



# Inhaltsverzeichnis

1	Logbuch.....	3
1.1	Eintrag: Erstellen der Projektvorgabe.....	3
1.2	Eintrag: Meilenstein 2 .....	4
1.3	Eintrag: Meilenstein 3 .....	6
1.4	Eintrag: Meilenstein 4 .....	8

# 1 Logbuch

## 1.1 Eintrag: Erstellen der Projektvorgabe

**Name:** Kay Hertenstein

**Projekt/Tätigkeit:** Erstellen der technischen Anforderungen

**Zeitraum:** 08.02.25 – 01.03.25

---

### **Tätigkeiten / Aufgaben:**

Gemäss der Modulvorgabe, erstelle ich das Dokument für die Semesterarbeit/Projektarbeit. Durch das genaue erstellen der Dokumentation werde ich mir den konkreten Aufgaben bewusst, welche ich umsetzen werde. Ich merke wie eine sauber erstellte Dokumentation ein wesentlicher Faktor ist für eine effiziente Softwareentwicklung.

### **Erreichte Ziele / Ergebnisse:**

Ich konnte die technischen Anforderungen definieren. Konkrete Eckdaten der Applikation werden festgelegt. Erfolgreich ein eigenes Mockup der Applikation erstellt.

### **Schwierigkeiten / Probleme:**

Definieren der Anforderungen und Zeitmanagement bezüglich Themen welche später im Modul unterrichtet werden. Einbringen des agilen Entwicklungsansatzes in Spezifikationen.

### **Lösungen / Maßnahmen:**

In der Dokumentation dokumentieren, dass sich die Spezifikationen/Präferenzen gemäss Unterricht gewählt werden mit Verweis auf prototypische Implementierungen.

### **Zusammenfassung:**

In der Softwareentwicklung spricht man oft von agilen Prozessen, diese in funktionalen Anforderungen miteinzubeziehen hat mich verunsichert. Durch das Erstellen der Spezifikation hat meine Idee eine konkrete Form erhalten. Viele weitere Möglichkeiten (Features) werden mir bewusst.

## 1.2 Eintrag: Meilenstein 2

**Name:** Kay Hertenstein

**Projekt/Tätigkeit:** Implementieren Kernansichten, Grundstruktur des Backends, Datenbankmodellierung

**Zeitraum:** 08.03.25 – 29.03.25

---

### **Tätigkeiten / Aufgaben:**

Gemäss dem Modulvorgabe starte ich mit der konkreten Umsetzung des Projekts gemäss den definierten Anforderungen. Ich habe im Frontend die Kernansichten erstellt. Dazu gehört die Kernansicht – für die Webseite also auch die Kernansicht des Dashboards. Die Kernansichten enthalten gemeinsame Komponenten und die Layoutstruktur für die spezifischen Ansichten. Im Backend habe ich gemäss meines Zeitmanagements die Grundstruktur eingerichtet sowie eine erste Datenbankmodellierung implementiert.

### **Erreichte Ziele / Ergebnisse:**

Ich habe die Ziele gemäss Zeitmanagement umsetzen können. Die ersten Resultate der prototypischen Inbetriebnahme waren sehr positiv – die Kernansichten funktionieren und die Datenbankmigrationen wurden erfolgreich durchgeführt. Durch die positiven Resultate konnte ich weitere Implementationen vornehmen als geplant.

### **Schwierigkeiten / Probleme:**

Meine Herausforderungen liegen in der Umsetzung des Zeitmanagements, es hat sich während der Umsetzung gezeigt, dass es für mich sinnvoller ist die gesamte Struktur eines Endpoints zu implementieren anstelle einer gestaffelten Umsetzung. Eine weitere Challenge war es ein konkretes Usermanagement zu definieren. Im Frontend beschäftigt mich die Performance.

