JMAN GROUP

# Employee Learning Platform

Major Project

# Contents

# 1 Project Overview

## 1.1.1 Brief introduction of the project.

This project aims to create a learning platform tailored for employees, incorporating crucial features such as secure login, event scheduling, user enrolment, email alerts, and capacity tracking. Additionally, I was focused on building a robust data infrastructure to support comprehensive reporting and analytics. Moreover, this project stands out by integrating predictive capabilities, leveraging user skills and experience in data science to recommend relevant events. This forward-thinking approach ensures a personalized learning journey for each employee, ultimately enhancing organizational effectiveness and employee satisfaction.

## 1.1.2 Purpose of the project.

The aim of this project is to develop a holistic learning platform tailored for employees within a company.

Key objectives include:

- Offering a centralized hub for employees to access diverse learning opportunities, such as training events and workshops, to enrich their skills and knowledge pertinent to their roles and professional growth.
- Implementing automated email notifications to ensure prompt communication with employees regarding new events, registration confirmations, and any updates or adjustments to scheduled activities, thereby encouraging enhanced engagement and participation.
- Establishing a data platform allows for the collection, cleaning, transformation, and modelling of relevant data, enabling the generation of operational reports and analysis to support decision-making processes aimed at governing and improving business operations related to employee development activities.
- Providing recommendation of events based on the user skills and experience within the organization ensures personalized learning experience

## 1.1.3 Goals of the project:

- **Development of a Learning Platform:** To Create a user-friendly platform where employees can easily access training resources and register for events. This includes implementing a secure login page, event creation functionality, user registration, and email notifications.
- **Data Platform Setup:** To establish a robust data platform capable of connecting to source systems and supporting the reporting needs of the Learning Platform. This involves extracting data from various sources, cleaning and transforming it, designing and building a data model.
- **Automation:** Automate processes wherever possible to reduce or eliminate manual interventions, ensuring efficiency and reliability in data handling and reporting.
- **Predictive Analytics:** Utilize employee skills and experience data to predict the type of training best suited for each individual, enhancing the personalization and effectiveness of the learning experience.

# 2 Project Scope

## 2.1.1 Web app support

| Platforms | Version |
|---|---|
| Google Chrome | 97 and above |
| Microsoft Edge | 97 and above |

### 2.1.2 Key features and functionalities of the web app.

**User Creation and Authentication:**

The Application provides a key feature only to admin and that is user creation. The Admin creates the user and the user can login using the default password that the admin created and change the password once entering the application.

**Admin Dashboard**:

The admin can create Events which will be displayed to the users for registrations. The admin can also view the users who are registered for particular event and can also edit the details of the particular event.

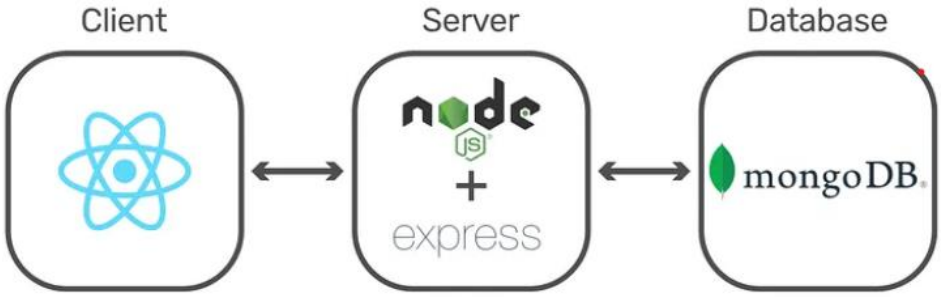
**Employee Dashboard**:

The user can register for the events that the admin created and also the user can add the events to Wishlist which can be registered later

# 3 Architecture and Technology Stack

### 3.1.1 Overall architecture of the web application.

The above architecture describes that the application is built using the MERN stack. The frontend comprises of ReactJS and CSS.
The backend is connected using Node and Express and used the MongoDB as database to store the details of the  User, Event, Event Registrations, Wishlist etc..




.

**Key functionalities:**

**Front end Application:**

- User Authentication: Facilitates secure user registration and login processes, ensuring data integrity and confidentiality. Administrators can create employee accounts securely, employing username/password authentication methods.
- Admin Dashboard: Empowers administrators with a centralized dashboard to oversee event management tasks. This includes functionalities such as creating events, updating event details, managing user registrations, and editing event information through intuitive interfaces.
- Employee Dashboard: Offers employees personalized dashboards showcasing relevant information like registered events, upcoming events, and event wishlist management, enhancing user engagement and interaction.
- Responsive Design: Ensures seamless user experience across diverse devices and screen sizes, encompassing desktops, tablets, and smartphones, through a responsive user interface.

**Back end Services:**

- Authentication Service: Manages authentication and authorization processes for both administrators and employees, verifying user credentials and ensuring secure access to platform functionalities.

