

Mobile Application Development

Higher Diploma in Science in Computer Science

Produced
by

Eamonn de Leastar (edeleastar@wit.ie)

Department of Computing, Maths & Physics
Waterford Institute of Technology

<http://www.wit.ie>

<http://elearning.wit.ie>

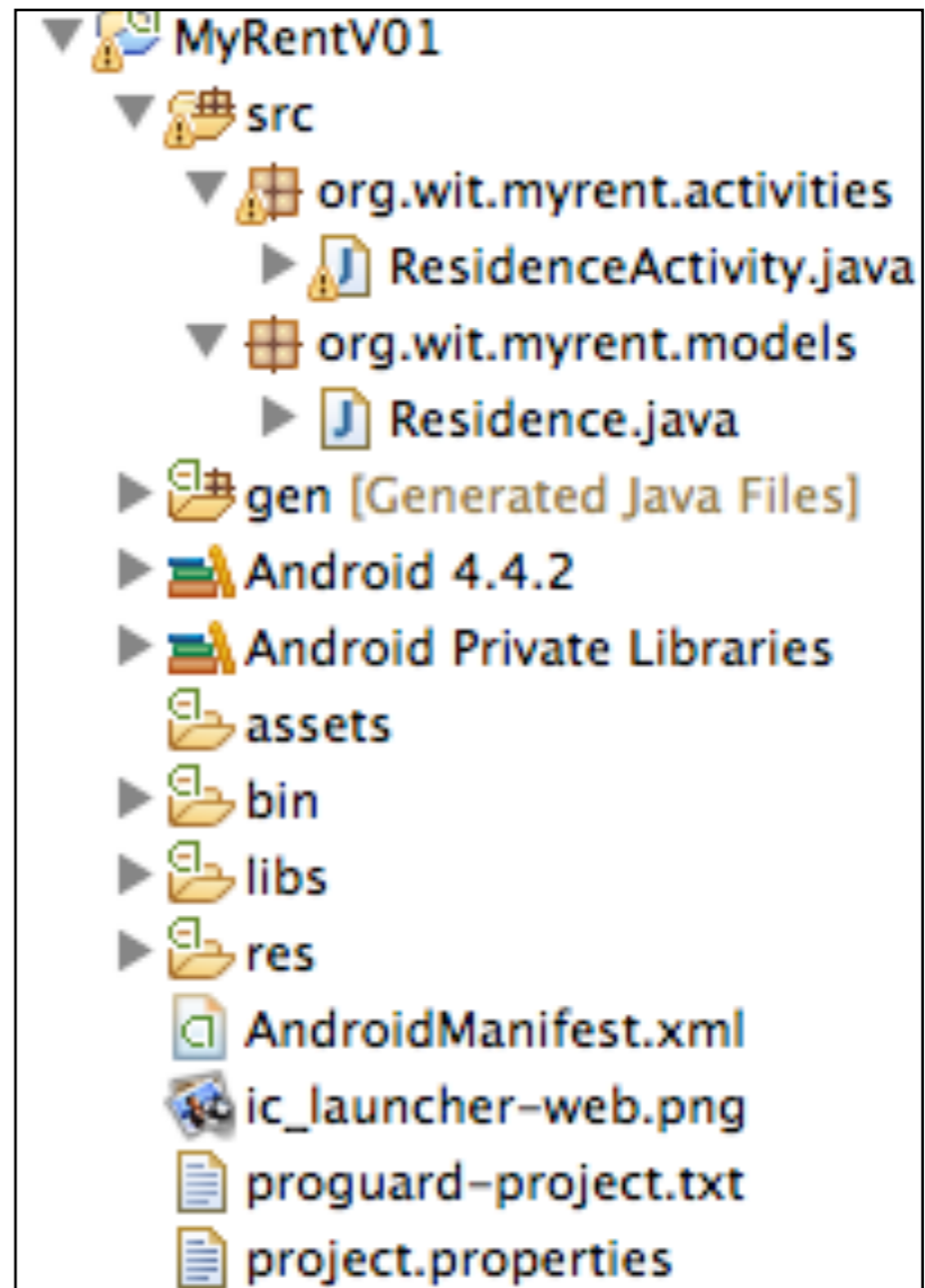
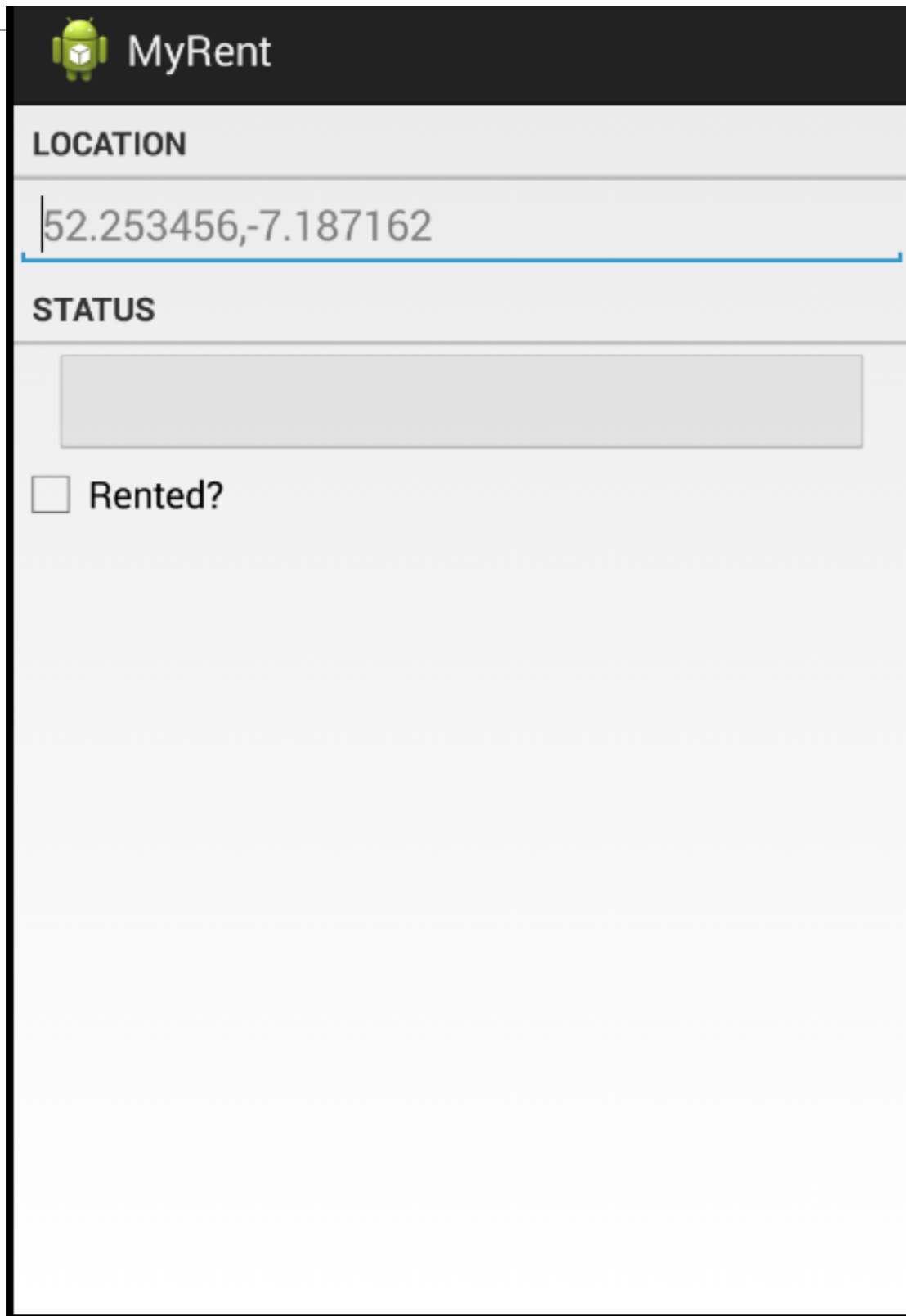


Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRCE

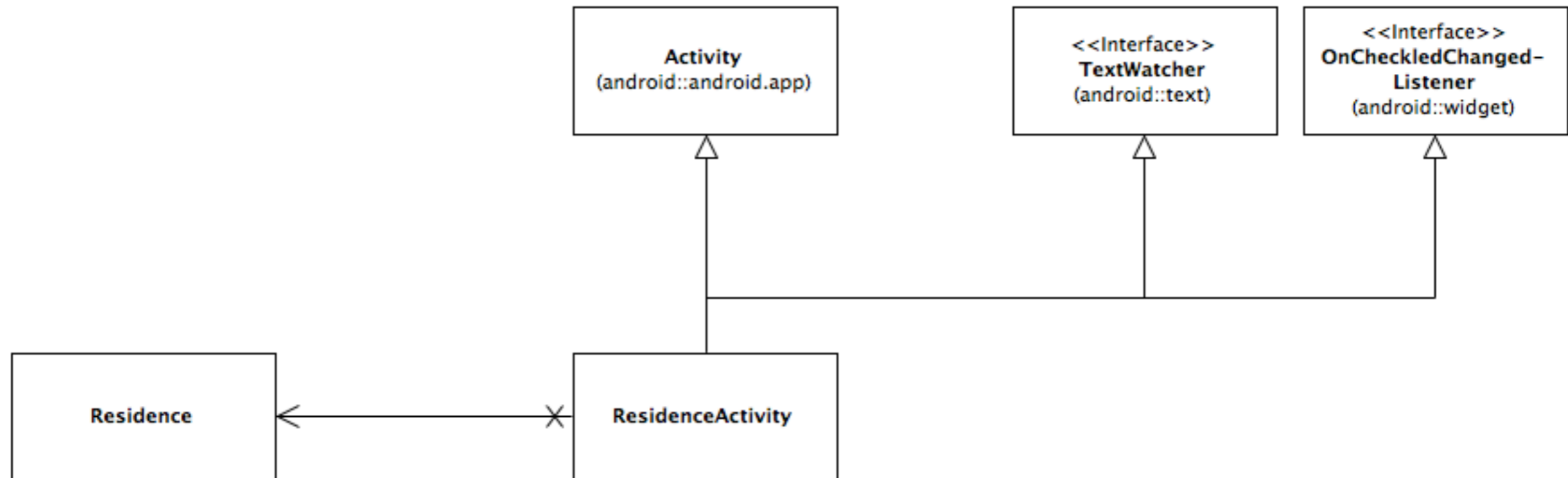


MyRent v01-03

Version 01



MyRent-v01 UML



Model Class

- UUID - a unique id for an object

```
public class Residence
{
    public UUID id;

    public String geolocation;
    public Date date;
    public boolean rented;

    public Residence()
    {
        id = UUID.randomUUID();
        this.date = new Date();
    }

    public String getDateString()
    {
        return "Registered: " + DateFormat.getDateTimeInstance().format(date);
    }
}
```

```

public class ResidenceActivity extends Activity implements TextWatcher, OnCheckedChangeListener
{
    private EditText geolocation;
    private CheckBox rented;
    private Button dateButton;
    private Residence residence;

    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_residence);

        geolocation = (EditText) findViewById(R.id.geolocation);
        dateButton = (Button) findViewById(R.id.registration_date);
        rented = (CheckBox) findViewById(R.id.isrented);

        residence = new Residence();

        geolocation.addTextChangedListener(this);
        geolocation.setText(residence.geolocation);
        dateButton.setEnabled(false);
        rented.setChecked(residence.rented);
        rented.setOnCheckedChangeListener(this);
    }
    @Override
    public void onCheckedChanged(CompoundButton arg0, boolean isChecked)
    {
        Log.i(this.getClass().getSimpleName(), "rented Checked");
        residence.rented = isChecked;
    }
    @Override
    public void afterTextChanged(Editable c)
    {
        Log.i(this.getClass().getSimpleName(), "geolocation " + c.toString());
        residence.geolocation = c.toString();
    }
    @Override
    public void beforeTextChanged(CharSequence arg0, int arg1, int arg2, int arg3)
    {}
    @Override
    public void onTextChanged(CharSequence arg0, int arg1, int arg2, int arg3)
    {}
}

