

# Collector

## Abschluss Präsentation - Semesterarbeit

Mathias Weigert

Zürcher Hochschule für angewandte Wissenschaften

22. Juni 2016

# Inhaltsverzeichnis

1 Ausgangssituation

2 Konzept

3 Anforderungen

- App
- Datenbank

4 Umsetzung

- Allgemeines
- Code Design
- View

## Ausgangssituation

Immer wieder komme ich in die Situation das ich auf Flohmärkten und in Brockenhäusern Bücher, Filme oder Konsolenspiele kaufe nur um später zu Hause festzustellen, dass ich diese bereits besitze.

Deshalb habe ich nach einer Lösung gesucht, Medien aus diesen Kategorien einfach zu verwalten.

Im Google Play Store konnte ich die ein oder andere App finden, welche Sammlungen aus einer der Kategorien verwalten kann, aber keine welche in der Lage ist Bücher, Filme und Konsolenspiele zu verwalten.

# Ausgangssituation

Name	Bücher	Filme	Spiele	Scanner
My books	Ja	Nein	Nein	Ja
Meine Bücher Pro	Ja	Nein	Nein	Ja
Codex	Ja	Nein	Nein	Ja
Movie Collection	Nein	Ja	Nein	Ja
Movielicious	Nein	Ja	Nein	Ja
Nintendo Collection	Nein	Nein	Ja (NES)	Nein

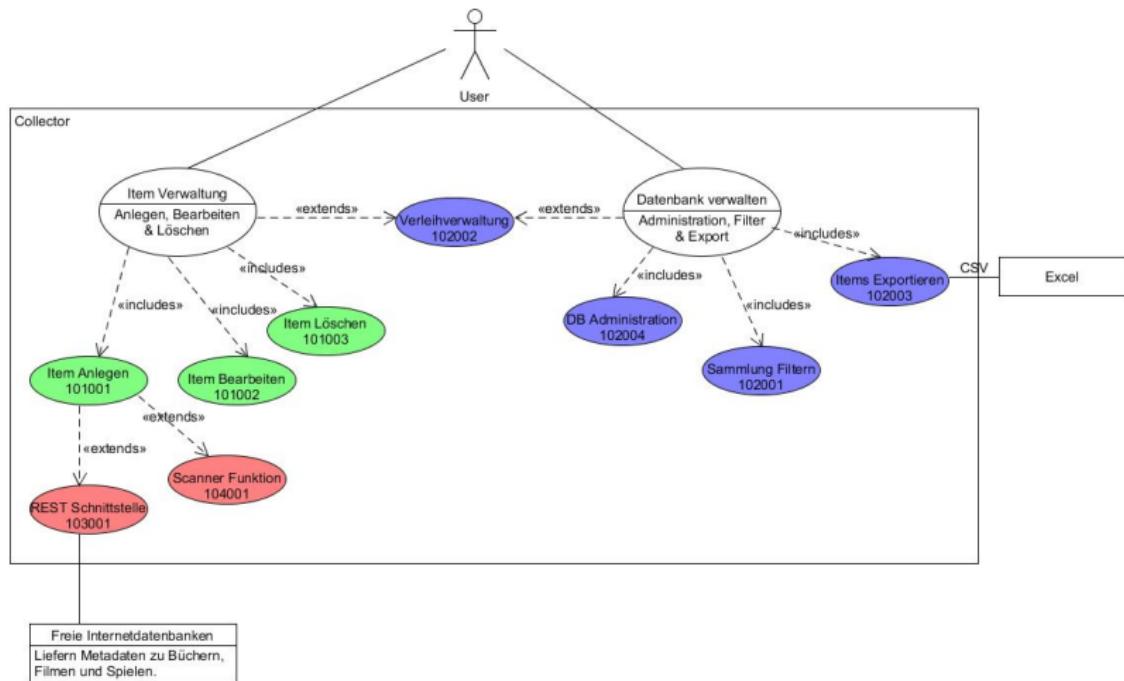
Name	Export	Verleih	Preis	Rating
My books	Ja	Ja	Gratis	3.7
Meine Bücher Pro	CSV	Nein	CHF 3.65	4.0
Codex	CSV, XML	Ja	Gratis	4.2
Movie Collection	CSV, XML	Ja	CHF 2.15	4.5
Movielicious	Ja	Ja	CHF 2.50	3.2
Nintendo Collection	Nein	Nein	CHF 1.47	5.0