- User Management Service: Enables efficient management of user accounts, encompassing user registration and role-based access control to delineate administrative privileges and employee access levels.
- Event Management Service: Facilitates comprehensive event management, including event creation, modification, deletion, and registration oversight, streamlining administrative tasks and enhancing user convenience.
- Notification Service: Delivers real-time updates and notifications to users concerning event registrations, updates, and relevant announcements, fostering enhanced communication and engagement.

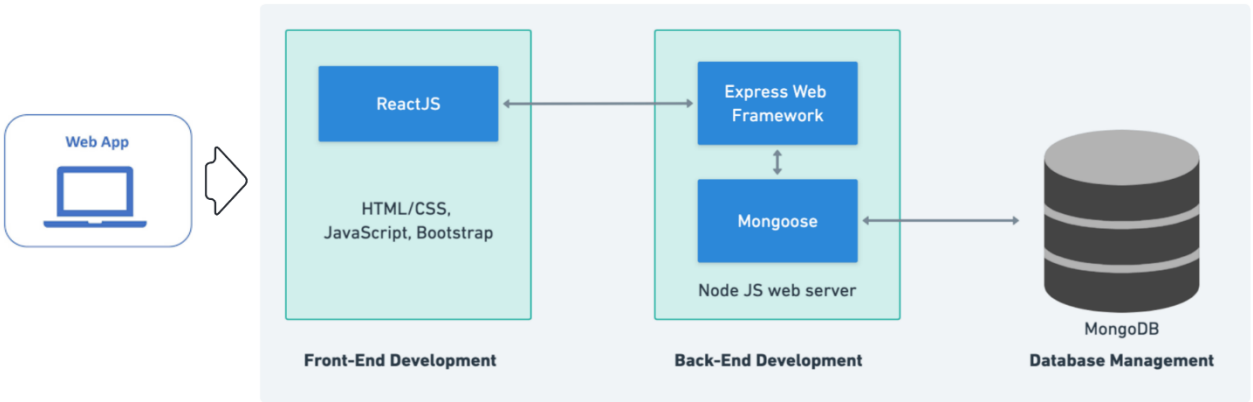
**Data Engineering:**

- Data Extraction: Fake data is generated from open source like mockaroo which is then used for cleaning process
- Data Cleaning and Transformation: Cleanses and preprocesses extracted data to ensure data quality and consistency, transforming it into a standardized format suitable for subsequent analysis.
- Reporting Tables: The reporting tables are created within the snowflake. It visualizes the transformations done using dbt.

**Data Science:**

- Feature Engineering: Extracts key user attributes, event characteristics, and historical attendance data to inform the recommendation model.
- Model Training: Utilizes the dataset to train machine learning models for generating personalized training predictions, fostering continuous skill development among employees.
- Prediction: The prediction of event name based on the user's skills and experience is done using machine learning model. For more accurate prediction I have also predicted using Collaborative filtering

### 3.1.1 Architecture




The Architecture diagram of this application explains that the technology stack used to develop the application. ReactJS is used as the front-end library to build the user interface. The backend is built using Node.js and Express.js. Express.js is a web application framework for Node.js that simplifies the process of building web applications. The back-end handles HTTP requests, processes data, and sends a response to the front-end.

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

**Technology Stack for the Web Application Platform:**

The technology stack for the web application comprises the following Technology stack

**React JS**:

React JS is an open-source JavaScript library for building user interfaces. It is widely used for creating interactive and dynamic web applications with a focus on component-based architecture.

**Axios:**

Axios is a JavaScript library for making HTTP requests in both the browser and Node.js environments. With a simple and intuitive API, Axios simplifies the process of sending and receiving data between the client and server. It supports asynchronous operations and works seamlessly with JSON data.

**Node.js:**

Node.js is a versatile and powerful platform for building modern web applications, APIs, and micro-services, offering high performance, scalability, and a vibrant ecosystem of tools and libraries. Its combination of JavaScript on both the client and server-side streamlines development and enables full-stack JavaScript development.

**MongoDB:**

MongoDB is a versatile and scalable database solution that is well-suited for a wide range of use cases, including web applications, mobile apps, content management systems, and real-time analytics. Its flexible data model, scalability, and rich feature set make it a popular choice for modern application development.

### 3.1.3 Rationale behind the chosen technology stack.

In developing the Web Application Platform for Employee Learning Platform, we have chosen React JS as the primary technology stack. The rationale behind this selection is based on the following considerations:

**Cross-Platform Compatibility:**

While React.js itself is primarily focused on building web applications, the React ecosystem provides various tools, frameworks, and libraries that enable developers to achieve cross-platform compatibility for their applications, extending the reach and usability of React applications across different platforms and devices.

**Time and Cost Efficiency:**

By utilizing React JS, we can optimize development resources and streamline the development process. This efficiency enables faster time-to-market, ensuring the application reaches the organisation sooner.

