer();  
 setSymbolArray();  
  
 }  
 *private void* initializer() {  
 *// text views initialized* timerTextView=findViewById(R.id.*timerTextView*);  
 scoreTextView=findViewById(R.id.*scoreTextView*);  
 queryTextView=findViewById(R.id.*queryTextView*);  
 answerEditText=findViewById(R.id.*answerEditText*);  
 randomQueryVariable =*new* Random();  
 stopGame=*false*;  
 }  
  
 *private void* setSymbolArray(){  
 symbolArray.add("+");  
 symbolArray.add("-");  
 }  
  
}

### \_math\_mission\_setting.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"  
 *android:paddingLeft*="32dp"  
 *android:paddingRight*="32dp"  
 *tools:context*=".math\_mission\_setting">  
  
 <TextView  
 *android:id*="@+id/titleMathMissionSetting"  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:gravity*="center"  
 *android:text*="Math Mission"  
 *android:textColor*="@color/schema\_heading"  
 *android:textSize*="36sp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintLeft\_toLeftOf*="parent"  
 *app:layout\_constraintRight\_toRightOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.1" />  
  
 <TextView  
 *android:id*="@+id/mathQueryDifficultPreviewTextView"  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:gravity*="center"  
 *android:text*="10+2"  
 *android:textColor*="@color/schema\_heading"  
 *android:textSize*="54sp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintLeft\_toLeftOf*="parent"  
 *app:layout\_constraintRight\_toRightOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.3" />  
  
 <SeekBar  
 *android:id*="@+id/difficultyLevelSeekBar"  
 *style*="@style/Widget.AppCompat.SeekBar.Discrete"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_marginStart*="32dp"  
 *android:layout\_marginEnd*="32dp"  
 *android:textAlignment*="center"  
 *android:max*="2"  
 *android:progress*="3"  
 *android:thumbTint*="@color/schema\_heading"  
 *android:progressTint*="@color/primary\_text\_light"  
  
  
  
 *android:thumb*="@drawable/ic\_baseline\_fiber\_manual\_record\_24"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintLeft\_toLeftOf*="parent"  
 *app:layout\_constraintRight\_toRightOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.4" />  
  
  
 <TextView  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:text*="Easy"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.1"  
 *app:layout\_constraintLeft\_toLeftOf*="parent"  
 *app:layout\_constraintRight\_toRightOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.425" />  
  
 <TextView  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:text*="Normal"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.5"  
 *app:layout\_constraintLeft\_toLeftOf*="parent"  
 *app:layout\_constraintRight\_toRightOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.425" />  
  
 <TextView  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:text*="Hard"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintHorizontal\_bias*="0.9"  
 *app:layout\_constraintLeft\_toLeftOf*="parent"  
 *app:layout\_constraintRight\_toRightOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.425" />  
 <TextView  
 *android:id*="@+id/numberOfQuestionTextView"  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:gravity*="center"  
 *android:text*="Number of Questions"  
 *android:textColor*="@color/schema\_heading"  
 *android:textSize*="24sp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintLeft\_toLeftOf*="parent"  
 *app:layout\_constraintRight\_toRightOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.7" />  
  
 <android.widget.GridLayout  
 *android:id*="@+id/numberOfQuestions"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="wrap\_content"  
  
  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintLeft\_toLeftOf*="parent"  
 *app:layout\_constraintRight\_toRightOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent"  
 *app:layout\_constraintVertical\_bias*="0.85">  
  
 <Button  
 *android:id*="@+id/oneQuestion"  
 *android:layout\_width*="72dp"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_row*="0"  
 *android:layout\_rowWeight*="1"  
 *android:layout\_column*="0"  
 *android:layout\_columnWeight*="1"  
 *android:layout\_gravity*="fill"  
 *android:layout\_margin*="10dp"  
 *android:background*="@color/schema\_heading"  
 *android:tag*="1"  
 *android:text*="1"  
 *android:textSize*="18sp" />  
  
 <Button  
 *android:id*="@+id/twoQuestion"  
 *android:layout\_width*="72dp"  
 *android:layout\_height*="wrap\_content"  
  
 *android:layout\_row*="0"  
 *android:layout\_rowWeight*="1"  
 *android:layout\_column*="1"  
 *android:layout\_columnWeight*="1"  
 *android:layout\_gravity*="fill"  
 *android:layout\_margin*="10dp"  
 *android:background*="@color/schema\_heading"  
 *android:tag*="2"  
 *android:text*="2"  
 *android:textSize*="18sp"  
  
