**Employee Learning Platform**

Mini Project

Contents

[1 Project Overview 2](#_Toc137748514)

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

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

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

[2 Project Scope 3](#_Toc137748518)

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

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

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

[3.1.1 Overall architecture of the web app. 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 Web App Components 6](#_Toc137748526)

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

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

[5 User Interface Design 8](#_Toc137748529)

[5.1.1 User interface (UI) design approach. 8](#_Toc137748530)

[6 Methodology for Data 9](#_Toc163818504)

[6.1.1 Methodology for data engineering. 12](#_Toc137748532)

[6.1.2 Methodology for Machine Learning. 13](#_Toc137748533)

[7 Project Timeline and Resources 15](#_Toc137748535)

[7.1.1 Estimated project timeline, including major milestones. 15](#_Toc137748536)

[7.1.2 Roles and responsibilities of the project team members. 15](#_Toc137748537)

[8 Risks and Mitigation Strategies 15](#_Toc137748538)

[8.1.1 Potential risks and challenges associated with the project. 15](#_Toc137748539)

[9 Improvements 11](#_Toc163818511)

[9.1.1 Improvements – Full Stack. 11](#_Toc163818512)

[9.1.2 Improvements – Data Engineering. 11](#_Toc163818513)

# Project Overview

### Brief introduction of the project.

Our project aims to develop a learning platform for employees, with key features such as a secure login page, event creation, user registration, email notifications, and capacity management. The establishment of a robust data platform to support reporting and analytics, ultimately aiming to enhance employee learning and organizational effectiveness.

### Purpose of the project.

The project aims to develop an all-encompassing learning platform tailored for organizational employees.

The key objectives of the project include:

• To offer employees a unified hub for accessing diverse learning avenues, such as training sessions and workshops, bolstering their skills and career progression.

• Utilizing automated email alerts to maintain consistent communication with employees about upcoming events, registration status, and any modifications or updates to the event schedule, enhancing engagement and involvement.

• Setting up a data infrastructure for gathering, refining, transforming, and analyzing pertinent data, facilitating the creation of actionable reports and insights to guide decision-making processes in optimizing and overseeing business operations linked to employee skill development initiatives.

### Goals of the project:

* Create an intuitive learning platform ensuring smooth navigation for administrators and employees, guaranteeing a seamless user experience.
* Design of the Core application tier including:

o A user-friendly Front End for employees, accessible through a web application.

o Utilize JMAN's provided Back End API.

The application will support Android version 10.0 and above, and iOS 11 and above

# Project Scope

### Web app support

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

### Key features and functionalities of the mobile app.

**User Registration and Authentication:**

Intuitive registration procedure enabling admins to establish employee profiles.

Robust authentication protocols, featuring username/password combinations.

**Personalised Admin Interface:**

Adapted dashboard allowing admins to craft events (training), modify event details, and manage user enrollments.

**Personalised Employee Interface:**

Individualized dashboard for every employee, showcasing pertinent details like event registrations.

Instantaneous notifications about forthcoming events.

**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 mobile app.

Create an employee-focused learning platform with essential features like a secure login, event management, user sign-up, email alerts, and capacity oversight. Establish a data platform for reporting and analytics, aiming to amplify both employee learning and organizational efficiency.

**Front-end Application:**

* User Authentication: This component manages user sign-up and login procedures, offering an admin-friendly interface to securely establish employee accounts with username/password validation mechanisms.
* Admin Interface: A dashboard for administrators to oversee events (training), modify event details, and manage user enrollments, featuring user-friendly interfaces for CRUD operations on events.
* Employee Interface: An individualized dashboard for each employee, showcasing pertinent details such as enrolled events, all events, and event interest indicators.

**Back-end Services:**

* Authentication Mechanism: Manages both admin and employee (user) authentication and authorization, validating user details and issuing access tokens for verified users.
* User Management: Manages user accounts, including registration and role-based access control (admin vs. employee).
* Event Supervision: Supervises training events with necessary CRUD operations
* Notification Mechanism: Sends instantaneous updates and alerts to users about event registrations, alterations via their email.

**Data Engineering Pipeline:**

* Data Extraction: Connects to MongoDB to snowflake to extract relevant data related to training events, user profiles, etc. Connecting snowflake to dbt to get real-time data from the snowflake.
* Data Cleaning and Transformation: Cleans and preprocesses the extracted data to ensure data quality and consistency, transforming it into a suitable format for analysis.
* Reporting Tables: Creates reporting tables and views to support operational reporting and analytics, providing insights into event attendance, user engagement, etc.

**Machine Learning Pipeline:**

* Feature Engineering: Extracts relevant features from the data, including user attributes, event characteristics etc., to feed into the recommendation model.
* Model Training: Trains machine learning models using data to generate personalized training recommendations for employees.

### Architecture

### 

Full Stack Architecture



Data Engineering Architecture

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

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

The technology stack for the Mobile Application Platform for Employee leasrning Platform will include the following components:

**React JS:**

A JavaScript library for building user interfaces, providing a declarative, component-based approach to UI development that enables efficient and scalable web applications.

**Axios:**

A popular HTTP client for JavaScript that allows making asynchronous HTTP requests to fetch or send data from/to a server, offering features like promise-based requests and interceptors for handling request and response transformations.

**Node.js:**

A runtime environment that allows executing JavaScript code server-side, facilitating building fast and scalable network applications using an event-driven, non-blocking I/O model.

**Cascading Style Sheets:**

CSS is a stylesheet language used for describing the look and formatting of a document written in HTML. It enables the separation of content from presentation, allowing developers to style web pages with colors, fonts, and layouts.

**MongoDB:**

A NoSQL database system that stores data in flexible, JSON-like documents, providing high performance, scalability, and flexibility for data storage and retrieval in modern web applications.

**Snowflake:**

A cloud-based data warehousing platform that offers a fully-managed service for storing, processing, and analyzing large volumes of structured and semi-structured data, enabling fast and scalable analytics.

**DBT:**

A tool for transforming data in the data warehouse, providing a modeling layer that allows defining business logic, version control, and documentation, enabling analysts and data engineers to build and manage data pipelines efficiently.

**Python:**

A versatile programming language known for its simplicity and readability, widely used for web development, data analysis, artificial intelligence, scientific computing, and automation, offering a rich ecosystem of libraries and frameworks for various applications.

