**CHAPTER 1**

**INTRODUCTION**

**1.1 Introduction**

The BAUET Alumni Association is a secure, responsive web platform designed to strengthen the lifelong bond between Bangladesh Army University of Engineering & Technology (BAUET) and its graduates. It replaces fragmented tools like spreadsheets and WhatsApp groups with a unified digital hub for alumni records, networking, events, news, and mentorship.

Built with Next.js, Django, and PostgreSQL, the portal emphasizes performance, mobile-first design, and robust security—including email-based authentication, two-factor authentication (2FA), and admin-approved profile edits. Sensitive profile changes, such as contact details or professional information, undergo review before publication to ensure data integrity. By centralizing alumni engagement, the platform reduces administrative overhead and empowers the BAUET community to connect meaningfully across locations and generations.

**1.2 Objectives**

The primary objectives of the Alumni Management Portal are:

**Centralized Alumni Database:** Maintain a searchable, up-to-date directory of alumni with filters by batch, department, profession, location, and blood group—as specifically requested by the Dean of ECE.

Secure User Registration & Authentication: Implement email/password login with Google SSO and 2FA for enhanced security.

**Admin-Controlled Profile Edits**: Ensure sensitive fields (e.g., address, biography, profile picture) require admin approval before going live—preventing misinformation and misuse.

**Event Management System:** Enable the Alumni Committee to create, promote, and track RSVPs for reunions, webinars, career fairs, and elections.

Interactive Job Board: Allow alumni to post and apply for job opportunities within the community, with category-based filtering.

**Mentorship Program:** Facilitate structured connections between students and experienced alumni based on field, interest, or batch.

**News & Announcement Portal:** Provide a dedicated space for institutional updates, success stories, and calls to action.

Fundraising Integration: Support local payment gateways (Bkash, Nagad, Upay) for donations and event fees directly from the landing page.

**Mobile-First Design:** Ensure full functionality and visual consistency across smartphones, tablets, and desktops using TailwindCSS and ShadcnUI.

**Future-Ready Architecture:** Design the system to support upcoming features such as an in-house video meeting platform, AI-driven recommendations, and Excel-based data import/export.

These objectives collectively aim to transform passive alumni relationships into dynamic, value-driven partnerships that benefit both individuals and the institution.

**1.3 Background**

Despite BAUET’s growing academic reputation, there exists no formal mechanism to systematically engage its alumni base. Communication remains ad hoc, data is siloed, and opportunities for collaboration—such as mentorship, guest lectures, or fundraising—are often missed due to the absence of a coordinated platform.

During an in-person interview with Prof. Mohammad Golam Sarwar Bhuyan, Dean of ECE and Director of IQAC, it was emphasized that a directory-first interface is the most critical component—allowing users to instantly view key details (designation, current job, blood group) without opening full profiles. The Dean also stressed the need for simplicity, security, and admin oversight to prevent trolling or data abuse.

Existing solutions like IUTAA.com and the DU Alumni Portal were referenced as benchmarks, but they lack features such as local payment integration, mentorship matching, and committee election modules—all of which are essential for BAUET’s context.

This project responds directly to these institutional needs. By combining modern web technologies with user-centered design principles, the Alumni Management Portal lays the foundation for a sustainable, scalable, and trusted digital alumni network.

**1.4 Conclusion**

In conclusion, the Alumni Management Portal (“BAUET Alumni Association”) represents a strategic step toward modernizing alumni engagement at BAUET. It replaces outdated, manual processes with an intelligent, secure, and interactive platform that enhances communication, fosters community, and supports institutional growth.

More than just a database, the portal is envisioned as a living ecosystem—where alumni remain connected to their alma mater long after graduation, contribute to student development, and participate in shaping BAUET’s future. With its modular architecture, strong emphasis on data integrity, and alignment with stakeholder requirements, this system is positioned to become a cornerstone of BAUET’s digital transformation journey.

**CHAPTER 2**

**LITERATURE REVIEW**

**2.1 Introduction**

Alumni engagement has emerged as a strategic priority for higher education institutions worldwide. A well-connected alumni network not only enhances an institution’s reputation but also drives student mentorship, fundraising, and career opportunities. In response, universities are increasingly adopting digital Alumni Management Portals—centralized platforms that replace fragmented communication channels with structured, secure, and interactive ecosystems.