**Large Developer Community and Ecosystem:**

React JS 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:**

React JS component-based architecture promotes code reusability. By developing reusable components, we can efficiently manage the application's UI elements and logic across different screens. This approach not only simplifies development but also facilitates maintenance and future updates, as changes made to shared components propagate throughout the application.

In developing the Data Engineering Pipeline for Employee Learning Platform, we have chosen Snowflake and Data Build Tool (dbt) as the primary technology stack. The rationale behind this selection is based on the following considerations:

**Scalability:**

Snowflake is a cloud-based data warehouse that offers elastic scalability. It can handle large volumes of data and concurrent queries, making it suitable for growing businesses and handling fluctuating workloads without performance degradation.

**Security and Compliance:**

Snowflake provides robust security features, including role-based access control (RBAC), encryption at rest and in transit, and compliance certifications such as SOC 2 and HIPAA. This ensures data security and compliance with regulatory requirements.

**dbt for Data Transformation:**

dbt is a powerful data transformation tool that allows data engineers and analysts to build, test, and deploy data transformation pipelines using SQL. It promotes a modular and version-controlled approach to data transformation, making it easy to manage and collaborate on data workflows.

**Community Support:**

Both Snowflake and dbt have active and supportive communities of users and contributors. This provides access to resources, best practices, and community-developed packages and plugins that can enhance the functionality and usability of the platforms.

In developing the Machine Learning Model for Employee Learning Platform, we have chosen python as the primary technology stack. The rationale behind this selection is based on the following considerations:

**Rich Ecosystem of Libraries:**

Python boasts a vast and robust ecosystem of libraries and frameworks specifically tailored for machine learning and data science, such as TensorFlow, PyTorch, scikit-learn, pandas, NumPy, and matplotlib. These libraries provide powerful tools for data manipulation, modelling, visualization, and evaluation, allowing developers to efficiently implement and experiment with ML algorithms.

Based on these considerations, we have chosen the above technology stack as our primary technology stack.

# 4 Web App Components

## 4.1.1 Main Components of the web app.

- Login
- AdminDashboard
- UserCreation
- EventCreation
- DisplayEventsAdmin
- UserEventRegistration
- DisplayRegisteredEvents
- ForgotPassword

## 4.1.2 Purpose of each component.

### *4.1.2.1 Login*

The login feature grants system access to administrators and regular users alike. Administrators are guided to the Admin Dashboard, equipped with options to oversee user and event management. Regular users are seamlessly directed to the User Event Registration page, facilitating effortless event participation.

### *4.1.2.2 AdminDashboard*

The Admin Dashboard is like the control center for admins. It's where they can easily do important tasks. They have three main options: making new users, creating events, and managing events by showing, editing, or deleting them.

#### 4.1.2.3 *UserCreation*

In the Admin Dashboard, admins can add new users quickly using the User Creation feature. They just need to enter basic info like name, email, and role. This helps make sure each user is correctly identified and gets the right access. Admins can also set permissions for each user, deciding what they can and can't do in the system.

#### 4.1.2.4 *EventCreation*

In the Admin Dashboard, admins have essential features to manage the system efficiently. The User Creation tool enables admins to easily add new users, inputting basic details like name, email, and assigning roles for specific access permissions. Additionally, the Event Creation feature empowers admins to organize events seamlessly by entering important details such as event name, location, date, and time. This ensures precise scheduling and coordination for all events within the system.

#### 4.1.2.5 *DisplayEventsAdmin*

The DisplayEventAdmin component showcases created events for administrators, offering a streamlined view of all scheduled activities within the system. Admins can easily access and review event details for effective management and coordination.

#### 4.1.2.6 *UserEventRegistration*

Event Creation enables users to create events, fostering registration and attendance. This feature streamlines event management and participation, enhancing user engagement and facilitating seamless coordination within the system.

#### 4.1.2.8 *DisplayRegisteredEvents*

The DisplayEvents feature showcases registered events to individual users, providing them with a personalized view of upcoming activities. This facilitates easy access to relevant event information and enhances user engagement within the platform.

#### 4.1.2.9 *ForgotPassword*

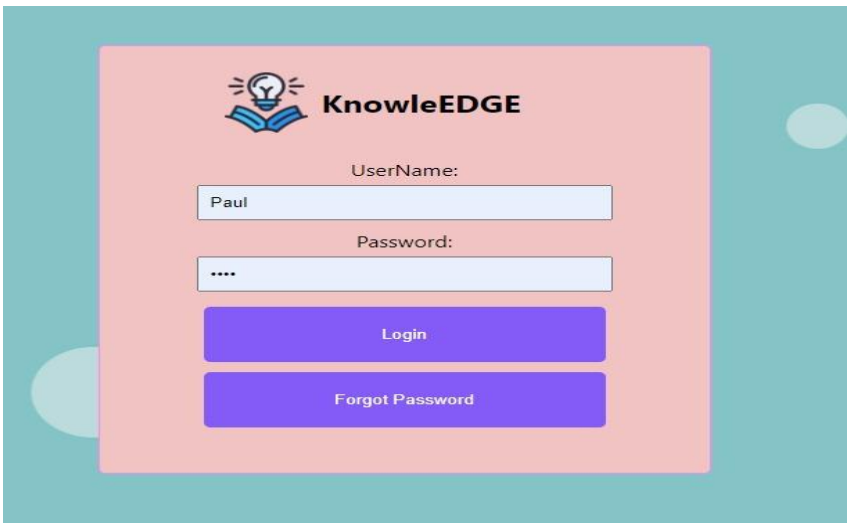
The feature provides employees and admins with a streamlined and secure method to regain access to their accounts if they forget their passwords.

