Set a date for movable holidays

Software Documentation

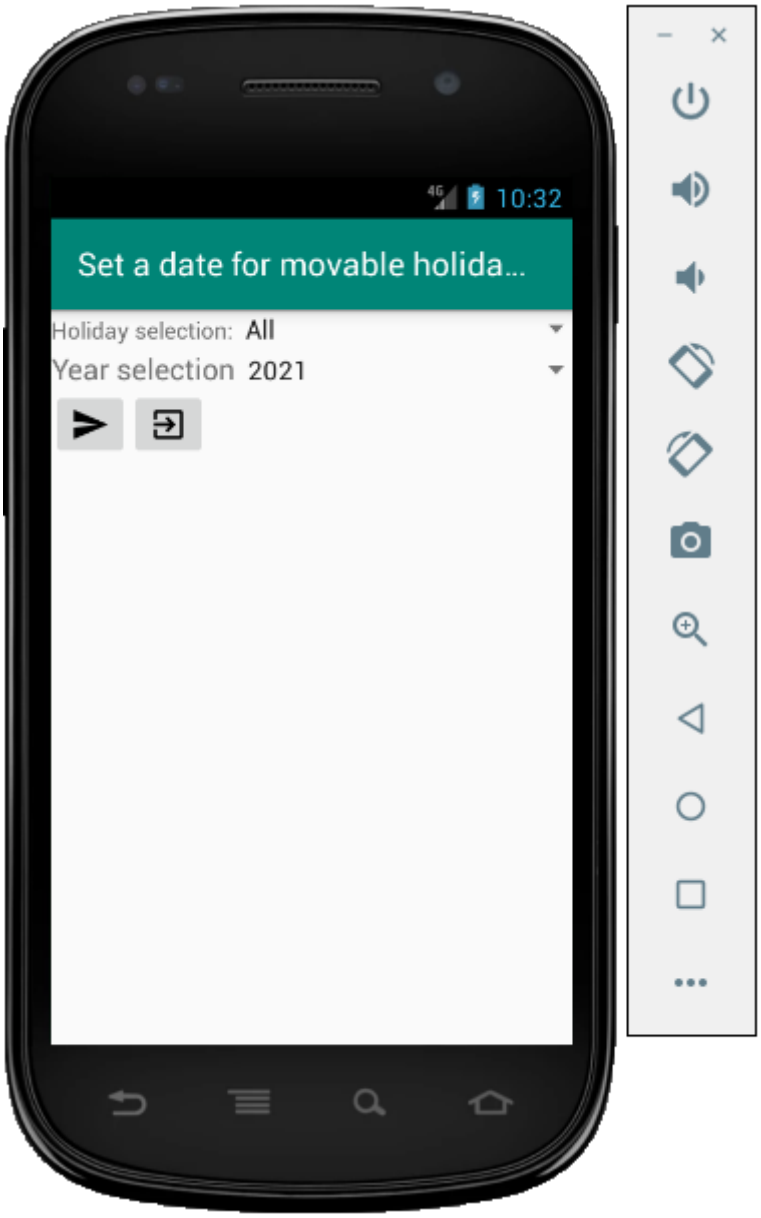
Author: matiwa

Table of contents

Introduction

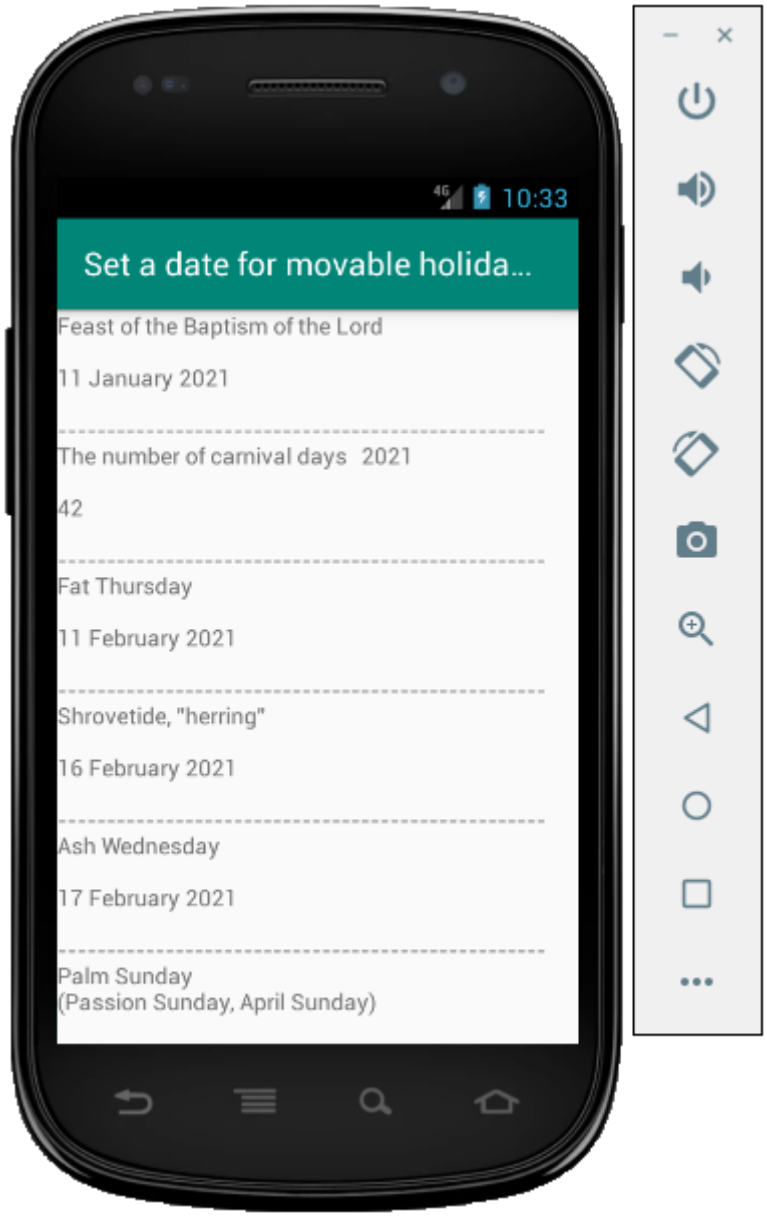
This software documentation includes: description of the application's operation, what is needed for use, algorithms used, interface description and source code description. This application is used to calculate the dates of moving holidays.

