



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИИТ)

Кафедра математического обеспечения и стандартизации ИТ

ОТЧЁТ ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ №4

по дисциплине

«Разработка мобильных приложений»

Отчет представлен к

рассмотрению:

Студенты группы ИНБО-04-20

«28» февраля 2022 г.

(подпись)

Ло В.Х.

Преподаватель

« »

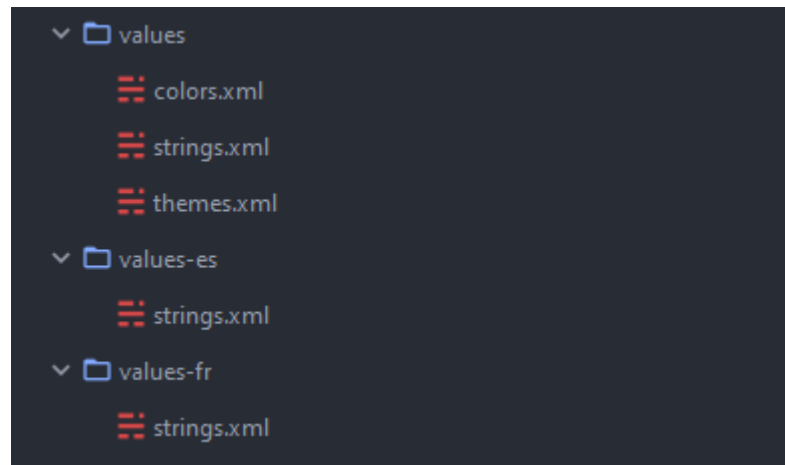
2022 г.

(подпись)

Фандеев И.И.

Москва, 2022г.

1. Поддержка различных языков. Создать директории для поддержки различных языков.



```
1  <!--English-->
2  <resources>
3      <string name="title">My Application</string>
4      <string name="hello_world">Hello World!</string>
5  </resources>
```

```
1  <!--Russian-->
2  <?xml version="1.0" encoding="utf-8"?>
3  <resources>
4      <string name="title"><span lang="ru_RU">Мое приложение</span></string>
5      <string name="hello_world"><span lang="ru_RU">Привет мир</span>!</string>
6  </resources>
```

```
1  <!--French-->
2  <?xml version="1.0" encoding="utf-8"?>
3  <resources>
4      <string name="title">Mon Application</string>
5      <string name="hello_world">Bonjour le monde !</string>
6  </resources>
```

Обеспечить использование строковых ресурсов.

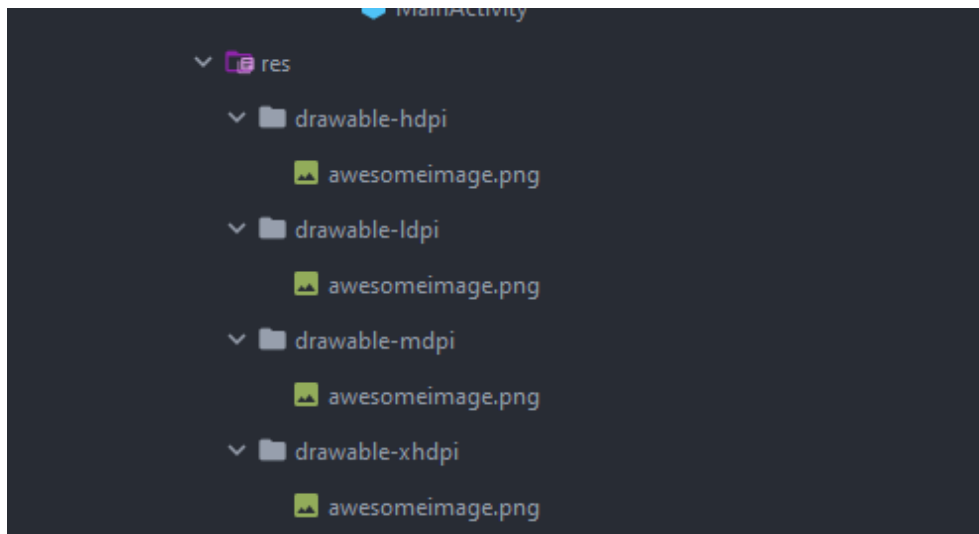
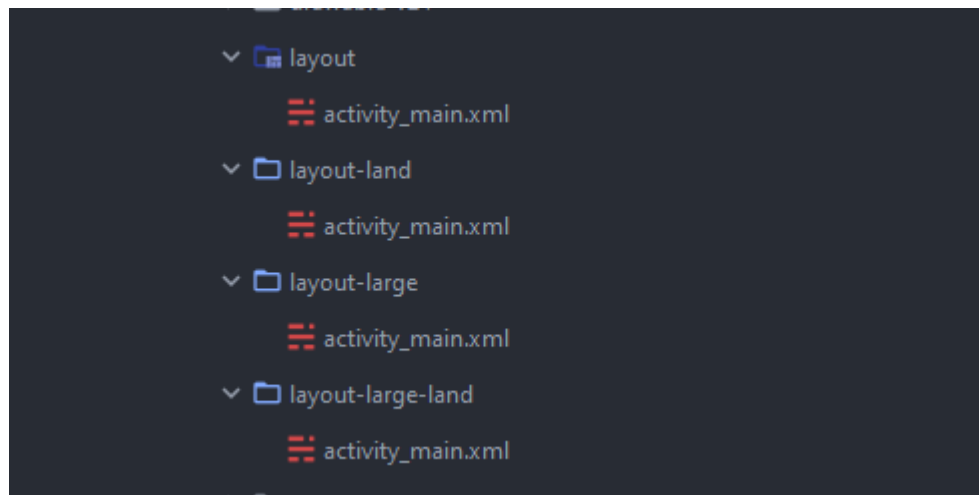
Использование строковых ресурсов в файле activity_main.xml.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.cc
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".MainActivity">
8
9      <TextView
10          android:layout_width="wrap_content"
11          android:layout_height="wrap_content"
12          android:text="@string/hello_world"
13          app:layout_constraintBottom_toBottomOf="parent"
14          app:layout_constraintLeft_toLeftOf="parent"
15          app:layout_constraintRight_toRightOf="parent"
16          app:layout_constraintTop_toTopOf="parent" />
17
18  </androidx.constraintlayout.widget.ConstraintLayout>
```

Использование строковых ресурсов в файле MainActivity .

