

Employee Skill Assessment & Competency Tracking

Major Project

Contents

- 1 Project Overview 2**
 - 1.1.1 Brief introduction of the project. 2
 - 1.1.2 Purpose of the project. 2
 - 1.1.3 Goals of the project: 2
- 2 Project Scope 2**
 - 2.1.1 Web app support..... 2
 - 2.1.2 Key features and functionalities of the web app. 2
- 3 Architecture and Technology Stack 3**
 - 3.1.1 Overall architecture of the mobile app. 3
 - 3.1.2 Architecture 4
 - 3.1.3 Technology stack (e.g., programming languages, frameworks, libraries). 4
 - 3.1.4 Rationale behind the chosen technology stack. 5
- 4 Web App Components 7**
 - 4.1.1 Main components of the mobile app. 7
 - 4.1.2 Purpose of each component. 7
- 5 User Interface Design 7**
 - 5.1.1 User interface (UI)..... 7
- 6 Work with Data 8**
 - 6.1.1 Approach for the Data Engineering. 8
 - 6.1.2 Approach for the Machine Learning..... 9
- 7 Project Resources 11**
 - 7.1.1 Roles and responsibilities of the project team members. 11
- 8 Risks and Mitigation Strategies..... 11**
 - 8.1.1 Potential risks and challenges associated with the project..... 11
- 9 Improvements 11**
 - 9.1.1 Improvements – Full Stack. 11
 - 9.1.2 Improvements – Data Engineering. 12

1 Project Overview

1.1.1 Brief introduction of the project.

This system tracks employee skills and competencies based on completed courses and assessments, enhancing workforce development and providing personalized learning recommendations.

1.1.2 Purpose of the project.

The purpose of the Employee Skill Assessment & Competency Tracking project is to create a comprehensive system that enables organizations to effectively monitor and assess employee skills and competencies. By tracking completed courses and performance in assessments, the system aims to:

- Enhance workforce development and employee growth.
- Provide actionable insights for skill gaps and training needs.
- Facilitate personalized learning paths and course recommendations.
- Support data-driven decision-making in talent management.

1.1.3 Goals of the project:

- **Skill Tracking:** Develop a robust system to accurately track and assess employee skills and competencies.
- **Data Integration:** Integrate data from various sources to provide a comprehensive view of employee qualifications and training needs.
- **User-Friendly Interfaces:** Create intuitive dashboards and profile pages for administrators and employees to easily access and manage skill data.
- **Personalized Learning:** Implement machine learning algorithms to recommend targeted learning paths and courses based on individual skill sets and job requirements.
- **Continuous Improvement:** Enable ongoing assessments and updates to ensure the system evolves with organizational needs and employee development.

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 Registration and Authentication:

User-friendly registration process for admin to create employee accounts.

Secure authentication mechanisms, including username/password.

Personalized Admin Dashboard:

Customized dashboard for admins to create events (training), update event information and user registration.

Personalized Employee Dashboard:

Customized dashboard for each employee, displaying relevant information such as events registration.

Real-time updates on upcoming events.

Responsive Design:

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

3 Architecture and Technology Stack

3.1.1 Overall architecture of the app.

Develop a comprehensive system that tracks employee skills and competencies based on completed courses and assessments, enhancing workforce development and providing personalized learning recommendations. The system aims to support data-driven decision-making in talent management.

- **Frontend Application**
 - **User Registration and Authentication:**
 - This module handles user registration and authentication processes, providing a user-friendly interface for administrators to get employee accounts securely.
 - **Admin Dashboard:**
 - A personalized dashboard for administrators to monitor employee skills, update course information, and manage user registrations. It supports intuitive CRUD (Create, Read, Update, Delete) operations on skill assessments and competencies.
 - **Employee Dashboard:**
 - A dashboard for each employee, displaying relevant information such as completed courses, assessment results, and recommendations for further learning opportunities.
 - **Responsive Design:**
 - The frontend application is designed with a responsive user interface to ensure optimal viewing and usability across various devices and screen sizes, including desktops, tablets, and smartphones.
- **Backend Services**
 - **Authentication Service:**
 - Handles admin and employee (user) authentication and authorization, verifying user credentials and generating access tokens for authenticated users.
 - **User Management Service:**
 - Manages user accounts, including registration, profile updates, and role-based access control (admin vs. employee).
 - **Skill Management Service:**
 - Responsible for tracking employee skills and competencies, managing assessments, and facilitating personalized learning recommendations.
- **Data Engineering Pipeline**
 - **Data Extraction:**
 - Connects to various data sources to extract relevant data related to completed courses, employee profiles, and assessment performance.
 - **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 skill gaps, training needs, and employee growth.
- **Machine Learning Pipeline**
 - **Feature Engineering:**
 - Extracts relevant features from the data, including employee attributes, completed courses, and assessment scores, to feed into the recommendation model.
 - **Model Training:**
 - Trains machine learning models using historical data to generate personalized learning paths and course recommendations for employees.