# Konzept

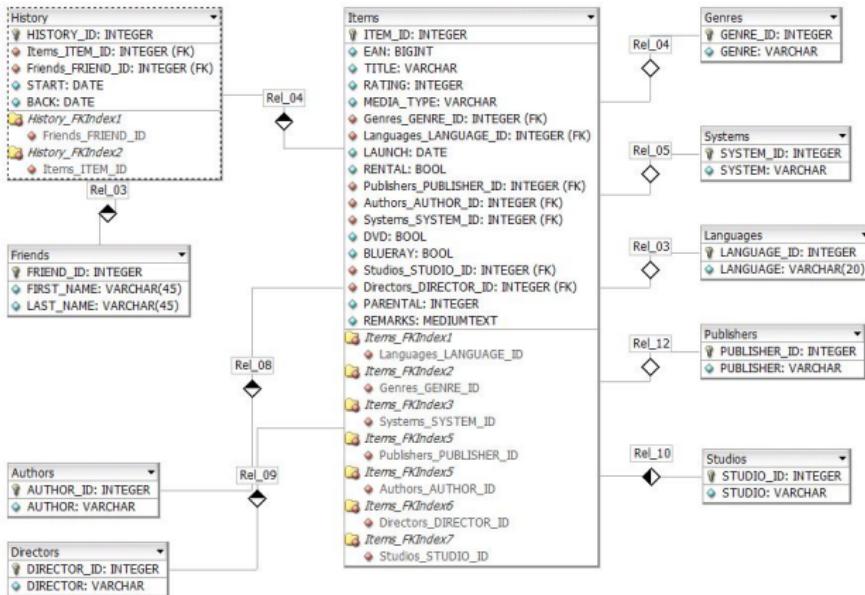
Anhand der Recherche wurde entschieden eine schlichte und intuitiv zu Bedienende App zu erstellen.

Welche den User bei der Verwaltung seiner Bücher, Spiele und Filme optimal unterstützt.

# Anforderungen: App



# Anforderungen: Datenbank



# Tools & System

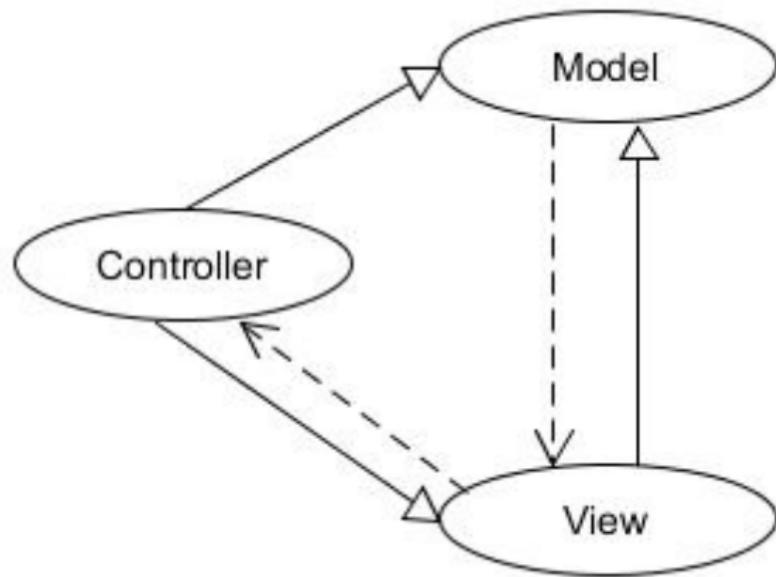
## Tools

- Code - Android Studio 1.5.1
- Dokumentation - TeXstudio 2.10.4
- UML Design - UMLet 14.2
- DB Design - DBDesigner 4

