

RESTful Services

Funshine

Teil 4: Designing the UI

Gestalten der UI

activity_weather.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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"
    android:background="#499eec"
    tools:context="at.htl.funshine.WeatherActivity">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:contentDescription=""
        app:srcCompat="@drawable/sunny"
        app:layout_constraintStart_toStartOf="parent"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        app:layout_constraintTop_toTopOf="parent"
        android:layout_marginTop="8dp" />

    <ImageView
        android:id="@+id/imageView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:srcCompat="@drawable/funshine_logo"
        app:layout_constraintStart_toEndOf="@+id/imageView"
        android:layout_marginStart="8dp"
        app:layout_constraintTop_toTopOf="parent"
        android:layout_marginTop="16dp"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginEnd="8dp"
        app:layout_constraintHorizontal_bias="0.037" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Today, May 1"
        android:textSize="25sp"
        app:layout_constraintStart_toStartOf="parent"
        android:layout_marginStart="8dp"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginEnd="8dp"
        android:layout_marginTop="16dp"
        app:layout_constraintTop_toBottomOf="@+id/imageView"
        app:layout_constraintHorizontal_bias="0.081" />
```

```
<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="78°"
    android:textSize="100sp"
    app:layout_constraintStart_toStartOf="parent"
    android:layout_marginStart="24dp"
    android:layout_marginLeft="24dp"
    android:layout_marginTop="0dp"
    app:layout_constraintTop_toBottomOf="@+id/textView2" />

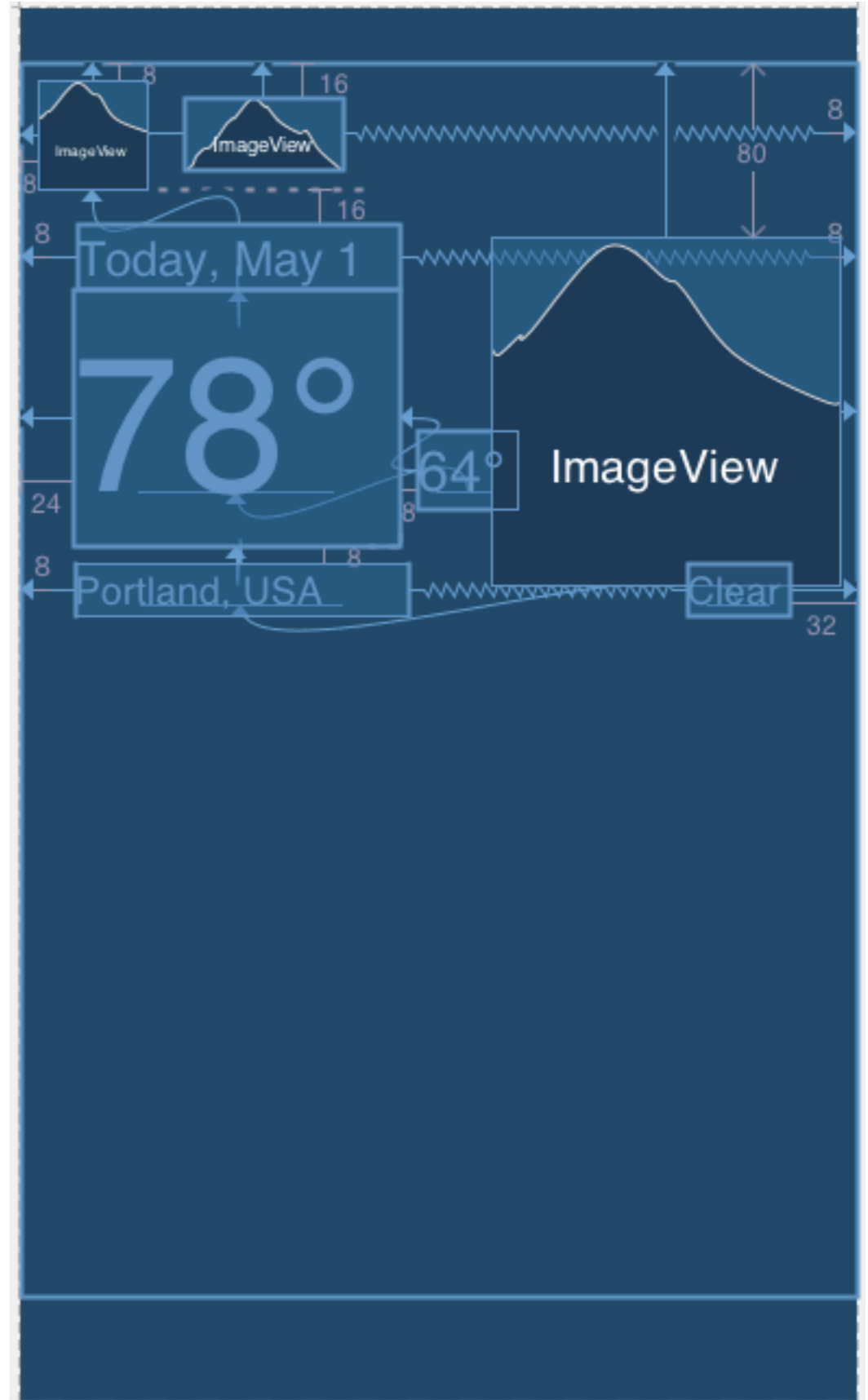
<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="64°"
    android:textSize="30sp"
    app:layout_constraintStart_toEndOf="@+id/textView3"
    android:layout_marginStart="8dp"
    android:layout_marginLeft="8dp"
    app:layout_constraintBaseline_toBaselineOf="@+id/textView3" />

