

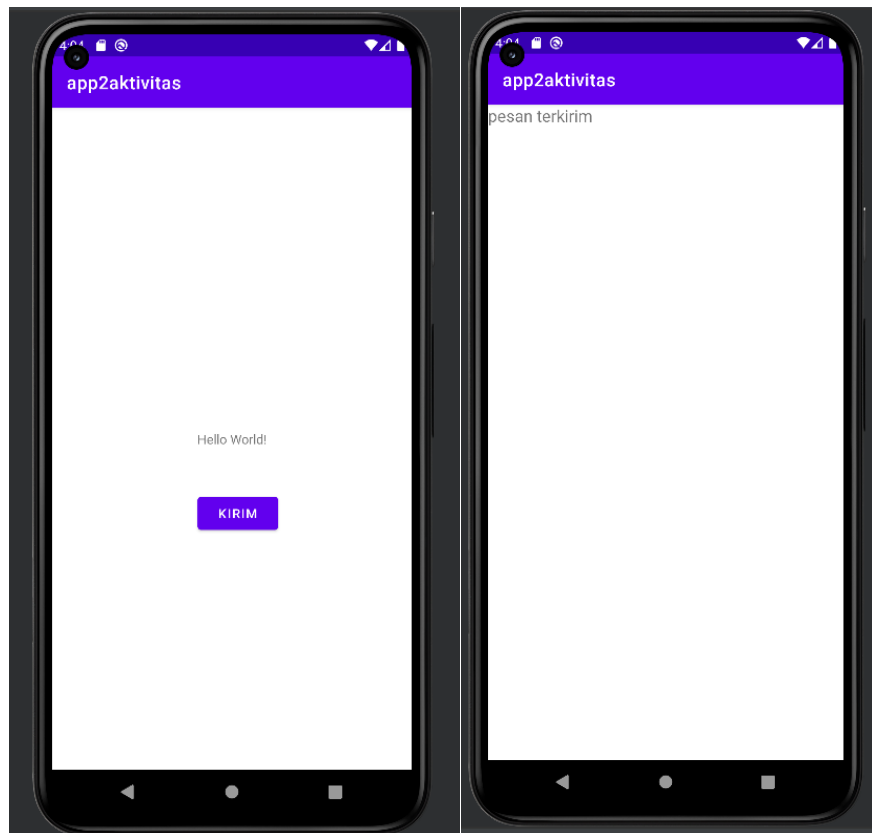
**Mata Kuliah : Pemrograman Mobile – TI – S1**

**Pertemuan : Minggu 2**

**NIM : A11.2021.13509**

**Nama : Ainindzi Nur Meiza Pudjianto**

## 1. App 2 Aktivitas



- MainActivity.java

```
package com.example.app2aktivitas;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

```

    }

    public void lihat(View view) {
        Intent intent = new Intent(MainActivity.this, MainActivity2.class);
        startActivity(intent);
    }
}

```

- MainActivity2.java

```

package com.example.app2aktivitas;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity2 extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
    }
}

```

- Activity\_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="159dp"
        android:layout_marginTop="419dp"
        android:layout_marginBottom="264dp"
        android:onClick="lihat"
        android:text=" kirim"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"

```

```

        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

- Activity\_main2.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="159dp"
        android:layout_marginTop="419dp"
        android:layout_marginBottom="264dp"
        android:onClick="lihat"
        android:text="kirim"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

- Colors.xml

