**Projektbeschreibung: Finanzmanager – Dein persönliches Finanz- & Asset-Dashboard**

**Ziel des Projekts**

Ziel dieser Anwendung ist es, eine kombinierte Web-App zu entwickeln, mit der Nutzer ihre persönlichen Finanzen verwalten und gleichzeitig ihre Investitionen im Blick behalten können. Die App vereint die Funktionalität eines Haushaltsbuchs mit einem modernen Asset-Watchlist und Investment-Tracking-System. Nutzer erhalten so einen umfassenden Überblick über ihre finanzielle Situation – inklusive klassischer Ausgaben sowie digitaler Vermögenswerte.

**Funktionen der Anwendung**

Die Web-App besteht aus zwei Hauptmodulen: einem Finanzmodul und einem Krypto-Modul.   
Im Finanzmodul können Benutzer Einnahmen und Ausgaben erfassen, bestimmten Kategorien wie "Miete", "Freizeit" oder "Essen" zuordnen. Zusätzlich ermöglicht die App die Anzeige aktueller Wechselkurse, um Ausgaben in verschiedenen Währungen zu vergleichen.

Im Krypto-Modul kann der Nutzer Kryptowährungen zur persönlichen Watchlist hinzufügen. Die App ruft dafür aktuelle Preisdaten von der CoinGecko-API (Krypto) oder von der twelvedata-API (Aktien) ab. Zu jedem Eintrag können eigene Notizen hinterlegt werden, etwa um Investmententscheidungen zu dokumentieren. Wenn ein Nutzer in ein Asset (z. B. eine Kryptowährung) investiert, wird der investierte Betrag automatisch als Ausgabe verbucht und vom aktuellen Kontostand abgezogen. Gleichzeitig erscheint die Investition als separater Eintrag im Finanzbereich mit folgenden Informationen: Investierter Betrag, Gekaufte Anzahl (Anteile) und Asset-Bezeichnung (z. B. BTC).

**Technologien im Einsatz**

* **Frontend:** Vue.js
* **Backend:** Java Spring Boot
* **Datenbank:** MySQL
  + Zugriff via Spring Data JPA
* **Externe REST-APIs:**
  + CoinGecko (Kryptopreise)
  + ExchangeRate.host (Wechselkurse)
  + Twelve Data (Aktienkurse)

**Mögliche Herausforderungen**

* API-Migration: Ursprünglich wurde eine API verwendet, die alle drei Funktionalitäten (Krypto-, Wechselkurs- und Aktiendaten) bereitstellte. Diese musste durch drei separate APIs ersetzt werden
* Datenbankwechsel: Migration von H2-Datenbank zu MySQL
* Datenbankrelationen: Nachträgliche Implementierung von Beziehungen zwischen Tabellen, da diese initial nicht vorhanden waren

**Projektplanung:**

* Ein Bild, das Text, Screenshot, Diagramm, Reihe enthält.

  KI-generierte Inhalte können fehlerhaft sein.**Ein Bild, das Text, Screenshot, Zahl, Schrift enthält.

  KI-generierte Inhalte können fehlerhaft sein.Gantt-Diagramm**

**Datenbankmodell**

* **ER-Diagramm**

Ein Bild, das Text, Screenshot, Diagramm, Schrift enthält.

KI-generierte Inhalte können fehlerhaft sein.