```

ResidenceActivity

MyRent

LOCATION

52.253456,-7.187162

STATUS

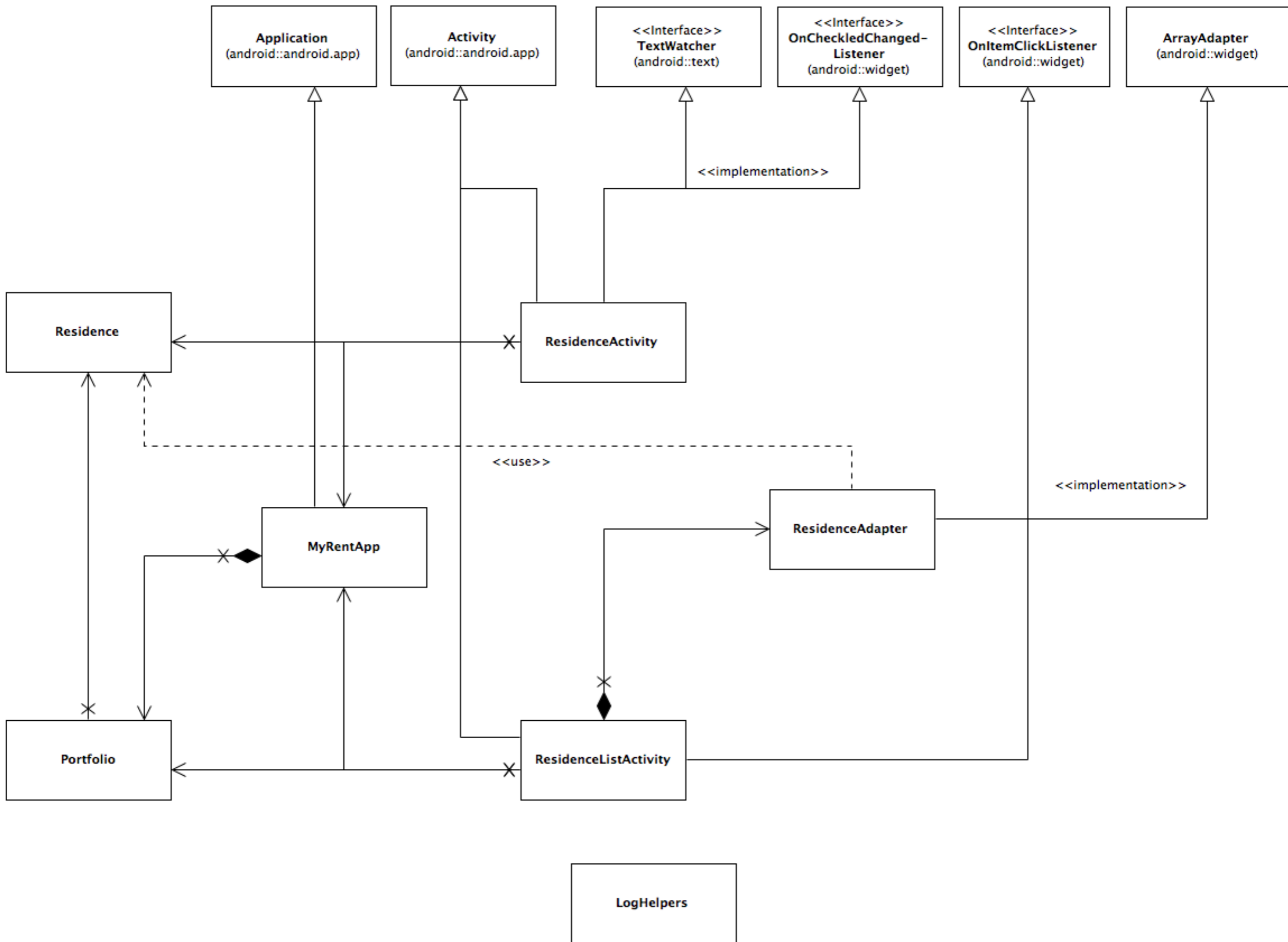
Rented?

MyRent v02

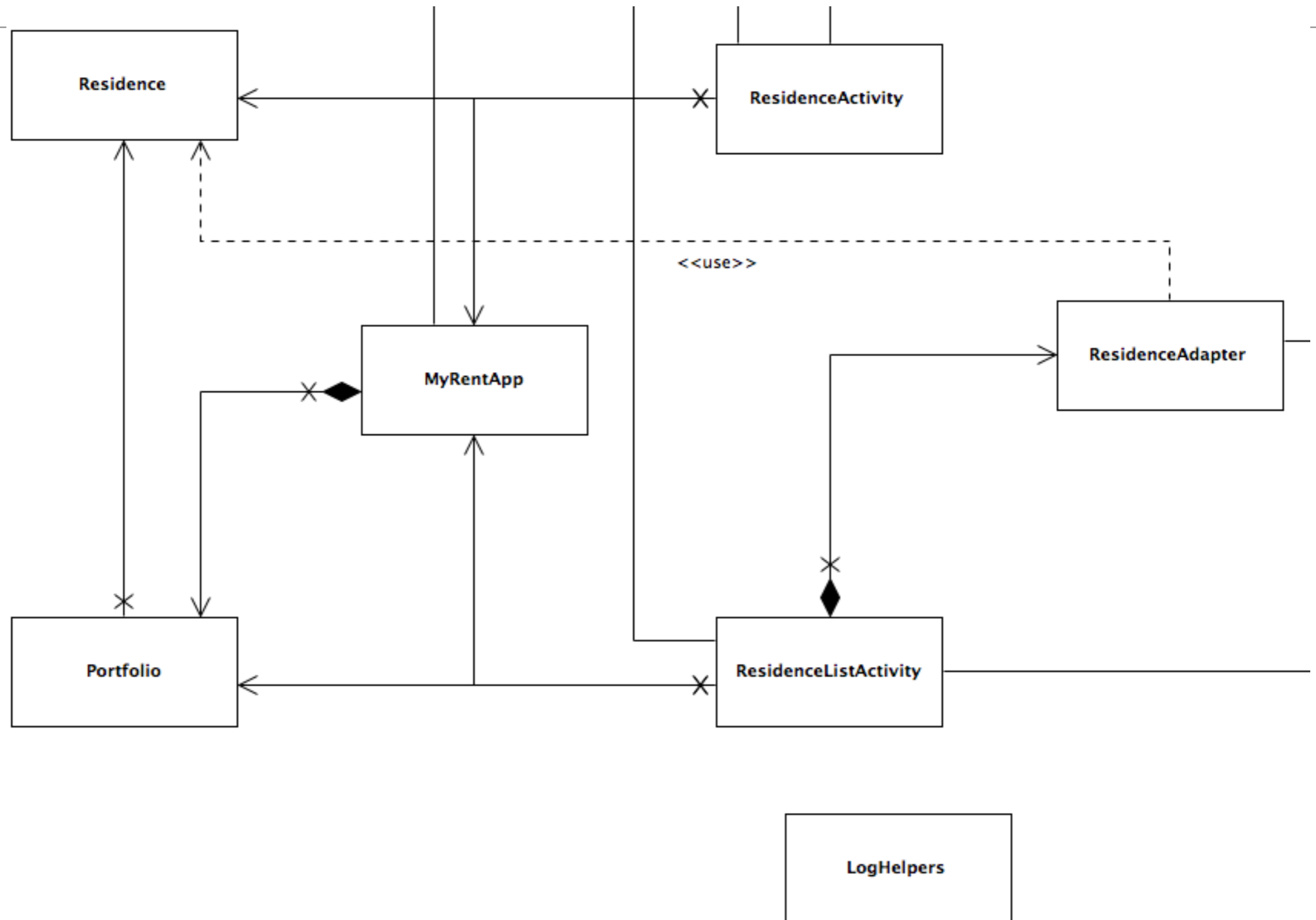
- MyRentV02
 - src
 - org.wit.android.helpers
 - LogHelpers.java
 - org.wit.myrent.activities
 - ResidenceActivity.java
 - ResidenceListActivity.java
 - org.wit.myrent.app
 - MyRentApp.java
 - org.wit.myrent.models
 - Portfolio.java
 - Residence.java
 - gen [Generated Java Files]
 - Android 4.4.2
 - Android Private Libraries
 - assets
 - bin
 - libs
 - res
 - AndroidManifest.xml
 - ic_launcher-web.png
 - proguard-project.txt
 - project.properties