```

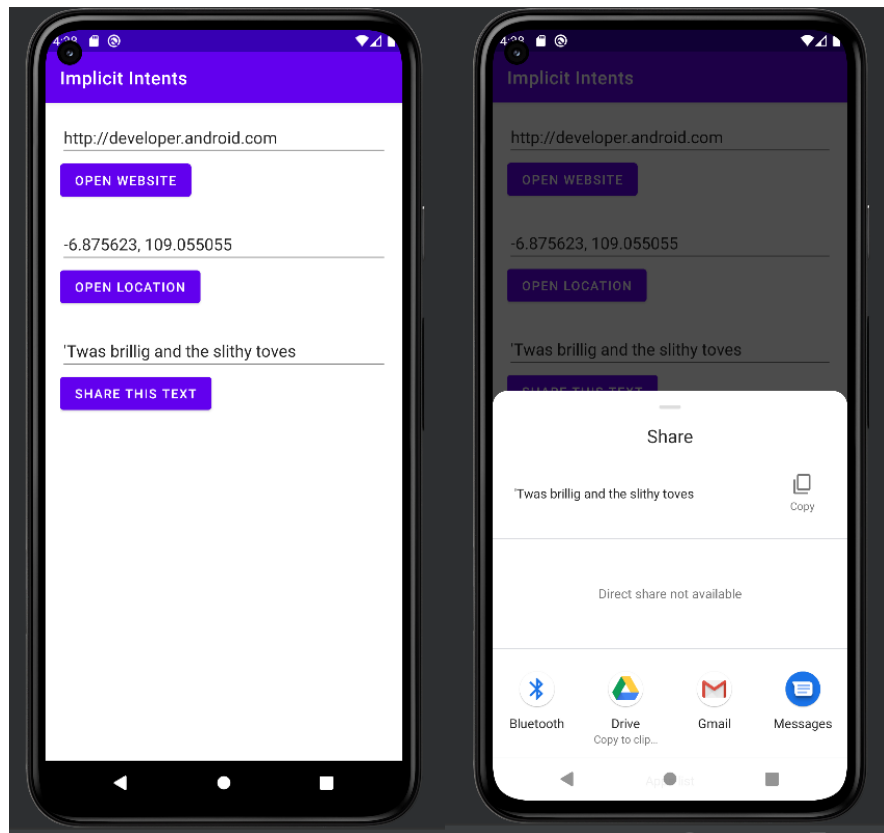
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
</resources>

```

- Strings.xml

```
<resources>
    <string name="app_name">app2aktivitas</string>
    <string name="text_header">pesan terkirim</string>
</resources>
```

## 2. App Implicit Intents



- MainActivity.java

```
package com.example.implicitintents;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ShareCompat;

public class MainActivity extends AppCompatActivity {
```

```

// EditText view for the website URI
private EditText mWebsiteEditText;
// EditText view for the location URI
private EditText mLocationEditText;
// EditText view for the share text
private EditText mShareTextEditText;
/**
 * Initializes the activity.
 *
 * @param savedInstanceState The current state data
 */
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    mWebsiteEditText = (EditText) findViewById(R.id.website_edittext);
    mLocationEditText = (EditText) findViewById(R.id.location_edittext);
    mShareTextEditText = (EditText) findViewById(R.id.share_edittext);
}
/**
 * Handles the onClick for the "Open Website" button. Gets the URI
 * from the edit text and sends an implicit intent for that URL.
 *
 * @param view The view (Button) that was clicked.
 */
public void openWebsite(View view) {
    // Get the URL text.
    String url = mWebsiteEditText.getText().toString();
    // Parse the URI and create the intent.
    Uri webpage = Uri.parse(url);
    Intent intent = new Intent(Intent.ACTION_VIEW, webpage);
    // Find an activity to hand the intent and start that activity.
    if (intent.resolveActivity(getPackageManager()) != null) {
        startActivity(intent);
    } else {
        Log.d("ImplicitIntents", "Can't handle this intent!");
    }
}
/**
 * Handles the onClick for the "Open Location" button. Gets the
 * location
 * text from the edit text and sends an implicit intent for that
 * location.
 *
 * The location text can be any searchable geographic location.
 *
 * @param view The view (Button) that was clicked.
 */
public void openLocation(View view) {
    // Get the string indicating a location. Input is not validated it is
    // passed to the location handler intact.
    String loc = mLocationEditText.getText().toString();
    // Parse the location and create the intent.
    Uri addressUri = Uri.parse("geo:0,0?q=" + loc);
    Intent intent = new Intent(Intent.ACTION_VIEW, addressUri);
    // Find an activity to handle the intent, and start that activity.
    if (intent.resolveActivity(getPackageManager()) != null) {

```

```

        startActivity(intent);
    } else {
        Log.d("ImplicitIntents", "Can't handle this intent!");
    }
}