 />  
  
 <Button  
 *android:id*="@+id/threeQuestion"  
 *android:layout\_width*="72dp"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_row*="0"  
 *android:layout\_rowWeight*="1"  
 *android:layout\_column*="3"  
 *android:layout\_columnWeight*="1"  
 *android:layout\_gravity*="fill"  
 *android:layout\_margin*="10dp"  
 *android:background*="@color/schema\_heading"  
 *android:tag*="3"  
 *android:text*="3"  
  
 />  
 </android.widget.GridLayout>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

### Math\_mission\_setting.java

*package* com.example.manageme;  
  
*import* androidx.appcompat.app.AppCompatActivity;  
  
*import* android.content.Intent;  
*import* android.content.*SharedPreferences*;  
*import* android.graphics.drawable.Drawable;  
*import* android.os.Bundle;  
*import* android.util.Log;  
*import* android.widget.Button;  
*import* android.widget.SeekBar;  
  
*import* com.google.gson.Gson;  
*import* com.google.gson.reflect.TypeToken;  
  
*import* java.lang.reflect.*Type*;  
*import* java.util.ArrayList;  
  
*public class* math\_mission\_setting *extends* AppCompatActivity {  
  
 Intent intent;  
 *SharedPreferences* sharedPreferences;  
 ArrayList<task> arrayList;  
 *int* position;  
 SeekBar seekBar;  
 Gson gson;  
 *Type* type;  
 String detail;  
 Button button1;  
 Button button2;  
 Button button3;  
 MathMission mathMissionTask;  
 *int* numberOfQuestions;  
  
  
 @Override  
 *public void* onCreate(Bundle savedInstanceState) {  
 *super*.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 initializer();  
 gson = *new* Gson();  
 type = *new* TypeToken<ArrayList<task>>() {  
 }.getType();  
  
 arrayList = gson.fromJson(detail, type);  
 *if* (arrayList == *null*) { *// base case* arrayList = *new* ArrayList<>();  
 } *else* {  
 *//seekBar.setProgress(arrayList.get(position).getMathMission().getDifficultLevel());  
 try* {  
 numberOfQuestions = arrayList.get(position).getMathMission().getNumberOfQuestions();  
 } *catch* (Exception e) {  
 mathMissionTask.setDifficultLevel(1); *// set default value first when --> already not set* mathMissionTask.setNumberOfQuestions(1);  
 }  
  
  
  
 arrayList.get(position).setMathMission(mathMissionTask);  
 String update = gson.toJson(arrayList);  
 sharedPreferences.edit().putString("identity\_task",detail ).apply();  
 }  
  
 *//seekBar.setProgress(1);  
  
// seekBar.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
// @Override  
// public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
// Log.i("Progress: ", String.valueOf(progress));  
// }  
//  
// @Override  
// public void onStartTrackingTouch(SeekBar seekBar) {  
//  
// }  
//  
// @Override  
// public void onStopTrackingTouch(SeekBar seekBar) {  
//  
// }  
// });* }  
  
 *private void* initializer() {  
 intent = getIntent();  
 seekBar=findViewById(R.id.*difficultyLevelSeekBar*);  
 Log.*i*("seek bar ", String.*valueOf*(seekBar.getProgress()));  
 sharedPreferences = getSharedPreferences("com.example.manage", *MODE\_PRIVATE*);  
 position = intent.getIntExtra("position", -1);  
 mathMissionTask = *new* MathMission();  
 detail = sharedPreferences.getString("identity\_task", *null*);  
 button1=findViewById(R.id.*oneQuestion*);  
 button2=findViewById(R.id.*twoQuestion*);  
 button3=findViewById(R.id.*threeQuestion*);  
  
 }  
  
}

## Android Manifest

<?*xml version*="1.0" *encoding*="utf-8"?>  
<manifest *xmlns:android*="http://schemas.android.com/apk/res/android"  
 *package*="com.example.manageme">  
  
