B.M.S. COLLEGE OF ENGINEERING BENGALURU
Autonomous Institute, Affiliated to VTU

Object Oriented Modeling Mini Project Report

Alumni Association platform

Submitted in partial fulfillment for the award of degree of

Bachelor of Engineering
in
Computer Science and Engineering

Submitted by:

Rohit Ramchandra Gandhi **1BM23CS417**
S Gajanana Nayak **1BM22CS227**
Rishabh Kumar **1BM22CS221**

Department of Computer Science and Engineering
B.M.S. College of Engineering
Bull Temple Road, Basavanagudi, Bangalore 560 019
2024-2025

B.M.S. COLLEGE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We, AAAA (1BM21CS001) students of 5th Semester, B.E, Department of Computer Science and Engineering, BMS College of Engineering, Bangalore,hereby declare that, this Object Oriented Modeling Mini Project entitled " ***Alumni Association platform***" has been carried out in Department of CSE, B.M.S. College of Engineering, Bangalore during the academic semester September 2024 -January 2025. I also declare that to the best of our knowledge and belief, the Object Oriented Modeling(23CS5PCOOM) mini Project report is not from part of any other report by any other students.

Signature of the Candidate

Rohit Ramchandra Gandhi (1BM23CS417)
S Gajanana Nayak (1BM22CS227)
Rishabh Kumar (1BM22CS221)

B.M.S. COLLEGE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the OOMD Mini Project titled "*Alumni Association platform***" has been carried out by Rohit Ramchandra Gandhi (1BM23CS417), S Gajanana Nayak(1BM22CS227), Rishabh Kumar (1BM22CS221)**

during the academic year 2023-2024.

Signature of the Faculty in Charge

Table of Contents

Problem Statement

Problem Statement Title: Implementation of the Alumni Association platform for the University/Institute.

Problem Statement ID:1609

Description: A comprehensive Alumni Association platform for a University/Institute, encompassing web applications, aims to address challenges effectively. Detailed Description: The proposed Alumni Association platform for the Government Engineering College will feature robust functionalities accessible through web applications: 1. Alumni Registration: User-friendly registration processes on web platforms, allowing alumni to join the association, update their profiles, and stay connected with peers and the institution.2. Donation Portal: Secure mechanisms on web platform for alumni to contribute donations easily, fostering a culture of philanthropy. 3. Networking Hub: Dedicated sections on both platforms to connect alumni based on shared interests, professions, and geographic locations, facilitating professional networking, mentorship, and collaboration opportunities. 4. Job Portal: Integrated job search and posting features accessible via web apps, enabling alumni to explore career opportunities, post job openings, and connect with potential employers within the alumni network. 5.Alumni Directory: Search functionalities available on web platforms to find alumni based on different criteria such as graduation year, field of study, industry, location, etc., 6. Success Story Tracking: Features on web apps to showcase and track alumni achievements, success stories, and notable contributions to society, inspiring current students and fostering pride among alumni. 7.Events and Reunions: Announcements, registrations, and management tools available on both platforms for organizing alumni events, reunions, workshops, and professional development sessions to maintain engagement and connection. 8.Feedback and Surveys: Channels on both web applications for alumni to provide feedback on their experiences. The platform will prioritize user experience, security, and scalability across both web and mobile applications to cater to the diverse needs of the Government Engineering College's alumni community. Expected Solution: Implementation of the Alumni Association platform for the Government Engineering College, comprising both web and mobile applications, is expected to achieve several positive outcomes: Enhanced Alumni Engagement: Seamless access to networking, career opportunities, and alumni events through web applications will strengthen connections among alumni, fostering a vibrant and active community. 9.Increased Philanthropic Support: Convenient donation processes accessible via web platform will encourage alumni to contribute towards the college's growth and development initiatives. 10.Career Advancement: Access to job postings, mentorship opportunities, and professional networking on website will support alumni in their career growth and advancement. 11.Knowledge Sharing: Exchange of knowledge, experiences, and best practices facilitated through both web and mobile apps will enrich professional development and lifelong learning initiatives. 12.Community Building: Interactive features available on web apps will nurture a sense of belonging and camaraderie among alumni, strengthening their bond with the institution. In summary, the Alumni Association platform for the University/Institute, integrated with web applications, aims to create a dynamic and supportive ecosystem where alumni can connect, contribute, and thrive, thereby enriching the overall educational experience.

Software Requirement Specification

1. Introduction

1.1 Purpose

The purpose of this document is to define the software requirements for the Alumni Association platform for the Government Engineering College. This platform's available in web application, will provide a centralized system to foster engagement among alumni, facilitate donations, enable networking opportunities, and showcase success stories, among other functionalities.

1.2 Scope

The Alumni Association platform will feature functionalities like alumni registration, donation portals, networking hubs, job postings, and event management. The system is intended to enhance alumni engagement, support philanthropic contributions, and promote career advancement through a user-friendly, secure, and scalable solution. web applications will ensure accessibility and convenience for a diverse user base.

1.3 Definitions, Acronyms, and Abbreviations

- **Alumni**: Graduates of the Government Engineering College.
- **SRS**: Software Requirements Specification.
- **UI**: User Interface..
- **CRUD**: Create, Read, Update, Delete.
- **Portal**: A dedicated section within the platform for a specific purpose, e.g., Donation Portal.

1.4 References

- IEEE Recommended Practice for Software Requirements Specifications (IEEE Std 830-1998).
- https://sih.gov.in/sih2024PS ,Problem statement ID 1609

1.5 Overview

This document outlines the functional, non-functional, and technical requirements of the Alumni Association platform. It includes system features, user interactions, and external interfaces necessary for successful implementation.

---

2. Overall Description

2.1 Product Perspective

The Alumni Association platform is a new system designed to enhance alumni engagement by integrating web and mobile applications. It will replace manual and fragmented processes with a unified, automated solution that supports multiple functionalities for alumni interaction and contribution.

2.2 Product Features

- **Alumni Registration**: User-friendly onboarding process.
- **Donation Portal**: Secure and transparent donation mechanisms.
- **Networking Hub**: Connect alumni based on commonalities.
- **Job Portal**: Enable job search and postings.
- **Alumni Directory**: Search alumni by various criteria.
- **Success Story Tracking**: Highlight and track achievements.
- **Events and Reunions**: Manage announcements and registrations.
- **Feedback and Surveys**: Channels for user input and suggestions.