MyRent	
52.253456, -7.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
53.253456, -8.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
54.253456, -9.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
55.253456, -10.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
56.253456, -11.187162	
Registered: Aug 27, 2014 1:20:35 PM	
57.253456, -12.187162	
Registered: Aug 27, 2014 1:20:35 PM	
58.253456, -13.187162	
Registered: Aug 27, 2014 1:20:35 PM	
59.253456, -14.187162	
Registered: Aug 27, 2014 1:20:35 PM	
60.253456, -15.187162	
Registered: Aug 27, 2014 1:20:35 PM	
61.253456, -16.187162	
Registered: Aug 27, 2014 1:20:35 PM	
62.253456, -17.187162	
Registered: Aug 27, 2014 1:20:35 PM	
63.253456, -18.187162	
Registered: Aug 27, 2014 1:20:35 PM	

MyRent	
LOCATION	
<input type="text" value="52.253456,-7.187162"/>	
STATUS	
<input type="text"/>	
<input type="checkbox"/> Rented?	



MyRent-v02 UML



Residence

```
public class Residence
{
    public UUID id;

    public String geolocation;
    public Date date;
    public boolean rented;

    public Residence()
    {
        id = UUID.randomUUID();
        this.date = new Date();
    }

    public String getDateString()
    {
        return "Registered: " + DateFormat.getDateTimeInstance().format(date);
    }
}
```

```

public class Portfolio
{
    public ArrayList<Residence> residences;

    public Portfolio()
    {
        residences = new ArrayList<Residence>();
        this.generateTestData();
    }

    public void addResidence(Residence residence)
    {
        residences.add(residence);
    }

    public Residence getResidence(UUID id)
    {
        Log.i(this.getClass().getSimpleName(), "UUID parameter id: "+ id);

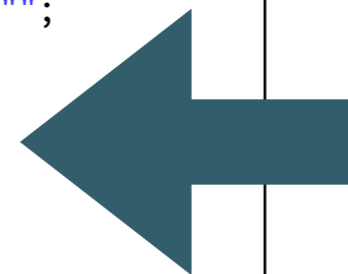
        for (Residence res : residences)
        {
            if(id.equals(res.id))
            {
                return res;
            }
        }
        info(this, "failed to find residence. returning first element array to avoid crash");
        return null;
    }

    private void generateTestData()
    {
        for(int i = 0; i < 100; i += 1)
        {
            Residence r = new Residence();
            r.geolocation = (52.253456 + i) % 90 + " " + (-7.187162 - i) % 180 + "";
            if(i%2 == 0)
            {
                r.rented = true;
            }
            else
            {
                r.rented = false;
            }
            residences.add(r);
        }
    }
}

```

Portfolio

- Generate random Residence objects to exercise UI



MyRentApp

```
public class MyRentApp extends Application
{
    public Portfolio portfolio;

    @Override
    public void onCreate()
    {
        super.onCreate();
        portfolio = new Portfolio();

        info(this, "RentControl app launched");
    }
}
```

- ‘Singleton’
Application object -
only on MyRentApp
created.
- Use this to create a
single portfolio object
for the application

