**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное автономное образовательное учреждение

высшего образования «Уральский федеральный университет

имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий - РТФ

Центр ускоренного обучения

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ 7**

по дисциплине «Разработка мобильных приложений»

**Тема: Unit 1 Начало работы “Неявные намерения”**

Студент группы РИЗ-300028у: А.С. Севостьянов

Преподаватель: С.И.Тимошенко,

доц., к.т.н.

**Екатеринбург 2022**

содержание

[1 Постановка задачи 3](#_Toc118041527)

[2 Результаты проделанной работы 4](#_Toc118041528)

[3 Анализ полученных результатов 5](#_Toc118041529)

## 1 Постановка задачи

Создать новое приложение, чтобы поэкспериментировать с implicit Intent. Реализовать неявное Intent, которое открывает веб-страницу, и другое, которое открывает местоположение на карте. Реализовать действие для совместного использования фрагмента текста. Создать новое приложение, которое может принимать неявное Intent значение для открытия веб-страницы.

.

## Результаты проделанной работы

Программный код проекта файл, проекта MainActivity.

package com.example.android.twoactivities;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.support.v7.app.AppCompatActivity;  
import android.util.Log;  
import android.view.View;  
import android.widget.EditText;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
 // Логирование  
 private static final String LOG\_TAG = MainActivity.class.getSimpleName();  
 // Уникальный тег, необходимый для намерения дополнительно  
 public static final String EXTRA\_MESSAGE  
 = "com.example.android.twoactivities.extra.MESSAGE";  
 // Уникальный тег для ответа с намерением  
 public static final int TEXT\_REQUEST = 1;  
  
 // Редактировать текстовое представление сообщения  
 private EditText mMessageEditText;  
 // TextView для заголовка ответа  
 private TextView mReplyHeadTextView;  
 // TextView для тела ответа  
 private TextView mReplyTextView;  
  
 @Override  
 protected void onStart() {  
 super.onStart();  
 Log.d(LOG\_TAG, "onStart");  
 }  
  
  
 @Override  
 public void onSaveInstanceState(Bundle outState) {  
 super.onSaveInstanceState(outState);  
 if (mReplyHeadTextView.getVisibility() == View.VISIBLE) {  
 outState.putBoolean("reply\_visible", true);  
 outState.putString("reply\_text", mReplyTextView.getText().toString());  
 }  
 }  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 // Регистрируйте запуск метода onCreate().  
 Log.d(LOG\_TAG, "-------");  
 Log.d(LOG\_TAG, "onCreate");  
  
 // Инициализируйте все переменные представления.  
 mMessageEditText = findViewById(R.id.editText\_main);  
 mReplyHeadTextView = findViewById(R.id.text\_header\_reply);  
 mReplyTextView = findViewById(R.id.text\_message\_reply);  
  
 // Восстановите состояние.  
 if (savedInstanceState != null) {  
 boolean isVisible = savedInstanceState  
 .getBoolean("reply\_visible");  
 if (isVisible) {  
 mReplyHeadTextView.setVisibility(View.VISIBLE);  
 mReplyTextView.setText(savedInstanceState  
 .getString("reply\_text"));  
 mReplyTextView.setVisibility(View.VISIBLE);  
 }  
 }  
 }  
  
 */\*\*  
 Обрабатывает повторный щелчок по кнопке "Отправить". Получает значение основного  
 Редактирует текст, создает намерение и запускает второе действие с  
 это намерение. Возвращаемое намерение из второго действия - onActivityResult().  
  
 \*/* public void launchSecondActivity(View view) {  
 Log.d(LOG\_TAG, "Button clicked!");  
 Intent intent = new Intent(this, SecondActivity.class);  
 String message = mMessageEditText.getText().toString();  
 intent.putExtra(EXTRA\_MESSAGE, message);  
 startActivityForResult(intent, TEXT\_REQUEST);  
 }  
  
 */\*\*  
 Обрабатывает данные в возвращаемом намерении из SecondActivity.  
 \*/* @Override  
 public void onActivityResult(int requestCode, int resultCode, Intent data) {  
 super.onActivityResult(requestCode, resultCode, data);  
  
