**EduVault Project Report**

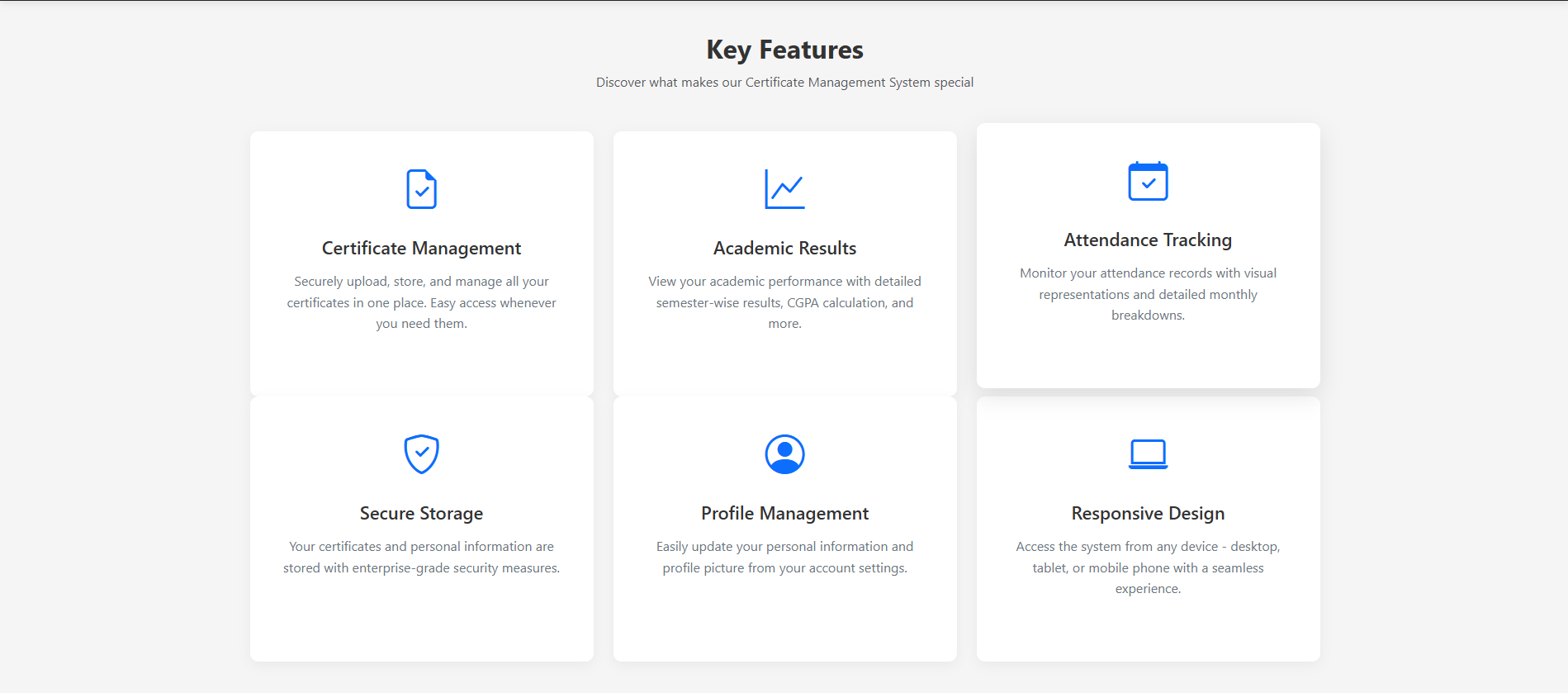
**1. Project Overview**

EduVault is a robust and comprehensive digital system meticulously designed to empower educational institutions with the efficient and secure management of critical student data. This includes academic certificates, detailed academic records (such as grades and transcripts), and precise attendance information. The primary objective of EduVault is to centralize, streamline, and automate these traditionally manual and often fragmented administrative tasks. By offering a secure, accessible, and user-friendly platform, EduVault aims to significantly reduce administrative overhead, improve data accuracy, and enhance the overall experience for both institutional staff and students. It serves as a single source of truth for student-related information, promoting transparency and accessibility.