2.3 User Classes and Characteristics

- **Alumni**: Graduates with varying levels of technical expertise.
- **Administrators**: College personnel managing the platform.
- **Donors**: Alumni or external contributors interested in donating.
- **Event Organizers**: Personnel or volunteers organizing alumni events.

2.4 Operating Environment

- **Web Application**: Compatible with modern browsers (Chrome, Firefox, Safari, Edge).
- **Hosting**: Cloud-based infrastructure ensuring scalability and reliability.

2.5 Design and Implementation Constraints

- Compliance with data protection laws (e.g., GDPR).
- Platform must handle high traffic during peak events (e.g., reunions).

2.6 Assumptions and Dependencies

- Reliable internet connectivity for accessing the platform.
- Alumni database provided by the college.
- Integration with third-party payment gateways for donations.

---

3. Specific Requirements

3.1 Functional Requirements

3.1.1 Alumni Registration

- Alumni can register using a simple form.
- CRUD operations for user profiles.
- Email and mobile number verification.

3.1.2 Donation Portal

- Secure payment gateway integration.
- Donation tracking and reporting.
- Automated tax receipts for donors.

3.1.3 Networking Hub

- Filters for connecting alumni based on profession, location, and interests.
- Messaging functionality.

3.1.4 Job Portal

- Alumni can post and search for jobs.
- Notifications for new job postings.

3.1.5 Alumni Directory

- Searchable database of alumni profiles.
- Filters for graduation year, industry, and more.

3.1.6 Success Story Tracking

- Submission form for alumni to share stories.
- Administrator approval workflow.

3.1.7 Events and Reunions

- Event creation and registration system.
- Notifications and reminders for events.

3.1.8 Feedback and Surveys

- Feedback submission forms.
- Survey creation and participation tracking.

3.2 Non-Functional Requirements

3.2.1 Performance Requirements

- The platform should support up to 10,000 simultaneous users.
- Response time for actions must be under 2 seconds.

3.2.2 Security Requirements

- All data transmission must use HTTPS.
- Multi-factor authentication for administrators.
- Data encryption for sensitive information.

3.2.3 Usability Requirements

- Intuitive and accessible UI.
- Support for multiple languages.

3.2.4 Scalability

- The system should scale seamlessly to accommodate growing user bases.

4. External Interface Requirements

4.1 User Interfaces

- Web: Responsive design compatible with desktops and tablets.

4.2 Hardware Interfaces

- Standard computing devices for web access.
- Smartphones with Android or iOS for mobile access.

4.3 Software Interfaces

- Integration with third-party APIs for payment gateways and messaging services.
- Compatibility with the college's existing systems.

4.4 Communication Interfaces

- Email and push notifications for user alerts.
- SMS integration for critical updates.

---

5. Other Non-Functional Requirements

5.1 Legal and Regulatory Requirements

- Adherence to data privacy laws (GDPR, CCPA).
- Accessibility compliance (WCAG 2.1).

5.2 Ethical Considerations

- Transparent handling of alumni data.
- Ethical fundraising practices.

---

6. Appendices

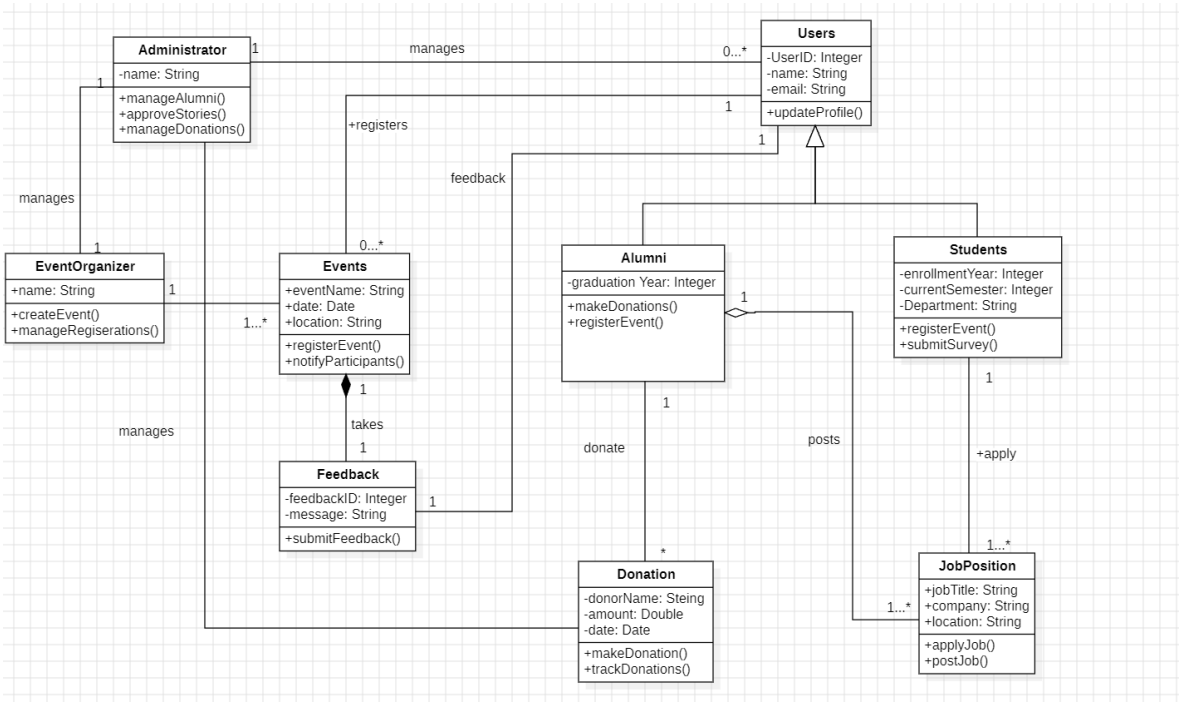
6.1 Glossary

- **GDPR**: General Data Protection Regulation.
- **CCPA**: California Consumer Privacy Act.
- **WCAG**: Web Content Accessibility Guidelines.

6.2 References

- https://sih.gov.in/sih2024PS ,Problem Statement ID 1609
- IEEE Standards documentation.

Class Modeling



Class Model 1.1

1. Users

Description:
This is the parent class that generalizes common attributes and methods for **Alumni** and **Students**. It ensures that both user types share a consistent structure for managing profiles and basic operations.