Describing of the application’s operation

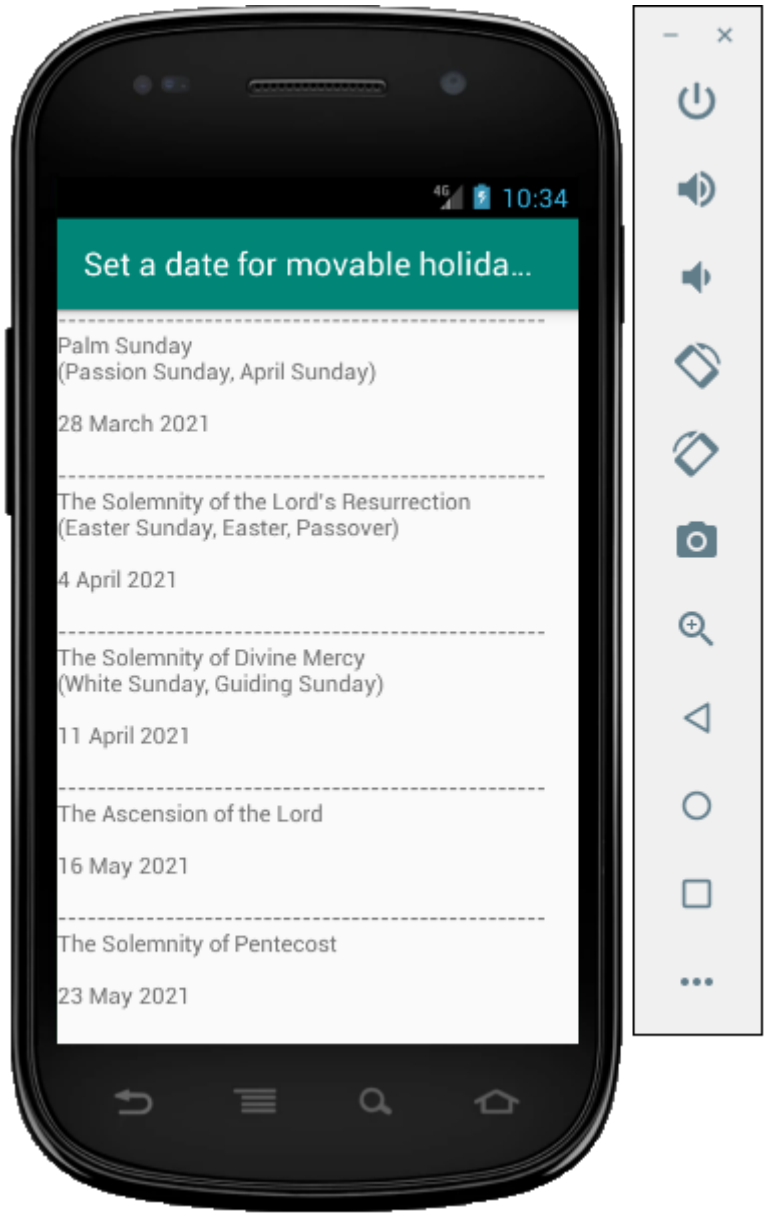


Drawing 1: The beginning of the application's operation [own study]

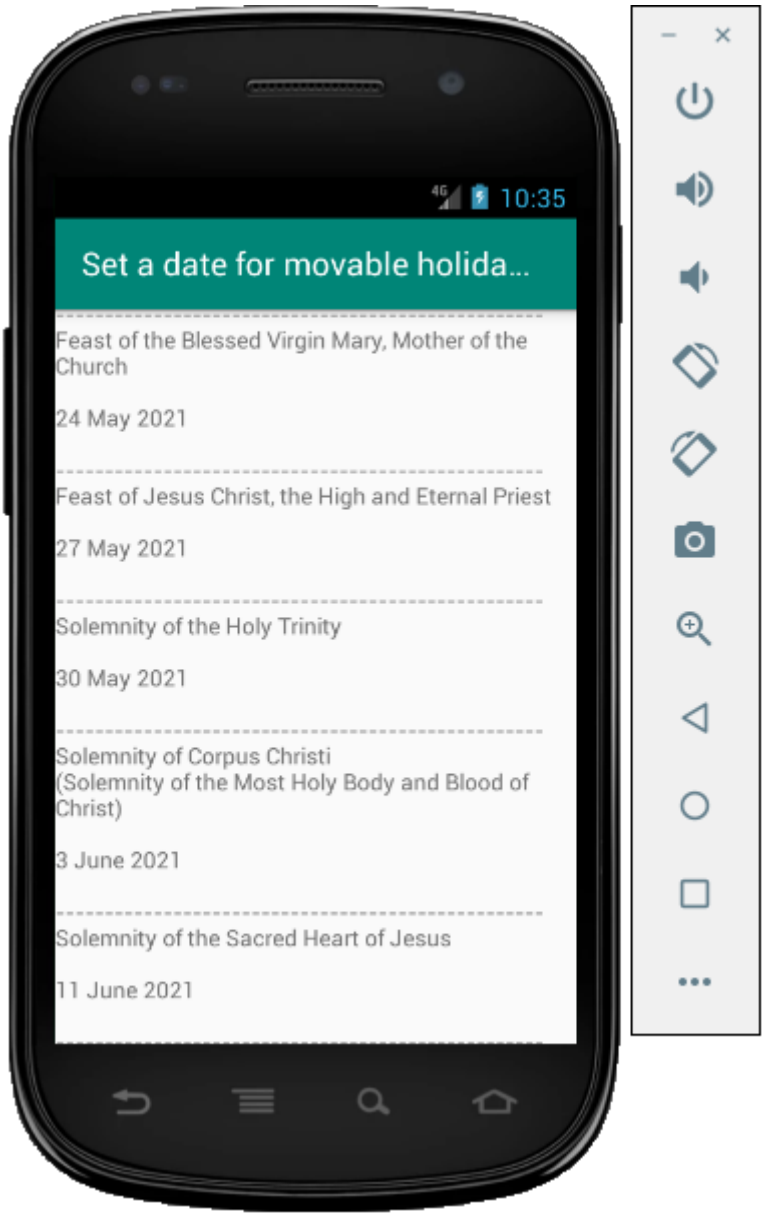
From the drop-down list, the user selects the holiday that he wants to calculate the date. He may choose one or all of the holidays. By default, date calculation is selected for all options. As the second user decides for which year the calculations are to be made. By default, the current year is selected. The user can select the year between 30 and 2999 because in these years the value of the two factors A and B is known in the Gaussian method. Jesus Christ was crucified on the 7th April 30 or 3rd April 33. After confirming the data, calculations take place and the result is displayed.



