**High-Level Design Document**

Mini Project

Contents

[1 Project Overview 2](#_Toc137748514)

[1.1 Brief introduction of the project. 2](#_Toc137748515)

[1.2 Purpose of the project. 2](#_Toc137748516)

[1.3 Goals of the project: 2](#_Toc137748517)

[2 Project Scope 3](#_Toc137748518)

[2.1.1 Platforms and devices the application will support (e.g., iOS, Android). 3](#_Toc137748519)

[2.1.2 Key features and functionalities of the application. 3](#_Toc137748520)

[3 Architecture and Technology Stack 3](#_Toc137748521)

[3.1.1 Overall architecture of the application 3](#_Toc137748522)

[3.1.2 Architecture 4](#_Toc137748523)

[3.1.3 Technology stack (e.g., programming languages, frameworks, libraries). 4](#_Toc137748524)

[3.1.4 Rationale behind the chosen technology stack. 5](#_Toc137748525)

[4 Mobile App Components 6](#_Toc137748526)

[4.1.1 Main components of the application. 6](#_Toc137748527)

[4.1.2 Purpose of each component. 6](#_Toc137748528)

[5 User Interface Design 8](#_Toc137748529)

[5.1.1 User interface (UI) design 10](#_Toc137748530)

# Project Overview

## Brief introduction of the project.

The project focuses on the development of an Employee Training Management System (ETMS), aimed at revolutionizing the training process for interns and employees within an organization. The envisioned platform will serve as a centralized hub for accessing, tracking, and managing training activities, with the overarching goal of enhancing learning outcomes and streamlining administrative processes.

## Purpose of the project.

The purpose of this project is to empower patients and enhance their experience by providing them with a comprehensive instrument tracking solution.

The key goals of the project include:

* Streamline Training Processes: Employee Training Management System (ETMS) will centralize and automate training administration, making it easier for interns and employees to access, enroll in, and track training activities.
* Enhance Learning Outcomes: User-friendly platform with personalized training plans and assessments to improve employee engagement, knowledge retention, and skill development, ultimately driving organizational growth and success.

## Goals of the project:

* Develop an intuitive user interface: Create a user-friendly interface that is easy to navigate and understand, ensuring interns and employees can access training materials and track progress without confusion.
* Ensure cross-platform compatibility: Build a responsive web application that works seamlessly across different devices and screen sizes, allowing users to access training resources from desktops, laptops, tablets, and mobile phones.
* Enhance accessibility and availability: Ensure the web application is accessible from any location with internet connectivity, enabling interns and employees to engage in training activities at their convenience, promoting continuous learning.Deployment to App Store and Play Store.
* Facilitate real-time progress tracking: Implement features for tracking training progress and performance metrics in real-time, enabling users to monitor their learning journey, identify areas for improvement, and stay motivated.

# Project Scope

## Platforms and devices the mobile app will support (e.g., iOS, Android).

|  |  |
| --- | --- |
| Platforms | Version |
| iOS | 11 and above |
| Android | 10 and above |

## Key features and functionalities of the mobile app.

**User Registration and Authentication:**

Streamlined registration process for interns and employees to create their accounts.

Secure authentication mechanisms, including username/password

**Personalized User Profiles:**

Customized profiles for each user, containing details such as name, email, role, and department.

Ability for users to update their profile information and preferences as needed..

**Interactive Training Dashboard:**

Tailored dashboard for each user, providing an overview of their training progress, upcoming sessions, and recommended modules.

**Responsive Design:**

Mobile-friendly design to support various devices, including iPads and smartphones

Responsive user interface for optimal viewing and usability across different screen sizes

# Architecture and Technology Stack

## Overall architecture of the application.

The Employee Training Management System follows a robust and scalable architecture to ensure optimal performance, security, and usability. The architecture consists of the following components:

**Client-Side Interface:**

The client-side interface serves as the user-facing component of the Employee Training Management System (ETMS) application. It is designed to provide interns and employees with an intuitive and responsive user interface, enabling them to access their personalized dashboards, track training progress, and receive notifications.

**Server-Side Layer:**

The server-side layer hosts the application logic and processes user requests received from the client-side layer. It includes server-side programming languages (e.g., Node.js, Python) and frameworks responsible for handling business logic, authentication, authorization, and data processing. This layer interacts with the database layer to fetch or store data and responds to client requests with appropriate data or actions.

