DYNAMIC WEB APPLICATION WEEK 10 DOCUMENTATION

Outline

This app is a simple health tracker system that allows users to record physical activities. The main features are user registration, login, adding activities and view old ones. Each activity has information such as duration, distance and optional notes.

The goal of the project is to practise working with server-side JavaScript, EJS and MySQL. The system validates user input, stores passwords.

The design focuses on usability and ease. Pages are consistent in terms of the styling and focuses on user-friendliness.

Architecture

The application uses a two-tier architecture with an application tier built using Node.js, Express, EJS templates, and Express-Session, and a data tier implemented with MySQL.

Data Model

The system uses two relational tables: users and activities. The users table stores personal information and a bcrypt-hashed password. The activities table stores exercise entries and references the user via a foreign key (user_id). This ensures that each activity belongs to exactly one user.

User Functionality

The application has several features designed to guide users smoothly through registering, logging in, and recording activities.

Home Page

The home page introduces the application and provides links to register, log in, or add/view activities.

Registration

Users can make an account by putting their details into a form. The server validates the input using Express-Validator and hashes passwords using bcrypt then storing it into the database. After registration, the user is shown a confirmation screen offering a login or return home option.

Login

Users enter their username and password. The system retrieves the stored bcrypt hash and compares it with the submitted password. The login success page displays a welcome message and a button to navigate home.

Adding Activities

The "Add Activity" page provides a form where users enter the type of activity, duration, distance, date, and optional notes. When submitted, the app inserts the activity into the database using the user’s session ID. A success page confirms the activity was added and offers options to go home, add activities or view the activity list.

Viewing Activities

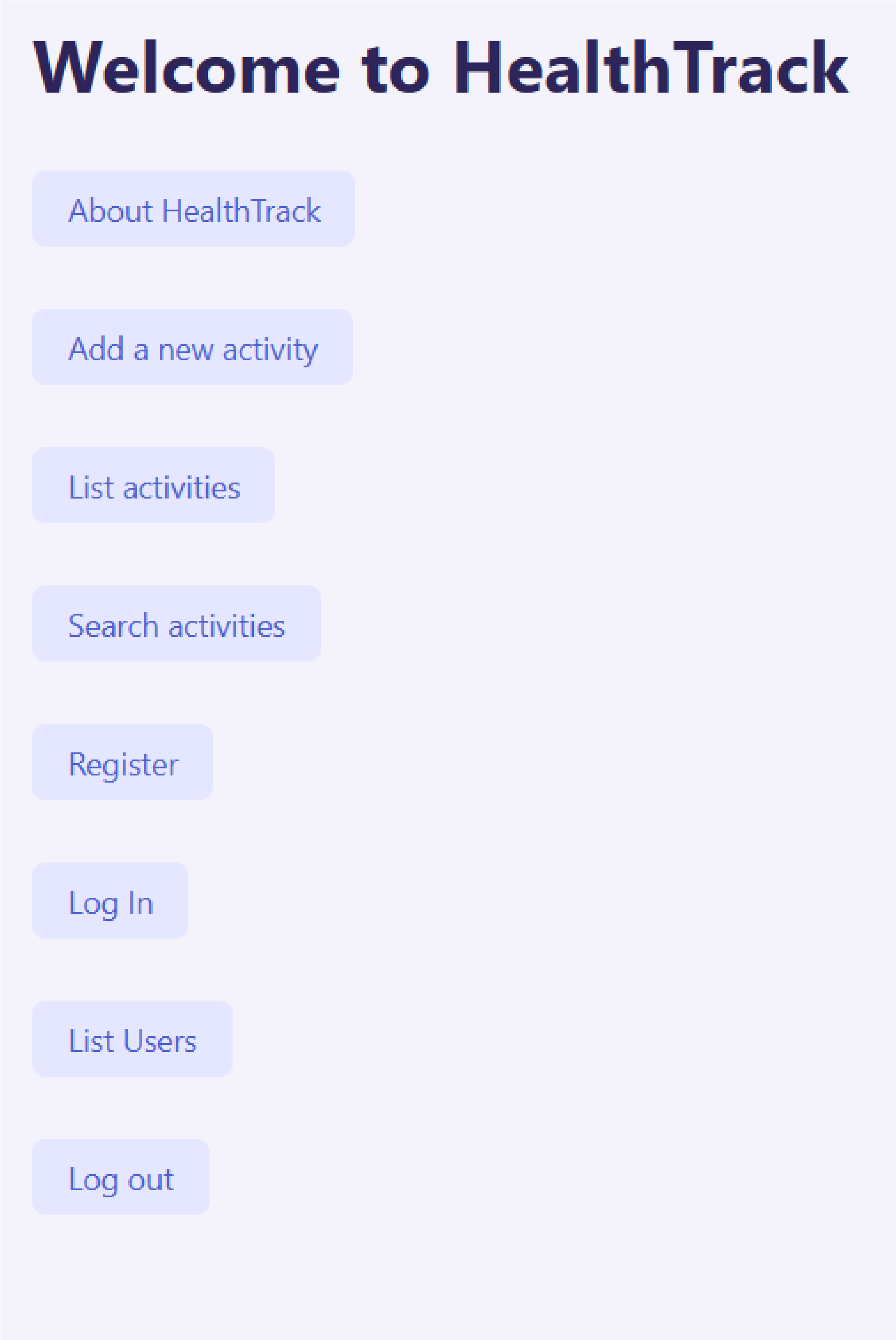
The list page retrieves all activities linked to the logged-in user and displays them in a clean layout. Each entry shows the activity type, duration, distance, date, and notes. A button on this page allows the user to immediately go to the “Add Activity” form.

About Page

The About page explains the purpose of the app and includes a navigation link back to the Home page for convenience.

Visual Design

The interface uses a simple layout clean styling. Buttons are styled to stand out from normal text, and each page includes clear sections to help users understand where they are and what actions they can take next. The design focuses on readability and consistency.



Welcome to HealthTrack

AI Declaration

AI assistance was used to help debug some errors, improve my understanding of bcrypt hashing and session handling. The code, design, and final implementation were completed by me. AI was used as support, not for generating solutions. All code included in the submission was written, adapted, and tested by me to finish the assignment.