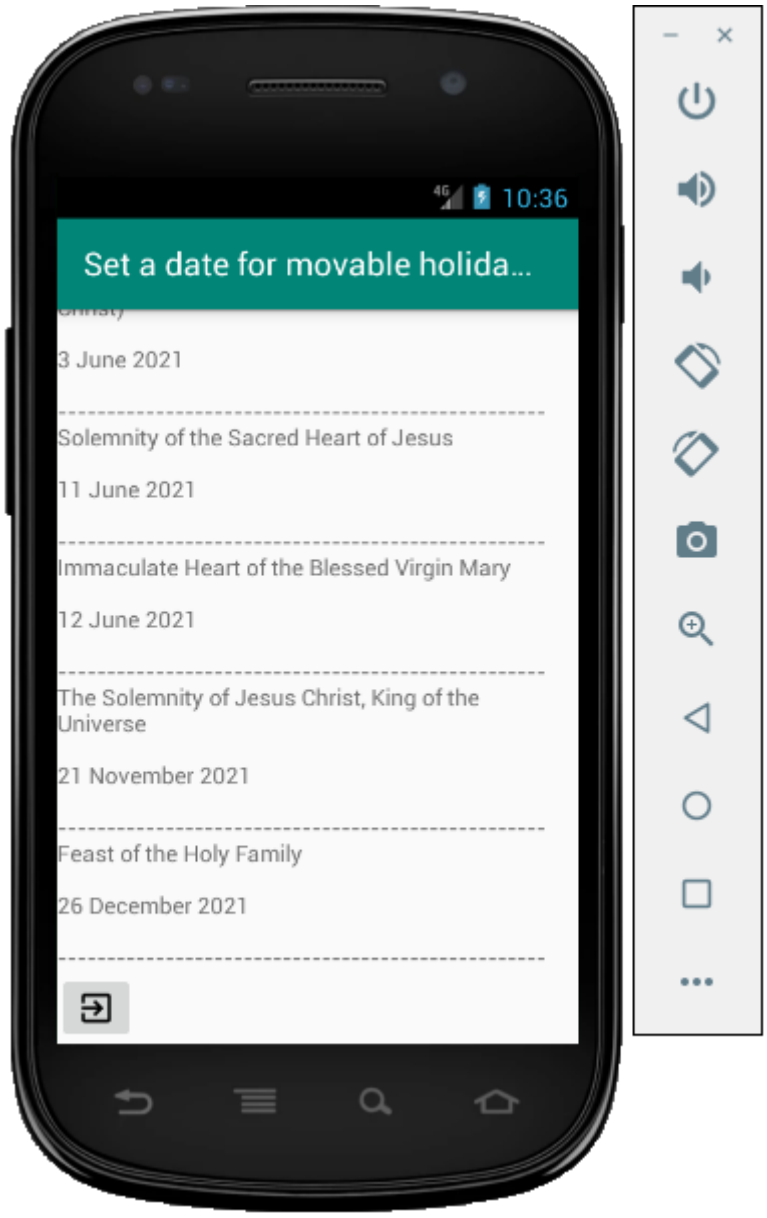
Drawing 2: Result for all options part 1 [own study]



Drawing 3: Result for all options part 2 [own study]



Drawing 4: Result for all options part 3 [own study]



Drawing 5: Result for all options part 4 [own study]

This is the correct use of the program. There may be coding errors that have not been detected by the developer.

What is needed for use?

The application requires installation on the Android operating system with a minimum API level Android 4.0 (IceCreamSandwich), i.e. on all mobile devices with this system.

Algorithm used

The three holidays are based only on an algorithm for calculating the day of the week. There are:

- Feast of the Baptism of the Lord (First Sunday after 6th January - calculations for 6th January)
- The Solemnity of Jesus Christ, King of the Universe (last Sunday in Ordinary Time of the year and the fifth Sunday before Christmas Day [25th December] – calculations for Christmas Day)
- Feast of the Holy Family (Sunday after Christmas, if 25th December does not fall on Sunday, otherwise the holiday is moved to 30th December)

The remaining holidays depend on the date of Easter with the deduction or addition of the appropriate number of days. The algorithm for calculating a leap year should also be taken into account.

The calculation of the date of Easter is based on the Gaussian method for the Gregorian calendar. For this reason, the user can select the range of years between 30 and 2999 as the values of A and B are known for these years. Jesus Christ was crucified on the 7th April 30 or 3rd April 33.

Gauss method

For the Gregorian calendar

The method of calculating this date was given by the German mathematician C.F. Gauss.

Two numbers A and B are needed for the calculations. Their values can be read from the table below :

Lata	A	B	Wyjątki (rok)	
			I rodzaju	II rodzaju
33–1582	15	6	brak	brak
1583–1699	22	2	1609	brak
1700–1799	23	3	brak	brak
1800–1899	23	4	brak	brak
1900–2099	24	5	1981, 2076	1954, 2049
2100–2199	24	6	2133	2106
2200–2299	25	0	2201, 2296	brak
2300–2399	26	1	brak	brak
2400–2499	25	1	2448	brak
2500–2599	26	2	brak	brak
2600–2699	27	3	2668	brak
2700–2899	27	4	2725, 2820	brak
2900–2999	28	5	brak	brak

Drawing 6: Table of A and B values and 1st and 2nd type exceptions [1]

It should be remembered that the Julian calendar was in force until 1582. Thus, the values of A and B in other years refer only to the Gregorian calendar.

