JMAN GROUP

Catalyst - Employee Training Management System [ETMS]

Mihir Mahim - INT051

Final Project

Table of Contents

1 Project Overview

1.1.1 Brief introduction of the project.

The **Catalyst** (an Employee Training Management System) is crafted to simplify and automate the training procedures for both interns and employees within the company. This system offers an intuitive platform for overseeing training schedules, monitoring advancements, administering evaluations, and producing customized reports.

1.1.2 Purpose of the project.

The aim of this project is to bolster organizations and enrich their interactions by furnishing them with a comprehensive tool for tracking instruments. The primary objectives of the project include:

- **Enhancing Organizational Experience**: By furnishing a user-friendly and intuitive interface, the platform endeavors to empower administrators in managing their training journey. Users will enjoy convenient access to real-time information concerning their training plans, module dates, and assessment history, fostering a more immersive and personalized experience.

- **Enhancing User Engagement:** Through personalized dashboards, users will gain access to vital information regarding their assessment scores and report history, enabling them to analyze their training performance effectively.

1.1.3 Goals of the project:

- Designing **UI/UX** components for the **Employee Training Management System** to support both user and admin functionalities.
- The specific tasks for the Development Phase will be defined during the **Requirements Analysis Phase**. The goal by the end of the development phase is to deliver a **Minimum Viable Product (MVP)** as scoped during the Requirements phase.
- Designing the Core application tier, which includes:
    - Front End tier for the **ETMS,** accessible through a web browser.

2 Project Scope

The ETMS will tackle these obstacles by offering a centralized platform for creating, distributing, and overseeing training activities. It will streamline processes, enhance visibility, and boost user engagement through personalized assessments and feedback.

2.1.1 Key features and functionalities of the web app.

Intuitive User Authentication: A straightforward authentication procedure allowing users to access their accounts securely. Utilizing secure authentication methods, including email/password verification.

Simplified User Interface: A user-friendly interface design aimed at enhancing ease of use and navigation for all users.

Personalized User Dashboard: Tailored dashboards for each user, showcasing pertinent information such as plan details, module specifics, assessment scores, and personalized reports.

Admin Functionalities: An administrative dashboard enabling oversight of all users. Features include user management, addition of training categories/plans/modules, and the ability to input assessment scores and generate personalized performance reports.

3 Architecture and Technology Stack

3.1.1 Overall architecture of the ETMS

1. Presentation Layer:

Description: The presentation layer is the topmost layer of the architecture, responsible for handling user interactions and displaying information to users.

Components:

- User Interface (UI): Provides interactive dashboards and interfaces for users, administrators, and approvers.
- Client-Side Logic: Manages client-side interactions and communicates with the backend server.

Technologies:

- **React.js** (and libraries like Chakra UI) for frontend development.
- **CSS** (with frameworks like Tailwind) for styling.

2. Application Layer:

Description: The application layer contains the business logic and functionality of the **ETMS** Components:

- Controllers: Handle incoming requests from the presentation layer and orchestrate business operations.

Technologies:

- **Node.js** with **Express.js** for building RESTful APIs.
- **JavaScript** for server-side logic.

3. Domain Layer:

Description: The domain layer encapsulates the core domain concepts and entities of the ETMS.

Components:

- User: Represents user accounts within the system, including roles, permissions, training plans and performances.
- Performance: Represents individual performance and proficiency levels in each topic tracked within the system.
- Module: Represents training modules for employee and interns.
- Training Plan: Represents the training plan with topics, start date and end date.

4. Data Access Layer:

Description: The data access layer manages interactions with the underlying data storage systems.

Components:

- Database Connection Pool: Manages connections to the database to ensure efficient data access.

Technologies:

- MongoDB for storing user data, skill information, certifications, and project details.

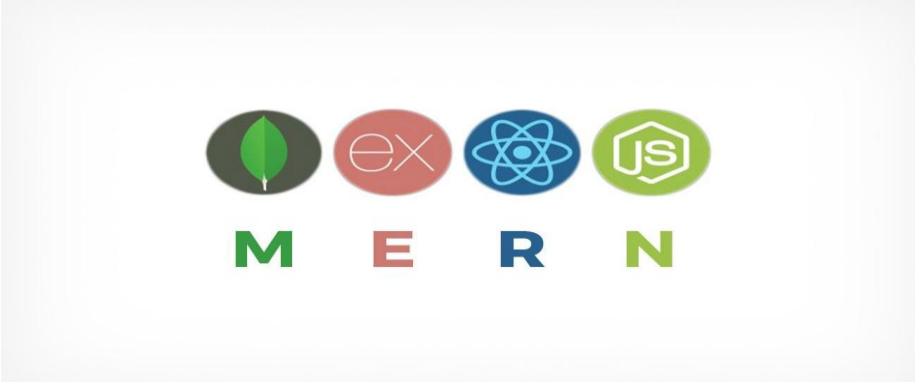
3.1.2 Technology stack (e.g., programming languages, frameworks, libraries).

Frontend:

- Framework: **React.js**
- Styling: **CSS** (Tailwind)
- **API Requests: Axios (for making HTTP requests to the backend and other APIs)**

Backend:

- Framework: **Node.js** with **Express.js**
- Database: **MongoDB**
- Authentication: **JSON Web Tokens** (JWT) for securing API endpoints



Data Engineering:

- **Python**: Used for data injection to Snowflake, data preprocessing, and any custom data manipulation tasks.
- **Snowflake**: Cloud data platform used for storing and managing structured and semi-structured data.
- **Dbt (Data Build Tool)**: Used for data modeling, transforming, and organizing data within Snowflake.

Data Analytics:

- **Power BI**: Business intelligence tool used for creating interactive reports and dashboards.

Data Science:

- **Python** (Pandas, Numpy, Matplotlib, Sklearn)

4 Project Stages:

4.1.1 Front-End (React.js):

Description:

- Develops the user interface components of the **ETMS** using React.js, a popular JavaScript library for building UIs.

Activities:

- Designing and implementing UI components such as forms, tables, and dashboards.
- Managing state and user interactions using React.js features like state management and hooks.
- Integrating with backend APIs to fetch and update data asynchronously.
- Ensuring responsiveness and cross-browser compatibility.

Tools:

- React.js, React Router (for navigation), Axios (for HTTP requests) and various other Js packages like CORS, PapaParser etc.

4.1.2 Back-End (Node.js):

Description:

- Implements the server-side logic and APIs of the ETMS using Node.js, a JavaScript runtime environment.
- Implements the server side logic for using machine learning model in the web application.

Activities:

- Developing RESTful API endpoints to handle requests from the frontend and interact with the database.
- Implementing authentication and authorization mechanisms to secure access to resources.
- Writing middleware functions for request validation, error handling, and logging.
- Integrating with external services and APIs as needed.