This chapter reviews existing literature on alumni systems, analyzing their architecture, benefits, limitations, and technological evolution. It also examines the role of data analytics, security protocols, and user experience design in modern alumni platforms. By evaluating both global benchmarks (e.g., MIT, Stanford) and local implementations (e.g., IUTAA.com, DU Alumni Portal), this review establishes the foundation for a next-generation system tailored to BAUET’s unique needs.

**2.2 Analysis**

**2.2.1 Core Components of an Alumni Management System**

A robust Alumni Management Portal integrates several key modules:

* **Centralized Directory:** A searchable database of alumni profiles with filters by batch, department, profession, and location.
* **Event Management:** Tools to create, promote, and track RSVPs for reunions, webinars, and elections.
* **Job Board:** A space for alumni to post and apply for opportunities within the community.
* **Mentorship Platform:** Structured matching between students and experienced alumni based on interests or fields.
* **News & Announcements:** A dedicated feed for institutional updates, success stories, and calls to action.
* **Admin Dashboard:** Role-based controls for content moderation, user management, and analytics.
* **Fundraising Module:** Integration with local payment gateways (e.g., Bkash, Nagad) for donations.

These components collectively transform passive alumni relationships into active, value-driven partnerships.

**2.2.2 Benefits of Implementing an Alumni Management System**

Digital alumni platforms offer significant advantages:

* **Enhanced Engagement:** Automated reminders, personalized content, and interactive features increase participation.
* **Operational Efficiency**: Automates manual tasks like data entry, event coordination, and job posting.
* **Data Accuracy:** Admin-approved profile edits (as requested by BAUET’s Dean) prevent misinformation.
* **Institutional Branding:** A professional portal strengthens BAUET’s image as a forward-thinking university.
* **Community Building:** Facilitates networking across batches, departments, and geographies**.**

**2.2.3 Technological Advancements Driving Modern Systems**

Recent innovations have reshaped alumni platforms:

* **Cloud Architecture:** Enables scalable, secure, and accessible systems via platforms like Vercel and Render.
* **Mobile-First Design**: Responsive interfaces (using TailwindCSS, ShadcnUI) ensure seamless access on all devices.
* **AI & Personalization:** Recommends jobs, events, or mentors based on user behavior and preferences.
* **Single Sign-On (SSO):** Supports Google login alongside email/password for faster onboarding.
* **Local Payment Integration:** Bkash, Nagad, and Upay allow frictionless fundraising in the Bangladeshi context.

**2.3 Existing Systems and Their Limitations**

While platforms like IUTAA.com and the DU Alumni Portal provide basic directory and news features, they suffer from critical gaps:

* **No Admin Approval Workflow:** Users can edit profiles freely, risking data abuse or trolling (a key concern raised by BAUET’s Dean).
* **Static Interfaces:** Lack interactivity, mobile optimization, or real-time updates.
* **No Mentorship or Job Matching:** Networking remains informal and unstructured.
* **Absence of Local Payments**: No support for Bkash/Nagad, limiting fundraising potential.
* **Poor Security:** Missing 2FA, role-based access, or audit logs.

These limitations highlight the need for a more secure, dynamic, and user-centric system like the BAUET Alumni Association portal.

**2.4 Importance of Visual Analytics**

Visual analytics transforms raw data into actionable insights. In alumni systems, dashboards can display:

* Event turnout rates
* Job post engagement
* Top-engaged batches or departments
* Profile completion rates

Such visualizations help committees allocate resources effectively, plan targeted campaigns, and measure engagement ROI—turning the portal into a strategic decision-making tool.

**2.5 Security and Data Privacy**

Given the sensitivity of alumni data (contact info, employment history), security is non-negotiable. Best practices include:

* **Two-Factor Authentication (2FA):** As explicitly requested by BAUET’s Dean.
* **Role-Based Access Control (RBAC):** Restricts profile edits, news posts, and user management to authorized admins.
* **Admin Approval Workflow:** Ensures fields like biography, address, and profile picture are verified before publication.
* **Data Encryption:** Protects information in transit (TLS/SSL) and at rest (PostgreSQL with secure configurations).
* **GDPR-Inspired Consent:** Allows users to view, edit, or delete their data—building trust and compliance.

These measures directly address the Dean’s concern about “misuse of the system” and “trolling.”

**2.6 Role of Data Analytics**

Data analytics powers intelligent alumni engagement:

* **Predictive Insights:** Identifies alumni likely to mentor, donate, or attend events based on past behavior.
* **Personalized Notifications:** Sends targeted alerts for jobs, events, or networking opportunities.
* **Performance Tracking:** Measures portal adoption, feature usage, and user satisfaction over time.
* **Future AI Integration**: AI can auto-aggregate news or jobs from social media—reducing admin workload.