**Database Layer**:

The database layer stores and manages the application's data, including user profiles, training modules, session schedules, and assessment results. It uses database management systems (e.g., MySQL, PostgreSQL, MongoDB) that store structured or unstructured data according to the application's requirements.

## Technology stack (e.g., programming languages, frameworks, libraries).

**Programming Languages:**

* JavaScript: Used for both front-end and back-end development.
* Python: Utilized for scripting, data processing, and machine learning tasks.

**Frameworks and Libraries:**

* React.js: A JavaScript library for building interactive user interfaces. Utilized for the client-side interface to create dynamic and responsive user experiences.
* Node.js: A JavaScript runtime environment enabling server-side development, utilized for building the back-end logic and APIs.
* Express.js: A minimalist web framework for Node.js, used for creating RESTful APIs and handling server-side routing and middleware.
* MongoDB: A NoSQL database for storing and managing training data, user information, and session details.
* Snowflake: A cloud-based data warehousing platform for storing and analyzing large volumes of data generated by the application.
* dbt (Data Build Tool): Used for data transformation and modeling within Snowflake, enabling the creation of analytical datasets and documentation generation.
* scikit-learn: A machine learning library for Python, utilized for building and training machine learning models to predict training outcomes and provide personalized recommendations.
* Material UI: A popular React UI framework providing pre-designed components and themes for building modern, responsive web applications.
* React Navigation: React Navigation is a navigation library for React Native applications. It provides a flexible and customizable solution for handling navigation between screens, including stack navigation, tab navigation, and drawer navigation.
* Axios: Axios is a widely used JavaScript library for making HTTP requests from the application to the server. It simplifies the process of sending and receiving data, handling API calls, and managing network requests in the mobile app.

## Rationale behind the chosen technology stack.

The selection of the technology stack for the Employee Training Management System (ETMS) web application is based on several key considerations:

**Cross-Platform Compatibility:**

React, chosen for its ability to build user interfaces, ensures cross-platform compatibility by enabling the creation of web applications that can be accessed seamlessly across different devices and operating systems. This eliminates the need for separate development efforts for various platforms, optimizing user accessibility and reducing development complexity.

**Time and Cost Efficiency:**

The selection of React and associated libraries such as React Navigation and Material-UI contributes to time and cost efficiency. React's component-based architecture streamlines development processes, allowing for rapid prototyping, iterative development, and efficient code maintenance. Additionally, the availability of pre-designed UI components in Material-UI accelerates development cycles, reducing time-to-market and overall development costs.

**Native-Like Performance:**

React Native allows us to develop mobile applications with native-like performance and user experience. It achieves this by utilizing native components and APIs specific to each platform, resulting in smooth animations, fast response times, and seamless integration with device features. This native performance enhances the overall usability and responsiveness of the instrument tracking application.

**Large Developer Community and Ecosystem:**

React Native boasts a vast and active developer community. This thriving community provides extensive support, resources, and libraries that accelerate development and problem-solving. The availability of a wide range of open-source libraries and components allows us to leverage existing solutions and integrate additional functionalities efficiently.

**Code Reusability:**

The chosen technology stack, centered around React, promotes code reusability through its modular architecture and component-based design. Components developed using React can be easily reused across different parts of the application, minimizing redundant code and enhancing development efficiency. This not only reduces development time but also ensures consistency and scalability in code management.

**Flexibility and Customizability:**

React and its associated libraries offer a high degree of flexibility and customizability, allowing developers to tailor the ETMS application to specific requirements and user preferences. React's ecosystem provides a wide range of third-party libraries and tools that enable developers to extend functionality, integrate with external services, and customize user interfaces according to project needs.

**Data Management and Analysis:**

MongoDB serves as the database for storing and managing training data, offering flexibility in schema design and scalability for handling diverse data types.

Snowflake was chosen as the data warehousing platform for its cloud-based architecture and powerful analytics capabilities, facilitating advanced data analysis and reporting.

# Mobile App Components

## Main components of the application.

1. **User Management Module:**

• Allows administrators to create, update, and manage user accounts for employees, trainers, and administrators.

• Provides functionalities for user authentication, role-based access control, and user profile management.

1. **Training Management Module:**

• Enables administrators to create, schedule, and manage training programs, modules, and sessions.

• Facilitates the assignment of training materials, resources, and assessments to users based on their roles and requirements.

