

Meine App

Badi-App

Version 1.0.0, 29. April 2019 | Remy Stettler, Flynn Gassmann

Inhalt

[1 Abstract (Kurzbeschreibung) 2](#_Toc8211228)

[2 Konkurrenzanalyse 3](#_Toc8211229)

[2.1.1 Konkurrenz Apps 3](#_Toc8211230)

[2.1.2 Wie kann sich unser App von den Konkurrenten abheben? 3](#_Toc8211231)

[3 User Stories 4](#_Toc8211232)

[4 Mockups 5](#_Toc8211233)

[4.1.1 Änderungen 5](#_Toc8211234)

[5 Technische Realisierung 6](#_Toc8211235)

[5.1.1 MainActivity 6](#_Toc8211236)

[5.1.2 BadiDetailsActivity 6](#_Toc8211237)

[5.1.3 BadiLocationActivity 6](#_Toc8211238)

[5.1.4 Badi 6](#_Toc8211239)

[5.1.5 Becken 6](#_Toc8211240)

[5.1.6 WieWarmJsonParser 6](#_Toc8211241)

[5.1.7 BadiDao 6](#_Toc8211242)

[6 Testing 7](#_Toc8211243)

[6.1 Manuelle UI-Tests 7](#_Toc8211244)

[6.2 Testauswertung 9](#_Toc8211245)

[7 Fazit 10](#_Toc8211246)

[7.1.1 Was lief gut/schlecht? 10](#_Toc8211247)

[7.1.2 Wie seid ihr mit dem Endergebnis zufrieden? 10](#_Toc8211248)

[7.1.3 Was habt ihr gelernt? 10](#_Toc8211249)

[7.1.4 War alles vorhanden oder was fehlte noch? 10](#_Toc8211250)

# Abstract (Kurzbeschreibung)

In unserem Projekt geht es um eine App, welche Schwimmbäder auflistet und anzeigt. Der User bekommt eine Liste von Schwimmbädern, welche er dann auch noch nach Badi filtern kann. Er kann auf ein beliebiges Element klicken, welches dann in einer Einzelansicht erscheint. In der Einzelansicht findet man die Temperatur der Schwimmbecken und vom Ort der Badi. Den Ort kann man sich dann auch auf einer Karte ansehen.

# Konkurrenzanalyse

### Konkurrenz Apps

#### Schwimmbad-Finder

Die WebApp Schwimmbad-Finder ist eine Konkurrenz-App: <https://www.chip.de/downloads/webapp-Schwimmbad-Finder_164052336.html>  
**Positiv:** Schöne Webseite und Auflistung der Schwimmbäder. Gute Suchfunktion. **Negativ:** Nicht alle Schwimmbäder haben ein Foto vorhanden.

#### Google Suche

Google-Suche ist zwar direkt nicht eine App, jedoch zeigt Google Schwimmbäder in der Nähe an mit den Angaben zum Schwimmbad an: <https://www.google.com/search?rlz=1C1GCEB_enCH846CH846&ei=b7vGXNrjC8KcsAfSgaSgAg&q=schwimmbad&oq=schwimmbad&gs_l=psy-ab.3..0i67l8j0l2.385094.385252..385430...0.0..0.114.182.1j1......0....1..gws-wiz.......0i71.qps2WHbB6Nc>  
**Positiv:** Schwimmbäder in der Nähe werden automatisch angezeigt. Man kann einfach «schwimmbad» in der Suche eingeben und es werden von Google automatisch Ergebnisse angezeigt. **Negativ:** Ist nicht direkt eine App.

#### Badi-Info

Badi-Info ist auch ein Web-App. Jedoch nach meiner Meinung ist sie nicht so schön gestaltet, jedoch ist es eine Konkurrenz: <https://www.badi-info.ch/schwimmbad-karten.html>  
**Positiv:** Auflistung von den Schwimmbäder Gruppiert in Kantone/Städte vorhanden.  
**Negativ:** Die Webseite sieht nicht so schön aus. Verwendet Google-Maps für die Automatischen Ergebnisse.

### Wie kann sich unser App von den Konkurrenten abheben?

Unsere Badi-App kann sich durch ein kompaktes und schönes Design, von den anderen Apps abheben. Das App soll auch nicht zu Komplex sein zu bedienen.