### **Lösungen / Maßnahmen:**

Ich habe während der Entwicklung folgenden Massnahmen / Lösungen getroffen:

1. **Zeitmanagement** – einen flexiblen Ansatz wählen, ggf. Zeitplan überarbeiten
2. **Usermanagement** (Für Benutzer die sich registrieren und die Applikation verwenden): Nach mehreren Ansätzen – habe ich entschieden das implementierte Usermanagement zu verwenden welches Django anbietet und die Benutzereigenschaften selbständig zu erweitern.
3. **Performanceprobleme:** Ich habe während des Entwicklungsprozesses Mühe mit der Performance – meine SPA hat eine initiale Ladezeit von > 15 Sekunden. Als Massnahme werde ich eine Prototypische Produktionsversion in einer Testumgebung prüfen (Apache Server später nach Möglichkeit auf NGINX mit Unicorn).
4. **Performance** bezüglich http-Requests: Unsicherheit bezüglich des initialen Payloads – sollen Daten initial geladen werden oder erst beim entsprechendem Seitenaufruf. Als Massnahme nehme ich Rücksprache mit einem Dozenten.

### Vorbereitung für die Projektbesprechung:

In der Projektbesprechung würde ich gerne diskutieren, ob meine Implementierung des User-Managements den gängigen Best Practices entspricht. Zudem möchte ich meine Bedenken zur Performance klären – insbesondere, ob es sinnvoller ist, Daten initial zu laden oder erst bei Bedarf abzurufen. Je nach Erkenntnissen könnten auch weiterführende Themen wie Caching-Mechanismen relevant werden.

**Zusammenfassung:**

Meilenstein 2 war für mich eine spannende Phase der technischen Implementierung. Anfangs hatte ich einige Schwierigkeiten mit der UI-Library MUI, doch inzwischen bin ich mit der Integration zufrieden. Besonders die erfolgreiche Umsetzung der prototypischen API-Endpunkte war wertvoll, da ich dadurch bereits Themen wie die JWT-Authentifizierung vorwegnehmen konnte. Ein persönlicher Meilenstein war es, als die Tokens erfolgreich authentifiziert wurden. Ich freue mich auf die kommende Projektbesprechung und bin zuversichtlich, dass ich meine offenen Fragen in absehbarer Zeit klären kann.

**Nachtrag Besprechung Meilenstein 2:**

Gemäss Feedback des Dozenten, sollte die Projektarbeit mit Businesslogik im Backend Teil ergänzt werden. Mögliche Ansätze für die Umsetzung dieser Vorgabe ist das Erstellen von Analysen und diese im Frontend darstellen. Die genaue Umsetzung ist bis Meilenstein 3 in der Projektdokumentation festgehalten.

### 1.3 Eintrag: Meilenstein 3

**Name:** Kay Hertenstein

**Projekt/Tätigkeit:** Implementieren UI-Logik, Implementierung API-Endpunkte & Geschäftslogik, Authentifizierung

**Zeitraum:** 05.04.25 – 03.05.25

---

#### **Tätigkeiten / Aufgaben:**

Gemäss definiertem Zeitplan habe ich im Frontend die Aufgabe *Implementierung der UI-Logik mit Testdaten* umgesetzt, während im Backend die Aufgabe *Implementierung der API-Endpunkte, Businesslogik und Implementierung von JWT* war.

#### **Erreichte Ziele / Ergebnisse:**

Frontend / Implementierung UI-Logik mit Testdaten

Ich habe das Dashboard mit der entsprechenden UI-Logik und CRUD Operationen erstellt.

Commits: 4fcd4f54, d01b2d38m, b068f18f

Die Home und Search Ansichten mit entsprechender UI-Logik habe ich umgesetzt.

Commits: 6e2fcc9a, ce4d14de, e1853ddb

Das Login und Registrationsverfahren sowie die Implementierung von JWT, habe ich erfolgreich optimiert, sowie die Authentifizierungsabläufe wurden integriert.

Commits: be5d48ba, 75c347d2

Allgemeine Verbesserungen und Optimierungen welche nicht konkret spezifiziert sind wurden durchgeführt.

#### Backend / Implementierung der API-Endpunkte, Businesslogik und Implementation von JWT

Die Businesslogik und Endpunkte für Kernfunktionalitäten wurden implementiert.

