Task ##P/C – Spike: [Core 1 – Software development for mobile devices – COS30017]

Goals:

- This app is a simple one-activity game that makes use of layouts and localisation.
- On completion of this task, I will demonstrate that I am able to work with a single activity app, handle activity states, use Logs, create layouts, implement listeners effectively and enable localisation.

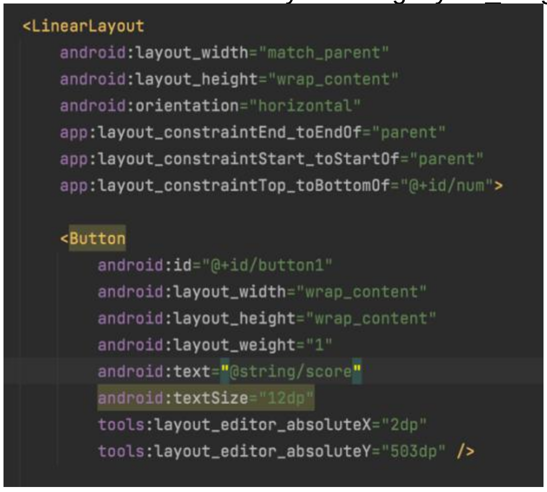
Tools and Resources Used

- Android Studio
- Website google developer
- Github

Knowledge Gaps and Solutions

Gap 1: Be able to create two layouts for the app, portrait, and landscape. Using linear and constrain lay out

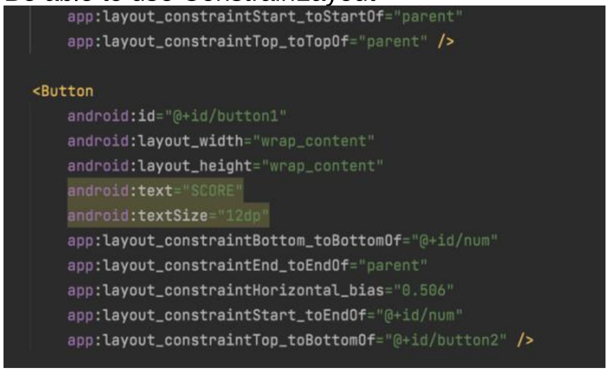
- Be able to add another landscape layout
- Be able to use LinearLayout using layout_weight



```xml
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/num">

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="@string/score"
        android:textSize="12dp"
        tools:layout_editor_absoluteX="2dp"
        tools:layout_editor_absoluteY="503dp" />
```

-
- Be able to use ConstrainLayout

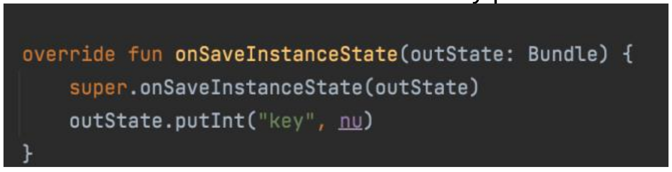


```xml
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="SCORE"
        android:textSize="12dp"
        app:layout_constraintBottom_toBottomOf="@+id/num"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.506"
        app:layout_constraintStart_toEndOf="@+id/num"
        app:layout_constraintTop_toBottomOf="@+id/button2" />
```

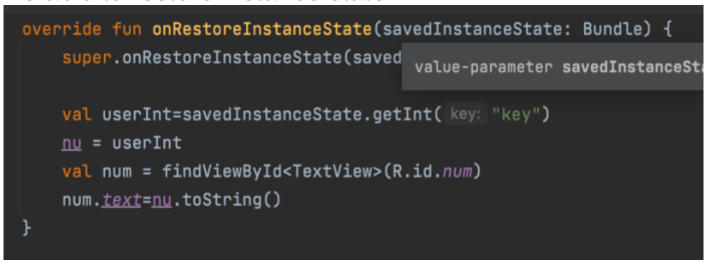
Gap 2: Be able to use saveInstanceState to save the score

- Be able to save instance state with a key pair



```
override fun onSaveInstanceState(outState: Bundle) {
    super.onSaveInstanceState(outState)
    outState.putInt("key", nu)
}
```