### Rationale behind the chosen technology stack.

In developing the Mobile Application Platform for Instrument Tracking in Healthcare, we have chosen React Native as the primary technology stack. The rationale behind this selection is based on the following considerations:

**Cross-Platform Compatibility:**

React enables building a single codebase for all the platforms thus drastically reducing development time. This approach ensures a consistent user experience across platforms by leveraging a shared codebase.

**Time and Cost Efficiency:**

Using React streamlines the development process, letting us write code once for both the main platforms . This efficiency saves time and cost, accelerating the application's time-to-market.

**Large Developer Community and Ecosystem:**

React benefits from an extensive developer community offering robust support and libraries. This active community accelerates development by providing existing solutions and problem-solving resources.

**Code Reusability:**

React's component-based architecture promotes code reusability across different screens. By developing reusable components, we simplify both development and maintenance, as changes to shared components apply throughout the app.

**Flexibility and Customizability:**

React provides flexibility through customization and integration with native modules. This flexibility allows us to meet specific requirements, integrate with existing systems, and tailor the application to unique needs.

**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, abbreviated for "Data Build Tool," stands as an open-source tool empowering data analysts and engineers to conduct data transformations within their data warehouse. Operating under a "transformation-as-code" paradigm, dbt facilitates the authoring of SQL queries to convert raw data into analytics-prepared tables efficiently.

**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.

**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, 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 chose the mentioned above stack as the optimal technology stack for developing our Web Application Platform.

# Mobile App Components

### Main components of the mobile app.

* Welcome
* Login
* Changepass
* LandingPage
* Create
* Event
* EditEvent
* DeleteEvent
* RegisterEvent
* DisplayRegEvents
* DisplayLikedEvents
* Skills

### Purpose of each component.

#### Welcome

#### The welcome page serves as an entry point for both admins and users, offering navigation to the login section for accessing the platform. This user-friendly interface ensures seamless access and navigation for all types of users.

#### Login

The login page serves as the gateway for users and admins to access their dedicated dashboards. Through this component, both groups authenticate and gain entry to their respective interfaces.

#### ChangePass

This component allows users who have forgotten their passwords to initiate a password reset process. When triggered, an email containing a link is sent from the admin to the user, enabling them to securely reset their password to a new one of their choice.

#### LandingPage

The landing page serves as the initial interface for both users and administrators, directing them to their respective dashboards. The admin dashboard features options to create new events and add users, facilitating the management of events and user accounts. Conversely, the user dashboard provides functionalities to register for events and manage skills, enabling users to engage with available learning opportunities. Additionally, both dashboards include a logout button, allowing users and administrators to securely exit their sessions when needed.

#### Create

This page facilitates the user creation process for administrators by allowing them to input necessary user details. Once the admin completes the user creation, an automated email is sent to the newly created user notifying them of their account setup.

#### Event

This feature allows administrators to input event details and create new events within the system. Once an event is created, an automated email notification is sent to all users, informing them about the newly created event. This ensures that all users are promptly informed and can register or participate in the event as needed.

#### EditEvent

This page ensures that the event modification is done properly. Once an event is edited, an automated email notification is sent to all users, informing them about the newly edited event. This ensures that all users are promptly informed and can register or participate in the event as needed.

#### DeleteEvent

This component enables administrators to remove existing events from the system.

#### RegisterEvent

This page allows users to accept event invitations, facilitating their participation in the selected events. Additionally, users have the option to like or unlike specific events, indicating their interest and preferences within the platform. Once they accept invitation, an automated email notification is sent to that users, informing them about their successful acceptance

#### DisplayRegEvents

This shows all the events for which the user has accepted the invitation.

#### DisplayLikedEvents

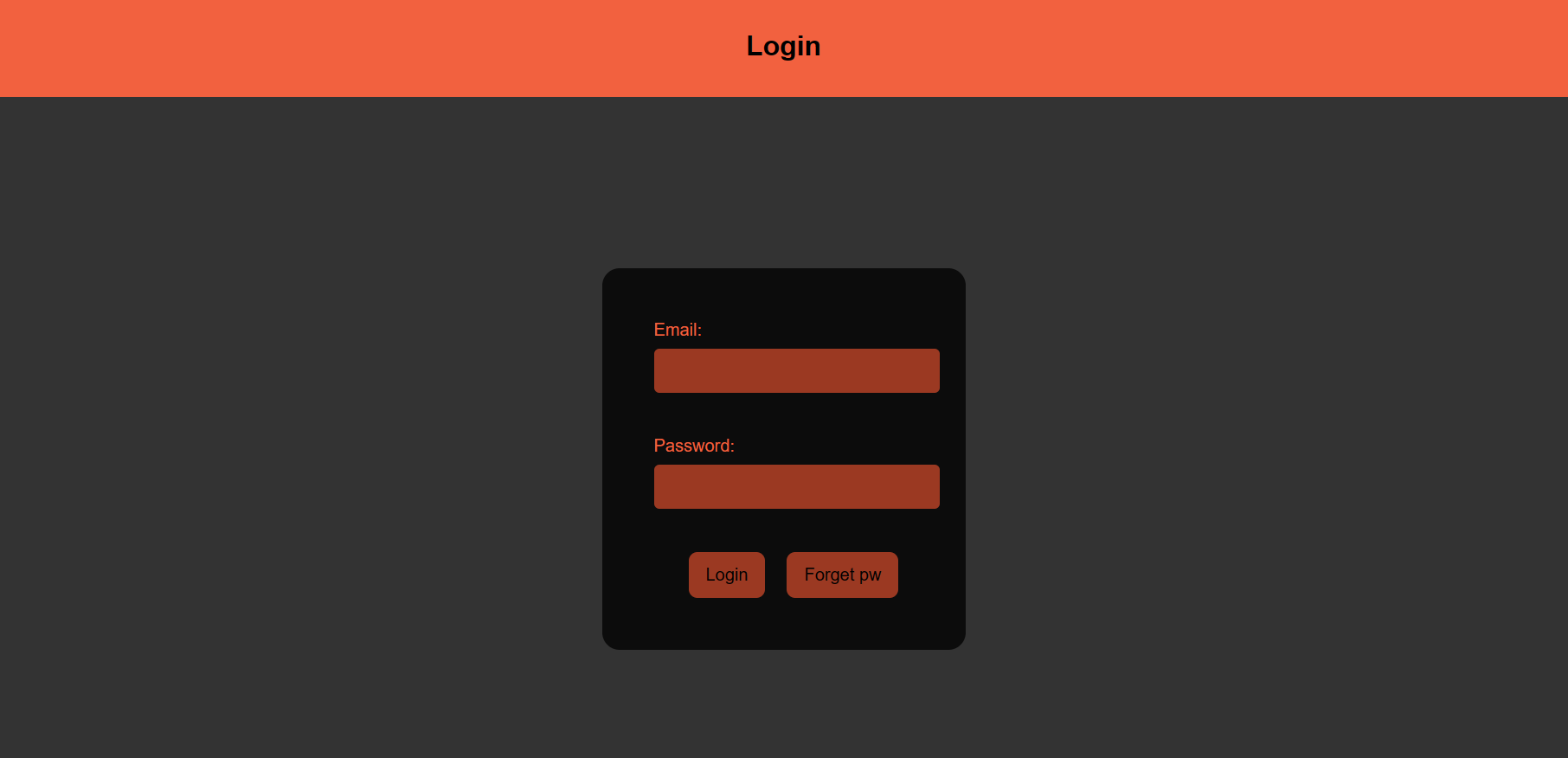
#### This shows all the events for which the user has pressed the like button.