/**
 * Handles the onClick for the "Share This Text" button. The
 * implicit intent here is created by the
 * {@link ShareCompat.IntentBuilder}
 * class. An app chooser appears with the available options for
 * sharing.
 *
 * ShareCompat.IntentBuilder is from the v4 Support Library.
 *
 * @param view The view (Button) that was clicked.
 */
public void shareText(View view) {
    // Get the shared text.
    String txt = mShareTextEditText.getText().toString();
    // Build the share intent with the mimetype text/plain and launch
    // a chooser for the user to pick an app.
    ShareCompat.IntentBuilder
        .from(this)
        .setType("text/plain")
        .setChooserTitle("Share this text with: ")
        .setText(txt)
        .startChooser();
}
}

```

- Activity\_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:paddingBottom="@dimen/activity_vertical_margin"
        android:paddingLeft="@dimen/activity_horizontal_margin"
        android:paddingRight="@dimen/activity_horizontal_margin"
        android:paddingTop="@dimen/activity_vertical_margin"
        tools:context="com.example.android.implicitintents.MainActivity"
        android:orientation="vertical">
        <EditText
            android:id="@+id/website_edittext"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/edittext_uri" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

```

```

        <Button
            android:id="@+id/open_website_button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="24dp"
            android:onClick="openWebsite"
            android:text="@string/button_uri" />

        <EditText
            android:id="@+id/location_edittext"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/edittext_loc" />

        <Button
            android:id="@+id/open_location_button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="24dp"
            android:onClick="openLocation"
            android:text="@string/button_loc" />

        <EditText
            android:id="@+id/share_edittext"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/edittext_share" />

        <Button
            android:id="@+id/share_text_button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="24dp"
            android:onClick="shareText"
            android:text="@string/button_share" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

```

- Colors.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#FF4081</color>
</resources>

```

- Dimens.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>

```

```

    <!-- Default screen margins, per the Android Design guidelines. -->
    <dimen name="activity_horizontal_margin">16dp</dimen>
    <dimen name="activity_vertical_margin">16dp</dimen>
</resources>

```

- Strings.xml

```

<resources>
    <!-- Title of app -->
    <string name="app_name">Implicit Intents</string>
    <!-- Default text for Open Website -->
    <string name="edittext_uri">http://developer.android.com</string>
    <!-- Button label for Open Website [CHAR LIMIT=20]-->
    <string name="button_uri">Open Website</string>
    <!-- Default text for Open Location [CHAR LIMIT=30]-->
    <string name="edittext_loc">-6.875623, 109.055055</string>
    <!-- Button label for Open Location [CHAR LIMIT=20]-->
    <string name="button_loc">Open Location</string>
    <!-- Default text for Share This Text [CHAR LIMIT=NONE]-->
    <string name="edittext_share">\'Twas brillig and the slithy
toves</string>
    <!-- Button label for Share This Text [CHAR LIMIT=20]-->
    <string name="button_share">Share This Text</string>
</resources>