Tools:

- Node.js, Express.js (as the web application framework), JWT (for authentication), Mongoose (for database interaction), CORS, Flask (for hosting the ML model)

4.1.3 Database (MongoDB):

Description:

- Manages data storage and retrieval for the ETMS using MongoDB, a flexible and scalable NoSQL database solution known for its JSON-like document storage and high-performance capabilities.

Activities:

- Designing JSON-like document structures to represent entities such as users, modules, performances, and training plan.
- Creating collections, indexes, and validations to organize and enhance data storage efficiency.
- Implementing data integrity and relationships between collections using MongoDB's flexible schema design.

- Performing CRUD (Create, Read, Update, Delete) operations to interact with the database.

Tools:

- MongoDB Atlas

4.1.4 Data Engineering:

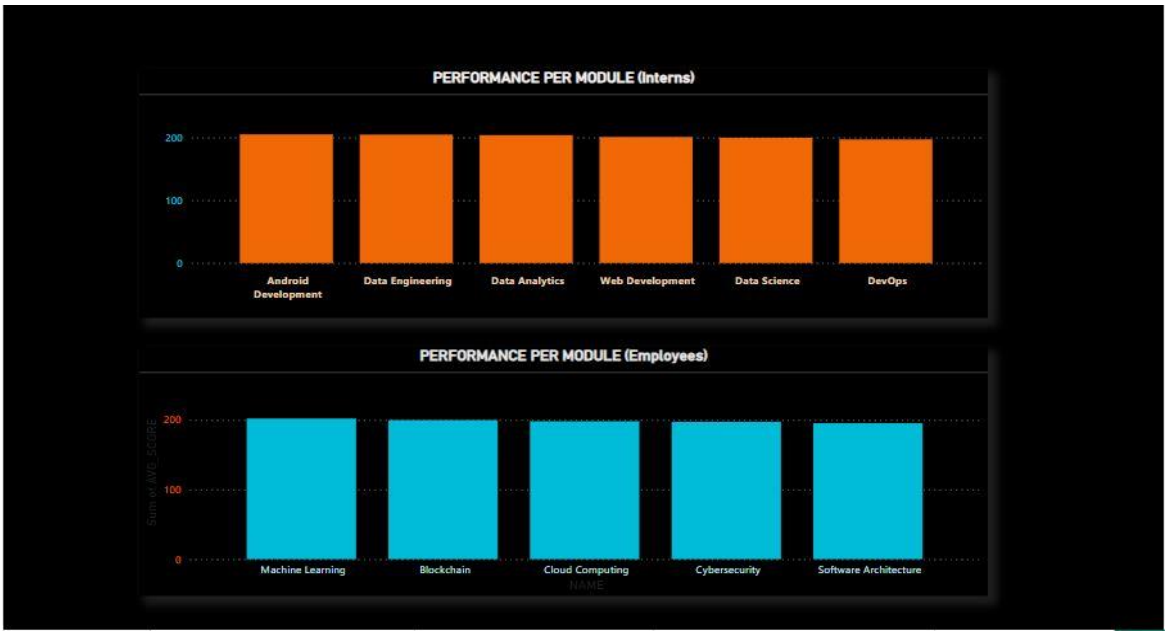
1. **Python Code for Pipeline:**
    - Develop Python code using libraries like Pandas, for transferring the data from MongoDB to Snowflake.
    - Use Snowflake's Python connector to establish a connection and load data into Snowflake tables.
2. **Snowflake Data Warehouse:**
    - Create a Snowflake database named **DATA_SHIP**
3. **DBT (Data Build Tool):**
    - Utilize **DBT** to perform data modeling, transformation, and management tasks for KPIs as follows -
        -
    - Write DBT models for staging and data mart layers to define data transformations and relationships.
    - Execute DBT commands to run models, generate SQL scripts, and deploy changes to Snowflake.

4.1.5 Data Analytics:

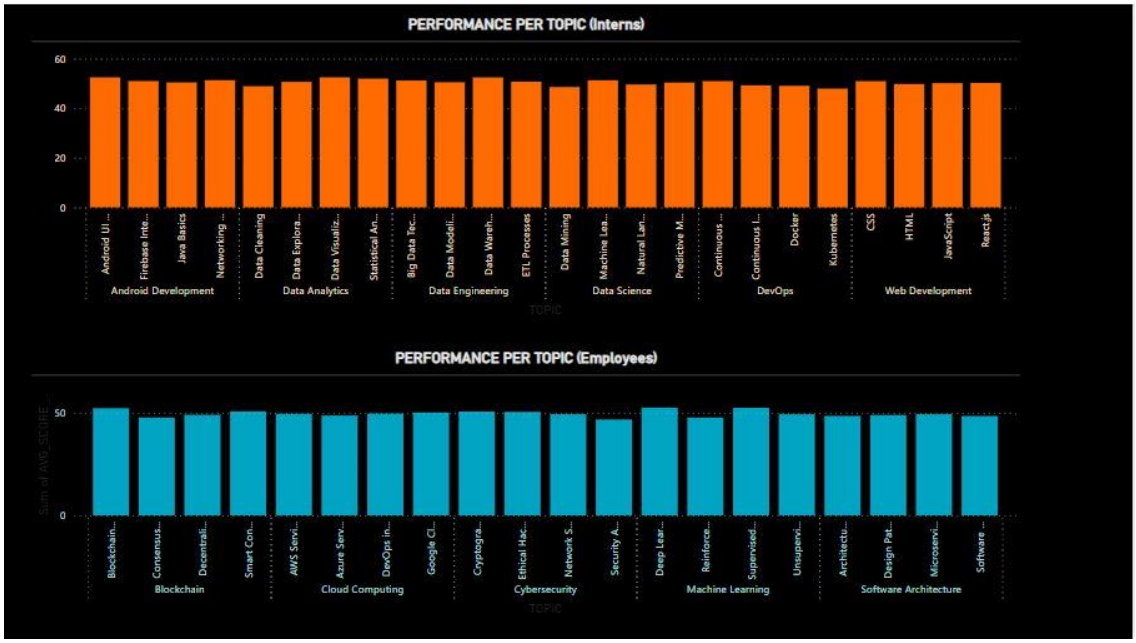
1. **Power BI Activities:**
    - Connect Power BI to the Snowflake warehouse's data mart layer to access cleaned and processed data.
    - Design interactive dashboards using Power BI's drag-and-drop interface to visualize insights derived from the data.
    - Incorporate various visualization types like charts, graphs, tables, and maps to represent different aspects of the data.

2. **Power BI Dashboard:**
    - The Power BI Dashboard is based up of various key performance indicators.
    - The activity included connecting to snowflake via Power BI and importing all the cleaned tables to create various visualisations.

