Medical Care Register

Name: Aniq Ahmed   MtrNr.: 1490381

Medical Care Register

- Medical Care Register for Doctors, Orthopedic Shoe-Makers and Patient to monitor the treatment.

- **Doctor/Orthopedic Shoe-Maker**:
  Dashboard: to view basic analytics of data present.
  Patient details: to visualize diagnostic data, questionnaire response and comparison to look at the progress.
  Analytics: To view detailed analysis using charts of diagnostic data of all patients
  Diagnostic sheet: To fill in a diagnosis for a new or old patient.

- **Patient:**
  Dashboard: To visualize the diagnosis.
  Questionnaires: To fill in questionnaire data prompted by doctor/orthopedic shoe-maker.

My Role

- Build the frontend using react
  Doctor/Orthopedic Shoe-Maker: Dashboard, Analytics, Patient Details
  Patient: Dashboard, Questionnaires

- Maintained the frontend GitLab repository:
  Branching strategy (your_branch->dev->main)
  Merge Requests, task tracking, code reviews.

Tech Stack

- React with Tailwind for styling

- Axios for Api calls

- React Router for navigation

- Lucid Charts and Recharts for graphs and charts

- React Hooks for state-management

```
src/
 ├── api/
 │    ├── api.js
 ├── components/
 │       ├.../
 ├── pages/
 │      ├doctor/
 │       ├.../
 ├── hooks/
 ├── utils/
 ├── data/
 ├── layouts/
 ├── images/
```

Key Features

- Doctor Dashboard

Doctor Dashboard

Shooo | 🏠 Dashboard | 👥 Patients | 📄 Diagnostic Sheet | 📈 Analytics | Navbar to navigate | ↪ Dr. Updated Name 👤

Total Patients
6
Active cases

Average Age
51
Years

High Risk Cases
10
Type VI-VII

Diabetes Type 2
5
Patients

Age Distribution   Expand

Avg Age: **51** | Age Range: **0-69**

Gender Distribution   Expand

Male: **1** | Female: **4**

Risk Type Distribution by Foot   Expand

High: **10** | Medium: **0** | Low: **0**

Stat Cards to show basic analytics

View More Analytics

| Enter Patient ID | **Go to Patient**

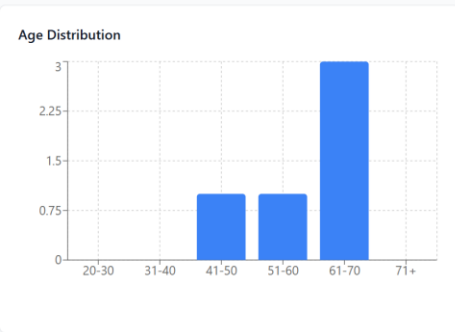Patient Id field to navigate directly to patient details

Key Features

- Doctor Dashboard

- Analytics

Analytics

Total Patients
6
Active cases

Average Age
51
Years

High Risk Cases
10
Type VI-VII

Diabetes Type 2
5
Patients

Age Distribution

Gender Distribution

● Male (16.7%)   ● Female (66.7%)

Risk Type Distribution by Foot

■ Left ■ Right

Diabetes Duration vs. Age

BMI Category Distribution

Neuropathy Prevalence (% with PNP Signs)
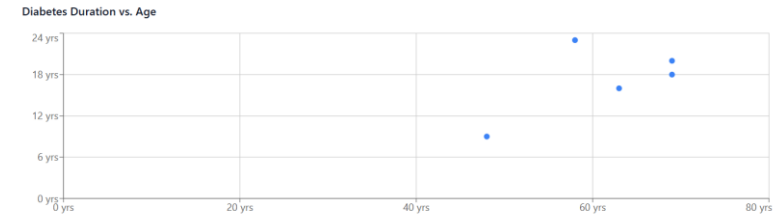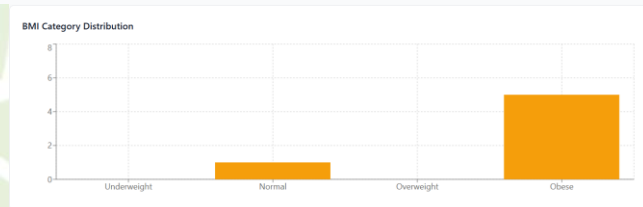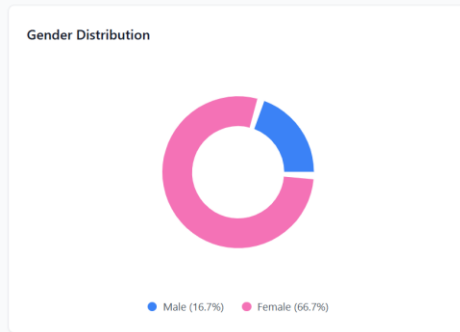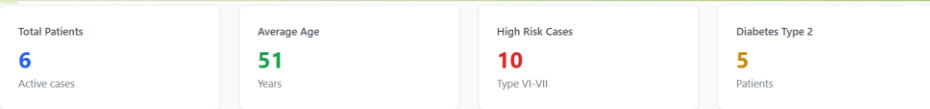
Multiple Charts to show overall patients data

