

Escuela Superior Politécnica del Litoral

Aplicaciones Móviles y

Servicios Telemáticos

Paralelo 1

Primera evaluación práctica

MegaCine

Msig. Adriana Collaguazo

Grupo 6

Abel Silva

Bolívar Núñez

11/29/2019

Descripción del código:

-MainActivity.java

Se creó los tres métodos, de los cuales el primero “cambiarVentana” permite direccionarnos directamente a una de las actividades de pregunta según el valor aleatorio que se obtenga del Random, cuyo valor máximo es 4 incrementado 1; este valor es utilizado en el método “otraActivity” y es utilizado por todas las actividades, por lo que, siempre lo vamos a encontrar. El otro método, es el “ventanaPolíticas”, el cual nos redirige a la actividad “políticas” en donde podremos visualizar las reglas del juego.

```
public void cambiarVentana(View view){
    Random rand = new Random();
    int n = rand.nextInt( bound: 4)+1;
    otraActivity(n);
}

public void ventanaPolíticas(View view){
    Intent a2 = new Intent( packageContext this , politicas.class);
    startActivity(a2);
}

public void otraActivity(int n){
    if (n==1) {
        Intent intent = new Intent( packageContext this, Pregunta1.class);
        intent.putExtra( name: "i", i);
        startActivity(intent);
    }
    if (n==2) {
        Intent intent = new Intent( packageContext this, Pregunta2.class);
        intent.putExtra( name: "i", i);
        startActivity(intent);
    }
    if (n==3) {
        Intent intent = new Intent( packageContext this, Pregunta3.class);
        intent.putExtra( name: "i", i);
        startActivity(intent);
    }
    if (n==4) {
        Intent intent = new Intent( packageContext this, Pregunta4.class);
        intent.putExtra( name: "i", i);
        startActivity(intent);
    }
    if (n==5) {
        Intent intent = new Intent( packageContext this, Pregunta5.class);
        intent.putExtra( name: "i", i);
        startActivity(intent);
    }
}
```

-MainActivity.xml

En la interfaz gráfica se utilizó la view de ImageView, para colocar el logo de la app, TextView para la colocación de la información e integrantes, y dos botones que permiten ingresar al juego o ver las reglas del juego a través de los respectivos métodos implementados en el archivo java respectivo. Cada view se le modificó el espaciado entre márgenes, tamaño de texto e incluso color de texto y fondo.

```
<ImageView
    android:id="@+id/imagen"
    android:layout_width="match_parent"
    android:layout_height="241dp"
    app:srcCompat="@drawable/megacine" />

<TextView
    android:id="@+id/textView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="30dp"
    android:layout_marginRight="20dp"
    android:text="TRIVIA MÁGICA"
    android:textAlignment="center"
    android:textColor="#33332F"
    android:textSize="24dp" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Desarrollado por Abel Silva y Bolívar Nuñez"
    android:textAlignment="center"
    android:layout_marginTop="60dp"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:textColor="#292623"
    android:textSize="15dp"
    android:textStyle="italic"/>

<Button
    android:id="@+id/btnEmpezar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="50dp"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:textSize="30dp"
    android:background="@color/colorAccent"
    android:textColor="@color/white"
    android:onClick="cambiarVentana"
    android:text="Empezar" />

<Button
    android:id="@+id/btnPolíticas"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="50dp"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:onClick="ventanaPolíticas"
    android:text="Mostrar Políticas" />
```

-Pregunta1.java, Pregunta2.java, Pregunta3.java, Pregunta4.java, Pregunta5.java

El formato para las actividades de pregunta son similares, con el único cambio en la validación de la respuesta correcta. Estas actividades, a más de contener el método “otraActivity” anteriormente descrita, tiene el método btn_pregunta que permite validar el botón que contenga la respuesta correcta. En caso de que el usuario acierte el valor de “i” incrementara en 1 e implementara el método otraActivity para direccionar a la siguiente pregunta de forma aleatoria solo si el valor de i no es igual a 5, caso contrario la persona gana el juego y es direccionado a la actividad “pantallaFinal”. Por otro lado, si el usuario no acierta pierde el juego y es direccionado a la actividad “PantallaPerdido”. Cabe recalcar que anterior a esto, se instaciaron los respectivos botones con el valor de la id de los botones que se encuentran en el respectivo archivo xml.

```
private Button btnAsk_1, btnAsk_2, btnAsk_3, btnAsk_4;
private TextView txtPregunta;
int i;

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

    btnAsk_1 = findViewById(R.id.btnAsk_1);
    btnAsk_2 = findViewById(R.id.btnAsk_2);
    btnAsk_3 = findViewById(R.id.btnAsk_3);
    btnAsk_4 = findViewById(R.id.btnAsk_4);
    txtPregunta = findViewById(R.id.textPregunta);

    i = getIntent().getIntExtra( name: "i", defaultValue: 0);
}

public void btn_Pregunta (View view){
    Random rand = new Random();
    int n = rand.nextInt( bound: 4)+1;

    if (view.getId()==R.id.btnAsk_4) {
        i+=1;
        if( i==5){
            Intent intent = new Intent( packageContext: this, pantallaFinal.class);
            startActivity(intent);
        }else{
            otraActivity(n);
        }
    }else{
        Intent intent = new Intent( packageContext: this, PantallaPerdido.class);
        intent.putExtra( name: "i", i);
        startActivity(intent);
    }
}
```