3.1.2 Architecture

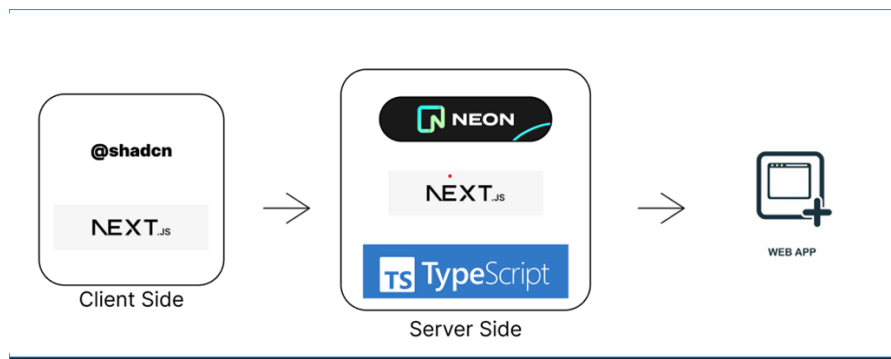


Figure 1 Full Stack Architecture

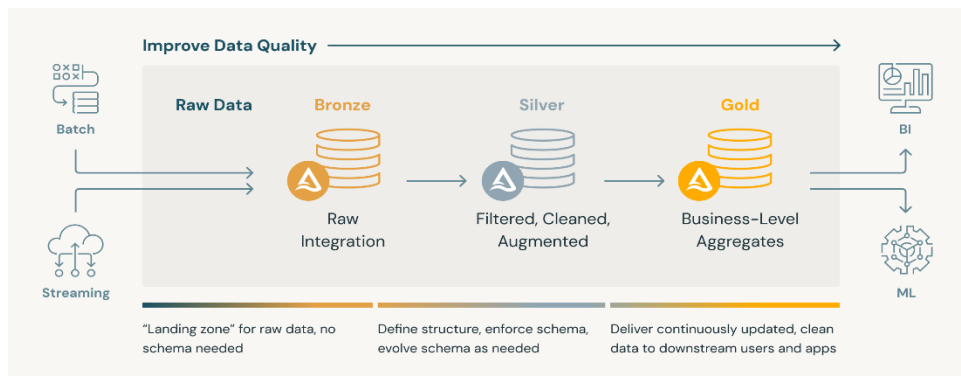


Figure 2 Data Engineering Architecture

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

- **Next.js with TypeScript**
Next.js, paired with TypeScript, is utilized for building the web application. This framework provides server-side rendering, static site generation, and efficient routing, making it well-suited for creating a fast, scalable, and SEO-friendly platform. TypeScript enhances code maintainability and reduces errors through type safety.
- **Shadcn for UI Components:**
Shadcn is used for designing and implementing user interface components. It offers a set of pre-built, customizable components that streamline the development process and ensure a consistent and attractive UI across the application.
- **Prisma as an ORM:**
Prisma serves as the Object-Relational Mapping (ORM) tool for database access. It simplifies data modeling and database interactions, allowing developers to write type-safe queries and manage database migrations easily. This enhances productivity and reduces the likelihood of errors in database operations.
- **Next.js Server Components:**
Next.js Server Components are leveraged for backend operations. This allows for rendering components on the server, which can lead to improved performance and user experience by reducing the amount of client-side JavaScript needed. It also facilitates direct access to data and server-side logic, enhancing the application's efficiency.
- **Neon PostgreSQL for Application Data:**

Neon PostgreSQL is employed as the primary database solution due to its cloud-native architecture, scalability, and suitability for handling diverse web application use cases. With its reliable and efficient management of relational data, Neon PostgreSQL supports the storage and retrieval of data related to employee skills, events, and user profiles, making it an ideal choice for managing the platform's core data.