# 5 User Interface Design

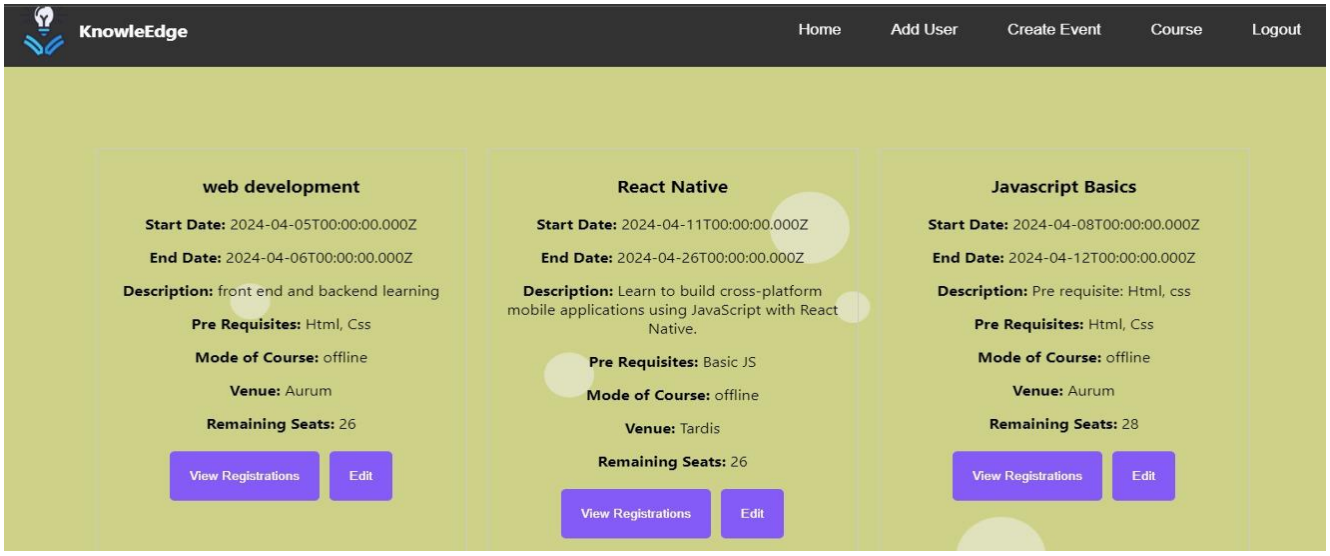
User Interface (UI) design in this project encompasses user research and analysis to understand user needs, wireframing and prototyping for layout visualization, visual design for aesthetic appeal and branding alignment, responsive design for cross-device compatibility, accessibility considerations for inclusivity, usability testing for feedback-driven refinement, and seamless integration with front-end development for accurate implementation of design elements.