Si nos damos cuenta cada ves que se direcciona a otra actividad se observa la línea intent.putExtra(). Este método propio de Intent, el cual nos permite enviar el valor de “i” para realizar las respectivas validaciones e incrementos según los aciertos. El valor de la variable i es extraído y guardado en la instancia con el mismo nombre de variable en las demás actividades.

```
intent.putExtra( name: "i", i);
startActivity(intent);
```

```
i = getIntent().getIntExtra( name: "i", defaultValue: 0);
```

-Pregunta1xml, Pregunta2.xml, Pregunta3.xml, Pregunta4.xml, Pregunta5.xml

De igual forma que su respectivo archivo java, los archivos xml de las preguntas tienen un formato similar cuyo valor de cada view cambia, pero su estructura no. El archivo xml de las preguntas está conformado por un TextView, un ImageView y 4 botones. El ImageView contendrá la imagen relacionada a la pregunta del TextView; y los botones contendrá las posibles respuestas. Todos los botones implementan onClick el cual nos direcciona al método anteriormente descrito en su archivo java “bnt_Pregunta”. Cada View es separado de bordes, se le dio estilos de texto, color, tamaño de letra y espaciado respectivo para una mejor visualización.

```

<ImageView
    android:id="@+id/imageGame"
    android:layout_width="match_parent"
    android:layout_height="200dp"
    android:layout_marginLeft="50dp"
    android:layout_marginRight="50dp"
    android:layout_marginTop="50dp"
    app:srcCompat="@drawable/titanic" />

<TextView
    android:id="@+id/textPregunta"
    android:layout_width="match_parent"
    android:layout_marginLeft="50dp"
    android:layout_marginRight="50dp"
    android:layout_marginTop="30dp"
    android:textStyle="italic"
    android:textAlignment="center"
    android:textSize="20dp"
    android:layout_height="52dp"
    android:text="Qual es el actor principal?"/>

<Button
    android:id="@+id/btnAsk_1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="50dp"
    android:layout_marginRight="50dp"
    android:layout_marginTop="15dp"
    android:text="Mario Cansas"
    android:onClick="btn_Pregunta" />

<Button
    android:id="@+id/btnAsk_2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="50dp"
    android:layout_marginRight="50dp"
    android:layout_marginTop="15dp"
    android:text="Liam Neson"
    android:onClick="btn_Pregunta"/>

<Button
    android:id="@+id/btnAsk_3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="50dp"
    android:layout_marginRight="50dp"
    android:layout_marginTop="15dp"
    android:text="Will Smith"
    android:onClick="btn_Pregunta"/>

<Button
    android:id="@+id/btnAsk_4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="50dp"
    android:layout_marginRight="50dp"
    android:layout_marginTop="15dp"
    android:text="Leonardo DiCarpio"
    android:onClick="btn_Pregunta" />

```

-políticas.java - políticas.xml

La activity políticas se utilizó únicamente la parte del desarrollo de la interfaz grafica para presentar las reglas del juego a través de dos TextView. Esta se inicia al presionar el botón de políticas del menú principal.

```

public class políticas extends AppCompatActivity {

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

<TextView
    android:id="@+id/textView5"
    android:layout_width="match_parent"
    android:layout_height="70dp"
    android:text="Políticas"
    android:layout_marginTop="50dp"
    android:layout_marginRight="20dp"
    android:layout_marginLeft="20dp"
    android:textAlignment="center"
    android:textColor="#33332F"
    android:textSize="25dp"

/>

<TextView
    android:id="@+id/textView6"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginRight="20dp"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="50dp"
    android:text="-) Solo tendrá un intento -) Son 5 preguntas relacionada..."
    android:textSize="20dp"
    android:textStyle="italic"
    android:textColor="#33332F"