# User Stories

1. Als Benutzer möchte ich in der Suchleiste eine Badi eingeben, sodass die gesuchte Badi angezeigt wird.
2. Als Benutzer möchte ich eine Liste bekommen, auf welcher ich dann ein Schwimmbad-Element auswählen kann.
3. Als Benutzer möchte ich auf der Einzelansicht des Schwimmbads die Temperatur der Schwimmbecken und des Ortes sehen.
4. Als Benutzer möchte ich auf einen Button klicken können, wo mir dann den Ort vom Schwimmbad angezeigt wird.
5. Als Benutzer möchte ich den Ort des Schwimmbads auf einer Karte sehen, sodass ich die Stadt sehen kann in welcher sich die Badi befindet.
6. Als Benutzer möchte ich die Ansicht mit der Karte mit einem Zurück-Button verlassen können und wieder zurück auf die Einzelansicht gelangen.
7. Als Benutzer möchte ich die Einzelansicht mit einem Zurück-Button verlassen können und auf die Listenansicht mit den Schwimmbädern gelangen.
8. Als Benutzer möchte ich eine Fehlermeldung bekommen, wenn ich keine Internet-Verbindung auf meinem Android-Smartphone habe, sodass ich merke, dass ich eine Internet-Verbindung herstellen muss.
9. Als Benutzer möchte Fehlermeldungen bestätigen können, sodass die Fehlermeldung verschwindet.
10. Als Benutzer möchte ich auf der Karte hinein- und herauszoomen können, damit ich mir den Ort gut ansehen kann.

# Mockups

BadiLocationActivity

BadiDetailActivity

MainActivity

### Änderungen

Die Searchbar auf der Main Page ist nun nicht mehr unter der Toolbar, sondern man findet sie in der Toolbar als Icon.

In der Einzelansicht befindet sich kein Wetter-Icon unter der Toolbar, aber zusätzlich sieht man noch eine Beliebige Information, welche von der Badi angegeben wurde.

Die Karte führt nicht direkt zum Ort der Badi, sondern zur Stadt der Badi. Es hätte uns zu viel Zeit beansprucht, dies noch zu implementieren.

In der Searchbar kann man nicht mehr nach Ort, sondern nach dem Namen filtern.

# Technische Realisierung

### MainActivity

Die MainActivity ist die Hauptansicht des Projekts. Auf der MainActivity wird eine Liste angezeigt mit Badi-Elementen. Mit einem Klick auf ein Element in der Liste wird man auf die Einzelansicht des gewählten Elements geleitet.

### BadiDetailsActivity

Die BadiDetailsActivity ist die Einzelansicht des Projekts. Auf dieser Activity wird die ausgewählte Badi von der MainActivity, detailliert angezeigt mit mehreren Informationen, wie zum Beispiel die Temperatur des Ortes von der Badi, die Temperatur der Becken von der Badi und eine Information von der Badi. Auf der Activity befindet sich auch noch ein Button der auf die BadiLocationActivity weiterleitet.

### BadiLocationActivity

Auf der BadiLocationActivity wird eine Karte, welche auf das Zentrum der Stadt von der Badi gerichtet ist, angezeigt.

### Badi

Die Klasse Badi ist das Objekt für die Badi, sie hat folgende Attribute:

* id : int
* name : string
* kanton : string
* ort : string
* becken : List<Becken>
* adresse : string
* information : string

### Becken

Die Becken sind auch Objekte und sind am relevantesten für die Einzelansicht. Die Klasse hat folgende Attribute:

* name : string
* temperatur : double
* status : string

### WieWarmJsonParser

Die Klasse WieWarmJsonParser wird dafür verwendet um die API-Abfragen zu verarbeiten.

### BadiDao

Die Klasse BadiDao ist eigentlich die «Datenbank» von der Applikation. Die Klasse beinhaltet eine Methode, welche die Badi-Objekte erstellt.

# Testing

## Manuelle UI-Tests

Im Projekt befinden sich noch 5 Unit-Tests und 3 Espresso Tests.

|  |  |
| --- | --- |
| Abschnitt | Innhalt |
| ID | ST-01 |
| Anforderungen | Userstory 1 |
| Vorbedingungen | Die Suchfunktion funktioniert und gibt Ergebnisse zurück. |
| Ablauf | 1. Die App wird gestartet und die Startliste wird angezeigt 2. Auf das Such-Icon klicken 3. Die gewünschte Badi eingeben |
| Erwartetes Resultat | Die Liste wird nach dem Wunsch des Benutzers aktualisiert. |