ResidenceListActivity

```
public class ResidenceListActivity extends Activity implements OnItemClickListener
{
    private ListView listView;
    private Portfolio portfolio;
    private ResidenceAdapter adapter;

    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setTitle(R.string.app_name);
        setContentView(R.layout.activity_residencelist);

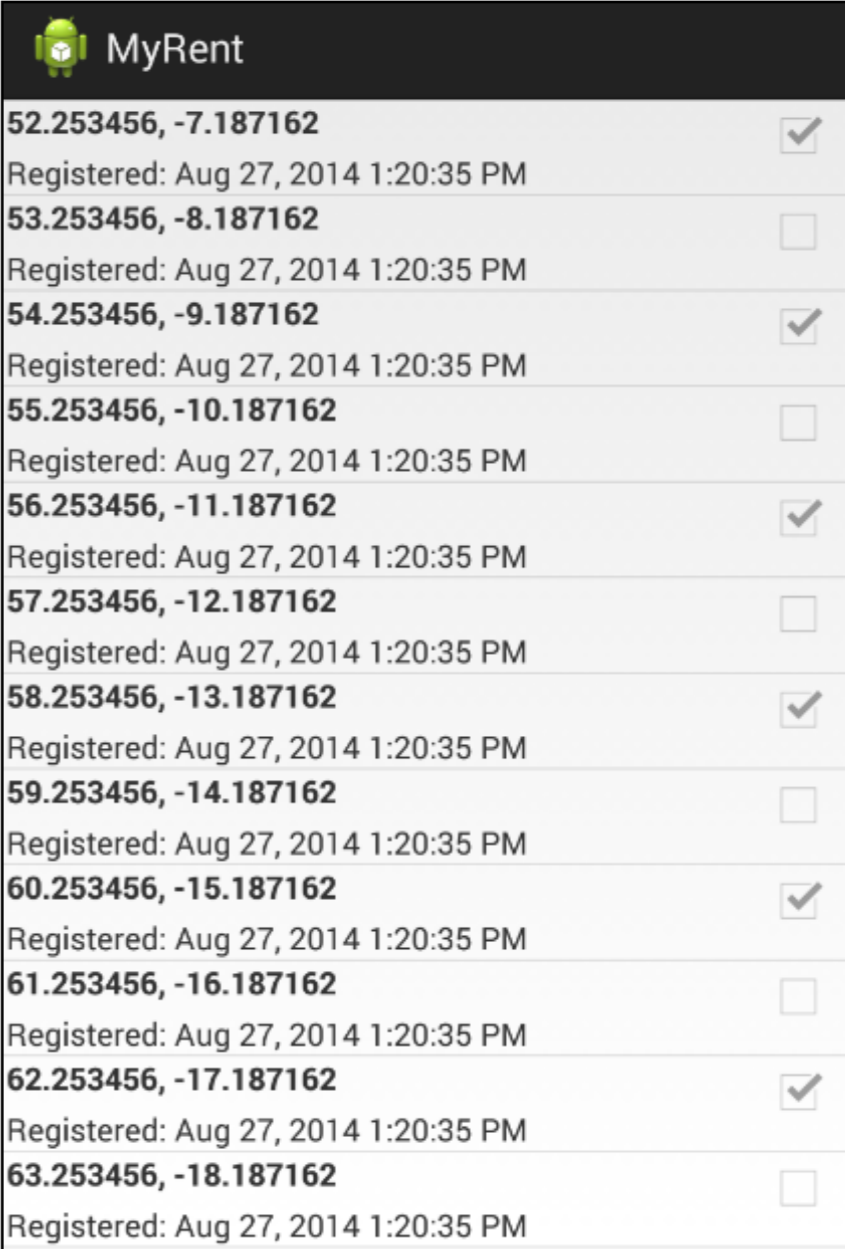
        listView = (ListView) findViewById(R.id.residenceList);

        MyRentApp app = (MyRentApp) getApplication();
        portfolio = app.portfolio;

        adapter = new ResidenceAdapter(this, portfolio.residences);
        listView.setAdapter(adapter);
        listView.setOnItemClickListener(this);
    }

    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position, long id)
    {
        Residence residence = adapter.getItem(position);
        Intent intent = new Intent(this, ResidenceActivity.class);
        intent.putExtra("RESIDENCE_ID", residence.id);
        startActivity(intent);
    }

    @Override
    public void onResume()
    {
        super.onResume();
        adapter.notifyDataSetChanged();
    }
}
```