```

-pantallaFinal.java pantallaFinal.xml

En la activity de pantallaFinal se utiliza la interfaz grafica para mostrar el código qr de la entrada que logro ganar el usuario al acertar las 5 preguntas, así como un mensaje e imagen de felicitaciones y un respectivo botón para la salida de la aplicación. De igual forma, se realizó la respectiva modificación de estilos y tamaño de texto, y espaciado. El botón de salir implementa onClick para llamar al método salir del archivo java. Este método permite cerrar la aplicación, el código utilizado permite crear una nueva activity que inicializa la actividad principal, pero antes de que se inicie se crea un flag que elimina la todas las activities, siendo este método mas complejo ya que el anterior (finish()) se tenia problemas para cerrar completamente la aplicación.

```

public void salir(View view) {
    Intent intent = new Intent(Intent.ACTION_MAIN);
    intent.addCategory(Intent.CATEGORY_HOME);
    intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
    startActivity(intent);
}

```

```

<TextView
    android:id="@+id/textViewGanado"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="50dp"
    android:layout_marginRight="20dp"
    android:text="HA GANADO UNA ENTRADA AL CINE"
    android:textAlignment="center"
    android:textColor="#292623"
    android:textSize="20dp" />

<ImageView
    android:id="@+id/imageView"
    android:layout_width="match_parent"
    android:layout_height="177dp"
    app:srcCompat="@drawable/felicidades" />

<ImageView
    android:id="@+id/imageView2"
    android:layout_width="match_parent"
    android:layout_height="204dp"
    app:srcCompat="@drawable/qr" />

<Button
    android:id="@+id/btnMainScreen"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:onClick="salir"
    android:text="Salir" />

```

-PantallaPerdido.java

De igual forma que las demás activities descritas, en esta se utiliza el método `otraActivity` para direccionar a las demás activities de pregunta y la extracción del dato de “i” para la respectiva validación. En esta parte se implementa el método `salir` (anteriormente descrito en la activity `pantallaFinal`) y el método `volver a jugar` que nos redirecciona a una activity de pregunta de forma random con la implementación del método `otraActivity` y se instancia el valor de `i` en cero, ya que el juego se reinicia, debe empezar desde el inicio para poder ganar el premio. También se instancia el valor del `textView3` que muestra el mensaje de cuantas ha podido acertar antes de haber fallado la pregunta.

```

Button btnJugar;
Button btnSalir;
TextView textView3;
int i;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_pantalla_perdido);
    btnJugar = (Button) findViewById(R.id.btnVolverJugar);
    btnSalir = (Button) findViewById(R.id.btnSalir);
    textView3 = (TextView) findViewById(R.id.textView3);

    i = getIntent().getIntentExtra( name: "i", defaultValue: 0);
    textView3.setText("Has logrado alcanzar "+i+ " aciertos");
}

public void salir(View view) {
    Intent intent = new Intent(Intent.ACTION_MAIN);
    intent.addCategory(Intent.CATEGORY_HOME);
    intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
    startActivity(intent);
}

public void volverJugar(View view) {
    i=0;
    Random rand = new Random();
    int n = rand.nextInt( bound: 4)+1;
    otraActivity(n);
}

public void otraActivity(int n){

```

-PantallaPerdido.xml

En el archivo xml se muestra el desarrollo de la interfaz gráfica, la cual está compuesta por dos TextView, que muestran un mensaje de consolación y otro de cuantas preguntas acertamos; un ImageView que contiene la imagen de una cara triste; y dos botones que implementan el método volverJugar y salir del archivo respectivo java a través de onClick. Todas las view son separadas de los márgenes y espaciadas de manera proporcional con el fin de dar una buena presentación de la interfaz.

```
<ImageView
    android:id="@+id/imageView3"
    android:layout_width="match_parent"
    android:layout_height="211dp"
    android:layout_marginTop="50dp"
    app:srcCompat="@drawable/triste" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="match_parent"
    android:layout_height="42dp"
    android:text="Lo siento has perdido"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:textAlignment="center"
    android:textSize="30dp"
    android:textColor="#33332F"/>

<TextView
    android:id="@+id/textView3"
    android:layout_width="match_parent"
    android:layout_height="58dp"
    android:text="Has respondido una pregunta mal"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:textAlignment="center"
    android:textSize="15dp"
    android:textColor="#33332F"/>

<Button
    android:id="@+id/btnVolverJugar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:onClick="volverJugar"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:text="Volver a jugar" />

<Button
    android:id="@+id/btnSalir"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    android:onClick="salir"
    android:text="Salir" />
```

Enlace GitHub:

https://github.com/bolisteward/AMST_MegaCine.git