Attributes:

- user ID: Unique identifier for each user.
- name: Name of the user.
- email: Contact email for the user.

Methods:

- updateProfile(): Enables users to modify their profile details.

Relevance:
The **Users** class reduces redundancy by grouping shared functionality of Alumni and Students, making the system scalable and easier to maintain.

---

2. Alumni

Description:
A subclass of **Users** tailored for alumni, containing additional attributes and methods specific to graduates.

Attributes:

- graduationYear: The year the alumni graduated.

Methods:

- registerEvent(): Allows alumni to register for events.
- makeDonation(): Facilitates financial contributions to the institution.

Relevance:
This class provides targeted functionality for alumni, such as event participation and donations, which are essential for their engagement and support.

---

3. Students

Description:
A subclass of **Users** designed for current students, focusing on their active participation in events and surveys.

Attributes:

- enrollmentYear: The year the student enrolled in the institution.
- currentSemester: The student's current academic semester.

Methods:

- registerEvent(): Enables students to join events.
- submitSurvey(): Allows students to provide feedback or participate in surveys.

Relevance:
Students form a distinct user base with their own needs, such as interacting with alumni and providing feedback to improve the system.

---

4. Administrator

Description:
Handles backend operations, including managing users, approving success stories, and monitoring donations.

Attributes:

- name: Name of the administrator.

Methods:

- manageAlumni(): Adds or updates alumni profiles.
- approveStories(): Reviews and approves alumni-submitted success stories.
- manageDonations(): Tracks and reports on donation activity.

Relevance:
Administrators are crucial for maintaining the integrity and smooth functioning of the platform.

---

5. Event

Description:
Represents an alumni or student event. Events help foster community and engagement.

Attributes:

- eventName: Name of the event.
- date: Date of the event.
- location: Venue or platform for the event.

Methods:

- registerEvent(): Allows users to register for the event.
- notifyParticipants(): Sends notifications to registered users.

Relevance:
Events are a key engagement tool to bring alumni and students together, promoting networking and collaboration.

---

6. JobPosting

Description:
Manages job opportunities posted by alumni or administrators, helping other alumni and students.

Attributes:

- jobTitle: Title of the job position.
- company: Company offering the position.
- location: Job location.

Methods:

- applyJob(): Allows users to apply for jobs.
- postJob(): Lets users or administrators post job openings.

Relevance:
Supports career advancement and networking among alumni and students, fostering professional growth.

---

7. Donation

Description:
Facilitates financial contributions from alumni and other donors.

Attributes:

- donorName: Name of the donor.
- amount: Amount donated.
- date: Date of donation.

Methods:

- makeDonation(): Processes donation transactions.
- trackDonations(): Tracks and reports donation history.

Relevance:
Donations help sustain the institution's initiatives and projects, making it a vital feature of the platform.

---

8. Feedback

Description:
Captures input and suggestions from users to improve the platform and its services.

Attributes:

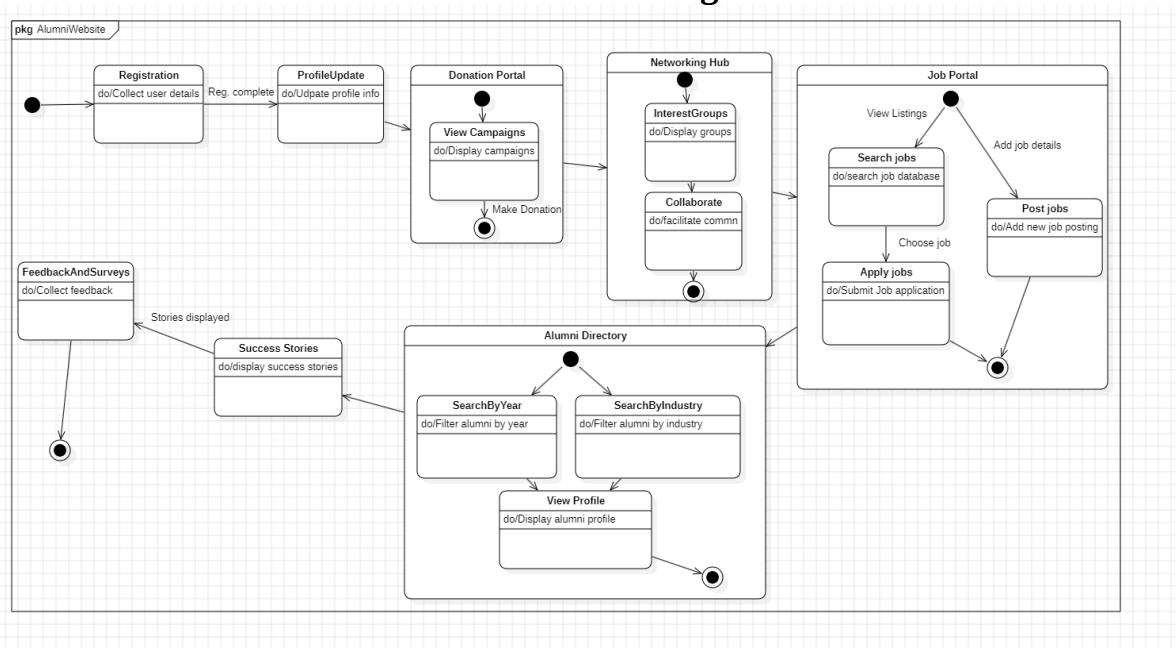
- feedbackID: Unique identifier for each feedback.
- message: The content of the feedback.

Methods:

- submitFeedback(): Allows users to provide feedback.

Relevance:
Feedback ensures continuous improvement of the platform based on user needs and experiences.

