

Projekt Softwaretechnik  
**Bewerbungstracker**

Jannika Börner, Marcus Klein, Norman Münzer, Lukas Reinhardt,  
Lara Seline Hippenstiel

Alexander Brendel, Klemens Morbe  
Prof. Dr. Jörg Nitzsche

Studiengang: Softwaretechnik und Medieninformatik  
Fakultät: Informationstechnik  
Hochschule Esslingen  
Wintersemester 2025/26

---

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>II</b>
<b>1 Meilenstein 01</b>	<b>1</b>
1.1 Problemstellung . . . . .	1
1.2 Lösungsansatz . . . . .	1
1.3 Zielgruppe . . . . .	1
1.4 Geforderter Funktionsumfang . . . . .	1
1.5 UI-Entwürfe . . . . .	2
1.5.1 Erster UI-Entwurf . . . . .	2
1.5.2 Verfeinerung des UI-Konzepts . . . . .	2
1.6 Architektur der Software . . . . .	3
1.6.1 Übersicht zentraler Komponenten . . . . .	3
1.6.2 Beschreibung des allgemeinen Ablaufs . . . . .	4
1.7 Projektmanagement . . . . .	4
1.7.1 Meetings . . . . .	4
1.7.2 Kommunikation . . . . .	5
1.8 Aufwandsschätzung . . . . .	5

# Abbildungsverzeichnis

1.1	Bevorzugtes 1. Mock-Up: Front Page . . . . .	6
1.2	Bevorzugtes 1. Mock-Up: Bewerbungsseite . . . . .	6
1.3	Ausgearbeitetes 2. Mock-Up Kalender . . . . .	7
1.4	Ausgearbeitetes 2. Mock-Up Bewerbungen . . . . .	7
1.5	Ausgearbeitetes 2. Mock-Up Unternehmensansicht . . . . .	8
1.6	Ausgearbeitetes 2. Mock-Up Dokumente . . . . .	8
1.7	Architektur: Zentrale Komponenten . . . . .	9
1.8	Aufwandsschätzung: Gantt-Diagramm Stand Meilenstein 1 . . . . .	10

# 1 Meilenstein 01

## 1.1 Problemstellung

Bei der Jobsuche werden viele Bewerbungen für unterschiedliche Stellenangebote verfasst und verschickt. Im Laufe des Bewerbungsprozesses müssen Daten, wie Termine, Adressen, Ansprechpartner, Gehalt und mehr, verwaltet werden. Diese Daten befinden sich dabei in der Regel verstreut in Notizen, Mails, Zetteln und Spreadsheets. Dadurch verliert man sehr leicht die Übersicht und die Organisation wird schnell zum Chaos. Die Folge ist ein stressiger und unstrukturierter Bewerbungsprozess.

## 1.2 Lösungsansatz

Mit einer zentralen Anwendung können alle relevanten Informationen an einem Ort verwaltet werden. Durch eine intuitive Benutzeroberfläche wird das Sammeln und Eintragen von Informationen effizient gestaltet und der Nutzer erhält einen gut strukturierten Überblick. Mit Hilfe von persönlichen Bewertungen und Notizen vom Anwender, kann dieser bei der Entscheidungsfindung unterstützt werden.

## 1.3 Zielgruppe

Die Zielgruppe umfasst Schüler, Studierende, Berufseinsteiger und Berufserfahrene, die sich beruflich umorientieren möchten. Diese müssen ihre Bewerbungen oft unter Druck und mit begrenzter Zeit verfassen und verwalten. Einige, wie beispielsweise Studierende, müssen noch anderen Verpflichtungen nachkommen. Insbesondere diese benötigen eine Lösung, die ihnen bei der Strukturierung des Bewerbungsprozesses unter die Arme greift.

## 1.4 Geforderter Funktionsumfang

In der App soll es möglich sein folgende Informationen bezüglich eines Unternehmens einzutragen und zu verwalten:

- Name und Logo des Unternehmens
- Kontaktperson
- Leistungen/Benefits
- Gehaltsspielraum
- Distanz zum Wohnort
- Mitarbeiteranzahl
- Persönliches Rating
- Eigene Notizen

- Kurze Unternehmensbeschreibung

Zusätzlich soll es möglich sein, diese Basisinformationen von einer KI zu sammeln und eintragen zu lassen.