<ImageView
    android:id="@+id/imageView3"
    android:layout_width="160dp"
    android:layout_height="160dp"
    app:srcCompat="@drawable/sunny"
    app:layout_constraintStart_toEndOf="@+id/textView4"
    android:layout_marginStart="8dp"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginEnd="8dp"
    app:layout_constraintTop_toTopOf="parent"
    android:layout_marginTop="80dp"
    app:layout_constraintHorizontal_bias="1.0" />

<TextView
    android:id="@+id/textView5"
    android:layout_width="153dp"
    android:layout_height="wrap_content"
    android:text="Portland, USA"
    android:textSize="20sp"
    app:layout_constraintStart_toStartOf="parent"
    android:layout_marginStart="8dp"
    app:layout_constraintEnd_toStartOf="@+id/textView6"
    android:layout_marginEnd="8dp"
    app:layout_constraintHorizontal_bias="0.12"
    android:layout_marginTop="8dp"
    app:layout_constraintTop_toBottomOf="@+id/textView3" />

<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Clear"
    android:textSize="20sp"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginEnd="32dp"
    android:layout_marginRight="32dp"
    app:layout_constraintBaseline_toBaselineOf="@+id/textView5" />

</android.support.constraint.ConstraintLayout>
```

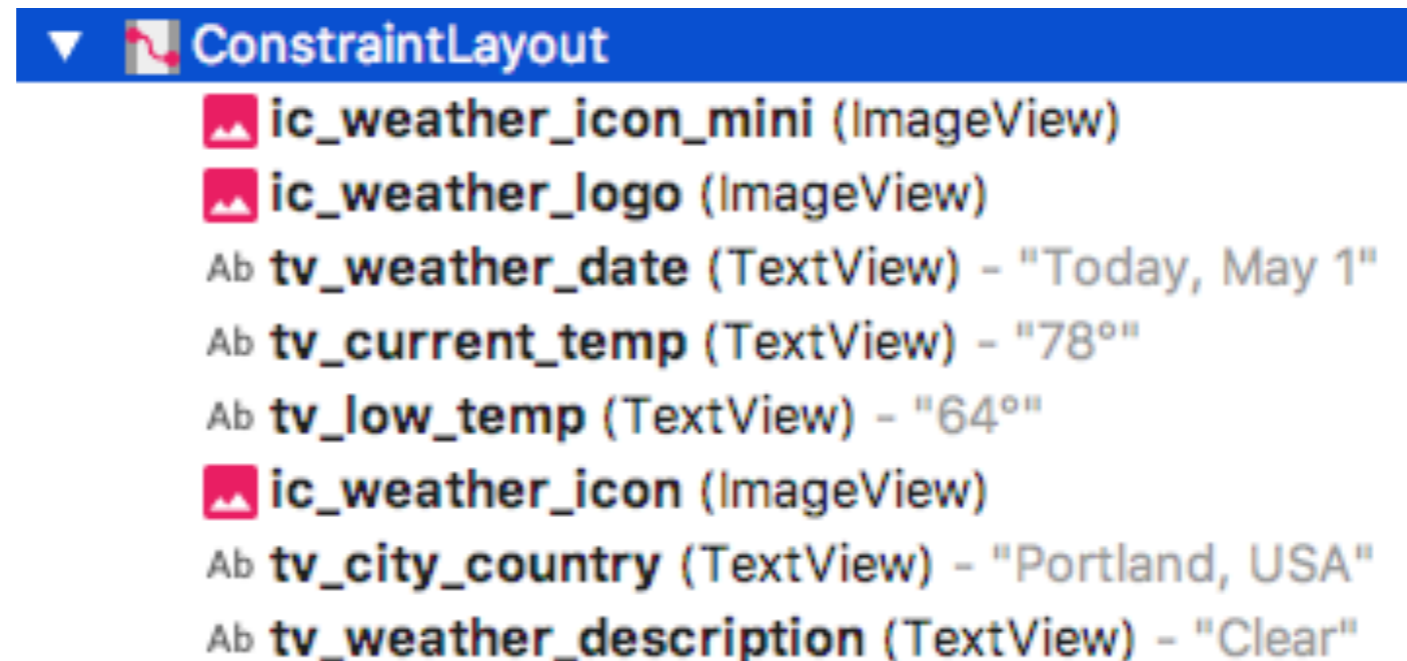


Naming conventions




https://github.com/ribot/android-guidelines/blob/master/project_and_code_guidelines.md

<http://stackoverflow.com/questions/12870537/android-naming-convention>

```
Button - btn
EditText - et
TextView - tv
Checkbox - chk
RadioButton - rb
ToggleButton - tb
Spinner - spn
Menu - mnu
ListView - lv
GalleryView - gv
LinearLayout - ll
RelativeLayout - rl
```



