


































**Name: Pinky Marick**

**Roll No.: T23075**

**Subject: Android Mobile Programming**

**Project name: Guess the Jumbled word**

- ▼  app
  - ▼  manifests
    -  AndroidManifest.xml
  - ▼  java
    - ▼  com.example.wordgame
      -  CategoryActivity
      -  GameActivity
      -  HowToPlayActivity
      -  MainActivity
      -  ScoreActivity

- ▼  res
  - ▼  anim
    -  pulse\_animation.xml
    -  slide\_in.xml
  - ▼  drawable
    -  game\_logo.jpg
    -  gradient\_background.xml
    -  ic\_launcher\_background.xml
    -  ic\_launcher\_foreground.xml
  - ▼  font
    -  game\_font.xml
  - ▼  layout
    -  activity\_category.xml
    -  activity\_game.xml
    -  activity\_how\_to\_play.xml
    -  activity\_main.xml
    -  activity\_score.xml
  - >  mipmap
  - ▼  values
    -  colors.xml
    -  strings.xml
    -  styles.xml
  - >  themes (2)

## AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.wordgame">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity android:name=".HowToPlayActivity"/>
        <activity android:name=".CategoryActivity" />
        <activity android:name=".GameActivity" />
        <activity android:name=".ScoreActivity" />

    </application>
</manifest>
```

## libs.version.toml (version catalog)

```
[versions]
material = "1.5.0"

[libraries]
material = { module = "com.google.android.material:material", version.ref = "material" }
```

# MainActivity.java

```
package com.example.wordgame;
```

```
import android.content.Intent;
```

```
import android.os.Bundle;
```

```
import android.view.View;
```

```
import android.view.animation.Animation;
```

```
import android.view.animation.AnimationUtils;
```

```
import android.widget.Button;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```

```
        Button startButton = findViewById(R.id.startButton);
```

```
        Button howToPlayButton = findViewById(R.id.howToPlayButton);
```

```
        // Add button animation
```

```
        Animation pulse = AnimationUtils.loadAnimation(this, R.anim.pulse_animation);
```

```
        startButton.startAnimation(pulse);
```

```
        startButton.setOnClickListener(v -> {
```

```
            Intent intent = new Intent(MainActivity.this, CategoryActivity.class);
```

```
            startActivity(intent);
```

```
        });
```

```
        howToPlayButton.setOnClickListener(v -> {
```

```
            Intent intent = new Intent(MainActivity.this, HowToPlayActivity.class);
```

```
            startActivity(intent);
```

```
        });
```

```
    }
```

```
}
```

## HowToPlayActivity.java

```
package com.example.wordgame;
```

```
import android.os.Bundle;
```

```
import android.view.View;
```

```
import android.widget.Button;
```

```
import android.widget.TextView;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
public class HowToPlayActivity extends AppCompatActivity {  
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_how_to_play);
```

```
  
        TextView instructions = findViewById(R.id.instructionsText);
```

```
        Button backButton = findViewById(R.id.backButton);
```

```
  
        // Set game instructions
```

```
        instructions.setText("1. Choose a category.\n" +
```

```
            "2. A jumbled word will be shown with a clue.\n" +
```

```
            "3. Type the correct answer and press Submit.\n" +
```

```
            "4. Use hints if needed (-2 points per hint).\n" +
```

```
            "5. Try to score the highest points before you run out of attempts!");
```

```
  
        backButton.setOnClickListener(v -> finish()); // Close activity when back button is  
        clicked
```

```
    }
```

```
}
```

## CategoryActivity.java

```
package com.example.wordgame;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.view.animation.AnimationUtils;
import android.view.animation.Animation;
import androidx.appcompat.app.AppCompatActivity;

public class CategoryActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_category);

        Button[] categoryButtons = new Button[]{
            findViewById(R.id.animalsButton),
            findViewById(R.id.sportsButton),
            findViewById(R.id.foodButton)
        };

        // Add animation for buttons
        Animation slideIn = AnimationUtils.loadAnimation(this, R.anim.slide_in);
        for (int i = 0; i < categoryButtons.length; i++) {
            Button button = categoryButtons[i];
            button.startAnimation(AnimationUtils.loadAnimation(this, R.anim.slide_in));
            button.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {
                    String category = button.getText().toString();
                    Intent intent = new Intent(CategoryActivity.this, GameActivity.class);
                    intent.putExtra("CATEGORY", category);
                    startActivity(intent);
                }
            });
        }
    }
}
```