Then follow 6 steps:

1. We divide the number of the year by 19 and find the remainder of a.
2. We divide the number of the year by 4 and find the remainder of b.
3. We divide the number of the year by 7 and find the remainder of c.
4. We multiply the remainder a by 19, add the number A to the product, divide the sum by 30 and find the remainder d.
5. We divide the sum of the products 2b + 4c + 6d + B by 7 and find the remainder of e.
6. We add the sum of the remainder of d + e to the date of March 22 and we get the date of Easter.

If the date is above March 31, it should be converted to the corresponding day in April. You can also check if d + e <10. If so, then Easter is (d + e + 22) March. If not, then (d + e - 9) April.

There are exceptions to the above rule:

- An exception of the first type occurs when d = 29 and e = 6, i.e. Easter would fall on April 26. Then it is always celebrated a week in advance, i.e. on April 19. This accident happened in 1609 and 1981.
- The second kind of exception is when d = 28 and e = 6 and dividing 11A + 11 by 30 gives a remainder less than 19 (or simply a> 10). Then, according to the above algorithm, Easter in 1954 was to fall on April 25, and was celebrated on April 18; the second exception - as Gauss wrote in 1807 - "has not yet occurred and will not occur for the first time until 1954".

As you can see, the years with the exceptions up to 2999 are given in the above table of A and B ratios for specific time periods.

For example, we'll set a date for Easter in 2008.

2008: 19 = 105 and the remainder 13. a = 13.

2008: 4 = 502 and the remainder 0.b = 0.

2008: 7 = 286 and the remainder 6. c = 6.

(19 × a + A): 30 → (19 × 13 + 24): 30 = 271: 30 = 9 and the remainder 1.d = 1.

(2 × b + 4 × c + 6 × d + B): 7 → (2 × 0 + 4 × 6 + 6 × 1 + 5): 7 → 35: 7 = 5 and the rest 0.

d + e + 22 → 1 + 0 + 22 = 23, that is, March 23 is the date of Easter in 2008, according to the Gregorian calendar.[1]

Movable holidays - holidays whose celebrations are not related to a date (a specific day of the month).

Movable holidays in the Catholic Church

Holidays associated with Easter

- Ash Wednesday - the first day of Lent, 46 days before Easter Sunday
- Palm Sunday - the Sunday preceding Easter Sunday
- Holy Week - the week between Palm Sunday and Easter Sunday; it concludes with the Paschal Triduum:
    - Holy Thursday
    - Good Friday
    - Holy Saturday

- Easter Sunday - the first Sunday after the first full moon after the vernal equinox (March 20/21)
- Easter Monday - the second day of Easter
- The Solemnity of Divine Mercy - 2nd Sunday of Easter (one week after Easter)
- Ascension Day - 40 days after Easter (Thursday) or on the 7th Easter Sunday
- Pentecost - 50 days after Easter.

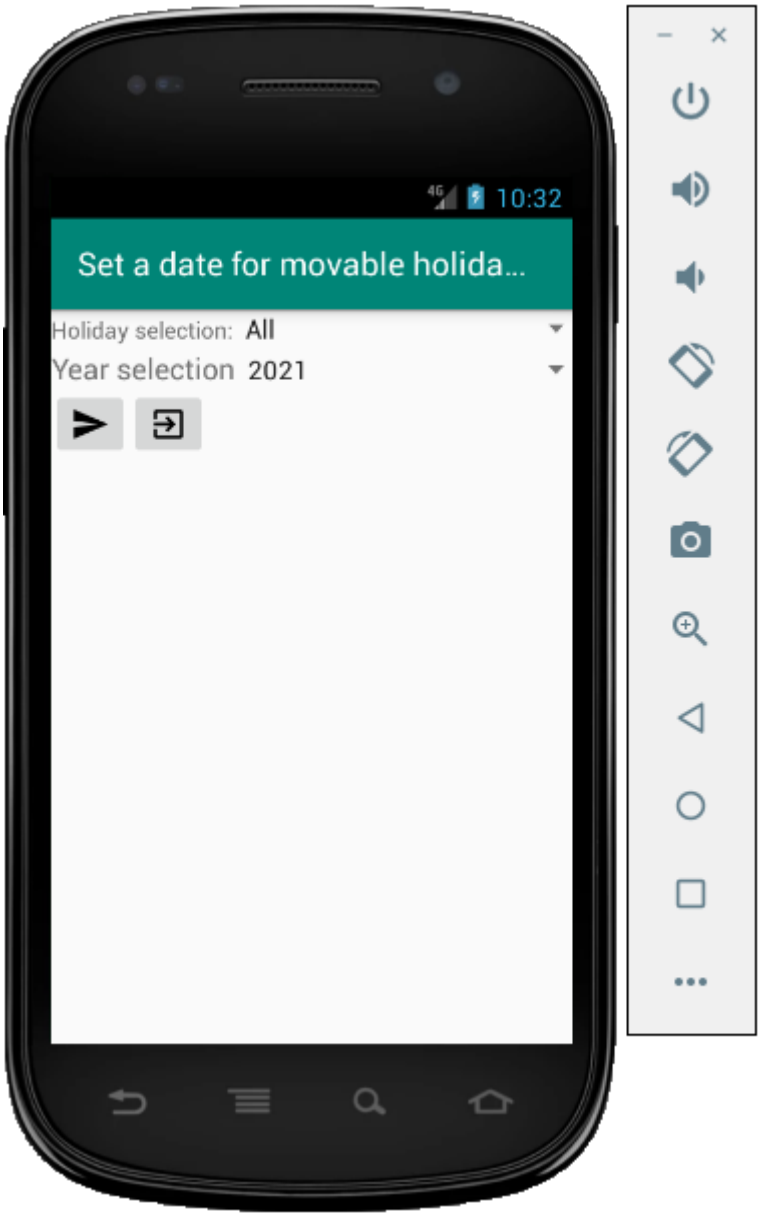
Days not related to Easter, but their date depends on Easter

- Feast of the Blessed Virgin Mary, Mother of the Church - Monday after Pentecost
- Feast of Jesus Christ, the High and Eternal Priest - Thursday after Trinity
- The Solemnity of the Holy Trinity - the first Sunday after Trinity
- Solemnity of the Most Holy Body and Blood of Christ - Thursday after the Holy Trinity
- Solemnity of the Sacred Heart of Jesus - on Friday after the second Sunday after Pentecost
- Memorial of the Immaculate Heart of the Blessed Virgin Mary - Saturday after the Solemnity of the Sacred Heart of Jesus.