Die App soll basierend auf den erfassten Informationen und der Gewichtung unterschiedlicher Schwerpunkte, ausgewählt durch den Nutzer, bei der Entscheidungsfindung helfen. Hierzu soll eine Entscheidungsmatrix benutzt werden können.

Des Weiteren ist es gewünscht, dass Dokumente bezüglich der Bewerbung innerhalb der App hochgeladen und gesammelt werden.

Die Daten sollen in einer Datenbank abgespeichert werden und es soll dem Nutzer möglich sein, diese Daten von jedem beliebigen Gerät aus mithilfe seiner Login-Daten abzurufen. Außerdem sind weitere Optionen für bessere Zugänglichkeit erwünscht:

- ein Dark & Light Theme
- Mehrsprachigkeit

Das gesamte Projekt soll Open-Source sein und inklusive vollständiger Dokumentation veröffentlicht werden.

Aus dem 1. Gesprächstermin (siehe Protokoll vom 07.10.25) ergaben sich zusätzlich folgende Anforderungen:

- Funktion zum Sortieren und Filtern nach z. B. Termin, Gehalt, Bewerbungsdatum, Unternehmensgröße, Entfernung, etc.
- Settings mit Basisinformationen über den Nutzer

## 1.5 UI-Entwürfe

### 1.5.1 Erster UI-Entwurf

Bei Betrachtung der ersten Mock-Ups (siehe Abbildungen 1.1, 1.2) wurden einige Punkte beschlossen:

- Name: Bewerbungstracker statt Jobtracker
- Leiste oben mit Widgets
- Neue Bewerbungen hinzufügen durch Pop-Up Menü oder ähnliches mit Eingabemöglichkeiten

### 1.5.2 Verfeinerung des UI-Konzepts

#### Startseite/Kalender

In der überarbeiteten Version dienen der Kalender und die Terminübersicht gleichzeitig auch als Startseite (siehe Abbildung 1.3). Im oberen Bereich befindet sich, wie bei der 1. UI-Version beschlossen, eine Navigationsleiste, über die man zwischen den Hauptbereichen “Termine”, “Bewerbungen” und “Dokumenten” wechseln kann.

Darunter befindet sich eine Navigationsleiste mit einer kompakten Terminübersicht. Hier können alle Termine eingetragen und tabellarisch mit Informationen zu Unternehmen, Art des Termins, Datum, Uhrzeit und auch möglichen To-Dos angezeigt werden. Termine können über das Plus-Symbol hinzugefügt werden, wobei die Verwaltung einfach über die Bearbeiten und Löschen-Symbole bei jedem Eintrag machbar ist.

Im unteren Bereich ist ein Kalender-Widget, das die Termine des aktuellen Monats visuell darstellt. Termine werden farblich nach Typ unterschieden, wodurch der Nutzer einen schnellen Überblick über alle Fristen und Termine bekommen kann.

## Bewerbungen

Im zweiten Hauptbereich, “Bewerbungen”, (siehe Abbildung 1.4) erhält man eine Übersicht über alle Bewerbungen. Im oberen Bereich befindet sich weiterhin die Navigationsleiste. Darunter folgen drei Bedienfelder mit einer Suchfunktion, einer Sortier- und Filterfunktion und einem Bedienfeld um neue Bewerbungen zu erfassen und hinzuzufügen. Unter den Bedienfeldern wird der Seiteninhalt in Kachelansicht angezeigt. Jede Kachel repräsentiert eine Firma, bei der eine Bewerbung eingereicht wurde. In der Kachel sind das Logo des Unternehmens, sowie der Firmenname und der nächste relevante Termin zu sehen. Durch das Klicken auf eine Bewerbung kommt man in die Übersicht des Unternehmens und des Jobs (siehe Abbildung 1.5). In dieser Ansicht werden Informationen zum Unternehmen und zur beworbenen Position dargestellt. Im oberen Bereich erscheinen das Firmenlogo sowie ein Bewertungssystem in Form von Sternen, das anzeigt, wie gut der Job und das Unternehmen den persönlichen Wünschen entsprechen. Oben links in dem Fenster sieht man einen Pfeil um wieder zurück in die Bewerbungsübersicht zu kommen. Darunter folgen in Feldern zuerst der Firmenname und dann ein Feld mit dem nächsten relevanten Termin. Im nächsten Feld wird eine Kurzbeschreibung des Unternehmens eingefügt. Darunter folgen Kontaktpersonen, darunter der Gehaltsspielraum und die Distanz. Als letztes gibt es die Perks zum Arbeitsplatz wie z.B. Homeoffice, Geschäftswagen, usw.