1. **Progress Tracking Module:**

• Allows employees to track their training progress, view completed modules, and monitor upcoming sessions.

• Provides visualizations and reports on individual and team progress to administrators for performance monitoring.

1. **Assessment and Reporting Module:**

• Facilitates the creation and administration of assessments, quizzes, and tests to evaluate user knowledge and skills.

• Generates personalized assessment reports for users based on their performance, providing insights and recommendations for improvement.

1. **Machine Learning Integration:**

• Integrates machine learning models for predictive analytics, allowing the system to predict training outcomes, recommend personalized learning paths, and estimate completion times.

1. **Integration with External Tools:**

• Integrates with external tools and services such as Power BI for advanced analytics and visualization, enhancing decision-making and strategic planning capabilities.

1. **Documentation and Documentation Generation Module:**

• Automatically generates documentation for data models, lineage, and dependencies using dbt, ensuring transparency and compliance in data management processes.

## Purpose of each component.

**1. User Management Module:**

• Purpose: Manages user accounts, authentication, and permissions within the system.

• Allows administrators to create, update, and delete user accounts.

• Ensures secure access to training resources based on user roles and permissions.

**2. Training Management Module:**

• Purpose: Facilitates the creation, scheduling, and administration of training programs.

• Allows administrators to define training modules, sessions, and resources.

• Enables assignment of training materials and assessments to users.

**3. Progress Tracking Module:**

• Purpose: Tracks and monitors the progress of users in their training activities.

• Allows employees to view their completed modules, upcoming sessions, and overall progress.

• Provides administrators with insights into individual and team progress for performance monitoring.

**4. Assessment and Reporting Module:**

• Purpose: Evaluates user performance through assessments and generates reports.

• Administers quizzes, tests, and assessments to measure user knowledge and skills.

• Generates personalized assessment reports based on user performance, providing insights and recommendations.

**5. Machine Learning Integration:**

• Purpose: Integrates machine learning models for predictive analytics and personalized recommendations.

• Predicts training outcomes, completion times, and recommends tailored learning paths for users.

**6. Integration with External Tools:**

• Purpose: Integrates with external tools and services for advanced analytics and visualization.

• Enables the use of tools like Power BI for data analysis, reporting, and decision-making.

• Enhances the system's capabilities by leveraging external resources and expertise.

**7. Documentation and Documentation Generation Module:**

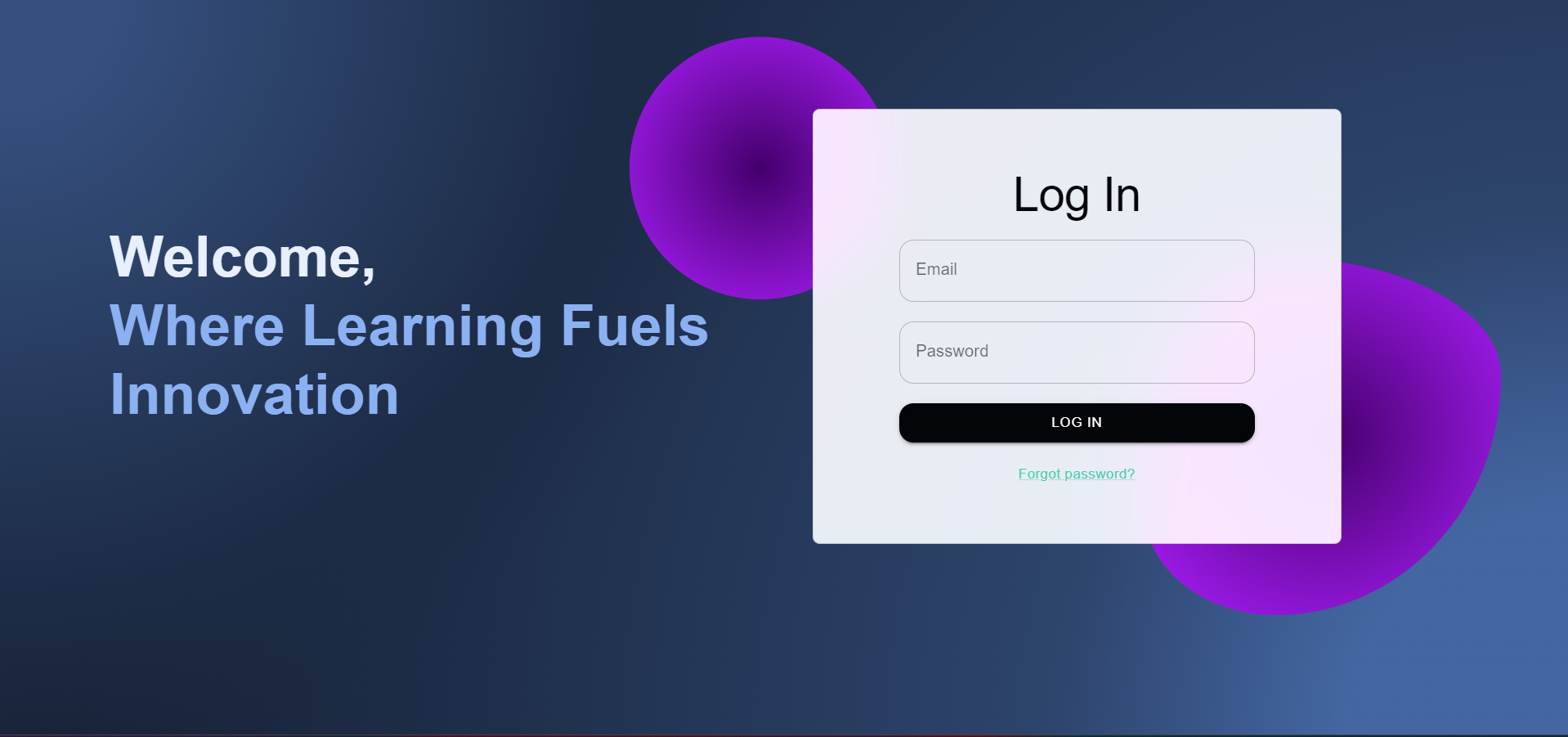
• Purpose: Generates documentation for data models, lineage, and dependencies.