```

- Styles.xml

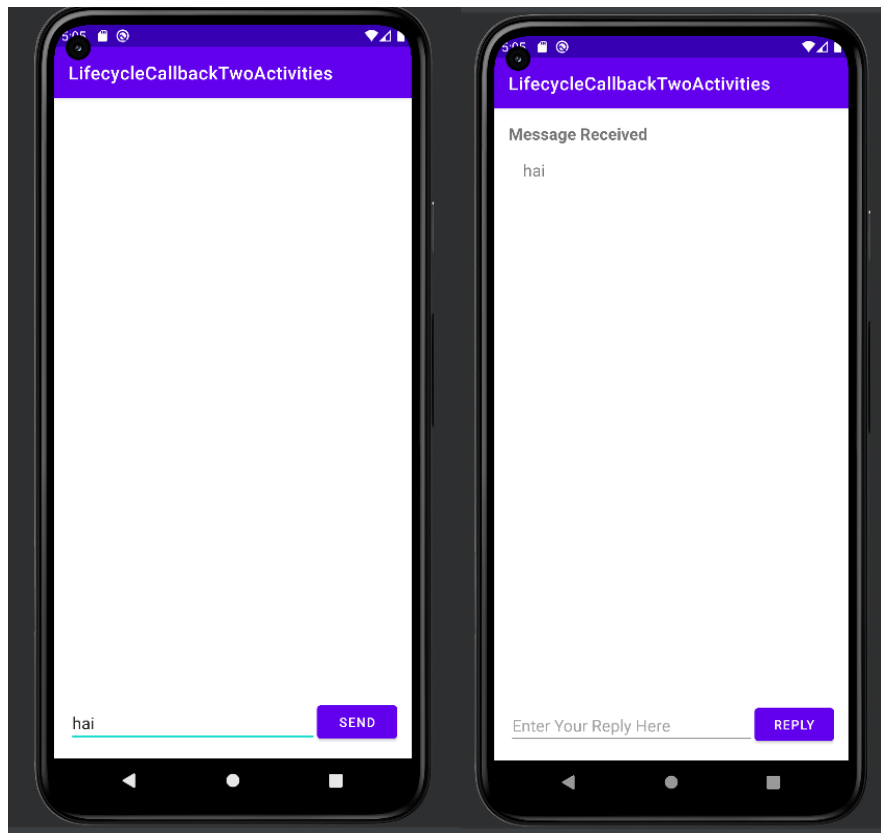
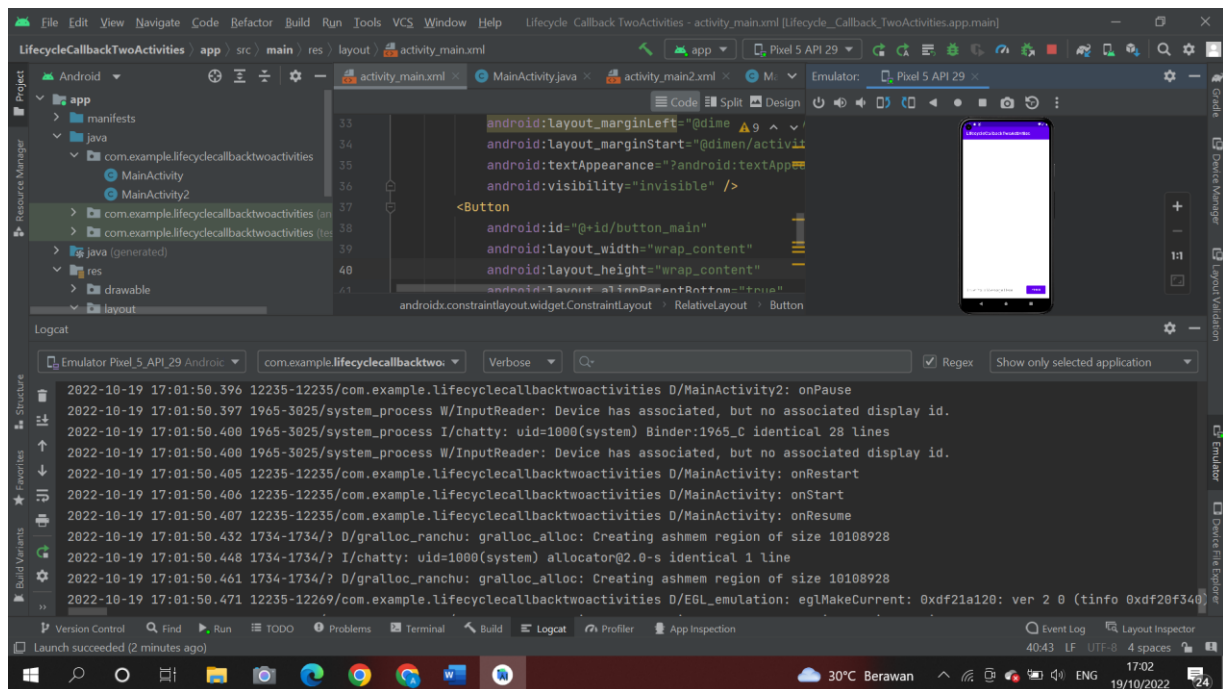
```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>
</resources>

