Medical Care Register

GROUP 06 (Brown)
Final Presentation – Individual Contribution

Course:
"**Software Development Project - AI5094 (SoSe25)„**

Instructor:
"Michael Jahn"

Name: **Huzaifa Khatri** MtrNr.: **1492947**

Project Management

Hochschule Fulda
University of Applied Sciences

Project Summary

- Create a Change Data Capture (CDC) tool (Register)
- Use case: Medical registry for diabetic foot care
- Must follow EU standards and data protection laws

Business Requirements:

- Doctors can register new patients and manage their medical records.
- A unique Patient ID is automatically generated after the first diagnosis.
- Patients can log into the portal using their Patient ID to complete questionnaires.
- Doctors and shoemakers can assign and monitor patient diagnoses and questionnaire.
- Patients have limited access to their data, while doctors have full access to all patient records.

Stakeholders:

- **Patients:** Individuals who receive diagnoses and submit questionnaires after visits. They gain portal access using a Patient ID assigned after the first diagnosis.
- **Doctors / Orthopedic-shoemakers:** Healthcare professionals responsible for creating patients, assigning diagnoses, and monitoring questionnaire submissions to track health progress.

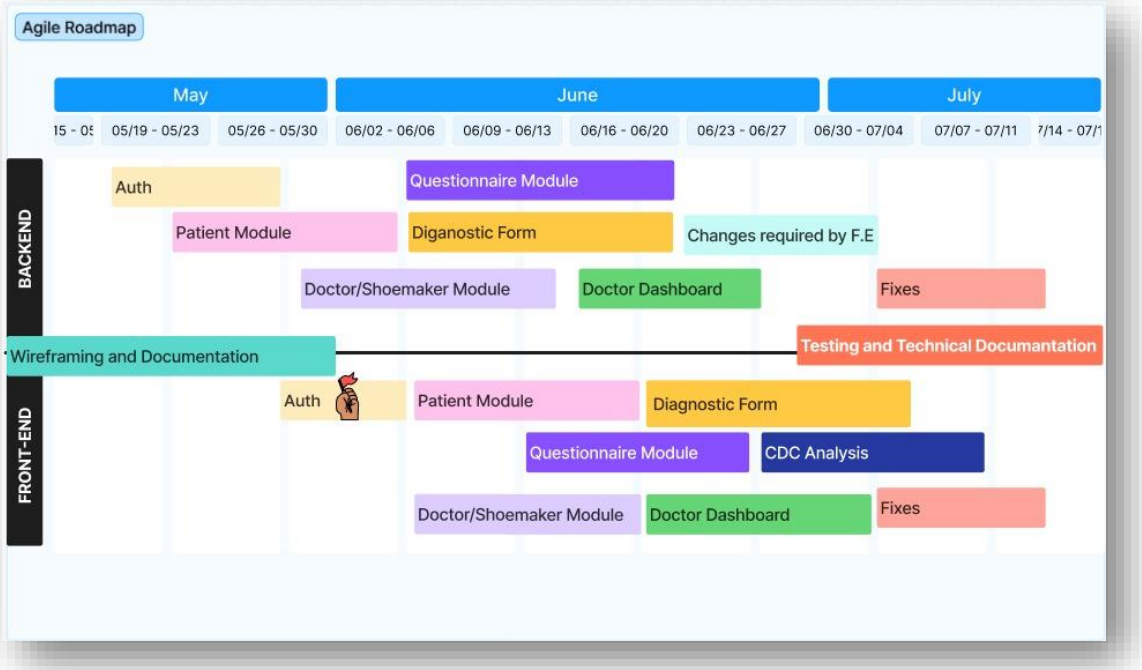
Project Management

- Analyzed the project roadmap, outlining sprint-wise milestones and deliverables
- Facilitated weekly meetings, including sprint planning, retrospectives, and daily coordination
- Managed task distribution and team responsibilities to ensure balanced workload
- Tracked project progress, monitored deadlines, and flagged risks early
- Ensured documentation consistency and maintained records of requirements and design decisions
- Coordinated between frontend and backend teams to align development timelines
- Helped drive team focus and fostered a collaborative, goal-oriented work environment