## 5.1.1 User interface (UI)

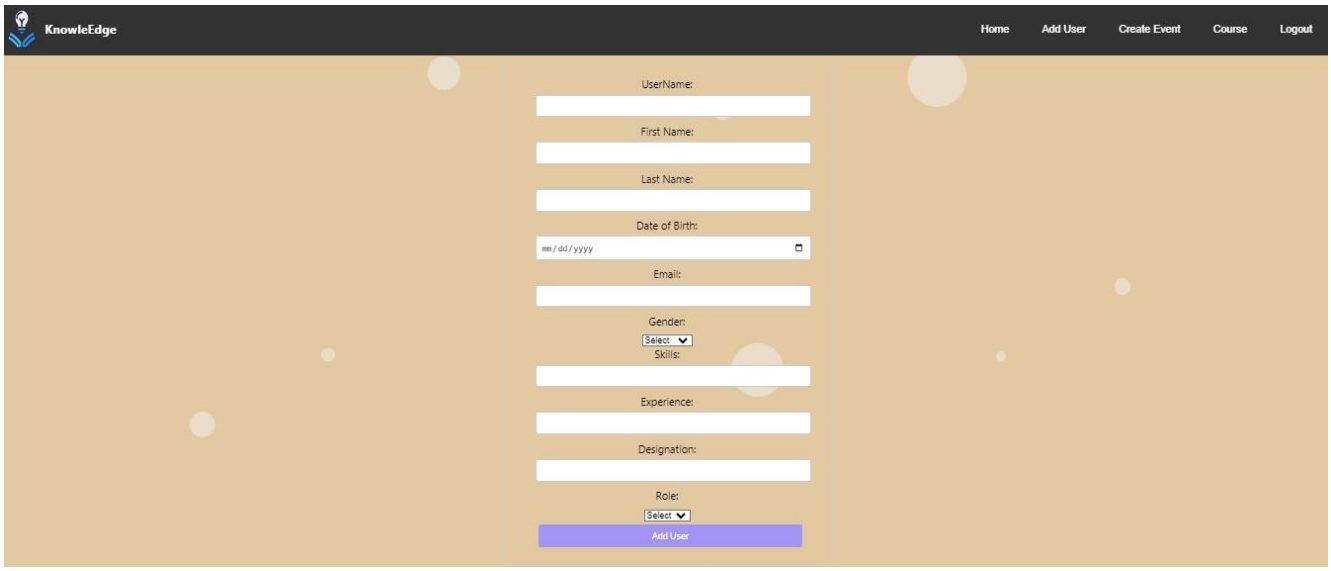
**Login Page**



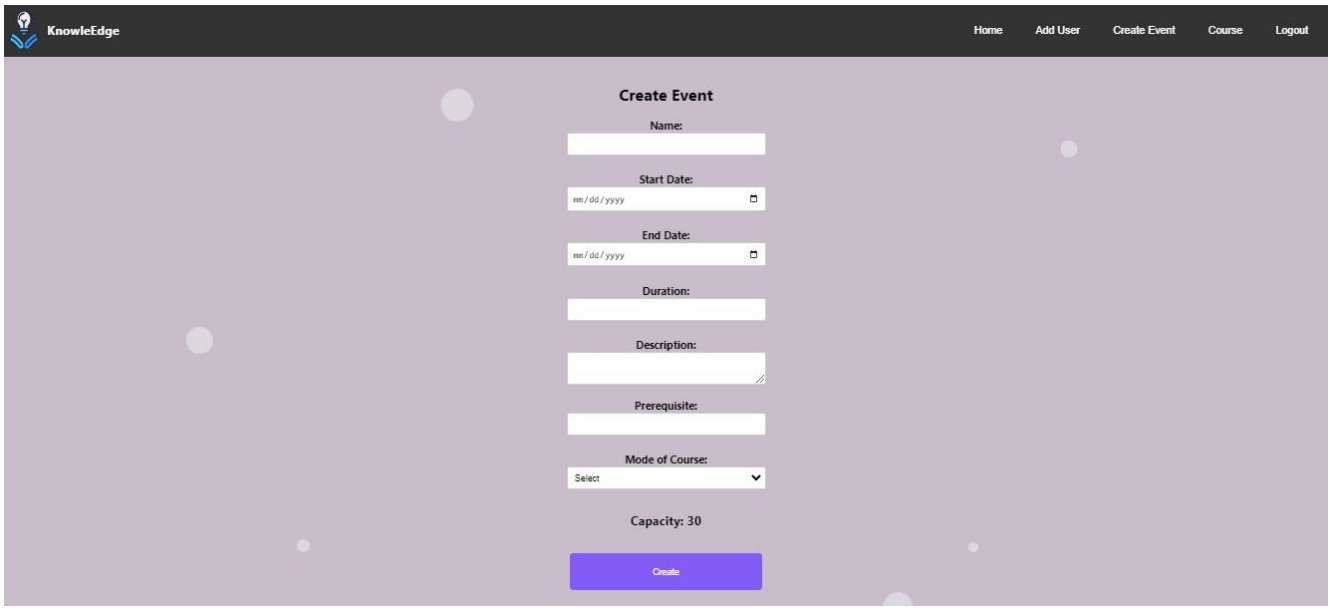
## Admin Dashboard



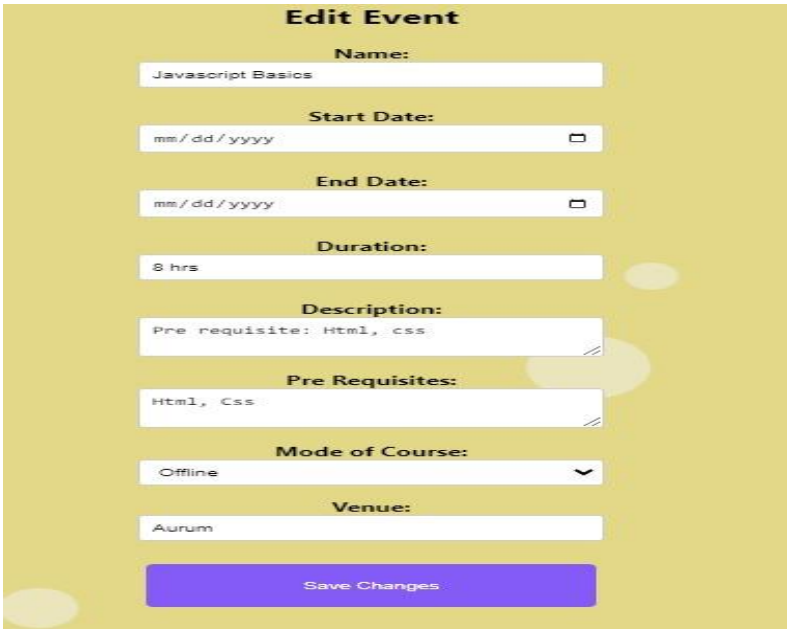

## User Creation



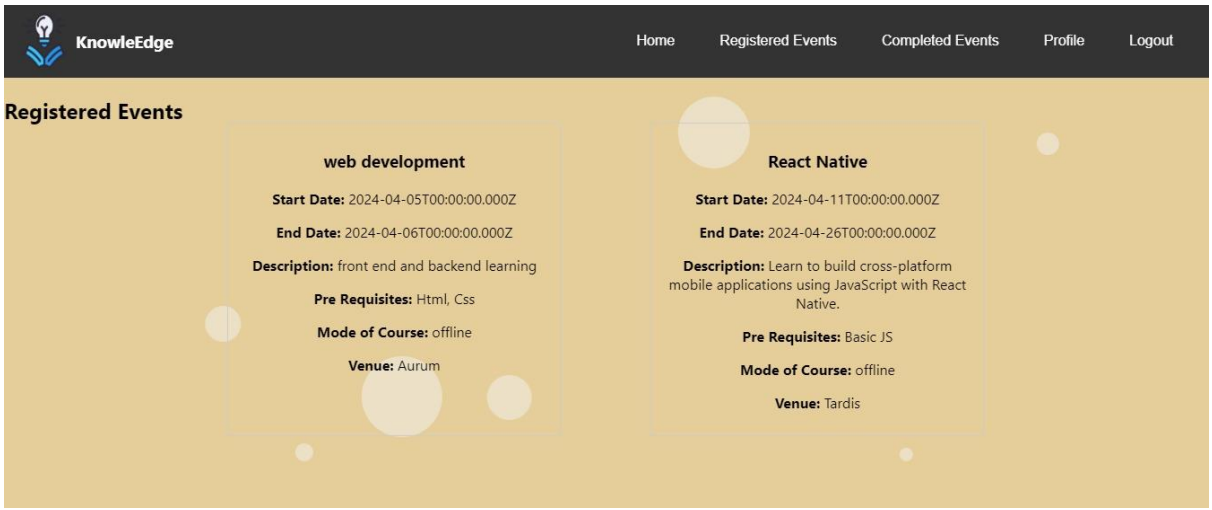

## Event Creation



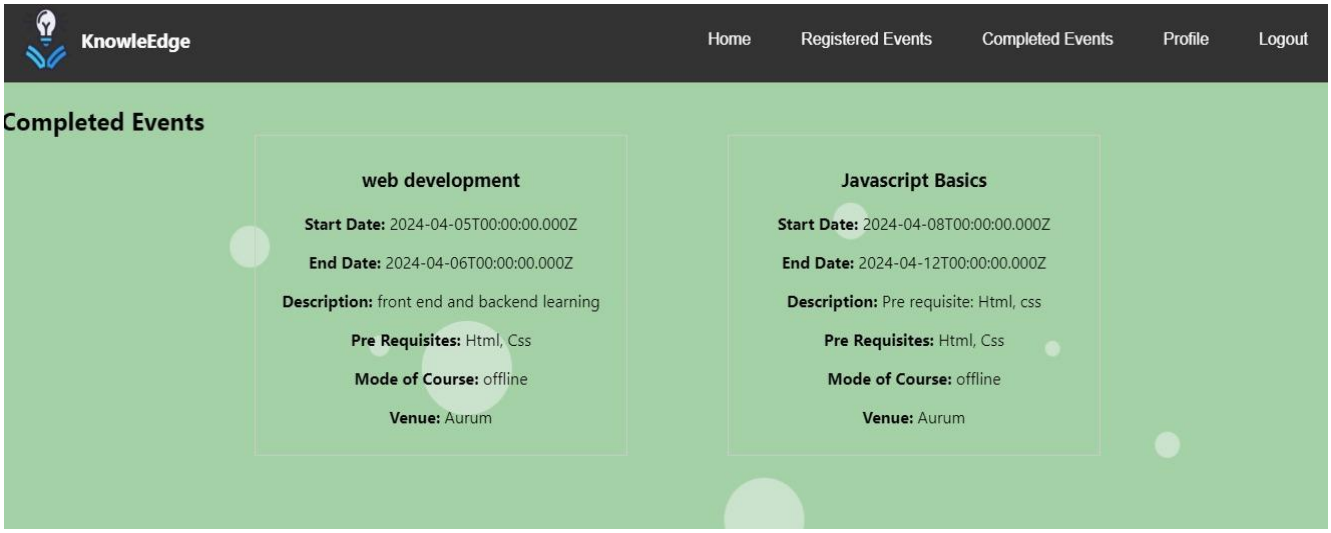
## Edit Event



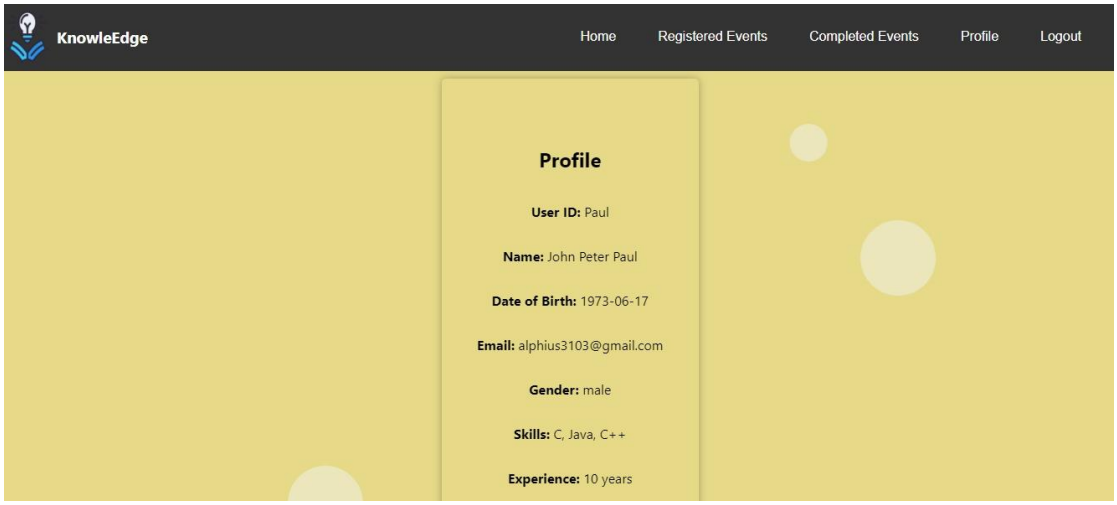

## Display Registered Events to the Users




## Display Completed Events to the Users



**Profile of the user**




# 6 Work with Data

Testing and quality assurance are essential processes in software development that aim to ensure the reliability, functionality, and overall