• Provides transparency and compliance in data management processes.

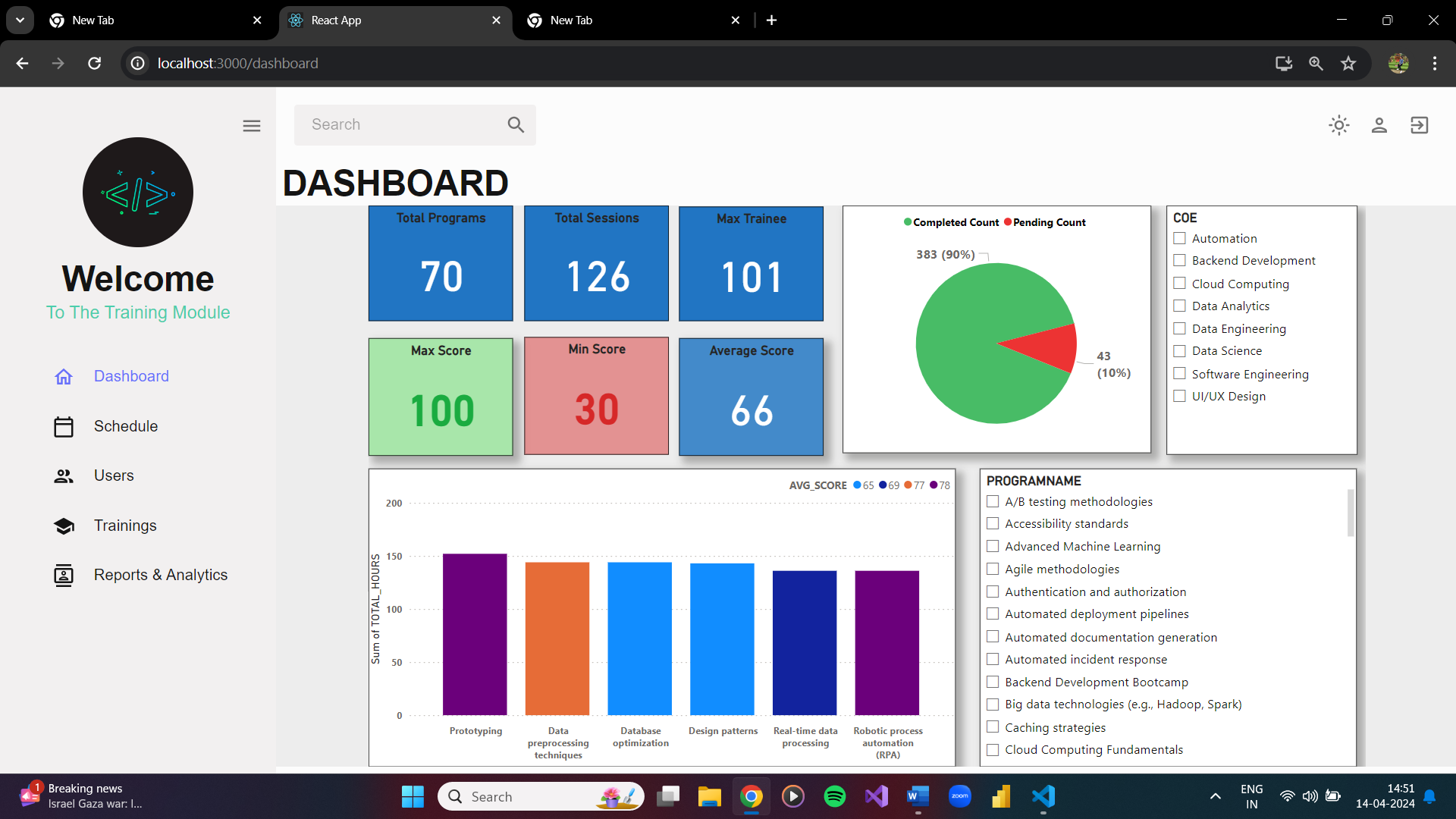