Project Management

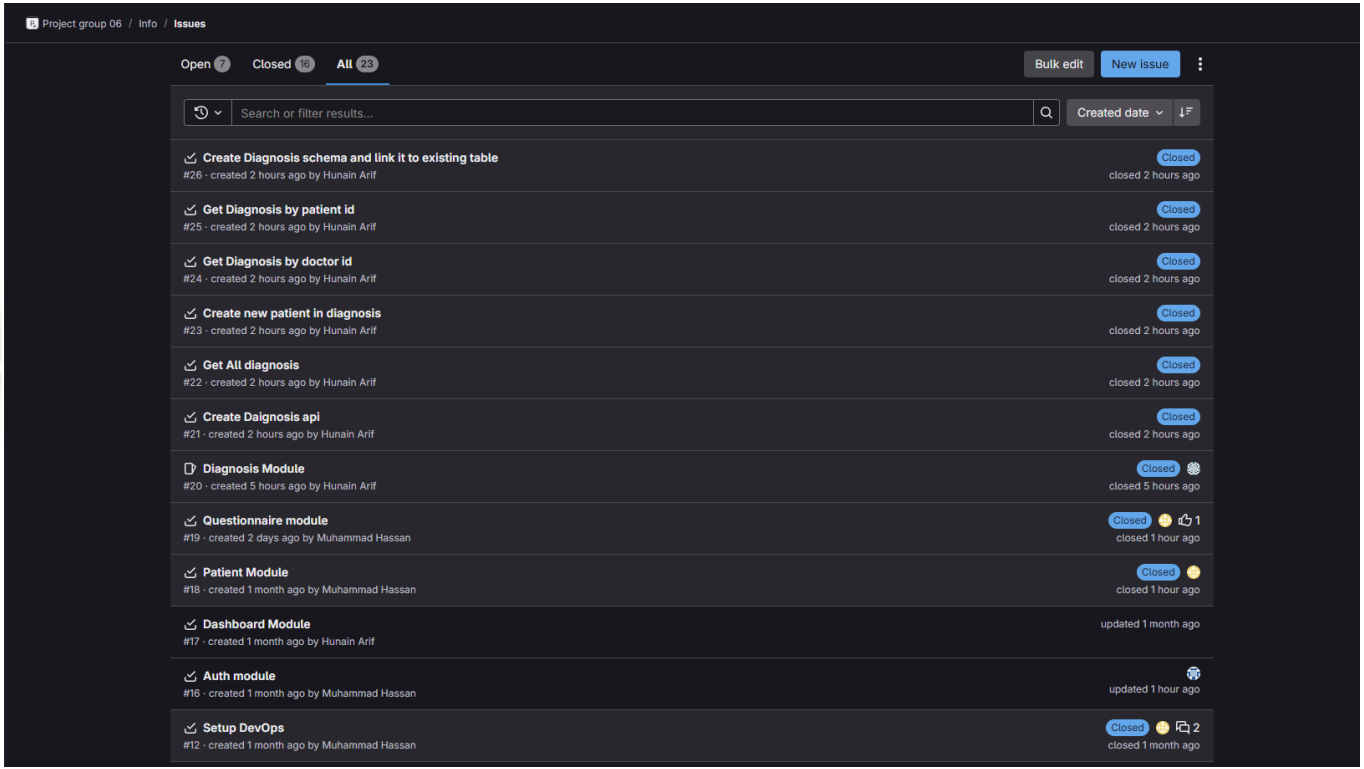
Hochschule Fulda
University of Applied Sciences

Roadmap:



Project Management

Hochschule Fulda
University of Applied Sciences



Contributed to Technical and Requirement documentation

Actively contributed to the creation and maintenance of **requirement specifications** and **technical documentation**, ensuring clear communication of system functionalities, workflows, and development standards. This documentation served as a vital reference for the development team and supported **project alignment and traceability** throughout the lifecycle.