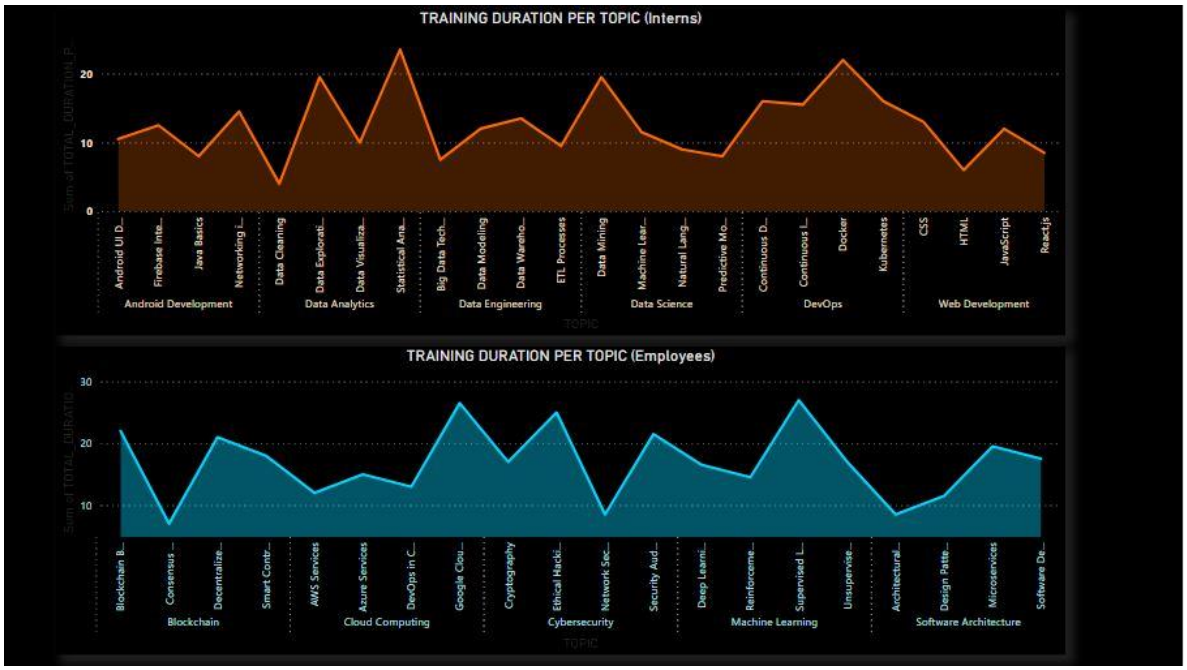
KPI 1 - Overall performance per module



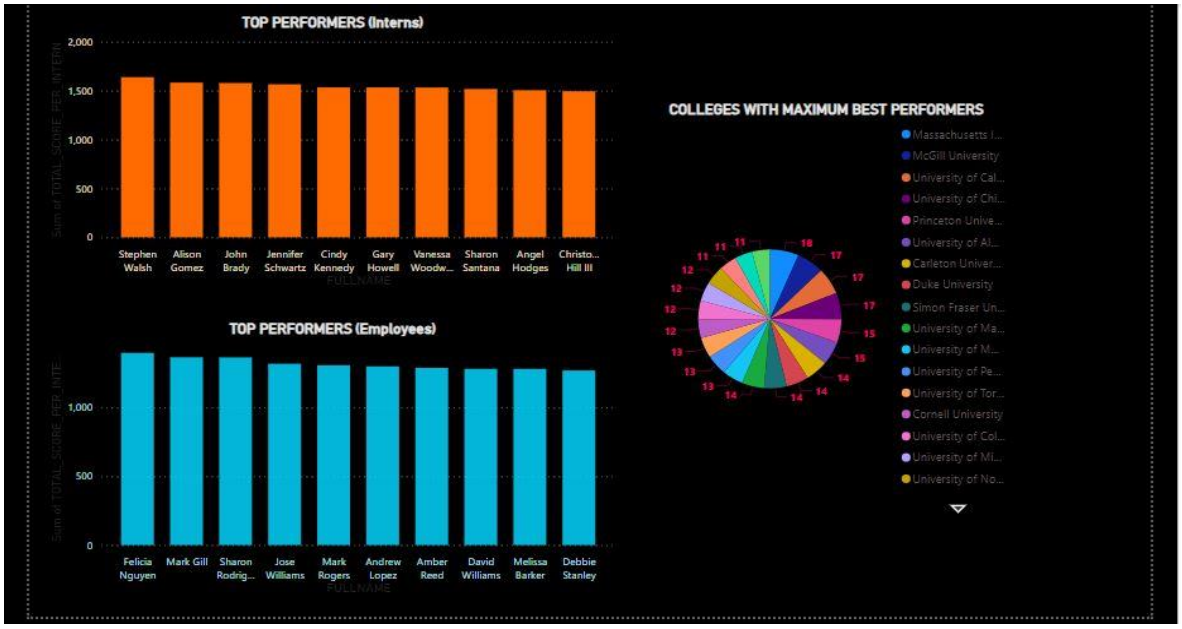
KPI 2 - Overall performance per topic within a module



KPI 3 - Overall performance per topic within a module



KPI 4 & 5 - Top performers & Maximum performers per college



4.1.6 Data Science:

1. **Python Code for Data Science:**

    - Utilised Python packages like **Pandas, Numpy and Matplotlib** for data pre-processing and EDA.
    - Used **SKlearn** for machine learning model implementation.
    - Used **joblib** package to make a pickle file for the ML model, load and use it in **flask API.**

5 APP Guide

5.1.1 User Dashboard:

Overview: The User Dashboard is tailored for individual users (admin, intern and employee) within the organization to manage their skills, certifications, and project contributions.

Features:

- **Profile:** View and manage your personal details such as name, email, phone number, linked profile and other personal details.
- **Scheduler:** Access calendar and schedule training plans for interns and employees as an admin. View the pre-set schedules and plan as an intern or employee.
- **Create functionalities for admin:** Add users, upload performance and set new modules for the intern or the employees.
- **SmartSession for admin:** ML driven predictions as to how long a session might last, for admins to set trainings accurately.
- **Track Training functionalities for intern/employee:** Detailed report about the training for interns/employees,
- **Strong authentication:** Robust login and logout systems for secure entry and exit from the web app.
- **Intelligent Authorization:** Intelligently devised authorization logic.
- **Password Reset :** Reset password securely in case n user forgets or needs to update it for account access.

5.1.2 Admin Dashboard:

Overview: The Admin Dashboard provides administrators with comprehensive control and management capabilities over the Skill Matrix System.