**2. Key Features**

EduVault offers a comprehensive suite of functionalities, each engineered to enhance the administrative efficiency and improve the student experience within an educational setting:

* **User Authentication**: This core feature provides a secure and highly reliable system for both new user registration and existing user login. It employs modern authentication protocols to ensure that only authorized individuals can gain access to the platform and their respective data. This includes mechanisms for password hashing, secure session management, and potentially multi-factor authentication, safeguarding sensitive student and institutional information from unauthorized access and maintaining data integrity.
* **Certificate Management**: This module offers a complete lifecycle management solution for student certificates. It enables educational institutions to securely upload digital copies of various certificates (e.g., degree certificates, participation certificates, awards). Once uploaded, these certificates can be organized, easily viewed by authorized users, conveniently downloaded for personal records or sharing, and, when necessary, securely deleted. This digital repository eliminates the need for physical certificate storage, reduces the risk of loss or damage, and facilitates quick retrieval and verification.
* **Academic Results Integration**: A standout feature, this module provides seamless, real-time integration with external academic results APIs, specifically the JNTUH results API. This integration allows for the direct retrieval and dynamic display of students' academic performance within the EduVault platform. By automating the fetching of results, it significantly reduces manual data entry, minimizes the potential for human error, and ensures that students and administrators always have access to the most current and accurate academic records.
* **Attendance Tracking**: This functionality offers robust tools for monitoring and visualizing student attendance records. It allows for the systematic recording of student presence in classes, lectures, or other academic activities. The data is then processed and presented through intuitive visualizations, such as graphs and charts, providing valuable, at-a-glance insights into student participation, punctuality, and overall engagement. This helps in identifying attendance patterns and supporting student success initiatives.
* **Profile Management**: This feature empowers all users, including students and staff, to easily manage and update their personal information within the system. Users can modify their contact details, personal particulars, and upload or change their profile pictures. This ensures that all user data within the system is always current, accurate, and reflects the latest information, improving communication and personalization within the platform.
* **Admin Dashboard**: A dedicated, intuitive, and powerful dashboard is provided for administrators. This central control panel offers a comprehensive overview of the entire EduVault system, presenting key metrics, system status, and quick access to administrative functions. It provides centralized control and management capabilities over user accounts, data categories, system settings, and various functionalities, enabling administrators to efficiently oversee and maintain the platform.
* **Responsive Design**: The application is meticulously built with a responsive design philosophy. This means the user interface automatically adapts and optimizes its layout, content, and interactive elements to provide an optimal viewing and interactive experience across a wide array of devices. Whether accessed on a large desktop monitor, a laptop, a tablet, or a small mobile phone screen, EduVault ensures consistent usability, readability, and functionality without requiring separate versions of the application.



**3. Installation Guide: From Scratch to Setup**

To get the EduVault project up and running on your local development machine or server, follow these detailed step-by-step instructions. Each step is crucial for a successful setup.

**Step 3.1: Prerequisites**

Before you begin the installation process, ensure that your system meets the following software requirements. Having these installed beforehand will prevent common setup issues:

* **Node.js**: This is a JavaScript runtime environment that allows you to execute JavaScript code outside of a web browser. It is essential for running the EduVault backend. Node.js typically comes bundled with npm (Node Package Manager), which is used to install project dependencies. You can download the latest stable version from the official Node.js website.
* **Git**: Git is a distributed version control system used for tracking changes in source code during software development. You will use Git to clone (download) the EduVault project repository from GitHub to your local machine. Install Git from its official website or via your operating system's package manager.
* **MySQL Server**: EduVault uses MySQL as its relational database management system to store all persistent data, including user information, certificate details, and attendance records. Ensure you have a running MySQL server instance. You can download and install MySQL Community Server, or use a tool like XAMPP/WAMP/MAMP which bundles MySQL with Apache and PHP.

**Step 3.2: Clone the Repository**

This step involves downloading the entire EduVault project source code from its GitHub repository to your local machine.

1. **Open your terminal or command prompt**: Navigate to the directory where you want to store your project files (e.g., C:\Projects on Windows, or ~/Documents/Projects on macOS/Linux).
2. **Execute the clone command**: Type the following command and press Enter:
3. git clone https://github.com/Harish2B3/eduvault.git

This command will create a new directory named eduvault (or whatever the repository name is) containing all the project files.

1. **Navigate into the project directory**: Once the cloning is complete, change your current directory to the newly created project folder:
2. cd eduvault

All subsequent commands will be executed from within this directory.

**Step 3.3: Install Dependencies**

After cloning the repository, you need to install all the external libraries and packages that the EduVault project relies on. These are listed in the package.json file.