State Modeling



State Diagram 2.1

States:

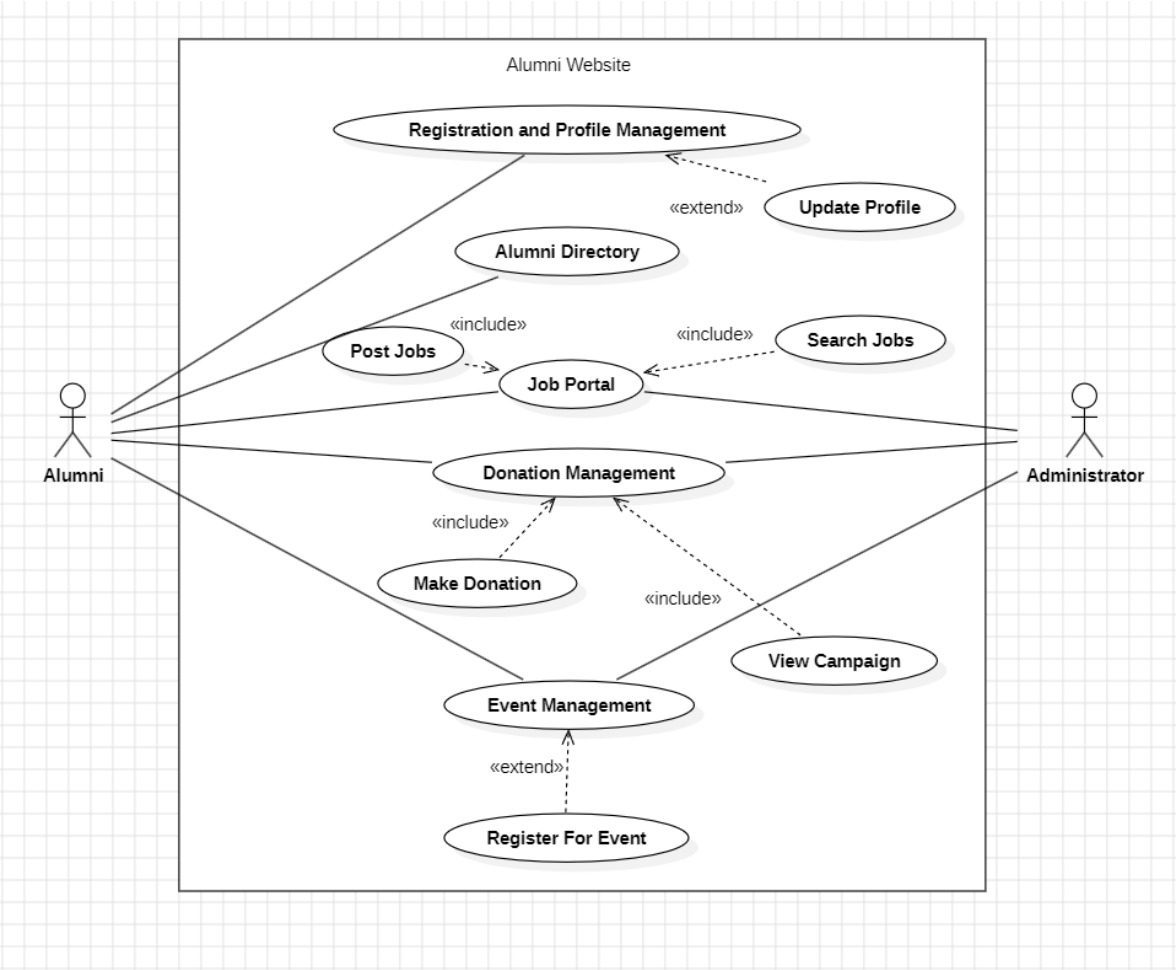
1. **Registration:**
   - **Relevance**: This is the initial state where new users provide their details to sign up for the platform. It ensures that all users are registered before accessing the system's features.
2. **Profile Update:**
   - **Relevance**: After registration, users can update their profiles, such as adding more details or modifying existing information. This keeps user data accurate and current.
3. **Donation Portal:**
   - **Relevance**: Represents the state where users interact with the donation system. Users can view donation campaigns and make contributions, supporting the institution's fundraising efforts.
4. **Networking Hub:**
   - **Relevance**: This state focuses on fostering connections among alumni through interest groups and collaboration features. It facilitates communication and engagement.
5. **Job Portal:**
   - **Relevance**: Allows users to interact with job-related functionalities, including searching for job listings, posting new opportunities, and applying for jobs. This state supports career development for alumni and students.
6. **Alumni Directory:**
   - **Relevance**: Provides a searchable database of alumni. Users can filter alumni by graduation year or industry and view detailed profiles, promoting networking and professional connections.

7. **Feedback and Surveys:**
    - **Relevance**: Enables users to provide feedback or participate in surveys, ensuring continuous improvement of the platform and collecting insights from the alumni community.
8. **Success Stories:**
    - **Relevance**: Displays inspiring achievements of alumni. This state highlights the institution's success and motivates other users to contribute and participate actively.

Events:

1. **Registration Complete (Reg. complete):**
    - **Relevance**: Marks the transition from the registration process to the ability to update profiles, signifying a successful onboarding of new users.
2. **Make Donation:**
    - **Relevance**: Indicates the user has contributed to a campaign, transitioning them to a state where their action supports the institution.
3. **View Listings:**
    - **Relevance**: Represents the action of accessing job opportunities posted on the platform, helping users explore relevant roles.
4. **Add Job Details:**
    - **Relevance**: Marks the transition where a user posts a new job opportunity, contributing to the job portal.
5. **Choose Job:**
    - **Relevance**: Indicates a user has selected a job listing to apply for, moving to the application process.
6. **Stories Displayed:**
    - **Relevance**: Indicates that success stories are being shown to users, inspiring and engaging the community.
7. **Apply Jobs:**
    - **Relevance**: Marks the event where a user submits an application for a selected job, progressing their career development.