quality of a software product minimizing risks, and ultimately delivering a stable and dependable software product to the end users.

## 6.1.1 Approach for the Data Engineering.

**Generating Fake Data using Mockaroo**

In this phase, synthetic data is created to simulate real-world scenarios using Mockaroo. The generated dataset encompasses various user profiles, event characteristics, and relevant information.

**Uploading Fake Data into Snowflake**

The next step involves uploading the generated fake data into Snowflake, a cloud-based data warehousing platform renowned for its scalability and performance

**Data Transformation with dbt**

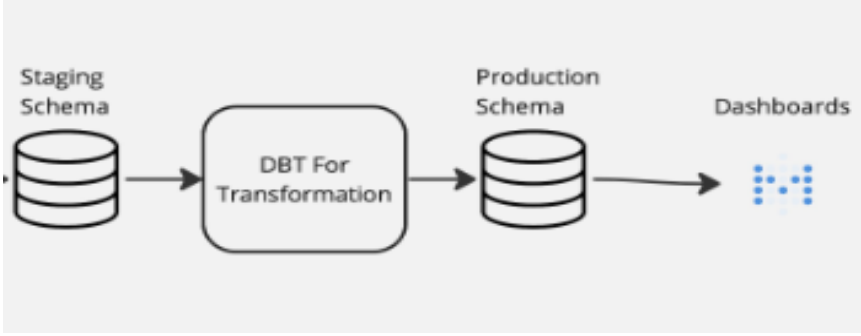
Once the simulated or fake data is securely stored within Snowflake, a cloud-based data warehousing platform known for its scalability and security features, it undergoes further processing using dbt (data build tool).Dbt is an open-source tool specifically designed for data transformation and analytics engineering purposes

**Applying Transformations**

Within dbt, various transformations are applied to the fake data stored in Snowflake. These transformations include data cleaning, aggregation, filtering, and other operations aimed at preparing the data for analysis and generating actionable insights.