- Neon PostgreSQL, combined with pgAdmin, enables seamless data management and transfer into dbt for transformation. Neon PostgreSQL provides robust data storage and relational capabilities, while pgAdmin supports data management and interaction with the database, facilitating the platform's data transformation and storage needs.

Python:

Python is a versatile and popular programming language widely used in machine learning (ML) and artificial intelligence (AI) applications due to its simplicity, readability, and extensive ecosystem of libraries and frameworks.

3.1.4 Rationale behind the chosen technology stack.

Next.js has been selected as the primary framework for building the web application, coupled with TypeScript for added type safety and improved development experience. This combination offers enhanced scalability, better developer productivity, and a strong foundation for building robust web applications.

Cross-Platform Compatibility:

While Next.js is focused on server-rendered and statically generated web applications, it offers features such as API routes and built-in support for incremental static regeneration, which enables enhanced cross-platform compatibility. The Next.js ecosystem also provides tools and libraries that support a consistent experience across various platforms and devices.

Time and Cost Efficiency:

Utilizing Next.js and TypeScript streamlines the development process by offering built-in routing, optimized performance, and robust tooling. These features contribute to faster time-to-market, allowing the platform to be deployed within the organization more swiftly.

Large Developer Community and Ecosystem:

Next.js, backed by a large and active developer community, offers extensive resources, libraries, and support. This strong ecosystem accelerates development and problem-solving, with many open-source tools and components readily available. TypeScript further enhances this by catching errors early and improving code maintainability, making it easier to integrate additional functionalities as needed.

Code Reusability:

NextJS's component-based architecture, combined with TypeScript, promotes code reusability. By developing reusable components, we can efficiently manage the application's UI elements and logic across different screens. This approach simplifies development and facilitates maintenance and future updates, as changes made to shared components can be propagated throughout the application seamlessly.

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 mobile app.

- Admin Dashboard
- Admin Login
- Admin Profile

4.1.2 Purpose of each component.

4.1.2.1 Admin Dashboard

The **Admin Dashboard** serves as the central hub for administrators, providing access to key areas of the platform, including:

- Team Lead Information: Displays details about team leads within the organization.
- Employee List: Admins can view a comprehensive list of employees. By clicking on an employee's name, the admin can navigate to that employee's specific dashboard for detailed skill tracking and development information.
- Organizational Metrics: Displays general organizational metrics, offering insights into employee engagement, skill distribution, and other relevant data.

4.1.2.2 Admin Login

The **Admin Login** provides a secure entry point for administrators to access the platform. It grants administrative privileges, enabling access to features and data that are restricted from regular users. This secure interface ensures that only authorized personnel can manage sensitive information and oversee platform operations.

4.1.2.3 Admin Profile

The **Admin Profile** page allows administrators to view and manage their personal profile information. This page includes options for updating contact information and resetting passwords, providing admins with quick access to personal account settings.

