Zita Trukse

My Birthday App
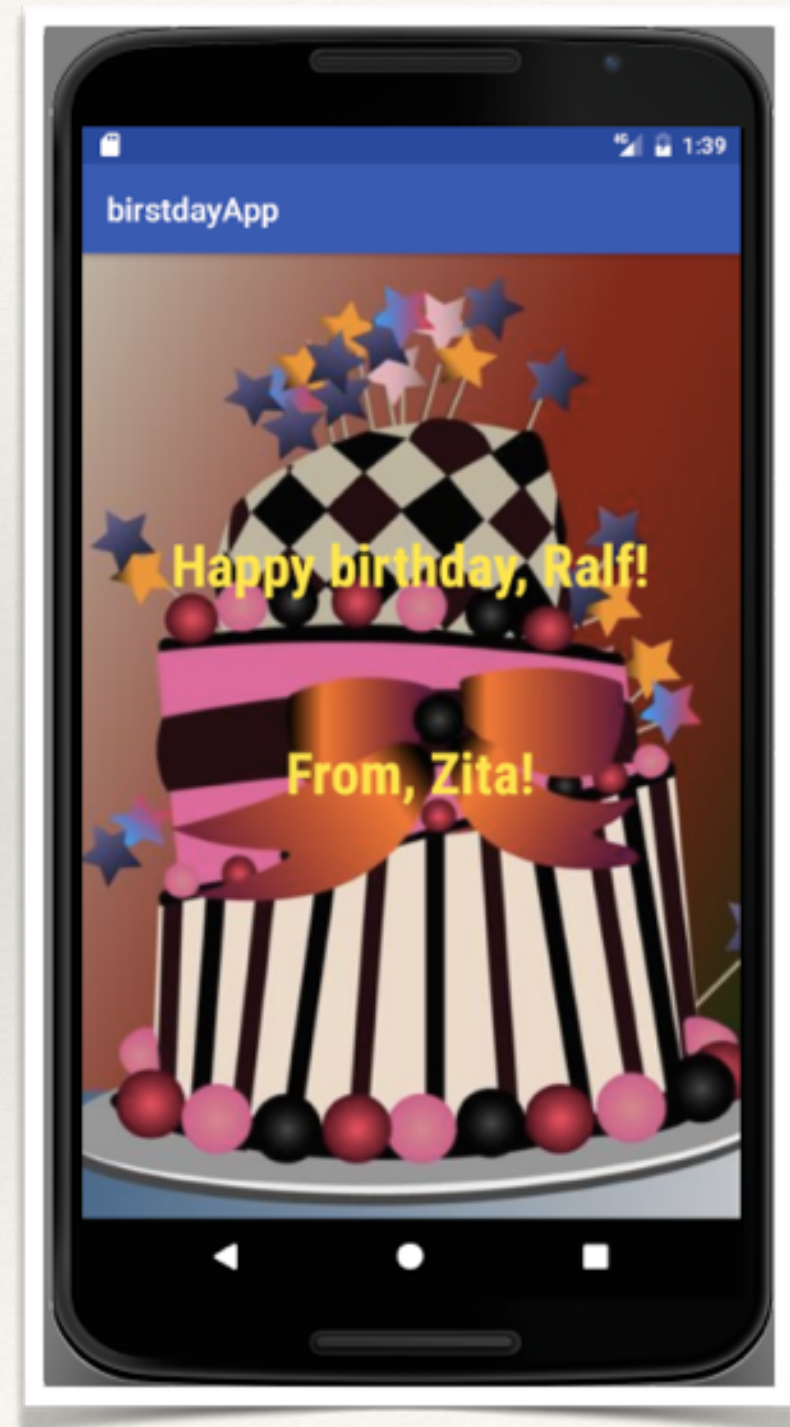