```



### 3. Aplikasi Lifecycle Callback TwoActivities



- MainActivity.java

```

package com.example.lifecyclecallbacktwoactivities;

import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    // Class name for Log tag
    private static final String LOG_TAG =
        MainActivity.class.getSimpleName();
    // Unique tag required for the intent extra
    public static final String EXTRA_MESSAGE =
        "com.example.android.twoactivitieslifecycle.extra.MESSAGE";
    // Unique tag for the intent reply
    public static final int TEXT_REQUEST = 1;
    // EditText view for the message
    private EditText mMessageEditText;
    // TextView for the reply header
    private TextView mReplyHeadTextView;
    // TextView for the reply body
    private TextView mReplyTextView;
    /**
     * Initializes the activity.
     *
     * @param savedInstanceState The current state data.
     */
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Log.d(LOG_TAG, "-----");
        Log.d(LOG_TAG, "onCreate");
        // Initialize all the view variables.
        mMessageEditText = (EditText) findViewById(R.id.editText_main);
        mReplyHeadTextView = (TextView)
            findViewById(R.id.text_header_reply);
        mReplyTextView = (TextView) findViewById(R.id.text_message_reply);
        // Restore the saved state. See onSaveInstanceState() for what gets saved.
        if (savedInstanceState != null) {
            boolean isVisible =
                savedInstanceState.getBoolean("reply_visible");
            // Show both the header and the message views. If isVisible is
            // false or missing from the bundle, use the default layout.
            if (isVisible) {
                mReplyHeadTextView.setVisibility(View.VISIBLE);
                mReplyTextView.setText(savedInstanceState.getString("reply_text"));
                mReplyTextView.setVisibility(View.VISIBLE);
            }
        }
    }
    /**

```

```

    * Handle the onClick for the "Send" button. Gets the value of the main
    EditText,
    * creates an intent, and launches the second activity with that
    intent.
    *
    * The return intent from the second activity is onActivityResult().
    * @param view The view (Button) that was clicked.
    */
public void launchSecondActivity(View view) {
    Log.d(LOG_TAG, "Button clicked!");
    Intent intent = new Intent(this, MainActivity2.class);
    String message = mMessageEditText.getText().toString();
    intent.putExtra(EXTRA_MESSAGE, message);
    startActivityForResult(intent, TEXT_REQUEST);
}

/**
 * Handle the data in the return intent from SecondActivity.
 *
 * @param requestCode Code for the SecondActivity request.
 * @param resultCode Code that comes back from SecondActivity.
 * @param data Intent data sent back from SecondActivity.
 */
@Override
public void onActivityResult(int requestCode, int resultCode, Intent
    data) {
    super.onActivityResult(requestCode, resultCode, data);
// Test for the right intent reply
    if (requestCode == TEXT_REQUEST) {
// Test to make sure the intent reply result was good.
        if (resultCode == RESULT_OK) {
            String reply =
                data.getStringExtra(MainActivity2.EXTRA_REPLY);
// Make the reply head visible.
            mReplyHeadTextView.setVisibility(View.VISIBLE);
// Set the reply and make it visible.
            mReplyTextView.setText(reply);
            mReplyTextView.setVisibility(View.VISIBLE);
        }
    }
}

/**
 * Lifecycle callback for start.
 */
@Override
protected void onStart() {
    super.onStart();
    Log.d(LOG_TAG, "onStart");
}

/**
 * Lifecycle callback for restart.
 */
@Override
public void onRestart() {
    super.onRestart();
    Log.d(LOG_TAG, "onRestart");
}
}

```

```

    * Lifecycle callback for resume.
    */
    @Override
    public void onResume() {
        super.onResume();
        Log.d(LOG_TAG, "onResume");
    }

    /**
     * Lifecycle callback for pause.
     */
    @Override
    public void onPause() {
        super.onPause();
        Log.d(LOG_TAG, "onPause");
    }

    /**
     * Lifecycle callback for stop.
     */
    @Override
    public void onStop() {
        super.onStop();
        Log.d(LOG_TAG, "onStop");
    }

    /**
     * Lifecycle callback for destroy.
     */
    @Override
    public void onDestroy() {
        super.onDestroy();
        Log.d(LOG_TAG, "onDestroy");
    }

    /**
     * Save the instance state if the activity is restarted (for example,
     * on device rotation.)
     * Here you save the contents of the reply text, and whether the reply
     * is visible.
     *
     * @param outState The state data.
     */
    @Override
    protected void onSaveInstanceState(Bundle outState) {
        super.onSaveInstanceState(outState);
        // If the heading is visible, we have a message that needs to be saved.
        // Otherwise we're still using default layout.
        if (mReplyHeadTextView.getVisibility() == View.VISIBLE) {
            outState.putBoolean("reply_visible", true);
            outState.putString("reply_text",
                mReplyTextView.getText().toString());
        }
    }
}