# User Interface Design

### User interface (UI) design .

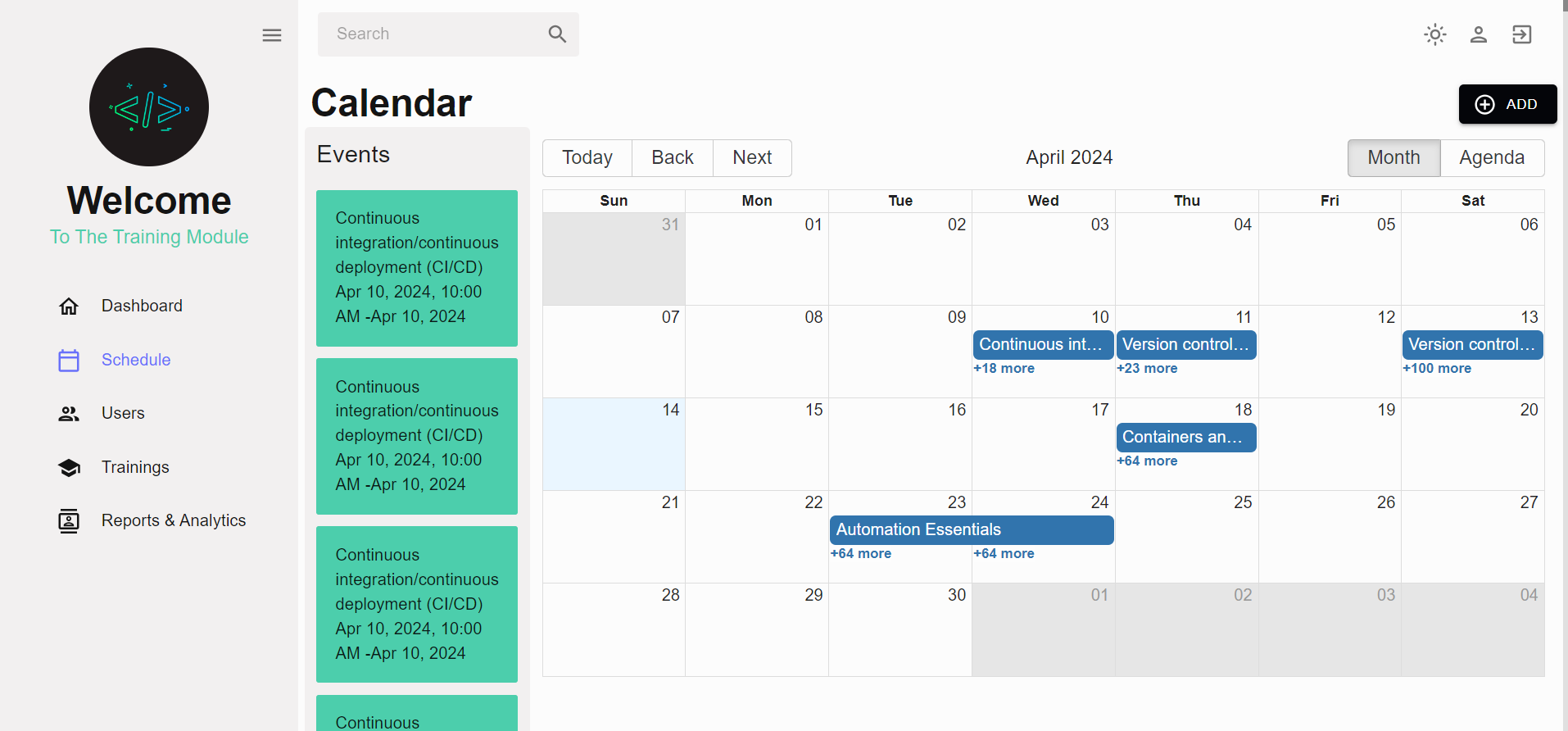