 <uses-permission *android:name*="com.android.alarm.permission.SET\_ALARM"/>  
 <application  
 *android:allowBackup*="true"  
 *android:icon*="@mipmap/ic\_launcher"  
 *android:label*="@string/app\_name"  
 *android:roundIcon*="@mipmap/ic\_launcher\_round"  
 *android:supportsRtl*="true"  
 *android:theme*="@style/Theme.ManageMe">  
 <activity *android:name*=".math\_mission\_setting"></activity>  
 <activity *android:name*=".math\_mission" />  
 <activity *android:name*=".splashView">  
 <intent-filter>  
 <action *android:name*="android.intent.action.MAIN" />  
 <category *android:name*="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 <activity *android:name*=".WelcomePage" />  
 <activity *android:name*=".taskEditorActivity" />  
 <activity *android:name*=".MainActivity" />  
 <receiver *android:name*=".AlertReceiver"/>  
 <meta-data  
 *android:name*="preloaded\_fonts"  
 *android:resource*="@array/preloaded\_fonts" />  
 </application>  
  
</manifest>

## Supporting Class

### MathMission.java

*package* com.example.manageme;  
  
*public class* MathMission {  
  
 *int* difficultLevel;  
 String numberOfQuestions;  
 *public* MathMission() {  
 *this*.difficultLevel = 0;  
 *this*.numberOfQuestions = *null*;  
 }  
 *public* MathMission(*int* difficultLevel, String numberOfQuestions) {  
 *this*.difficultLevel = difficultLevel;  
 *this*.numberOfQuestions = numberOfQuestions;  
 }  
  
 *public int* getDifficultLevel() {  
 *return* difficultLevel;  
 }  
  
 *public void* setDifficultLevel(*int* difficultLevel) {  
 *this*.difficultLevel = difficultLevel;  
 }  
  
 *public* String getNumberOfQuestions() {  
 *return* numberOfQuestions;  
 }  
  
 *public void* setNumberOfQuestions(String numberOfQuestions) {  
 *this*.numberOfQuestions = numberOfQuestions;  
 }  
}

### task.java

*package* com.example.manageme;  
  
*import* android.renderscript.RenderScript;  
  
*import* java.sql.Time;  
*import* java.util.Calendar;  
  
*public class* task {  
  
 String taskTitle;  
 String taskDescription;  
 Calendar due;  
 Boolean priority;  
 MathMission mathMission;  
  
 *public* MathMission getMathMission() {  
 *return* mathMission;  
 }  
  
 *public void* setMathMission(MathMission mathMission) {  
 *this*.mathMission = mathMission;  
 }  
  
 *public* Calendar getDue() {  
 *return* due;  
 }  
  
 *public void* setDue(Calendar due) {  
 *this*.due = due;  
 }  
  
 *public* Boolean getPriority() {  
 *return* priority;  
 }  
  
 *public void* setPriority(Boolean priority) {  
 *this*.priority = priority;  
 }  
  
 *public* task(String taskTitle, String taskDescription) {  
 *this*.taskTitle = taskTitle;  
 *this*.taskDescription = taskDescription;  
 }  
  
 *public* String getTaskTitle() {  
 *return* taskTitle;  
 }  
  
 *public* String getTaskDescription() {  
 *return* taskDescription;  
 }  
  
 *public void* setTaskTitle(String taskTitle) {  
 *this*.taskTitle = taskTitle;  
 }  
  
 *public void* setTaskDescription(String taskDescription) {  
 *this*.taskDescription = taskDescription;  
 }  
  
}

### AlertReciver,java

*package* com.example.manageme;  
  
*import* android.content.BroadcastReceiver;  
*import* android.content.Context;  
*import* android.content.Intent;  
  
*import* androidx.core.app.NotificationCompat;  
  
*public class* AlertReceiver *extends* BroadcastReceiver {  
 Intent intent;  
 @Override  
 *public void* onReceive(Context context, Intent intent) {  
  
 NotificationHelper notificationHelper = *new* NotificationHelper(context);  
 NotificationCompat.Builder notificationBuilder = notificationHelper.getChannelNotification();  
 notificationHelper.getManager().notify(1, notificationBuilder.build());  
 }  
}

### NotificationHelper.java

*package* com.example.manageme;  
  
*import* android.annotation.TargetApi;  
*import* android.app.NotificationChannel;  
*import* android.app.NotificationManager;  
*import* android.content.Context;  
*import* android.content.ContextWrapper;  
*import* android.os.Build;  
*import* androidx.core.app.NotificationCompat;  
  
*public class* NotificationHelper *extends* ContextWrapper {  
  
  
 *public static final* String *channelID* = "channelID";  
 *public static final* String *channelName* = "Manage Me";  
 *private* NotificationManager notificationManager;  
  