```
1  package com.example.my_fouth_application;
2
3  import ...
4
5
6
7
8  <> public class MainActivity extends AppCompatActivity {
9
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         setContentView(R.layout.activity_main);
14
15         TextView textView = new TextView( context: this);
16         textView.setText(R.string.hello_world);
17     }
18 }
```

2. Поддержка устройств с различными экранами. Обеспечить создание различной разметки. Обеспечить использование различных изображений.



3. Поддержка различных версий Android. Указать минимальную и целевую версии API. Обеспечить получение версии Android во время выполнения приложения.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="com.example.my_fouth_application">
4      <uses-sdk android:minSdkVersion="4" android:maxSdkVersion="31" />
5
6      <application
7          android:allowBackup="true"
8          android:icon="@mipmap/ic_launcher"
9          android:roundIcon="@mipmap/ic_launcher_round"
10         android:supportsRtl="true"
11         android:theme="@style/Theme.My_fouth_application">
12         <activity
13             android:name=".MainActivity"
14             android:exported="true">
15             <intent-filter>
16                 <action android:name="android.intent.action.MAIN" />
17
18                 <category android:name="android.intent.category.LAUNCHER" />
19             </intent-filter>
20         </activity>
21     </application>
22
23 </manifest>

```

4. Обеспечить использование встроенных тем и стилей

```

<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.AppCompat.DayNight.Dialog">

    <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>
</application>

```

5. Жизненный цикл явлений. Обеспечить запуск явлений. Обеспечить функции обратного вызова жизненного цикла. Обеспечить объявление главного явления. Обеспечить создание экземпляра явления. Обеспечить приостановку и возобновление явлений. Обеспечить остановку, запуск и перезапуск явлений. Обеспечить пересоздание явлений. Сохранить состояния экземпляра явлений. Восстановить состояния экземпляра явлений.

Обеспечить объявление главного явления.

```
<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>
```

Обеспечить создание экземпляра явления.

```
1  package com.example.my_fouth_application;
2
3  import ...
10
11  <? public class MainActivity extends AppCompatActivity {
12      TextView mTextView;
13      @RequiresApi(api = Build.VERSION_CODES.ICE_CREAM_SANDWICH)
14      @Override
15      protected void onCreate(Bundle savedInstanceState) {
16          super.onCreate(savedInstanceState);
17          setContentView(R.layout.activity_main);
18
19          mTextView = (TextView) findViewById(R.id.text_message);
20          if (Build.VERSION.SDK_INT ≥ Build.VERSION_CODES.HONEYCOMB){
21              ActionBar actionBar = getActionBar();
22              actionBar.setHomeButtonEnabled(false);
23          }
24
25      }
26  }
```

Обеспечить приостановку и возобновление явлений.

```
private void release(){
}

public void onPause(){
    super.onPause();

    if(mCamera != null){
        mCamera.release();
        mCamera = null;
    }
}

public void onResume(){
    super.onResume();
    if(mCamera != null){
        initializeCamera();
    }
}
```

Обеспечить остановку, запуск и перезапуск явлений.

```
protected void onStop(){
    super.onStop();
    ContentValues values = new ContentValues();
    values.put(NotePad.Notes.COLUMN_NAME_NOTE,getCurrentNoteText());
    values.put(NotePad.Notes.COLUMN_NAME_TITLE,getCurrentNoteText());
    Uri mUri;
    getContentResolver().update(mUri,values, where: null, selectionArgs: null);
}

protected void onStart(){
    super.onStart();
    LocationManager locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
    boolean gpsEnabled = locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER);
    if(!gpsEnabled){

    }
}

protected void onRestart(){
    super.onRestart();
}
```

Сохранить состояния экземпляра явлений.

```

static final String STATE_SCORE = "playerScore";
static final String STATE_LEVER = "playerLever";

public void onSaveInstanceState(Bundle savedInstanceState) {

    super.onSaveInstanceState(savedInstanceState);
    savedInstanceState.putInt(STATE_SCORE, mCurrentScore);
    savedInstanceState.putInt(STATE_LEVER, mCurrentLever);
}

```

Восстановить состояния экземпляра явлений.

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (savedInstanceState  $\neq$  null) {
        mCurrentScore = savedInstanceState.getInt(STATE_SCORE);
        mCurrentLever = savedInstanceState.getInt(STATE_LEVER);
    } else {

    }

}

public void onRestoreInstanceState(Bundle savedInstanceState) {
    super.onRestoreInstanceState(savedInstanceState);
    mCurrentScore = savedInstanceState.getInt(STATE_SCORE);
    mCurrentLever = savedInstanceState.getInt(STATE_LEVER);
}

```