**Log In Page**



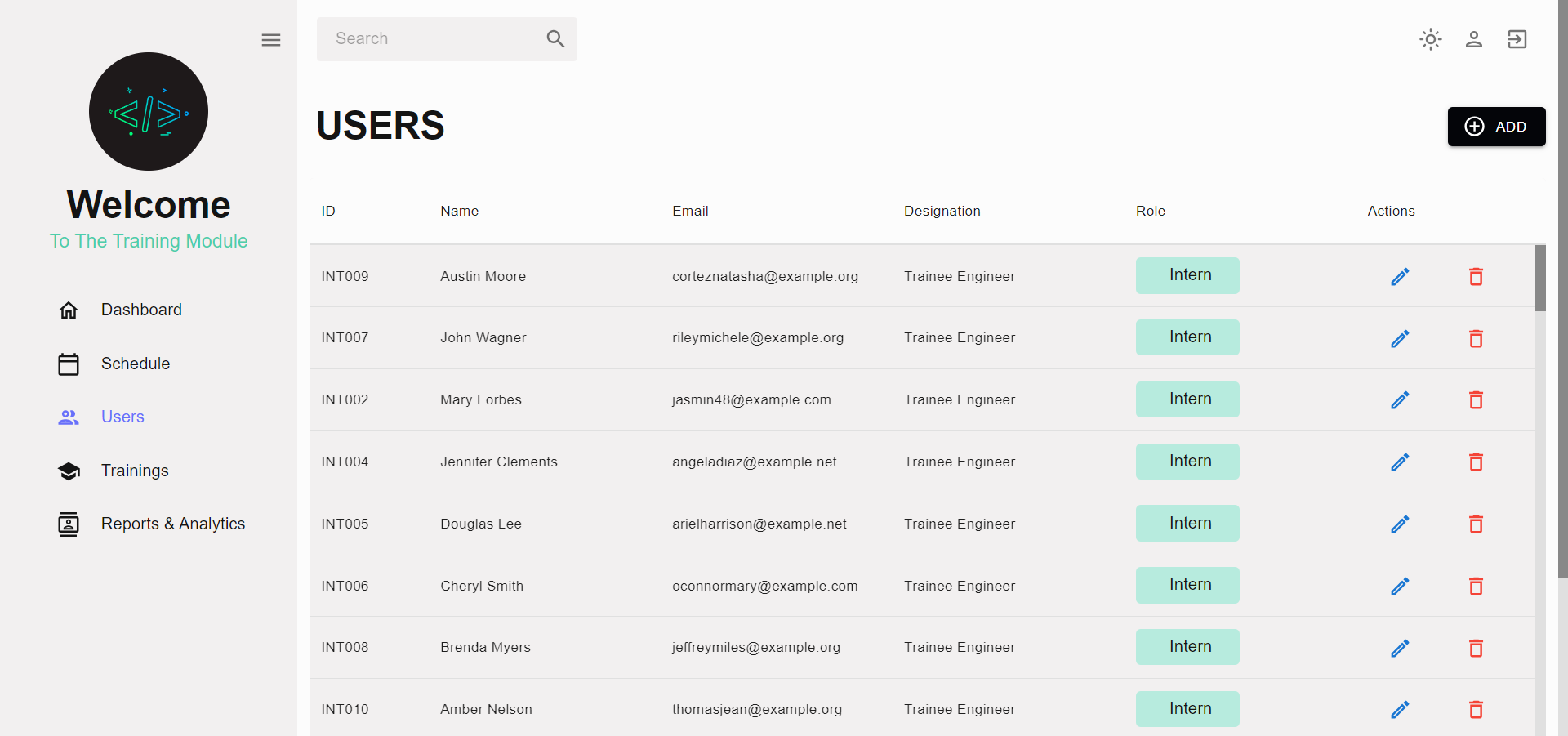
**Admin Dashboard**



