



DEPARTMENT OF INFORMATION

SCIENCE AND TECHNOLOGY

**IT5702
MOBILE COMPUTING
SEMESTER - 7**

WORD UNSCRAMBLING GAME

**DONE BY
ARVIND P
2019115021
B. Tech IT
2019 - 2023**

ABSTRACT:

In today's world, it is imperative for people to have a sharp and incisive mind to cope with everyday tasks. Often, the survivability and fate of projects at companies depend on project team members staying mentally active. In order to provide that, they need to exercise their minds by performing activities for the brain. These need not be complicated, and can be as simple as a word unscrambling game.

In a word unscrambling game, a scrambled word, typically representing a seven letter word, is presented to the player. The player has to find out the correct word as fast as they can within a specified time frame. Upon guessing the correct word, the user can click on a button to generate a new scrambled word. The application allows the user to note their highscore, which is the number of words they have guessed correctly before the timer for a scrambled word runs out.

TABLE OF CONTENTS:

1. INTRODUCTION	4
1.1. BACKGROUND	4
1.2. PROBLEM STATEMENT	4
1.3. SOLUTION OVERVIEW	5
2. DESIGN	7
2.1. REGISTRATION AND SIGN IN	7
2.2. GAMEPLAY	8
2.3. ADMINISTRATOR FEATURES	8
3. IMPLEMENTATION	9
3.1. JAVA FILES	9
3.2. XML FILES	41
3.3. SCREENSHOTS	67
4. HARDWARE AND SOFTWARE	79
4.1. GOOGLE FIREBASE	79
4.2. ANDROID STUDIO	81
5. RESULTS AND CONCLUSION	83
5.1 RESULTS	83
5.2 CONCLUSION AND FUTURE WORK	83
6. REFERENCES	84

LIST OF FIGURES

1. REGISTRATION AND SIGN IN	6
2. GAMEPLAY	7
3. ADMINISTRATOR FEATURES	7

CHAPTER 1

INTRODUCTION

1.1. BACKGROUND:

Often, people become stressed at their workplace. This would cause them to perform relatively less at their jobs. Most of such jobs require the person to maintain an active, deductive, problem-solving, and critical-thinking mindset. Stress at the workplace can impair all of these vital traits that are needed to contribute to a project.

To circumvent this risk, we need not look far for solutions. A solution of a word unscrambling game suffices in keeping one's mind active and working.

1.2. PROBLEM STATEMENT:

In order to effectively curb the potential of stress to derail a person's active mindset and keep them, a word unscrambling game is to be developed as an easy-to-use-and-play solution.

1.3. SOLUTION OVERVIEW:

The Word Unscrambling Game made with Android Studio IDE is meant to follow a simplistic but engaging playstyle.

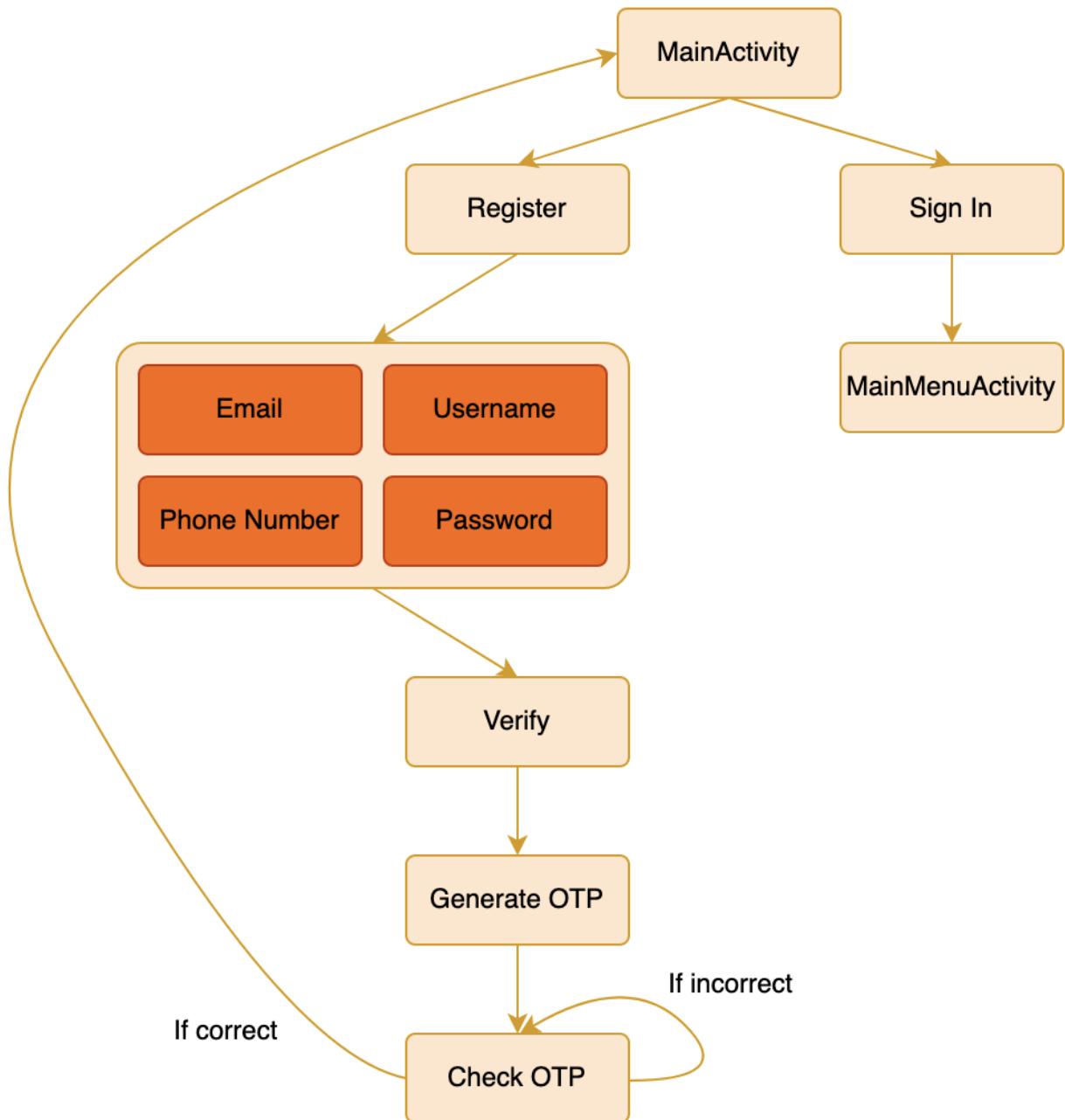
In the game, a scrambled word, typically representing a seven letter word, is presented to the player. The player has to find out the correct word as fast as they can within a specified time frame. Upon guessing the correct word, the user can click on a button to generate a new scrambled word. The application allows the user to note their highscore, which is the number of words they have guessed correctly before the timer for a scrambled word runs out.

In the game, the administrator can also remove users, and add or remove words. To adjust the difficulty level, the administrator can add words with a different number of letters, and also modify the timer accordingly.

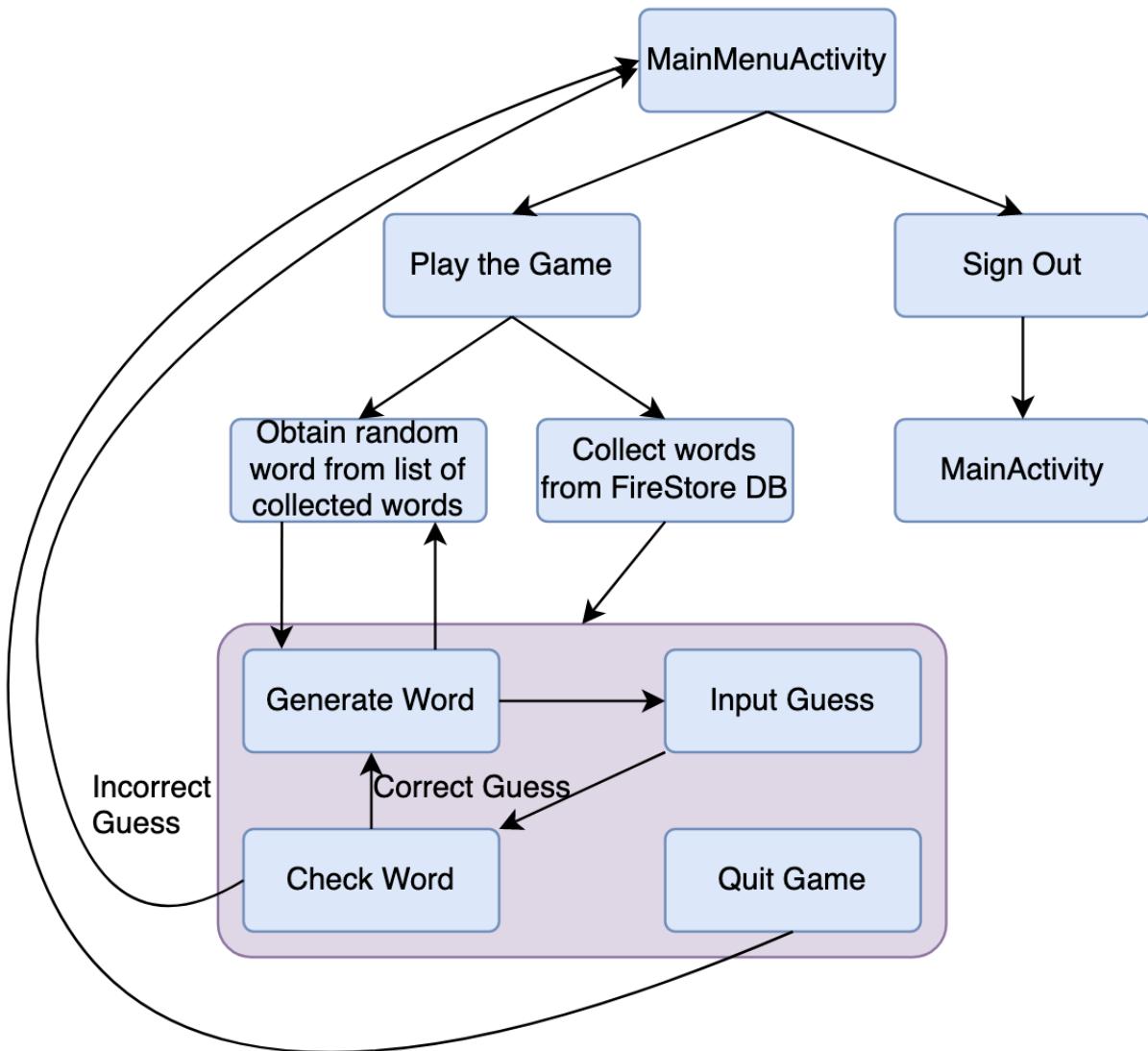
CHAPTER 2

DESIGN

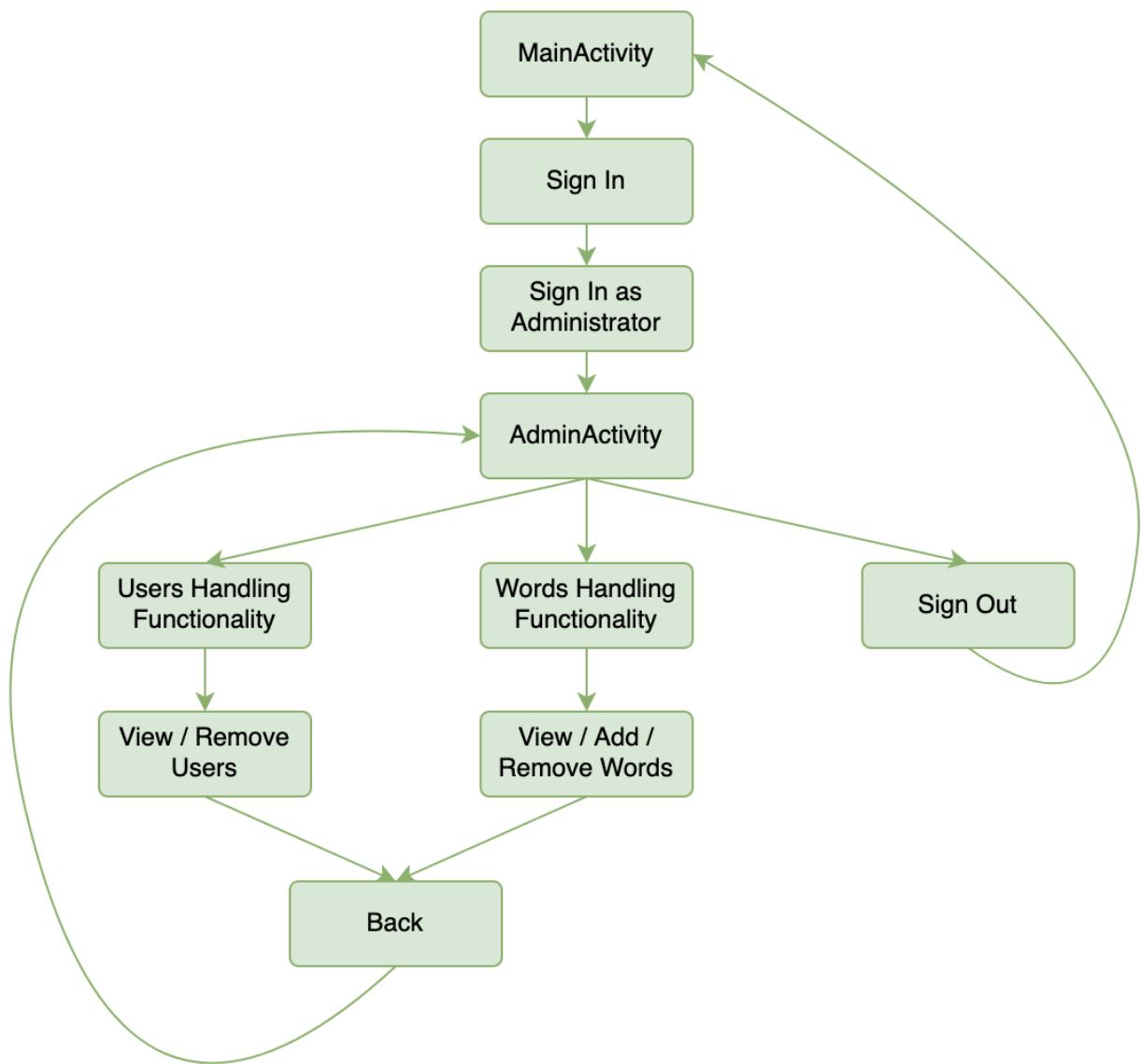
2.1. REGISTRATION AND SIGN IN:



2.2. GAMEPLAY:



2.3. ADMINISTRATOR FEATURES:



CHAPTER 3

IMPLEMENTATION

3.1. JAVA FILES:

MainActivity.java:

```
package com.arvindp.unscramblethewords;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        getWindow().setNavigationBarColor(ContextCompat.getColor(this,
R.color.cinnamon));
        setContentView(R.layout.activity_main);

        Button buttonReg = findViewById(R.id.btnRegister);
        buttonReg.setOnClickListener(view -> openRegisterActivity());

        Button buttonSignIn = findViewById(R.id.btnSignIn);
        buttonSignIn.setOnClickListener(view -> openSignInActivity());
    }

    public void openRegisterActivity() {
        Intent intent = new Intent(this, RegisterActivity.class);
        startActivity(intent);
    }

    public void openSignInActivity() {
        Intent intent = new Intent(this, SignInActivity.class);
    }
}
```

```
        startActivity(intent);
    }
}
```

MainMenuActivity.java:

```
package com.arvindp.unscramblethewords;

import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

public class MainMenuActivity extends AppCompatActivity {

    Integer score = 0;
    String Email, documentID;
    TextView scoreTextView;

    @SuppressLint("SetTextI18n")
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        getWindow().setNavigationBarColor(ContextCompat.getColor(this,
R.color.cinnamon));
        setContentView(R.layout.activity_main_menu);

        Intent intent = getIntent();
        score = intent.getIntExtra("high_score", 0);
        Email = intent.getStringExtra("current_email");
        documentID = intent.getStringExtra("documentID");

        scoreTextView = findViewById(R.id.subtext2);
        scoreTextView.setText("Highest streak: " + score);

        Button buttonPlay = findViewById(R.id.btnPlay);
        buttonPlay.setOnClickListener(view -> openPlayActivity());
    }
}
```

```

        Button buttonSignOut = findViewById(R.id.btnExit);
        buttonSignOut.setOnClickListener(view -> openMainActivity());
    }

    public void openPlayActivity()
    {
        Intent intent = new Intent(this, PlayActivity.class);
        intent.putExtra("high_score", score);
        intent.putExtra("current_email", Email);
        intent.putExtra("documentID", documentID);
        startActivity(intent);
    }

    public void openMainActivity()
    {
        Intent intent = new Intent(this, MainActivity.class);
        startActivity(intent);
    }
}