Features:

- **User:** Manages user details including name, email, role, and department within the ETMS.
- **Module:** Represents a distinct training unit or topic within the system's training plans.
- **Add Module Trainer:** Enables the assignment of a trainer or instructor to a specific training module.
- **Add User Module:** Associates a user with a training module, allowing them access to relevant training materials.

- **Performance:** Tracks and evaluates user performance based on completed modules and assessments.
- **Add Quiz:** Allows creation and addition of quizzes to training modules for assessing user knowledge.
- **Add Module Quiz:** Links quizzes specifically to training modules for targeted assessment.
- **Module Status:** Displays the progress and completion status of each module for users within the system.

6 Workflow:

- **Login Flow**: User logs in using credentials.
- **Profile Management**: User updates profile information.
- **Training Access**: User navigates training plans and modules.
- **Progress Tracking**: User monitors training progress.

6.1.1 Data Flow

User data is retrieved from the database upon login.
Training plans and module details are dynamically fetched based on user roles.
Assessment results are stored and updated in the database for reporting purposes.

7 Data Model

7.1 Data Model Description:

- **Admin:**

  This collection stores the admin credentials and personal information such as phone number, Linked In profile url, academic qualifications.

- **Users:**

  This collection stores the intern and employee credentials and personal information such as phone number, Linked In profile url, academic qualifications.

- **Modules:**

  This collection stores the modules and the corresponding details such as topics, start date and end date for both interns and employee.

- **Intern training plan:**

  This collection stores the training plans and the corresponding details such as the trainer name, time of training for only the interns.

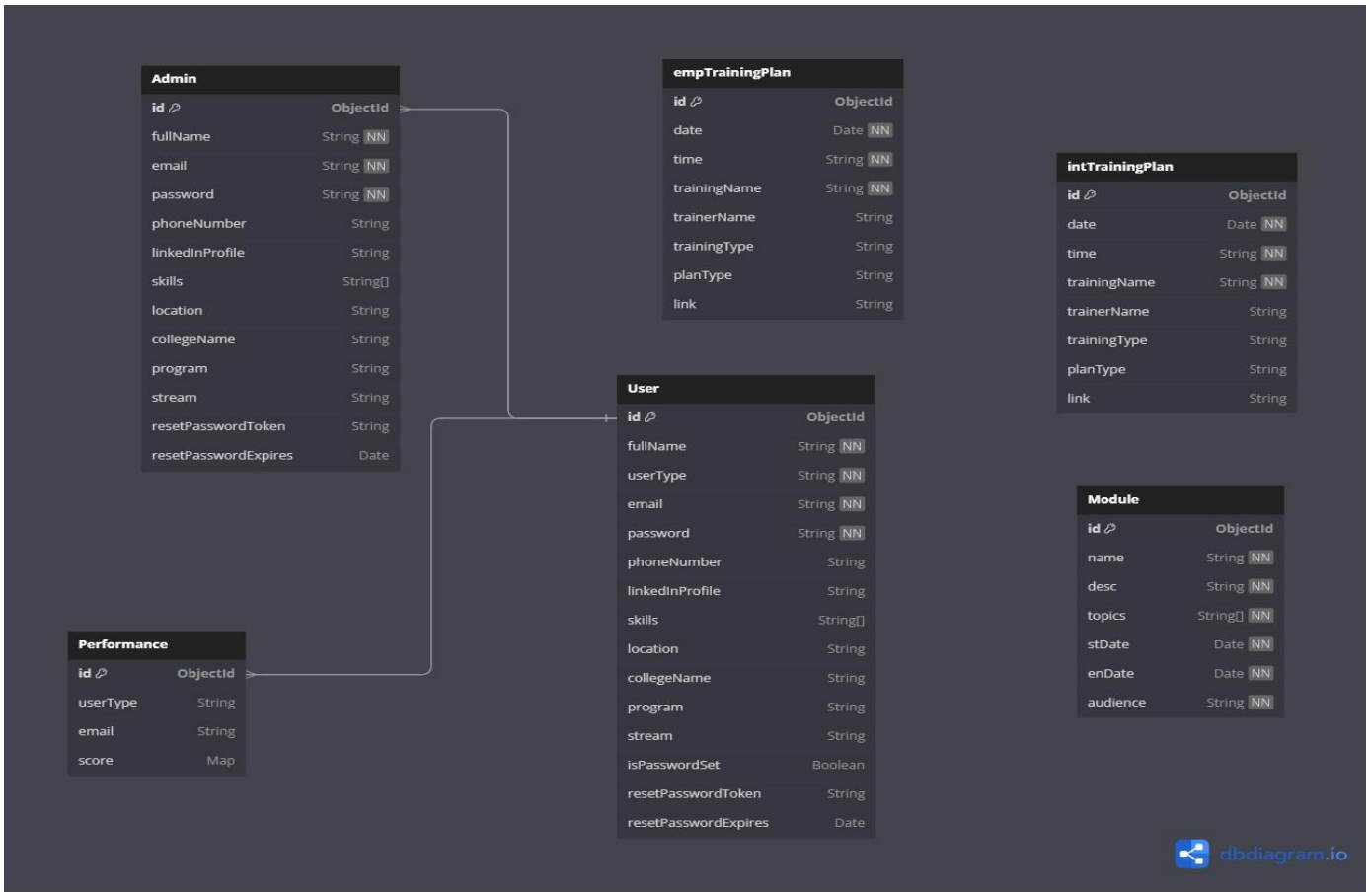
- **Employee training plan:**

  This collection stores the training plans and the corresponding details such as the trainer name, time of training for only the employees.