5 User Interface Design

5.1.1 User interface (UI)

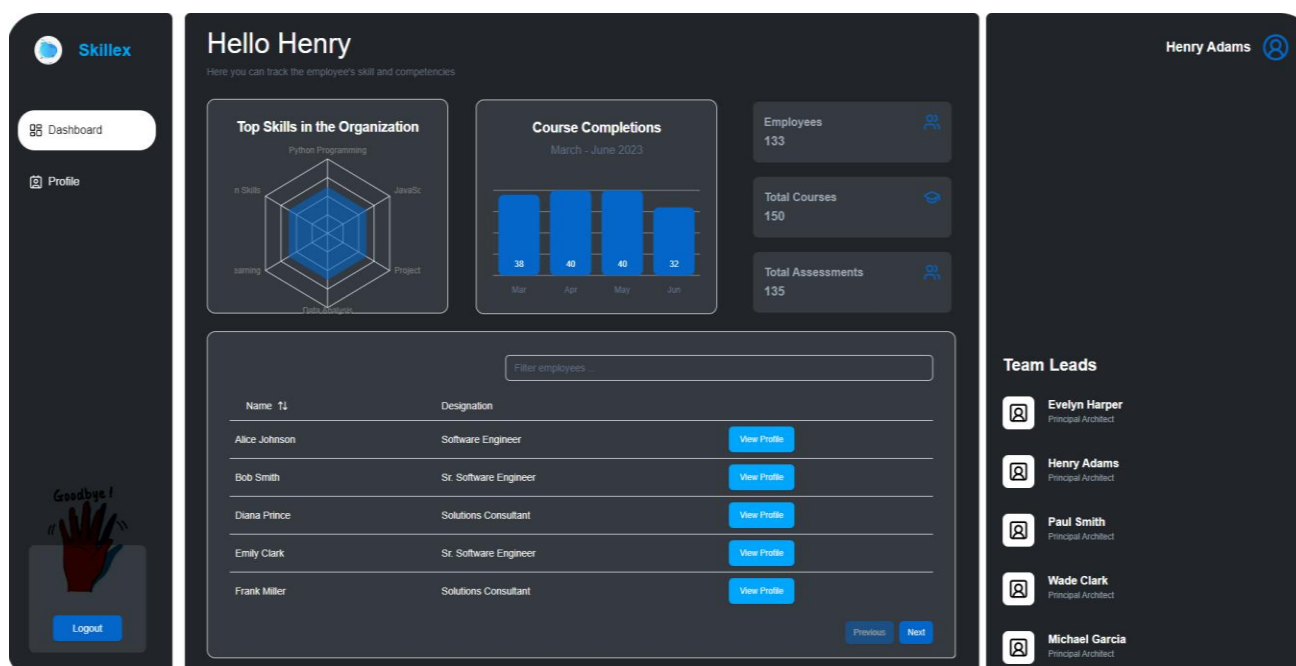


Figure 3 Admin Dashboard

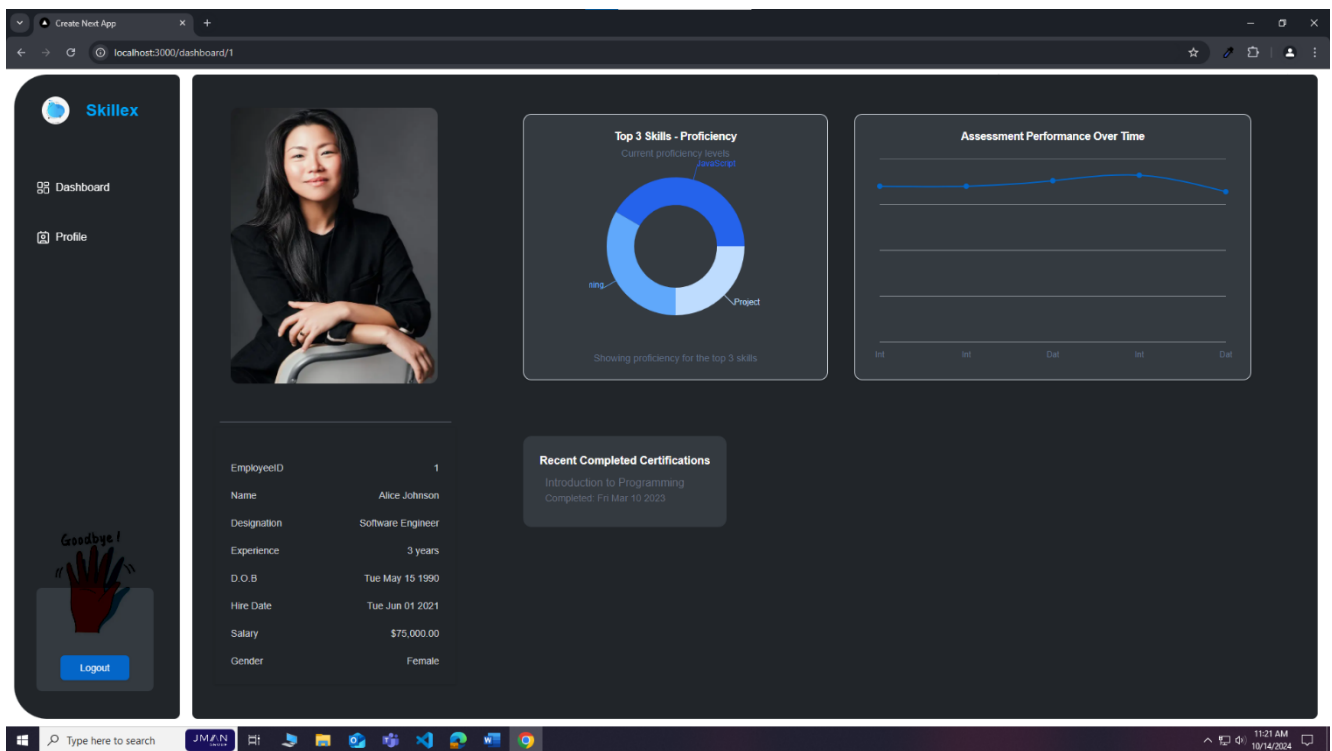


Figure 4 Employee Dashboard

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.

