Технически университет - София

Документация на Курсов проект

по дисциплина „Програмиране за мобилни устройства“

На тема „Truth Or Dare“

Изработиле:

Велко Миланов, КСИ, гр. 44, ф-н № 123220014

Деян Павловски.КСИ,гр. 44, ф-н № 123220023

Съдържание

3. ................................................................................................Увод

4. ..........................................Анализ на съществуващи разработки

5. ...................................................................................Проектиране

8. ..................................................................Софтуерна реализация

19. ......................................Потребителско ръководство(тестване)

23. ...................................................................................Заключение

23. ……………............................................Използвана литература

Увод

Truth or Dare е популярна игра, която се играе от хора по целия свят. Тази апликация предоставя уникален начин за игра на тази забавна игра, която може да бъде играна с приятели или със случайни хора от целия свят.

С помощта на тази апликация, можете да изберете дали да играете Truth или Dare. Ако изберете Truth, ще бъдете задавани въпроси, на които трябва да отговорите честно. Ако изберете Dare, ще бъдете предизвикани да изпълните определена задача или действие.

Тази апликация е идеална за забавление на парти, вечери или когато желаете да се забавлявате с приятели. Играта е създадена за всички възрасти и можете да я играете с когото пожелаете.

Truth or Dare е игра, която предизвиква хората да се забавляват и да се отварят един на друг. Тази апликация предлага голям брой въпроси и предизвикателства, които са подходящи за всички възрасти и за различни обстановки.

Така че, ако искате да се забавлявате с приятели или да срещнете нови хора и да се запознаете по-добре, тази апликация може да бъде перфектният избор за вас. Просто изберете Truth or Dare и започнете да се забавлявате.

Анализ на съществуващи разработки

1.Truth or Dare – Free party game (iOS, Android)

Положителни страни:

Възможност за персонализация на въпроси и предизвикателства;

Лесен за използване интерфейс;

Възможност за избор на категории на въпросите или предизвикателствата.

Отрицателни страни:

Не са предоставени достатъчно въпроси или предизвикателства;

Неинтуитивен начин за добавяне на собствени въпроси или предизвикателства;

Приложението има реклами, които могат да бъдат досадни.

2.Truth or Dare – Dirty & Couple Game (iOS, Android)

Положителни страни:

Възможност за избор на категория на въпросите или предизвикателствата;

Съдържа много въпроси и предизвикателства;

Предлага различни нива на трудност за въпросите или предизвикателствата.

Отрицателни страни:

Приложението има много реклами, които могат да бъдат досадни

Интерфейсът може да бъде по-интуитивен

Не предоставя възможност за персонализация на играта.

Обобщавайки, можем да кажем, че всички приложения предлагат различни ползи и имат своите отрицателни страни. Някои от тях предлагат персонализация на играта, като добавяне на собствени въпроси или предизвикателства, докато други не предлагат тази опция. Всички приложения имат разнообразие от въпроси и предизвикателства, но при някои може да има повторения. Интуитивният интерфейс е предимство за всички приложения, но много реклами може да доведат до досада и да намалят потребителското изживяване.

Проектиране

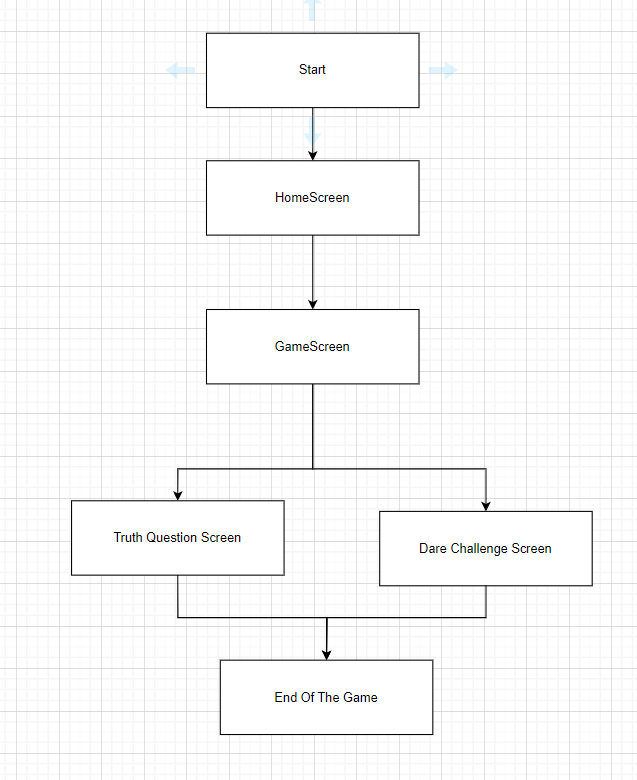
Приложението е разработено с помощта на “Android Studio”, приложение с отворен код за разработка на Android приложения. Като такова, в основата си то използва езикът за програмиране Java за осъществяване на автоматизирани дейности в себе си и XML файлове за описание на външния вид на потребителския си интерфейс.

Продуктът ще бъде използван от млади хора, насочени към забавление и игра, най-вероятно в групи или на партия. Поради тази причина,има опция и възможност за персонализация на играта, за да се отговори на предпочитанията на различни групи от потребители.