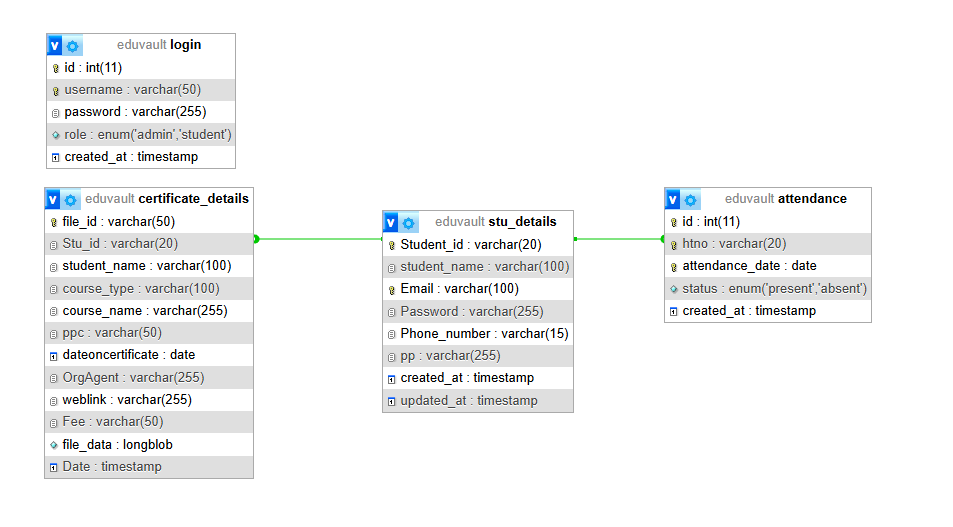
1. **Execute the npm install command**: While inside the eduvault project directory, run:
2. npm install

This command reads the dependencies and devDependencies sections of your package.json file and downloads all the necessary Node.js modules into a node\_modules folder within your project. This process may take a few moments depending on your internet connection.

**Step 3.4: Database Setup**

This final installation step involves preparing your MySQL database to store EduVault's data by creating the necessary database and importing its predefined schema.

1. **Create a MySQL Database**:
   * Open your preferred MySQL client. This could be a graphical user interface (GUI) tool like MySQL Workbench, phpMyAdmin, DataGrip, or simply the MySQL command-line client.
   * Execute the following SQL file to create database, “eduvault.sql”
   * Ensure the command executes successfully.



1. **Import Database Schema**:
   * Locate the eduvault.sql file within your cloned eduvault project directory. This file contains all the SQL commands necessary to create the tables, define their columns, set up relationships, and potentially insert initial data.
   * **Using MySQL Command Line**: If you are comfortable with the command line, navigate to the directory where eduvault.sql is located (or provide its full path) and run:
   * mysql -u your\_mysql\_username -p eduvault < eduvault.sql

Replace your\_mysql\_username with the MySQL username you used to create the database (e.g., root). You will be prompted to enter the password for this MySQL user. This command pipes the SQL file's content into the eduvault database.

* + **Using a GUI Tool (e.g., MySQL Workbench)**:
    - Open MySQL Workbench (or your chosen GUI tool) and connect to your MySQL server.
    - In the Schemas navigator, right-click on the eduvault database you just created and select "Set as Default Schema" or similar.
    - Go to "Server" > "Data Import" (or "Import/Restore" or "Execute SQL Script" depending on the tool).
    - Select the eduvault.sql file as the source for your import.
    - Ensure the target schema/database is eduvault.
    - Start the import process. This will create all the tables and populate any initial data defined in the SQL file.

**4. Execution Instructions**

Once the EduVault project has been successfully installed and configured, you can start the application server. The method you choose depends on whether you are running it for development or for general use.

* **Start the Application (Production Mode)**: This command is typically used when you want to run the application in a stable, production-like environment. It starts the server using the default Node.js execution.
* npm start

Upon execution, the terminal will usually display messages indicating that the server has started, often including the URL (e.g., Server running on http://localhost:5000) where the application can be accessed in your web browser. This mode is generally optimized for performance.

* **Start in Development Mode (with Auto-Restart)**: This mode is highly recommended for developers as it significantly enhances the development workflow. It uses a tool (like nodemon if configured in package.json) to monitor changes in your project files.
* npm run dev

When you save changes to your code, the server will automatically detect these changes and restart, allowing you to see the effects of your modifications instantly without manually stopping and starting the server. This speeds up debugging and iterative development. The output in the terminal will be similar to npm start, indicating the server's status and access URL.