```

RegisterActivity.java:

```

package com.arvindp.unscramblethewords;

import android.content.Intent;
import android.os.Bundle;
import android.text.method.PasswordTransformationMethod;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

import com.google.android.material.button.MaterialButton;
import com.google.firebase.firestore.FirebaseFirestore;

public class RegisterActivity extends AppCompatActivity {

    EditText email, username, password, phno;
    MaterialButton registerBtn, backBtn;
    FirebaseFirestore firestore;
}

```

```

String Email, Username, Password, Phno, emailRegex, passwordRegex,
phoneNumberRegex;

@Override
protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);
    getWindow().setNavigationBarColor(ContextCompat.getColor(this,
R.color.cinnamon));
    setContentView(R.layout.activity_register);

    firestore = FirebaseFirestore.getInstance();
    email = findViewById(R.id.email);
    username = findViewById(R.id.username);
    password = findViewById(R.id.password);
    phno = findViewById(R.id.phno);

    password.setTransformationMethod(PasswordTransformationMethod.getInstance());

    emailRegex = "^(.+@[\\S]+)$";
    passwordRegex = "^(?=.*[0-9])"
        + "(?=.*[a-z])"
        + "(?=.*[A-Z])"
        + "(?=.*[$%^])"
        + "(?=\\S+.{8,20})$";
    phoneNumberRegex = "^[789]\\d{9}$";

    registerBtn = findViewById(R.id.btnRegister);
    registerBtn.setOnClickListener(view -> {

        Email = email.getText().toString();
        Username = username.getText().toString();
        Password = password.getText().toString();
        Phno = phno.getText().toString();

        if (!Email.isEmpty() && !Username.isEmpty() && !Password.isEmpty() &&
            !Phno.matches(phoneNumberRegex)) {
            if (!Email.matches(emailRegex))
                Toast.makeText(RegisterActivity.this,"Invalid email...",Toast.LENGTH_SHORT).show();
            else if (!Password.matches(passwordRegex))

```

```

        Toast.makeText(RegisterActivity.this,"Password does not meet
requirements...", Toast.LENGTH_SHORT).show();
        else if (!Phno.matches(phoneNumberRegex))
            Toast.makeText(RegisterActivity.this,"Not a valid phone number
format!", Toast.LENGTH_SHORT).show();
    }
    //else if (Email.contains("@admin."))
    //    Toast.makeText(RegisterActivity.this,"You cannot register with
that email!", Toast.LENGTH_SHORT).show();
    //}
    else {
        openVerifyAccActivity();
    }
}
else {
    Toast.makeText(RegisterActivity.this,"One or more fields is empty...", Toast.LENGTH_SHORT).show();
}
});

backBtn = findViewById(R.id.btnPrev);
backBtn.setOnClickListener(view -> openMainActivity());
}

public void openVerifyAccActivity() {
    Intent intent = new Intent(this, VerifyAccActivity.class);
    intent.putExtra("phoneNumber", Phno);
    intent.putExtra("email", Email);
    intent.putExtra("username", Username);
    intent.putExtra("password", Password);
    startActivity(intent);
}

public void openMainActivity() {
    Intent intent = new Intent(this, MainActivity.class);
    startActivity(intent);
}
}

```

VerifyAccActivity.java:

```
package com.arvindp.unscramblethewords;
```

```
import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.widget.EditText;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.material.button.MaterialButton;
import com.google.firebase.FirebaseException;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.PhoneAuthCredential;
import com.google.firebase.auth.PhoneAuthOptions;
import com.google.firebase.auth.PhoneAuthProvider;
import com.google.firebaseio.firestore.FirebaseFirestore;

import java.util.HashMap;
import java.util.Map;
import java.util.concurrent.TimeUnit;

public class VerifyAccActivity extends AppCompatActivity {

    EditText otp;
    MaterialButton btnGenerateOTP, btnVerify;
    FirebaseAuth mAuth;
    String Email, Username, Password, Phno, verificationID;
    String encryptedEmail, encryptedUsername, encryptedPassword, encryptedHighscore,
    encryptedPhno;

    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        getWindow().setNavigationBarColor(ContextCompat.getColor(this,
R.color.cinnamon));
    }
}
```

```

setContentView(R.layout.activity_verify_acc);

Intent intent = getIntent();
Phno = intent.getStringExtra("phoneNumber");
Email = intent.getStringExtra("email");
Username = intent.getStringExtra("username");
Password = intent.getStringExtra("password");

otp = findViewById(R.id.otp);
btnGenerateOTP = findViewById(R.id.btnGenerateOTP);
btnVerify = findViewById(R.id.btnVerify);
mAuth = FirebaseAuth.getInstance();

btnGenerateOTP.setOnClickListener(view -> {

    sendVerificationCode(Phno);
});

btnVerify.setEnabled(false);
btnVerify.setOnClickListener(view -> {
    if(TextUtils.isEmpty(otp.getText().toString())) {
        Toast.makeText(VerifyAccActivity.this, "Please enter the OTP...", Toast.LENGTH_SHORT).show();
    }
    else
        verifyCode(otp.getText().toString());
});
}

private void sendVerificationCode(String phoneNumber) {
    PhoneAuthOptions options = PhoneAuthOptions.newBuilder(mAuth)
        .setPhoneNumber("+91" + phoneNumber)
        .setTimeout(60L, TimeUnit.SECONDS)
        .setActivity(this)
        .setCallbacks(mCallbacks)
        .build();

    PhoneAuthProvider.verifyPhoneNumber(options);
}

private final PhoneAuthProvider.OnVerificationStateChangedCallbacks mCallbacks =
    new PhoneAuthProvider.OnVerificationStateChangedCallbacks() {

```

```

@Override
public void onVerificationCompleted(@NonNull PhoneAuthCredential
phoneAuthCredential) {

    final String code = phoneAuthCredential.getSmsCode();
    if(code != null) {
        verifyCode(code);
    }
}

@Override
public void onVerificationFailed(@NonNull FirebaseException firebaseException)
{

    Toast.makeText(VerifyAccActivity.this, "Could not verify account...", 
Toast.LENGTH_SHORT).show();
}

@Override
public void onCodeSent(@NonNull String s, @NonNull
PhoneAuthProvider.ForceResendingToken token)
{
    super.onCodeSent(s, token);
    verificationID = s;

    Toast.makeText(VerifyAccActivity.this, "An OTP has been sent to your
registered phone number.", Toast.LENGTH_SHORT).show();
    btnVerify.setEnabled(true);
}

private void verifyCode(String Code) {

    PhoneAuthCredential credential =
PhoneAuthProvider.getCredential(verificationID, Code);
    signInByCredentials(credential);
}

private void signInByCredentials(PhoneAuthCredential phoneAuthCredential) {

```

```

FirebaseFirestore firestore = FirebaseFirestore.getInstance();

FirebaseAuth firebaseAuth = FirebaseAuth.getInstance();
firebaseAuth.signInWithCredential(phoneAuthCredential)
    .addOnCompleteListener((OnCompleteListener<AuthResult>) task -> {

        if(task.isSuccessful()) {
            Toast.makeText(VerifyAccActivity.this, "Your account has been
verified!", Toast.LENGTH_SHORT).show();

            try {
                encryptedEmail = AES.encrypt>Email;
                encryptedUsername = AES.encrypt$Username;
                encryptedPassword = AES.encrypt$Password;
                encryptedHighscore = AES.encrypt"0";
                encryptedPhno = AES.encrypt$Password;

            } catch (Exception exception) {
                exception.printStackTrace();
            }

            Map<String, Object> user = new HashMap<>();
            user.put("email", encryptedEmail);
            user.put("username", encryptedUsername);
            user.put("password", encryptedPassword);
            user.put("highscore", encryptedHighscore);
            user.put("phone number", encryptedPhno);

            firestore.collection("users")
                .add(user)
                .addOnSuccessListener(docRef ->
Toast.makeText(VerifyAccActivity.this, "Registration Successful!",
Toast.LENGTH_SHORT).show())
                    .addOnFailureListener(exception -> {
                        Toast.makeText(VerifyAccActivity.this,
"Registration Failed...", Toast.LENGTH_SHORT).show();
                        exception.printStackTrace();
                    });
            }
        else {
            Toast.makeText(VerifyAccActivity.this, "OTP provided is
incorrect...", Toast.LENGTH_SHORT).show();
        }
    });
}

```

```

        }

    });

    openMainActivity();
}

public void openMainActivity() {
    Intent intent = new Intent(this, MainActivity.class);
    startActivity(intent);
}
}

```

SignInActivity.java:

```

package com.arvindp.unscramblethewords;

import android.content.Intent;
import android.os.Bundle;
import android.text.method.PasswordTransformationMethod;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

import com.google.android.material.button.MaterialButton;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.QuerySnapshot;

import java.util.ArrayList;
import java.util.List;
import java.util.Objects;

public class SignInActivity extends AppCompatActivity {

    EditText email, password;
    MaterialButton signInBtn, prevBtn;
    FirebaseFirestore firestore;
    Integer score = 0;
    String encryptedEmail, decryptedPassword, decryptedUsername;
    String Email, emailRegex, Password, documentID;
}

```

```

List documentsList;

@Override
protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);
    getWindow().setNavigationBarColor(ContextCompat.getColor(this,
R.color.cinnamon));
    setContentView(R.layout.activity_sign_in);

    firestore = FirebaseFirestore.getInstance();
    email = findViewById(R.id.email);
    password = findViewById(R.id.password);
    password.setTransformationMethod(PasswordTransformationMethod.getInstance());

    emailRegex = "^(.+)@(.+)$";

    signInBtn = findViewById(R.id.btnSignIn);
    signInBtn.setOnClickListener(view -> {

        Email = email.getText().toString();
        Password = password.getText().toString();

        if (!Email.isEmpty() && !Password.isEmpty()) {
            if (!Email.matches(emailRegex)) {
                if (!Email.matches(emailRegex))
                    Toast.makeText(SignInActivity.this, "Invalid email...", Toast.LENGTH_SHORT).show();
            } else {

                encryptedEmail = AES.encrypt(Email);
                firestore.collection("users")
                    .whereEqualTo("email", encryptedEmail)
                    .get()
                    .addOnCompleteListener(task -> {
                        if (task.isSuccessful()) {

                            QuerySnapshot querySnapshot = task.getResult();
                            documentsList = new ArrayList<>();
                            documentsList = querySnapshot.getDocuments();

                            if (!documentsList.isEmpty()) {

```

```

        DocumentSnapshot document = (DocumentSnapshot)
documentsList.get(0);

        documentID = document.getId();
        decryptedPassword =
AES.decrypt(document.getString("password"));

        assert decryptedPassword != null;
        if (decryptedPassword.equals(Password))
        {
            decryptedUsername =
AES.decrypt(document.getString("username"));
            if (Email.contains("@admin.com")) {

                Toast.makeText(SignInActivity.this,
"Credentials verified! Welcome Admin " + decryptedUsername + "!",
Toast.LENGTH_SHORT).show();
                openAdminActivity();

            } else {

                Toast.makeText(SignInActivity.this,
"Credentials verified! Welcome " + decryptedUsername + "!",
Toast.LENGTH_SHORT).show();
                score =
Integer.parseInt(Objects.requireNonNull(AES.decrypt(document.getString("highscore"))));
;

                openMainMenuActivity();
            }
        } else
            Toast.makeText(SignInActivity.this,
"Incorrect password...", Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(SignInActivity.this, "Could not
identify account with email: " + Email, Toast.LENGTH_SHORT).show();
        }
    } else {
        Toast.makeText(SignInActivity.this, "Sign in failed
with: " + task.getException(), Toast.LENGTH_SHORT).show();
    }
}
}

```

```

        }
        else {
            Toast.makeText(SignInActivity.this,"One or more fields is empty...",  

Toast.LENGTH_SHORT).show();
        }
    });

prevBtn = findViewById(R.id.btnPrev);
prevBtn.setOnClickListener(view -> openMainActivity());
}

public void openMainActivity() {
    Intent intent = new Intent(this, MainActivity.class);
    startActivity(intent);
}

public void openAdminActivity() {
    Intent intent = new Intent(this, AdminActivity.class);
    startActivity(intent);
}

public void openMainMenuActivity() {
    Intent intent = new Intent(this, MainMenuActivity.class);
    intent.putExtra("high_score", score);
    intent.putExtra("current_email", Email);
    intent.putExtra("documentID", documentID);
    startActivity(intent);
}
}
}

```

PlayActivity.java:

```

package com.arvindp.unscramblethewords;

import android.content.Intent;
import android.os.Bundle;
import android.os.CountDownTimer;
import android.util.Log;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

```

```
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

import com.google.android.material.button.MaterialButton;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.QuerySnapshot;

import java.util.ArrayList;
import java.util.Collections;
import java.util.LinkedList;
import java.util.List;
import java.util.Locale;
import java.util.Random;

public class PlayActivity extends AppCompatActivity {

    private long mTimeLeftInMillis = 36000;
    private CountDownTimer mCountDownTimer;
    Integer highscore = 0, score = 0;
    String Email, ans, encryptedHighscore, documentID;
    DocumentSnapshot document;
    List documentsList;
    MaterialButton genWordBtn, answerCheckBtn, exitBtn;
    TextView timerTextView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        getWindow().setNavigationBarColor(ContextCompat.getColor(this,
R.color.cinnamon));
        setContentView(R.layout.activity_play);

        Intent intent = getIntent();
        highscore = intent.getIntExtra("high_score", 0);
        Email = intent.getStringExtra("current_email");
        documentID = intent.getStringExtra("documentID");

        FirebaseFirestore firestore = FirebaseFirestore.getInstance();
        EditText answer = findViewById(R.id.answer);
        timerTextView = findViewById(R.id.timer);
    }
}
```

```

exitBtn = findViewById(R.id.btnExit);
exitBtn.setEnabled(false);

firestore.collection("word")
    .get()
    .addOnCompleteListener(task -> {
        if (task.isSuccessful()) {

            QuerySnapshot querySnapshot = task.getResult();

            documentsList = new ArrayList<>();

            documentsList = querySnapshot.getDocuments();
            TextView scrambledWord = findViewById(R.id.scrambledword);
            answerCheckBtn = findViewById(R.id.btnAnswerCheck);
            answerCheckBtn.setEnabled(false);

            genWordBtn = findViewById(R.id.btnGenWord);
            genWordBtn.setOnClickListener(view -> {

                exitBtn.setEnabled(true);
                answerCheckBtn.setEnabled(true);
                genWordBtn.setEnabled(false);

                Random rand = new Random();
                if (!documentsList.isEmpty()) {

                    int index = rand.nextInt(documentsList.size());
                    document = (DocumentSnapshot) documentsList.get(index);
                    ans = AES.decrypt(document.getString("word"));
                    assert ans != null;

                    LinkedList<String> list = new LinkedList<>();
                    String[] chars = ans.split("");
                    Collections.addAll(list, chars);

                    // To ensure the scrambled word is not the same as the
actual word
                    LinkedList<String> temp = new LinkedList<>(list);
                    while (temp.equals(list)) {

```

```

        Collections.shuffle(list);
    }

    StringBuilder scrWord = new StringBuilder();
    for (int i=0; i<list.size(); i++) {
        scrWord.append(list.get(i));
    }

    scrambledWord.setText(scrWord.toString());

    mCountDownTimer = new CountDownTimer(mTimeLeftInMillis,
1000) {

    @Override
    public void onTick(long millisUntilFinished) {
        mTimeLeftInMillis = millisUntilFinished;

        int minutes = (int) (mTimeLeftInMillis / 1000)
/ 60;
        int seconds = (int) (mTimeLeftInMillis / 1000)
% 60;

        String timeLeftFormatted =
String.format(Locale.getDefault(), "%02d:%02d", minutes, seconds);
        timerTextView.setText(timeLeftFormatted);
    }

    @Override
    public void onFinish() {
        Toast.makeText(PlayActivity.this, "Time's up!
The word was \"" + ans + "\". Final number of words unscrambled: " + score,
Toast.LENGTH_SHORT).show();

        if (score > highscore) {
            highscore = score;
            encryptedHighscore =
AES.encrypt(String.valueOf(highscore));

            firestore.collection("users")
                .document(documentID)
                .update("highscore",
encryptedHighscore);
    }
}