```

- MainActivity2.java

```
package com.example.lifecyclecallbacktwoactivities;
```

```

import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity2 extends AppCompatActivity {
    // Unique tag for the intent reply.
    public static final String EXTRA_REPLY =
        "com.example.android.twoactivitieslifecycle.extra.REPLY";
    // Class name for Log tag.
    private static final String LOG_TAG =
        MainActivity2.class.getSimpleName();
    // EditText for the reply.
    private EditText mReply = null;
    /**
     * Initializes the activity.
     *
     * @param savedInstanceState The current state data
     */
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
    // Initialize view variables.
        mReply = (EditText) findViewById(R.id.editText_second);
    // Get the intent that launched this activity, and the message in
    // the intent extra.
        Intent intent = getIntent();
        String message = intent.getStringExtra(MainActivity.EXTRA_MESSAGE);
    // Put that message into the text_message TextView
        TextView textView = (TextView) findViewById(R.id.text_message);
        if (textView != null) {
            textView.setText(message);
        }
        Log.d(LOG_TAG, "-----");
        Log.d(LOG_TAG, "onCreate");
    }
    /**
     * Handle the onClick for the "Reply" button. Gets the message from the
     * second EditText,
     * creates an intent, and returns that message back to the main
     * activity.
     *
     * @param view The view (Button) that was clicked.
     */
    public void returnReply(View view) {
    // Get the reply message from the edit text.
        String reply = mReply.getText().toString();
    // Create a new intent for the reply, add the reply message to it as an
    // extra,
    // set the intent result, and close the activity.
        Intent replyIntent = new Intent();
        replyIntent.putExtra(EXTRA_REPLY, reply);
    }
}

```

```

        setResult(RESULT_OK, replyIntent);
        Log.d(LOG_TAG, "End SecondaryActivity");
        finish();
    }

    /**
     * Lifecycle callback for start.
     */
    @Override
    protected void onStart() {
        super.onStart();
        Log.d(LOG_TAG, "onStart");
    }

    /**
     * Lifecycle callback for restart.
     */
    @Override
    public void onRestart() {
        super.onRestart();
        Log.d(LOG_TAG, "onRestart");
    }

    /**
     * Lifecycle callback for resume.
     */
    @Override
    public void onResume() {
        super.onResume();
        Log.d(LOG_TAG, "onResume");
    }

    /**
     * Lifecycle callback for pause.
     */
    @Override
    public void onPause() {
        super.onPause();
        Log.d(LOG_TAG, "onPause");
    }

    /**
     * Lifecycle callback for stop.
     */
    @Override
    public void onStop() {
        super.onStop();
        Log.d(LOG_TAG, "onStop");
    }

    /**
     * Lifecycle callback for destroy.
     */
    @Override
    public void onDestroy() {
        super.onDestroy();
        Log.d(LOG_TAG, "onDestroy");
    }
}