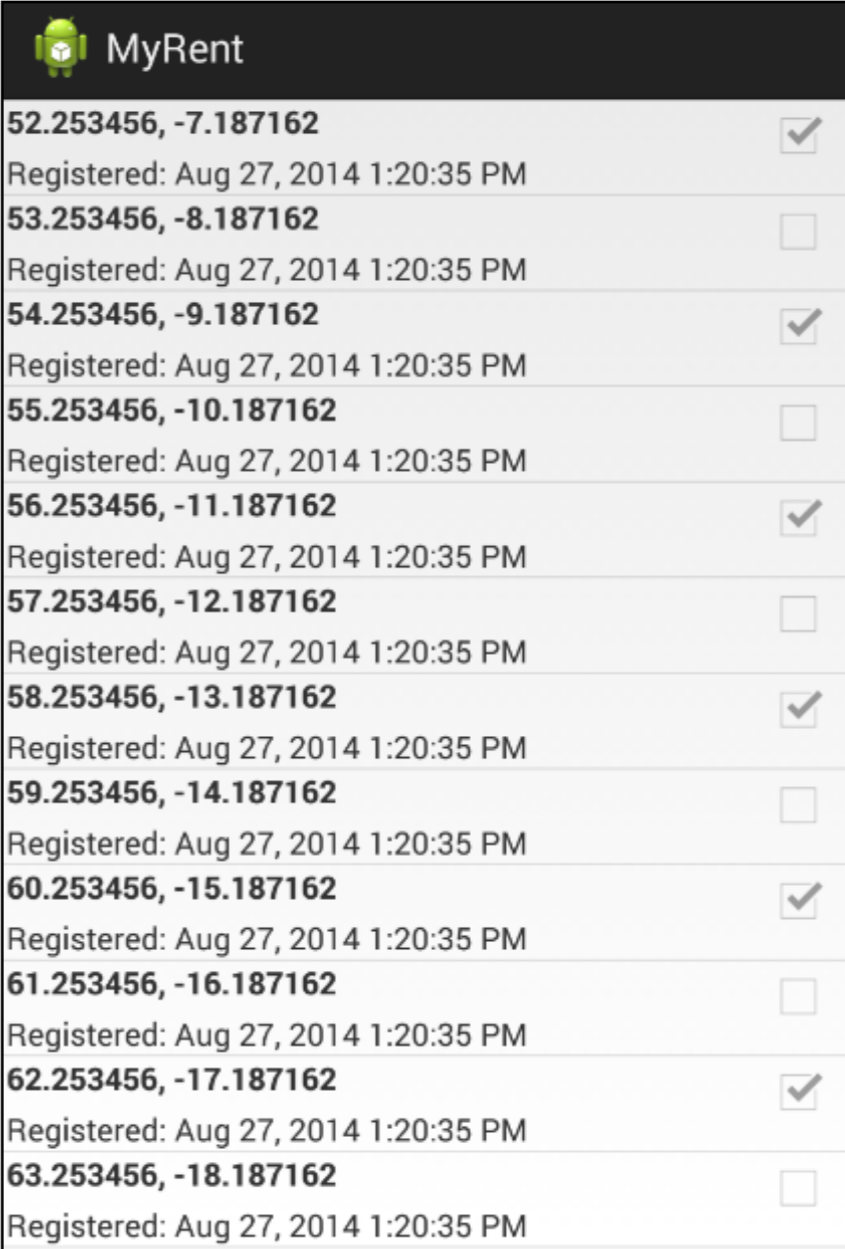


MyRent	
52.253456, -7.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
53.253456, -8.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
54.253456, -9.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
55.253456, -10.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
56.253456, -11.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
57.253456, -12.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
58.253456, -13.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
59.253456, -14.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
60.253456, -15.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
61.253456, -16.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
62.253456, -17.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
63.253456, -18.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	

ResidenceListActivity

```
//...
@Override
public void onItemClick(AdapterView<?> parent, View view, int position, long id)
{
    Residence residence = adapter.getItem(position);
    Intent intent = new Intent(this, ResidenceActivity.class);
    intent.putExtra("RESIDENCE_ID", residence.id);
    startActivity(intent);
}
//...
```

- Retrieve the Residence object by its position in the list
- Create a new Intent to start ResidenceActivity class
- Before starting it, put the ID of the object we retrieved into the 'extra' information passed to the intent



MyRent	
52.253456, -7.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
53.253456, -8.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
54.253456, -9.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
55.253456, -10.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
56.253456, -11.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
57.253456, -12.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
58.253456, -13.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
59.253456, -14.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
60.253456, -15.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
61.253456, -16.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
62.253456, -17.187162	<input checked="" type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	
63.253456, -18.187162	<input type="checkbox"/>
Registered: Aug 27, 2014 1:20:35 PM	

```

public class ResidenceActivity extends Activity implements TextWatcher,
OnCheckedChangeListener
{
    private EditText geolocation;
    private CheckBox rented;
    private Button dateButton;

    private Residence residence;
    private Portfolio portfolio;

    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_residence);

        geolocation = (EditText) findViewById(R.id.geolocation);
        dateButton = (Button) findViewById(R.id.registration_date);
        rented = (CheckBox) findViewById(R.id.isrented);

        geolocation.addTextChangedListener(this);
        dateButton.setEnabled(false);
        rented.setOnCheckedChangeListener(this);

        MyRentApp app = (MyRentApp) getApplication();
        portfolio = app.portfolio;

        UUID resId = (UUID) getIntent().getExtras().getSerializable("RESIDENCE_ID");
        residence = portfolio.getResidence(resId);
        if (residence != null)
        {
            updateControls(residence);
        }
    }
}
//...