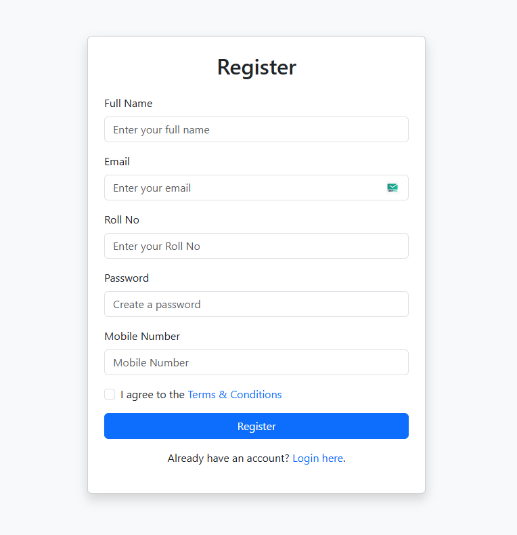
**5. User Interaction: Registration and Login**

After successfully executing the EduVault application, it will be accessible via your web browser. The exact URL will depend on the PORT configured in your .env file (e.g., http://localhost:3000 or http://localhost:5000). The initial interaction for new users involves registration, followed by the login process for subsequent access.

**Step 5.1: Registration**

The registration process allows new users to create an account within the EduVault system, granting them personalized access to its features.

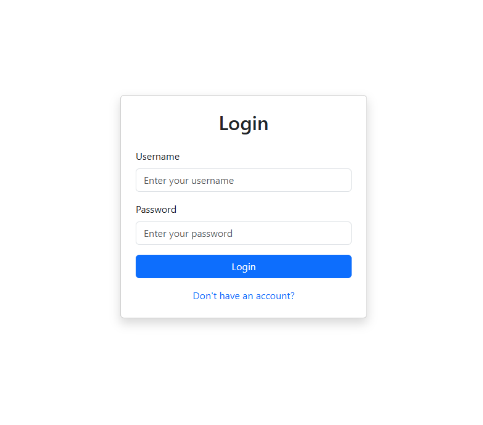
1. **Access the Application**: Open your preferred web browser (e.g., Chrome, Firefox, Edge) and enter the application's URL (e.g., http://localhost:3000). This will typically lead you to the application's landing page or a default public view.
2. **Locate Registration**: On the landing page or navigation bar, actively look for a prominent "Register," "Sign Up," or "Create Account" link or button. Click on it to proceed to the registration form.
3. **Fill Registration Form**: You will be presented with a form requiring specific personal details. Carefully enter the following information:
   * **Full Name**: Your complete name.
   * **Email Address**: A valid and unique email address, which will often serve as your primary identifier for logging in.
   * **Password**: Choose a strong, unique password. It's crucial to use a combination of uppercase and lowercase letters, numbers, and special characters.
   * **Confirm Password**: Re-enter your chosen password exactly to ensure there are no typos.
   * **Any other required user-specific information**: Depending on the system's configuration, you might also be asked for details like a Student ID, your role (e.g., "Student," "Faculty"), or other demographic information.
4. **Submit**: After carefully reviewing all the entered details for accuracy, click the "Register" or "Sign Up" button. Upon successful submission, you should receive a confirmation message (e.g., "Registration Successful!") or be automatically redirected to the login page, prompting you to log in with your newly created credentials.



**Step 5.2: Login**

The login process allows registered users to securely access their personalized dashboard and the features tailored to their role within EduVault.

1. **Access the Application**: If you are not already on the application's page, open your web browser and navigate to the application's URL (e.g., http://localhost:3000).
2. **Locate Login**: On the landing page or navigation bar, identify and click on the "Login" or "Sign In" link or button. This will take you to the login form.
3. **Enter Credentials**: In the provided fields, input your registered email address (or username, if the system uses usernames) and the password you set during registration. Ensure there are no typos.
4. **Submit**: Click the "Login" or "Sign In" button. The system will then authenticate your credentials against its database. Upon successful authentication, you will be redirected to your personalized user dashboard or the main application interface, granting you access to your specific functionalities. If authentication fails, an error message (e.g., "Invalid credentials") will typically be displayed.



