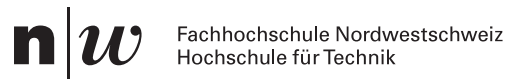


Bachelor Thesis

# Spektrometer App

Anbindung Spektrometer an mobiles Device

Andreas Lüscher, Raphael Bolliger



Dozent: Martin Gwerder  
Auftraggeber: Andreas Hueni

Windisch, 6. März 2017

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Ziel der Arbeit . . . . .	1
1.2	Hilfestellungen . . . . .	1
1.3	Erreichtes . . . . .	1
1.4	Leserführung . . . . .	1
<b>2</b>	<b>Theoretischer Teil</b>	<b>2</b>
2.1	Einleitung . . . . .	2
2.2	Datenhaltung . . . . .	2
2.3	Berechnungen . . . . .	2
2.3.1	Dark Current . . . . .	2
2.3.2	White Reference . . . . .	2
2.3.3	Radiance . . . . .	2
<b>3</b>	<b>Umsetzung</b>	<b>3</b>
3.1	Einleitung . . . . .	3
3.2	Anforderungen . . . . .	3
3.3	Technologien . . . . .	3
3.3.1	Entwicklungsumgebung . . . . .	3
3.3.2	Programmiersprache . . . . .	3
3.3.3	Abhängigkeiten . . . . .	3
3.4	Architektur . . . . .	3
3.4.1	Grobarchitektur . . . . .	3
3.4.2	Core . . . . .	4
3.4.3	iOS App . . . . .	4
3.5	Verbindung zum Spektrometer . . . . .	4
3.5.1	TCP Manager . . . . .	4
3.5.2	Command Manager . . . . .	4
3.6	Datenverarbeitung . . . . .	4
3.6.1	Spectrum Parser . . . . .	4
3.6.2	Indico File Reader . . . . .	4
3.6.3	Indico File Writer . . . . .	4
3.6.4	File Write Manager . . . . .	4
3.8	UI . . . . .	4
3.9	Validierung . . . . .	4
3.10	Einstellungen . . . . .	5
3.11	Gespeicherte Daten . . . . .	5
3.12	Indico Format . . . . .	5
3.13	Raw . . . . .	5
3.14	Reflectance . . . . .	5
3.15	Radiance . . . . .	5

# Abbildungsverzeichnis

### **Zusammenfassung**

Das Hauptziel des Projektes, ist es eine mobile Applikation zu erstellen, welche die RS3 Desktoplösung von ASD ablöst. RS3 ist eine Software, welche die Verbindung zu einem Spektralmessgerät herstellt, Messungen auslösen sowie die Resultate anzeigen kann. Das bisherige System besteht aus einem Laptop inklusive RS3 Software, welche jeweils auf ein Spektrometer abgestimmt ist. Neu soll eine mobile App ausreichen, um mehrere Spektrometer ansprechen zu können. Die App soll Forschende unterstützen, Messungen direkt vor Ort zu beurteilen und verwalten zu können.

# 1 Einleitung

## 1.1 Ziel der Arbeit

Das ist die Ausgangslage Das geologische Institut der Universität Zürich

## 1.2 Hilfestellungen

Das ist die Problemstellung

## 1.3 Erreichtes

Hier wird kurz beschrieben was bei der Arbeit erreicht wurde.

## 1.4 Leserführung

## 2 Theoretischer Teil

### 2.1 Einleitung

Die Projektarbeit enthält nur einen kurzen theoretischen Teil da sie nicht als Forschungsarbeit deklariert ist. Im wesentlichen mussten für die Umsetzung nur drei komplexere Berechnungen umgesetzt werden. Die jedoch bereits bekannt und in der Programmiersprache Java umgesetzt sind. Zur Vollständigkeit werden diese hier erwähnt und beschrieben.