 // Проверьте правильность ответа с намерением.  
 if (requestCode == TEXT\_REQUEST) {  
 //Проверьте, чтобы убедиться, что результат намеренного ответа был хорошим.  
 if (resultCode == RESULT\_OK) {  
 String reply = data.getStringExtra(SecondActivity.EXTRA\_REPLY);  
  
 //Заголовок ответа видимым.  
 mReplyHeadTextView.setVisibility(View.VISIBLE);  
  
 // Установите ответ и сделайте его видимым.  
 mReplyTextView.setText(reply);  
 mReplyTextView.setVisibility(View.VISIBLE);  
 }  
 }  
 }  
  
 @Override  
 protected void onPause() {  
 super.onPause();  
 Log.d(LOG\_TAG, "onPause");  
 }  
  
 @Override  
 protected void onRestart() {  
 super.onRestart();  
 Log.d(LOG\_TAG, "onRestart");  
 }  
  
 @Override  
 protected void onResume() {  
 super.onResume();  
 Log.d(LOG\_TAG, "onResume");  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 Log.d(LOG\_TAG, "onStop");  
 }  
  
 @Override  
 protected void onDestroy() {  
 super.onDestroy();  
 Log.d(LOG\_TAG, "onDestroy");  
 }  
}

Программный код проекта файл, проекта SecondActivity.

package com.example.android.twoactivities;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.support.v7.app.AppCompatActivity;  
import android.util.Log;  
import android.view.View;  
import android.widget.EditText;  
import android.widget.TextView;  
  
*/\*\*  
 Второе действие определяет второе действие в приложении. Он запущен  
 из намерения с сообщением и отправляет намерение обратно со вторым  
 сообщение.  
 \*/*public class SecondActivity extends AppCompatActivity {  
 // Логирование.  
 private static final String LOG\_TAG  
 = SecondActivity.class.getSimpleName();  
 // Уникальный тег для ответа с намерением.  
 public static final String EXTRA\_REPLY =  
 "com.example.android.twoactivities.extra.REPLY";  
  
 // Отредактируйте текст для ответа.  
 private EditText mReply;  
  
 @Override  
 protected void onStart() {  
 super.onStart();  
 Log.d(LOG\_TAG, "onStart");  
 }  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_second);  
  
 // Инициализировать переменные представления.  
 mReply = findViewById(R.id.editText\_second);  
  
 // Получить намерение, которое запустило это действие, и сообщение в  
 // намерение дополнительное.  
 Intent intent = getIntent();  
 String message = intent.getStringExtra(MainActivity.EXTRA\_MESSAGE);  
  
 // Поместите это сообщение в text\_message TextView.  
 TextView textView = findViewById(R.id.text\_message);  
 textView.setText(message);  
 }  
  
 */\*\*  
 Обрабатывает повторный щелчок по кнопке "Ответить". Получает сообщение из  
 второго EditText, создает намерение и возвращает это сообщение обратно к  
 основному действию.  
  
 \*/* public void returnReply(View view) {  
 // Получите ответное сообщение из отредактированного текста.  
 String reply = mReply.getText().toString();  
  
 // Создайте новое намерение для ответа, добавьте к нему ответное сообщение  
 // в качестве дополнительного, установите результат намерения и закройте действие.  
 Intent replyIntent = new Intent();  
 replyIntent.putExtra(EXTRA\_REPLY, reply);  
 setResult(RESULT\_OK, replyIntent);  
 Log.d(LOG\_TAG, "End SecondActivity");  
 finish();  
 }  
  
 @Override  
 protected void onPause() {  
 super.onPause();  
 Log.d(LOG\_TAG, "onPause");  
 }  
  
 @Override  
 protected void onRestart() {  
 super.onRestart();  
 Log.d(LOG\_TAG, "onRestart");  
 }  
  
 @Override  
 protected void onResume() {  
 super.onResume();  
 Log.d(LOG\_TAG, "onResume");  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 Log.d(LOG\_TAG, "onStop");  
 }  
  
 @Override  
 protected void onDestroy() {  
 super.onDestroy();  
 Log.d(LOG\_TAG, "onDestroy");  
 }  
}