## Dokumente

Im letzten Hauptbereich der Anwendung, “Dokumente” (siehe Abbildung 1.6), befindet sich die Dokumentenablage. Hier werden alle Dokumente zu den Bewerbungen angezeigt und verwaltet. Über das Bedienfeld im oberen Bereich der Seite kann man neue Dokumente hochladen. Darunter werden alle bereits hochgeladenen Dokumente in Kachelansicht gelistet. In jeder Kachel stehen das Upload-Datum und die Art des Dokuments. Die Dokumente sind über Bearbeitungs- und Löschsymbole in den jeweiligen Kacheln verwaltbar, wodurch eine einfache und intuitive Handhabung ermöglicht wird.

## 1.6 Architektur der Software

Die Architektur der Software setzt sich aus vier zentralen Komponenten zusammen, welche in Docker betrieben werden. Im Diagramm (siehe Abbildung 1.7) erkennt man wie das Frontend, das Backend, das Authentifizierungssystem und die Datenbank miteinander interagieren.

### 1.6.1 Übersicht zentraler Komponenten

#### 1. Frontend (React)

- Läuft im Browser des Nutzers, wo dieser mit Weboberfläche interagiert
- Leitet beim ersten Zugriff zur Authentifizierung (Keycloak) weiter
- Erhält vom IAM (Keycloak) ein Access Token nach erfolgreicher Anwendung  
→ Token wird bis Session-Ende für alle weitere Anfragen an das Backend (Spring Boot) genutzt

#### 2. Authentifizierung (Keycloak)

- Das Identity- und Access-Management-System (IAM) Keycloak übernimmt Authentifizierung

- Überprüft Anmeldedaten und gibt bei erfolgreichem Login ein Access Token an das Frontend (React) zurück
3. Backend (Spring Boot)
    - Ermöglicht den Zugriff auf REST-API-Endpunkte
    - Empfängt Anfragen des Frontends (React), überprüft die Gültigkeit des Access Tokens
      - Gültig: Verarbeitet die Anfrage weiter und ruft Daten von der Datenbank (PostgreSQL) ab
      - Ungültig: Automatische Weiterleitung zur Authentifizierung (Keycloak)
  4. Datenbank (PostgreSQL)
    - Strukturelles Speichern der Daten
    - Kommuniziert über JDBC mit dem Backend (Spring Boot)
      - Ermöglicht das Abrufen, Speichern und Aktualisieren der Nutzerdaten und Anwendungsinformationen

### 1.6.2 Beschreibung des allgemeinen Ablaufs

1. Nutzer öffnet die Webanwendung im Browser und wird bei fehlendem gültigen Access Token an Keycloak-Loginseite weitergeleitet.
2. Access Token wird nach erfolgreicher Anmeldung an das Frontend übergeben, während Nutzer zeitgleich zurück zur Webanwendung geleitet wird.
3. Nutzer interagiert, wodurch eine Anfrage mit Access Token an Spring Boot Backend übersendet wird.
4. Das Backend überprüft die Validität des Tokens und ruft, speichert bzw. ändert ggf. Daten in der PostgreSQL-Datenbank ab.
5. Das Backend übergibt die Daten an das Frontend, welches diese anzeigt.
6. Ablauf wiederholt sich ab Schritt 3. bis die Session des Nutzers beendet wird.

## 1.7 Projektmanagement

Die Softwareentwicklung verläuft in Anlehnung an Scrum. Es findet ein wöchentliches Meeting statt, das den Beginn des nächsten Sprints markiert. Projektaufgaben werden in einer Art Kanban-Board auf GitHub verwaltet. Zur Versionsverwaltung wird ein privates Repository auf GitHub in einer für das Projekt erstellten Organisation verwendet.

### 1.7.1 Meetings

Das regelmäßige Meeting findet jeden Freitag um 14:00 Uhr über Microsoft Teams statt. Darin präsentiert das Team seine Fortschritte aus dem letzten Sprint und legt zusammen mit dem Kunden die Aufgaben für den nächsten Sprint fest. Zusätzlich bietet das Meeting die Möglichkeit, Anforderungen zu spezifizieren und Rückfragen zu stellen. Der Leiter des Meetings wird vom Team gestellt und rotiert wöchentlich. Ein Protokoll wird vom Team gemeinsam in einem Hedge-Doc-Dokument erstellt. Im Anschluss bespricht sich das Team in einem halbstündigen Meeting. Bei weiterem Gesprächsbedarf steht dem Team ein weiteres 30-minütiges Zeitfenster für ein zusätzliches Meeting zur Verfügung.