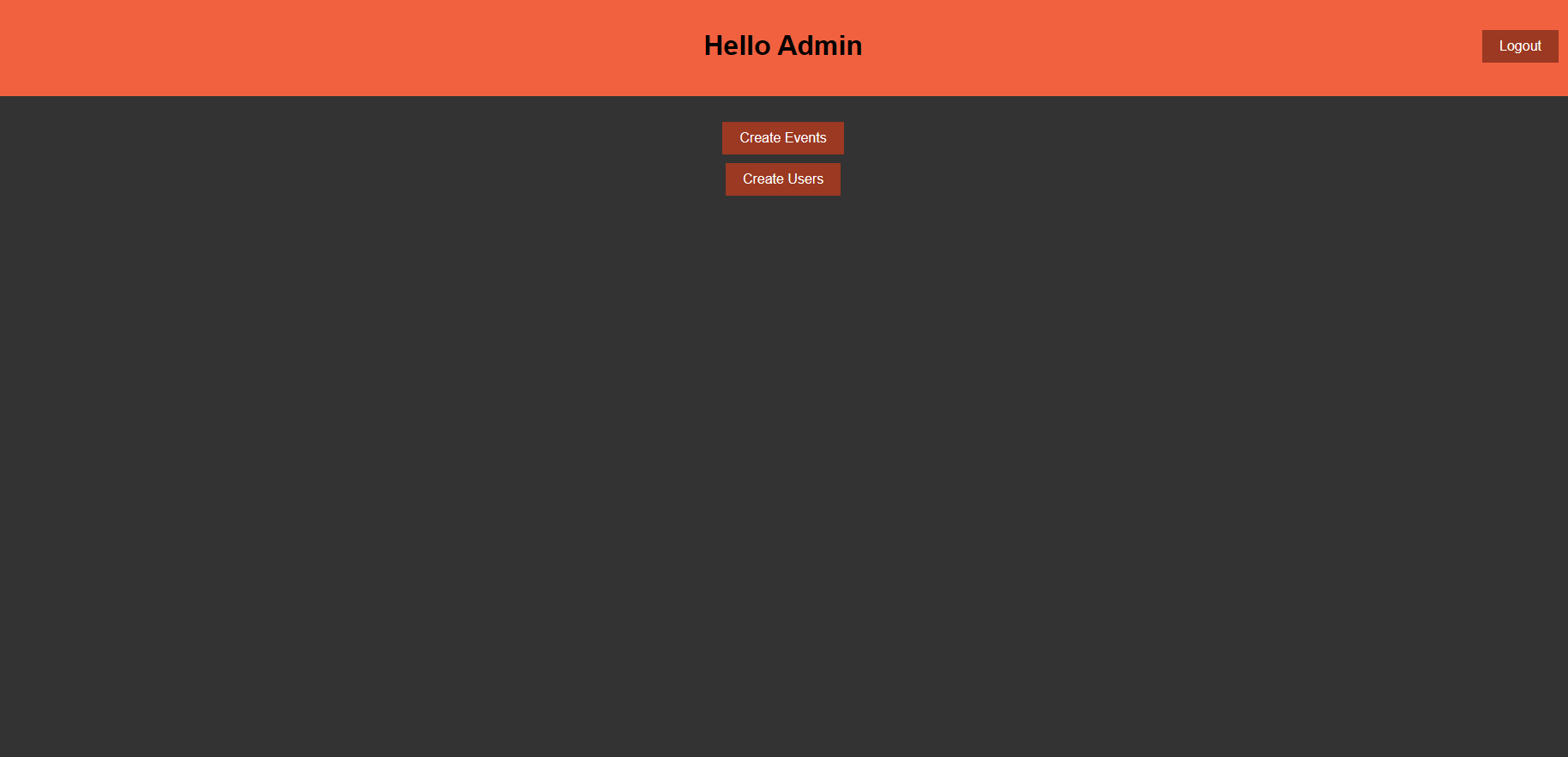
#### Skills

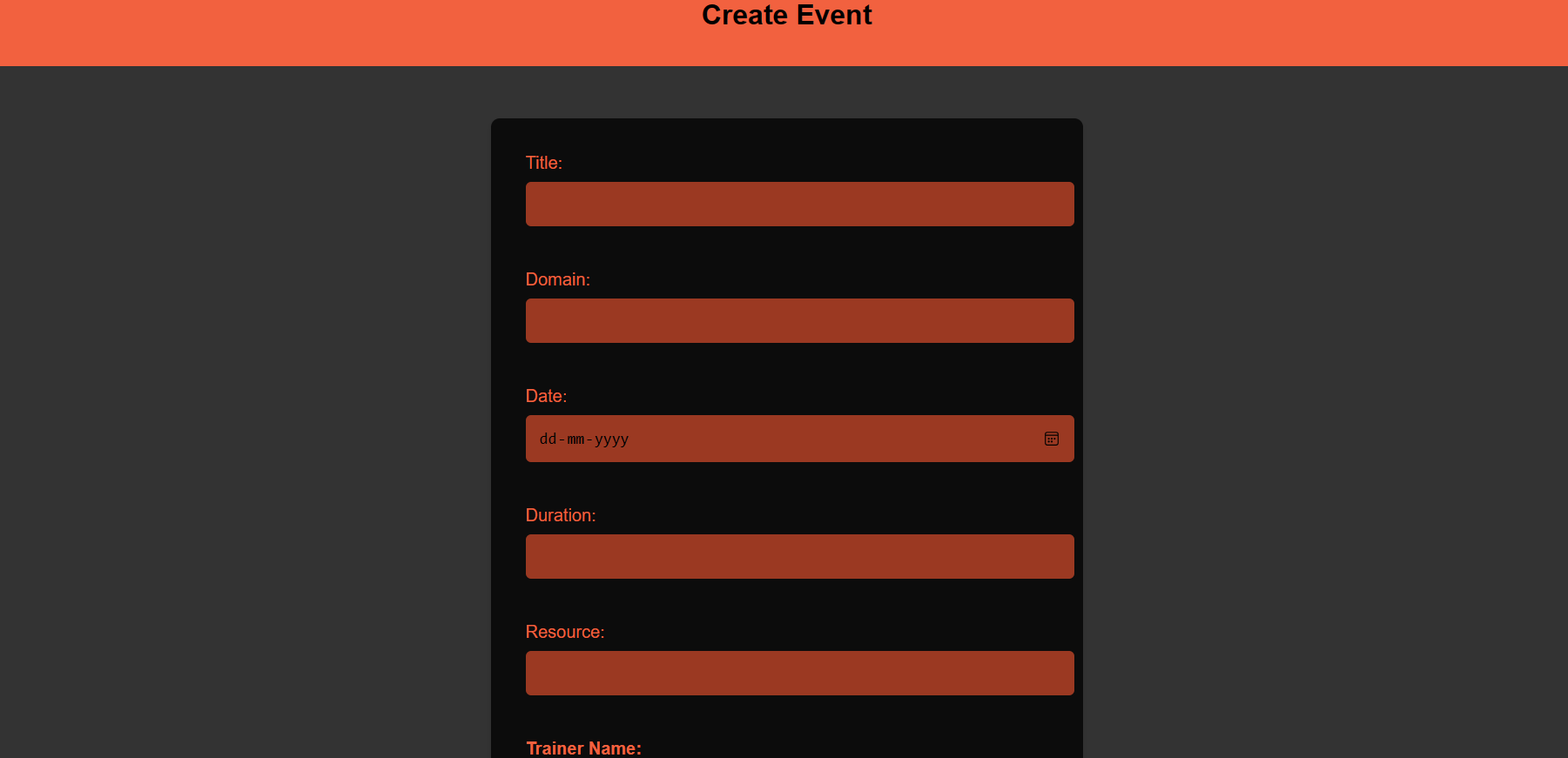
Through this page, a user will be able to successfully add their skills and their details