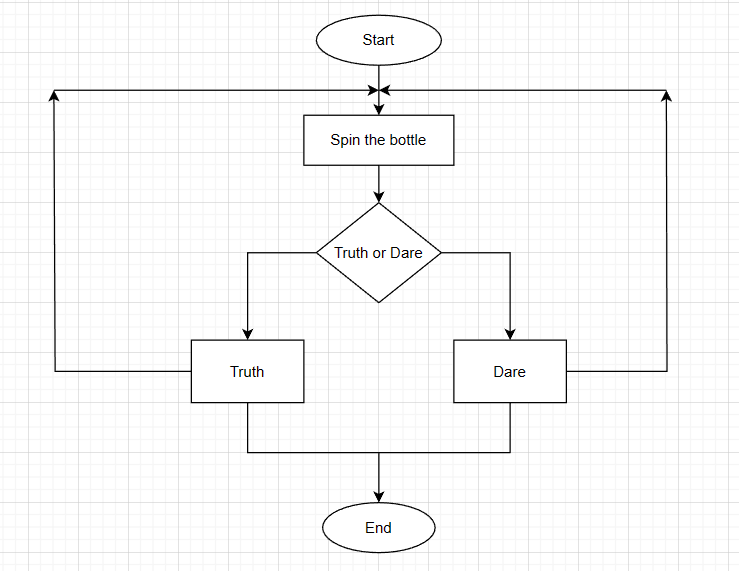
Входните данни за приложението ще бъдат въпроси и предизвикателства. За да се управляват тези данни, изградени са следните структури: "Въпроси" и "Предизвикателства".Въпросите и предизвикателствата ще се съхраняват в низ от стригове,има изграден модел на класовете за тези две същности.

Приложението достъпно за потребителя,има изграден потребителски интерфейс с интуитивна навигация. Главният екран включва опции за стартиране на игра и персонализация на играта. Основната страница на играта включва опция за избор на "Truth" или "Dare",също има и опция за добавяне на нови "Въпроси" и "Предизвикателства".

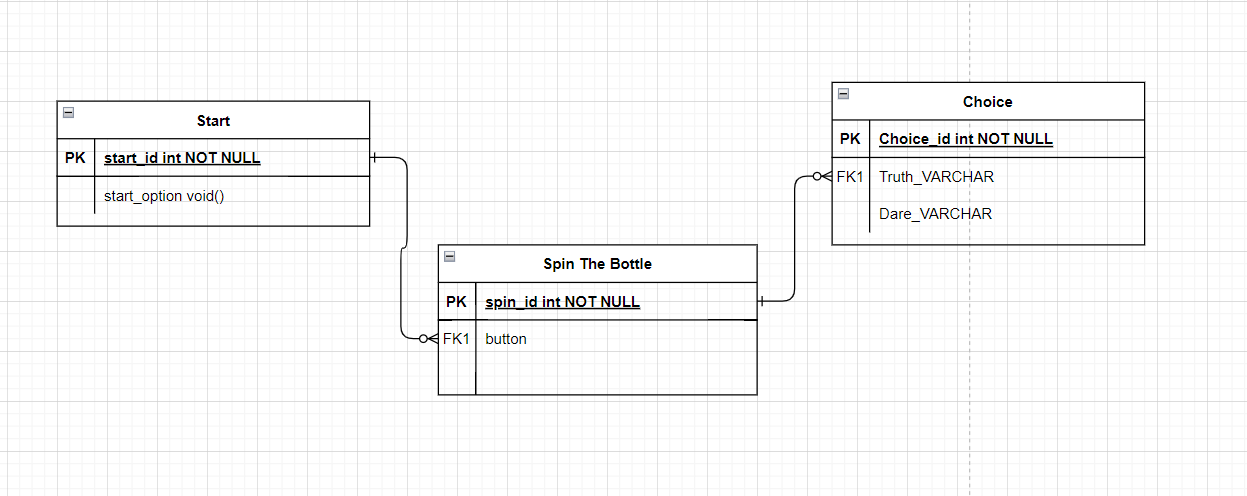
Блокова Схема:



Алгоритъм:



ER диаграм:



Софтуерна реализация

1.Клас “StartActivity”:

public class StartActivity extends AppCompatActivity {

private Button start, truth, dare;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

//setRequestedOrientation(ActivityInfo.SCREEN\_ORIENTATION\_PORTRAIT);

setContentView(R.layout.activity\_start);

start = findViewById(R.id.start);

truth = findViewById(R.id.truth);

dare = findViewById(R.id.dare);

start.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), MainActivity.class));

}

});

truth.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), TruthActivity.class));

}

});

dare.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), DareActivity.class));

}

});

}

}

В StartActivity класа има три бутона: start, truth и dare. Когато потребителят натисне бутона start, те ще бъдат прехвърлени към MainActivity класа. Ако потребителят натисне бутона truth, те ще бъдат прехвърлени към TruthActivity класа. Ако потребителят натисне бутона dare, те ще бъдат прехвърлени към DareActivity класа.