The screenshot shows a blue header bar for a **ConstraintLayout**. Below it, several views are listed with their IDs and types:

-  **ic_weather_icon_mini** (ImageView)
-  **ic_weather_logo** (ImageView)
- Ab **tv_weather_date** (TextView) - "Today, May 1"
- Ab **tv_current_temp** (TextView) - "78°"
- Ab **tv_low_temp** (TextView) - "64°"
-  **ic_weather_icon** (ImageView)
- Ab **tv_city_country** (TextView) - "Portland, USA"
- Ab **tv_weather_description** (TextView) - "Clear"

WeatherActivity.java

```
private GoogleApiClient mGoogleApiClient;  
private final int PERMISSION_LOCATION = 111;  
private List<DailyWeatherReport> weatherReportList = new ArrayList<
```

```
private ImageView weatherIcon;  
private ImageView weatherIconMini;  
private TextView weatherDate;  
private TextView currentTemp;  
private TextView lowTemp;  
private TextView cityCountry;  
private TextView weatherDescription;
```

```
@Override
```

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_weather);
```

```
    weatherIcon = (ImageView) findViewById(R.id.ic_weather_icon);  
    weatherIconMini = (ImageView) findViewById(R.id.ic_weather_icon  
    weatherDate = (TextView) findViewById(R.id.tv_weather_date);  
    currentTemp = (TextView) findViewById(R.id.tv_current_temp);  
    lowTemp = (TextView) findViewById(R.id.tv_low_temp);  
    cityCountry = (TextView) findViewById(R.id.tv_city_country);  
    weatherDescription = (TextView) findViewById(R.id.tv_weather_de
```

```
    mGoogleApiClient = new GoogleApiClient.Builder(this)  
        .addApi(LocationServices.API)  
        .enableAutoManage(this, this)  
        .addConnectionCallbacks(this)  
        .addOnConnectionFailedListener(this)  
        .build();
```

```
}
```