|  |  |
| --- | --- |
| Abschnitt | Innhalt |
| ID | ST-02 |
| Anforderungen | Userstory 2/3 |
| Vorbedingungen | In der Listenansicht ist es möglich auf eine gewünschte Badi zu klicken und somit in die Einzelansicht hinüber zu gehen. |
| Ablauf | 1. Die Liste der Badis wird angezeigt 2. Wenn ich nun auf eine der vielen Badis klicke werde ich zur Einzelansicht verlinkt |
| Erwartetes Resultat | Die Einzelansicht mit weiteren Infos und Details zu jeder Badi wird angezeigt |

|  |  |
| --- | --- |
| Abschnitt | Innhalt |
| ID | ST-03 |
| Anforderungen | Userstory 6/7 |
| Vorbedingungen | Es sind funktionierende Backbuttons vorhanden |
| Ablauf | 1. Navigieren sie zur Location-Ansicht und drücken Sie dort den Back-Button in der Toolbar. 2. Sie wurden erfolgreich zur Einzelansicht zurückgeleitet. 3. Drücken Sie in der Einzelansicht den Back-Button in der Toolbar. 4. Sie wurden erfolgreich zur Listenansicht weitergeleitet und sehen die Badis. |
| Erwartetes Resultat | Man wird durch die Back Buttons jeweils eine Seite zurück transportiert |
| Abschnitt | Innhalt |
| ID | ST-04 |
| Anforderungen | Userstory 8/9 |
| Vorbedingungen | Wenn der User keine Internetverbindung hat kommt eine entsprechende Fehlermeldung. |
| Ablauf | 1. Versuchen sie zur Einzelansicht einer Badi zu Navigieren OHNE Internetverbindung. 2. Es erscheint eine Fehlermeldung welche mitteilt, dass zurzeit keine Daten gelesen werden können. 3. Wenn man die Fehlermeldung bestätigt wird diese geschlossen und man wird eine Seite zurückgeleitet. |
| Erwartetes Resultat | Eine Funktionierende Fehlermeldung die man auch schliessen kann |

|  |  |
| --- | --- |
| Abschnitt | Innhalt |
| ID | ST-05 |
| Anforderungen | Userstory 5/6 |
| Vorbedingungen | In der Einzelansicht kann man mit einem Button zu einer Location Ansicht gelangen in der man den genauen Standort der Badi sieht |
| Ablauf | 1. Man wählt zuerst eine Badi in der Liste aus und klickt darauf 2. In der Einzelansicht kann man nun auf einen weiteren Button klicken um zu einer Location Ansicht zu gelangen 3. Da wird dann schliesslich eine Karte mit dem Standort der Stadt in welcher die Badi ist, angezeigt |
| Erwartetes Resultat | Die dritte Ansicht der App, die Location Ansicht mit einer Karte des Standorts der Stadt von der Badi wird geöffnet. |

Alle Tests wurden erfolgreich ausgeführt von Vinicius.

## Testauswertung

Es hab keine Fehlerhafte UI-Tests

|  |  |  |
| --- | --- | --- |
| ID | Erfolgreich | Bemerkungen |
|  |  |  |

# Fazit

### Was lief gut/schlecht?

+Das Testing

+Die Listenansicht

+Die Einzelansicht

+Der PDF Auftrag

- Die Location Ansicht

- Zu wenig Zeit

### Wie seid ihr mit dem Endergebnis zufrieden?

Eigentlich gut, wir sind den Mockups mehr oder weniger treu geblieben und konnten alles Geplante auch umsetzten. Es gab jedoch ein wenig Zeitdruck und wenn man mehr Zeit gehabt hätte, hätte man sicher eine viel spektakulärere App kreieren können.

### Was habt ihr gelernt?

* Wie man mit Android Studios arbeitet
* Wie man sich in Android Studios zurechtfindet
* Wie man mit Java eine Android basierte App erstellen kann.
* Ein kleiner refresh zu wie man mit Github zusammenarbeiten kann.
* Wie man Automatisierte Espresso Test erstellt
* Wie man mit einem Virtuellen Android Emulator arbeitet
* Was Manuele UI Tests sind
* Wie man mit Gradle Scripts arbeitet
* Wie man in Android Studios eine Java API integriert
* Was in Java eine Activity und ein Intent ist und was der LifeCycle von einem Activity ist.
* Wie man mit Java eine Webconnection herstellt
* Was Datenpersistenz ist

### War alles vorhanden oder was fehlte noch?

Eigentlich haben wir alles was wir mit den Mockups geplant haben auch umgesetzt, allerdings mussten wir im Nachhinein ein paar Sachen verändern wie z.B. konnten wir keine Google Maps in unsere App integrieren, sondern eine Osmdroid Map. Oder die Internet Connection Fehlermeldung ist auch nicht genau, dass was wir in den Mockups geschrieben hatten.