Interaction Modeling



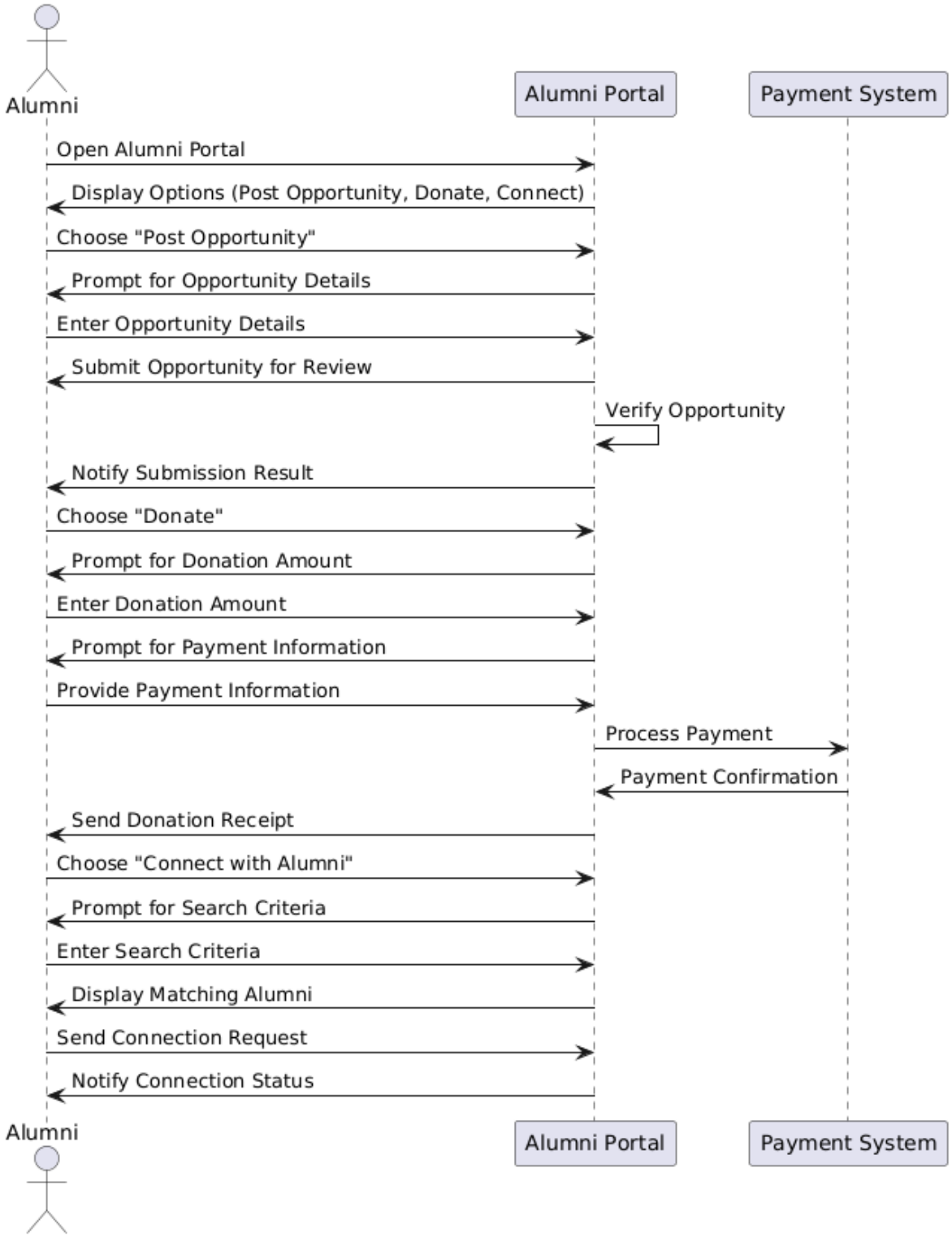
Use Case Diagram 3.1

Actors :

1. **Alumni:**
   - **Relevance**:
     Alumni are the primary users of the platform, engaging with features like registration, profile management, job postings, donations, event registrations, and networking. They represent the core audience whose participation drives the platform's success.
2. **Administrator:**
   - **Relevance**:
     Administrators manage the platform's functionality, oversee data integrity, curate events, monitor donations, and ensure smooth operation. Their role ensures the platform remains reliable and user-friendly.

Use Cases:

1. **Registration and Profile Management:**
    - **Relevance**:
      This use case ensures that alumni can register, create, and manage their profiles on the platform. Accurate and updated profiles are essential for networking, job applications, and other functionalities.
2. **Update Profile** (Extends: Registration and Profile Management):
    - **Relevance**:
      This extended use case allows alumni to modify their profiles post-registration, ensuring their information stays current for job searches and directory access.
3. **Alumni Directory:**
    - **Relevance**:
      Facilitates networking by enabling alumni to search for and connect with peers based on various filters like graduation year, industry, and location.
4. **Job Portal:**
    - **Relevance**:
      Enables alumni to interact with job-related functionalities, including searching for jobs, applying for positions, and posting job opportunities, supporting career development.
5. **Search Jobs** (Include: Job Portal):
    - **Relevance**:
      Provides alumni with access to available job listings, ensuring they can find relevant career opportunities efficiently.
6. **Post Jobs** (Include: Job Portal):
    - **Relevance**:
      Allows alumni to share job opportunities with their peers, fostering a supportive community and aiding professional growth.
7. **Donation Management:**
    - **Relevance**:
      This use case supports philanthropic contributions, enabling alumni to make donations and administrators to manage campaigns.
8. **Make Donation** (Include: Donation Management):
    - **Relevance**:
      Enables alumni to financially support the institution through an easy and secure mechanism.
9. **View Campaign** (Include: Donation Management):
    - **Relevance**:
      Allows users to explore ongoing donation campaigns and choose initiatives they wish to support.
10. **Event Management:**
    - **Relevance**:
      Facilitates the organization and management of alumni events and reunions, keeping the community engaged.
11. **Register for Event** (Extend: Event Management):
    - **Relevance**:
      Enables alumni to sign up for specific events, streamlining participation and fostering connections through organized gatherings.



Sequence Diagram 3.2

Key Components:

1. **Actors/Entities**:
    - **Alumni**: The primary user interacting with the system.

- **Alumni Portal**: The platform facilitating various services (e.g., posting opportunities, donating, and connecting).
- **Payment System**: A third-party or integrated system for handling payment transactions.

2. **Lifelines**:
   - Vertical dashed lines represent the lifespan of each entity during the interaction.
   - Arrows represent the flow of actions/messages between the entities.
3. **Interactions**: These are categorized into three main activities:
   - Posting an Opportunity
   - Making a Donation
   - Connecting with Alumni

Detailed Flow Explanation:

***1*. Posting an Opportunity:**

- **Step 1**: The **Alumni** opens the Alumni Portal.
- **Step 2**: The portal displays options like "Post Opportunity," "Donate," and "Connect."
- **Step 3**: The **Alumni** selects the "Post Opportunity" option.
- **Step 4**: The system prompts the **Alumni** to enter opportunity details.
- **Step 5**: The **Alumni** submits the opportunity for review.
- **Step 6**: The **Alumni Portal** sends the opportunity details for verification.
- **Step 7**: The system notifies the **Alumni** about the submission result (e.g., success or rejection).

