

Meine App

App-Name

Version 1.0.0, 1. Dezember 2015 | Vorname Nachname des Autors

Inhalt

[1 Abstract (Kurzbeschreibung) 2](#_Toc510016994)

[2 Mockups 3](#_Toc510016995)

[3 Technische Realisierung 4](#_Toc510016996)

[4 Testing 5](#_Toc510016997)

[4.1 Manuelle UI-Tests 5](#_Toc510016998)

[4.2 Testauswertung 6](#_Toc510016999)

[5 Fazit 7](#_Toc510017000)

# Abstract (Kurzbeschreibung)

Im Modul-335 haben wir den Auftrag bekommen, in 2er Teams eine Android App zu programmieren. Die Rahmenbedingungen waren, dass jedes Teammitglied an allen Teilen des Projekts arbeitet und die Applikation mindestens 3 verschiedene Activities und eine komplexe Komponente beinhaltet. Bei der komplexen Komponente handelt es sich um den Gebrauch einer API oder eines Sensors zum Beispiel.

Unser Ziel war es natürlich eine möglichst gute Note zu erreichen, weshalb wir eine App mit einem Sensor programmierten. Wir haben als Sensor das Mikrofon gewählt, weil wir interessiert daran waren eine Art Stimmgerät zu programmieren, welche den Pitch eines aufgenommenen Tons anzeigt, da wir beide Gitarre spielen. Und so entstand die Idee hinter AudioGraph.

# Mockups

1. **Startactivity**Innerhalb der StartActivity wird dem Benutzer zentral ein Login-Formular präsentiert. Oberhalb des Formulars wird das Logo der App platziert. Die Eingabefelder sollten möglichst in den oberen zwei Drittel des Bildschirms platziert werden damit die Tastatur diese nicht überdeckt.  
   Unter den Eingabefelder sind zwei Buttons platziert, «Login» und «Registrieren». Dabei soll der Login-Button farblich hervorgehoben werden da dieser öfters benutzt wird. Der «Registrieren»-Button wird im Normalfall einmal benutzt deshalb soll dieser neutral oder sogar weniger prominent dargestellt werden.
2. **Badi-Galerie**In der «Badi-Galerie»-Ansicht wird dem Benutzer ein Grid mit den favorisierten Schwimmbäder angezeigt. Jedes einzelne Schwimmbad wird mit einem Bild präsentiert. In der in der unteren Ecke jedes Bildes wird die aktuelle Temperatur angezeigt. Darunter den Namen der Badi und der Ort. In der Auflistung muss auf und ab navigiert werden können da man mehr Schwimmbäder hinzufügen kann als auf dem Display Platz haben. Ein Floating-Action-Button mit einem Plus-Icon ist unten rechts am Screen platziert um neue Schwimmbäder in die eigene Liste aufzunehmen. In der Actionbar am oberen Rand wird rechts ein Kontextmenü platziert welches mit dem entsprechenden Button geöffnet werden kann. Darin sind die Punkte Einstellungen, Hilfe und Logout zu finden.
3. **Badidetails**…
4. **Badiauswahl**…
5. **Registration**…

# Technische Realisierung

Beschreibt hier, wie ihr eure komplexe Komponente technisch umgesetzt habt. Zur Darstellung der technischen Umsetzung wird ein UML-Diagramm empfohlen, welches zusätzlich in Textform beschrieben wird. Erklärt kurz die wichtigsten Klassen und Methoden und deren Zusammenspiel. Eine Fachperson, welche dieses Kapitel liest, sollte schnell nachvollziehen können, wie die externe Komponente realisiert wurde.