Key Features

- Doctor Dashboard

- Analytics

- Patient Details

Patient Details

Recent Patients

Patients table to summarize data for each patient with risk type filter and sorting on age, gender, diabetes type and last visit

All Risk Types ▼

PATIENT ID	AGE ⇅	GENDER ⇅	DIABETES ⇅	RISK TYPE	ULCER	LAST VISIT ⇅
886203005	Unbekannt	Unbekaant	Typ 2		No	2025-07-16
569003969	68 years	Female	typ 2	Typ V Typ IV	Yes	2024-09-09
502332701	63 years	Female	typ 2	Typ IV Typ IV	Yes	2025-04-04
745134217	47 years	Female	typ 2	Typ IV Typ IV	No	2025-03-03
660726077	69 years	Male	typ 2	Typ IV Typ IV	Yes	2024-04-23
377643390	58 years	Female	typ 2	Typ IV Typ VII	Yes	2020-07-09

Patient Details

Patient Details: 377643390

Individual Patient Diagnostic details

Details | Questionnaires | Diagnostic

Allgemeine Informationen

Alter: **58 Jahre**   Diabetes Typ: **typ 2**   Letzte Untersuchung: **2020-07-09**

Geschlecht: **weiblich**   Dauer der Diabetes: **23 Jahre** Anlagen: **N/A**

Größe: **177 cm**   Diabetesschulung: **ja**

Gewicht: **66 kg**   MRSA: **nein**

Kriterien

kontralateraleAmputation: **N/A**   arthropathie: **N/A**   amputationGroßzehe: **N/A**

motorischeEinschränkung: **N/A**   standunsicherheit: **ja**   adipositas: **ja**

dialysepflichtig: **N/A**   berufStehbelastung: **ja**   visuseinschränkung: **N/A**

fehlgeschlageneVorversorgung: **N/A** fußdeformitätenDruckerhöhung: **ja**

Schuhversorgung

linksTyp4DiabetecFußbettung: **ja**   rechtsTyp7Entlastungsorthese: **ja**

Funktionelle Tests

Linker Fuß   **Rechter Fuß**

Push-Up: **positiv**   Push-Up: **positiv**

Dorsalextension: **eingeschränkt**   Dorsalextension: **eingeschränkt**

Plantarflexion: **eingeschränkt**   Plantarflexion: **eingeschränkt**

Silf Test: **N/A**   Silf Test: **N/A**

Vorfuß Entlastung: **proniert**   Vorfuß Entlastung: **supiniert**

Vorfuß Belastung: **supiniert**   Vorfuß Belastung: **N/A**

Zehen Krallenzehen: **N/A**   Zehen Krallenzehen: **ja**

Belastung Erster Strahl: **nein**   Zehen Drehen Ein: **ja**

Coleman Test: **N/A**   Belastung Erster Strahl: **ja**

Gang Innenrotiert: **mittel**   Coleman Test: **N/A**

Gang Außenrotiert: **stark**   Coleman Test Höhe: **N/A**

                                        Gang Innenrotiert: **mitte**

                                        Gang Außenrotiert: **stark**

Erstellt am: **14.2.2023, 13:44:10**   Aktualisiert am: **14.2.2023, 13:44:10**   Befundbogen ID: **4**   Partner ID: **2**