2.Клас “MainActivity“:

public class MainActivity extends AppCompatActivity {

private Button btn, truthBtn, dareBtn;

private ImageView imgView;

private Random random = new Random();

private int lastDirection;

private MediaPlayer mp;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

btn = findViewById(R.id.button);

truthBtn = findViewById(R.id.btn1);

dareBtn = findViewById(R.id.btn2);

imgView = findViewById(R.id.imageView);

truthBtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), TruthActivity.class));

}

});

dareBtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

startActivity(new Intent(getApplicationContext(), DareActivity.class));

}

});

}

@Override

protected void onResume() {

super.onResume();

truthBtn.setEnabled(false);

dareBtn.setEnabled(false);

btn.setEnabled(true);

}

public void spin(View view) {

int newDirection = random.nextInt(5400);

float pivotX = imgView.getWidth()/2;

float pivotY = imgView.getHeight()/2;

Animation rotate = new RotateAnimation(lastDirection, newDirection, pivotX, pivotY);

rotate.setDuration(2000);

rotate.setFillAfter(true);

rotate.setAnimationListener(new Animation.AnimationListener() {

@Override

public void onAnimationStart(Animation animation) {

mp = MediaPlayer.create(MainActivity.this, R.raw.audio);

mp.start();

btn.setEnabled(false);

}

@Override

public void onAnimationEnd(Animation animation) {

mp.stop();

mp.release();

mp = null;

truthBtn.setEnabled(true);

dareBtn.setEnabled(true);

}

@Override

public void onAnimationRepeat(Animation animation) {

}

});

lastDirection = newDirection;

imgView.startAnimation(rotate);

}

}

В MainActivity класа има три бутона: btn, truthBtn и dareBtn, както и един ImageView: imgView. Когато потребителят натисне бутона truthBtn, те ще бъдат прехвърлени към TruthActivity класа. Ако потребителят натисне бутона dareBtn, те ще бъдат прехвърлени към DareActivity класа.

Методът spin() се използва, когато потребителят натисне бутона btn, който ще върти бутилката с помощта на анимация. Методът включва звуков ефект при старта на анимацията и спиране на звука, когато анимацията приключи.

В onResume() метода са зададени началните стойности на бутоните, като truthBtn и dareBtn са забранени, а btn е активен.

3.Клас “TruthActivity”:

public class TruthActivity extends AppCompatActivity {

private ArrayList<TruthItem> truthList;

private Toolbar toolbar;

private RecyclerView recyclerView;

private RecyclerView.Adapter adapter;

private RecyclerView.LayoutManager layoutManager;

SharedPreferences sharedPreferences;

SharedPreferences.Editor editor;

Gson gson = new Gson();

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_values);

sharedPreferences = getSharedPreferences("mySharedPreference", Context.MODE\_PRIVATE);

editor = sharedPreferences.edit();

toolbar = findViewById(R.id.toolbar);

setSupportActionBar(toolbar);

truthList = new ArrayList<>();

recyclerViewConfig();

populateDefaultData();

if(sharedPreferences.contains("UserTruths"))

populateUserData(sharedPreferences.getString("UserTruths", null));

}

public void populateDefaultData() {

Values values = new Values();

for(int i=0; i<values.truths.length; i++)

truthList.add(new TruthItem(values.truths[i]));

}

public void populateUserData(String jsonTruths) {

String[] truths = gson.fromJson(jsonTruths, String[].class);

for(int i=0; i<truths.length; i++)

truthList.add(new TruthItem(truths[i]));

}

public void recyclerViewConfig() {

// config for RV

recyclerView = findViewById(R.id.recyclerView);

//performance

recyclerView.setHasFixedSize(true);

layoutManager = new LinearLayoutManager(this);

adapter = new TruthAdapter(truthList);

recyclerView.setLayoutManager(layoutManager);

recyclerView.setAdapter(adapter);

}

public void showDialog() {

final Dialog dialog = new Dialog(this);

dialog.requestWindowFeature(Window.FEATURE\_NO\_TITLE);

dialog.setCancelable(false);

dialog.setContentView(R.layout.custom\_dialog\_box);

dialog.getWindow().setBackgroundDrawable(new ColorDrawable(Color.TRANSPARENT));

// setup buttons

final EditText input = dialog.findViewById(R.id.editText);

Button dismiss = dialog.findViewById(R.id.dismiss);

Button add = dialog.findViewById(R.id.add);

add.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String mText = input.getText().toString();

if(mText.isEmpty())

Toast.makeText(getApplicationContext(), "Empty Text", Toast.LENGTH\_LONG).show();

else{

updateUserData(mText);

adapter.notifyDataSetChanged();

Toast.makeText(getApplicationContext(), "Successfully Added", Toast.LENGTH\_SHORT).show();

}

dialog.dismiss();

}

});

dismiss.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

dialog.cancel();

}

});

dialog.show();

}

public void updateUserData(String string) {

ArrayList<String> textList = new ArrayList<>();

if(sharedPreferences.contains("UserTruths")) {

String jsonTruths = sharedPreferences.getString("UserTruths", null);

String[] truths = gson.fromJson(jsonTruths, String[].class);

for(int i=0; i<truths.length; i++)

textList.add(truths[i]);

}

textList.add(string);

editor.putString("UserTruths", gson.toJson(textList));

editor.apply();

truthList.add(new TruthItem(string));

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.menu, menu);

return super.onCreateOptionsMenu(menu);

}

@Override

public boolean onOptionsItemSelected(@NonNull MenuItem item) {

switch (item.getItemId()) {

case R.id.action\_add:

showDialog();

return true;

default:

return super.onOptionsItemSelected(item);

}

}

}

Този код представлява активност в нашето приложение, която позволява на потребителите да добавят или изтриват въпроси за истината. Класът има RecyclerView, който показва списъка с въпроси за истината и има диалогов прозорец, който се появява, когато потребителят иска да добави нов въпрос. Добавените въпроси за истината се запазват в SharedPreferences.