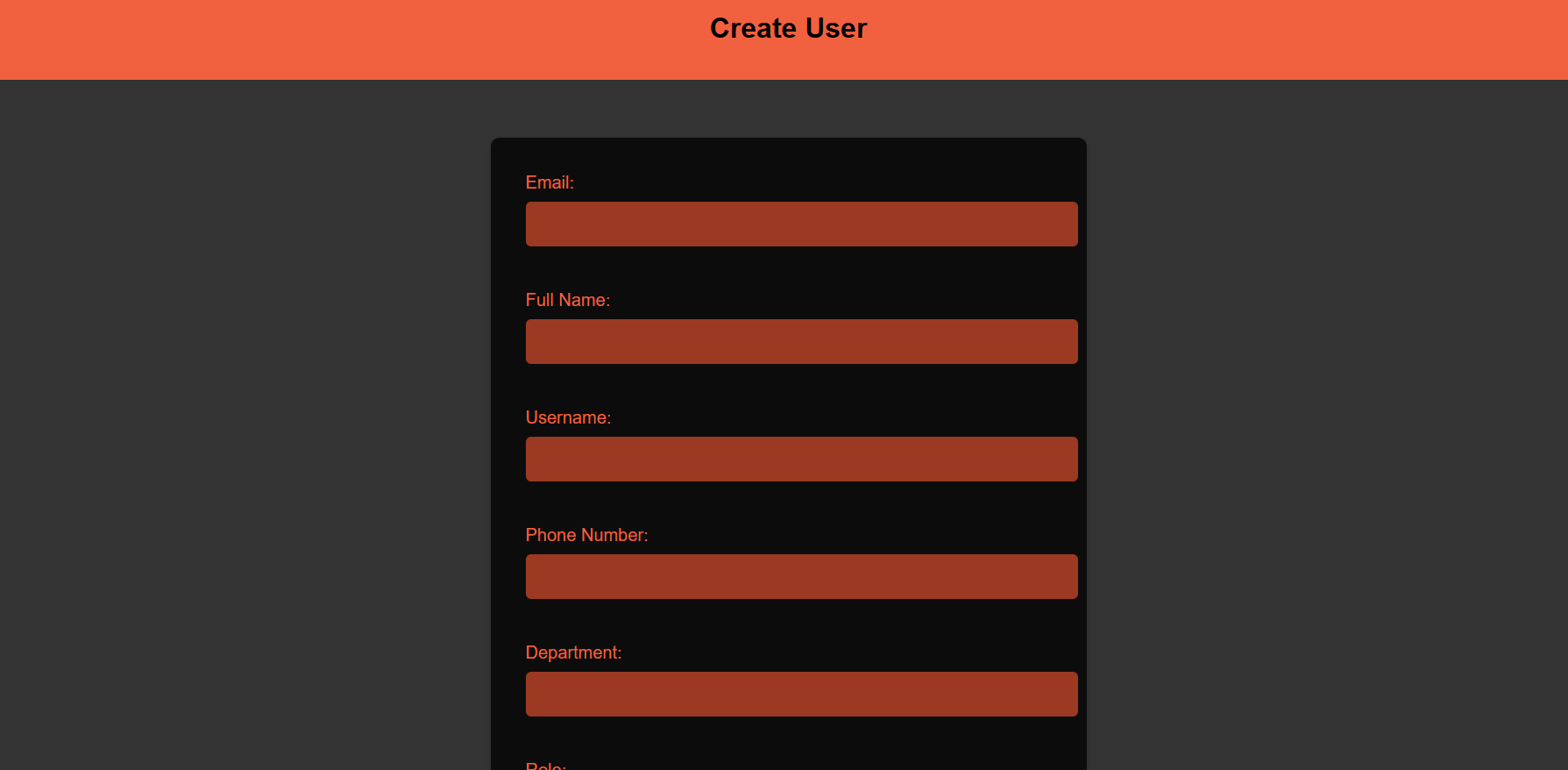
# User Interface Design

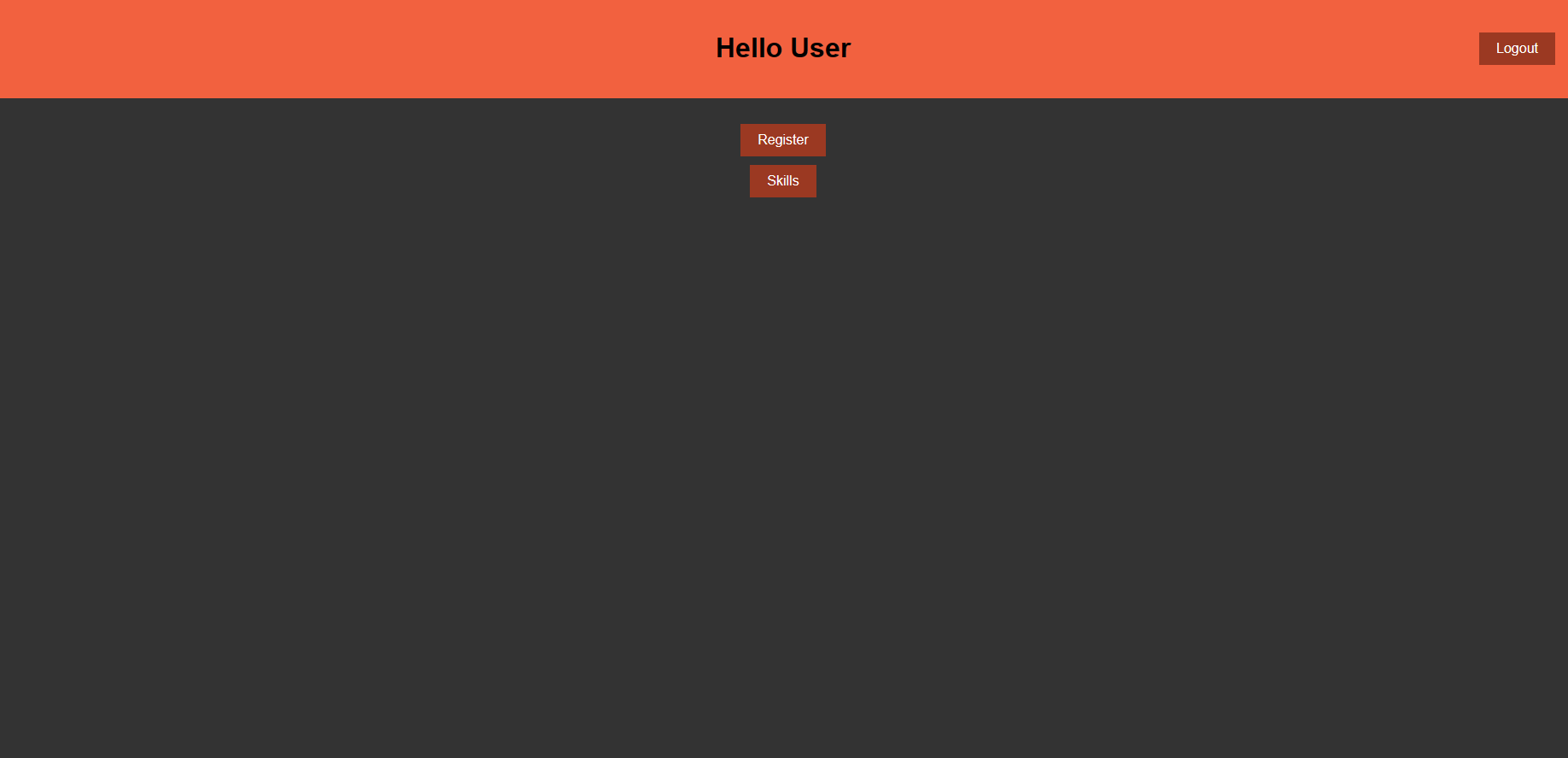
### User interface (UI) design approach.