---

2. Making a Donation:

- **Step 1**: The **Alumni** chooses the "Donate" option in the portal.
- **Step 2**: The portal prompts the **Alumni** to enter the donation amount.
- **Step 3**: The **Alumni** provides the donation amount.
- **Step 4**: The system prompts for payment information.
- **Step 5**: The **Alumni** enters payment details.
- **Step 6**: The **Alumni Portal** forwards the payment information to the **Payment System**.
- **Step 7**: The **Payment System** processes the payment and sends a confirmation to the portal.
- **Step 8**: The **Alumni** receives a donation receipt.

---

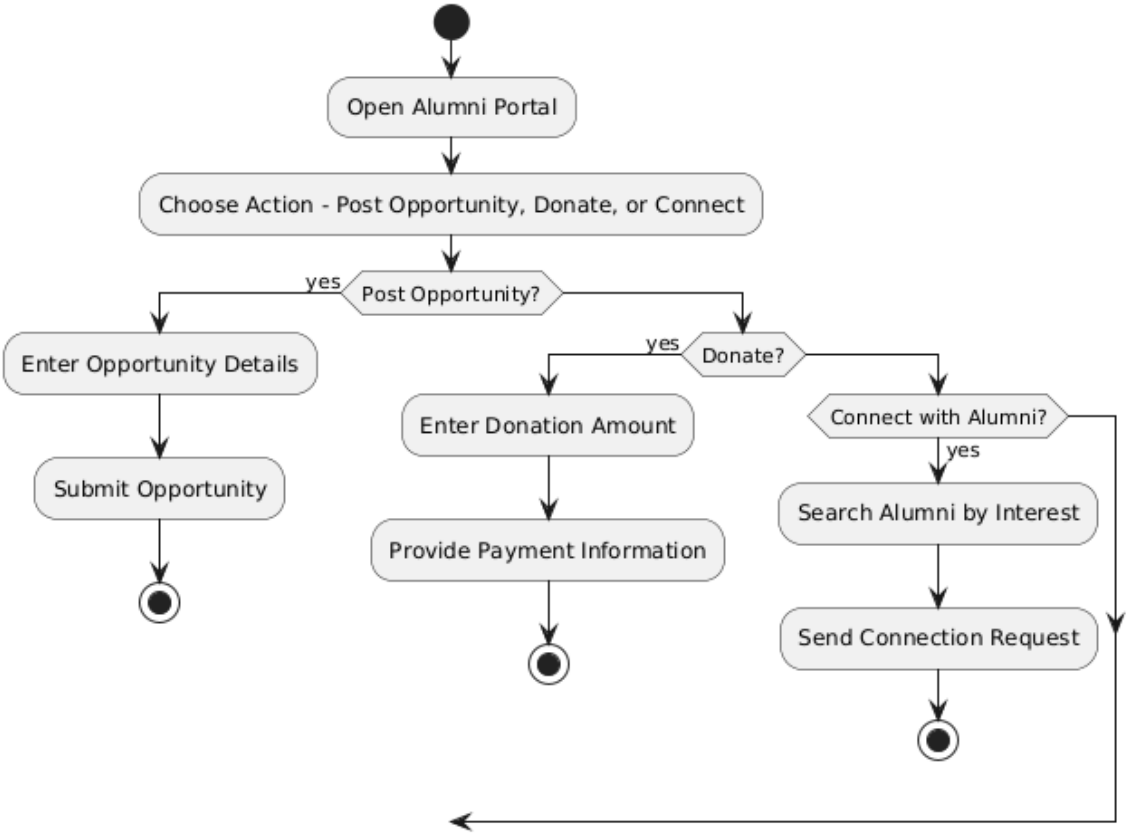
*3***. Connecting with Alumni**:

- **Step 1**: The **Alumni** selects the "Connect with Alumni" option.
- **Step 2**: The portal prompts the **Alumni** to enter search criteria (e.g., location, graduation year, profession).
- **Step 3**: The **Alumni** provides the search criteria.
- **Step 4**: The system displays a list of matching alumni.
- **Step 5**: The **Alumni** sends a connection request to a selected match.
- **Step 6**: The system notifies the **Alumni** about the connection status (e.g., accepted, pending, rejected).

Key Points:

- The **Alumni Portal** acts as a mediator between the **Alumni** and other processes (opportunity verification and payment processing).
- Arrows indicate both requests (outgoing) and responses (incoming), ensuring clear communication.
- The **Payment System** only gets involved during the donation process, indicating modular design.

This sequence diagram effectively models the flow for an Alumni Portal system by dividing its functionalities into discrete, manageable processes.



Activity Diagram 3.4

Key Features of the Diagram:

1. **Swimlanes**:
    - Although there are no explicit swimlanes in this diagram, the actions can be logically grouped by functionality:
        - **Post Opportunity** actions.
        - **Donation** actions.
        - **Connection** actions.
2. **Start and End Points**:

- **Start Point**: Represented by the black filled circle at the top, indicating the beginning of the process.
- **End Points**: Represented by black circles with an outline, indicating the completion of each workflow path.

3. **Decision Nodes**:
    - Represented by diamond shapes, these nodes split the control flow based on user decisions:
        - *"Post Opportunity?"*: Determines whether the user wants to proceed with posting an opportunity.
        - *"Donate?"*: Directs the user to the donation flow if selected.
        - *"Connect with Alumni?"*: Routes the user to the connection flow if chosen.
4. **Activity Nodes**:
    - Represented by rounded rectangles, these describe specific actions or tasks performed by the user or the system.
5. **Control Flow**:
    - Solid arrows indicate the direction of the process flow.
    - The flow splits at decision nodes (diamonds) and merges back into a single path after completing each branch.

Detailed Workflow Explanation:

1. **Start**:
    - The process begins when the user opens the Alumni Portal.
2. **Choose an Action**:
    - The user is presented with three options: **Post Opportunity**, **Donate**, or **Connect with Alumni**.
3. **Post Opportunity Path**:
    - If the user selects "Post Opportunity":
        - They are prompted to enter the opportunity details.
        - Once details are entered, the user submits the opportunity.
        - This concludes the posting process.
4. **Donation Path**:
    - If the user selects "Donate":
        - They are prompted to enter the donation amount.
        - After providing the donation amount, they proceed to payment information.
        - Completing the payment concludes the donation process.
5. **Connect with Alumni Path**:
    - If the user selects "Connect with Alumni":
        - They are prompted to search for alumni based on interest or criteria.
        - The user sends a connection request to the chosen alumni.
        - This concludes the connection process.
6. **End**:
    - Each path ends independently after completing the respective task.