4.Клас “DareActivity”:

public class DareActivity extends AppCompatActivity {

private ArrayList<TruthItem> truthList;

private Toolbar toolbar;

private RecyclerView recyclerView;

private RecyclerView.Adapter adapter;

private RecyclerView.LayoutManager layoutManager;

SharedPreferences sharedPreferences;

SharedPreferences.Editor editor;

Gson gson = new Gson();

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_values);

sharedPreferences = getSharedPreferences("mySharedPreference", Context.MODE\_PRIVATE);

editor = sharedPreferences.edit();

toolbar = findViewById(R.id.toolbar);

setSupportActionBar(toolbar);

truthList = new ArrayList<>();

recyclerViewConfig();

populateDefaultData();

if(sharedPreferences.contains("UserDares"))

populateUserData(sharedPreferences.getString("UserDares", null));

}

public void populateDefaultData() {

Values values = new Values();

for(int i=0; i<values.dares.length; i++)

truthList.add(new TruthItem(values.dares[i]));

}

public void populateUserData(String jsonDares) {

String[] dares = gson.fromJson(jsonDares, String[].class);

for(int i=0; i<dares.length; i++)

truthList.add(new TruthItem(dares[i]));

}

public void recyclerViewConfig() {

// config for RV

recyclerView = findViewById(R.id.recyclerView);

//performance

recyclerView.setHasFixedSize(true);

layoutManager = new LinearLayoutManager(this);

adapter = new TruthAdapter(truthList);

recyclerView.setLayoutManager(layoutManager);

recyclerView.setAdapter(adapter);

}

public void showDialog() {

final Dialog dialog = new Dialog(this);

dialog.requestWindowFeature(Window.FEATURE\_NO\_TITLE);

dialog.setCancelable(false);

dialog.setContentView(R.layout.custom\_dialog\_box);

dialog.getWindow().setBackgroundDrawable(new ColorDrawable(Color.TRANSPARENT));

// setup buttons

final EditText input = dialog.findViewById(R.id.editText);

Button dismiss = dialog.findViewById(R.id.dismiss);

Button add = dialog.findViewById(R.id.add);

add.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String mText = input.getText().toString();

if(mText.isEmpty())

Toast.makeText(getApplicationContext(), "Empty Text", Toast.LENGTH\_LONG).show();

else{

updateUserData(mText);

adapter.notifyDataSetChanged();

Toast.makeText(getApplicationContext(), "Successfully Added", Toast.LENGTH\_SHORT).show();

}

dialog.dismiss();

}

});

dismiss.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

dialog.cancel();

}

});

dialog.show();

}

public void updateUserData(String string) {

ArrayList<String> textList = new ArrayList<>();

if(sharedPreferences.contains("UserDares")) {

String jsonDares = sharedPreferences.getString("UserDares", null);

String[] dares = gson.fromJson(jsonDares, String[].class);

for(int i=0; i<dares.length; i++)

textList.add(dares[i]);

}

textList.add(string);

editor.putString("UserDares", gson.toJson(textList));

editor.apply();

truthList.add(new TruthItem(string));

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.menu, menu);

return super.onCreateOptionsMenu(menu);

}

@Override

public boolean onOptionsItemSelected(@NonNull MenuItem item) {

switch (item.getItemId()) {

case R.id.action\_add:

//TODO: add

showDialog();

return true;

default:

return super.onOptionsItemSelected(item);

}

}

}

В Dare Activity се създава списък с действия и има функционалност за добавяне на нови действия от потребителя. Списъкът се пази във SharedPreferences и се актуализира в RecyclerView, който се конфигурира в метода recyclerViewConfig(). Когато потребителят кликне върху бутона за добавяне, се показва диалогов прозорец, който съдържа текстово поле и два бутона - "Добави" и "Отмени". Ако потребителят въведе текст в полето и кликне върху "Добави", текстът се добавя към списъка и се актуализира RecyclerView.