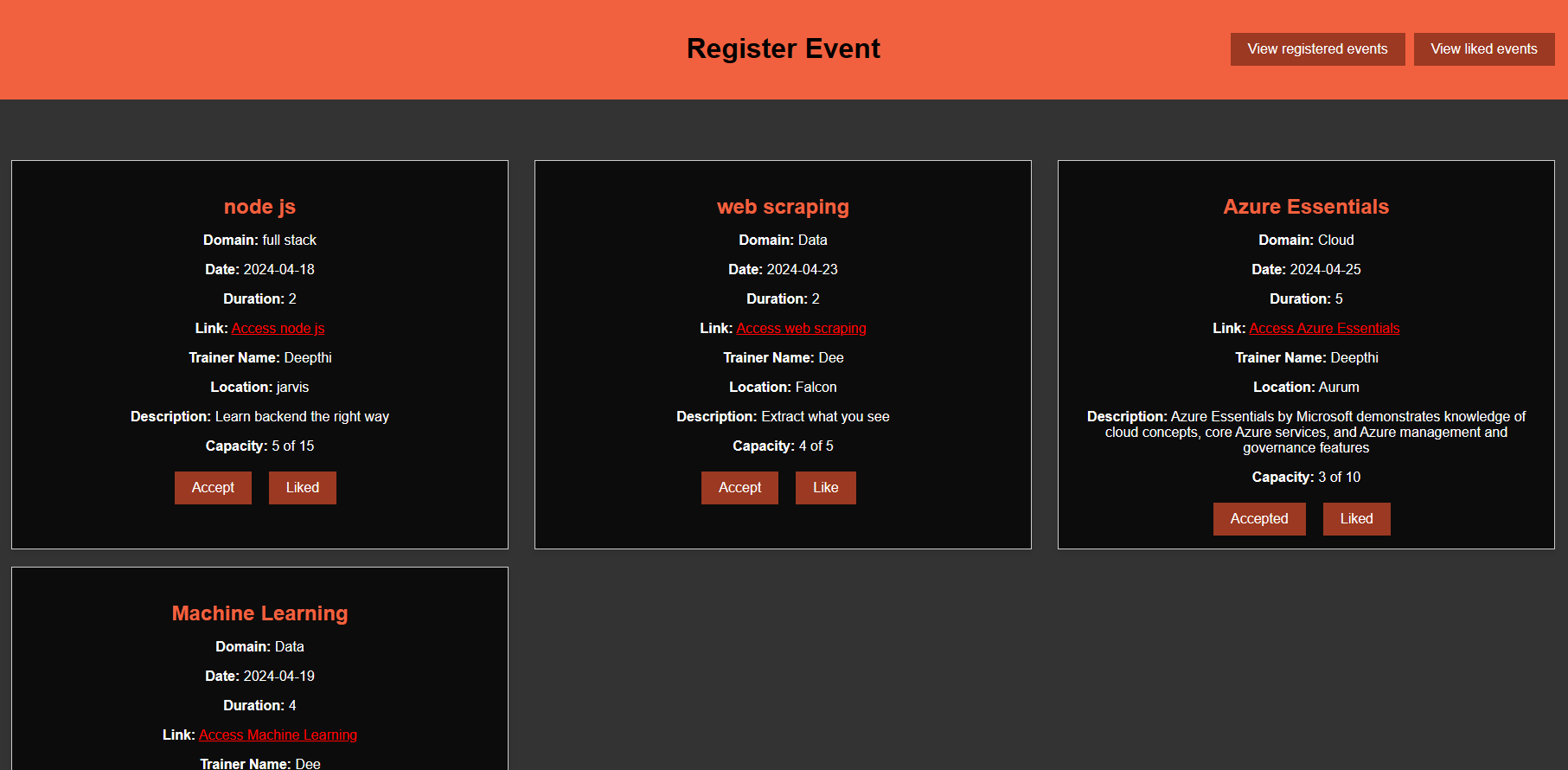


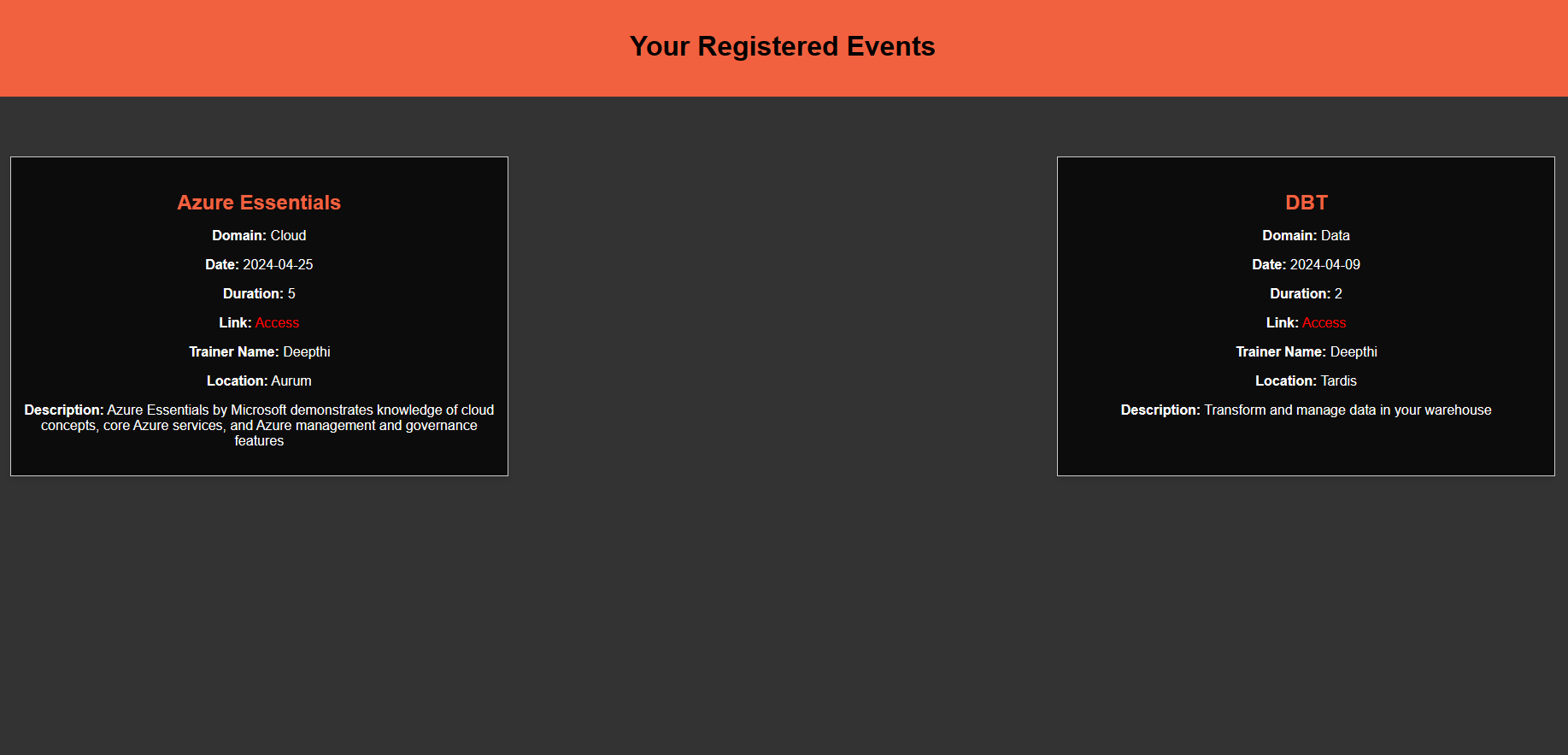


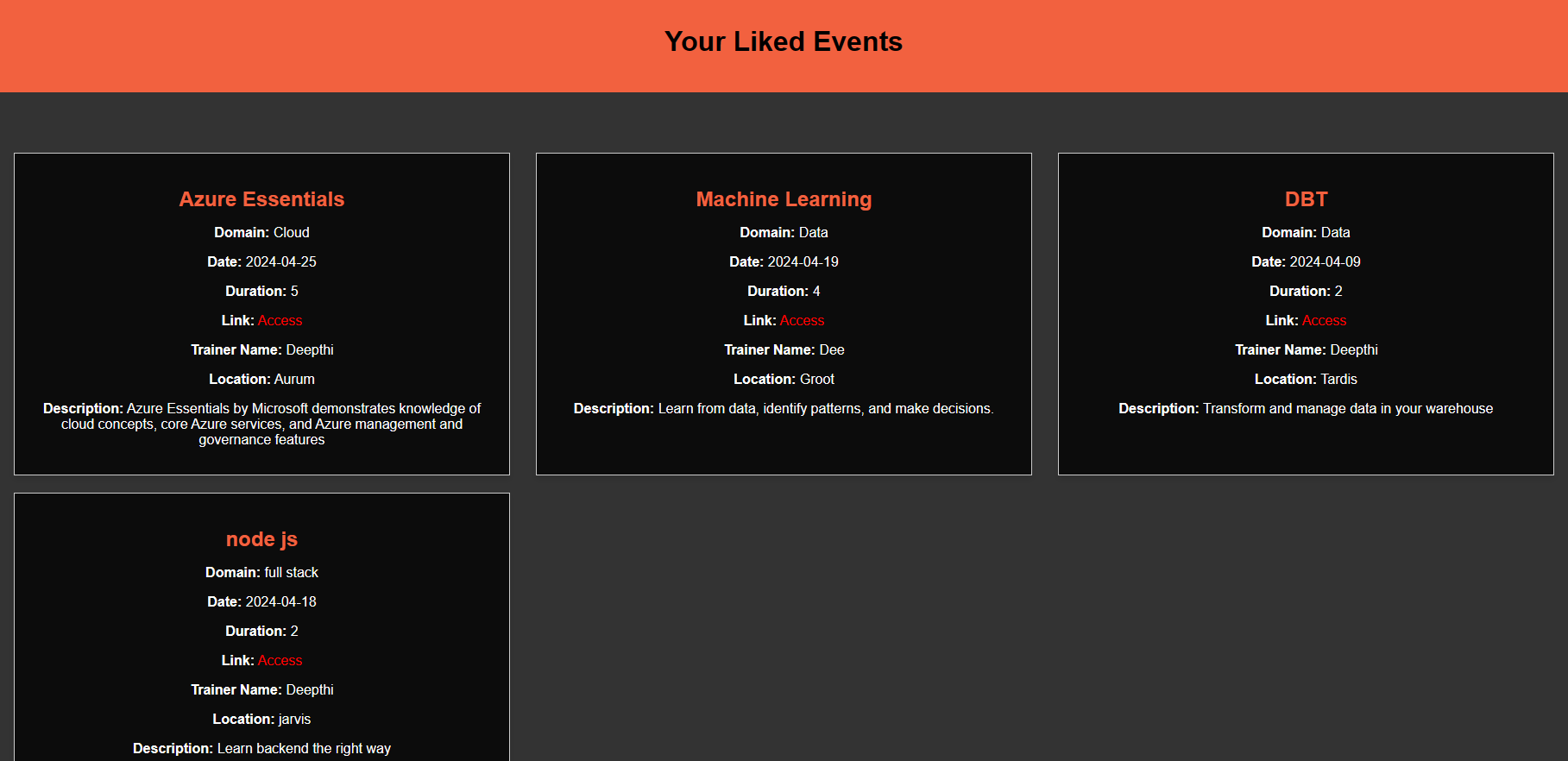


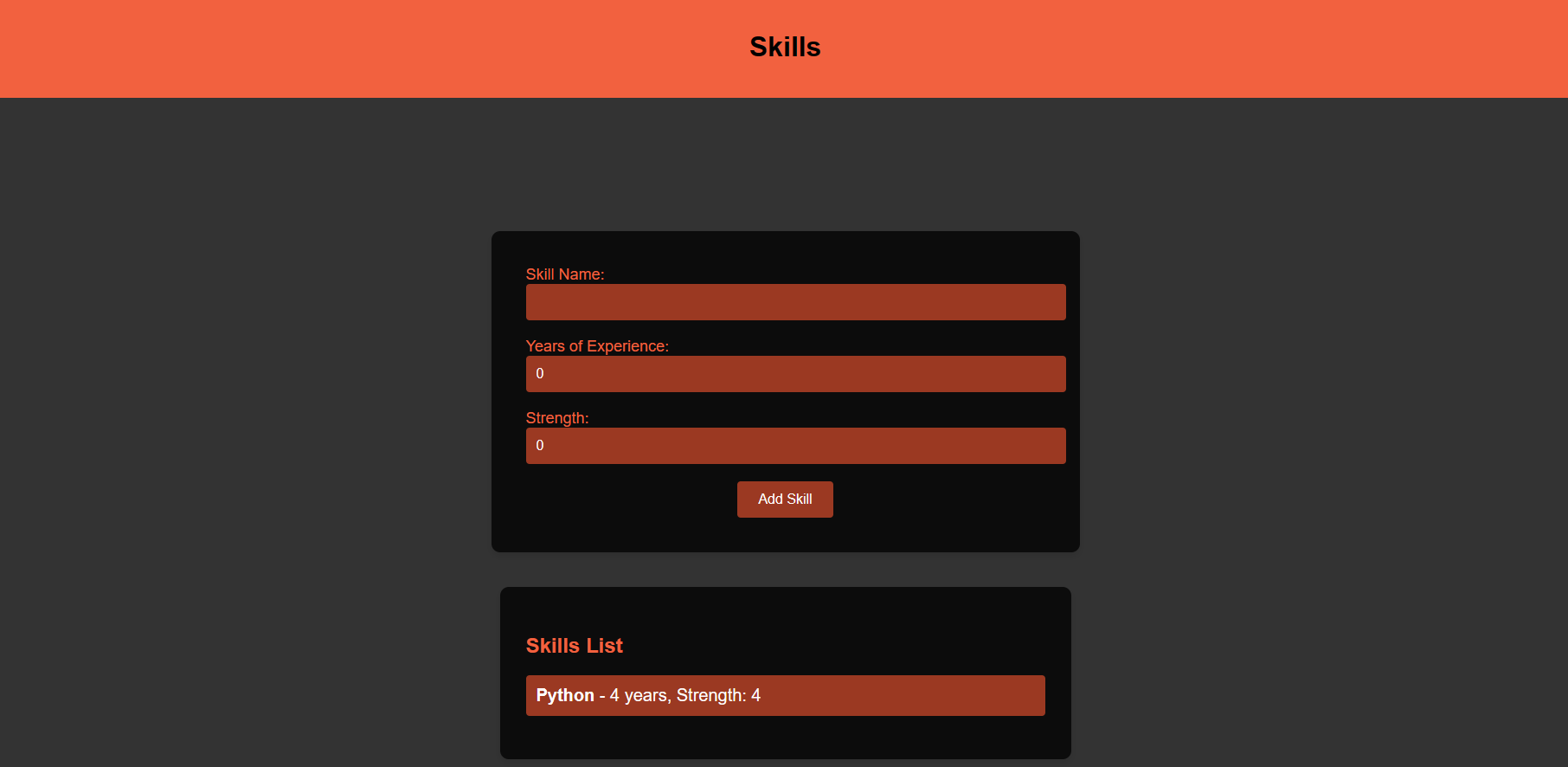












# Methodology for Data

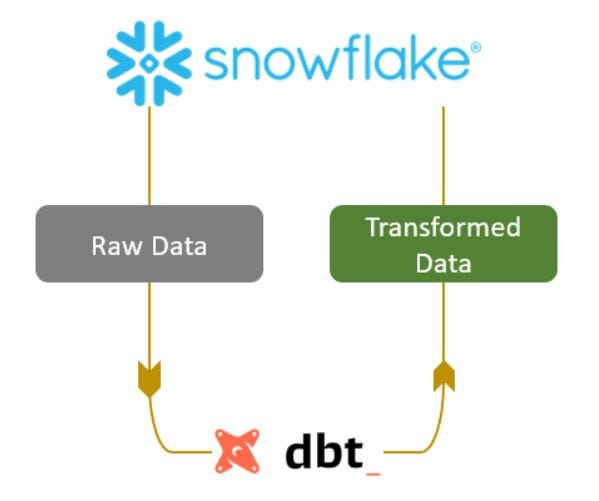
Testing and quality assurance are essential processes in software development that aim to ensure the reliability, functionality, and overall quality of a software product.

### Methodology for Data Engineering

**Snowflake Integration:**

Snowflake Integration involves leveraging Snowflake as the primary data storage solution. By utilizing the snowflake-connector, you establish a connection between Snowflake and your MongoDB database, facilitating seamless data transfer and synchronization between the two platforms.

**DBT:**By incorporating dbt into your data workflow, you streamline transformation of data marts within Snowflake, ensuring consistent and reliable data preparation. This integration enhances the efficiency of analytics by standardizing data transformation processes and enabling more straightforward, automated workflows.