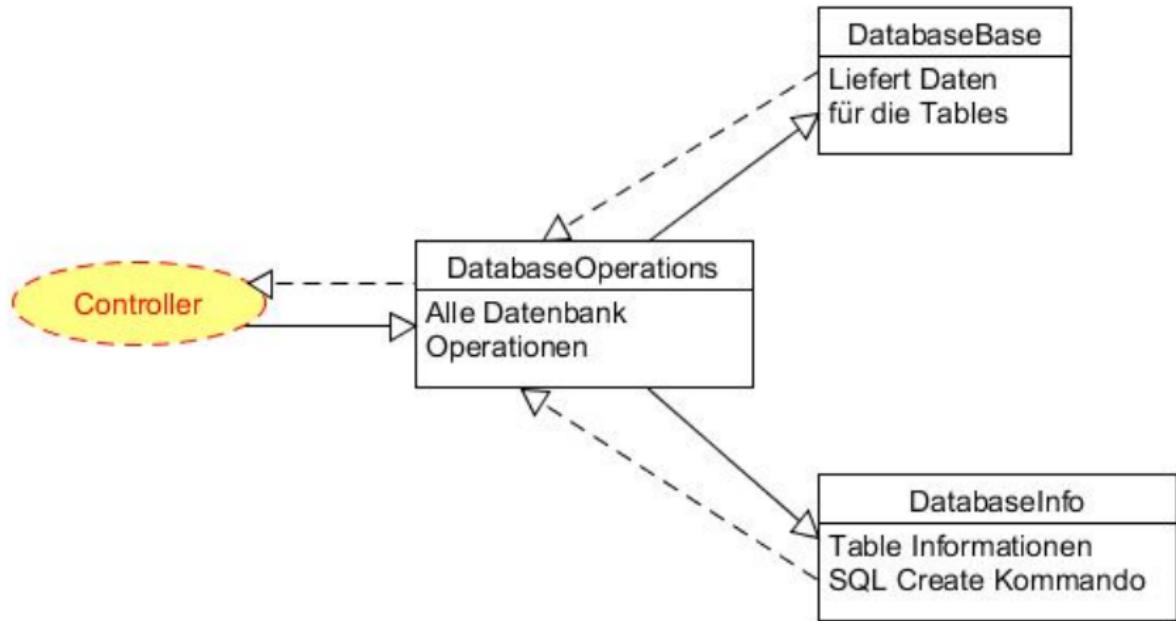
## Systemvoraussetzung (App)

- Android Version 6.0 (API 23)
- Zugriff: Netzwerk
- Zugriff: Internet
- Zugriff: Externer Speicher