### 1.7.2 Kommunikation

Für die Kommunikation ist ein Discord-Server eingerichtet. Darin befinden sich eine Reihe von Textkanälen und ein allgemeiner Sprachkanal. Davon sind jeweils ein Textkanal für Dokumente und Protokolle reserviert. Des Weiteren stehen Textkanäle für Fragestellungen zur Verfügung. Für die Teamkommunikation stehen gesonderte Textkanäle bereit, die nur vom Team eingesehen werden können. Einer dient dabei für tägliche Fortschrittsmeldungen der Teammitglieder, um eine Art Daily-Standup zu ermöglichen.

## 1.8 Aufwandsschätzung

Für eine initiale Aufwandsschätzung wurden die Meilensteine als feste Werte in das Gantt-Diagramm, Abbildung 1.8, eingetragen und die Anforderungen der einzelnen Meilensteine als Aufgaben vermerkt und abgeschätzt. Dabei wurden initial die Wochenenden mitgezählt, jedoch von der Software nicht akzeptiert, daher sind manche Aufgaben aufgrund des Wochenendes überdimensioniert. Technische Aufgaben, welche für die Funktionalität des Projekts abgeschlossen werden müssen sind nicht eingetragen. Da es noch kein technisches Konzept und damit keine genaue Aufgaben gibt, sind diese nicht aufgeführt. In die Setup-Schätzungen wurden verschiedene Zeiten der Verfügbarkeit und wahrscheinliche technische Probleme eingerechnet.