Untersuchung vom 14.2.2023

Risikotyp Kategorisierung

Links: Typ IV   Rechts: Typ VII

Klinischer Befund

Linker Fuß   **Rechter Fuß**

Vibrationsempfindung: **4**   Vibrationsempfindung: **3**

Mikrofilament: **ja**   Mikrofilament: **ja**

Hinweis PNP: **ja**   Hinweis PNP: **ja**

Hinweis PAVK: **ja**   Hinweis PAVK: **ja**

Fußstatus

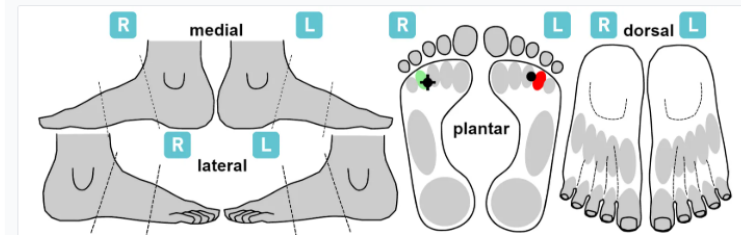**Linker Fuß**   **Rechter Fuß**

Fußstatus: **auffällig**   Fußstatus: **auffällig**

Amputation: **nein**   Amputation: **nein**

Osteoarthropathie: **unbekannt**   Osteoarthropathie: **unbekannt**

Ulcus: **abgeheilt**   Ulcus: **akut**

Pedografischer Befund: **extrem**   Pedografischer Befund: **extrem**

Fußdeformitäten: **hochgradig**   Fußdeformitäten: **lotabweichung**

Podologische Behandlung: **ja**

Letzte Behandlung: **2020-07-09**

Fußstatus Grafik
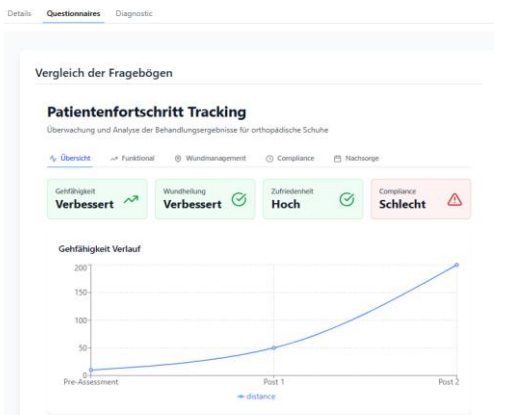
m4pr: ulcus_abgeheilt   m4pl: ulcus_akut

Key Features

- Doctor Dashboard

- Analytics

- Patient Details

- Questionnaire Comparison

Questionnaire Comparison

Questionnaire Pre and Post comparison using charts and cards to visualize progress in patient

Key Features

- Doctor Dashboard

- Analytics

- Patient Details

- Questionnaire Comparison

- Patient Dashboard

Patient Dashboard

Patient Dashboard to visualize diagnostic data

Key Features

- Doctor Dashboard

- Analytics

- Patient Details

- Questionnaire Comparison

- Patient Dashboard

- MOS Questionnaire

MOS Questionnaire

Hochschule Fulda
University of Applied Sciences

MOS Questionnaire
Vorerhebung
✓ Previous responses restored

Auto Save Functionality saves the progress and on next loading patient can continue where they left

A — B — ✓ — D — E — F — G — H — I
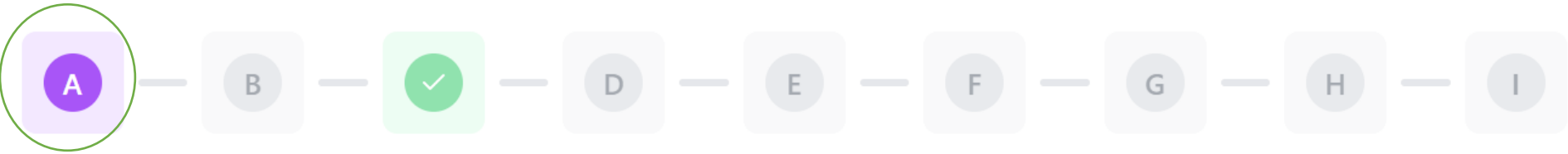
Vorerhebung

0% Complete (0/17 questions)

Multiple progress bars to show patient where they currently are.
A,B,C... for section wise progress.
Percentage for overall progress
Page numbers to show how many pages are left
Page progress at bottom
Can move forward only if current page is filled completely

A. Persönliche Angaben
Vorerhebung (Schritt 1 von 9)

‹ Zurück   Weiter ›

MOS Questionnaire

Speaker button to read aloud the question

🔊 **1. Wie weit können Sie aktuell gehen?** *

ICF: d450 Gehen

Emojis to visualize the answer   Read Aloud on every option as it is clicked

🏠 Ich kann mich nur in meiner Wohnung bewegen (0-10 Meter)

👋 Ich kann zum Nachbarn gehen (10-50 Meter)

🚶 Ich kann bis zur Straßenecke gehen (50-200 Meter)

🏪 Ich kann zu Geschäften in der Nachbarschaft gehen (200 Meter - 1 Kilometer)

🚶💪 Ich kann längere Strecken ohne Pause gehen (mehr als 1 Kilometer)

MOS Questionnaire

Conditional rendering of questions according to user response

MOS Questionnaire
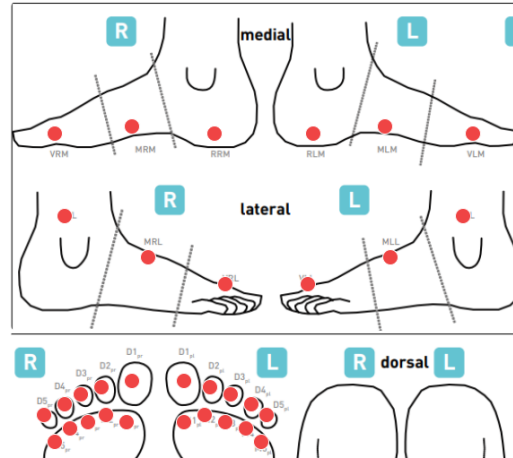
Questionnaires table to show assigned questionnaires and progress of each questionnaire