5.Клас“Activity\_Main.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">

<ImageView

android:id="@+id/imageView"

android:layout\_width="160dp"

android:layout\_height="230dp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintBottom\_toTopOf="@id/button"

app:srcCompat="@drawable/spin" />

<Button

android:id="@+id/button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/spin"

android:textColor="#ffffff"

android:textSize="18dp"

android:onClick="spin"

app:layout\_constraintBottom\_toTopOf="@id/linearLayout"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/imageView"/>

<LinearLayout

android:id="@+id/linearLayout"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:paddingHorizontal="10dp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintTop\_toBottomOf="@id/button">

<Button

android:id="@+id/btn1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginVertical="10dp"

android:layout\_weight="1"

android:text="@string/truth"

app:layout\_constraintBottom\_toBottomOf="parent"/>

<Button

android:id="@+id/btn2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginVertical="10dp"

android:layout\_weight="1"

android:text="@string/dare"

app:layout\_constraintBottom\_toBottomOf="parent"/>

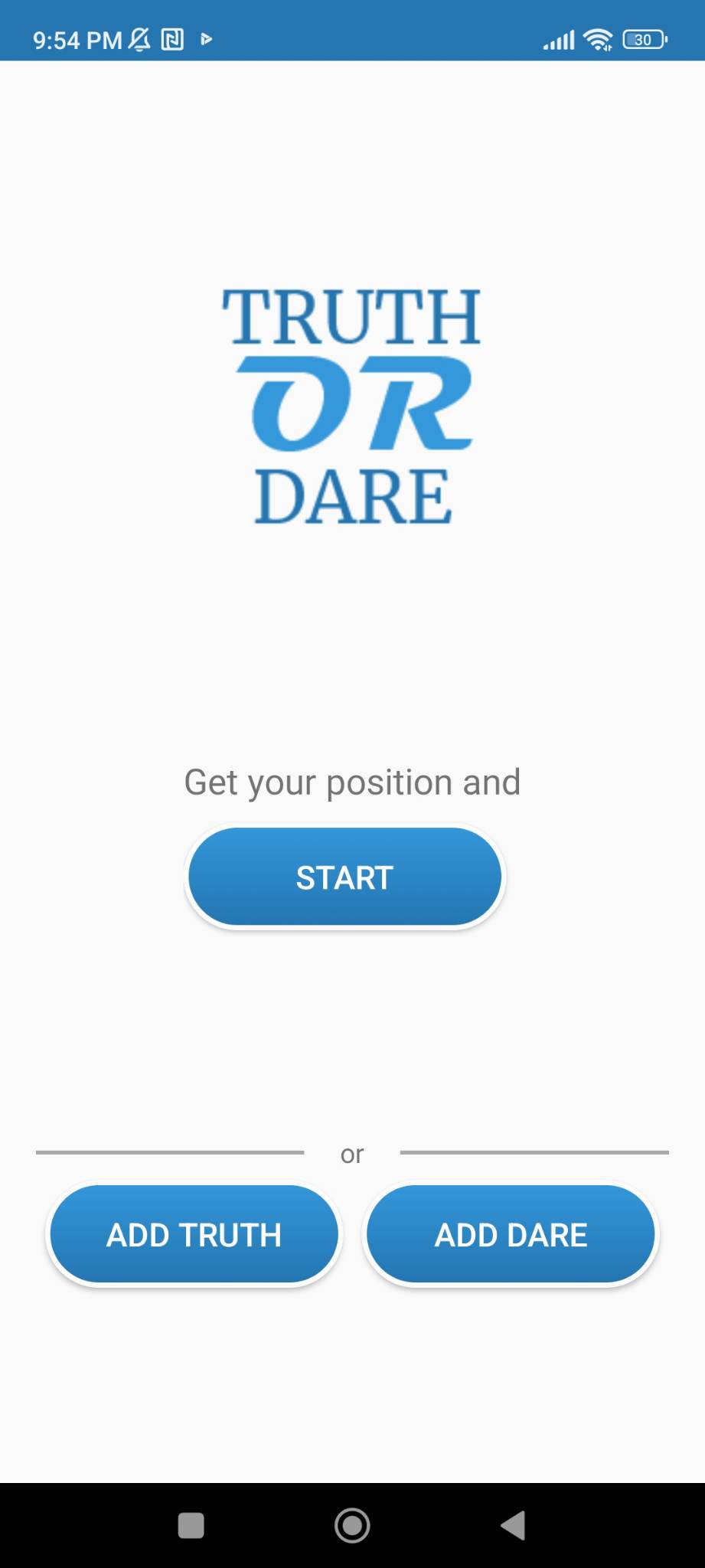
</LinearLayout>

</androidx.constraintlayout.widget.ConstraintLayout>

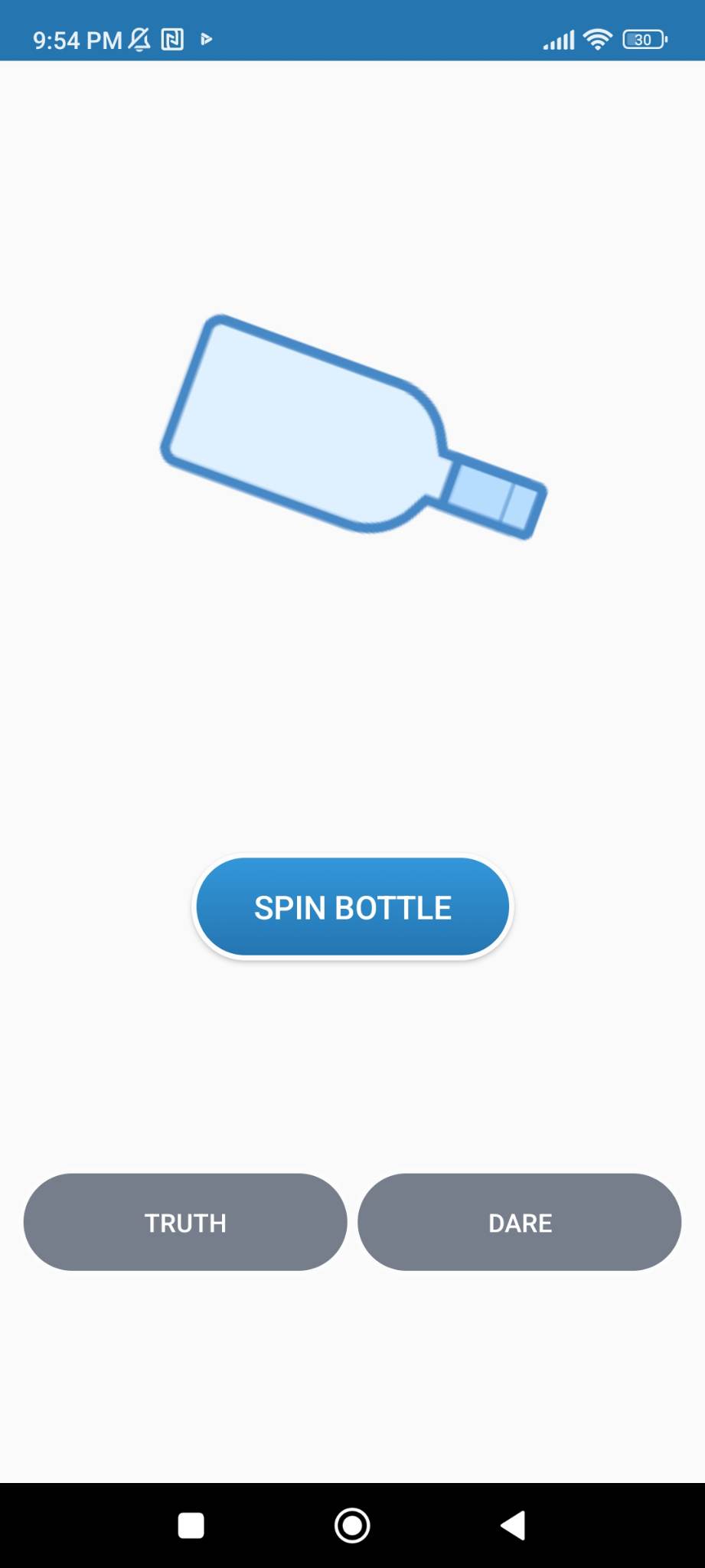
Това е XML код за дизайн на екран в Android приложение. Използва се ConstraintLayout, който е компонент за позициониране на елементи върху екрана. В дизайна има ImageView за показване на изображение, Button за стартиране на игра и два Button-a в LinearLayout за избор на "Truth" или "Dare" действия. Дизайнът е на английски език, но може да бъде преведен на български за по-лесно разбиране.

Потребителско ръководство(тестване)