## 1 Meilenstein 01



Abbildung 1.1: Bevorzugtes 1. Mock-Up: Front Page

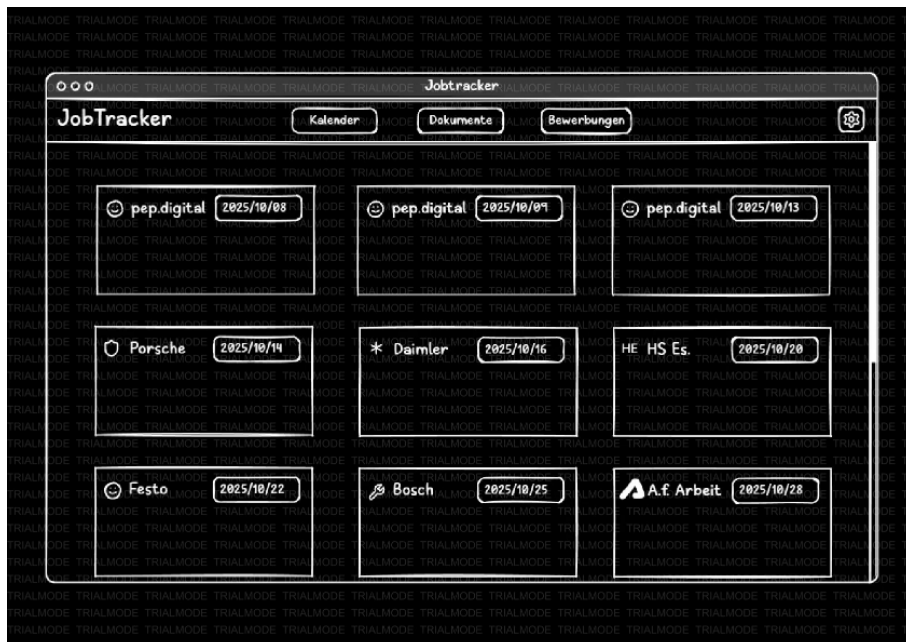


Abbildung 1.2: Bevorzugtes 1. Mock-Up: Bewerbungsseite

## 1 Meilenstein 01

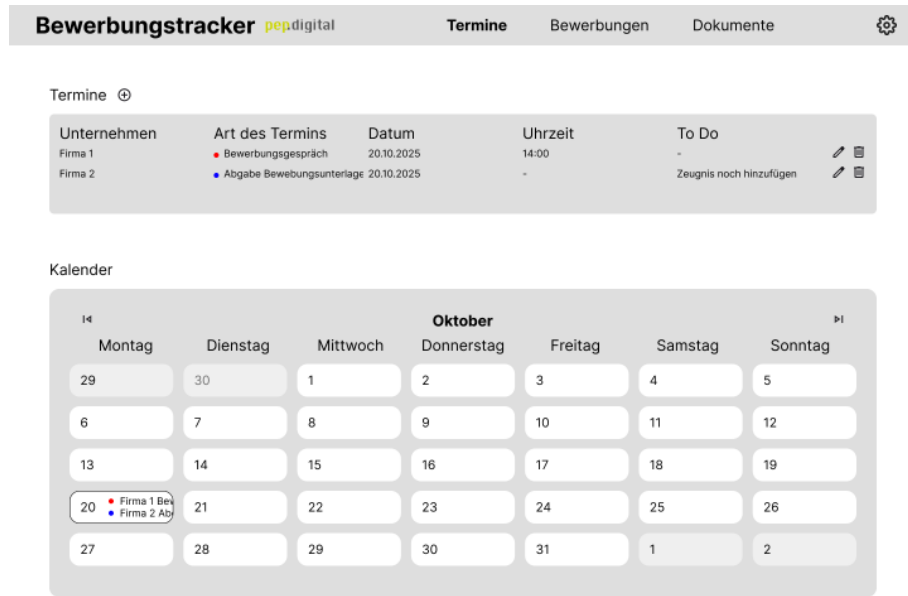


Abbildung 1.3: Ausgearbeitetes 2. Mock-Up Kalender

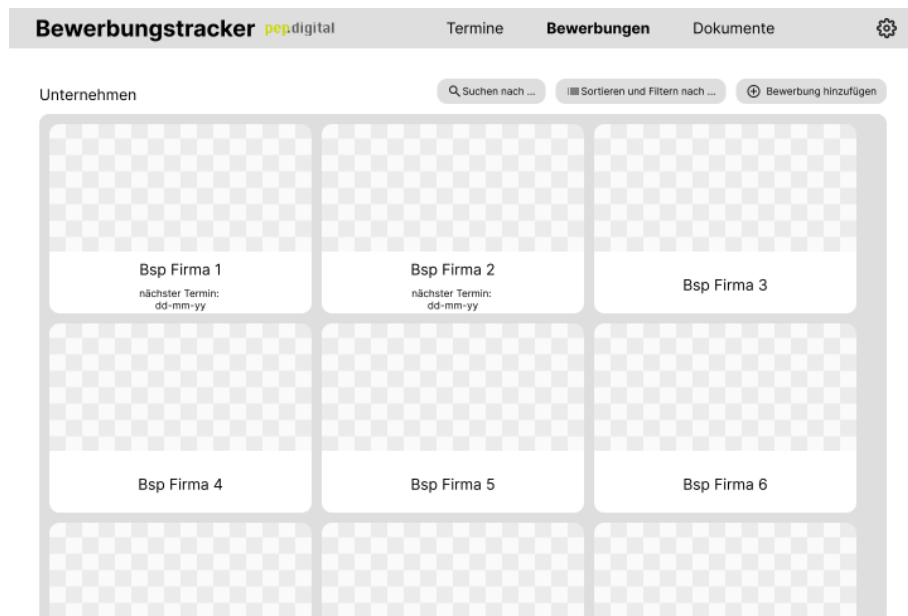


Abbildung 1.4: Ausgearbeitetes 2. Mock-Up Bewerbungen

## 1 Meilenstein 01

The mock-up shows a user interface for 'Bewerbungstracker pepdigital'. The top navigation bar includes 'Termine', 'Bewerbungen', and 'Dokumente'. The main content area is titled 'Unternehmensansicht' and features a checklist of company details. Each item has a checkbox and a text input field for additional information.

- ☐ **Beispiel-Firma**
- ☐ **nächster Termin**
- ☐ **„Das Unternehmen ist ein moderner Dienstleister, der sich durch Innovationskraft, Flexibilität und kundenorientiertes Handeln auszeichnet. Mit einem breiten Leistungsspektrum und hoher fachlicher Kompetenz positioniert es sich als verlässlicher Partner in seiner Branche.“**
- ☐ **Kontaktperson**
- ☐ **Gehaltsspielraum**
- ☐ **Distanz**
- ☐ **Leistungen**
  - Flexible Arbeitszeiten
  - Homeoffice- oder Remote-Work-Möglichkeiten
  - Mitarbeiter-Events und Teambuilding
  - Kostenlose Getränke (Kaffee, Wasser)

Abbildung 1.5: Ausgearbeitetes 2. Mock-Up Unternehmensansicht

The mock-up shows the 'Dokumente' section of the application. It features a list of documents, each with a checkbox, a title, and a date input field. A 'Dokument hinzufügen' button is located at the top right of the list.