Following are the links to above mentioned documents:

- Technical Documentation for System Information - Technical Document
- Requirement Specification - Requirement Specification

Modularization

We decided to go with a modular approach by segregating the system in three major modules:

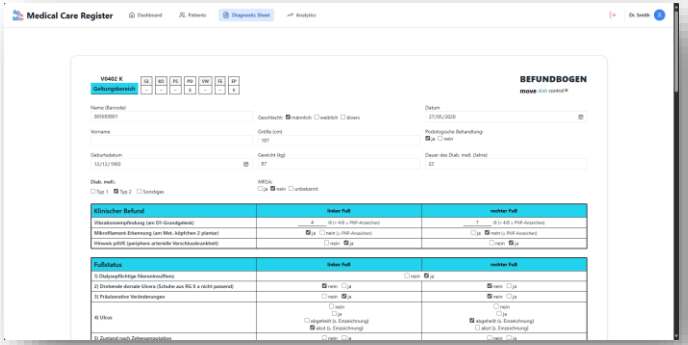
- **Patients & Doctors Module:** Documented Handled user management, including login, role-based access, and profile details. This module also managed the doctor's ability to view and assign diagnostic and questionnaire tasks to patients
- **Questionnaire Module:** Focused on capturing patient inputs through structured forms. Questionnaires were designed by doctors or orthopedic specialists and filled by patients. The module also supported storing, viewing, and analyzing historical responses for comparison.
- **Diagnostic Module:** Enabled doctors to fill out diagnostic sheets for both new and existing patients. Included tools for reviewing previous diagnoses, entering new data, and evaluating patient progress over time.

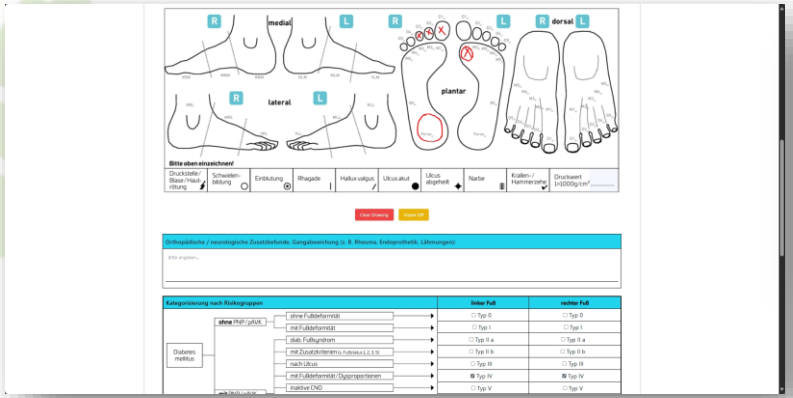
Development (Frontend)

Hochschule Fulda
University of Applied Sciences

Diagnostic Module

Developed the complete frontend code for the diagnostic module. This included multiple functionality for data storing, i.e. conditional inputs, custom drawing input, and multiple data store





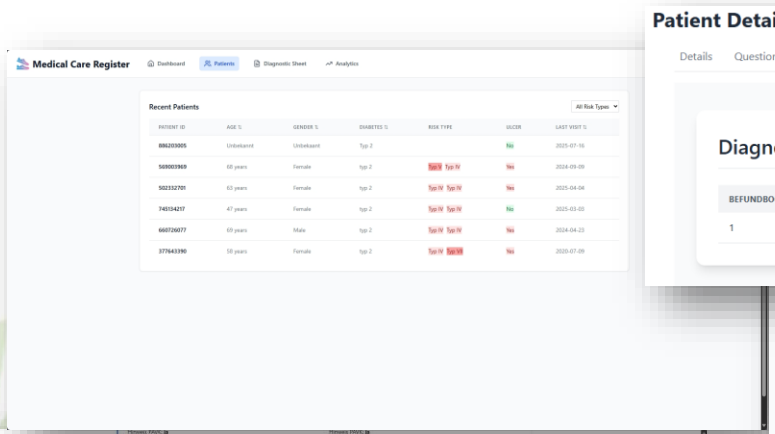
Development (Frontend)