**Connected dbt to Snowflake:**

An ETL (Extract, Transform, Load) pipeline is set up to pull data from Snowflake, process or modify it as needed, and then reload it back into Snowflake. This ensures data quality and consistency across the Snowflake database, enabling efficient analytics and reporting.

### Methodology for Machine Learning

**Exploratory Data Analysis (EDA):**

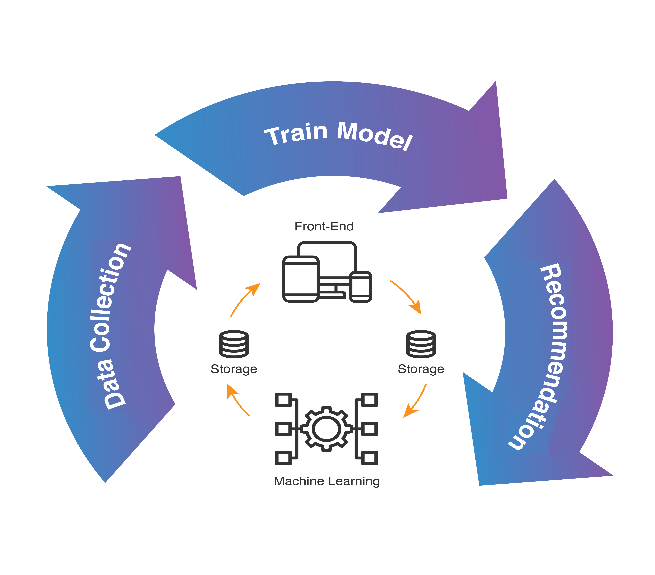
Exploratory Data Analysis (EDA) is a crucial initial step in any data analysis or machine learning project. It involves exploring and summarizing the main characteristics of a dataset to understand its underlying structure, patterns, and relationships.

**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 is the process of building and training machine learning models using the preprocessed data obtained from exploratory data analysis and