Fragebögen von Patient #377643390

#	TYP	STATUS	FORTSCHRITT	ERSTELLT AM	AKTUALISIERT AM
1	Pre	⌛ In Bearbeitung	29% 5 von 17 Fragen	13. Juli 2025	16. Juli 2025
2	Post	○ Nicht begonnen	0% 0 von 29 Fragen	16. Juli 2025	16. Juli 2025

Codes and Explanation

Import Statements

```
import React, { useState, useMemo, useEffect } from "react";
import StatsCard from "../../components/StatsCard";
import { useLatestPatients } from "../../hooks/useLatestPatients";
import { calculateAge } from "../../utils/calculateAge";
import { highRiskCount } from "../../utils/highRiskCount";
import AgeDistributionChart from "../../components/AgeDistributionChart";
import GenderDistribution from "../../components/GenderDistribution";
import RiskComparisonChart from "../../components/RiskComparisionChart";
import { useNavigate } from "react-router-dom";
import ChartWrapper from "../../components/ChartWrapper";
import { fetchDiagnosticData } from "../../api/api";
```

State Variable Declaration

```
const [showAgeChart, setShowAgeChart] = useState(false);
const [showGenderChart, setShowGenderChart] = useState(false);
const [showRiskChart, setShowRiskChart] = useState(false);
const [diagnosticData, setDiagnosticData] = useState([]);
const [loading, setLoading] = useState(true);
const [error, setError] = useState(null);
const navigate = useNavigate();
```

Use Effect for API calls

```
useEffect(() => {
  const fetchData = async () => {
    try {
      setLoading(true);
      const doctorId = localStorage.getItem("doctorId");

      if (!doctorId) {
        setError("Doctor ID not found in localStorage");
        setLoading(false);
        return;
      }

      const response = await fetchDiagnosticData(doctorId);
      setDiagnosticData(response);
      setError(null);
    } catch (err) {
      console.error("Error fetching diagnostic data:", err);
      setError("Failed to fetch diagnostic data");
    } finally {
      setLoading(false);
    }
  };
};
```

Axios to call Api from Backend

```
import axios from "axios";
import { mapDiagnoses } from '../utils/mapDiagnoses';

const BASE_URL = "https://pg06.regifor.de/api/v1";

const api = axios.create({
  baseURL: BASE_URL,
});

export const fetchQuestionnairesByPatientId = async (patientId) => {
  try {
    const response = await api.get(`/questionnaire/${patientId}`);
    return response.data.data;
  } catch (error) {
    console.error("Error fetching questionnaires:", error);
    return [];
  }
};
```

Custom Functions to Handle Data

```
const getPatientFriendlyValue = (value, type) => {···
};

const getRiskExplanation = (riskType) => {···
};

const getSensationExplanation = (value) => {···
};

const getFootStatusExplanation = (value) => {···
};
```

Codes and Explanation

- Questionnaire Code

```
// Initialize responses from saved data
useEffect(() => {...
}, [savedResponses, sections]);

// Find first incomplete section
const findFirstIncompleteSection = (responseState) => {...
};

// Modified to accept optional responseState parameter
const getVisibleQuestions = (section, responseState = responses) => {...
};
//Conditional question rendering
const shouldShowQuestion = (question, responseState = responses) => {...
};
//Get all visible questions with/without conditions
const getAllVisibleQuestions = () => {...
};
//Validate questions visibility is current section
const validateCurrentSection = () => {...
};
//Validate when a section is complete
const validateSectionCompletion = (section) => {...
};
//Check for next section
const canNavigateToSection = (targetIndex) => {...
};
```

```
// Auto-save functionality
useEffect(() => {...
}, [responses, onSave]);
//Mapping questionnaire response
const formatQuestionnaireResponses = (responses, sections) => {...
};
```

```
//Custom Hook to Read Aloud
import { useEffect, useState, useRef } from "react";

const useSpeech = () => {
  const [voices, setVoices] = useState([]);
  const pendingText = useRef(null);

  useEffect(() => {...
  }, []);

  const speakText = (text, voiceList = voices) => {...
  };

  const handleReadAloud = (text) => {...
  };

  return handleReadAloud;
};

export default useSpeech;
```

```
//Question rendering based on question type
const renderQuestionContent = () => {
  switch (question.type) {
    case "radio":...

    case "checkbox":...

    case "scale":...

    case "textarea":...

    case "text":...

    case "wounds_image":...

    default:
      return null;
  }
};
```