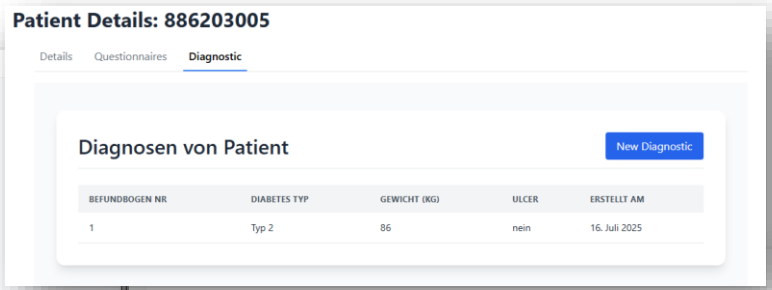
Technologies Used:

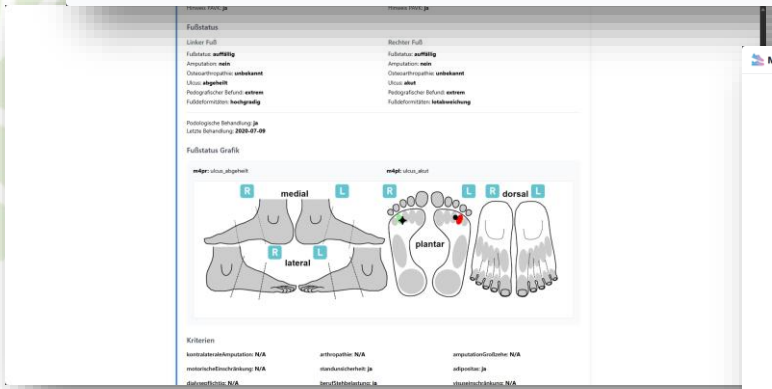
- React.js (Javascript, HTML5, CSS3, Bootstrap)
- Axios for Api calls
- React Router for navigation
- Tailwind Framework
- Windows Speech Systhesis
- Visual Studio Code
- Lucid Charts and Recharts for graphs and charts
- React Hooks for state-management

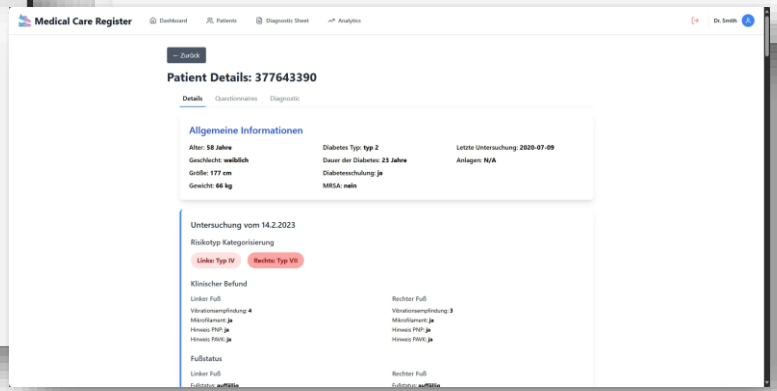
Development (Frontend)

Hochschule Fulda
University of Applied Sciences









Development (Frontend)