```

```

        openMainMenuActivity();
    }
}
}.start();
} else {
    Toast.makeText(PlayActivity.this, "Could not fetch a
word...", Toast.LENGTH_SHORT).show();
}
});

answerCheckBtn.setOnClickListener(view -> {

    String text = answer.getText().toString();

    if (!text.isEmpty()) {

        mCountDownTimer.cancel();
        answerCheckBtn.setEnabled(false);
        genWordBtn.setEnabled(true);

        answer.setText("");

        if (!text.equals(ans)) {
            Toast.makeText(PlayActivity.this, "That was not the
right word! The word was \"" + ans + "\". Final number of words unscrambled: " +
score, Toast.LENGTH_SHORT).show();

            if (score > highscore) {
                highscore = score;

                encryptedHighscore =
AES.encrypt(highscore.toString());
            }

            firestore.collection("users")
                .document(documentID)
                .update("highscore",
encryptedHighscore);
        }
        openMainMenuActivity();
    }
} else {
    mTimeLeftInMillis = 36000;
}
}
});

```

```

                score += 1;
                Toast.makeText(PlayActivity.this, "Correct word! "
+ score + " word(s) unscrambled", Toast.LENGTH_SHORT).show();
            }
        }
        else {
            Toast.makeText(PlayActivity.this, "Make a guess
first!", Toast.LENGTH_SHORT).show();
        }
    });

    exitBtn.setOnClickListener(view -> {
        mCountDownTimer.cancel();
        openMainActivity();
    });
}
)).addOnFailureListener(exception -> Log.d("this",
exception.getMessage())));
}

public void openMainActivity() {
    Intent intent = new Intent(this, MainActivity.class);
    intent.putExtra("high_score", highscore);
    intent.putExtra("current_email", Email);
    intent.putExtra("documentID", documentID);
    startActivity(intent);
}
}

```

AdminActivity.java:

```

package com.arvindp.unscramblethewords;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;

public class AdminActivity extends AppCompatActivity {

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);
    getWindow().setNavigationBarColor(ContextCompat.getColor(this,
R.color.cinnamon));
    setContentView(R.layout.activity_admin);

    Button buttonUsersHandle = findViewById(R.id.btnUsersHandle);
    buttonUsersHandle.setOnClickListener(view -> openUsersHandleActivity());

    Button buttonWordsHandle = findViewById(R.id.btnWordsHandle);
    buttonWordsHandle.setOnClickListener(view -> openWordsHandleActivity());

    Button buttonExit = findViewById(R.id.btnExit);
    buttonExit.setOnClickListener(view -> openMainActivity());
}

public void openUsersHandleActivity() {
    Intent intent = new Intent(this, UsersHandleActivity.class);
    startActivity(intent);
}

public void openWordsHandleActivity() {
    Intent intent = new Intent(this, WordsHandleActivity.class);
    startActivity(intent);
}

public void openMainActivity() {
    Intent intent = new Intent(this, MainActivity.class);
    startActivity(intent);
}
}

```

UserHandleActivity.java:

```

package com.arvindp.unscramblethewords;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Toast;

```

```
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.google.android.material.button.MaterialButton;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.QuerySnapshot;

import java.util.ArrayList;
import java.util.List;
import java.util.Objects;

public class UsersHandleActivity extends AppCompatActivity {

    private ArrayList<ExampleItem> mExampleList;
    List documentsList = new ArrayList<>();
    FirebaseFirestore firestore = FirebaseFirestore.getInstance();
    private ExampleAdapter mAdapter;
    MaterialButton backBtn;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        getWindow().setNavigationBarColor(ContextCompat.getColor(this,
R.color.cinnamon));
        setContentView(R.layout.activity_users_handle);

        mExampleList = new ArrayList<>();

        firestore.collection("users")
            .get()
            .addOnCompleteListener(task -> {
                if (task.isSuccessful()) {

                    QuerySnapshot querySnapshot = task.getResult();
                    documentsList = querySnapshot.getDocuments();

                    for (int i=0; i<documentsList.size(); i++) {

```

```

        DocumentSnapshot documentSnapshot = (DocumentSnapshot)
documentsList.get(i);

        String username =
AES.decrypt(Objects.requireNonNull(documentSnapshot.getString("username")));

        String email =
AES.decrypt(Objects.requireNonNull(documentSnapshot.getString("email")));

        assert email != null;
        if (!email.contains("@admin."))
            mExampleList.add(new
ExampleItem(R.drawable.ic_baseline_person, username, email));
    }

    buildRecyclerView();
}
});

backBtn = findViewById(R.id.btnExit);
backBtn.setOnClickListener(view -> openAdminActivity());
}

public void openAdminActivity() {
    Intent intent = new Intent(this, AdminActivity.class);
    startActivity(intent);
}

public void removeItem(int position) {

    firestore.collection("users")

.document(Objects.requireNonNull(AES.encrypt(mExampleList.get(position).getText2())))

.delete()
.addOnSuccessListener(unused ->
Toast.makeText(UsersHandleActivity.this,"Deletion
successful!",Toast.LENGTH_SHORT).show()
.addOnFailureListener(e -> {
    Toast.makeText(UsersHandleActivity.this,"Deletion
unsuccessful...",Toast.LENGTH_SHORT).show();
    e.printStackTrace();
}));
}

mExampleList.remove(position);

```

```

        mAdapter.notifyItemRemoved(position);
    }

    public void buildRecyclerView() {
        RecyclerView mRecyclerView = findViewById(R.id.recyclerView);
        mRecyclerView.setHasFixedSize(true);
        RecyclerView.LayoutManager mLayoutManager = new LinearLayoutManager(this);
        mAdapter = new ExampleAdapter(mExampleList);

        mRecyclerView.setLayoutManager(mLayoutManager);
        mRecyclerView.setAdapter(mAdapter);

        mAdapter.setOnItemClickListener(this::removeItem);
    }
}

```

WordsHandleActivity.java:

```

package com.arvindp.unscramblethewords;

import android.content.Intent;
import android.os.Bundle;
import android.widget.EditText;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.content.ContextCompat;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.android.material.button.MaterialButton;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.QuerySnapshot;

import java.util.ArrayList;
import java.util.HashMap;

```

```

import java.util.List;
import java.util.Map;
import java.util.Objects;

public class WordsHandleActivity extends AppCompatActivity {

    private ArrayList<ExampleItem2> mExampleList;
    List documentsList = new ArrayList<>(), documentsList2;
    FirebaseFirestore firestore = FirebaseFirestore.getInstance();
    private ExampleAdapter2 mAdapter;
    MaterialButton backBtn, addBtn;
    EditText text;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        getWindow().setNavigationBarColor(ContextCompat.getColor(this,
R.color.cinnamon));
        setContentView(R.layout.activity_words_handle);

        mExampleList = new ArrayList<>();

        firestore.collection("word")
            .get()
            .addOnCompleteListener(task -> {
                if (task.isSuccessful()) {

                    QuerySnapshot querySnapshot = task.getResult();
                    documentsList = querySnapshot.getDocuments();

                    for (int i=0; i<documentsList.size(); i++) {

                        DocumentSnapshot d = (DocumentSnapshot)
documentsList.get(i);
                        String word =
AES.decrypt(Objects.requireNonNull(d.getString("word")));
                        mExampleList.add(new ExampleItem2(word));
                    }
                }

                buildRecyclerView();
            }
    }
}

```

```

});;

text = findViewById(R.id.newWord);
addBtn = findViewById(R.id.btnAdd);
addBtn.setOnClickListener(view -> {
    String newWord = text.getText().toString();
    text.setText("");

    if (!newWord.matches("[a-z]{7}")) {
        Toast.makeText(WordsHandleActivity.this, "Invalid word syntax! (Must have 7 letters | Must be lowercase)", Toast.LENGTH_SHORT).show();
    }
    else
    {
        String encryptedNewWord = AES.encrypt(newWord);

        firestore.collection("word")
            .whereEqualTo("word", encryptedNewWord)
            .get()
            .addOnCompleteListener(task -> {

                QuerySnapshot querySnapshot = task.getResult();
                documentsList2 = new ArrayList<>();
                documentsList2 = querySnapshot.getDocuments();

                if (documentsList2.size() != 0) {
                    Toast.makeText(WordsHandleActivity.this, "\"" + newWord + "\" already exists!", Toast.LENGTH_SHORT).show();
                }
                else {
                    Map<String, Object> word = new HashMap<>();
                    word.put("word", encryptedNewWord);

                    firestore.collection("word")
                        .add(word)
                        .addOnSuccessListener(docRef ->
                            Toast.makeText(WordsHandleActivity.this, newWord + " has been added successfully!", Toast.LENGTH_SHORT).show())
                        .addOnFailureListener(exception -> {
                            Toast.makeText(WordsHandleActivity.this, "\"" + newWord + "\" could not be added...", Toast.LENGTH_SHORT).show();
                            exception.printStackTrace();
                        });
                }
            });
    }
}

```

```

        } ) ;

        mExampleList.add( new ExampleItem2( newWord ) );
        buildRecyclerView();
    }
} );
}

backBtn = findViewById(R.id.btnExit);
backBtn.setOnClickListener( view -> openAdminActivity() );
}

public void openAdminActivity() {
    Intent intent = new Intent(this, AdminActivity.class);
    startActivity(intent);
}

public void removeItem(int position) {

    if (mExampleList.size() > 2) {
        firestore.collection("word")
            .whereEqualTo("word",
AES.encrypt(mExampleList.get(position).getText()))
            .get()
            .addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
                @Override
                public void onComplete(@NonNull Task<QuerySnapshot> task) {

                    if (task.isSuccessful() && !task.getResult().isEmpty()) {

                        DocumentSnapshot documentSnapshot =
task.getResult().getDocuments().get(0);
                        String documentID = documentSnapshot.getId();

                        firestore.collection("word")
                            .document(documentID)
                            .delete()
                            .addOnSuccessListener(unused ->
Toast.makeText(WordsHandleActivity.this, "Deletion successful!",
Toast.LENGTH_SHORT).show())
                            .addOnFailureListener(exception -> {
}

```

```

        Toast.makeText(WordHandleActivity.this,
"Deletion unsuccessful...", Toast.LENGTH_SHORT).show();
                exception.printStackTrace();
            });
        } else {
            Toast.makeText(WordHandleActivity.this, "Deletion
unsuccessful...", Toast.LENGTH_SHORT).show();
        }
    }
}

mExampleList.remove(position);
mAdapter.notifyItemRemoved(position);
}
else {
    Toast.makeText(WordHandleActivity.this, "Cannot delete when there are two
words left!", Toast.LENGTH_SHORT).show();
}

}

public void buildRecyclerView() {
    RecyclerView m RecyclerView = findViewById(R.id.recyclerView);
    m RecyclerView.setHasFixedSize(true);
    RecyclerView.LayoutManager m LayoutManager = new LinearLayoutManager(this);
    mAdapter = new ExampleAdapter2(mExampleList);

    m RecyclerView.setLayoutManager(m LayoutManager);
    m RecyclerView.setAdapter(m Adapter);

    mAdapter.setOnItemClickListener(this::removeItem);
}
}
}

```

AES.java:

```

package com.arvindp.unscramblethewords;

import java.nio.charset.StandardCharsets;
import java.security.spec.KeySpec;
import java.util.Base64;
import javax.crypto.Cipher;

```

```

import javax.crypto.SecretKey;
import javax.crypto.SecretKeyFactory;
import javax.crypto.spec.IvParameterSpec;
import javax.crypto.spec.PBEKeySpec;
import javax.crypto.spec.SecretKeySpec;

class AES {
    private static final String SECRET_KEY = "word-unscrambling-game-secret-key";

    private static final String SALT = "salt-for-word-unscrambling-game";

    public static String encrypt(String strToEncrypt) {

        try {
            byte[] iv = { 0, 0, 0, 0, 0, 0, 0, 0,
                         0, 0, 0, 0, 0, 0, 0, 0 };

            IvParameterSpec ivspec = new IvParameterSpec(iv);

            SecretKeyFactory factory =
SecretKeyFactory.getInstance("PBKDF2WithHmacSHA256");

            KeySpec spec = new PBEKeySpec(SECRET_KEY.toCharArray(), SALT.getBytes(),
65536, 256);
            SecretKey tmp = factory.generateSecret(spec);
            SecretKeySpec secretKey = new SecretKeySpec(tmp.getEncoded(), "AES");

            Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
            cipher.init(Cipher.ENCRYPT_MODE, secretKey, ivspec);

            return
Base64.getEncoder().encodeToString(cipher.doFinal(strToEncrypt.getBytes(StandardCharsets.UTF_8)));
        }

        catch (Exception exception) {
            exception.printStackTrace();
        }

        return null;
    }

    public static String decrypt(String strToDecrypt) {

```

```

try {
    byte[] iv = { 0, 0, 0, 0, 0, 0, 0, 0,
                  0, 0, 0, 0, 0, 0, 0, 0 };

    IvParameterSpec ivspec = new IvParameterSpec(iv);

    SecretKeyFactory factory =
SecretKeyFactory.getInstance("PBKDF2WithHmacSHA256");

    KeySpec spec = new PBEKeySpec(SECRET_KEY.toCharArray(), SALT.getBytes(),
65536, 256);
    SecretKey tmp = factory.generateSecret(spec);
    SecretKeySpec secretKey = new SecretKeySpec(tmp.getEncoded(), "AES");

    Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5PADDING");
    cipher.init(Cipher.DECRYPT_MODE, secretKey, ivspec);

    return new
String(cipher.doFinal(Base64.getDecoder().decode(strToDecrypt)));
}

catch (Exception exception) {
    exception.printStackTrace();
}

return null;
}
}

```

ExampleAdapter.java:

```

package com.arvindp.unscramblethewords;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;

```

```
import androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;

public class ExampleAdapter extends RecyclerView.Adapter<ExampleAdapter.ExampleViewHolder> {

    private ArrayList<ExampleItem> mExampleList;
    private OnItemClickListener mListener;

    public interface OnItemClickListener {
        void onDeleteClick(int position);
    }

    public void setOnItemClickListener(OnItemClickListener listener) {
        mListener = listener;
    }

    public static class ExampleViewHolder extends RecyclerView.ViewHolder {

        public ImageView mImageView;
        public TextView mTextView1;
        public TextView mTextView2;
        public ImageView mDeleteImage;

        public ExampleViewHolder(View itemView, final OnItemClickListener listener) {

            super(itemView);
            mImageView = itemView.findViewById(R.id.imageView);
            mTextView1 = itemView.findViewById(R.id.textView);
            mTextView2 = itemView.findViewById(R.id.textView2);
            mDeleteImage = itemView.findViewById(R.id.image_delete);

            mDeleteImage.setOnClickListener(view -> {
                if (listener != null) {
                    int position = getAdapterPosition();
                    if (position != RecyclerView.NO_POSITION) {
                        listener.onDeleteClick(position);
                    }
                }
            });
        }
    }
}
```

```

}

public ExampleAdapter(ArrayList<ExampleItem> exampleList) {
    mExampleList = exampleList;
}

@NonNull
@Override
public ExampleViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
    View view =
LayoutInflator.from(parent.getContext()).inflate(R.layout.example_item, parent,
false);
    return new ExampleViewHolder(view, mListener);
}

@Override
public void onBindViewHolder(@NonNull ExampleViewHolder holder, int position) {

    ExampleItem currentItem = mExampleList.get(position);
    holder.mImageView.setImageResource(currentItem.getImageResource());
    holder.mTextView1.setText(currentItem.getText1());
    holder.mTextView2.setText(currentItem.getText2());
}

@Override
public int getItemCount() {
    return mExampleList.size();
}
}

```

ExampleAdapter2.java:

```

package com.arvindp.unscramblethewords;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