### 2.2 Datenhaltung

### 2.3 Berechnungen

#### 2.3.1 Dark Current

#### 2.3.2 White Reference

#### 2.3.3 Radiance

## 3 Umsetzung

Dieses Kapitel beschreibt die Umsetzung des Produktes. Die nachfolgenden Abschnitte werden schrittweise detaillierter und beschreiben den Aufbau sowie die Designentscheidungen des Softwarecodes.

### 3.1 Einleitung

### 3.2 Anforderungen

### 3.3 Technologien

#### 3.3.1 Entwicklungsumgebung

#### 3.3.2 Programmiersprache

Das Projekt wurde in der Programmiersprache Swift 3 umgesetzt und ist mit iOS 10 und höher kompatibel. Es wurde darauf geachtet, dass alle Abhängigkeiten ebenfalls in Swift umgesetzt sind. Dies erleichtert die Weiterentwicklung da kein, für ungeübte Entwickler, meist schwer lesbarer Objective C Code zum Einsatz kommt.

#### 3.3.3 Abhängigkeiten

### 3.4 Architektur

#### 3.4.1 Grobarchitektur

Um weite Teile des Sourcecodes erneut nutzen zu können, wurde das Projekt in zwei Teile aufgeteilt. Der Core beinhaltet den gesamten Code, welcher systemunabhängig ist.

Der restliche Code befindet sich im Ordner Spektrometer. Dieser ist iOS spezifisch und kann nicht einfach auf andere Plattformen portiert werden.

---

### **3.4.2 Core**

### **3.4.3 iOS App**

## **3.5 Verbindung zum Spektrometer**

### **3.5.1 TCP Manager**

### **3.5.2 Command Manager**

## **3.6 Datenverarbeitung**

### **3.6.1 Spectrum Parser**

### **3.6.2 Indico File Reader**

### **3.6.3 Indico File Writer**

### **3.6.4 File Write Manager**

## **3.7**

## **3.8 UI**

Alle ViewController sind in der Datei Main.storyboard enthalten. Dies war eine bewusste Entscheidung, um Entwicklern einen guten Überblick über den gesamten UI Ablauf zu ermöglichen. Einzig das Design für eine Zelle der Messübersichtstabelle wurde in eine eigne XIB-Datei ausgelagert.

Die Anordnung der Controls wurde im Storyboard gelöst. Kleinere Merkmale wurden jeweils im Code angepasst, indem von bestehenden Controls abgeleitet wurde.

## **3.9 Validierung**

Für die Validierung wurde von jedem benutzen Control eine Ableitung erstellt und das BaseValidationControl Protokoll implementiert. Dieses Protokoll enthält ein Property isValid welches den Gültikeitszustand des Objektes enthält.

In jedem ViewController, muss nun nur noch der ValidationManager aufgerufen werden und die Hauptview übergeben werden. Dieser ValidationManager prüft nun alle Subviews, welches das Protokoll implementieren.

Um die Validation für einen neuen ViewController hinzuzufügen, kann folgendermassen vorgegangen werden: 1. Erstellen Sie einen neuen ViewController 2. Fügen Sie ein Control hinzu und leiten sie von einem bestehenden ValidationControl ab. Oder erstellen Sie eine neue Ableitung eines Controls und implementieren Sie das BaseValidationControl Protokoll. 3. Rufen Sie die validateSubviews Methode des ValidationMangers auf.



---

## **3.10 Einstellungen**

Einstellungen, welche pro Applikation verfügbar sein müssen, werden in den sogenannten UserDefaults gespeichert. Dieser Speicher, kann sämtliche serialisierbaren Objekte speichern.

## **3.11 Gespeicherte Daten**

Alle Dateien, welche auf dem Gerät gespeichert sind,

## **3.12 Indico Format**

Um Messungen speichern zu können, bietet die Firma ASD ein Format namens Indico File Format an. Diese Projektarbeit stützt sich ebenfalls auf dieses Format.

## **3.13 Raw**

Bei sogenannten Raw Messungen, werden die Dateien welche vom Spektrometer gesendet werden, ohne weitere Manipulationen gespeichert.

## **3.14 Reflectance**

## **3.15 Radiance**