- **Performance:**

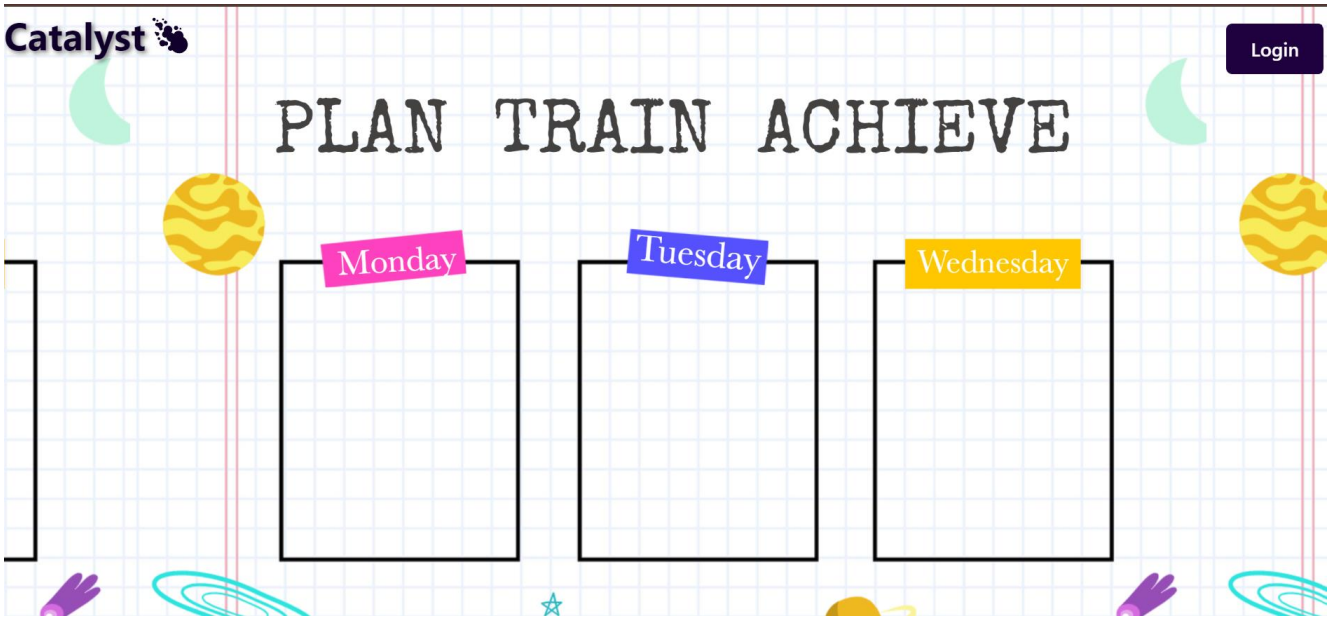
  This collection stores the performances in every topic for both interns and employees; references the user collection.