Holidays in other seasons of the liturgical year

- Feast of the Baptism of the Lord - Sunday after January 6
- Dedication of One's Own Church - last Sunday in October (in dedicated churches where the date of consecration is unknown)
- Solemnity of Jesus Christ the King of the Universe - last Sunday in Ordinary Time (fifth Sunday before 25th December)
- Feast of the Holy Family - Sunday after 25th December (if Christmas Day falls on Sunday then – 30th December).[2]

Interface description



Drawing 4: Graphical interface [own study]

The interface has the basic components available in the Android Studio development environment: TextView, ImageButton and Spinner.

Source code description

The project was made in the Java programming language, in the Android Studio programming environment. All work was done on the Windows 10 operating system. The application's source code looks like this.

```
package com.example.setadateformovableholidays;

public class Day {

    public static String week[] =
            {"Monday", "Tuesday", "Wednesday", "Thursday", "Friday",
"Saturday", "Sunday"};

    public static int numberOfDays[] =
            {0, 31, 59, 90, 120, 151, 181, 212, 243, 273, 304, 334};

    public static boolean leap(int year) {
        return ((year % 4 == 0  &&  year % 100 != 0) || year % 400
== 0);
    }

    public static int dayOfWeek(int day, int month, int year) {
        int dayOfYear;
        int yy, c, g;
        int result;

        dayOfYear = day + numberOfDays[month-1];
        if ((month > 2) && (leap(year) == true))
            dayOfYear++;

        yy = (year - 1) % 100;
        c = (year - 1) - yy;
        g = yy + (yy / 4);
        result = (((((c / 100) % 4) * 5) + g) % 7);
        result += dayOfYear - 1;
        result %= 7;

        return result;
    }
}
```

Listing 1: The source code for Day.java file [own study]

```
package com.example.setadateformovableholidays;

public class Easter {
    private static int getA(int year)
    {
        if (year <= 1582)
        {
            return 15;
        }
        if (year <= 1699)
```

```
    {
        return 22;
    }
    if (year <= 1899)
    {
        return 23;
    }
    if (year <= 2199)
    {
        return 24;
    }
    if (year <= 2299)
    {
        return 25;
    }
    if (year <= 2399)
    {
        return 26;
    }
    if (year <= 2499)
    {
        return 25;
    }
    if (year <= 2599)
    {
        return 26;
    }
    if (year <= 2699)
    {
        return 27;
    }
    if (year <= 2899)
    {
        return 27;
    }
    if (year <= 2999)
    {
        return 28;
    }

    return 0;
}

private static int getB(int year)
{
    if (year <= 1582)
    {
        return 6;
    }
```

```
    if (year <= 1699)
    {
        return 2;
    }
    if (year <= 1799)
    {
        return 3;
    }
    if (year <= 1899)
    {
        return 4;
    }
    if (year <= 2099)
    {
        return 5;
    }
    if (year <= 2199)
    {
        return 6;
    }
    if (year <= 2299)
    {
        return 0;
    }
    if (year <= 2499)
    {
        return 1;
    }
    if (year <= 2599)
    {
        return 2;
    }
    if (year <= 2699)
    {
        return 3;
    }
    if (year <= 2899)
    {
        return 4;
    }
    if (year <= 2999)
    {
        return 5;
    }

    return 0;
}

public static int easter(int year)
```

```
    {
        int a, b, c, d, e;

        a = year % 19;
        b = year % 4;
        c = year % 7;
        d = (a * 19 + getA(year)) % 30;
        e = (2 * b + 4 * c + 6 * d + getB(year)) % 7;
        if ((d == 29 && e == 6) ||
                (d == 28 && e == 6))
        {
            d -= 7;
        }
        return d + e;
    }

    private static boolean leap(int year)
    {
        return ((year % 4 == 0) && (year % 100 != 0)) || (year % 400
== 0);
    }

    public int dataldk(int year){ //number of carnival days from 6I
to 4II - 10III
        int dat;

        dat = 29 + easter(year);
        if(leap(year)) {
            dat=dat+1;
        }
        return dat;
    }

    public String datatc(int year) //-52days 29I-4III
    {
        int dat;
        int leaping=28;

        dat = 29 + easter(year);
        if(leap(year)){
            leaping=leaping+1;
            dat=dat+1;
        }
        if(dat>(31+leaping)){
            return dat%leaping+" March "+year;
        }
        else if(dat>31){
            return dat%31+" February "+year;
        }
```

```
    else{
        return dat+" January "+year;
    }
}

public String datao(int year) //-47days 3II-9III
{
    int dat;
    int leaping=28;

    dat = 3 + easter(year);
    if(leap(year)) {
        leaping = leaping + 1;
        dat = dat + 1;
    }
    if (dat > leaping) {
        return dat % leaping + " March " + year;
    }
    else {
        return dat + " February " + year;
    }
}

public String datasp(int year) //-46days 4II-10III
{
    int dat;
    int leaping=28;

    dat = 4 + easter(year);
    if(leap(year)) {
        leaping = leaping + 1;
        dat=dat+1;
    }
    if (dat > leaping) {
        return dat % leaping + " March " + year;
    } else {
        return dat + " February " + year;
    }
}

public String datanmp(int year) //-7days
{
    int dat;

    dat = 15 + easter(year);
    if (dat > 31) {
        return dat % 31 + " April " + year;
    }
    else {
```

```
            return dat + " March " + year;
        }
    }

    public String data(int year)
    {
        int dat;

        dat = 22 + easter(year);
        if (dat > 31) {
            return dat % 31 + " April " + year;
        }
        else {
            return dat + " March " + year;
        }
    }

    public String databn(int year) //+7days
    {
        int dat;

        dat = 29 + easter(year);
        if(dat > 61) {
            return dat % 61 + " May " + year;
        }
        else if (dat > 31) {
            return dat % 31 + " April " + year;
        }
        else {
            return dat + " March " + year;
        }
    }

    public String datawp(int year) //+42days or +39days
    {
        int dat;
        int guideline;//date at the earliest from 2004 VII Easter
Sunday (42 days), earlier 39 days (Thursday)
        if(year<2004) guideline=39;
        else guideline=42;

        dat = 22 + guideline + easter(year);
        if(dat > 92) {
            return dat % 92 + " June " + year;
        }
        else {
            return dat % 61 + " May " + year;
        }
    }
```

```
public String datazsp(int year) //+49days
{
    int dat;

    dat = 71 + easter(year);
    if(dat > 92) {
        return dat % 92 + " June " + year;
    }
    else {
        return dat % 61 + " May " + year;
    }
}

public String datanmpmk(int year) //+50days
{
    int dat;

    dat = 72 + easter(year);
    if(dat > 92) {
        return dat % 92 + " June " + year;
    }
    else {
        return dat % 61 + " May " + year;
    }
}

public String datajcniwk(int year) //+53days
{
    int dat;

    dat = 75 + easter(year);
    if(dat > 92) {
        return dat % 92 + " June " + year;
    }
    else {
        return dat % 61 + " May " + year;
    }
}

public String datatp(int year) //+56days
{
    int dat;

    dat = 78 + easter(year);
    if(dat > 92) {
        return dat % 92 + " June " + year;
    }
    else {
```

```
            return dat % 61 + " May " + year;
        }
    }

    public String databc(int year) //+60days
    {
        int dat;

        dat = 82 + easter(year);
        if(dat > 92) {
            return dat % 92 + " June " + year;
        }
        else {
            return dat % 61 + " May " + year;
        }
    }

    public String datanspj(int year) { //+68days 29V-2VII
        int dat;

        dat = 90 + easter(year);
        if (dat > 122) {
            return dat % 122 + " July " + year;
        } else if (dat > 92) {
            return dat % 92 + " June " + year;
        } else {
            return dat % 61 + " May " + year;
        }
    }

    public String datansnmp(int year) { //+69days
        int dat;

        dat = 91 + easter(year);
        if (dat > 122) {
            return dat % 122 + " July " + year;
        } else if (dat > 92) {
            return dat % 92 + " June " + year;
        } else {
            return dat % 61 + " May " + year;
        }
    }
}
```