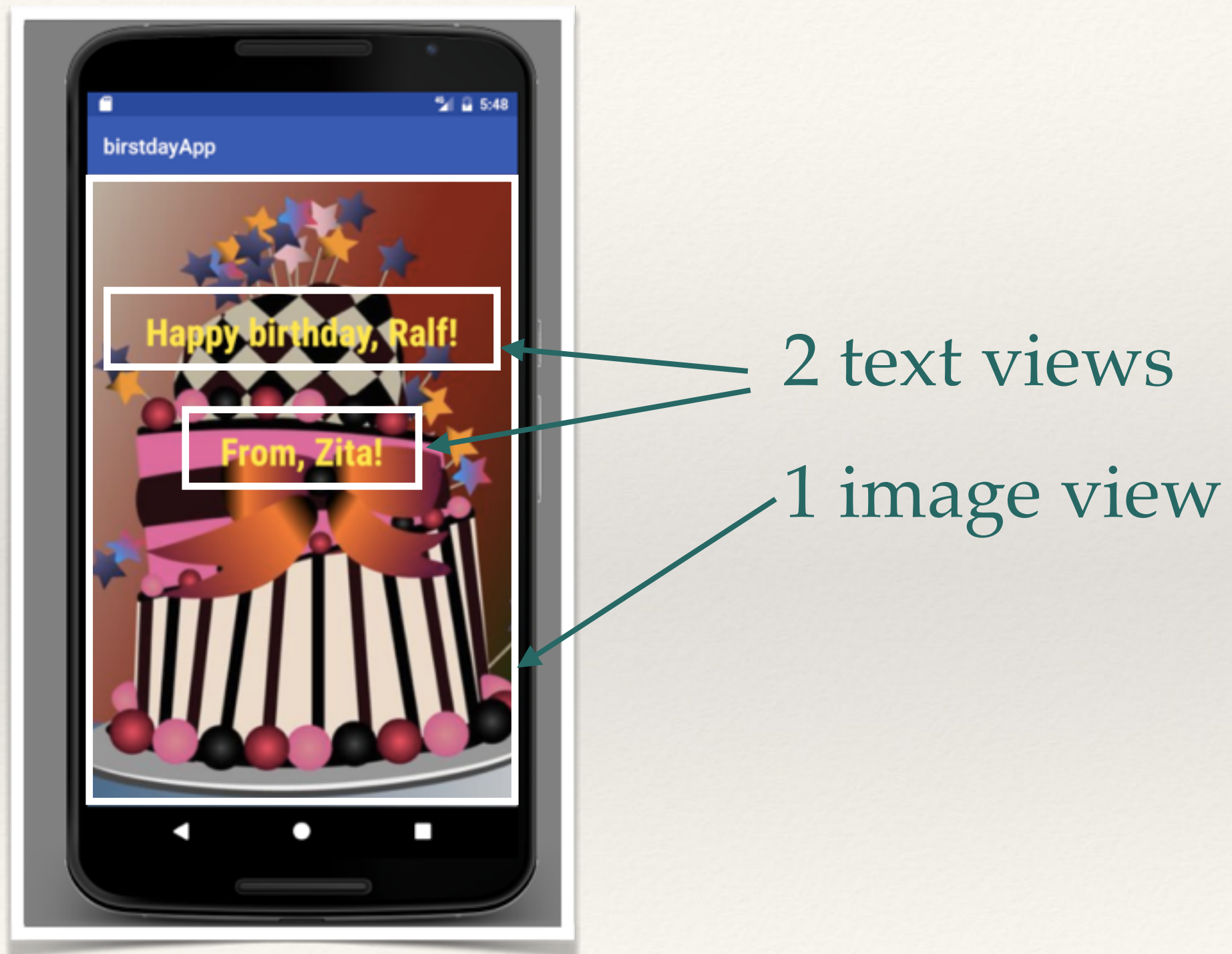
Android Studio