```

ResidenceActivity

The screenshot shows the 'MyRent' app interface. At the top, there is a header with the Android logo and the text 'MyRent'. Below the header, the form is divided into two sections: 'LOCATION' and 'STATUS'. The 'LOCATION' section has a text input field containing the coordinates '52.253456,-7.187162'. The 'STATUS' section has a large, empty rectangular input field, followed by a checkbox labeled 'Rented?' which is currently unchecked.

ResidenceActivity

```
public class ResidenceActivity extends Activity implements TextWatcher,
    OnCheckedChangeListener
{
    //...
    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        //...
        UUID resId = (UUID) getIntent().getExtras().getSerializable("RESIDENCE_ID");
        residence = portfolio.getResidence(resId);
        if (residence != null)
        {
            updateControls(residence);
        }
    }

    public void updateControls(Residence residence)
    {
        geolocation.setText(residence.geolocation);
        rented.setChecked(residence.rented);
        dateButton.setText(residence.getDateString());
    }
}
//...
```



The screenshot shows the MyRent app interface. At the top, there is a header with the Android logo and the text "MyRent". Below the header, there are two sections: "LOCATION" and "STATUS". The "LOCATION" section has a text input field containing the coordinates "52.253456,-7.187162". The "STATUS" section has a text input field that is currently empty, and a checkbox labeled "Rented?" which is currently unchecked.

- Retrieve the ID from the 'Extra' information
- Use the ID to recover the actual Residence object from the portfolio
- Send this residence information to the controls

ResidenceActivity - Event Handlers

```
@Override
public void onCheckedChanged(CompoundButton arg0, boolean isChecked)
{
    Log.i(this.getClass().getSimpleName(), "rented Checked");
    residence.rented = isChecked;
}

@Override
public void afterTextChanged(Editable c)
{
    Log.i(this.getClass().getSimpleName(), "geolocation " + c.toString());
    residence.geolocation = c.toString();
}

@Override
public void beforeTextChanged(CharSequence arg0, int arg1, int arg2, int arg3)
{
}

@Override
public void onTextChanged(CharSequence arg0, int arg1, int arg2, int arg3)
{
}
}
```

ResidenceAdapter

```
class ResidenceAdapter extends ArrayAdapter<Residence>
{
    private Context context;

    public ResidenceAdapter(Context context, ArrayList<Residence> residences)
    {
        super(context, 0, residences);
        this.context = context;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent)
    {
        LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
        if (convertView == null)
        {
            convertView = inflater.inflate(R.layout.list_item_residence, null);
        }
        Residence res = getItem(position);

        TextView geolocation = (TextView) convertView.findViewById(R.id.residence_list_item_geolocation);
        geolocation.setText(res.geolocation);

        TextView dateTextView = (TextView) convertView.findViewById(R.id.residence_list_item_dateTextView);
        dateTextView.setText(res.getDateString());

        CheckBox rentedCheckBox = (CheckBox) convertView.findViewById(R.id.residence_list_item_isrented);
        rentedCheckBox.setChecked(res.rented);

        return convertView;
    }
}
```

LogHelper

- Occasionally introduce helpers to simplify / encapsulate awkward looking code.
- Helpers should usually be static - and should aim to make the clients of the helper easier to read

```
public class LogHelpers
{
    public static void info(Object parent, String message)
    {
        Log.i(parent.getClass().getSimpleName(), message);
    }
}
```

AndroidManifest

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="org.wit.myrent"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="16"
        android:targetSdkVersion="19" />

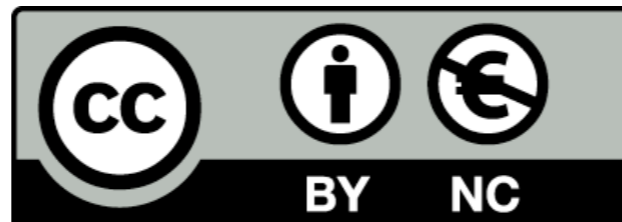
    <application
        android:name=".app.MyRentApp"
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >

        <activity
            android:name=".activities.ResidenceListActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity
            android:name=".activities.ResidenceActivity"
            android:label="@string/app_name" >
        </activity>

    </application>

</manifest>
```



Except where otherwise noted, this content is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

For more information, please see <http://creativecommons.org/licenses/by-nc/3.0/>