Splitting and Merging Control:

- **Splitting**:
  - The control flow splits at the decision nodes, directing the user to one of the three paths based on their choice.
- **Merging**:
  - Each path merges back into a single endpoint upon completing its specific workflow, ensuring the process terminates neatly.

UI Design with Screenshots

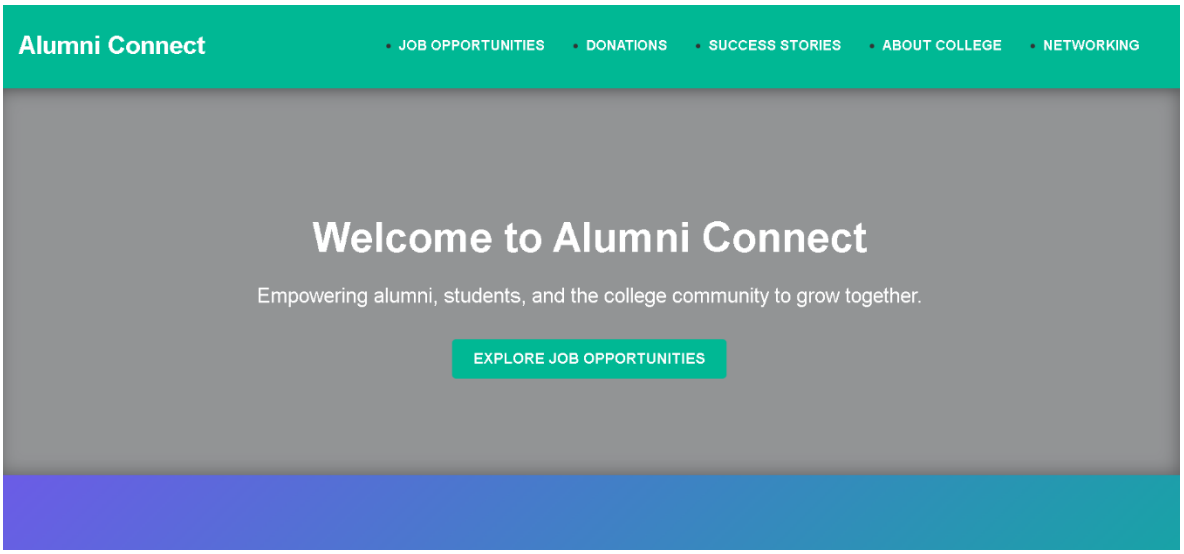


Diagram No: 4.1

Dashboard

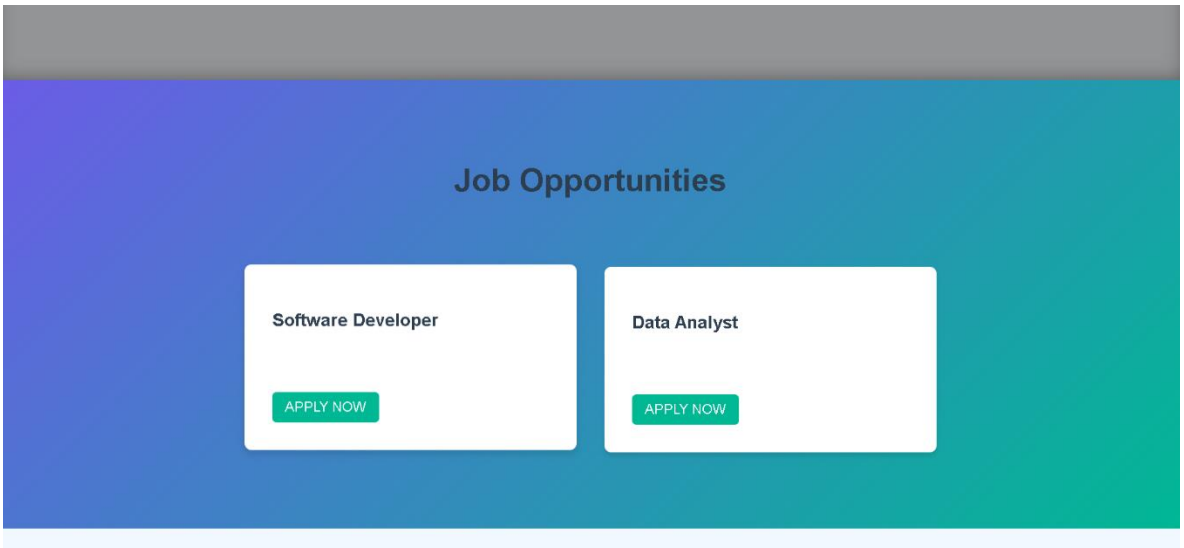


Diagram No:4.2

Jobs posting and application page

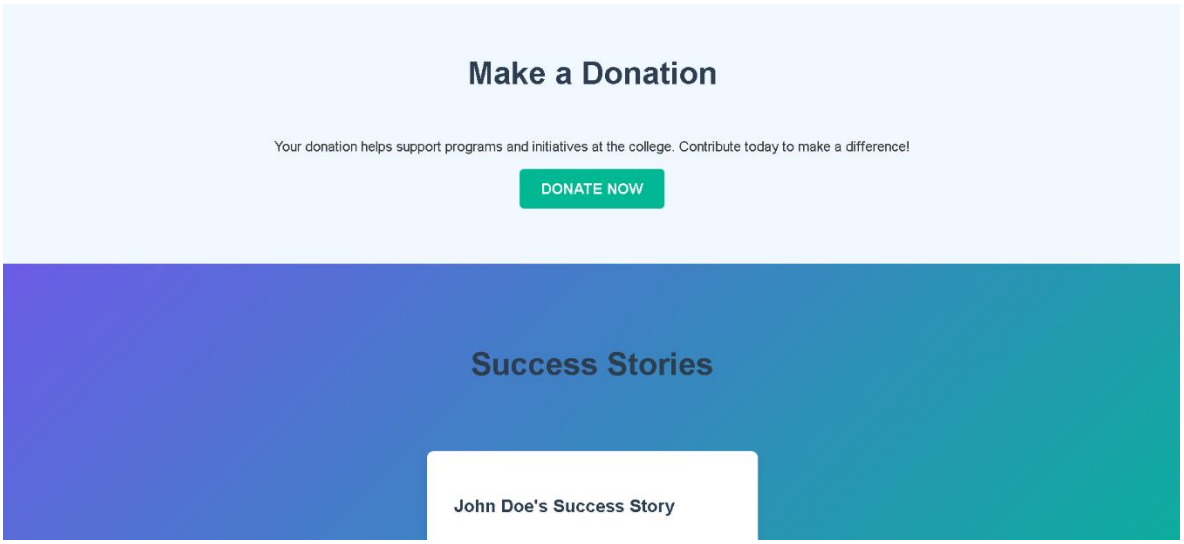


Diagram No:4.3

Donation and Success Story Page

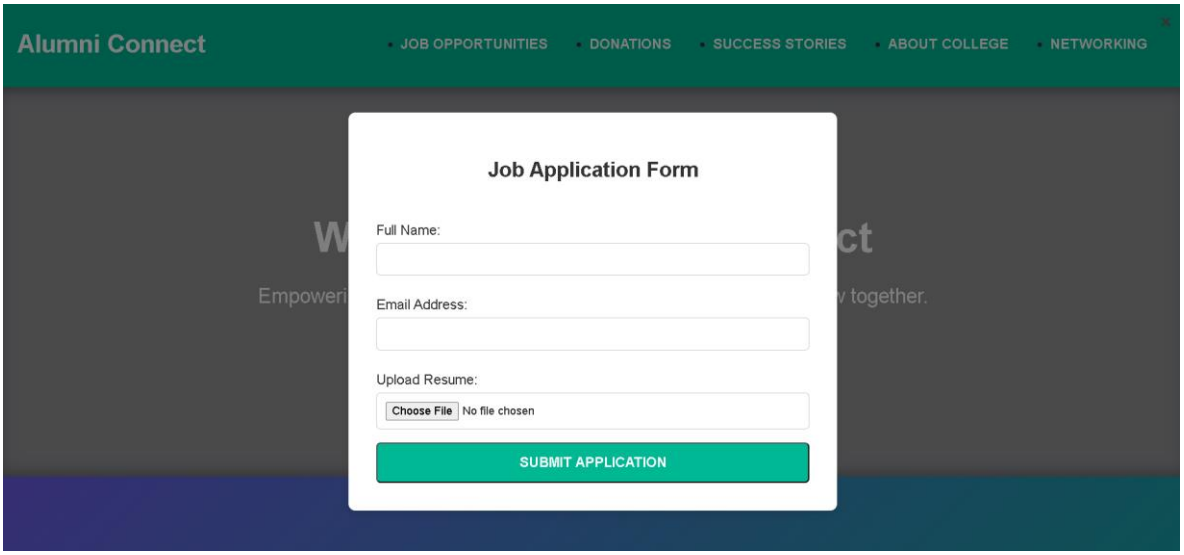


Diagram No:4.4

Job Application Form

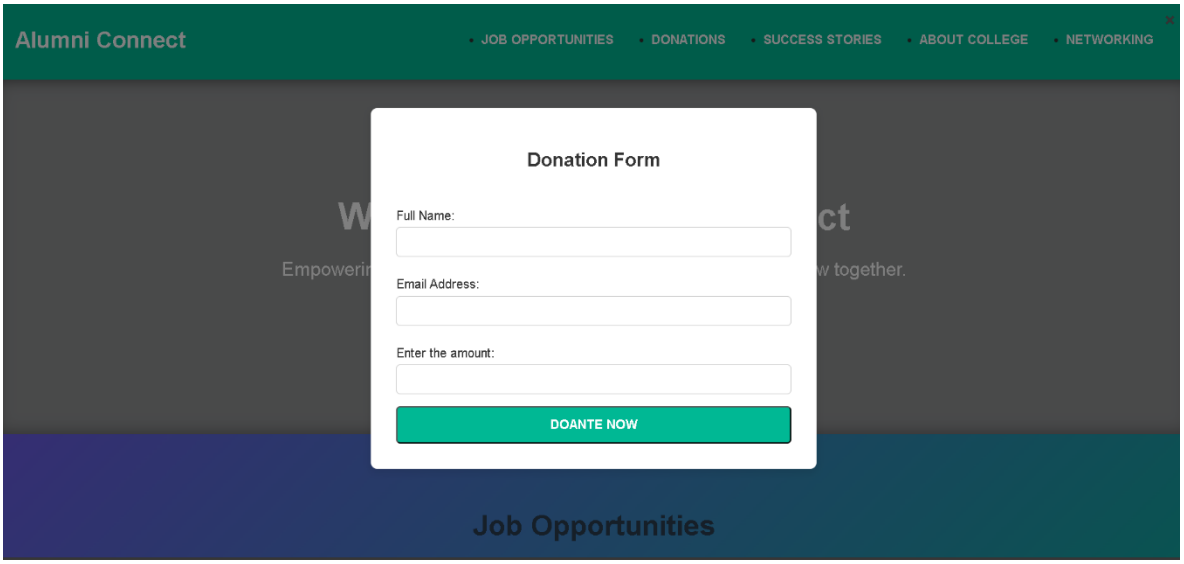


Diagram No:4.5

Donation Form

1. **Dashboard Page**:

- The landing page that provides an overview of the portal and navigation to other sections.
- Includes menus or buttons for accessing Job Opportunities, Donation, Success Stories, and Networking.
- Users can choose a specific feature to proceed further.
- Acts as the central hub for Alumni Portal interactions.

2. **Job Opportunities Page:**

- Displays a list of job opportunities posted by alumni or the institution.
- Users can click **"Apply Now"** to proceed with a job application form.
- Connects alumni with career opportunities and professional growth.

3. Make Donation Page:

- Allows alumni to contribute to the institution by filling out a **Donation Form**.
- Provides a simple way for alumni to support the college financially.

4. Success Stories Page:

- Showcases inspiring stories and achievements of alumni to encourage engagement.
- Builds a sense of community by highlighting alumni contributions.
- Motivates others to contribute through donations or sharing success stories.

5. Job Application Form:

- A dedicated page where users provide details to apply for a job opportunity.
- Typically collects personal, academic, and professional information.
- Includes an option to upload a resume or additional documents.
- Submits the application to the respective job poster for review.

References:

- IEEE Recommended Practice for Software Requirements Specifications (IEEE Std 830-1998).
- https://sih.gov.in/sih2024PS ,Problem statement ID 1609
- http://www.programsformca.com/2012/03/uml-diagrams-for-library-management.html