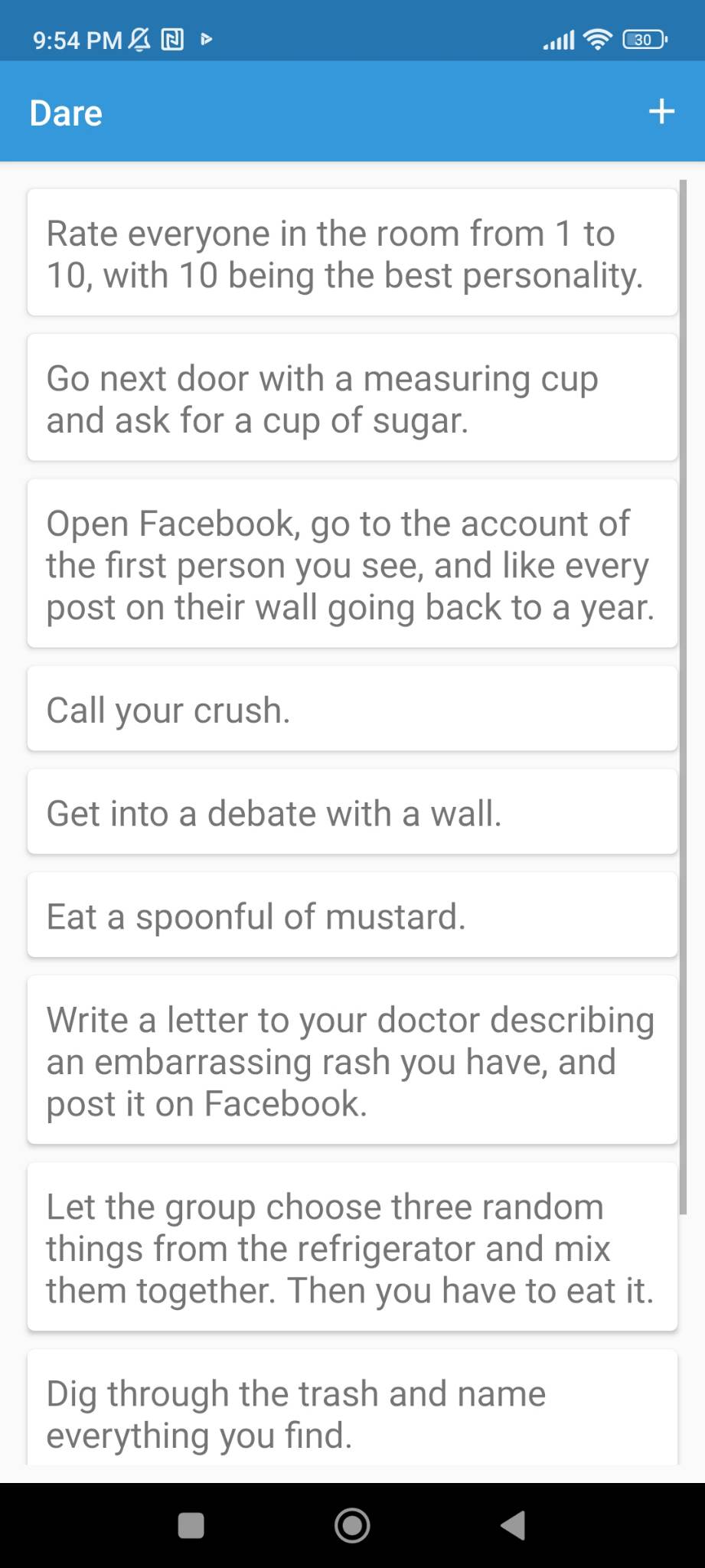
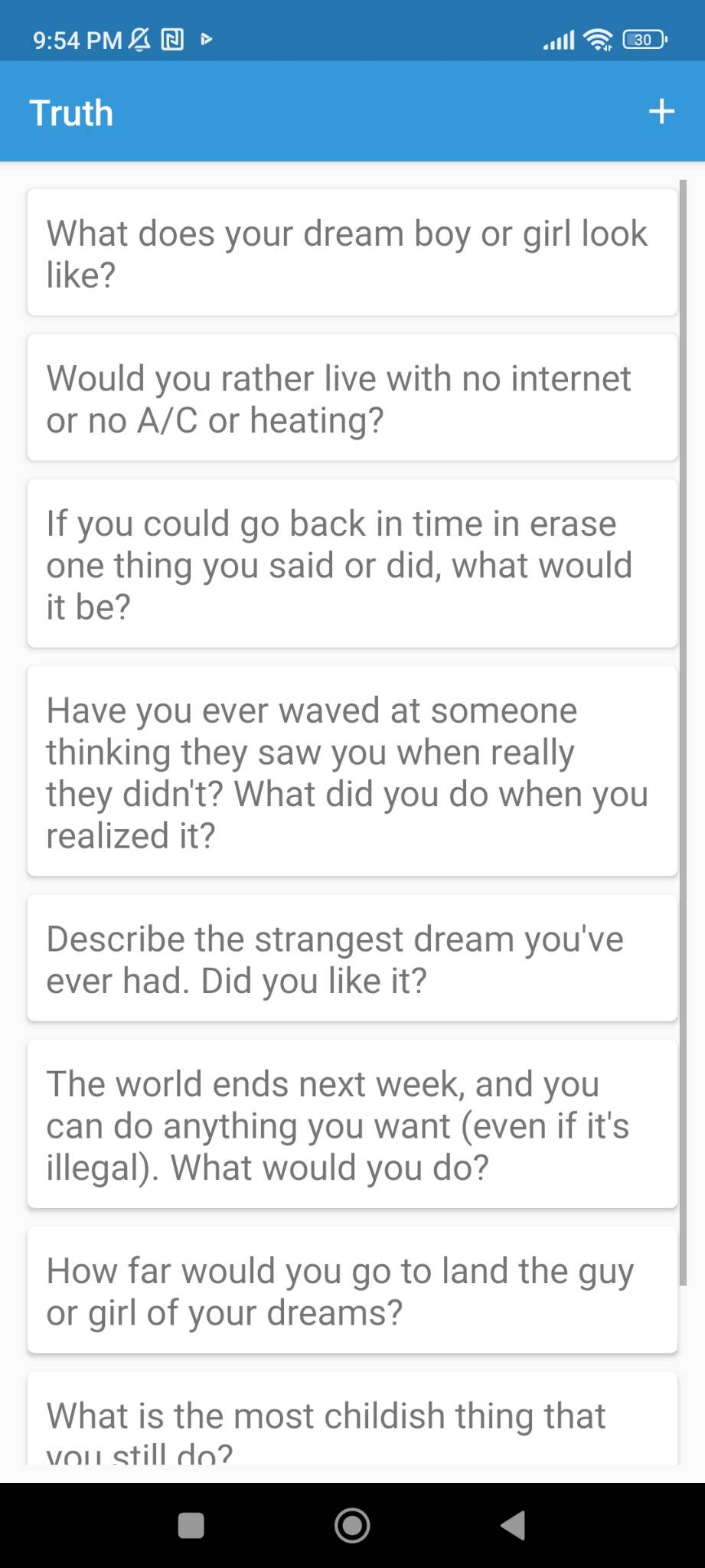
1.Това е потребителския интерфейс от “StartActivity” от тук потребителя избира и преминава или в “MainActivity” или в”TruthActivity” или в “DareActivity”.