Listing 2: The source code for Easter.java file [own study]

```
package com.example.setadateformovableholidays;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;
import android.widget.Spinner;
import android.widget.Toast;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.List;
import android.widget.ArrayAdapter;
import android.content.Intent;

public class MainActivity extends AppCompatActivity {
    Spinner holidays,years;
    int year0;
    ImageButton imgbtnc,imgbtne;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        initializecomponents();
    }

    private void initializecomponents(){
        holidays=findViewById(R.id.sh);
        years=findViewById(R.id.sy);
        imgbtnc=findViewById(R.id.imgbtnc);
        imgbtne=findViewById(R.id.imgbtne);

        List<String> list = new ArrayList<String>();
        list.add("All");
        list.add("Feast of the Baptism of the Lord");
        list.add("The number of carnival days");
        list.add("Fat Thursday");
        list.add("Shrovetide, \"herring\"");
        list.add("Ash Wednesday");
        list.add("Palm Sunday");
        list.add("The Solemnity of the Lord's Resurrection");
        list.add("The Solemnity of Divine Mercy");
        list.add("The Ascension of the Lord");
        list.add("The Solemnity of Pentecost");
        list.add("Feast of the Blessed Virgin Mary, Mother of the
Church");
        list.add("Feast of Jesus Christ, the High and Eternal
Priest");
```

```
        list.add("Solemnity of the Holy Trinity");
        list.add("Solemnity of Corpus Christi");
        list.add("Solemnity of the Sacred Heart of Jesus");
        list.add("Immaculate Heart of the Blessed Virgin Mary");
        list.add("The Solemnity of Jesus Christ, King of the
Universe");
        list.add("Feast of the Holy Family");
        ArrayAdapter<String> dataAdapter = new
ArrayAdapter<String>(this,
                android.R.layout.simple_spinner_item, list);

dataAdapter.setDropDownViewResource(android.R.layout.simple_spinner_
dropdown_item);
        holidays.setAdapter(dataAdapter);

        List<String> list2 = new ArrayList<String>();
        for(int i=30;i<3000;i++) list2.add(Integer.toString(i));
        ArrayAdapter<String> dataAdapter2 = new
ArrayAdapter<String>(this,
                android.R.layout.simple_spinner_item, list2);

dataAdapter2.setDropDownViewResource(android.R.layout.simple_spinner
_dropdown_item);
        years.setAdapter(dataAdapter2);

years.setSelection(Calendar.getInstance().get(Calendar.YEAR)-30);

        imgbtnc.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                calculate();
            }
        });
        imgbtne.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                finish();
            }
        });
    }

    private void calculate(){
        Intent i = new Intent(this, Result.class);
        year0=Integer.parseInt(years.getSelectedItem().toString());
        i.putExtra("year", year0);
        switch (String.valueOf(holidays.getSelectedItem())) {
            case "All":
                i.putExtra("option", 0);
                break;
```

```
            case "Feast of the Baptism of the Lord":
                i.putExtra("option", 1);
                break;
            case "The number of carnival days":
                i.putExtra("option", 2);
                break;
            case "Fat Thursday":
                i.putExtra("option", 3);
                break;
            case "Shrovetide, \"herring\"":
                i.putExtra("option", 4);
                break;
            case "Ash Wednesday":
                i.putExtra("option", 5);
                break;
            case "Palm Sunday":
                i.putExtra("option", 6);
                break;
            case "The Solemnity of the Lord's Resurrection":
                i.putExtra("option", 7);
                break;
            case "The Solemnity of Divine Mercy":
                i.putExtra("option", 8);
                break;
            case "The Ascension of the Lord":
                i.putExtra("option", 9);
                break;
            case "The Solemnity of Pentecost":
                i.putExtra("option", 10);
                break;
            case "Feast of the Blessed Virgin Mary, Mother of the
Church":
                i.putExtra("option", 11);
                break;
            case "Feast of Jesus Christ, the High and Eternal
Priest":
                i.putExtra("option", 12);
                break;
            case "Solemnity of the Holy Trinity":
                i.putExtra("option", 13);
                break;
            case "Solemnity of Corpus Christi":
                i.putExtra("option", 14);
                break;
            case "Solemnity of the Sacred Heart of Jesus":
                i.putExtra("option", 15);
                break;
            case "Immaculate Heart of the Blessed Virgin Mary":
                i.putExtra("option", 16);
```

```
                break;
            case "The Solemnity of Jesus Christ, King of the
Universe":
                i.putExtra("option", 17);
                break;
            case "Feast of the Holy Family":
                i.putExtra("option", 18);
                break;
            default:
                break;
        }
        startActivity(i);
    }
}
```