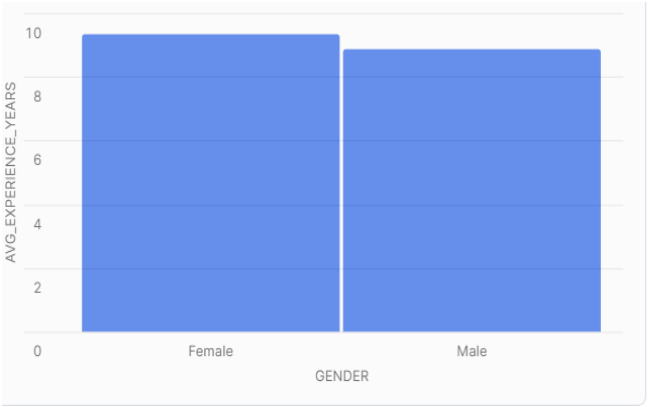
**Deriving Insights**

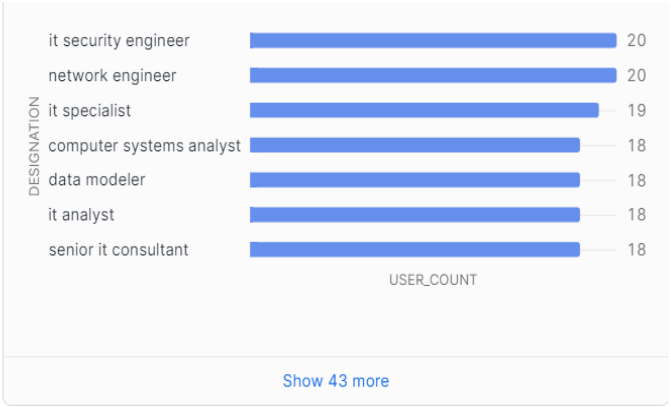
Data Build tool outputs insights derived from the transformed data, providing valuable analytics and insights into various aspects of the dataset. These insights encompass descriptive statistics, trend analysis, pattern recognition, and other metrics relevant to the specific use case, such as predicting events based on user skills and experience.



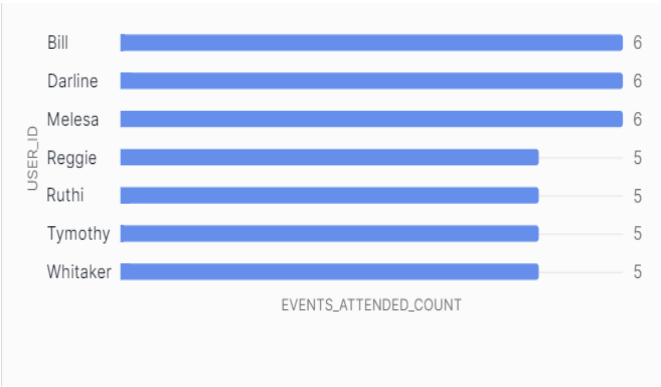
**Reporting and Analysis**

The Visualization of the transformed data is done using the dashboards in the snowflake.

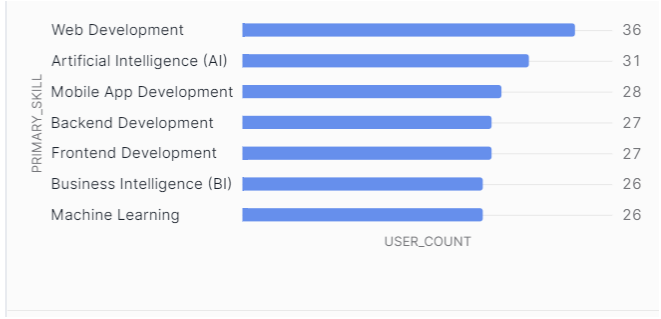











## 6.1.2 Approach for the Machine Learning.

**Exploratory Data Analysis (EDA):**

Exploratory Data Analysis (EDA) is a crucial initial step in any data analysis or machine learning project. It involves exploring and summarizing the main characteristics of a dataset to understand its underlying structure, patterns, and relationships.

**Feature Extraction:**

Feature extraction is the process of transforming raw data into a set of meaningful features that can be used as input for machine learning algorithms.

**Model Development:**

Model development involves constructing and training machine learning models with preprocessed data derived from exploratory data analysis and feature extraction. The objective is to produce predictive

models capable of providing precise forecasts on unseen data, thereby addressing real-world challenges and producing actionable insights.

For predicting events based on user skills and experience, employing random forest and decision tree algorithms is common. Additionally, collaborative filtering techniques are often utilized, leveraging user-item interaction data to generate personalized event recommendations.

# 7 Project Timeline and Resources

## 7.1.1 Roles and responsibilities.

In the realm of roles and responsibilities, I, Alphius Victoria Ashlee, take on a pivotal role within this individual project. With a primary focus on developing the full stack application, I spearhead the end-to-end creation of the application, ensuring seamless functionality and user experience across the frontend and backend.

Beyond application development, I delve into the domain of data engineering, where I am tasked with the intricate process of transforming data. Leveraging my expertise in data science, I navigate through datasets, employing various techniques and methodologies to refine raw data into valuable insights.

Furthermore, I take on the formidable challenge of predictive analytics, leveraging my skill set to forecast events relevant to user interaction. Through a combination of data manipulation, statistical analysis, and predictive modeling, I craft a personalized experience for users, tailored to their skills and experiences within the realm of data science.

## 7.1.2 Project Timeline.