WeatherActivity.java

```
public void downloadWeatherData(Location location) {
    final String fullCoords = URL_COORD + location.getLatitude() + "&lon=" + location.getLongitude();
    final String url = URL_BASE + fullCoords + URL_UNITS + URL_API_KEY;

    final JsonObjectRequest jsonRequest = new JsonObjectRequest(
        Request.Method.GET,
        url,
        null, // hier könnte man JSON-Daten im body mitschicken
        new Response.Listener<JSONObject>() {
            @Override
            public void onResponse(JSONObject response) {
                try {
                    ...

                }

                Log.v(LOG_TAG, "Name: " + cityName + " - Country: " + country);

            } catch (JSONException e) {
                Log.e(LOG_TAG, e.getMessage());
            }

            updateUI();

        }, new Response.ErrorListener() {
            @Override
            public void onErrorResponse(VolleyError error) {
                Log.v(LOG_TAG, "Err: " + error.getMessage());
            }
        });

    Volley.newRequestQueue(this).add(jsonRequest);
}
```

```
public void updateUI() {
    if (weatherReportList.size() > 0) {
        DailyWeatherReport report = weatherReportList.get(0);
    }
}
```


DailyWeatherReport.java

Für die einzelnen Wetterarten werden Konstanten angelegt.

```
public class DailyWeatherReport {  
  
    public static final String WEATHER_TYPE_CLOUDS = "Clouds";  
    public static final String WEATHER_TYPE_CLEAR = "Clear";  
    public static final String WEATHER_TYPE_RAIN = "Rain";  
    public static final String WEATHER_TYPE_WIND = "Wind";  
    public static final String WEATHER_TYPE_SNOW = "Snow";  
}
```

WeatherActivity.java

```
public void updateUI() {  
    if (weatherReportList.size() > 0) {  
        DailyWeatherReport report = weatherReportList.get(0);  
  
        switch (report.getWeather()) {  
            case DailyWeatherReport.WEATHER_TYPE_CLOUDS:  
                weatherIconMini.setImageDrawable(getResources().getDrawable(R.drawable.cloudy));  
                weatherIcon.setImageDrawable(getResources().getDrawable(R.drawable.cloudy));  
                break;  
            case DailyWeatherReport.WEATHER_TYPE_RAIN:  
                weatherIconMini.setImageDrawable(getResources().getDrawable(R.drawable.rainy));  
                weatherIcon.setImageDrawable(getResources().getDrawable(R.drawable.rainy));  
                break;  
            default:  
                weatherIcon.setImageDrawable(getResources().getDrawable(R.drawable.sunny));  
                weatherIconMini.setImageDrawable(getResources().getDrawable(R.drawable.sunny));  
        }  
  
        weatherDate.setText("Today, May 1");  
        currentTemp.setText(Integer.toString(report.getCurrentTemp()));  
        lowTemp.setText(Integer.toString(report.getMinTemp()));  
        cityCountry.setText(report.getCityName() + ", " + report.getCountry());  
        weatherDescription.setText(report.getWeather());  
    }  
}
```

Probelauf

- Nun geben wir noch eine fixe Geolocation ein, um die App auch am Emulator testen zu können

```
public void downloadWeatherData(Location location) {  
    //final String fullCoords = URL_COORD + location.getLatitude() + "&lon=" + location.getLongitude();  
    final String fullCoords = „?lat=25.778&lon=-80.2018“;  
    ...  
}
```

```
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    ...  
  
    downloadWeatherData(null);  
}
```

Beim Aufruf brauchen wir keine Location mitzugeben, da diese fix ist

Probelauf





Noch
Fragen?