feature extraction.The ultimate goal of model development is to create predictive models that can make accurate and reliable predictions on unseen data, thereby solving real-world problems and generating actionable insights.  
Using cosine similarity to build a recommendation system is a common approach, especially in content- based recommendation systems. In this model the employee skills are taken into consideration to suggest them the events to attend.



# Project Resources

### Roles and responsibilities of the project team members.

Nadiminti Deepthi - Involved in developing the full stack application, data engineering and data science

# Risks and Mitigation Strategies

### Potential risks and challenges associated with the project.

|  |  |  |  |
| --- | --- | --- | --- |
| S.NO | Risk / Challenges | Impact | Mitigation Plan |
| 1 | If there is a delay in API | Project deliverables will be delayed | Need to have all the API ahead of development team |

# Improvements

* **Implement Calender and Block the calendars:**

Implementing a calendar feature lets users manage upcoming events visually, enhancing the user experience. Blocking calendars marks full or past-deadline events as unavailable, preventing overbooking and maintaining accurate platform data.

* **Admins can't register for events:**

Restrict the ability for admins to register for events they are scheduled to conduct, ensuring accurate attendee lists, and preventing conflicts of interest.

* **Employee should be sent an email if he is the trainer:**

Implement automated email notifications to inform employees when they are assigned as trainers for specific events or tasks.

* **Date merging:**

Implement logic to merge overlapping date ranges or events to prevent conflicts and ensure accurate scheduling.

* **Automate Data Transfer from MongoDB to Snowflake:**

Implement an ETL pipeline to extract data from MongoDB, transform it if necessary, and load it into Snowflake.