Listing 3: The source code for MainActivity.java file [own study]

```
package com.example.setadateformovableholidays;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;
import android.widget.TextView;

public class Result extends AppCompatActivity {
    TextView tv;
    ImageButton imgbtne;
    int year0;
    int option;
    CharSequence buffer="";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_result);

        initializecomponents();
    }

    private void initializecomponents() {
        tv = findViewById(R.id.tv);
        imgbtne=findViewById(R.id.imgbtne);
        imgbtne.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                finish();
            }
```

```
        });
        year0 = getIntent().getIntExtra("year", year0);
        option = getIntent().getIntExtra("option", option);
        Easter easter;
        Day day;
        String screen = "";
        switch (option) {
            case 0: {
                day = new Day();
                easter = new Easter();
                screen = screen + "" + "----------------------------
----------------------" + "\r\n";
                screen = "Feast of the Baptism of the Lord\r\n\r\n";
                screen = screen + (13 - day.dayOfWeek(6, 1, year0))
+ " January " + year0;
                screen = screen + "\r\n\r\n" + "--------------------
------------------------------" + "\r\n";
                screen = screen + "The number of carnival days \t" +
year0 + "\r\n\r\n";
                screen = screen + easter.dataldk(year0);
                screen = screen + "\r\n\r\n" + "--------------------
------------------------------" + "\r\n";
                screen = screen + "Fat Thursday\r\n\r\n";
                screen = screen + easter.datatc(year0);
                screen = screen + "\r\n\r\n" + "--------------------
------------------------------" + "\r\n";
                screen = screen + "Shrovetide, \"herring\"\r\n\r\n";
                screen = screen + easter.datao(year0);
                screen = screen + "\r\n\r\n" + "--------------------
------------------------------" + "\r\n";
                screen = screen + "Ash Wednesday\r\n\r\n";
                screen = screen + easter.datasp(year0);
                screen = screen + "\r\n\r\n" + "--------------------
------------------------------" + "\r\n";
                screen = screen + "Palm Sunday\r\n(Passion Sunday,
April Sunday)\r\n\r\n";
                screen = screen + easter.datanmp(year0);
                screen = screen + "\r\n\r\n" + "--------------------
------------------------------" + "\r\n";
                screen = screen + "The Solemnity of the Lord's
Resurrection\r\n(Easter Sunday, Easter, Passover)\r\n\r\n";
                screen = screen + easter.data(year0);
                screen = screen + "\r\n\r\n" + "--------------------
------------------------------" + "\r\n";
                screen = screen + "The Solemnity of Divine
Mercy\r\n(White Sunday, Guiding Sunday)\r\n\r\n";
                screen = screen + easter.databn(year0);
                screen = screen + "\r\n\r\n" + "--------------------
------------------------------" + "\r\n";
```

```
                screen = screen + "The Ascension of the
Lord\r\n\r\n";
                screen = screen + easter.datawp(year0);
                screen = screen + "\r\n\r\n" + "----------
------------------------------" + "\r\n";
                screen = screen + "The Solemnity of
Pentecost\r\n\r\n";
                screen = screen + easter.datazsp(year0);
                screen = screen + "\r\n\r\n" + "----------
------------------------------" + "\r\n";
                screen = screen + "Feast of the Blessed Virgin Mary,
Mother of the Church\r\n\r\n";
                screen = screen + easter.datanmpmk(year0);
                screen = screen + "\r\n\r\n" + "----------
------------------------------" + "\r\n";
                screen = screen + "Feast of Jesus Christ, the High
and Eternal Priest\r\n\r\n";
                screen = screen + easter.datajcniwk(year0);
                screen = screen + "\r\n\r\n" + "----------
------------------------------" + "\r\n";
                screen = screen + "Solemnity of the Holy
Trinity\r\n\r\n";
                screen = screen + easter.datatp(year0);
                screen = screen + "\r\n\r\n" + "----------
------------------------------" + "\r\n";
                screen = screen + "Solemnity of Corpus
Christi\r\n(Solemnity of the Most Holy Body and Blood of
Christ)\r\n\r\n";
                screen = screen + easter.databc(year0);
                screen = screen + "\r\n\r\n" + "----------
------------------------------" + "\r\n";
                screen = screen + "Solemnity of the Sacred Heart of
Jesus\r\n\r\n";
                screen = screen + easter.datanspj(year0);
                screen = screen + "\r\n\r\n" + "----------
------------------------------" + "\r\n";
                screen = screen + "Immaculate Heart of the Blessed
Virgin Mary\r\n\r\n";
                screen = screen + easter.datansnmp(year0);
                screen = screen + "\r\n\r\n" + "----------
------------------------------" + "\r\n";
                screen = screen + "The Solemnity of Jesus Christ,
King of the Universe\r\n\r\n";//20-26 November
                screen = screen + (26 - day.dayOfWeek(25, 12,
year0)) + " November " + year0;
                screen = screen + "\r\n\r\n" + "----------
------------------------------" + "\r\n";
                screen = screen + "Feast of the Holy
Family\r\n\r\n";
```

```
                if (day.dayOfWeek(25, 12, year0) == 6)
                    screen = screen + "30 December " + year0;
                else
                    screen = screen + (31 - day.dayOfWeek(25, 12,
year0)) + " December " + year0;
                screen = screen + "\r\n\r\n" + "----------------------
------------------------------" + "";
                break;
            }
            case 1:
                day = new Day();
                screen = screen + "Feast of the Baptism of the
Lord\r\n\r\n";
                screen = screen + (13 - day.dayOfWeek(6, 1, year0))
+ " January " + year0;
                break;
            case 2:
                easter = new Easter();
                screen = screen + "The number of carnival days \t" +
year0 + "\r\n\r\n";
                screen = screen + easter.dataldk(year0);
                break;
            case 3:
                easter = new Easter();
                screen = screen + "Fat Thursday\r\n\r\n";
                screen = screen + easter.datatc(year0);
                break;
            case 4:
                easter = new Easter();
                screen = screen + "Shrovetide, \"herring\"\r\n\r\n";
                screen = screen + easter.datao(year0);
                break;
            case 5:
                easter = new Easter();
                screen = screen + "Ash Wednesday\r\n\r\n";
                screen = screen + easter.datasp(year0);
                break;
            case 6:
                easter = new Easter();
                screen = screen + "Palm Sunday\r\n(Passion Sunday,
April Sunday)\r\n\r\n";
                screen = screen + easter.datanmp(year0);
                break;
            case 7:
                easter = new Easter();
                screen = screen + "The Solemnity of the Lord's
Resurrection\r\n(Easter Sunday, Easter, Passover)\r\n\r\n";
                screen = screen + easter.data(year0);
                break;
```

```
            case 8:
                easter = new Easter();
                screen = screen + "The Solemnity of Divine
Mercy\r\n(White Sunday, Guiding Sunday)\r\n\r\n";
                screen = screen + easter.databn(year0);
                break;
            case 9:
                easter = new Easter();
                screen = screen + "The Ascension of the
Lord\r\n\r\n";
                screen = screen + easter.datawp(year0);
                break;
            case 10:
                easter = new Easter();
                screen = screen + "The Solemnity of
Pentecost\r\n\r\n";
                screen = screen + easter.datazsp(year0);
                break;
            case 11:
                easter = new Easter();
                screen = screen + "Feast of the Blessed Virgin Mary,
Mother of the Church\r\n\r\n";
                screen = screen + easter.datanmpmk(year0);
                break;
            case 12:
                easter = new Easter();
                screen = screen + "Feast of Jesus Christ, the High
and Eternal Priest\r\n\r\n";
                screen = screen + easter.datajcniwk(year0);
                break;
            case 13:
                easter = new Easter();
                screen = screen + "Solemnity of the Holy
Trinity\r\n\r\n";
                screen = screen + easter.datatp(year0);
                break;
            case 14:
                easter = new Easter();
                screen = screen + "Solemnity of Corpus
Christi\r\n(Solemnity of the Most Holy Body and Blood of
Christ)\r\n\r\n";
                screen = screen + easter.databc(year0);
                break;
            case 15:
                easter = new Easter();
                screen = screen + "Solemnity of the Sacred Heart of
Jesus\r\n\r\n";
                screen = screen + easter.datanspj(year0);
                break;
```

```
            case 16:
                easter = new Easter();
                screen = screen + "Immaculate Heart of the Blessed
Virgin Mary\r\n\r\n";
                screen = screen + easter.datansnmp(year0);
                break;
            case 17:
                day = new Day();
                screen = screen + "The Solemnity of Jesus Christ,
King of the Universe\r\n\r\n";//20-26 November
                screen = screen + (26 - day.dayOfWeek(25, 12,
year0)) + " November " + year0;
                break;
            case 18:
                day = new Day();
                screen = screen + "Feast of the Holy
Family\r\n\r\n";
                if (day.dayOfWeek(25, 12, year0) == 6)
                    screen = screen + "30 December " + year0;
                else
                    screen = screen + (31 - day.dayOfWeek(25, 12,
year0)) + " December " + year0;
                break;
            default:
                break;
        }
        buffer = screen;
        tv.setText(buffer);
    }
}
```