# GameActivity.java

```
package com.example.wordgame;
```

```
import android.content.Intent;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.View;
import android.view.inputmethod.EditorInfo;
import android.view.inputmethod.InputMethodManager;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import java.util.Random;
```

```
public class GameActivity extends AppCompatActivity {
    private String[] animalQuestions = {
        "The ____ is known as the king of the jungle.|LION|The animal has a mane",
        "A ____ can change its color to match its surroundings.|CHAMELEON|This reptile
has a long tongue",
        "The ____ is the tallest land animal.|GIRAFFE|It has a very long neck",
        "The ____ is the largest mammal on Earth.|BLUE WHALE|Lives in the ocean",
        "A group of ____ is called a pack.|WOLVES|They howl at the moon",
        "The ____ is the fastest land animal.|CHEETAH|Spotted and runs fast",
        "A ____ has black and white stripes.|ZEBRA|Looks like a striped horse",
        "The ____ has a pouch to carry its young.|KANGAROO|Hops on strong hind legs",
        "A ____ has eight legs and spins webs.|SPIDER|An arachnid",
        "The ____ is the only bird that can fly backward.|HUMMINGBIRD|Small and fast"
    };
};
```

```
private String[] sportQuestions = {
    "In ____, players hit a ball over a net.|TENNIS|This sport uses rackets",
    "A ____ has 11 players on each team.|SOCCER|Also known as football",
    "Players shoot hoops in _____.|BASKETBALL|Played on a court",
    "____ is played on ice with a puck.|HOCKEY|Players use sticks",
    "The ____ is used in baseball to hit the ball.|BAT|Made of wood or metal",
};
```

```

        "A ____ is used in cricket to hit the ball.|BAT|Also used in baseball",
        "In ____, players swim and throw a ball into the goal.|WATER POLO|Played in a
pool",
        "A player can score a touchdown in ____.|FOOTBALL|Popular in the U.S.",
        "____ is an Olympic sport involving balance beams and
rings.|GYMNASTICS|Athletes perform flips",
        "A ____ is used in golf to hit the ball.|CLUB|Comes in different sizes"
    };

```

```

private String[] foodQuestions = {
    "A ____ is made from ground beef between buns.|BURGER|Fast food favorite",
    "____ is made from tomatoes and often used on pasta.|SAUCE|Red liquid
condiment",
    "A ____ is a sweet frozen dairy dessert.|ICECREAM|Cold and creamy",
    "____ is a yellow dairy product used for cooking.|BUTTER|Made from milk",
    "A ____ is a red fruit often mistaken for a vegetable.|TOMATO|Used in salads",
    "____ is a tropical fruit with a spiky exterior.|PINEAPPLE|Sweet and juicy",
    "____ is a type of bread often used for sandwiches.|BAGUETTE|Long and crunchy",
    "A ____ is made from layers of pasta, meat, and cheese.|LASAGNA|Italian dish",
    "____ is a popular Japanese dish made with rice and raw fish.|SUSHI|Eaten with
chopsticks",
    "____ is a fizzy drink often served cold.|SODA|Comes in different flavors"
};

```

```

private String[][] allQuestions;
private int currentQuestionIndex = 0;
private int attempts = 3; // Set max attempts
private int score = 0;
private String currentAnswer;
private String currentHint;

```

```

private TextView questionText, jumbledText, scoreText, attemptsText;
private EditText answerInput;
private Button submitButton, hintButton;

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