Commits: 09f12ab6, 9ace17d7, 1713a7fc, b11ba3c1

Die Implementation von JWT (Authentifizierung generell) wurde umgesetzt:

Commit: 47f5ff2c

#### Testing / TDD

Im Zeitplan habe ich fürs Testing die Vorgabe ca. ab Meilenstein 3 definiert, als auch den Ansatz TDD definiert, ich habe diese Strategie teilweise umgesetzt, aber durch häufige Anpassungen nur für einzelne Komponenten durchgeführt welche einen fast-finalen Stand erreicht haben. Die Commits ea2dc4e4 und 09f12ab6 decken diese Tests ab, möchte aber darauf hinweisen das, dass diese noch im Entwicklungsstand sind.

#### **Übersicht Änderungen / Projektdokumentation:**

- Neu: FAFE ID-009: Passwort zurücksetzen via Email/Authorisierungslink.
- Neu: FAFE ID-010: Ein Benutzer soll die Möglichkeit haben sein Passwort zurückzusetzen.

- Neu: FAFE ID-010: Benachrichtigungen.
- Neu: FAFE ID-011: Kontakt/Supportformular.
- Neu: FAFE ID-011: Das System erstellt Analysen für Marketingzwecke.
- Neu: FAFE ID-012: Bereitstellen einer Übersicht der definierten Analysen.
- Neu: FAFE ID-013: Eine URL kann geteilt oder gespeichert werden.
- Ergänzung FAFE ID-008: Kann Anforderung definiert (Suche nach Dienstleister).
- Ergänzung FAFE ID-007: Kann Anforderung definiert (Suche nach Dienstleister).

### **Schwierigkeiten / Probleme:**

Eine Schwierigkeit die ich habe betrifft die Search Komponenten im Frontend, die Suchfunktion besitzt 4 Filter und eine Volltextsuche, diese aufeinander abzustimmen war machbar aber die Logik der einzelnen Komponenten wurde sehr umfangreich.

Ich habe ein unstabiles Verhalten in der Verwendung des http-only Cookies festgestellt. Die Implementierung des Cookies für die Speicherung des Refresh – Tokens konnte ich eigentlich erfolgreich umsetzen unter Verwendung eines lokalen ssl Zertifikates. Während der Entwicklung hatte ich allerdings ein unstabiles Verhalten (Möglicherweise durch Browser Updates)

Eine weitere Herausforderung sind die zusätzlichen Anforderungen fürs Backend, auch habe ich bei der Planung diverse Funktionalitäten wie Passwort zurücksetzen nicht beachtet.

### **Lösungen / Maßnahmen:**

Ich habe während der Entwicklung folgenden Massnahmen / Lösungen getroffen:

Ich versuche die Logik der Suchkomponenten zu Modularisieren und ggfs. Mit Services oder Utils zu arbeiten.

Damit eine stabile Authentifizierung gewährleistet ist werde ich lokal auf die Verwendung des http-only Cookies verzichten und als alternative mit dem Localstorage arbeiten.

Um die zusätzlichen Anforderungen umzusetzen verweise ich auf die definierte Reservezeit im Zeitplan.

### **Vorbereitung für die Projektbesprechung:**

Gerne würde ich best-practice ansprechen bezüglich der Komponenten meiner Suchfunktion (Commit e1853ddb)

Bestehen absolute No-Gos oder Umsetzungen welche ich überarbeiten sollte?

### **Zusammenfassung:**

Die Umsetzung des Meilenstein 3, ist meiner Meinung nach erfolgreich gelungen und ich habe die definierten Ziele gemäss Zeitmanagements umsetzen können. Ich habe im aktuellen

Stand meines Repos noch einige auskommentierte Stellen, gerne würde ich erwähnen, dass ich diese bewusst so belassen habe für die weiteren Entwicklungsschritte.