Listing 4: The source code for Result.java file [own study]

```
<?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
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        tools:ignore="MissingConstraints">
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Holiday selection:"
        android:textSize="14sp" />

    <Spinner
        android:id="@+id/sh"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Year selection"
        android:textSize="18sp" />

    <Spinner
        android:id="@+id/sy"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal">

    <ImageButton
        android:id="@+id/imgbtnc"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:srcCompat="@drawable/ic_send_black_24dp" />

    <ImageButton
        android:id="@+id/imgbtne"
        android:layout_width="wrap_content"
```

```
                android:layout_height="wrap_content"
                app:srcCompat="@drawable/ic_exit_to_app_black_24dp"
/>
        </LinearLayout>

    </LinearLayout>

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

Listing 5: The source code for activity_main.xml file [own study]

```
<?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=".Result">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical" >

            <TextView
                android:id="@+id/tv"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" />

            <ImageButton
                android:id="@+id/imgbtne"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                app:srcCompat="@drawable/ic_exit_to_app_black_24dp"
/>

        </LinearLayout>
    </ScrollView>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Listing 6: The source code for activity_result.xml file [own study]

List of drawings

List of listings

Bibliography:

[1] https://pl.wikipedia.org/wiki/Wielkanoc
[2] https://pl.wikipedia.org/wiki/%C5%9Awi%C4%99ta_ruchome