```



```
setContentView(R.layout.activity_game);
```

```
// Initialize UI Elements
```

```
questionText = findViewById(R.id.questionText);
```

```
jumbledText = findViewById(R.id.jumbledText);
```

```
scoreText = findViewById(R.id.scoreText);
```

```
attemptsText = findViewById(R.id.attemptsText);
```

```
answerInput = findViewById(R.id.answerInput);
```

```
submitButton = findViewById(R.id.submitButton);
```

```
hintButton = findViewById(R.id.hintButton);
```

```
// Get category from intent
```

```
String category = getIntent().getStringExtra("CATEGORY");
```

```
switch (category) {
```

```
    case "Animals":
```

```
        allQuestions = new String[][]{animalQuestions};
```

```
        break;
```

```
    case "Sports":
```

```
        allQuestions = new String[][]{sportQuestions};
```

```
        break;
```

```
    case "Food":
```

```
        allQuestions = new String[][]{foodQuestions};
```

```
        break;
```

```
}
```

```
setupQuestion();
```

```
// Submit Button Click Listener
```

```
submitButton.setOnClickListener(new View.OnClickListener() {
```

```
    @Override
```

```
    public void onClick(View v) {
```

```
        checkAnswer();
```

```
    }
```

```
});
```

```
// Hint Button Click Listener
```

```
hintButton.setOnClickListener(new View.OnClickListener() {
```

```
    @Override
```

```
    public void onClick(View v) {
```

```
        showHint();
    }
});
```

```
// Enable Enter Key to Work as Submit
```

```
answerInput.setOnEditorActionListener(new TextView.OnEditorActionListener() {
    @Override
    public boolean onEditorAction(TextView v, int actionId, KeyEvent event) {
        if (actionId == EditorInfo.IME_ACTION_DONE) {
            checkAnswer();
            return true;
        }
        return false;
    }
});
```

```
answerInput.setOnKeyListener(new View.OnKeyListener() {
    @Override
    public boolean onKey(View v, int keyCode, KeyEvent event) {
        if (event.getAction() == KeyEvent.ACTION_DOWN && keyCode ==
KeyEvent.KEYCODE_ENTER) {
            checkAnswer();
            return true;
        }
        return false;
    }
});
}
```

```
private void setupQuestion() {
    String[] questionData = allQuestions[0][currentQuestionIndex].split("\\|");
    String question = questionData[0];
    currentAnswer = questionData[1];
    currentHint = questionData[2];

    questionText.setText(question);
    jumbledText.setText(jumbleWord(currentAnswer));
    scoreText.setText("Score: " + score);
    attemptsText.setText("Attempts Left: " + attempts);
}
```

```
}
```

```
private String jumbleWord(String word) {  
    char[] chars = word.toCharArray();  
    Random random = new Random();  
    for (int i = chars.length - 1; i > 0; i--) {  
        int j = random.nextInt(i + 1);  
        char temp = chars[i];  
        chars[i] = chars[j];  
        chars[j] = temp;  
    }  
    return new String(chars);  
}
```

```
private void checkAnswer() {  
    String userAnswer = answerInput.getText().toString().trim().toUpperCase();  
  
    if (userAnswer.equals(currentAnswer)) {  
        score += 10;  
        Toast.makeText(this, "Correct!", Toast.LENGTH_SHORT).show();  
        currentQuestionIndex++;  
        attempts = 3; // Reset attempts on correct answer  
  
        if (currentQuestionIndex < allQuestions[0].length) {  
            setupQuestion();  
        } else {  
            showFinalScore();  
        }  
    } else {  
        attempts--;  
        if (attempts <= 0) {  
            showFinalScore();  
        } else {  
            Toast.makeText(this, "Wrong! Attempts left: " + attempts,  
Toast.LENGTH_SHORT).show();  
            attemptsText.setText("Attempts Left: " + attempts);  
        }  
    }  
}
```

```
    answerInput.setText("");  
    hideKeyboard();  
}
```

```
private void showHint() {  
    Toast.makeText(this, "Hint: " + currentHint, Toast.LENGTH_LONG).show();  
    score -= 2; // Penalty for using hint  
    scoreText.setText("Score: " + score);  
}
```

```
private void showFinalScore() {  
    Intent intent = new Intent(this, ScoreActivity.class);  
    intent.putExtra("SCORE", score);  
    startActivity(intent);  
    finish();  
}
```

```
private void hideKeyboard() {  
    View view = this.getCurrentFocus();  
    if (view != null) {  
        InputMethodManager imm = (InputMethodManager)  
getSystemService(INPUT_METHOD_SERVICE);  
        imm.hideSoftInputFromWindow(view.getWindowToken(), 0);  
    }  
}  
}
```

# ScoreActivity.java

```
package com.example.wordgame;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class ScoreActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_score);

        int finalScore = getIntent().getIntExtra("SCORE", 0);
        TextView scoreText = findViewById(R.id.finalScoreText);
        scoreText.setText("Final Score: " + finalScore);

        Button replayButton = findViewById(R.id.replayButton);
        Button menuButton = findViewById(R.id.menuButton);

        replayButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(ScoreActivity.this, CategoryActivity.class);
                startActivity(intent);
                finish();
            }
        });

        menuButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(ScoreActivity.this, MainActivity.class);
                intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                startActivity(intent);
                finish();
            }
        });
    }
}
```