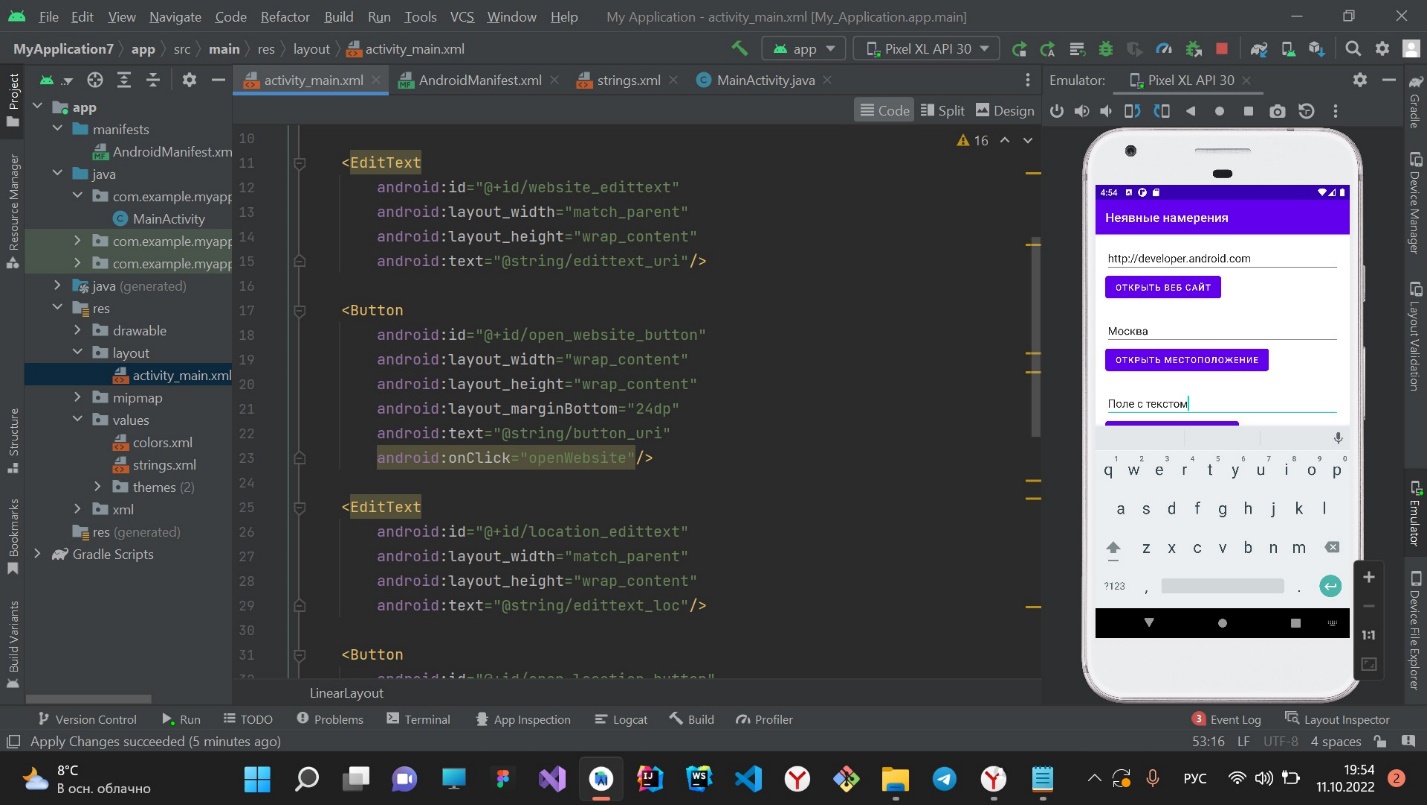
На рисунке 1 показан результат выполненной работы.

Рисунок 1 – Результат выполненой работы

## **Анализ полученных результатов**

В результате выполнения данной лабораторной работы мной был разработан проект, в котором я научился создавать неявное Intent и использовать его действия и категории.

Как использовать ShareCompat. вспомогательный класс для создания неявного Intent для обмена данными.

Как объявить, что ваше приложение может принимать неявное Intent значение, объявив Intent фильтры в AndriodManifest.xml файле.