```

- Activity\_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <RelativeLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:paddingBottom="@dimen/activity_vertical_margin"
        android:paddingLeft="@dimen/activity_horizontal_margin"
        android:paddingRight="@dimen/activity_horizontal_margin"
        android:paddingTop="@dimen/activity_vertical_margin"
        tools:context=".MainActivity">
        <TextView
            android:id="@+id/text_header_reply"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="@dimen/activity_vertical_margin"
            android:text="@string/text_header_reply"
            android:textAppearance="?android:textAppearanceMedium"
            android:textStyle="bold"
            android:visibility="invisible" />
        <TextView
            android:id="@+id/text_message_reply"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@+id/text_header_reply"
            android:layout_marginLeft="@dimen/activity_horizontal_margin"
            android:layout_marginStart="@dimen/activity_horizontal_margin"
            android:textAppearance="?android:textAppearanceMedium"
            android:visibility="invisible" />
        <Button
            android:id="@+id/button_main"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentBottom="true"
            android:layout_alignParentRight="true"
            android:layout_alignParentEnd="true"
            android:onClick="launchSecondActivity"
            android:text="@string/button_main" />
        <EditText
            android:id="@+id/editText_main"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentBottom="true"
            android:layout_toLeftOf="@+id/button_main"
            android:layout_toStartOf="@+id/button_main"
            android:hint="@string/editText_main" />
    </RelativeLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

```

- Activity\_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity2">
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:paddingBottom="@dimen/activity_vertical_margin"
        android:paddingLeft="@dimen/activity_horizontal_margin"
        android:paddingRight="@dimen/activity_horizontal_margin"
        android:paddingTop="@dimen/activity_vertical_margin"
        tools:context=".SecondActivity">

        <TextView
            android:id="@+id/text_header"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="@dimen/activity_vertical_margin"
            android:text="@string/text_header"
            android:textAppearance="?android:textAppearanceMedium"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/text_message"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@+id/text_header"
            android:layout_marginLeft="@dimen/activity_horizontal_margin"
            android:layout_marginStart="@dimen/activity_horizontal_margin"
            android:textAppearance="?android:textAppearanceMedium" />

        <Button
            android:id="@+id/button_second"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentBottom="true"
            android:layout_alignParentRight="true"
            android:layout_alignParentEnd="true"
            android:onClick="returnReply"
            android:text="@string/button_second" />

        <EditText
            android:id="@+id/editText_second"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentBottom="true"
            android:layout_toLeftOf="@+id/button_second"
            android:layout_toStartOf="@+id/button_second"
            android:hint="@string/editText_second" />

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



- Colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#FF4081</color>
</resources>
```

- Dimens.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources> <!-- Default screen margins, per the Android Design guidelines.
-->
    <dimen name="activity_horizontal_margin">16dp</dimen>
    <dimen name="activity_vertical_margin">16dp</dimen>
</resources>
```

- strings.xml

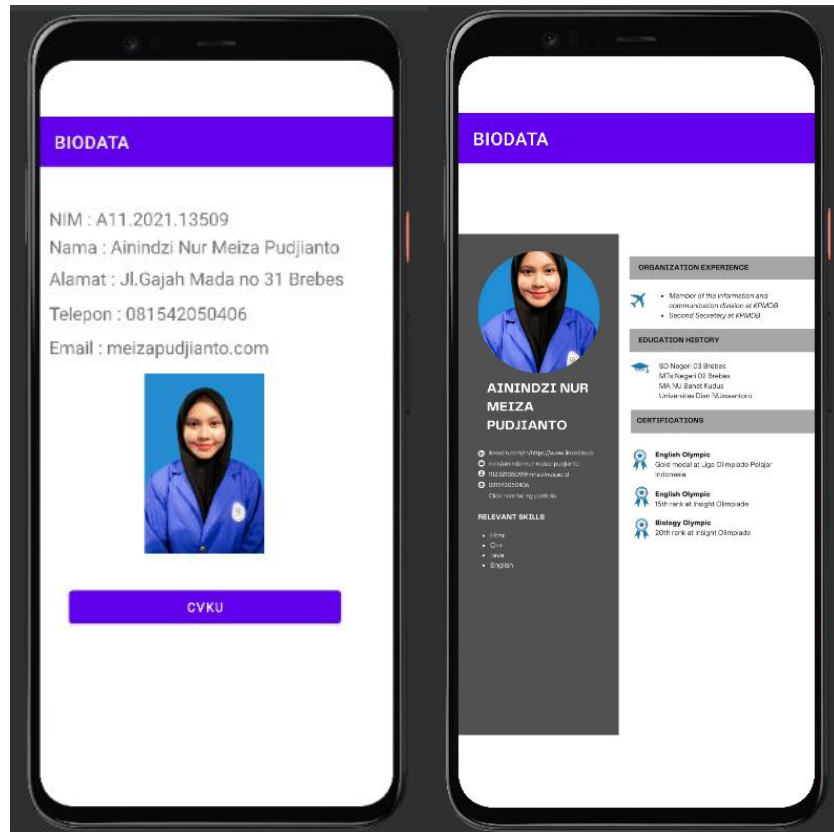
```
<resources>
    <string name="app_name">LifecycleCallbackTwoActivities</string>
    <!-- Title of second activity -->
    <string name="activity2_name">Second Activity</string>
    <!-- Message header text (in second activity) [CHAR LIMIT=30]-->
    <string name="text_header">Message Received</string>
    <!-- Button label in main activity [CHAR LIMIT=10]-->
    <string name="button_main">Send</string>
    <!-- Hint for message edit text in main activity [CHAR LIMIT=30]-->
    <string name="editText_main">Enter Your Message Here</string>
    <!-- Button label in second activity [CHAR LIMIT=10]-->
    <string name="button_second">Reply</string>
    <!-- Hint for reply edit text in second activity [CHAR LIMIT=30]-->
    <string name="editText_second">Enter Your Reply Here</string>
    <!-- Reply header text in main activity [CHAR LIMIT=30]-->
    <string name="text_header_reply">Reply Received</string>
</resources>
```

- styles.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>
</resources>
```

```
</style>  
</resources>
```