 *public* NotificationHelper(Context base) {  
 *super*(base);  
 *if* (Build.VERSION.*SDK\_INT* >= Build.VERSION\_CODES.*O*) {  
 createChannel();  
 }  
 }  
  
 @TargetApi(Build.VERSION\_CODES.*O*)  
 *private void* createChannel() {  
 NotificationChannel channel = *new* NotificationChannel(*channelID*, *channelName*, NotificationManager.*IMPORTANCE\_HIGH*);  
 getManager().createNotificationChannel(channel);  
 }  
 *public* NotificationManager getManager() {  
 *//if not initialized then initialize it will access  
 if* (notificationManager == *null*) {  
 notificationManager = (NotificationManager) getSystemService(Context.*NOTIFICATION\_SERVICE*);  
 }  
 *return* notificationManager;  
 }  
  
 *public* NotificationCompat.Builder getChannelNotification() {  
 *return new* NotificationCompat.Builder(getApplicationContext(), *channelID*)  
 .setContentTitle("Alarm!")  
 .setContentText("You have a task pending !!")  
 .setSmallIcon(R.drawable.*ic\_baseline\_alarm\_24*);  
 }  
  
  
}

## Resource Files

### Color.xml

<?*xml version*="1.0" *encoding*="utf-8"?>  
<resources>  
 <color *name*="purple\_200">#FFBB86FC</color>  
 <color *name*="purple\_500">#FF6200EE</color>  
 <color *name*="purple\_700">#FF3700B3</color>  
 <color *name*="teal\_200">#FF03DAC5</color>  
 <color *name*="teal\_700">#FF018786</color>  
 <color *name*="black">#FF000000</color>  
 <color *name*="white">#FFFFFFFF</color>  
  
  
 <color *name*="schema\_heading">#2276E6</color>  
 *<!-- light mode -->* <color *name*="selected\_edit\_text\_light">#eff0f2</color>  
 <color *name*="light\_mode">#f9f9fa</color>  
 <color *name*="primary\_text\_light">#222222</color>  
 <color *name*="secondary\_text\_light">#455965</color>  
  
 *<!-- dark mode -->* <color *name*="selected\_edit\_text\_dark">#282a33</color>  
 <color *name*="dark\_mode">#32343d</color>  
 <color *name*="primary\_text\_dark">#ffffff</color>  
 <color *name*="secondary\_text\_dark">#b0bdc6</color>  
  
</resources>

### Font\_cert.xml

<?*xml version*="1.0" *encoding*="utf-8"?>  
<resources>  
 <color *name*="purple\_200">#FFBB86FC</color>  
 <color *name*="purple\_500">#FF6200EE</color>  
 <color *name*="purple\_700">#FF3700B3</color>  
 <color *name*="teal\_200">#FF03DAC5</color>  
 <color *name*="teal\_700">#FF018786</color>  
 <color *name*="black">#FF000000</color>  
 <color *name*="white">#FFFFFFFF</color>  
  
  
 <color *name*="schema\_heading">#2276E6</color>  
 *<!-- light mode -->* <color *name*="selected\_edit\_text\_light">#eff0f2</color>  
 <color *name*="light\_mode">#f9f9fa</color>  
 <color *name*="primary\_text\_light">#222222</color>  
 <color *name*="secondary\_text\_light">#455965</color>  
  
 *<!-- dark mode -->* <color *name*="selected\_edit\_text\_dark">#282a33</color>  
 <color *name*="dark\_mode">#32343d</color>  
 <color *name*="primary\_text\_dark">#ffffff</color>  
 <color *name*="secondary\_text\_dark">#b0bdc6</color>  
  
</resources>

### Preloaded\_fonts.xml

<?*xml version*="1.0" *encoding*="utf-8"?>  
<resources>  
 <array *name*="com\_google\_android\_gms\_fonts\_certs">  
 <item>@array/com\_google\_android\_gms\_fonts\_certs\_dev</item>  
 <item>@array/com\_google\_android\_gms\_fonts\_certs\_prod</item>  
 </array>  
 <string-array *name*="com\_google\_android\_gms\_fonts\_certs\_dev">  
 <item>  
   
 </item>  
 </string-array>  
 <string-array *name*="com\_google\_android\_gms\_fonts\_certs\_prod">  
 <item>  
   
 </item>  
 </string-array>  
</resources>

# References

Coding in flow : <https://www.youtube.com/watch?v=tTbd1Mfi-Sk&t=434s>

System Alarm : <https://www.youtube.com/watch?v=LsKjw-IJQpI>