- **Prisma for Database Management:**

Prisma will be utilized as the ORM for managing database interactions. It streamlines data access and manipulation, providing a type-safe interface for querying and updating data within the Neon PostgreSQL database. This helps ensure that data integrity is maintained throughout the application.

- **Snowflake Integration:**

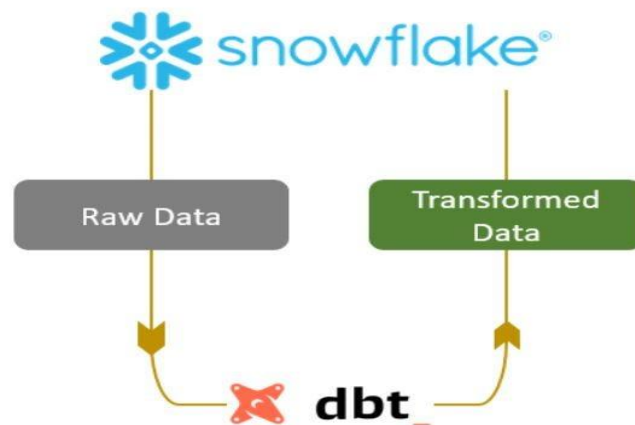
The platform will integrate Snowflake as the data warehousing solution. Data will be ingested from the Neon PostgreSQL database, with Prisma facilitating the export of relevant datasets. Snowflake's capabilities allow for the storage and processing of large volumes of data, enabling advanced analytics and reporting.

- **Data Pipeline with Next.js Server Components:**

The Next.js server components will handle backend operations and API requests for data processing. This architecture allows for efficient data retrieval and manipulation directly from the server, optimizing performance and enhancing user experience.

- **DBT for Data Transformation:**

By integrating dbt into the workflow, the process of transforming and preparing data within Snowflake will be automated and standardized. dbt will enable the creation of reliable data models and data marts, facilitating efficient analytics and reporting.



6.1.2 Approach for Machine Learning.

Exploratory Data Analysis (EDA)

- EDA is a vital first step in any data analysis or machine learning project.
- For the Employee Skill Tracking platform, EDA entails examining and summarizing key characteristics of datasets related to employee skills, training events, and engagement metrics.
- This analysis aids in uncovering the underlying structure, patterns, and relationships within the data, informing future analyses and decision-making.

Feature Extraction

- Feature extraction involves converting raw data into a set of meaningful features that serve as inputs for machine learning algorithms.
- In the context of the Employee Skill Tracking platform, relevant features might include employee skill levels, event attendance records, and other attributes indicative of employee development and engagement.
- This transformation enhances the quality of the data used for model training.

Model Development

- Model development encompasses building and training machine learning models using the pre-processed data derived from EDA and feature extraction.
- The objective is to create predictive models that deliver accurate and reliable insights on unseen data. In the Employee Skill Tracking project, this process aims to provide actionable insights that promote employee skill enhancement and engagement with training events.

Recommendation System

- A recommendation system based on cosine similarity is a widely used method, especially in content-based recommendation systems. This model considers employee skills to suggest relevant training events.
- By analysing the similarity between an employee's skills and the skills required for various training events, the system can generate personalized recommendations, thereby fostering employee development and increasing participation in training activities.