```

```
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;

public class ExampleAdapter2 extends
RecyclerView.Adapter<ExampleAdapter2.ExampleViewHolder> {

    private ArrayList<ExampleItem2> mExampleList;
    private ExampleAdapter2.OnItemClickListener mListener;

    public void setOnItemClickListener(ExampleAdapter2.OnItemClickListener listener) {
        mListener = listener;
    }

    public interface OnItemClickListener {
        void onDeleteClick(int position);
    }

    public static class ExampleViewHolder extends RecyclerView.ViewHolder {

        public TextView mTextView;
        public ImageView mDeleteImage;

        public ExampleViewHolder(View itemView, final
ExampleAdapter2.OnItemClickListener listener) {

            super(itemView);
            mTextView = itemView.findViewById(R.id.textView);
            mDeleteImage = itemView.findViewById(R.id.image_delete);

            mDeleteImage.setOnClickListener(view -> {
                if (listener != null) {
                    int position = getAdapterPosition();
                    if (position != RecyclerView.NO_POSITION) {
                        listener.onDeleteClick(position);
                    }
                }
            });
        }
    }
}
```

```

public ExampleAdapter2(ArrayList<ExampleItem2> exampleList) {
    mExampleList = exampleList;
}

@NonNull
@Override
public ExampleAdapter2.ExampleViewHolder onCreateViewHolder(@NonNull ViewGroup
parent, int viewType) {
    View v =
LayoutInflater.from(parent.getContext()).inflate(R.layout.example_item_2, parent,
false);
    ExampleAdapter2.ExampleViewHolder evh = new
ExampleAdapter2.ExampleViewHolder(v, mListener);
    return evh;
}

@Override
public void onBindViewHolder(@NonNull ExampleAdapter2.ExampleViewHolder holder, int
position) {

    ExampleItem2 currentItem = mExampleList.get(position);
    holder.mTextView.setText(currentItem.getText());
}

@Override
public int getItemCount() {
    return mExampleList.size();
}
}

```

ExampleItem.java:

```

package com.arvindp.unscramblethewords;

public class ExampleItem {
    private int mImageResource;
    private String mText1;
    private String mText2;
}

```

```
public ExampleItem(int imageResource, String text1, String text2) {
    mImageResource = imageResource;
    mText1 = text1;
    mText2 = text2;
}

public int getImageResource() {
    return mImageResource;
}

public String getText1() {
    return mText1;
}

public String getText2() {
    return mText2;
}

}
```

ExampleItem2.java:

```
package com.arvindp.unscramblethewords;

public class ExampleItem2 {

    private String mText;

    public ExampleItem2(String text) {
        mText = text;
    }

    public String getText() {
        return mText;
    }
}
```

3.2. XML FILES:

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg"
    android:padding="10dp"
    tools:context=".MainActivity">

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="25dp"
        app:cardCornerRadius="10dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        app:cardElevation="15dp"
        >

        <TextView
            android:id="@+id/maintext"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="29dp"
            android:fontFamily="cursive"
            android:gravity="center_vertical|center_horizontal"
            android:text="@string/unscramble_the_words"

            android:textColor="#000000"
            android:textSize="35sp"
            android:textStyle="bold" />

    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
```

```

        android:id="@+id/cardView2"
        android:layout_below="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="350dp"
        android:layout_marginBottom="25dp"
        app:cardCornerRadius="15dp"
        app:cardElevation="15dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
    >

    <com.google.android.material.button.MaterialButton
        android:id="@+id	btnRegister"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginStart="46.5dp"
        android:layout_marginTop="50dp"
        android:paddingBottom="10dp"
        android:text="Register" />

    <com.google.android.material.button.MaterialButton
        android:id="@+id	btnSignIn"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:layout_below="@+id	btnRegister"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="135dp"
        android:layout_marginStart="46.5dp"
        android:paddingBottom="10dp"
        android:text="@string/sign_in" />

</androidx.cardview.widget.CardView>

</RelativeLayout>

```

activity_main_menu.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

```

```
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:padding="10dp"
android:background="@drawable/bg"
tools:context=".MainMenuActivity">

<androidx.cardview.widget.CardView
    android:id="@+id/cardView"
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="25dp"
    app:cardCornerRadius="10dp"
    app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
    app:cardElevation="15dp"
    >

<TextView
    android:id="@+id/maintext"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="29dp"
    android:fontFamily="cursive"
    android:gravity="center_vertical|center_horizontal"
    android:textColor="#000000"
    android:textSize="35sp"
    android:textStyle="bold"
    android:text="@string/unscramble_the_words"/>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView2"
    android:layout_width="match_parent"
    android:layout_height="300dp"
    android:layout_below="@+id/cardView"
    android:layout_marginStart="25dp"
    android:layout_marginTop="50dp"
```

```
        android:layout_marginEnd="25dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        app:cardCornerRadius="15dp"
        app:cardElevation="15dp">

    <TextView
        android:id="@+id/subtext"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="45dp"
        android:fontFamily="cursive"
        android:gravity="center"

        android:text="@string/words_are_scrambled_except_for_the_first_letter_nguess_the_correct_words_within_30_seconds_each"
        android:textColor="@color/black"
        android:textSize="25sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/subtext2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/subtext"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="195dp"
        android:fontFamily="cursive"
        android:gravity="center"
        android:text=""
        android:textColor="@color/black"
        android:textSize="25sp"
        android:textStyle="bold">

    </TextView>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView3"
    android:layout_below="@+id/cardView2"
    android:layout_width="match_parent"
```

```

        android:layout_height="match_parent"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="50dp"
        android:layout_marginBottom="25dp"
        app:cardCornerRadius="10dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        app:cardElevation="15dp"
    >

<com.google.android.material.button.MaterialButton
    android:id="@+id	btnPlay"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="45dp"
    android:layout_marginEnd="45dp"
    android:layout_marginTop="25dp"
    android:layout_centerHorizontal="true"
    android:paddingBottom="10dp"
    android:text="@string/play_the_game" />

<com.google.android.material.button.MaterialButton
    android:id="@+id	btnExit"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_below="@+id	btnPlay"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="45dp"
    android:layout_marginTop="90dp"
    android:paddingBottom="10dp"
    android:text="@string/sign_out" />

</androidx.cardview.widget.CardView>

</RelativeLayout>

```

activity_register.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"

```

```
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:padding="10dp"
        android:background="@drawable/bg"
        tools:context=".RegisterActivity">

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="25dp"
        app:cardCornerRadius="10dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        app:cardElevation="15dp"
    >

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="29dp"
        android:fontFamily="cursive"
        android:gravity="center_vertical|center_horizontal"
        android:textColor="@color/black"
        android:textSize="35sp"
        android:textStyle="bold"
        android:text="@string/register_now"/>

    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView2"
        android:layout_below="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="25dp"
        android:layout_marginBottom="25dp"
        app:cardCornerRadius="15dp"
```

```
    app:cardElevation="15dp"
    app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
>

<com.google.android.material.textfield.TextInputLayout

    android:id="@+id/emaillayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="15dp"
    android:hint="@string/email">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text"
        android:maxLines="1" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout

    android:id="@+id usernamelayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="95dp"
    android:hint="@string/username">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text"
        android:maxLines="1" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
```

```
        android:id="@+id/passwordlayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="175dp"
        android:hint="@string/password">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text"
        android:maxLines="1" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout

    android:id="@+id/phnolayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="305dp"
    android:hint="@string/phone_number">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/phno"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text"
        android:maxLines="1" />
</com.google.android.material.textfield.TextInputLayout>

<TextView
    android:id="@+id/passwordInfo"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="240dp"
    android:gravity="center_vertical|center_horizontal">
```

```

        android:textColor="@color/black"
        android:textSize="15sp"
        android:textStyle="bold"
        android:text="Password must have:\n Atleast 1 letter, 1 digit, 1 symbol
from ($,%,=,>), and be of 8-20 characters."/>

```

```

<com.google.android.material.button.MaterialButton
    android:id="@+id/btnRegister"
    android:layout_below="@+id/passwordlayout"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="45dp"
    android:layout_marginEnd="45dp"
    android:layout_marginTop="375dp"
    android:layout_centerHorizontal="true"
    android:paddingBottom="10dp"
    android:text="Register" />

```

```

<com.google.android.material.button.MaterialButton
    android:id="@+id/btnPrev"
    android:layout_below="@+id/btnRegister"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="45dp"
    android:layout_marginEnd="45dp"
    android:layout_marginTop="440dp"
    android:layout_centerHorizontal="true"
    android:paddingBottom="10dp"
    android:text="@string/back" />

```

```

</androidx.cardview.widget.CardView>

```

```

</RelativeLayout>

```

activity_verify_acc.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

```

```
        android:background="@drawable/bg"
        android:padding="10dp"
        tools:context=".MainActivity">

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="25dp"
        app:cardCornerRadius="10dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        app:cardElevation="15dp"
    >

    <TextView
        android:id="@+id/maintext"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="29dp"
        android:fontFamily="cursive"
        android:gravity="center_vertical|center_horizontal"
        android:text="Verify your account"

        android:textColor="#000000"
        android:textSize="35sp"
        android:textStyle="bold" />

    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView2"
        android:layout_below="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="265dp"
        android:layout_marginBottom="25dp"
        app:cardCornerRadius="15dp"
```

```
    app:cardElevation="15dp"
    app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
>

<com.google.android.material.textfield.TextInputLayout

    android:id="@+id/emaillayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="25dp"
    android:hint="@string/enter_otp">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/otp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text"
        android:maxLines="1" />

</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.button.MaterialButton
    android:id="@+id/btnGenerateOTP"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_below="@+id/btnRegister"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="125dp"
    android:layout_marginStart="46.5dp"
    android:paddingBottom="10dp"
    android:text="@string/generate_otp" />

<com.google.android.material.button.MaterialButton
    android:id="@+id/btnVerify"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_below="@+id/btnRegister"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="200dp"
    android:layout_marginStart="46.5dp"
```

```
        android:paddingBottom="10dp"
        android:text="@string/verify" />

    </androidx.cardview.widget.CardView>

</RelativeLayout>
```

activity_sign_in.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="10dp"
    android:background="@drawable/bg"
    tools:context=".SignInActivity">

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="25dp"
        app:cardCornerRadius="10dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        app:cardElevation="15dp"
        >

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="29dp"
            android:fontFamily="cursive"
            android:gravity="center_vertical|center_horizontal"
            android:textColor="@color/black"
            android:textSize="35sp"
            android:textStyle="bold"
            android:text="@string/log_in"/>
```

```
</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView2"
    android:layout_below="@+id/cardView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="200dp"
    android:layout_marginBottom="25dp"
    app:cardCornerRadius="15dp"
    app:cardElevation="15dp"
    app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
    >

<com.google.android.material.textfield.TextInputLayout

    android:id="@+id/emaillayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="25dp"
    android:hint="@string/email">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text"
        android:maxLines="1" />
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout

    android:id="@+id/passwordlayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
```

```
        android:layout_marginTop="105dp"
        android:hint="@string/password">

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text"
        android:maxLines="1" />
</com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.button.MaterialButton
        android:id="@+id/btnSignIn"
        android:layout_below="@+id/passwordlayout"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="45dp"
        android:layout_marginEnd="45dp"
        android:layout_marginTop="200dp"
        android:layout_centerHorizontal="true"
        android:paddingBottom="10dp"
        android:text="Sign in" />

    <com.google.android.material.button.MaterialButton
        android:id="@+id/btnPrev"
        android:layout_below="@+id/btnSignIn"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="45dp"
        android:layout_marginEnd="45dp"
        android:layout_marginTop="265dp"
        android:layout_centerHorizontal="true"
        android:paddingBottom="10dp"
        android:text="@string/back" />
</androidx.cardview.widget.CardView>
</RelativeLayout>
```

activity_admin.xml:

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg"
    android:padding="10dp"
    tools:context=".PlayActivity">

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="150dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="25dp"
        app:cardCornerRadius="10dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        app:cardElevation="15dp"
        >

    <TextView
        android:id="@+id/admin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="32dp"
        android:fontFamily="cursive"
        android:gravity="center_vertical|center_horizontal"
        android:textColor="#000000"
        android:textSize="35sp"
        android:textStyle="bold"
        android:text="@string/unscramble_the_words_nadmin_page"/>

    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView2"
        android:layout_below="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginStart="25dp"
```

```
        android:layout_marginEnd="25dp"
        android:layout_marginTop="200dp"
        android:layout_marginBottom="25dp"
        app:cardCornerRadius="15dp"
        app:cardElevation="15dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
    >

<com.google.android.material.button.MaterialButton
    android:id="@+id/btnUsersHandle"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="46.5dp"
    android:layout_marginTop="50dp"
    android:paddingBottom="10dp"
    android:text="@string/users_handle" />

<com.google.android.material.button.MaterialButton
    android:id="@+id/btnWordsHandle"
    android:layout_below="@+id/btnUsersHandle"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="46.5dp"
    android:layout_marginTop="125dp"
    android:paddingBottom="10dp"
    android:text="@string/words_handle" />

<com.google.android.material.button.MaterialButton
    android:id="@+id/btnExit"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="46.5dp"
    android:layout_marginTop="200dp"
    android:paddingBottom="10dp"
    android:text="Sign Out" />

</androidx.cardview.widget.CardView>

</RelativeLayout>
```

activity_play.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg"
    android:padding="10dp"
    tools:context=".PlayActivity">

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="25dp"
        app:cardCornerRadius="10dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        app:cardElevation="15dp"
        >

        <TextView
            android:id="@+id/maintext"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="29dp"
            android:fontFamily="cursive"
            android:gravity="center_vertical|center_horizontal"
            android:textColor="#000000"
            android:textSize="35sp"
            android:textStyle="bold"
            android:text="@string/unscramble_the_word"/>

    </androidx.cardview.widget.CardView>

    <androidx.cardview.widget.CardView
        android:id="@+id/cardView2"
```

```
        android:layout_below="@+id/cardView"
        android:layout_width="match_parent"
        android:layout_height="250dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="50dp"
        app:cardCornerRadius="10dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        app:cardElevation="15dp"
    >

    <com.google.android.material.button.MaterialButton
        android:id="@+id/btnGenWord"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="46.5dp"
        android:layout_marginEnd="25dp"
        android:layout_marginTop="25dp"
        android:layout_centerHorizontal="true"
        android:paddingBottom="10dp"
        android:text="@string/generate_scrambled_word" />

    <TextView
        android:id="@+id/timer"
        android:layout_below="@+id/btnGenWord"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="100dp"
        android:fontFamily="cursive"
        android:gravity="center_vertical|center_horizontal"
        android:textColor="#000000"
        android:textSize="35sp"
        android:textStyle="bold"
        android:text="" />

    <TextView
        android:id="@+id/scrambledword"
        android:layout_below="@+id/timer"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
```

```
        android:layout_marginTop="160dp"
        android:gravity="center_vertical|center_horizontal"
        android:textColor="#000000"
        android:textSize="35sp"
        android:textStyle="bold"
        android:text=""/>

    </androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView3"
    android:layout_below="@+id/cardView2"
    android:layout_width="match_parent"
    android:layout_height="210dp"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="50dp"
    app:cardCornerRadius="10dp"
    app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
    app:cardElevation="15dp"
    >

    <com.google.android.material.textfield.TextInputLayout

        android:id="@+id/answerlayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/scrambledword"
        android:layout_marginTop="20dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        android:hint="@string/what_is_the_word">

        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/answer"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="text"
            android:maxLines="1" />
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.button.MaterialButton
```

```

        android:id="@+id/btnAnswerCheck"
        android:layout_below="@+id/answerlayout"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="90dp"
        android:layout_marginStart="46.5dp"
        android:layout_centerHorizontal="true"
        android:paddingBottom="10dp"
        android:text="@string/check_your_answer" />

<com.google.android.material.button.MaterialButton
    android:id="@+id.btnExit"
    android:layout_below="@+id/btnAnswerCheck"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="145dp"
    android:layout_marginStart="46.5dp"
    android:layout_centerHorizontal="true"
    android:paddingBottom="10dp"
    android:text="@string/quit_game" />

</androidx.cardview.widget.CardView>

</RelativeLayout>

```

activity_users_handle.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ebe5d8"
    tools:context=".UsersHandleActivity">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginBottom="150dp"

```

```

        android:background="@drawable/bg"
        android:padding="4dp"
        android:scrollbars="vertical" />

<androidx.cardview.widget.CardView
    android:id="@+id/cardView"
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="624.5dp"
    app:cardCornerRadius="10dp"
    app:cardElevation="15dp"
    app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
    >

<com.google.android.material.button.MaterialButton
    android:id="@+id.btnExit"
    android:layout_width="250dp"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="46.5dp"
    android:layout_marginTop="25dp"
    android:paddingBottom="10dp"
    android:text="Back" />

</androidx.cardview.widget.CardView>

</RelativeLayout>

```

activity_words_handle.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ebe5d8"
    tools:context=".UsersHandleActivity">

```