- ☐ **Lebenslauf**
- ☐ **Bewerbungsanschreiben Firma 1**
- ☐ **Bewerbungsanschreiben Firma 2**

Abbildung 1.6: Ausgearbeitetes 2. Mock-Up Dokumente

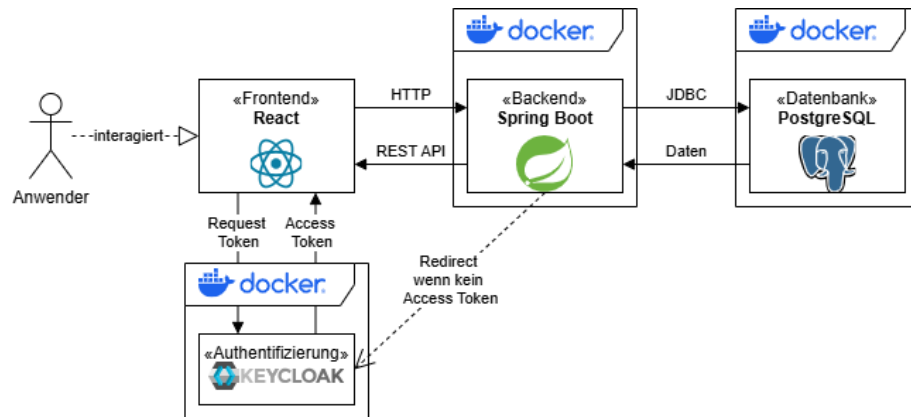


Abbildung 1.7: Architektur: Zentrale Komponenten

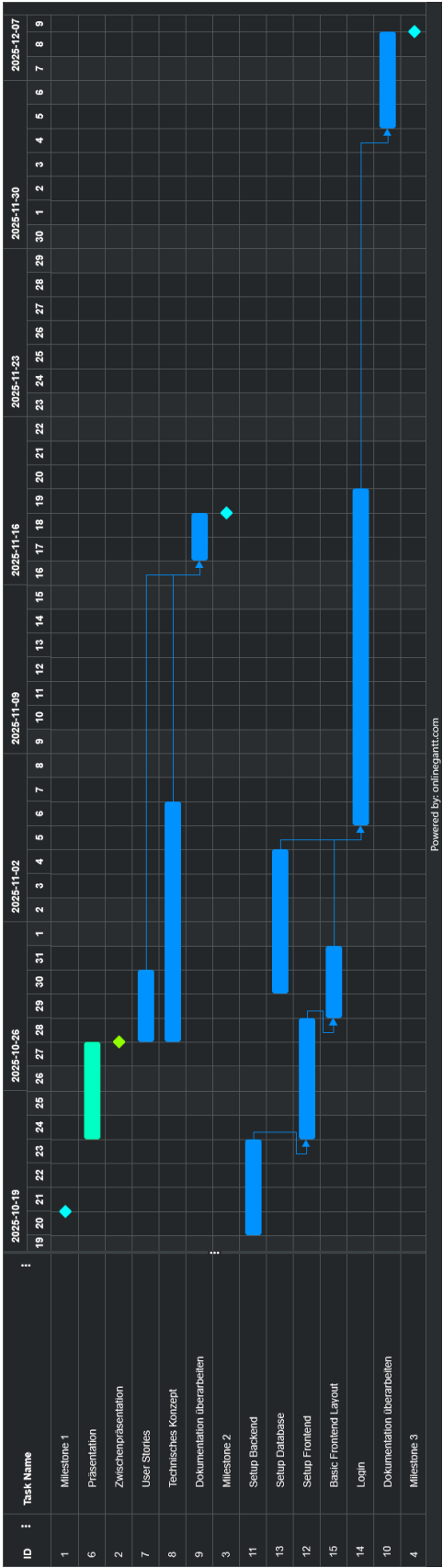


Abbildung 1.8: Aufwandsschätzung: Gantt-Diagramm Stand Meilenstein 1