## 1.4 Eintrag: Meilenstein 4

**Name:** Kay Hertenstein

**Projekt/Tätigkeit:** Testen der API-Endpunkte, UI-Tests, Refactoring generell, Feinabstimmung, Frontend finalisieren

**Zeitraum:** 04.04.25 – 31.04.25

---

### **Tätigkeiten / Aufgaben:**

Gemäss definiertem Zeitplan hatte ich bis zum Meilenstein 4 die Aufgaben: prüfen der API Endpunkte, UI-Tests, Refactoring und Finalisierung (Feinabstimmung)

### **Erreichte Ziele / Ergebnisse:**

#### Backend

Das Hauptthema im Meilenstein 4 war das erfüllen der Anforderung für Qualität- und Performancevorgaben. Ich habe ein gesamtes Refactoring im Backend durchgeführt:

- Code wurde kommentiert (mehrheitlich mit docstrings)
- Alle Tests implementiert
- Code Optimierungen gemäss Test Ergebnissen

Der Commit eb676d85 (Backend tests) zeigt diese Änderungen.

Die Ergebnisse der Qualitätssicherung war eine Testabdeckung von 98%. Mittels dem Monitoring-Tool Silk habe ich feststellen können das fast alle Requests im akzeptablen Bereich sind. Es bestehen kleinere Ausnahmen wo ich bis zur Abgabe noch die Datenbankabfragen optimieren werde.

Gemäss Besprechung von Meilenstein2 war ein wichtiger Punkt, dass das Projekt noch mehr Businesslogik im Backend enthalten sollte, siehe dazu Meilenstein 3 Übersicht Änderungen. Dies habe ich im Meilenstein 4 umsetzen können mit einer zusätzlichen App (Analytics). Auch kann das System nun Emails versenden im Bezug auf Passwort zurücksetzen und Kontaktformular. Da mittlerweile einige 'clean-ups' durchgeführt wurden Verweise ich hier auf den Stand des Repositorys.

#### Frontend

Im Frontend war gemäss Zeitplan geplant, dass sicherstellen der Qualität (tests) und die Feinabstimmung mit dem Backend zu implementieren. Ich habe zuerst die neu definierten Anforderungen umgesetzt, daher muss ich im Frontend bezüglich testing und Qualitätssicherung auf die definierte Reservezeit ausweichen. Trotzdem konnte ich folgende Punkte umsetzen:

- Applikation Full-Responsive finalisieren
- Darstellen der Statistiken im User-Dashboard
- Kontaktformular
- Diverse Views finalisiert (Detailansicht eines ausgewählten Services mit öffentlichem Profil des Anbieters, Agbs, Impressum, Datenschutz)
- Feinabstimmung mit Backend: abgleichen der Status und Fehlermeldungen.



Im Testing bin ich beim Stand der Abgabe Meilenstein 4 – auf ca 65 Unittests. Ich erwarte ca. 150 – 200 Tests bis zur finalen Abgabe.

#### Deployment

Um das Projekt möglichst einfach in Betrieb zu nehmen und mögliche Fehlerquellen auszuschliessen habe ich die optionale Anforderung ‘Deployment mit Docker’ umgesetzt.

#### **Schwierigkeiten / Probleme:**

Während der Umsetzung der Anforderungen bezüglich des Meilenstein 4, hatte ich häufig Schwierigkeiten im Prüfen der Applikation als ganzes System – im Zusammenspiel aller Komponenten. Ich musste mehrere Refactorings durchführen und API Endpunkte anpassen.

#### **Lösungen / Maßnahmen:**

Ausgiebige Tests und Kontrollen durchführen.

#### **Zusammenfassung:**

Ich bin zufrieden mit meiner Umsetzung des Meilenstein 4. Ich konnte die wichtigsten Punkte der Projekt Anforderungen umsetzen. Das sicherstellen der Qualität im Backend hat sich als sehr hilfreich herausgestellt, da sich daraus die Spezifikationen des erstellten Codes entnehmen lassen. Wichtige Erkenntnisse die ich in meine berufliche Tätigkeit übernehmen kann. Im Frontend bin ich eigentlich auch zufrieden da ich eine gute Usability erzielt habe mit den zusätzlichen Anforderungen, und bin gespannt was für Ergebnisse ich im Qualitätscheck bis zur finalen Abgabe erreiche.