My Birthday App

Birthday App in 3 steps

1. Step: Select the Views

2. Step: Position the Views

3. Step: Style the Views
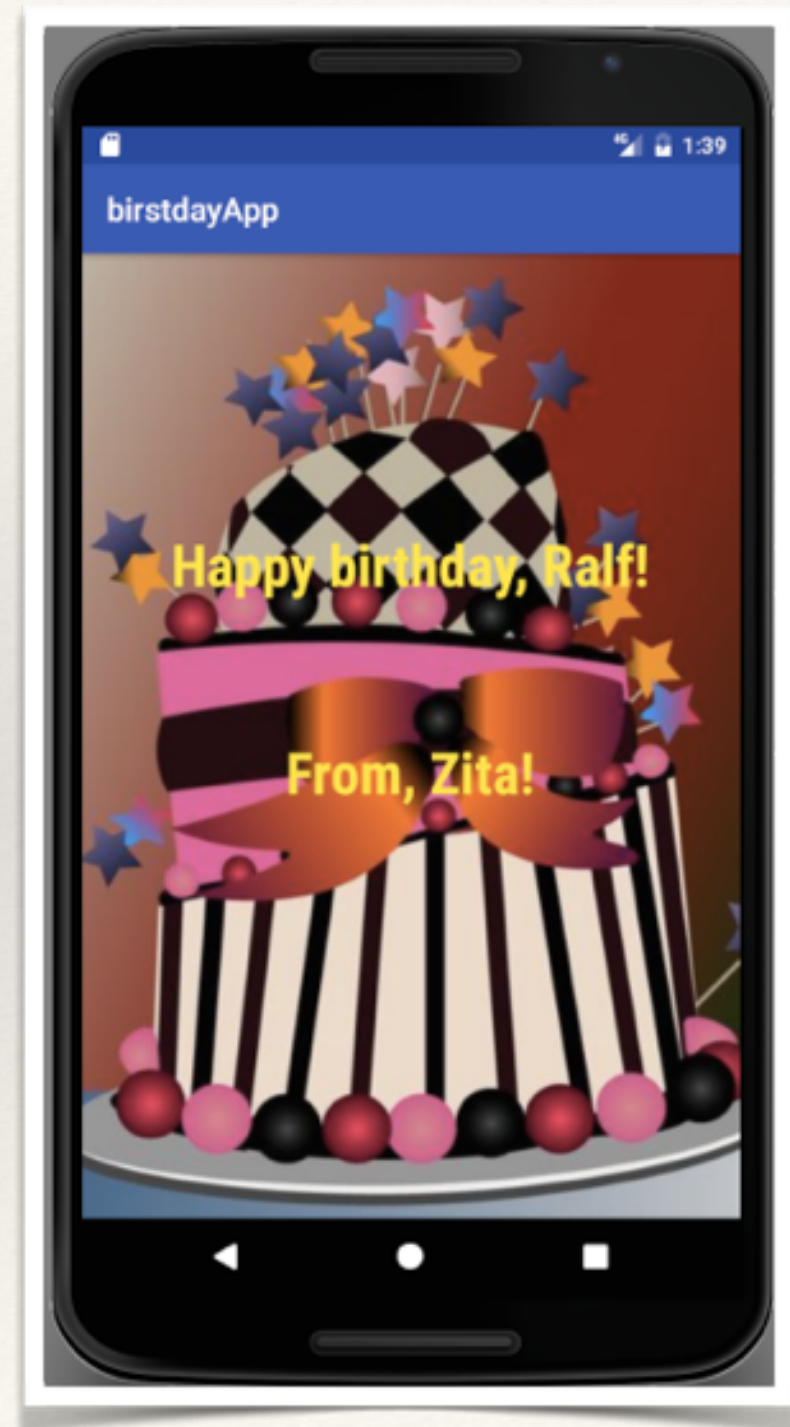
1. step: **Select** the views

2 text views

1 image view

2. step: **Position** the views

RelativeLayout:

- textViews are placed separate, relative

- textViews overlaps the imageView

Creation process of birthday card

inspiration from realistic, fancy birthday cakes

My Birthday cake –

illustrator

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context="com.example.zitatrukse.birstdayapp.MainActivity">


    <ImageView
        android:src="@drawable/cake"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scaleType="centerCrop"  />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Happy birthday, Ralf!"
        android:textSize="36sp"
        android:fontFamily="sans-serif-condensed"
        android:textColor="#ffeb3b"
        android:textStyle="bold"
        android:layout_centerHorizontal="true"
        android:paddingTop="170dp"
         />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="From, Zita!"
        android:textSize="36sp"
        android:fontFamily="sans-serif-condensed"
        android:textStyle="bold"
        android:textColor="#ffe83b"
        android:layout_centerHorizontal="true"
        android:paddingTop="300dp"
        />

</RelativeLayout>
```

relativeLayout opening tag

imageView

textView

textView

relativeLayout closing tag

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.zitatrukse.birstdayapp.MainActivity">
```

RelativeLayout open tag:

- android attributes specification by default from Android Studio

- attribute id

- layout_width and layout_height are „match_parent" because textViews and imageView have to cover all display.

```xml
<ImageView
    android:src="@drawable/cake"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
```

ImageView:

- attribute android:src shows were is placed image and how is named image , what is used for this ImageView.

- layout_width and layout_height is „match_parent" , because imageView has to cover all display.

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Happy birthday, Ralf!"
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="From, Zita!"
```

TextViews:

- layout_width and layout_height is „wrap_content" , because text has no background and layout_width and layout_height values are not relevant. There could also has been placed „match_parent".

- android:text value is message, what will display on the screen.

3. step: **Style** the views

now

goal

What do we need style?

- text size

- text color

- font family

- text placement

- image size

```
<ImageView
    android:src="@drawable/cake"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:scaleType="centerCrop"   />
```

The styling of ImageView:

- add attribute android:scaleType with value centerCrop, that means that an image is stretched proportionally to cover all screen.

```xml
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Happy birthday, Ralf!"
    android:textSize="36sp"
    android:fontFamily="sans-serif-condensed"
    android:textColor="#ffeb3b"
    android:textStyle="bold"
    android:layout_centerHorizontal="true"
    android:paddingTop="170dp"
    />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="From, Zita!"
    android:textSize="36sp"
    android:fontFamily="sans-serif-condensed"
    android:textStyle="bold"
    android:textColor="#ffe83b"
    android:layout_centerHorizontal="true"
    android:paddingTop="300dp"
    />
```

The styling of TextView:

- increases a text size to 36sp

- change font family to sans-serif-condensed

- use color pallet and choose specific yellow colors code

- text style boldIncreases a text size to 36

- text is placed in center there for we add attribute android:layout_centerHorizontal with value true

- add padding from top side to place each text in the write place.

My Birthday App