# Testing

## Manuelle UI-Tests

In diesem Kapitel definiert ihr die Tests die Ihr macht.   
Es müssen minimal 5 Tests gemacht werden. Dabei können entweder hier manuelle UI-Tests definiert werden oder sinnvolle Unit-Tests erstellt werden. Falls Unit-Tests geschrieben werden sollte hier mindestens der Verweis auf die Unit-Tests stehen.

|  |  |
| --- | --- |
| Abschnitt | Innhalt |
| ID | ST-01 |
| Anforderungen | Der Benutzer öffnet die App und wird aufgefordert, den Zugriff auf das Mikrofon anzunehmen oder abzulehnen. |
| Vorbedingungen | Der Benutzer muss die App installiert haben. |
| Ablauf | 1. Die App wird gestartet 2. Ein Toast erscheint, welcher um die Berechtigung bittet. 3. Der Button «Deny» wird geklickt. |
| Erwartetes Resultat | Das RecordFragment wird geladen. Eine Notification erscheint, dass die Berechtigung abgelehnt wurde. Der Record Button kann nicht gedrückt werden. |

|  |  |
| --- | --- |
| Abschnitt | Innhalt |
| ID | ST-02 |
| Anforderungen | Der Benutzer öffnet die App und wird aufgefordert, den Zugriff auf das Mikrofon anzunehmen oder abzulehnen. |
| Vorbedingungen | Der Benutzer muss die App installiert haben. |
| Ablauf | 1. Die App wird gestartet 2. Ein Toast erscheint, welcher um die Berechtigung bittet. 3. Der Button «Allow» wird geklickt. |
| Erwartetes Resultat | Das RecordFragment wird geladen. Der Record Button kann gedrückt werden. |

|  |  |
| --- | --- |
| Abschnitt | Innhalt |
| ID | ST-03 |
| Anforderungen | Der Benutzer möchte eine Audio aufnehmen. Er drückt auf Record und spricht. |
| Vorbedingungen | ST-02 |
| Ablauf | 1. Die App wird gestartet 2. Ein Toast erscheint, welcher um die Berechtigung bittet. 3. Der Button «Allow» wird geklickt. 4. «Record» wird gedrückt. 5. Benutzer spricht |
| Erwartetes Resultat | Der Recorder startet die Aufnahme. Der Pitch wird angezeigt, während der Benutzer spricht. Save Button erscheint. |

|  |  |
| --- | --- |
| Abschnitt | Innhalt |
| ID | ST-04 |
| Anforderungen | Der Benutzer hat eine Aufnahme erstellt und möchte diese abspeichern. Er klickt auf Save. |
| Vorbedingungen | ST-02; ST-03 |
| Ablauf | 1. Die App wird gestartet 2. Ein Toast erscheint, welcher um die Berechtigung bittet. 3. Der Button «Allow» wird geklickt. 4. «Record» wird gedrückt. 5. Benutzer spricht. 6. «Stop» wird gedrückt. 7. «Save» wird gedrückt. |
| Erwartetes Resultat | Die vom Recorder erstellte mp3 Datei wird abgespeichert. Daten werden dem Recycler hinzugefügt. Der Save Button wird nicht mehr angezeigt. |

## Testauswertung

Zusammenfassung aller durchgeführten Tests. Nur fehlgeschlagene Tests und Tests mit Bemerkungen müssen in der folgenden Tabelle aufgelistet werden.

|  |  |  |
| --- | --- | --- |
| ID | Erfolgreich | Bemerkungen |
| ST-01 | Ja | Der Testfall war erfolgreich, der Testperson 1 ist jedoch aufgefallen, dass es in der angezeigten Fehlermeldung noch einen Rechtschreibfehler gibt. |
| … | … | … |
|  |  |  |

# Fazit

Hier kommt eure Reflexion zum Projekt.

* Was lief gut/schlecht?
* Wie seid ihr mit dem Endergebnis zufrieden?
* Was habt ihr gelernt?
* War alles vorhanden oder was fehlte noch?
* Usw.
* Wir hätten weniger Zeit mit dem Design verschwenden sollen. Diese Tabs haben unendlich Zeit gekostet. Ohne dass das Schöne Design sonderlich brauchbar ist. Zugegeben es sieht ziemlich gut aus, aber das wäre nicht unbedingt notwendig gewesen.