```
<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/recyclerView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="4dp"
    android:scrollbars="vertical"
    android:background="@drawable/bg"
    android:layout_marginBottom="150dp"/>

<androidx.cardview.widget.CardView
    android:id="@+id/cardView"
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:layout_marginStart="25dp"
    android:layout_marginEnd="25dp"
    android:layout_marginTop="624.5dp"
    app:cardCornerRadius="10dp"
    app:cardElevation="15dp"
    app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
    >

<EditText
    android:id="@+id/newWord"
    android:layout_width="100dp"
    android:layout_height="40dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentBottom="true"
    android:gravity="center"
    android:layout_centerInParent="true"
    android:layout_marginStart="10dp"
    android:layout_marginTop="30dp"
    android:background="#99000000"
    android:textCursorDrawable="@null"
    android:textColor="#ffffff"
    />

<com.google.android.material.button.MaterialButton
    android:id="@+id/btnAdd"
    android:layout_width="115dp"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
```

```

        android:layout_centerVertical="true"
        android:layout_marginStart="120dp"
        android:layout_marginTop="25dp"
        android:paddingBottom="10dp"
        android:text="Add word" />

    <com.google.android.material.button.MaterialButton
        android:id="@+id/btnExit"
        android:layout_width="105dp"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="245dp"
        android:layout_marginTop="25dp"
        android:paddingBottom="10dp"
        android:text="Back" />

    </androidx.cardview.widget.CardView>

</RelativeLayout>

```

example_item.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="4dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    app:cardCornerRadius="4dp"
    app:cardElevation="15dp"
    app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
    >

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="4dp">

        <ImageView

```

```
        android:id="@+id/imageView"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:padding="2dp"
    />

<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Line 1"
    android:textSize="20sp"
    android:textStyle="bold"
    android:textColor="@android:color/black"
    android:fontFamily="cursive"
    android:layout_marginStart="25dp"
    android:layout_alignParentTop="true"
    android:layout_toEndOf="@+id/imageView"
/>

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Line 2"
    android:textSize="20sp"
    android:textColor="@android:color/black"
    android:fontFamily="cursive"
    android:layout_marginStart="25dp"
    android:layout_below="@+id/textView"
    android:layout_toEndOf="@+id/imageView"
/>

<ImageView
    android:id="@+id/image_delete"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:src="@drawable/ic_baseline_delete"
    android:layout_alignParentTop="true"
    android:layout_alignParentEnd="true"
/>
</RelativeLayout>
```

```
</androidx.cardview.widget.CardView>
```

example_item_2.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="4dp"
        android:layout_marginStart="25dp"
        android:layout_marginEnd="25dp"
        app:cardBackgroundColor="@color/material_on_primary_emphasis_high_type"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        app:cardCornerRadius="4dp"
        app:cardElevation="15dp"
    >

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="4dp">

        <TextView
            android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Line"
            android:textSize="35sp"
            android:textStyle="bold"
            android:textColor="@android:color/black"
            android:layout_alignParentTop="true"
            android:layout_marginStart="25dp"
        />

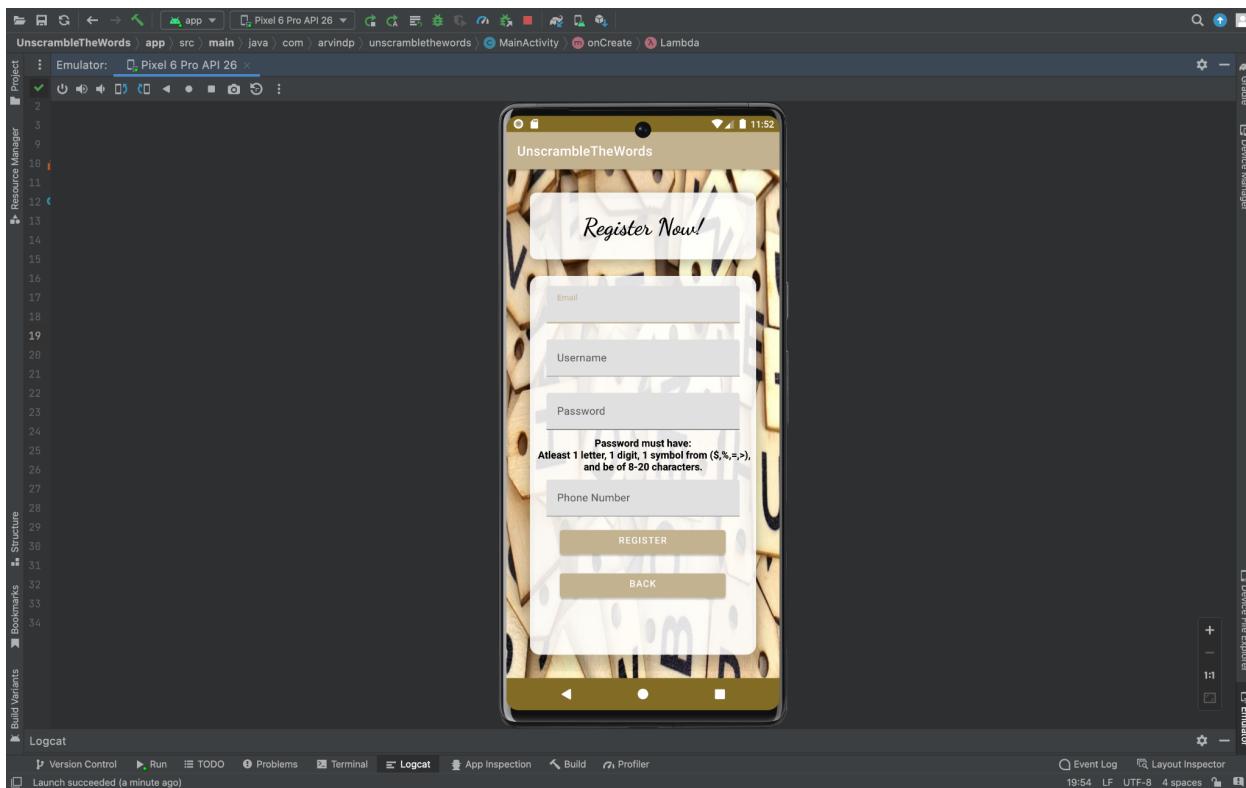
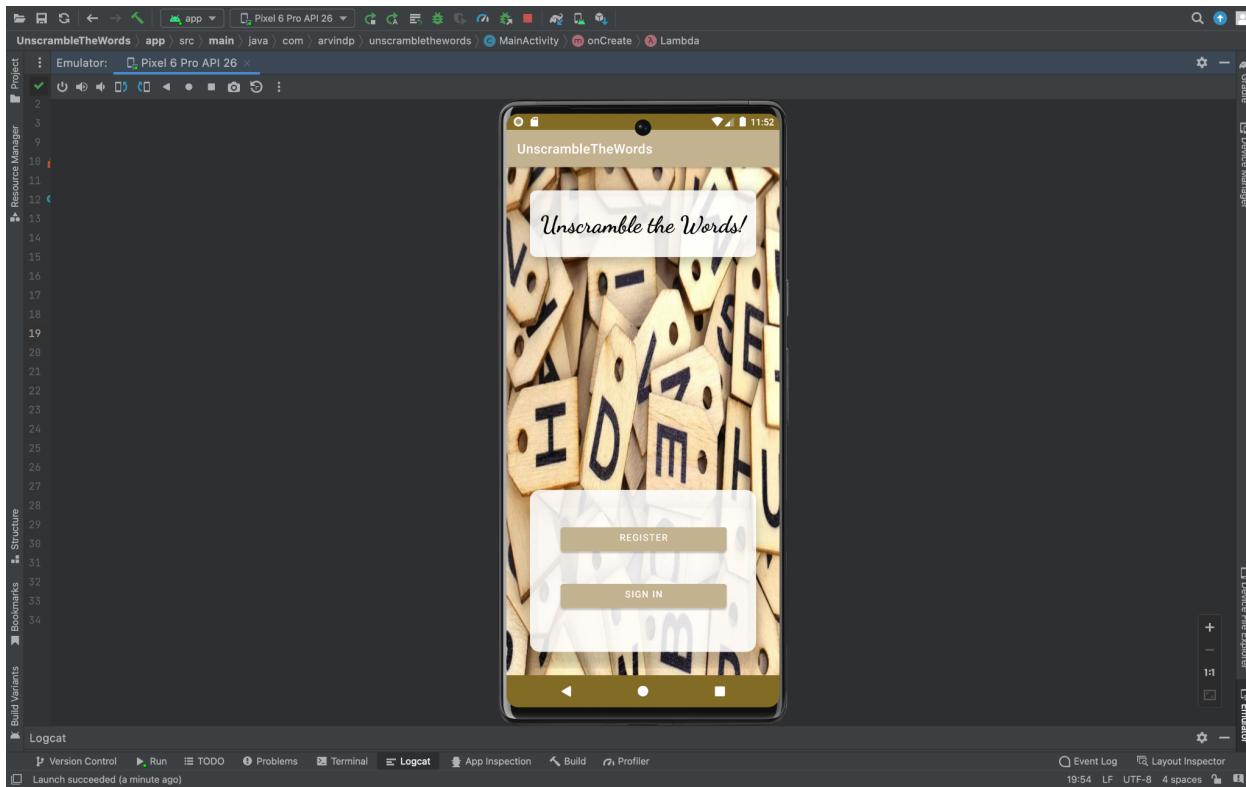
        <ImageView
            android:id="@+id/image_delete"
            android:layout_width="50dp"
            android:layout_height="50dp"
            android:src="@drawable/ic_baseline_delete"
            android:layout_alignParentTop="true"
        />
    
```

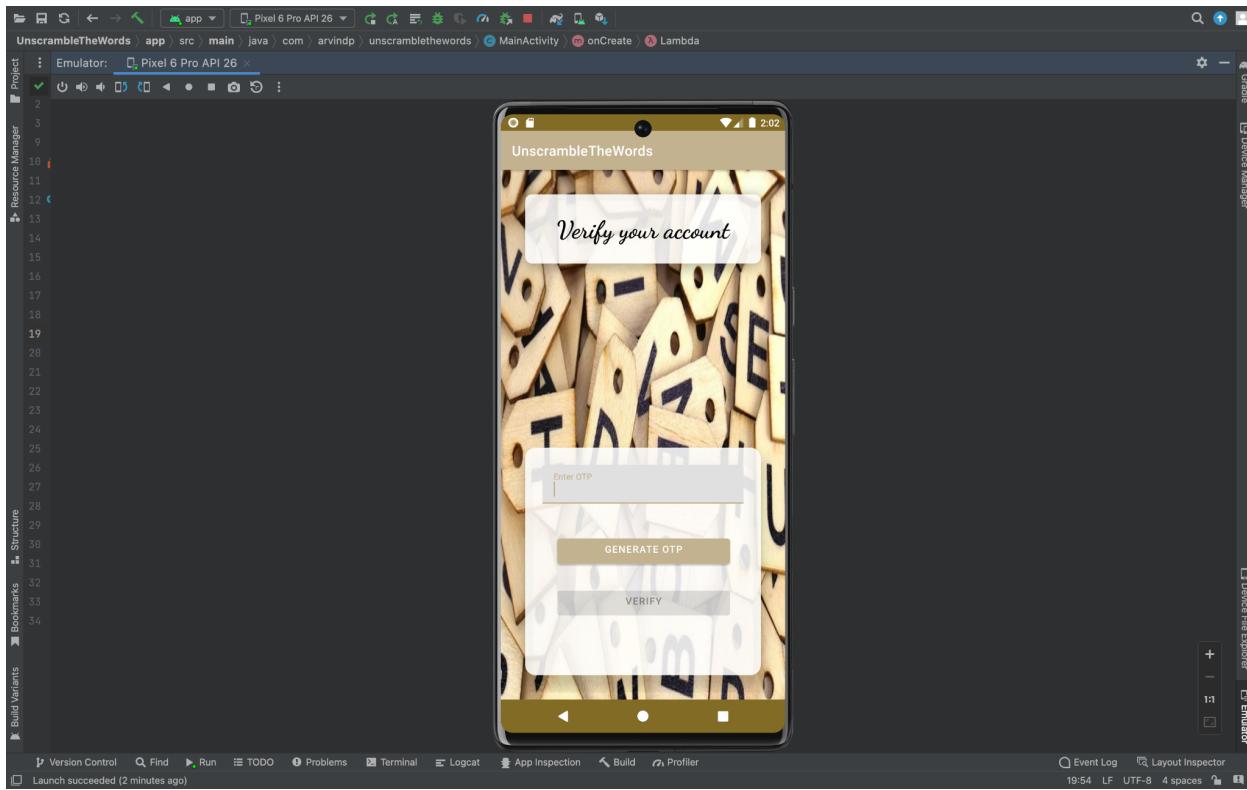
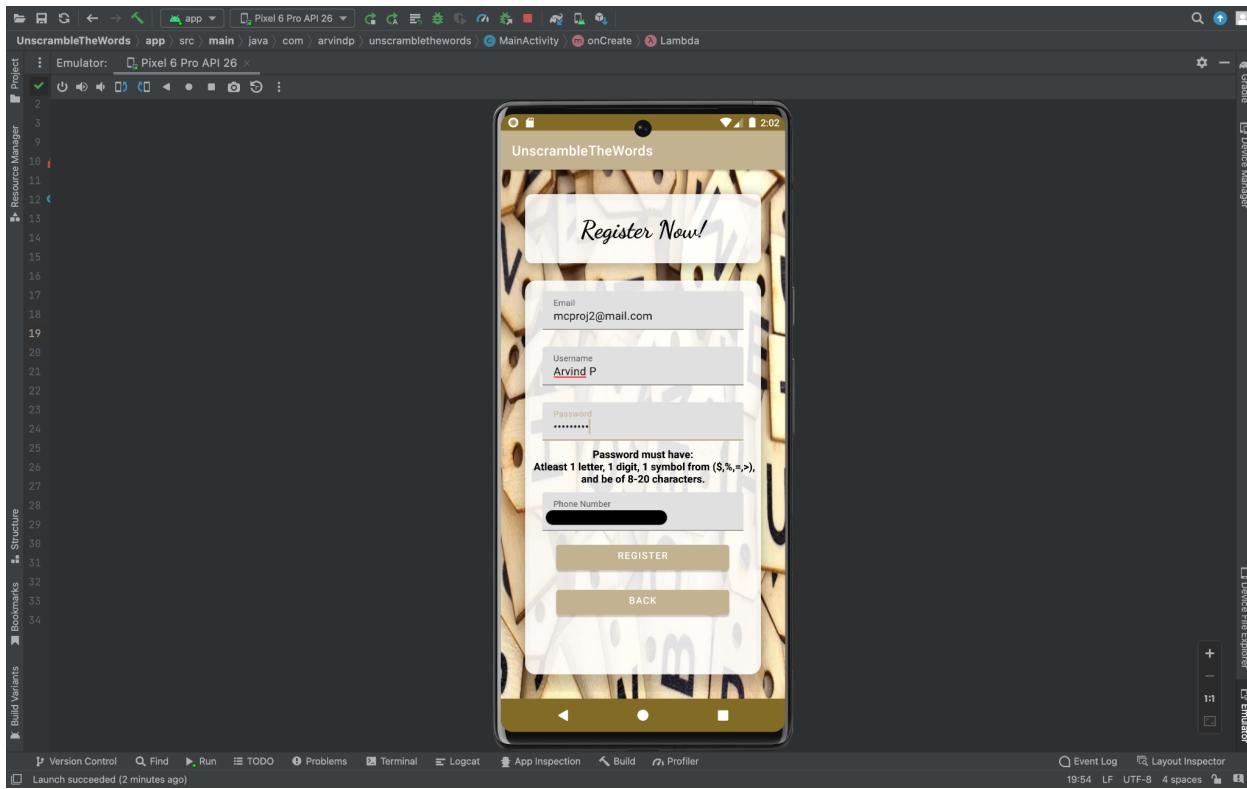
```
        android:layout_alignParentEnd="true"
    />
</RelativeLayout>