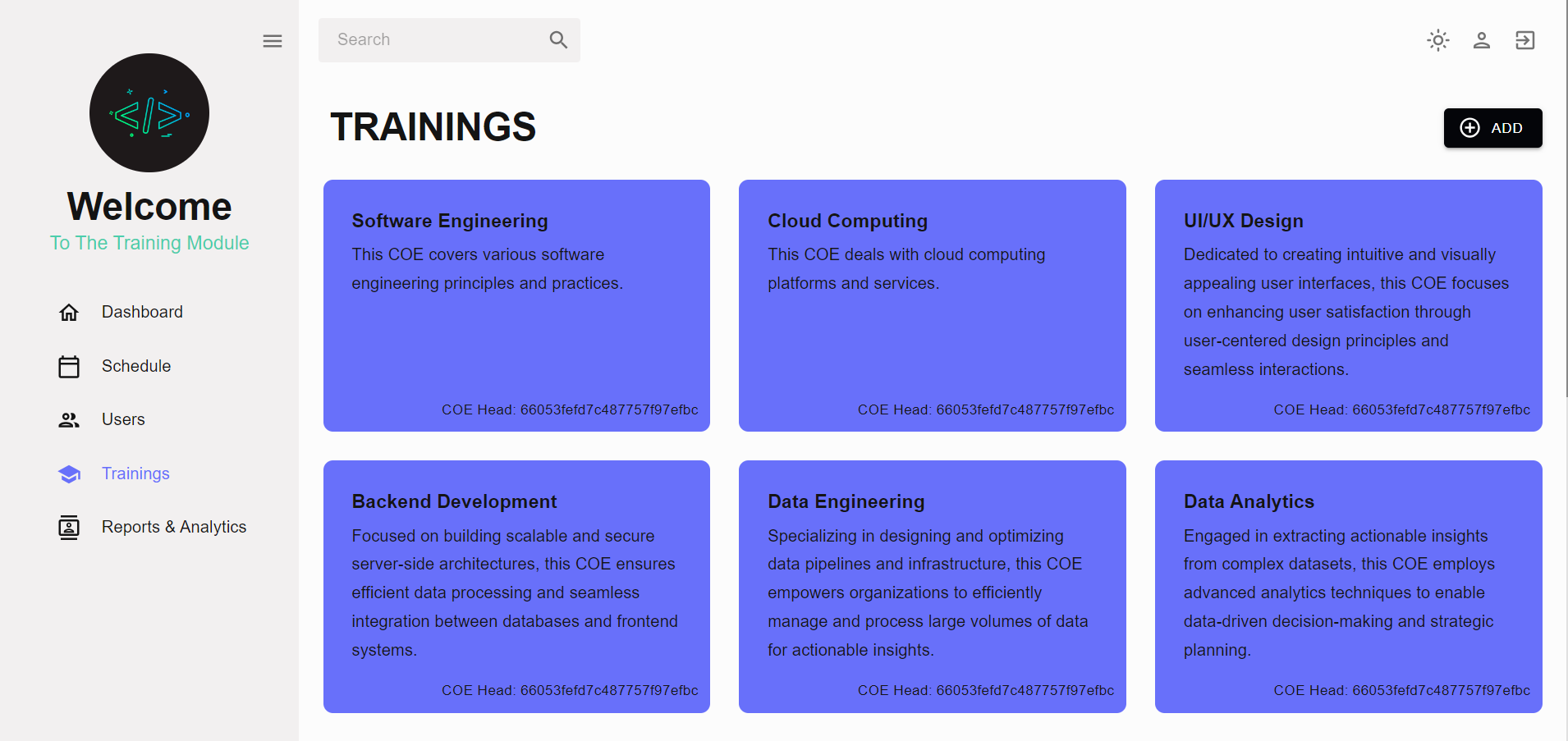
**Calendar**



**Users Detail Page**



**Trainings Page**

****