#### 4. App biodata + CV



- MainActivity.java

```
package com.example.biodatacv;  
  
import androidx.appcompat.app.AppCompatActivity;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
  
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
    }  
  
    public void button_cv (View view){  
        Intent i = new Intent(this, MainActivity2.class);  
        startActivity(i);  
    }  
}
```

```
}  
}
```

- MainActivity2.java

```
package com.example.biodatacv;  
  
import android.os.Bundle;  
import android.view.View;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import com.github.barteksc.pdfviewer.PDFView;  
  
public class MainActivity2 extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main2);  
        PDFView pdfView=findViewById(R.id.pdfv);  
        pdfView.fromAsset("CV.pdf").load();  
    }  
  
    public void button_cv(View view) {  
    }  
}
```

- Activity\_main.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:app="http://schemas.android.com/apk/res-auto"  
    xmlns:tools="http://schemas.android.com/tools"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    tools:context=".MainActivity">  
  
    <TextView  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:paddingLeft="10dp"  
        android:paddingTop="44dp"  
        android:text="NIM : A11.2021.13469"  
        android:textSize="20sp" />  
  
    <TextView  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:paddingLeft="10dp"  
        android:paddingTop="75dp"  
        android:text="Nama : Ainindzi Nur Meiza Pudjianto"  
        android:textSize="20sp" />
```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingLeft="10dp"
    android:paddingTop="110dp"
    android:text="Brebes"
    android:textSize="20sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingLeft="10dp"
    android:paddingTop="144dp"
    android:text="Telepon : 081542050406"
    android:textSize="20sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingLeft="10dp"
    android:paddingTop="180dp"
    android:text="Email : meizapudjianto@gmail.com"
    android:textSize="20sp" />

<ImageView
    android:id="@+id/imageView7"
    android:layout_width="200dp"
    android:layout_height="200dp"
    android:layout_marginTop="230dp"
    android:layout_marginLeft="100dp"
    app:srcCompat="@drawable/fotobiodata" />

<Button
    android:id="@+id/button"
    android:layout_width="303dp"
    android:layout_height="wrap_content"
    android:layout_alignBottom="@+id/imageView7"
    android:layout_alignParentStart="true"
    android:layout_centerHorizontal="true"
    android:layout_marginStart="46dp"
    android:layout_marginBottom="-81dp"
    android:onClick="button_cv"
    android:text="@string/button_cv"
    tools:ignore="UsingOnClickInXml" />

</RelativeLayout>

```

- Activity\_main2.java

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"

```

```
        android:layout_height="match_parent"
        tools:context=".MainActivity2">
        <com.github.barteksc.pdfviewer.PDFView
            android:id="@+id/pdfv"
            android:layout_width="match_parent"
            android:layout_height="match_parent">
        </com.github.barteksc.pdfviewer.PDFView>

    </RelativeLayout>
```

- Colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
</resources>
```

- Strings.xml

```
<resources>
    <string name="app_name">BIODATA</string>
    <string name="button_cv">CVKU</string>
</resources>
```