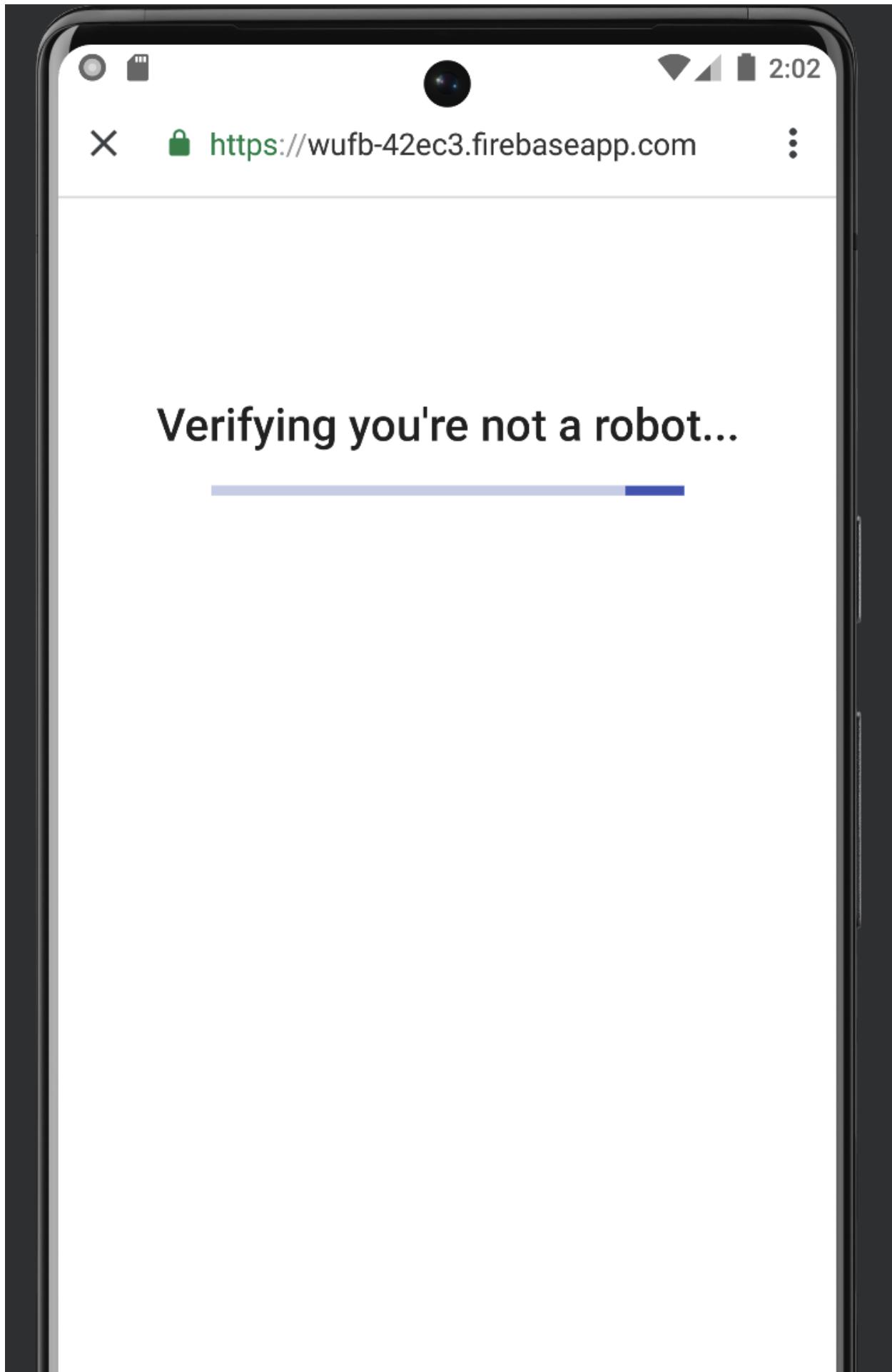
</androidx.cardview.widget.CardView>
```

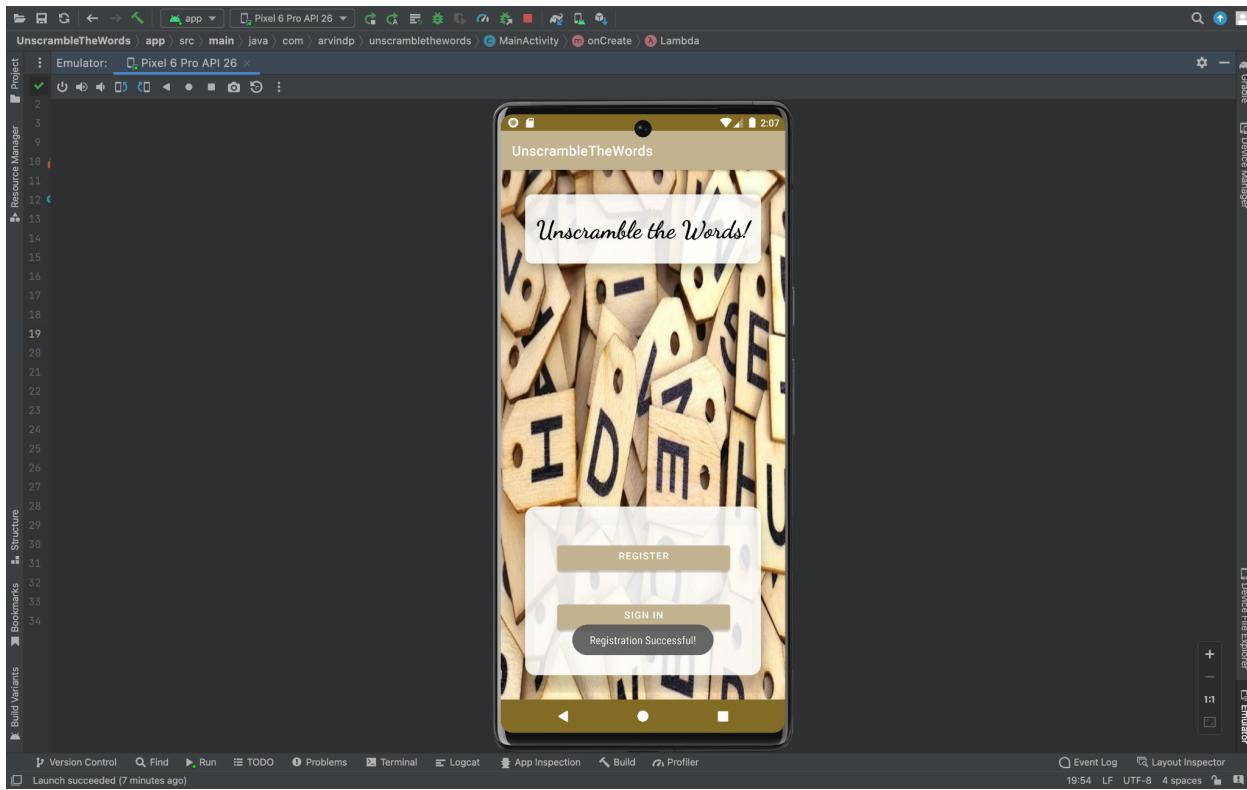
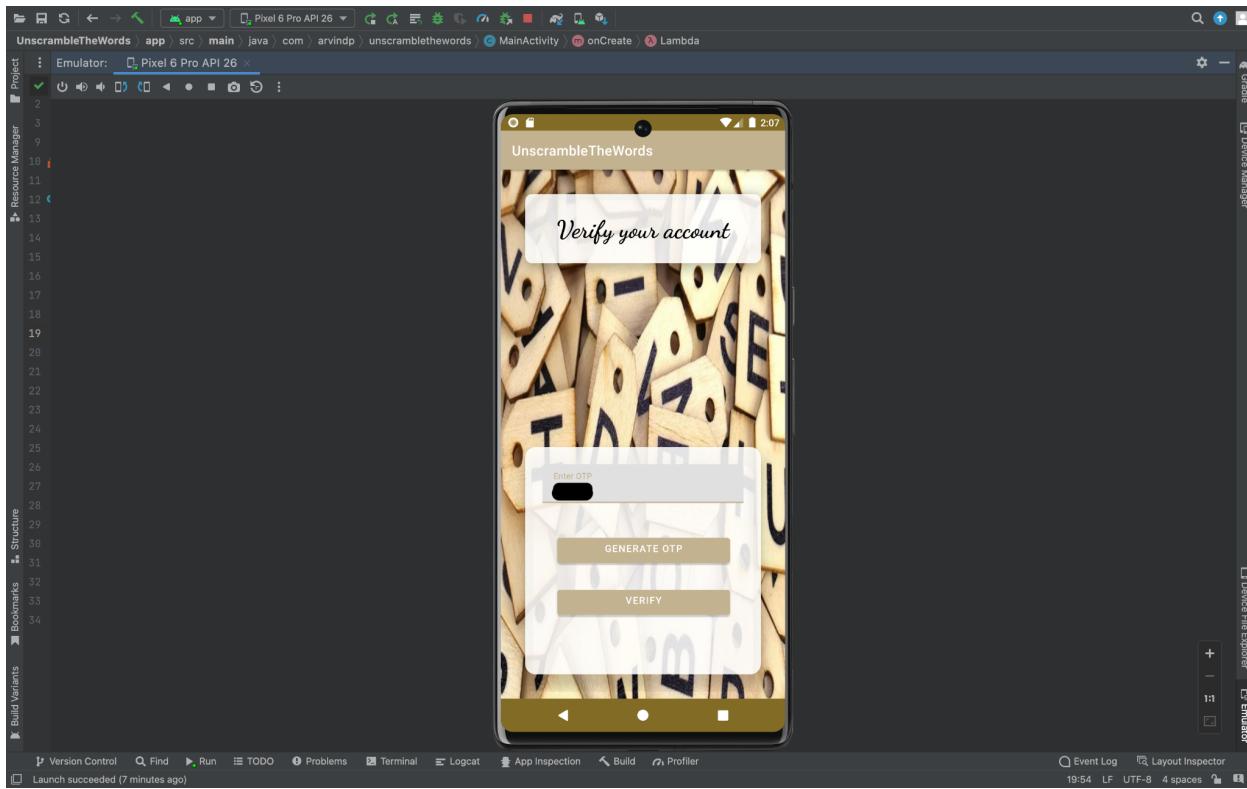
3.3. SCREENSHOTS:

REGISTRATION:

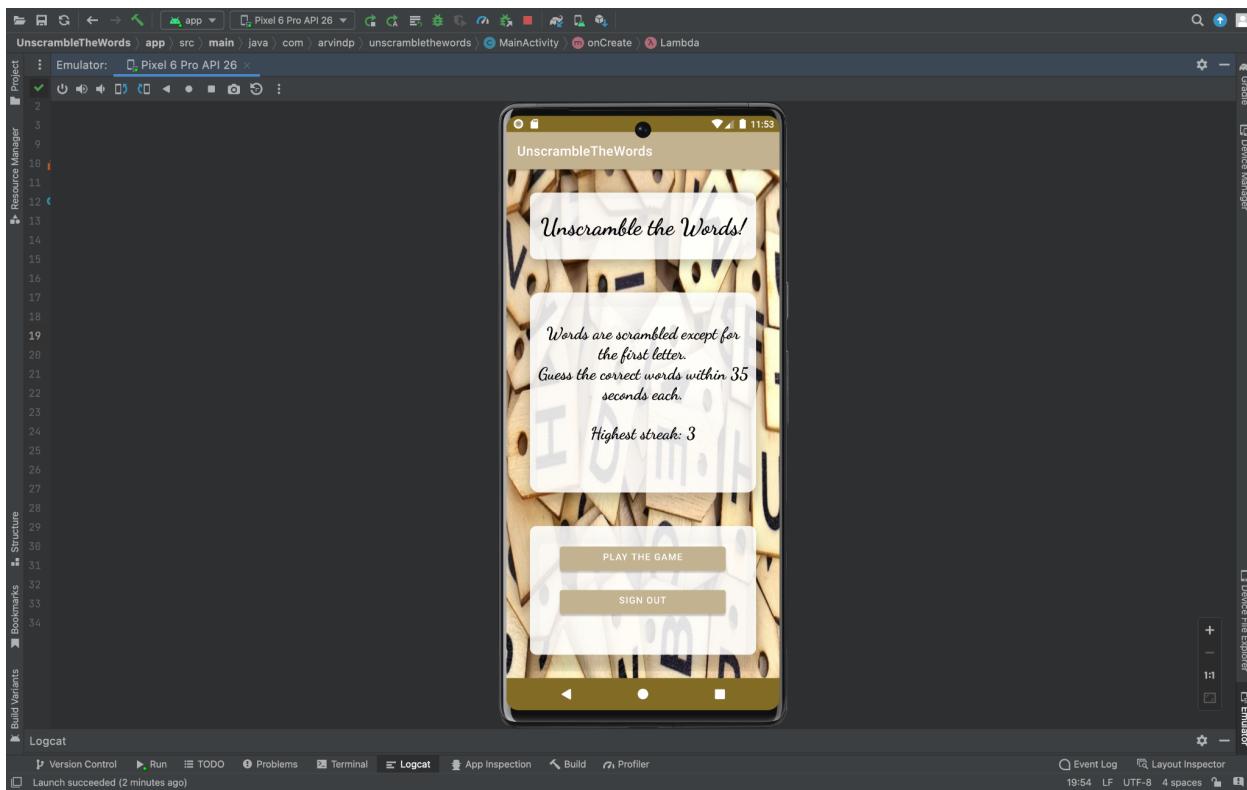
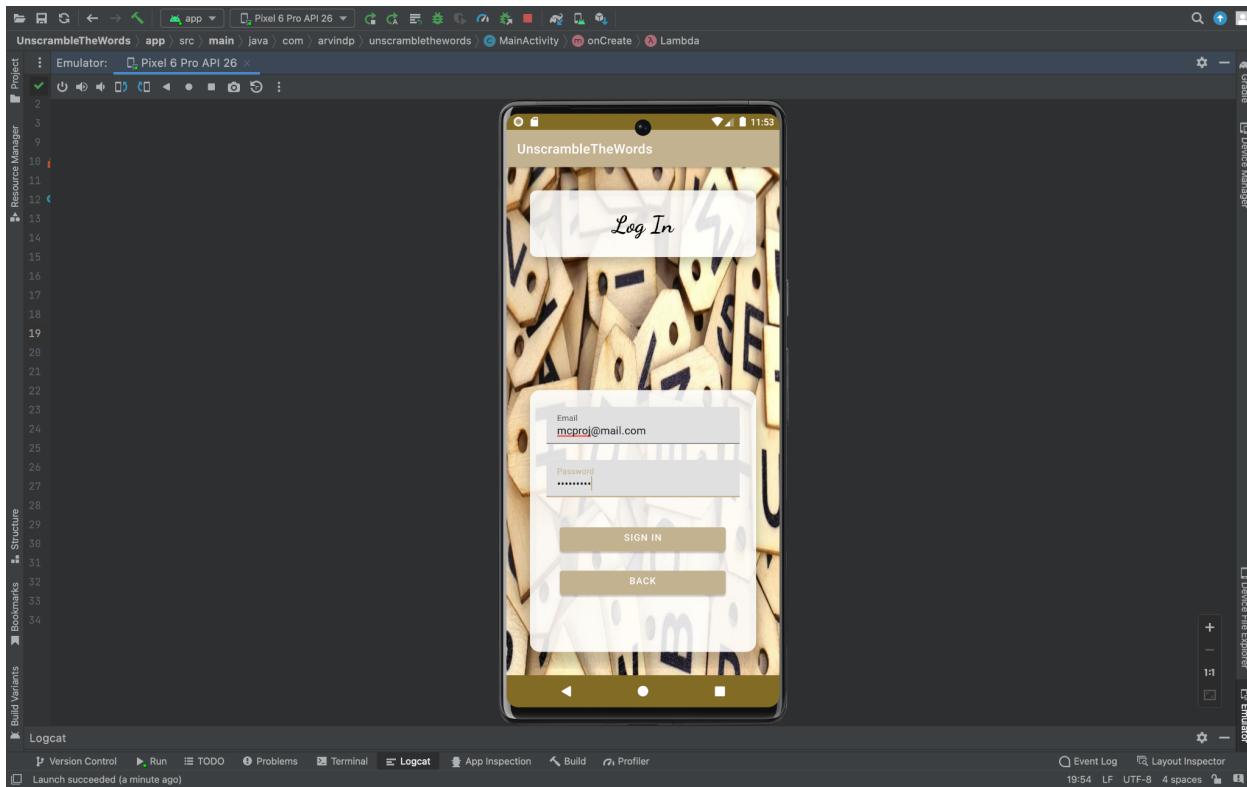




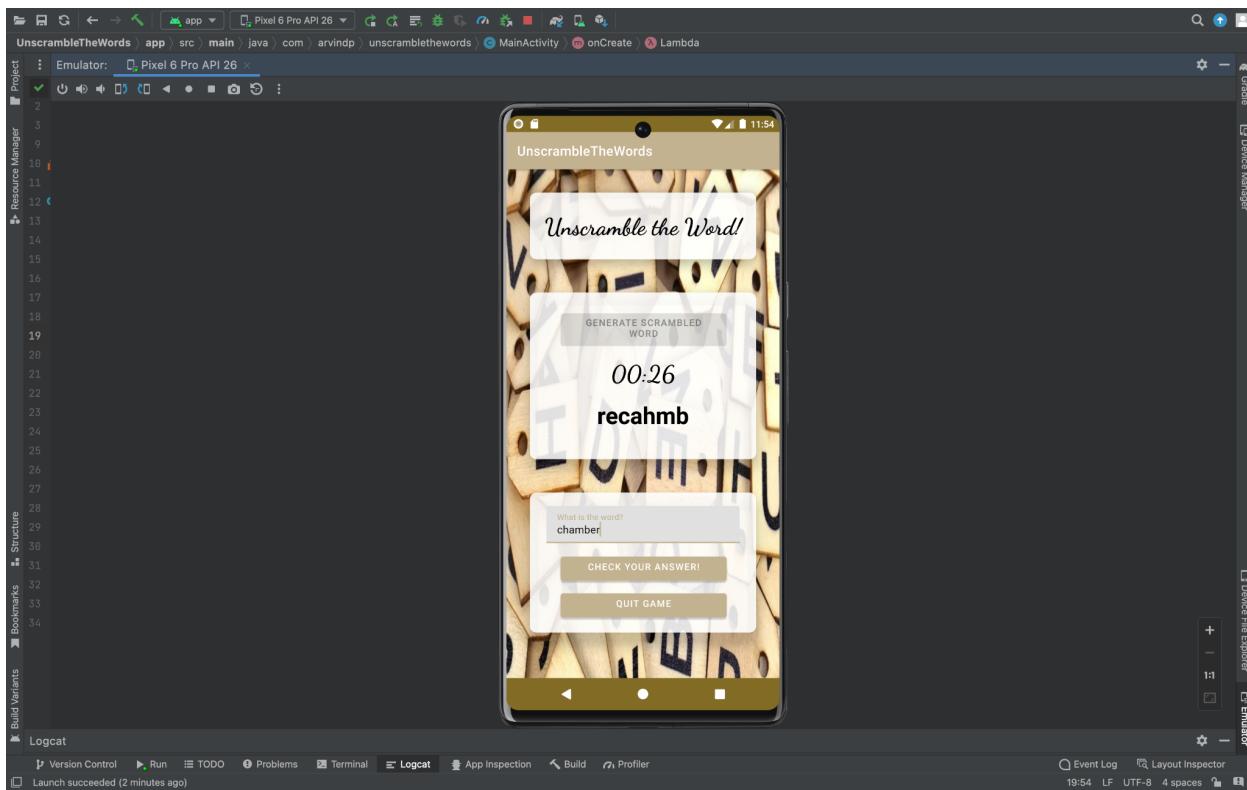
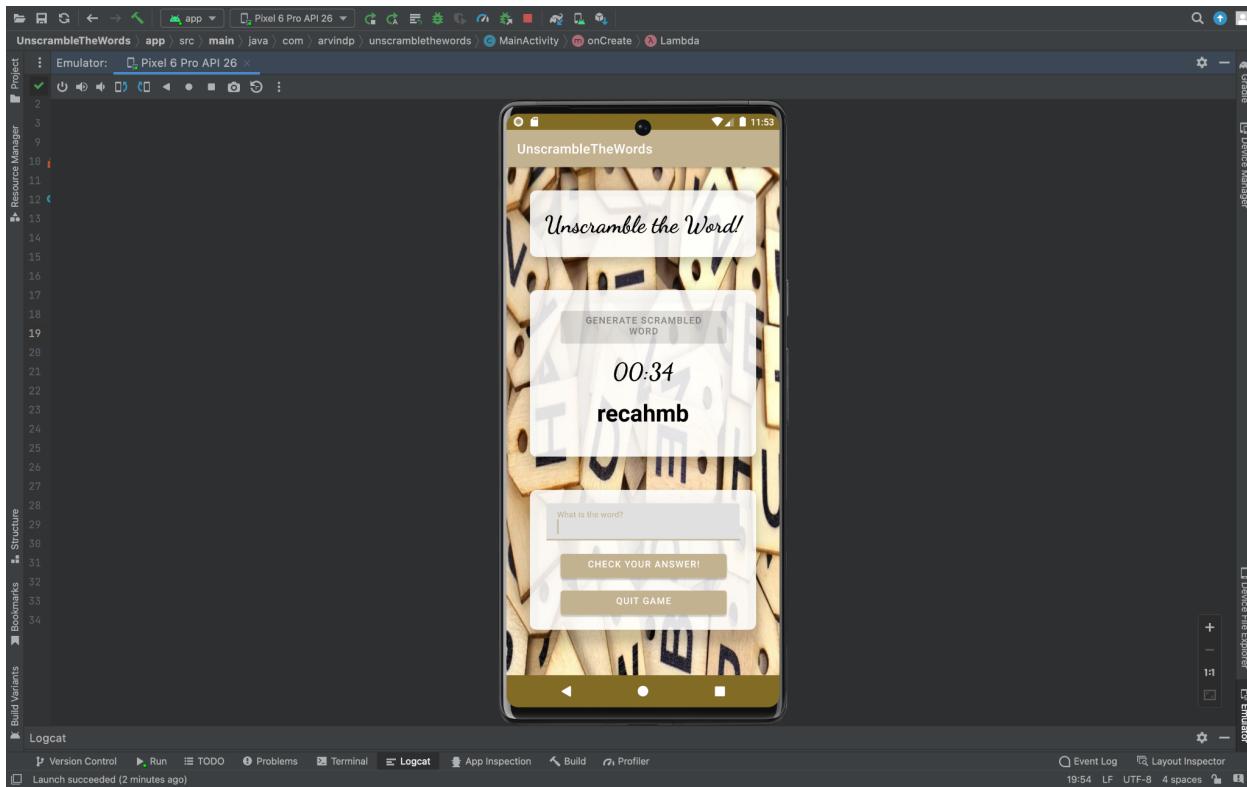


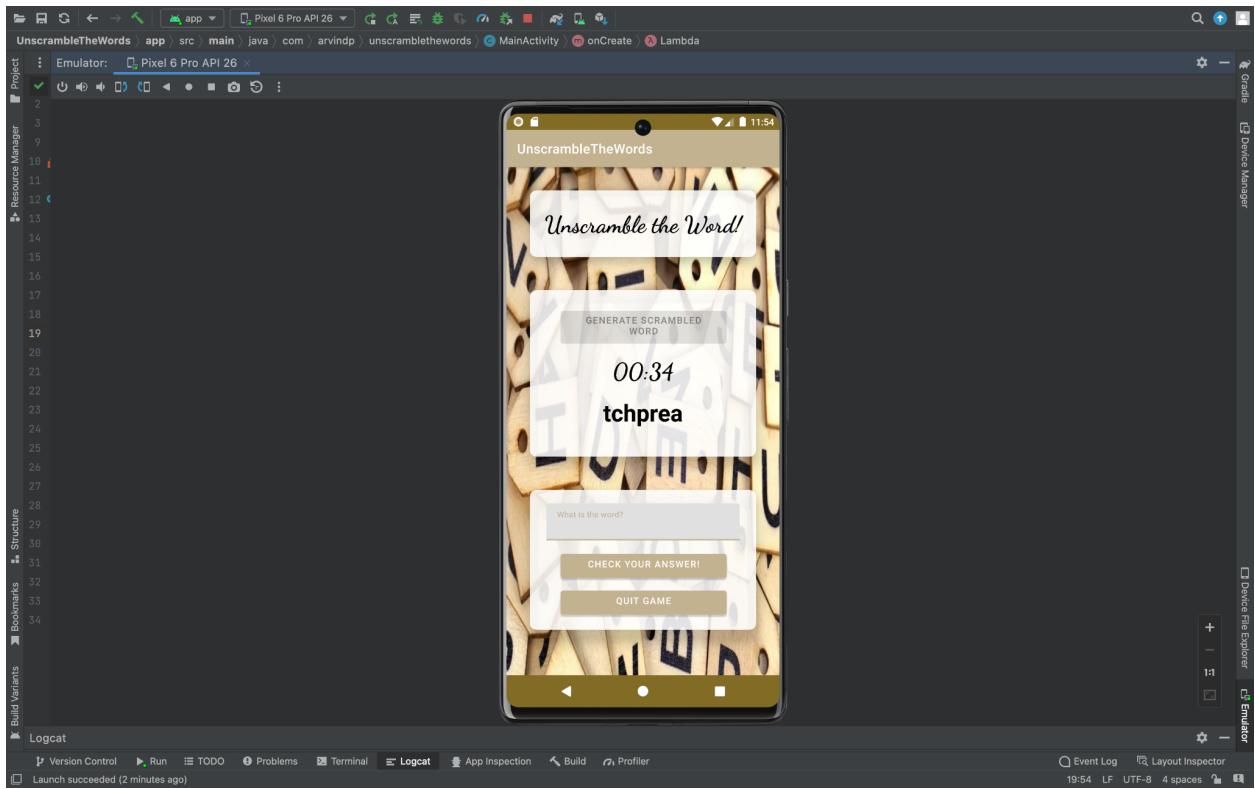


**SIGNING IN WITH A PRE-EXISTING ACCOUNT BEFORE
CREATION OF THE NEW ACCOUNT:**

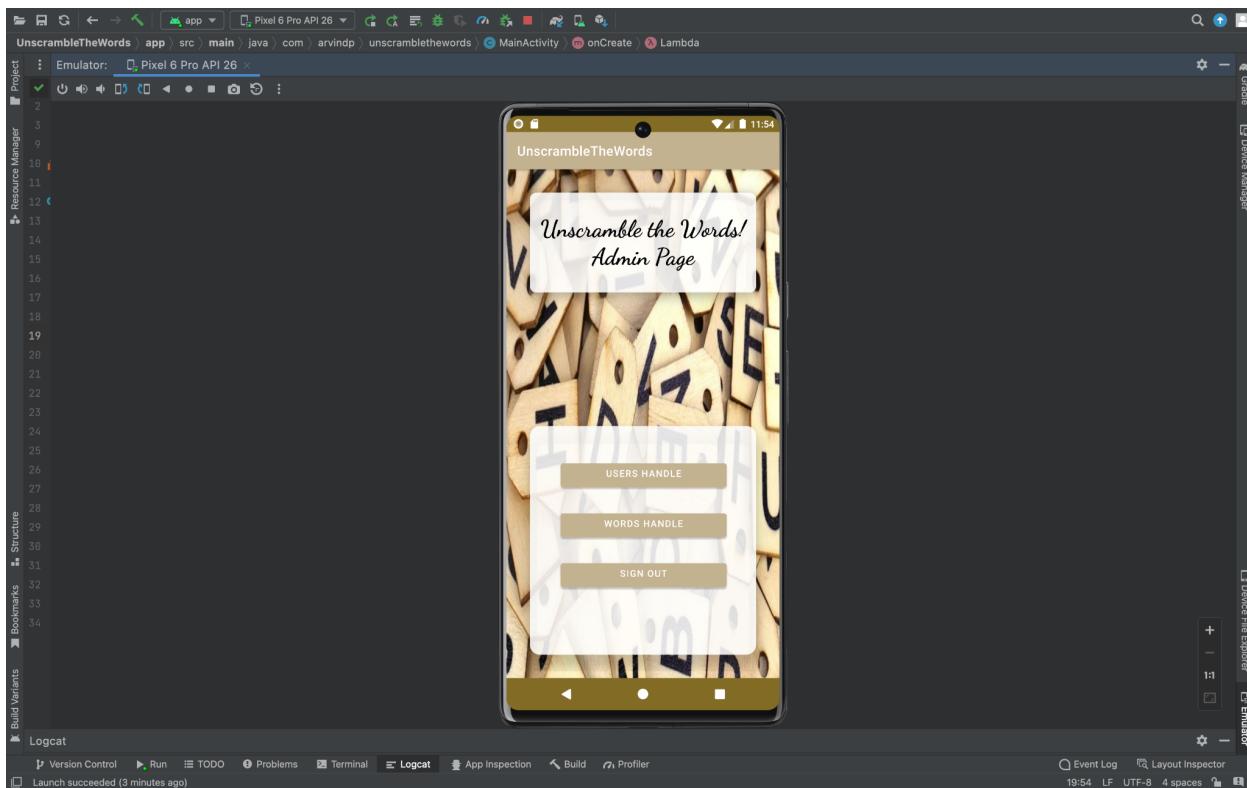
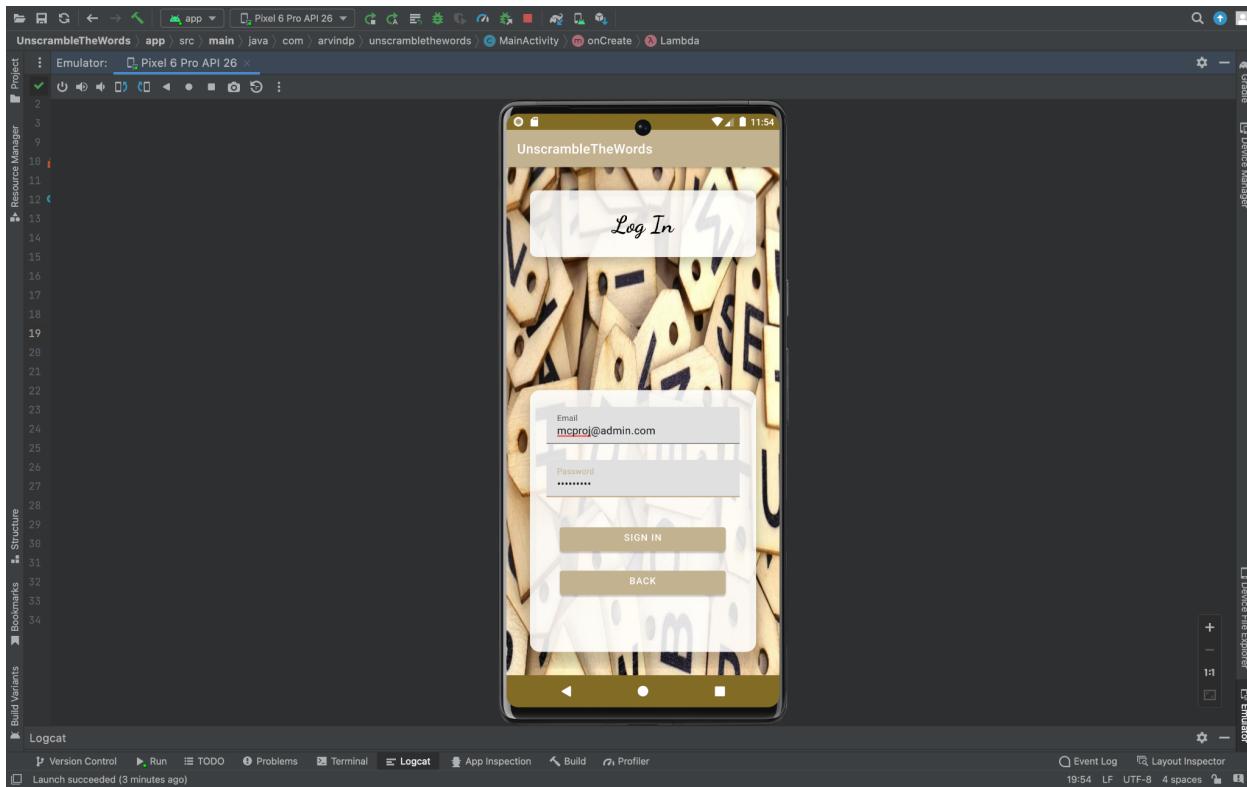


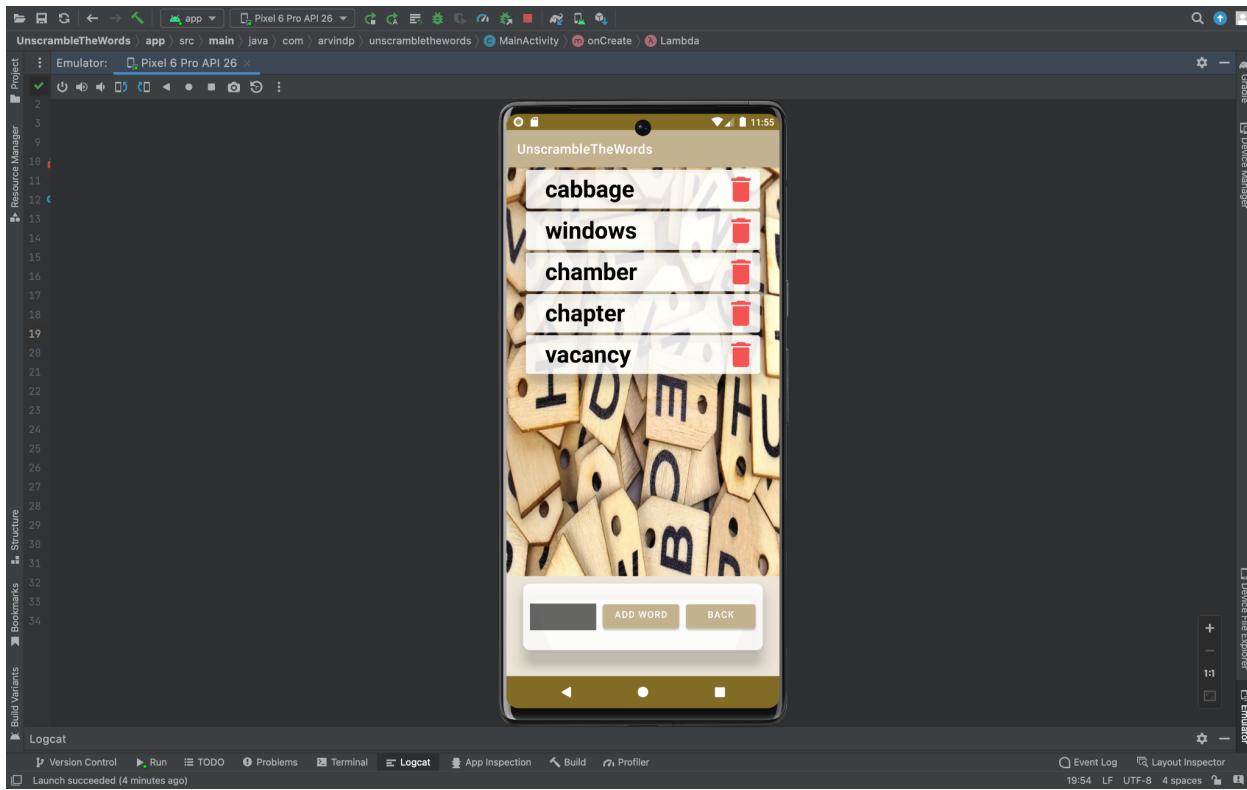
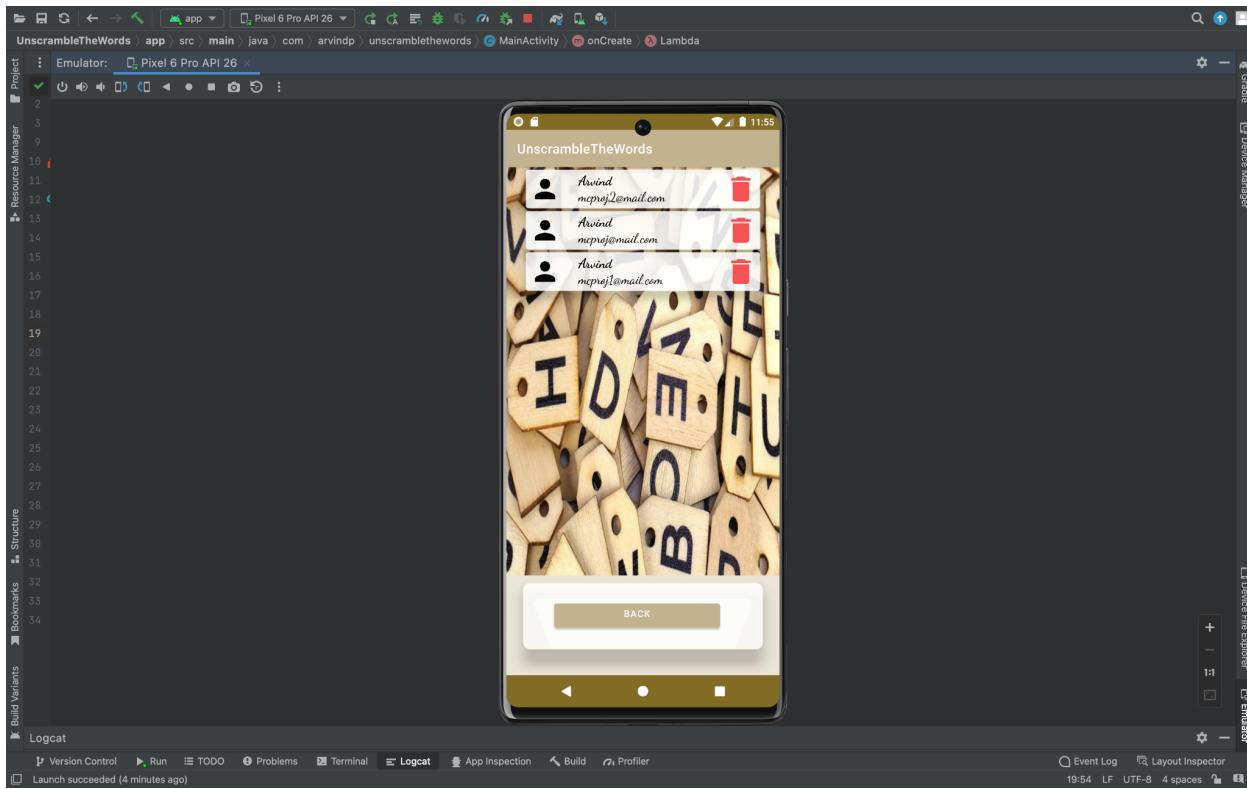
PLAYING THE GAME:





ADMINISTRATOR SIGN IN AND FEATURES:





CHAPTER 4

HARDWARE AND SOFTWARE

4.1. GOOGLE FIREBASE:

Firebase is a set of hosting services for any type of application (Android, iOS, Javascript, Node.js, Java, Unity, PHP, C++). It offers NoSQL and real-time hosting of databases, content, social authentication (Google, Facebook, Twitter and Github), and notifications, or services, such as a real-time communication server. In this project, Firebase Firestore DB and Authentication services have been made use of.

Firestore (NoSQL Database):

Cloud Firestore

Data Rules Indexes Usage

Panel view Query builder

wufb-42ec3 > users > V9KFgpowD538cikcKrWN

	users	V9KFgpowD538cikcKrWN
+ Start collection	+ Add document	+ Start collection
users	V9KFgpowD538cikcKrWN	+ Add field
word	X4o1U2f7Tf4LzZGhc8x3 YTxxy7pHmJTWDjJfh3YJ2 t013erDP6sADyPXkdiMr	email: "g8M4smXCLKPsSIXchYvfVTEv6LZLCd8YtH2OEkXah4=" highscore: "0zGNtwC7UvYEytCNCxS6fG==" password: "UZlDEq4x6q7lePBoElqhA==" phone number: "UZlDEq4x6q7lePBoElqhA==" username: "m081MpgGdT7OliQrOf8fg=="