## res/layout/activity\_main.xml

### activity\_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/gradient_background"
    android:padding="16dp">

    <ImageView
        android:id="@+id/logoImage"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="64dp"
        android:src="@drawable/game_logo"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/logoImage"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="32dp"
        android:text="TRICKY WORDS"
        android:textColor="@android:color/white"
        android:textSize="36sp"
        android:textStyle="bold"/>

    <com.google.android.material.button.MaterialButton
        android:id="@+id/startButton"
        style="@style/GameButton"
        android:layout_centerInParent="true"
        android:text="Start Game"
        android:layout_marginTop="48dp"/>
```

```
<com.google.android.material.button.MaterialButton
    android:id="@+id/howToPlayButton"
    style="@style/GameButton"
    android:layout_below="@id/startButton"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="16dp"
    android:text="How to Play"/>
</RelativeLayout>
```

## **res/layout/activity\_how\_to\_play.xml**

### **activity\_how\_to\_play.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:padding="24dp"
    android:background="@drawable/gradient_background">
    <TextView
        android:id="@+id/instructionsText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:textColor="@android:color/white"
        android:textStyle="bold"
        android:padding="6dp"
        android:lineSpacingExtra="20sp"
        android:gravity="start"
        android:textAlignment="viewStart"/>
    <com.google.android.material.button.MaterialButton
        android:id="@+id/backButton"
        style="@style/GameButton"
        android:text="Back"
        android:layout_marginTop="32dp"/>
</LinearLayout>
```

## res/layout/activity\_category.xml

### activity\_category.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:background="@drawable/gradient_background"
    android:padding="16dp">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select Category"
        android:textSize="32sp"
        android:textColor="@android:color/white"
        android:textStyle="bold"
        android:layout_marginBottom="48dp"/>

    <com.google.android.material.card.MaterialCardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:padding="16dp">

            <com.google.android.material.button.MaterialButton
                android:id="@+id/animalsButton"
                style="@style/GameButton"
                android:layout_gravity="center"
                android:text="Animals"
                android:layout_marginBottom="16dp"/>
        </LinearLayout>
    </com.google.android.material.card.MaterialCardView>
</LinearLayout>
```



```

        <com.google.android.material.button.MaterialButton
            android:id="@+id/sportsButton"
            style="@style/GameButton"
            android:layout_gravity="center"
            android:text="Sports"
            android:layout_marginBottom="16dp"/>

        <com.google.android.material.button.MaterialButton
            android:id="@+id/foodButton"
            style="@style/GameButton"
            android:layout_gravity="center"
            android:text="Food"/>
    </LinearLayout>
</com.google.android.material.card.MaterialCardView>
</LinearLayout>

```

## res/layout/activity\_game.xml

### activity\_game.xml

```

<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/gradient_background"
    android:padding="16dp">

    <com.google.android.material.card.MaterialCardView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="8dp"
        app:cardCornerRadius="15dp"
        app:cardElevation="8dp"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="50dp"> <!-- Moved the card slightly upwards -->

```

<LinearLayout

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="16dp">
```

<TextView

```
    android:id="@+id/scoreText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Score: 0"
    android:textSize="24sp"
    android:textColor="@color/accentColor"
    android:layout_gravity="end"/>
```

<TextView

```
    android:id="@+id/attemptsText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Attempts Left: 3"
    android:textSize="18sp"
    android:textColor="@color/accentColor"
    android:layout_gravity="center"
    android:layout_marginBottom="16dp"/>
```

<TextView

```
    android:id="@+id/questionText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="28sp"
    android:textColor="@color/textColor"
    android:layout_marginTop="16dp"
    android:layout_marginBottom="8dp"
    android:gravity="center"/>
```

<TextView

```
    android:id="@+id/jumbledText"
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
    android:textSize="32sp"
    android:textStyle="bold"
    android:textColor="@color/primaryColor"
    android:gravity="center"
    android:layout_marginBottom="32dp"/>
```

```
<com.google.android.material.textfield.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter your answer"
    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox">
```

```
    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/answerInput"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:imeOptions="actionDone"
        android:inputType="text"/>
```

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

```
<com.google.android.material.button.MaterialButton
    android:id="@+id/submitButton"
    style="@style/GameButton"
    android:layout_gravity="center"
    android:text="Submit"
    android:layout_marginTop="16dp"
    android:layout_marginBottom="16dp"/>
```

```
<com.google.android.material.button.MaterialButton
    android:id="@+id/hintButton"
    style="@style/GameButton"
    android:layout_gravity="center"
    android:text="Get Hint (-2 points)"
    app:backgroundTint="@color/accentColor"
    android:layout_marginBottom="24dp"/> <!-- Moved hint button up -->
```

```
</LinearLayout>
```

```
</com.google.android.material.card.MaterialCardView>
```