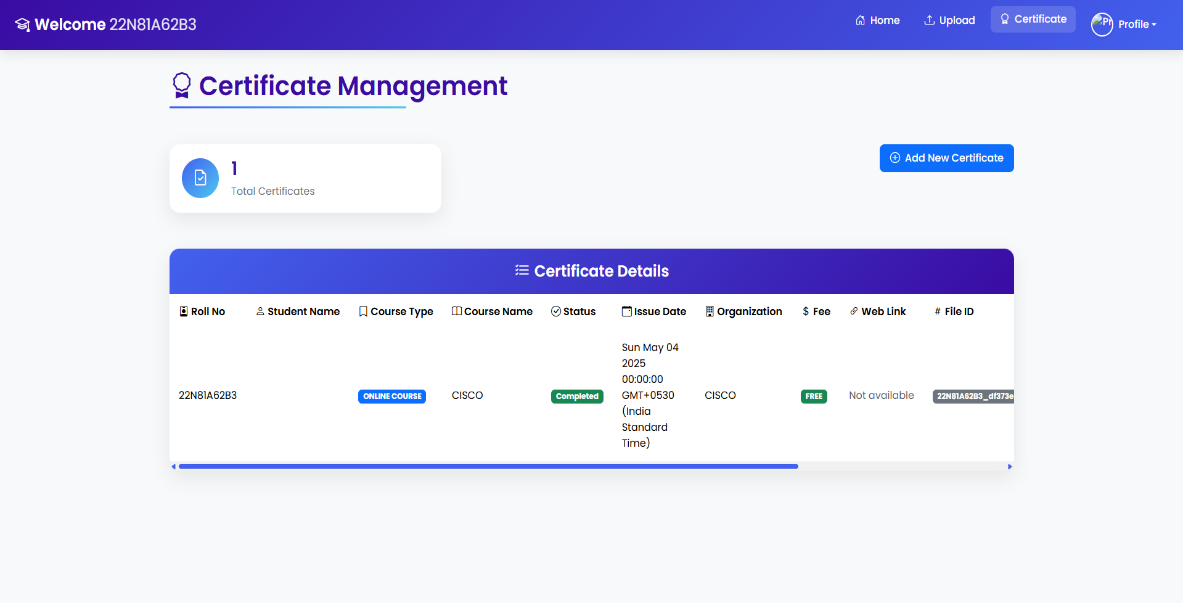
**6. Post-Login Functionalities: What You Can Do**

After successfully logging into EduVault, users gain access to a range of features and modules, the specific functionalities of which are determined by their assigned role within the system (e.g., student, administrator, faculty). Here's a detailed breakdown of the common functionalities available:

**6.1: For Students/General Users**

Students and general users typically have access to features that allow them to manage their personal academic and administrative information:

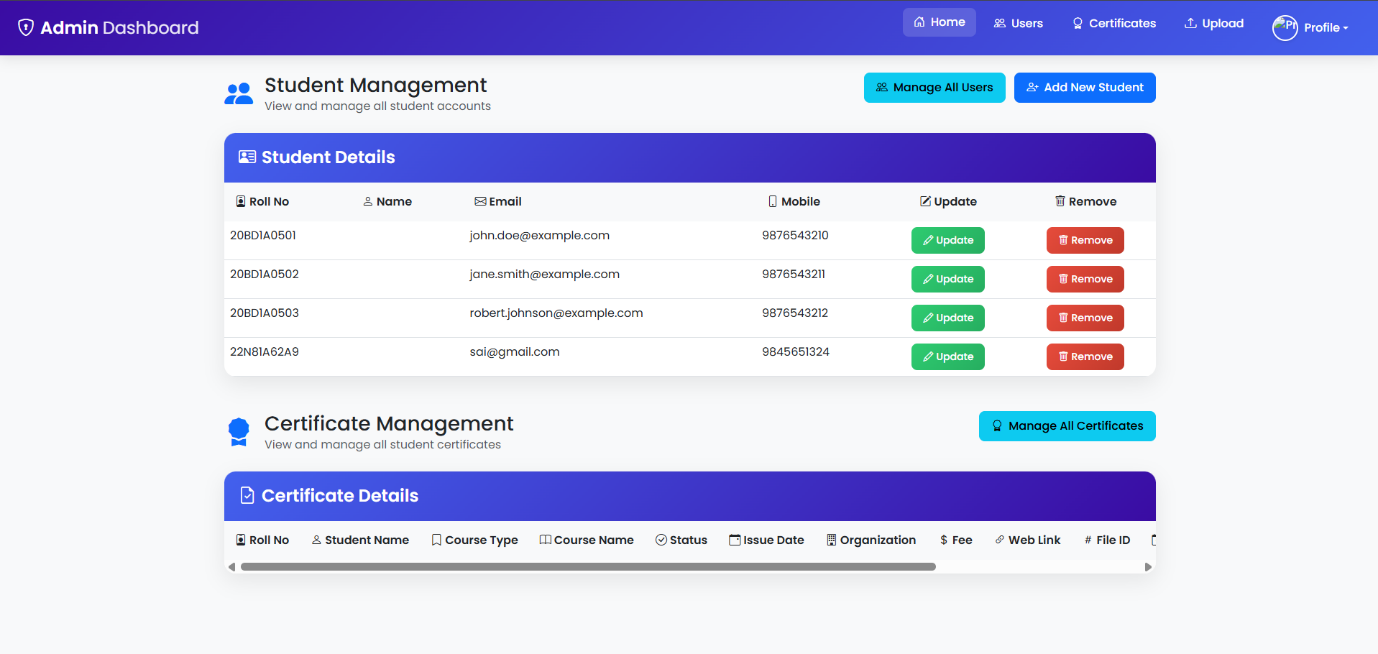
* **Manage Certificates**: This comprehensive sub-module allows students to interact with their digital certificates:
  + **Upload**: Students can securely upload new digital copies of their academic achievements (e.g., degree certificates, course completion certificates) or extracurricular awards. The system may support various file formats (e.g., PDF, JPEG).
  + **View**: A dedicated interface allows students to browse and view all their previously uploaded certificates in an organized manner, often with thumbnails or a list view for easy navigation.
  + **Download**: Students have the ability to download any of their uploaded certificates. This is useful for sharing with potential employers, for personal records, or for printing physical copies.
  + **Delete**: In cases of accidental uploads or outdated documents, students can securely remove certificates from their digital vault.
* **View Academic Results**: This section provides direct and up-to-date access to a student's academic performance. Leveraging the JNTUH results API integration, students can view their grades, marks, and overall academic standing for various semesters or courses. The data is presented clearly, often with options to filter by academic year or program.
* **Track Attendance**: This module offers students a transparent view of their attendance records. It provides detailed information on their presence in classes, lectures, or other mandatory academic sessions. The data is often presented with intuitive visualizations (e.g., bar charts showing attendance percentage, line graphs showing trends) to give a clear and immediate overview of their participation and punctuality.
* **Update Profile**: Students can personalize and maintain their personal information through this feature. They can navigate to their profile section to:
  + Update contact details (e.g., phone number, secondary email).
  + Modify personal particulars (e.g., address).
  + Upload or change their profile picture, allowing for a personalized user experience within the platform.



**6.2: For Administrators (via Admin Dashboard)**

Administrators, with their elevated privileges, gain access to a specialized and powerful dashboard that provides comprehensive control and oversight over the entire EduVault system:

* **User Management**: This core administrative function allows authorized personnel to:
  + **Add New Users**: Create new accounts for students, faculty, or other staff members.
  + **Edit Existing Profiles**: Modify user details, roles, or permissions as needed.
  + **Deactivate/Activate Accounts**: Temporarily suspend or reactivate user accounts, for instance, in cases of leave of absence or disciplinary actions.
  + **Reset Passwords**: Assist users by resetting their passwords if they are locked out or forget their credentials.



* **Content Management**: Administrators can oversee and manage the various types of content stored within the system, ensuring data accuracy and consistency. This includes:
  + Managing academic record structures.
  + Defining and categorizing different types of certificates.
  + Ensuring data integrity across all stored information.
* **System Analytics**: The admin dashboard often provides valuable insights into the system's usage and health. Administrators can view:
  + Overall system statistics (e.g., number of registered users, total certificates uploaded).
  + User activity logs (e.g., login times, actions performed), which can be crucial for auditing and security.
  + Other relevant data points to monitor the performance, stability, and overall health of the platform.
* **Configuration**: Administrators have the ability to adjust various system settings and configurations. This might include:
  + Setting up integration parameters for external APIs (like JNTUH results).
  + Managing notification settings.
  + Configuring user roles and permissions.
  + Performing system maintenance tasks.

This detailed report provides an exhaustive explanation of your EduVault project, covering its core purpose, extensive features, meticulous installation procedures, execution methods, user registration and login workflows, and the various functionalities available post-login for different user roles.