Hochschule Fulda
University of Applied Sciences

Custom Data Mapping Functionality

Developed a **custom mapping feature** to ensure that newly added data aligns with the existing data structure. This was crucial for maintaining **data consistency and compatibility**, especially in scenarios involving **data export** for stakeholders. The solution helped prevent structural mismatches and ensured a smooth flow of information across modules.

```
export const mapDiagnoses = (data) => {
  const allgemeineDaten = {
    geburtsdatum: data.geburtsdatum || null,
    groeße: data.groeße || null,
    gewicht: data.gewicht || null,
    geschlecht: data.geschlecht || null,
    diabetestyp: data.diabetestyp || null,
    dauerDiab: data.dauerDiab || null,
    diabSchulung: data.diabSchulung || null,
    anlage: data.anlage || null,
    anlagenSources: data.anlagenSources || null,
    mrsa: data.mrsa || null,
    datum: data.datum || null,
  };

  const klinischerBefund = {
    linksVibrationsempfindung: data.clinicalFindings?.[0]?.leftValue,
    rechtsVibrationsempfindung: data.clinicalFindings?.[0]?.rightValue,
    linksMikrofilament: data.clinicalFindings?.[1]?.leftValue === true ? "ja" : "nein",
    rechtsMikrofilament: data.clinicalFindings?.[1]?.rightValue === true ? "ja" : "nein",
    linksHinweisPAVK: data.clinicalFindings?.[2]?.leftValue === true ? "ja" : "nein",
    rechtsHinweisPAVK: data.clinicalFindings?.[2]?.rightValue === true ? "ja" : "nein",
    linksHinweisPNP: data.clinicalFindings?.[1]?.leftValue === 'nein' || data.clinicalFindings?.[0]?.leftValue < 4 ? 'ja' : 'nein',
    rechtsHinweisPNP: data.clinicalFindings?.[1]?.rightValue === 'nein' || data.clinicalFindings?.[0]?.rightValue < 4 ? 'ja' : 'nein',
  };

  const fussstatus = {
    linksFußstatus: data.fussStatus?.[0]?.value === true ? "ja" : "nein",
    rechtsFußstatus: data.fussStatus?.[0]?.value === true ? "ja" : "nein",
    linksUlcus: data.fussStatus?.[3]?.leftValue === true ? "ja" : "nein",
    rechtsUlcus: data.fussStatus?.[3]?.rightValue === true ? "ja" : "nein",
    linksAmputation: data.fussStatus?.[4]?.leftValue === true ? "ja" : "nein",
    rechtsAmputation: data.fussStatus?.[4]?.rightValue === true ? "ja" : "nein",
    linksFußdeformitäten: data.fussStatus?.[5]?.leftValue?.[0] === true ? "ja" : "nein",
    rechtsFußdeformitäten: data.fussStatus?.[5]?.rightValue?.[0] === true ? "ja" : "nein",
    linksOsteoarthropathie: data.fussStatus?.[7]?.leftValue?.[0] === true ? "ja" : "nein",
    rechtsOsteoarthropathie: data.fussStatus?.[7]?.rightValue?.[0] === true ? "ja" : "nein",
    linksPedograpischerBefund: "extrem",
    rechtsPedograpischerBefund: "extrem",
    podologischeBehandlung: data.diabSchulung || null,
    letztePodoBehandlung: data.datum || null,
  };

  const fussstatusGrafik = {
    grafikDaten: {
      m4pr: data.fussStatus?.[3]?.leftValue || [],
```

Development (Frontend)

Hochschule Fulda
University of Applied Sciences

New Questionnaire Functionality

Implemented a functionality that empowers doctors to **enable or assign new questionnaires** to patients based on specific medical requirements. This dynamic control allows healthcare providers to tailor the data collection process, ensuring that only relevant questionnaires are presented to each patient. The feature enhances **personalization, clinical accuracy**, and supports more effective **patient monitoring and diagnosis**.

Development (Frontend)

API Integration (Frontend)

- Integrated login functionality using a custom API function (logIn) with Axios.
- Handled role-specific credentials: email for doctors and ID for patients.
- Managed successful login redirection to respective dashboards.
- Used Toast notifications to provide instant user feedback for success or error.
- Stored relevant data (like patient ID) in localStorage when needed.
- Implemented error handling and fallback UI behavior for failed API responses.

Bug Fixing as per QA Feedback (Frontend)

Actively participated in **bug fixing cycles** based on detailed feedback from the QA team. This involved identifying root causes, resolving functional and UI-related issues, and ensuring compliance with the expected behavior of each feature. The iterative process helped enhance **system stability, user experience**, and maintain **quality standards** across all modules.

Following is the link that was used to keep track of Test Case for system:
Test Cases - Google Sheets

Thank you!