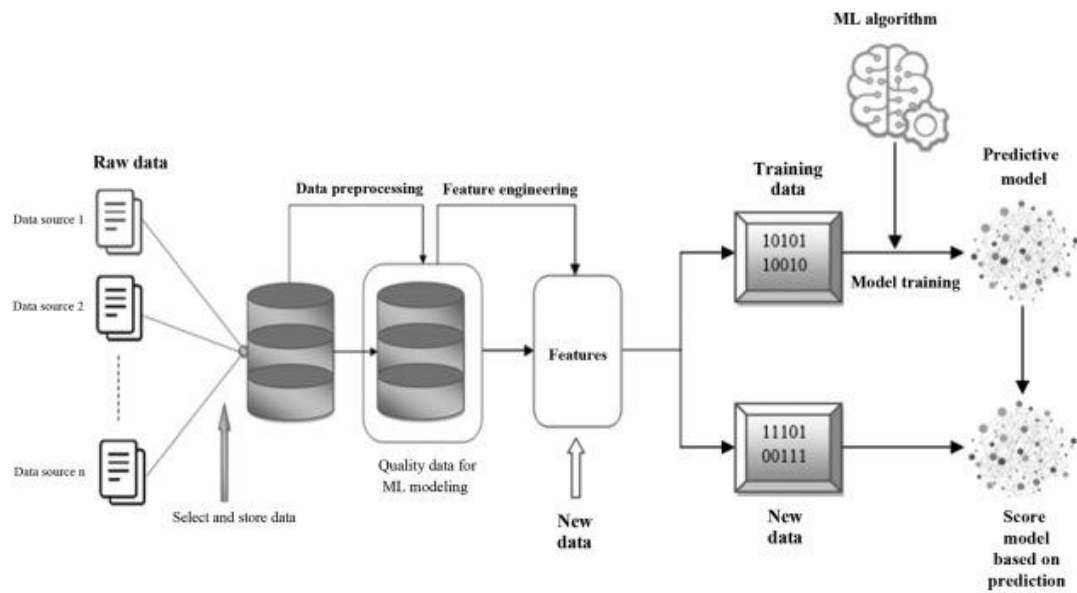


Figure 5 Machine Learning Architecture

7 Project Resources

7.1.1 Roles and responsibilities of the project team members.

Rishabh Saklani – Responsible for developing the full stack application, data engineering and data science

8 Risks and Mitigation Strategies

8.1.1 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
2	Database schema creation	Correct schema is required to work with data	Design the schema ahead of building the application

9 Improvements

9.1.1 Improvements – Full Stack.

- **Enhanced User Experience**

- Personalized Dashboards: Implement customizable dashboards for both employees and administrators. This would allow users to prioritize the information and tools that are most relevant to them, such as skill assessments, upcoming training events, and performance metrics.
- Responsive Design: Ensure that the web application is fully responsive and accessible on various devices (desktops, tablets, and mobile phones). This will enhance usability and engagement among employees who may access the platform from different devices.

2. Advanced Analytics

- Predictive Analytics: Integrate advanced predictive analytics capabilities that can forecast employee training needs based on past performance, skill gaps, and industry trends. This could help the organization proactively plan training programs.
- Skill Gap Analysis: Incorporate automated skill gap analysis tools that assess the skills of employees relative to their current roles and future career aspirations, suggesting relevant training events to bridge those gaps.

- **3. Gamification**

- Gamification Features: Introduce gamification elements such as badges, points, and leaderboards to motivate employees to participate in training events. This could enhance engagement and create a more interactive learning environment.
- Progress Tracking: Provide visual representations of individual learning paths and progress tracking for both employees and administrators. This would help users understand their development trajectory and stay motivated.

4. Integration and Collaboration

- Integration with Existing Tools: Enable integration with other HR and productivity tools (e.g., Slack, Microsoft Teams, and Learning Management Systems) to streamline communication and collaboration around training events and employee development.

- Collaborative Learning: Facilitate peer learning by allowing employees to share insights and experiences related to training events. This could be implemented through discussion boards, forums, or group projects.

5. Enhanced Feedback Mechanisms

- Real-Time Feedback: Implement real-time feedback mechanisms that allow employees to provide input on training events, helping organizers improve future offerings based on participant experiences.
- Post-Training Assessments: Incorporate post-training assessments or surveys to evaluate the effectiveness of training events. This feedback can be used to continuously improve content and delivery methods.

6. Security and Compliance

- Data Privacy and Security: Ensure that data privacy and security measures comply with relevant regulations (e.g., GDPR, CCPA) to protect employee information. Implement strong authentication and encryption protocols to safeguard sensitive data.
- Audit Trails: Create comprehensive audit trails to track changes in the system, including user activity, data access, and modifications to training events. This will enhance accountability and facilitate compliance audits.

7. Continuous Improvement and Support

- Regular Updates and Maintenance: Establish a regular schedule for software updates, maintenance, and feature enhancements based on user feedback and technological advancements.
- User Support and Resources: Provide extensive resources, such as tutorials, FAQs, and user support, to assist employees in navigating the platform and maximizing its features.

9.1.2 Improvements – Data Engineering.

- **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.