7.2 Entity-Relationship Diagram (ERD):

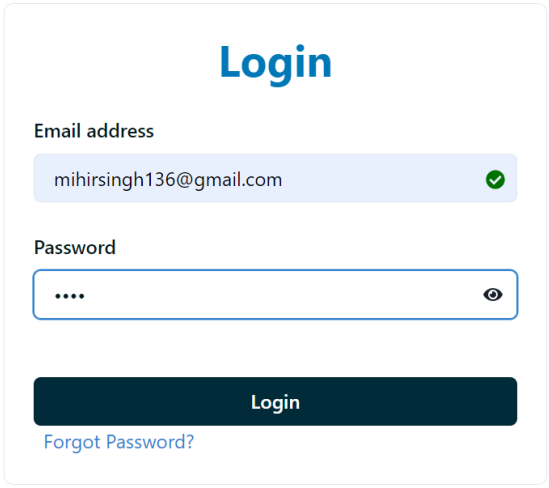


8 User Interface Design

8.1 Landing page

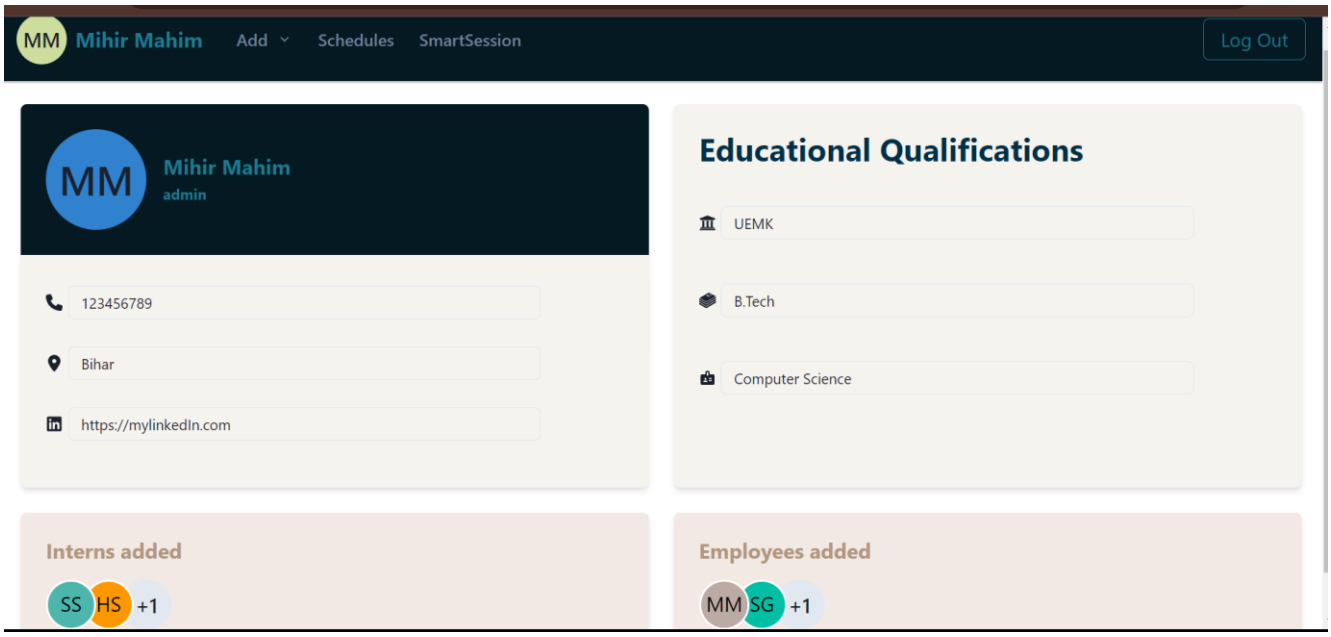


8.2 Login Page

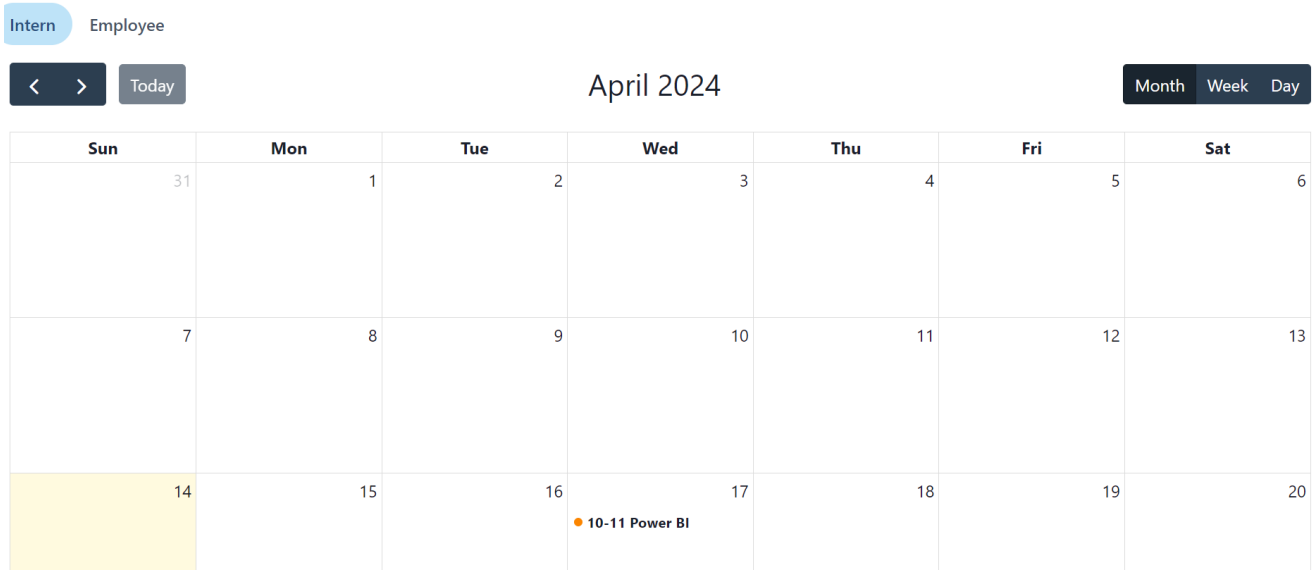


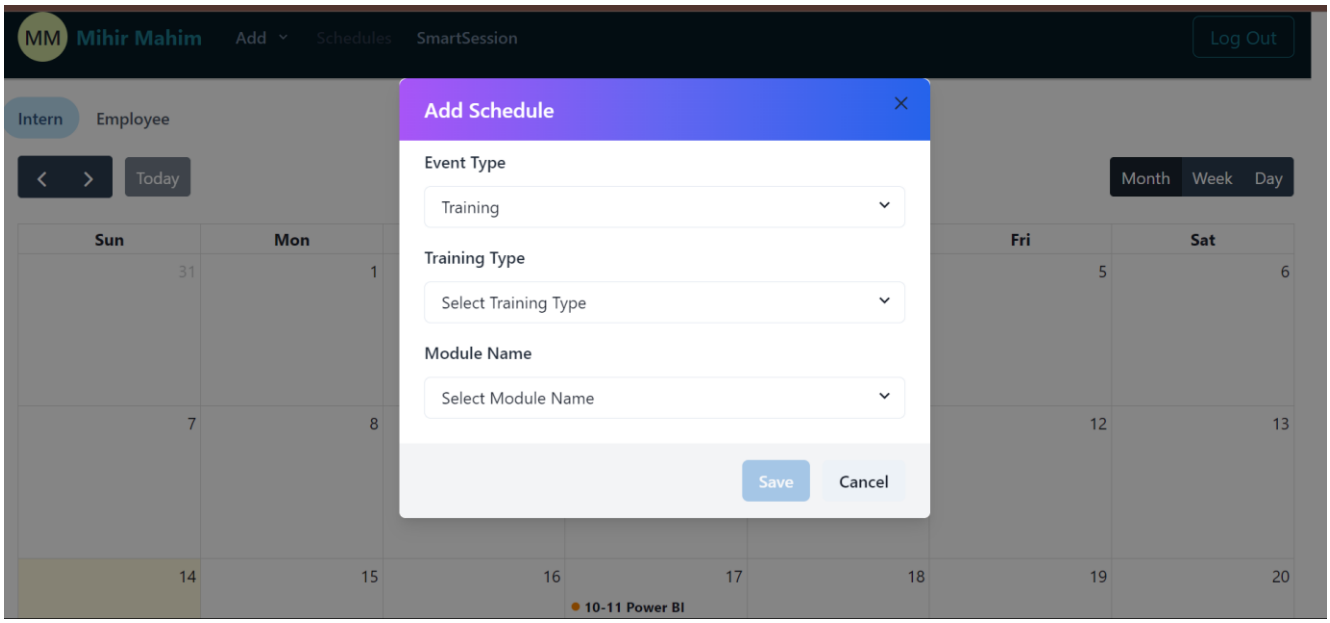
8.3 Admin Interface

8.3.1 Admin dashboard

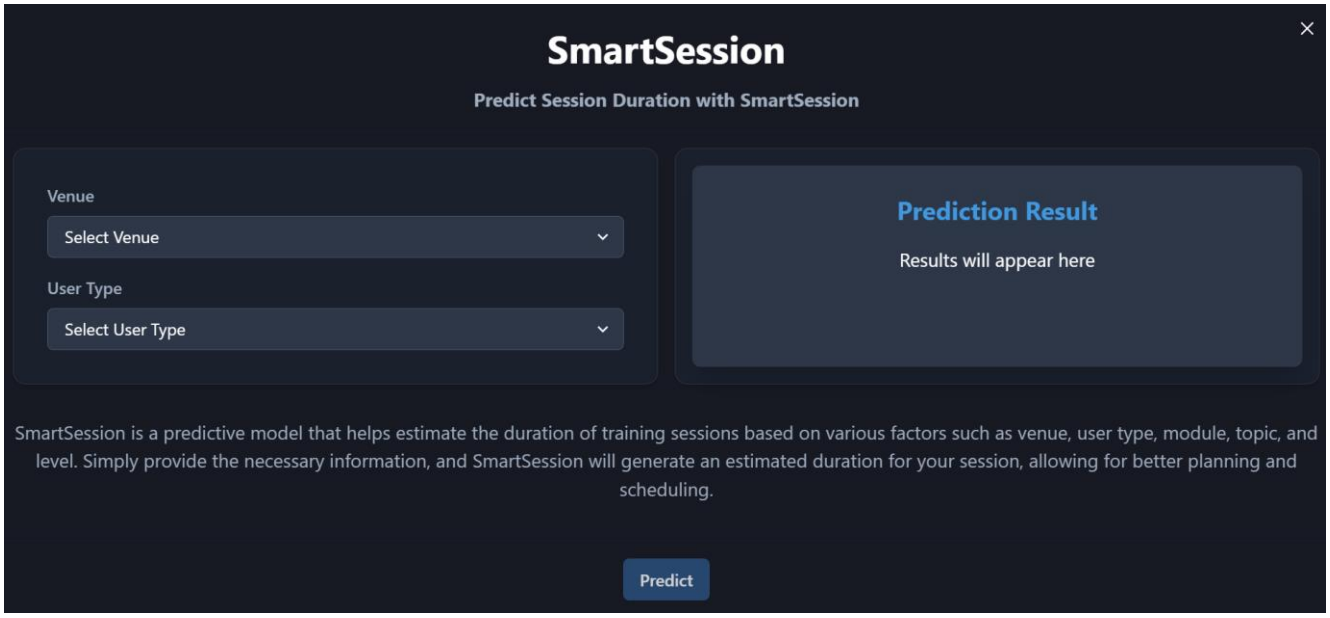


8.3.2 Scheduler

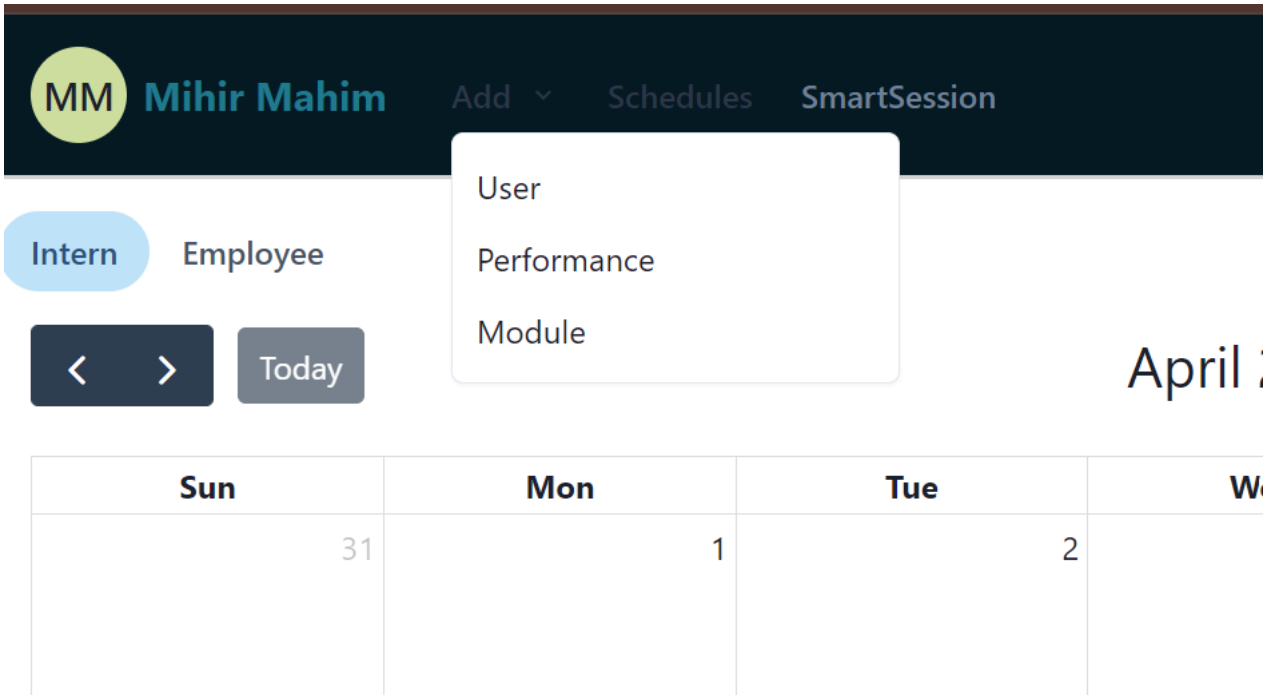




8.3.3 SmartSession

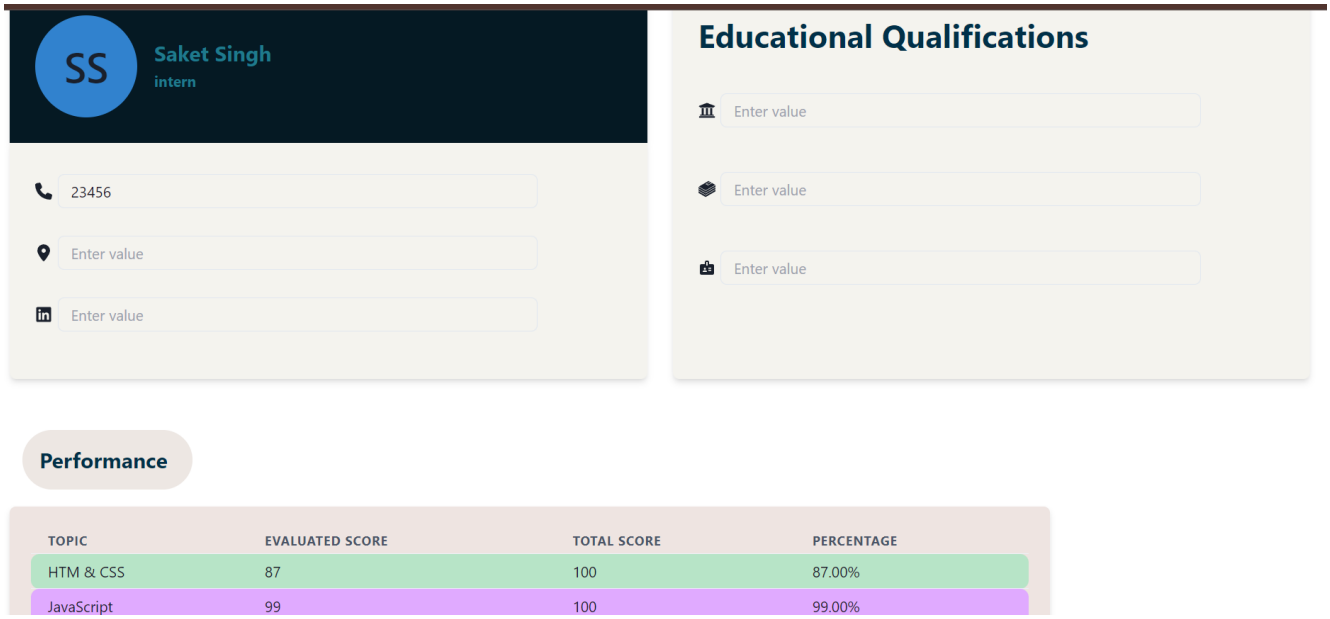


8.3.4 Create functionalities

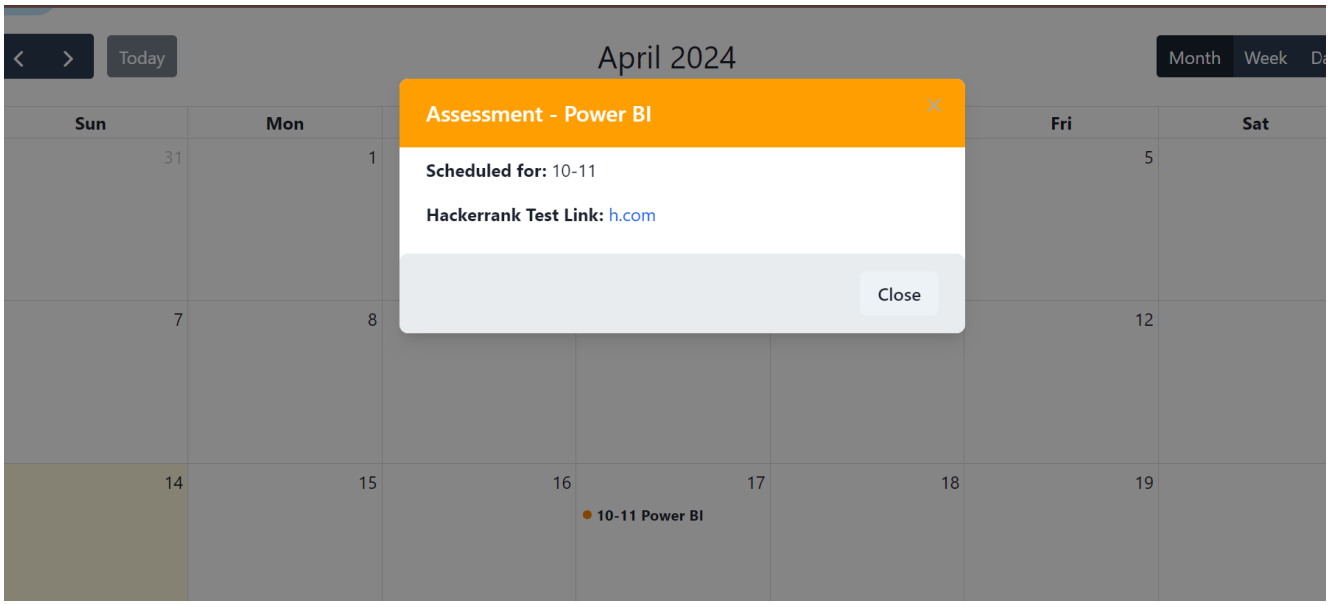


8.4 User Interface

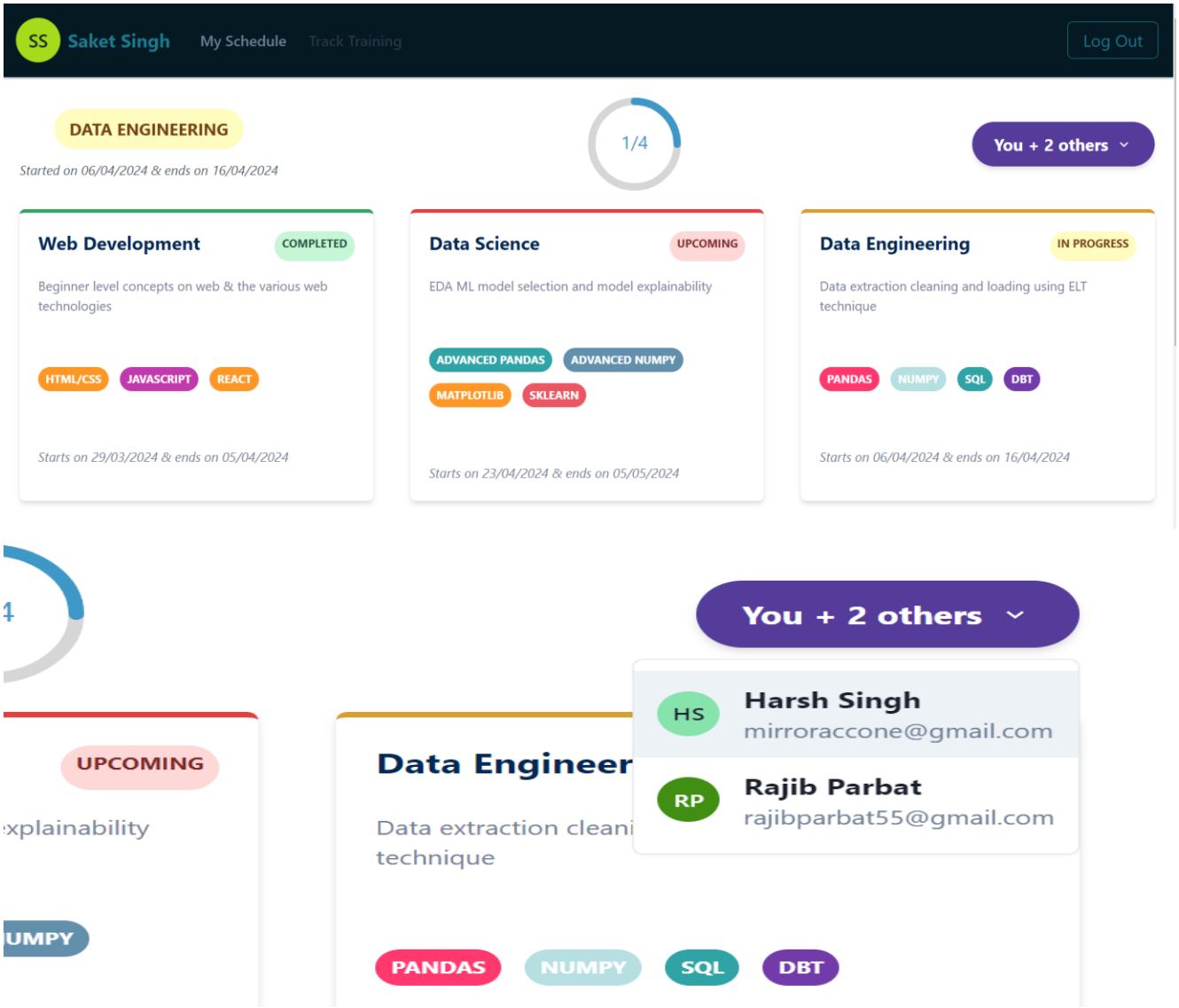
8.4.1 User Dashboard



8.4.2 My schedules



8.4.3 Track Training



9 Conclusion

The Catalyst Employee Training Management System (Catalyst ETMS) signifies a significant shift in our organization's approach to training and development. By consolidating training resources, automating processes, and offering personalized learning experiences, Catalyst greatly enhances the efficiency and effectiveness of our training endeavors.

With Catalyst in place, we've achieved several key objectives:

- Streamlining training administration, reducing manual effort, and administrative burdens. This empowers trainers and administrators to focus on delivering high-quality content rather than getting bogged down in logistical tasks.

- Providing users with a user-friendly interface for seamless access to training materials, progress tracking, and personalized learning paths. This fosters engagement and encourages proactive professional development.

- Offering robust performance monitoring and assessment capabilities, providing insights into user progress and proficiency levels. This data-driven approach facilitates the identification of improvement areas, addressing individual learning needs, and recognizing top performers.

Looking ahead, Catalyst will evolve to meet our organization's changing needs. Future enhancements may include advanced reporting and analytics for deeper insights into training effectiveness, integration with other systems to streamline data management, and additional interactive learning tools to enhance user engagement.

In essence, Catalyst represents a strategic investment in our organization's human capital. By empowering employees with the tools they need to succeed, we foster a culture of continuous learning and growth that drives success and innovation across all departments. Catalyst isn't just a system—it's a catalyst for individual and organizational excellence.