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

## res/layout/activity\_score.xml

### activity\_score.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:padding="16dp">

    <TextView
        android:id="@+id/finalScoreText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Final Score: 0"
        android:textSize="32sp"
        android:textStyle="bold"
        android:layout_marginBottom="32dp"/>

    <Button
        android:id="@+id/replayButton"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:text="Play Again"
        android:layout_marginBottom="16dp"/>

    <Button
        android:id="@+id/menuButton"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:text="Main Menu"/>
</LinearLayout>
```

## **res/anim/pulse\_animation.xml**

### **pulse\_animation.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:interpolator="@android:anim/linear_interpolator">
    <scale
        android:duration="1000"
        android:fromXScale="1.0"
        android:fromYScale="1.0"
        android:pivotX="50%"
        android:pivotY="50%"
        android:repeatCount="infinite"
        android:repeatMode="reverse"
        android:toXScale="1.1"
        android:toYScale="1.1" />
</set>
```

## **res/anim/slide\_in.xml**

### **slide\_in.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:interpolator="@android:anim/decelerate_interpolator">
    <translate
        android:duration="500"
        android:fromXDelta="100%"
        android:toXDelta="0%" />
    <alpha
        android:duration="500"
        android:fromAlpha="0.0"
        android:toAlpha="1.0" />
</set>
```

**res/drawable/game\_logo.jpg**



**res/drawable/gradient\_background.xml**

**gradient\_background.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
  <gradient
    android:startColor="#6200EE"
    android:endColor="#03DAC5"
    android:angle="135"/>
</shape>
```

## res/values/colors.xml

### colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="primaryColor">#6200EE</color>
    <color name="primaryDarkColor">#3700B3</color>
    <color name="accentColor">#03DAC5</color>
    <color name="buttonColor">#FF4081</color>
    <color name="backgroundColor">#F0F0F0</color>
    <color name="textColor">#212121</color>
</resources>
```

## res/values/styles.xml

### styles.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <!-- Base application theme -->
    <style name="AppTheme" parent="Theme.MaterialComponents.Light.NoActionBar">
        <item name="colorPrimary">@color/primaryColor</item>
        <item name="colorPrimaryDark">@color/primaryDarkColor</item>
        <item name="colorAccent">@color/accentColor</item>
        <item name="android:windowBackground">@color/backgroundColor</item>
        <item name="android:textColor">@color/textColor</item>

        <!-- Custom font for the entire app -->
        <item name="android:fontFamily">@font/game_font</item>

        <!-- Button style -->
        <item name="materialButtonStyle">@style/GameButton</item>
    </style>

    <!-- Custom button style -->
    <style name="GameButton" parent="Widget.MaterialComponents.Button">
        <item name="android:layout_width">200dp</item>
        <item name="android:layout_height">wrap_content</item>
        <item name="android:textSize">18sp</item>
```

```
<item name="android:padding">12dp</item>
<item name="cornerRadius">25dp</item>
<item name="android:textColor">@android:color/white</item>
<item name="backgroundTint">@color/buttonColor</item>
<item name="android:textAllCaps">false</item>
<item name="android:elevation">4dp</item>
</style>
```

```
<!-- Card style -->
```

```
<style name="GameCard" parent="Widget.MaterialComponents.CardView">
  <item name="cardCornerRadius">15dp</item>
  <item name="cardElevation">8dp</item>
  <item name="contentPadding">16dp</item>
  <item name="android:layout_margin">8dp</item>
</style>
```

```
<!-- Text styles -->
```

```
<style name="TitleText">
  <item name="android:textSize">32sp</item>
  <item name="android:textStyle">bold</item>
  <item name="android:textColor">@android:color/white</item>
  <item name="android:gravity">center</item>
</style>
```

```
<style name="QuestionText">
  <item name="android:textSize">24sp</item>
  <item name="android:textColor">@color/textColor</item>
  <item name="android:gravity">center</item>
  <item name="android:layout_margin">16dp</item>
</style>
```

```
<style name="ScoreText">
  <item name="android:textSize">20sp</item>
  <item name="android:textColor">@color/accentColor</item>
  <item name="android:textStyle">bold</item>
</style>
```

```
</resources>
```



