

Name	Sven Kreußler
Titel der App	Wetter app
GitHub Repository URL	<a href="https://github.com/SI-Classroom-Batch-007/ios-abschluss-SvenKreussler">https://github.com/SI-Classroom-Batch-007/ios-abschluss-SvenKreussler</a>

## Zielsetzung

Ziel der Abschlussaufgabe ist es, das Gelernte innerhalb des Swift-Moduls in einer eigenen App umzusetzen und diese App abschließend in einer Präsentation vorstellen und erklären zu können. Diese App soll anschließend als Teil eures Portfolios auf GitHub veröffentlicht werden.

## Anforderungen an die App

### Mindestens 5 Screens

Deine App muss mindestens 5 verschiedene Screens anzeigen. Jeder Screen soll dabei sinnvoll genutzt werden.

### Navigation

Zur grundlegenden Navigation kann ein `TabView` genutzt werden (muss aber nicht). Wichtig ist, dass mindestens **1 Sheet** und **1 `NavigationLink`** genutzt werden.

### Anzeigen von Daten innerhalb einer Liste

In einem deiner Views muss eine Liste angezeigt werden, welche sinnvoll integriert ist und nützliche Inhalte liefert. Optional können optional auch Swipe Actions integriert werden.

### Abspeichern von Daten (Core Data / Firebase)

Daten, die von deinen Usern generiert werden, sollen entweder zur Offline-Nutzung mit **Core Data** zur Verfügung stehen oder du integrierst **Firebase**, um die Daten zu synchronisieren. Wenn du dich für Firebase entscheidest, nutze den Firestore zum Speichern und Authentication, um User zu authentifizieren.

## API Call

Such dir eine API, welche thematisch relevanten Inhalt für deine App bereitstellt und nutze die API, indem du einen API Call integrierst. Die API kann mit oder ohne API-Key sein.

## MVVM

Zur Integration der bisher gelernten Softwarearchitektur muss auch deine App MVVM nutzen. Optional kannst du hier direkt mit Repositories arbeiten.

## Pitch

Schreibe in diesem Feld 3-5 Sätze, die beschreiben, welche Funktionalitäten deine App bieten wird und welchen Nutzen sie erfüllen soll. Stell dir vor, du beschreibst die App einem/einer Freund:in.

Jeder möchte über das Wetter informiert werden.  
Daher die App.  
Motivation: sehen ob das gelernte angewendet werden kann

## Aufschlüsselung der Anforderungen

Schreibe hier in jedes Feld (in Stichpunkten), wie deine App die einzelnen Anforderungen erfüllen soll und welche Technologien du dafür verwenden möchtest (z.B. NavigationStack für Navigation).

### Mindestens 5 Screens:

1) Standort, 2) 5 tages Vohersage, 3) sheet: Auswahl für Eigenschaften(UV, Wind, Regen ...), 4) Grafik der Eigenschaften, 5) einstellen der Standorte

### Navigation:

Home Nach links Wischen, führt zum anderen Standort, klicken auf tagesaktuellen Eintrag oder Vorhersage -> sheet, sheet-button -> Eigenschaften(UV, Wind, regen ...)  
Klicken auf Listen Symbol führt zur Standort Auswahl

### Anzeigen von Daten innerhalb einer Liste:

```
list(wetterModelView.fiveDaySum, id: \.wetterModelView.day) {}
```

### Abspeichern von Daten (Core Data / Firebase):

Im persistent Store Speicherfunktion schreiben und dann im ViewModel die Speicherfunktion aufrufen

### API Call:

```
JSONdecoder().decode(WetterResult.self)  
http://api.weatherapi.com/v1/forecast.json?key=8f5d32dc8b0a4092b5f81115230607&q=berlin
```

### MVVM:

Model gestaltet datenmodel.  
ViewModel beobachtet Modell. View Reflektiert ViewModel.

## Meilensteine

Die Projektphase ist in vier große Meilensteine unterteilt (alle zwei Wochen). Überlege dir, was du für deine App in dieser Zeit erreichen möchtest, z.B. UI oder bestimmte Features.

**Es ist Pflicht, dass jeden Freitag der aktuelle Stand der App auf Github gepusht wird!**

<b>Woche 1</b> <b>Freitag 17.11.23</b>	<ul style="list-style-type: none"><li>• Konzept erstellt</li><li>• Figma-Prototyp fertiggestellt</li><li>• Pflichtenheft ausgefüllt und abgegeben</li></ul>
<b>Woche 3</b> <b>Freitag 01.12.23</b>	Benutzer Oberfläche gestalten
<b>Woche 5</b> <b>Freitag 15.12.23</b>	Datenmodell implementieren und API Call
<b>Woche 7</b> <b>Freitag 05.01.23</b>	App finalisieren und Bugs ausmerzen

## Bonus Features

Die oben beschriebenen Anforderungen stellen lediglich Mindestanforderungen dar. Wenn ihr zusätzliche Features integrieren möchtet, ist das natürlich super, sowohl für die Bewertung als auch für euer Portfolio!

**Es sollten unbedingt zuerst die Mindestanforderungen erfüllt sein, bevor an Bonus-Features gearbeitet wird.**

Feature	Erklärung
Statistische Modelle zur Vorhersage nutzen	Übliche, statistische Modelle benutzen, um Vorhersagen zu treffen aufgrund von Daten
Wolken klassifizieren über visuelle Muster	Klassifizierung Modell für Wolken und deren Bedeutung ableiten
Anbindung an übliche Messeinstrumente	Gängige Mess Instrumente implementieren und nutzen

**Dieses Dokument muss unterschrieben werden, um einen "Vertrag" über die zu erbringende Leistung mit uns zu simulieren.**

***Mir ist bewusst, dass meine App anhand der Angaben in diesem Dokument bewertet wird.***

Berlin, 15.11

**Ort, Datum**

Sven Kreußler

**Unterschrift**