- Be able to restore instance state

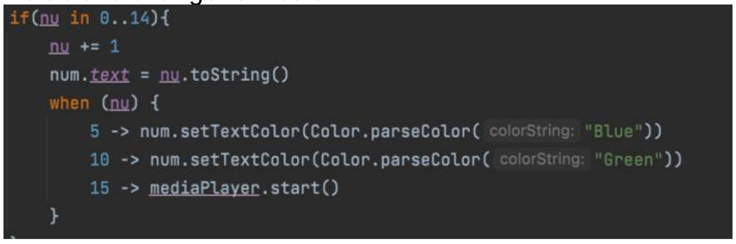


```
override fun onRestoreInstanceState(savedInstanceState: Bundle) {
    super.onRestoreInstanceState(saved  value-parameter savedInstanceSt

    val userInt=savedInstanceState.getInt( key: "key")
    nu = userInt
    val num = findViewById<TextView>(R.id.num)
    num.text=nu.toString()
}
```

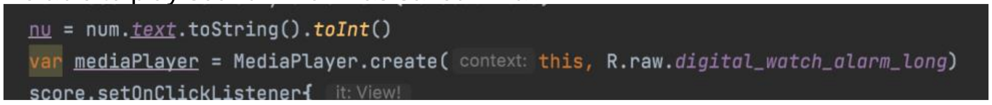
Gap 3: Be able to use when to change text color and play sound

- Be able to use when for different cases.
- Be able to change text color



```
if(nu in 0..14){
    nu += 1
    num.text = nu.toString()
    when (nu) {
        5 -> num.setTextColor(Color.parseColor( colorString: "Blue"))
        10 -> num.setTextColor(Color.parseColor( colorString: "Green"))
        15 -> mediaPlayer.start()
    }
```

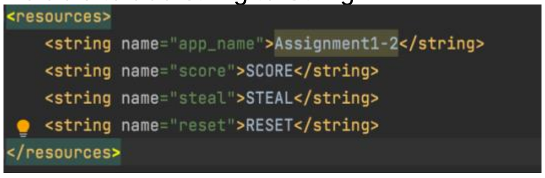
- Be able to play sound which was saved in raw



```
nu = num.text.toString().toInt()
var mediaPlayer = MediaPlayer.create( context: this, R.raw.digital_watch_alarm_long)
score.setOnClickListener{  it: View!
```

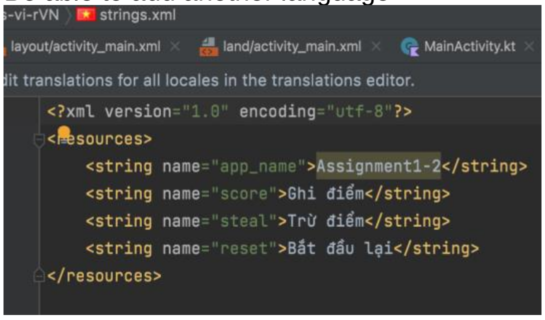
Gap 4: Be able to add a language to the project

- Be able to add string to String.xml



```
<resources>
    <string name="app_name">Assignment1-2</string>
    <string name="score">SCORE</string>
    <string name="steal">STEAL</string>
    <string name="reset">RESET</string>
</resources>
```

- Be able to add another language



```
s-vi-rVN 〉 strings.xml
layout/activity_main.xml ×   land/activity_main.xml ×   MainActivity.kt ×
dit translations for all locales in the translations editor.
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">Assignment1-2</string>
    <string name="score">Ghi điểm</string>
    <string name="steal">Trừ điểm</string>
    <string name="reset">Bắt đầu lại</string>
</resources>
```

Gap 5: Be able to use Log

A log is a record of what has happened. Typically it helps to diagnose problems or get certain insights on what is going on in an application's life cycle.
When rotating the screen, I use logs to check and realize that the activity is destroyed, and then recreated. And since, I can check whether the point is saved using instance state.

Open Issues and Recommendations

This section outlines any open issues, risks, and/or bugs, and highlights potential approaches for trying to address them in the future.

…