Cloud Firestore

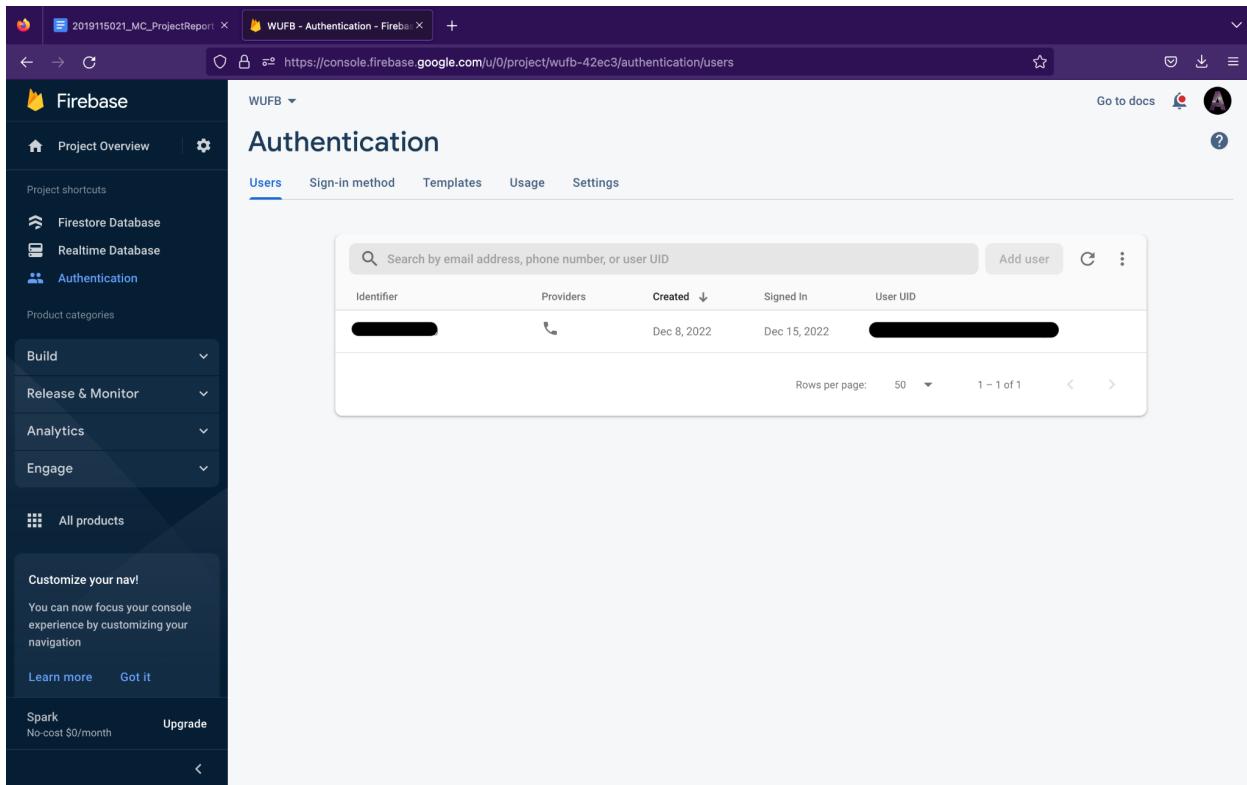
Data Rules Indexes Usage

Panel view Query builder

wufb-42ec3 > word > 8Y00QDnV94Mgo9IofsDV

	word	8Y00QDnV94Mgo9IofsDV
+ Start collection	+ Add document	+ Start collection
users	8Y00QDnV94Mgo9IofsDV	+ Add field
word	I47YNRzHh1UAeV014fNI N641Vo08EAlSxcuhwqsV N61Mr0ew9fG61uq5whT7 xEyDQ56yYAI7gUYNkLYq	word: "O3a4McqlPMY90Yqm1moeaQ=="

Authentication (SMS):

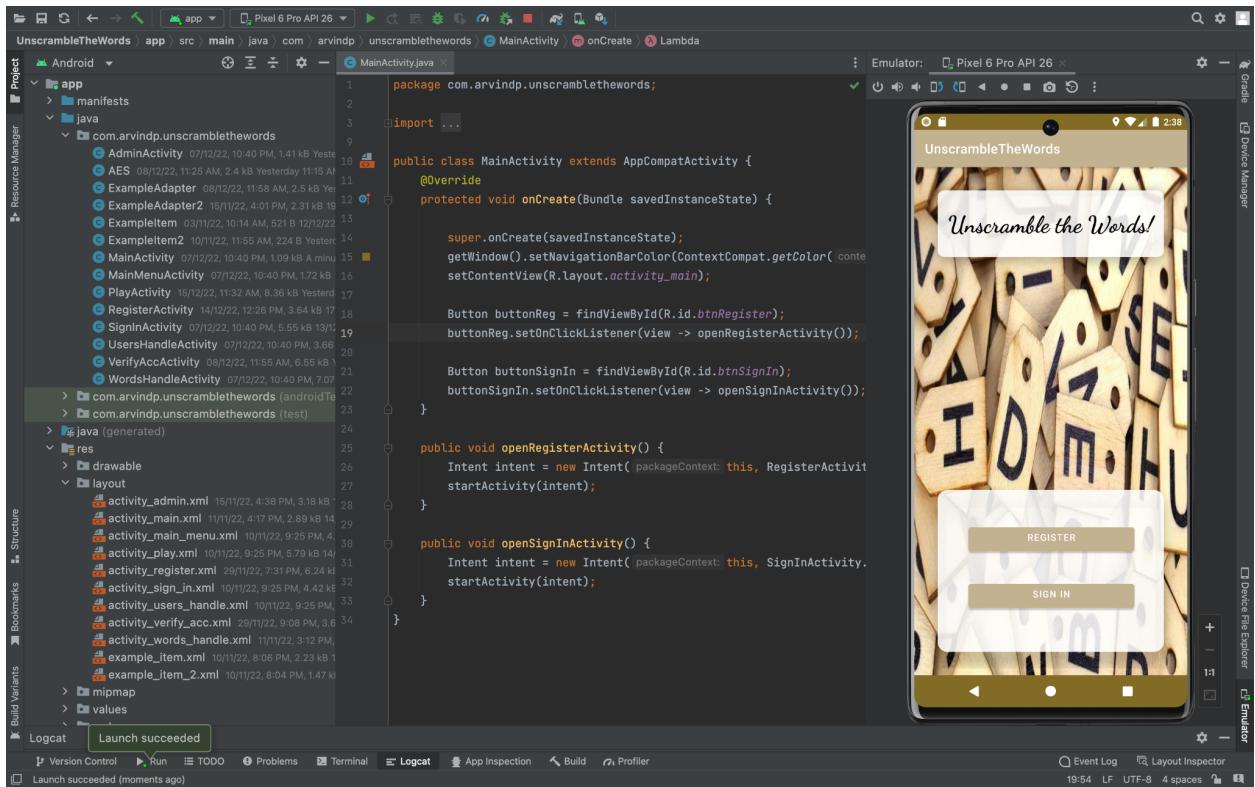


The screenshot shows the Firebase Authentication screen in the Google Cloud Platform console. The left sidebar includes links for Project Overview, Firestore Database, Realtime Database, and Authentication. The main content area is titled 'Authentication' and shows a table of users. The table has columns for Identifier, Providers, Created, Signed In, and User UID. One user entry is visible, with the Identifier and User UID redacted. The Providers column shows a phone icon. The Created date is Dec 8, 2022, and the Signed In date is Dec 15, 2022. A search bar at the top allows searching by email address, phone number, or user UID.

Identifier	Providers	Created	Signed In	User UID
[REDACTED]	Phone	Dec 8, 2022	Dec 15, 2022	[REDACTED]

4.2. ANDROID STUDIO:

Android Studio is the official multi-platform IDE for Google's Android OS, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.



CHAPTER 5

RESULTS AND CONCLUSION

5.1. RESULTS:

A fully functional word unscrambling android application has been developed. The application primarily makes use of Java, XML, and Firebase Firestore and Authentication. The user must be authenticated via a SMS with two-factor authentication before they are able to register with the application.

Once registered, the user is able to play the game and maintain a high score. A person cannot create an admin account directly from the application. An admin can remove users, and add or delete words from the database.

5.2. CONCLUSION AND FUTURE WORK:

Thus, an android application for a word unscrambling game has been successfully developed. As future work, I would like to improve upon the game by adding varying levels of difficulty. These would involve unscrambling 6, 7, and 8 letter words with different time limits for each word. I am also planning on adding statistics such as fastest guess, and average speed of guessing a word. This would enable the user to improve upon their skills by trying to get their best possible stats.

REFERENCES

<https://firebase.google.com/docs/android/setup>

<https://stackoverflow.com/questions/50035752/how-to-get-list-of-documents-from-a-collection-in-firebase-android>

<https://firebase.google.com/docs/firestore/query-data/get-data>

<https://developer.android.com/topic/libraries/support-library/packages.html#v7-recyclerview>

<https://developer.android.com/topic/libraries/support-library/packages.html#v7-cardview>

<https://www.geeksforgeeks.org/android-studio-tutorial/>

<https://codingzest.com/firebase-authentication-android/>

<https://stackoverflow.com/questions/46798981/firestore-how-to-get-random-documents-in-a-collection>

<https://stackoverflow.com/questions/14051962/change-edittext-password-mask-character-to-asterisk>