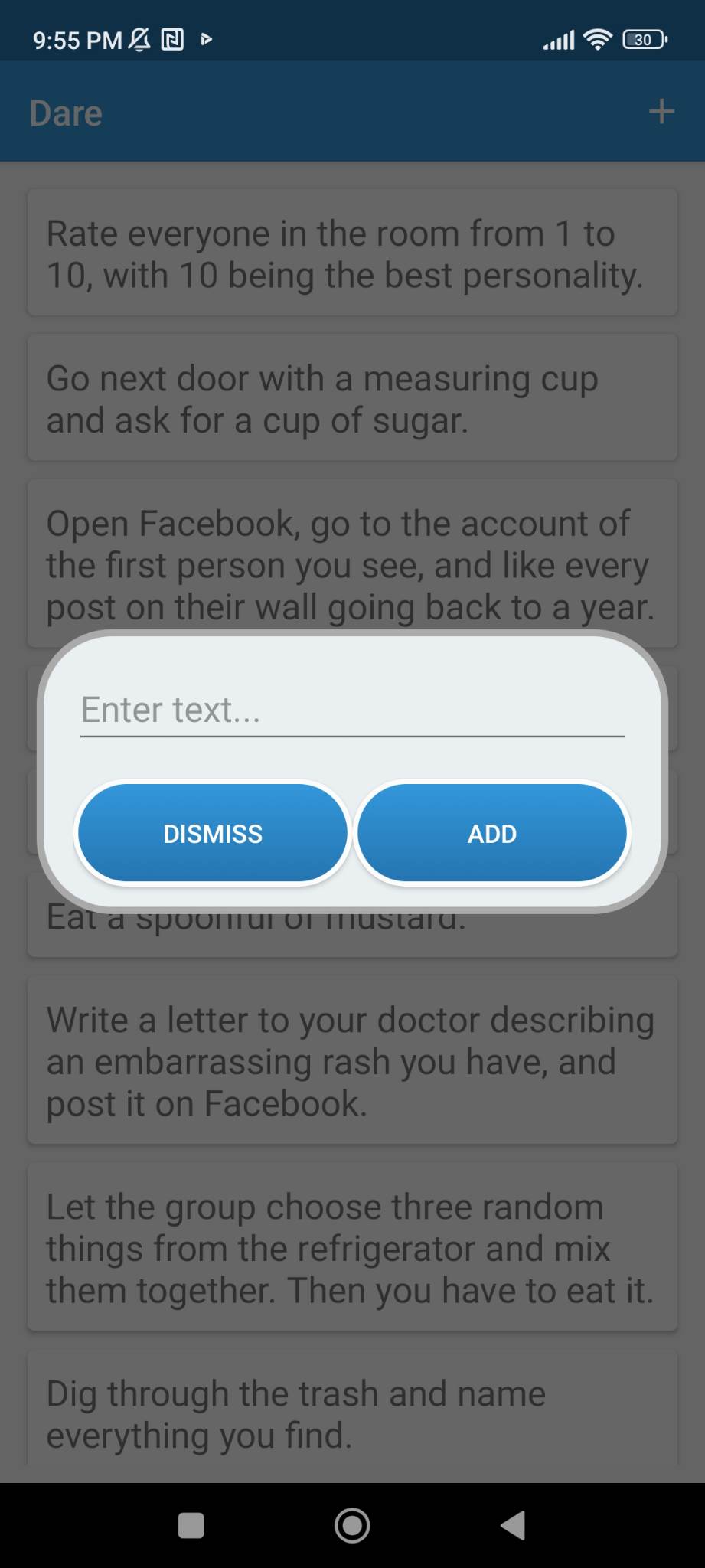
2.Това е потребителския интерфейс от “MainActivity”,потребителя тук първо я върти ботилката с бутона “Spin Bottle” и след тва отива ”TruthActivity” или в “DareActivity”.



3.Това са ”TruthActivity” и “DareActivity”.Където потребителя избира от дадените „Въпроси“ и „Предизвикателства“.



4.Това е екранът където потребителя може да додаде още въпроси или предизвикателства.



Заключение

Truth or Dare е забавна и интерактивна мобилна апликация, която предоставя възможност за игра на любимата игра на много потребители. Апликацията предоставя на потребителите широка гама от въпроси и предизвикателства, които могат да бъдат настроени и персонализирани.

Проектирането и реализацията на приложението бяха осъществени посредством използване на стандартни технологии и концепции на програмирането като класове, обекти, string масиви, if-else конструкции и други. Беше необходимо да се обърне внимание на детайлите и логиката на играта, за да се осигури добро потребителско изживяване.

Литература

1.<https://stackoverflow.com/search?q=android+studio&s=276b7beb-0ad1-43f7-bc2d-0cb247653b58>

2.<https://www.youtube.com/results?search_query=build+android+app>

3.<https://app.diagrams.net/?libs=general;er&src=about>

4. https://www.geeksforgeeks.org/android-animation-using-android-studio/