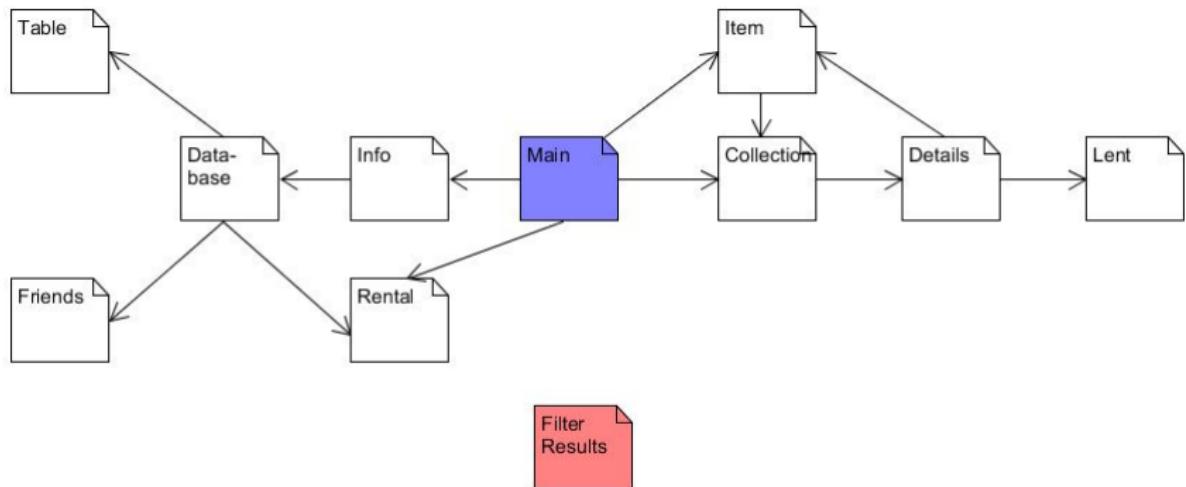
# MVC Pattern



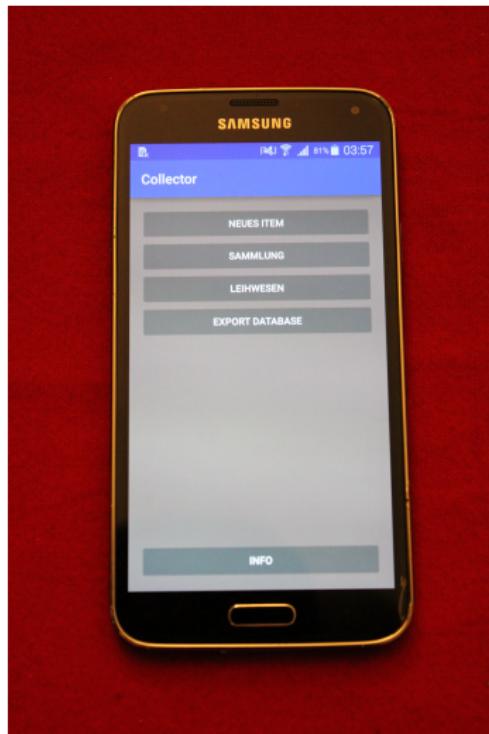
# Model



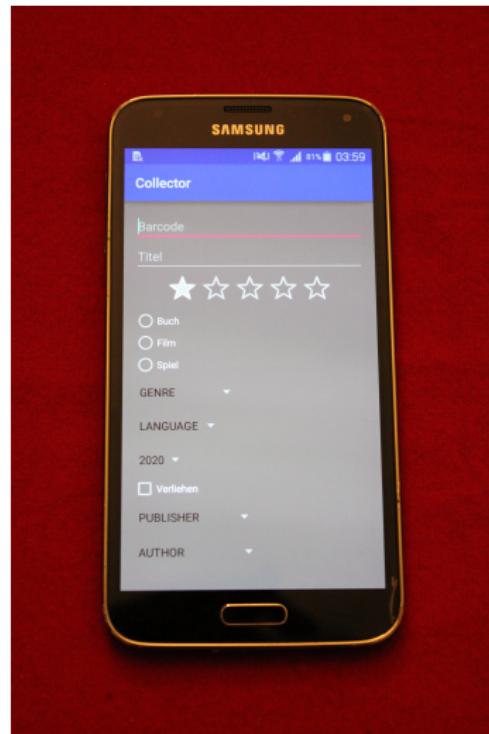
# View Steuerung



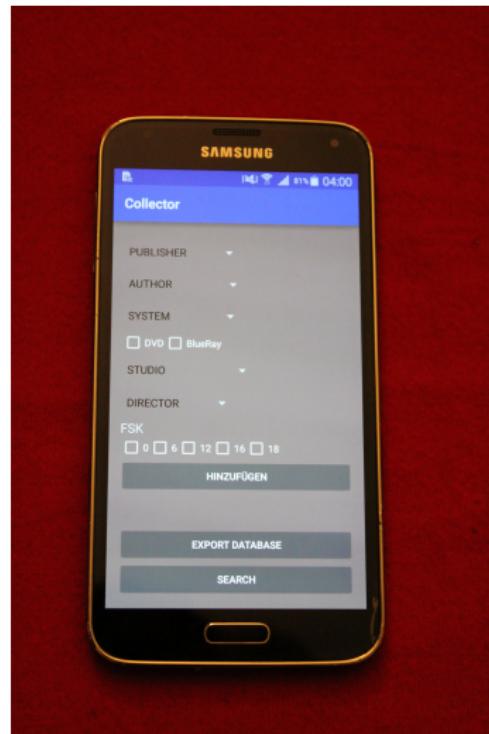
# Startbildschirm



# Neues Item/Filter/Export



# Neues Item/Filter/Export



# Button

Listing 1: View: Button

```
1 <Button  
2     android:id="@+id/btn_mng_author"  
3     android:layout_width="match_parent"  
4     android:layout_height="wrap_content"  
5     android:onClick="buttonOnClick"  
6     android:text="@string/btn_mng_author"/>
```

# Button Listener

Listing 2: Controller: Button Listener

```
//Button listener for the database activity
2    public void buttonOnClick(View v) {
3        switch (v.getId()) {
4            case R.id.btn_mng_author:
5                //Button author pressed by user.
6                if (debugMode) {
7                    Log.d("USERACTION", "Entering author administration.");
8                }
9                final Intent authorIntent = new Intent(this,
10                                              TableActivity.class);
11                authorIntent.putExtra("debugMode", debugMode);
12                authorIntent.putExtra("tableName", "AUTHOR");
13                startActivityForResult(authorIntent);
14                break;
15            }
16        }
17    }
18}
```

# Controller: ListView

Listing 3: Controller: ListView

```
// Controller einer ListView
2  if (anz > 0) {
3      int indexItemId = crs.getColumnIndex(DatabaseInfo.ITEMS_ID_COL);
4      do {
5          int id = crs.getInt(indexItemId);
6          Item item = new Item(ctx, id);
7          items.add(item);
8      } while (crs.moveToNext());
9
10     final ArrayAdapter adapter = new ArrayAdapter(this,
11             android.R.layout.simple_list_item_1, items);
12
13     list.setAdapter(adapter);
14 }
15
16 // Class: Item
17 @Override
18     public String toString() {
19         return title;
20     }
```

# Noch Fragen