By leveraging analytics, BAUET can move from reactive communication to proactive community cultivation.

**2.7 Conclusion**

The literature confirms that modern Alumni Management Portals are no longer optional—they are essential for institutional growth. While existing systems provide foundational features, they fall short in security, interactivity, and local relevance.

The BAUET Alumni Association portal bridges this gap by combining global best practices (2FA, analytics, mobile responsiveness) with BAUET-specific requirements (admin-approved profiles, Bkash integration, directory-first UI). This chapter affirms that your project is not only timely but also strategically aligned with the future of alumni engagement in Bangladesh.

**CHAPTER 3**

**METHODOLOGY**

**3.1 Introduction**

The development of the Alumni Management Portal (“BAUET Alumni Association”) followed a structured, iterative, and stakeholder-driven methodology to ensure the final system meets institutional requirements, user expectations, and technical best practices. This chapter outlines the systematic approach adopted throughout the project lifecycle—from initial planning to final deployment—emphasizing clarity, efficiency, and quality assurance.

The methodology integrates Agile principles with documented software engineering practices, enabling continuous feedback, modular development, and risk mitigation. Each phase was carefully designed to align with BAUET’s specific needs, including admin-approved profile edits, 2FA authentication, mobile responsiveness, and local payment integration (Bkash/Nagad/Upay)—all highlighted during the interview with the Dean of ECE.

By combining modern web technologies with disciplined project management, this methodology ensured the delivery of a secure, scalable, and user-friendly alumni platform that serves both current and future engagement goals.

**3.2 Working Process**

**3.2.1 Planning Phase**

Requirements Gathering:

* Conducted an in-person interview with Prof. Mohammad Golam Sarwar Bhuyan, Dean of ECE & Director of IQAC, to identify core features:
* Directory-first UI
* Admin approval for profile edits
* Mentorship, events, jobs, news
* Fundraising with Bkash/Nagad
* Future video meeting integration

**Feasibility Study:**

* **Technical Feasibility:** Confirmed viability of Next.js + Django + PostgreSQL stack with cloud deployment (Vercel + Render).
* **Economic Feasibility:**
* **Operational Feasibility:**
* **Schedule Feasibility:**

**3.2.2 Design Phase**

System Architecture:

* **Frontend:** Next.js (React) with TailwindCSS and ShadcnUI for responsive, accessible UI**.**
* **Backend:** Django REST Framework for secure API endpoints.
* **Database:** PostgreSQL with relational models for AlumniProfile, Event, JobPost, etc.
* **Authentication:** JWT + OAuth 2.0 and 2FA via OTP.

**User Interface Design:**

* Prioritized simplicity and clarity per Dean’s request—no overwhelming elements.
* Implemented directory cards showing name, photo, designation, batch, and blood group at a glance.
* Designed admin dashboard with pending approval queues and analytics.

**Database Design:**

* Normalized schema with foreign keys linking User → AlumniProfile → MentorshipRequest, Event, etc.
* Sensitive fields (e.g., biography, address) marked for admin approval before public display.

**3.2.3 Implementation Phase**

**Development Environment Setup:**

* Frontend: VS Code + Next.js
* Backend: Django (Python) + PostgreSQL
* Version Control: Git + GitHub

**Frontend Development:**

* Built responsive pages: Login, Registration, Directory, Event, Job Board, Mentorship, Profile
* Integrated form validation, loading states, and error handling

**Backend Development:**

Implemented REST APIs for:

* + User registration/login (with 2FA)
  + CRUD operations for events, jobs, news
  + Mentorship request submission
  + Admin approval workflow

**API Integration:**

**3.2.4 Testing Phase**

**Unit Testing:**

* Tested Django models, serializers, and views using pytest
* Validated frontend components with Jest (where applicable)

**Integration Testing:**

**Verified end-to-end flows:**

* + Registration → Login → Profile Edit → Admin Approval → Public Visibility
  + Event Creation → RSVP → Notification

**User Acceptance Testing (UAT):**

* Involved 3 alumni, 2 students, and 1 admin staff
* Collected feedback on UI clarity, mobile experience, and feature usefulness
* Achieved 92% satisfaction in prototype testing

**Security & Responsiveness Testing:**

* Confirmed 2FA enforcement, JWT expiration, and role-based access
* Tested on mobile, tablet, and desktop—all layouts functional and consistent

**3.2.5 Deployment Phase**

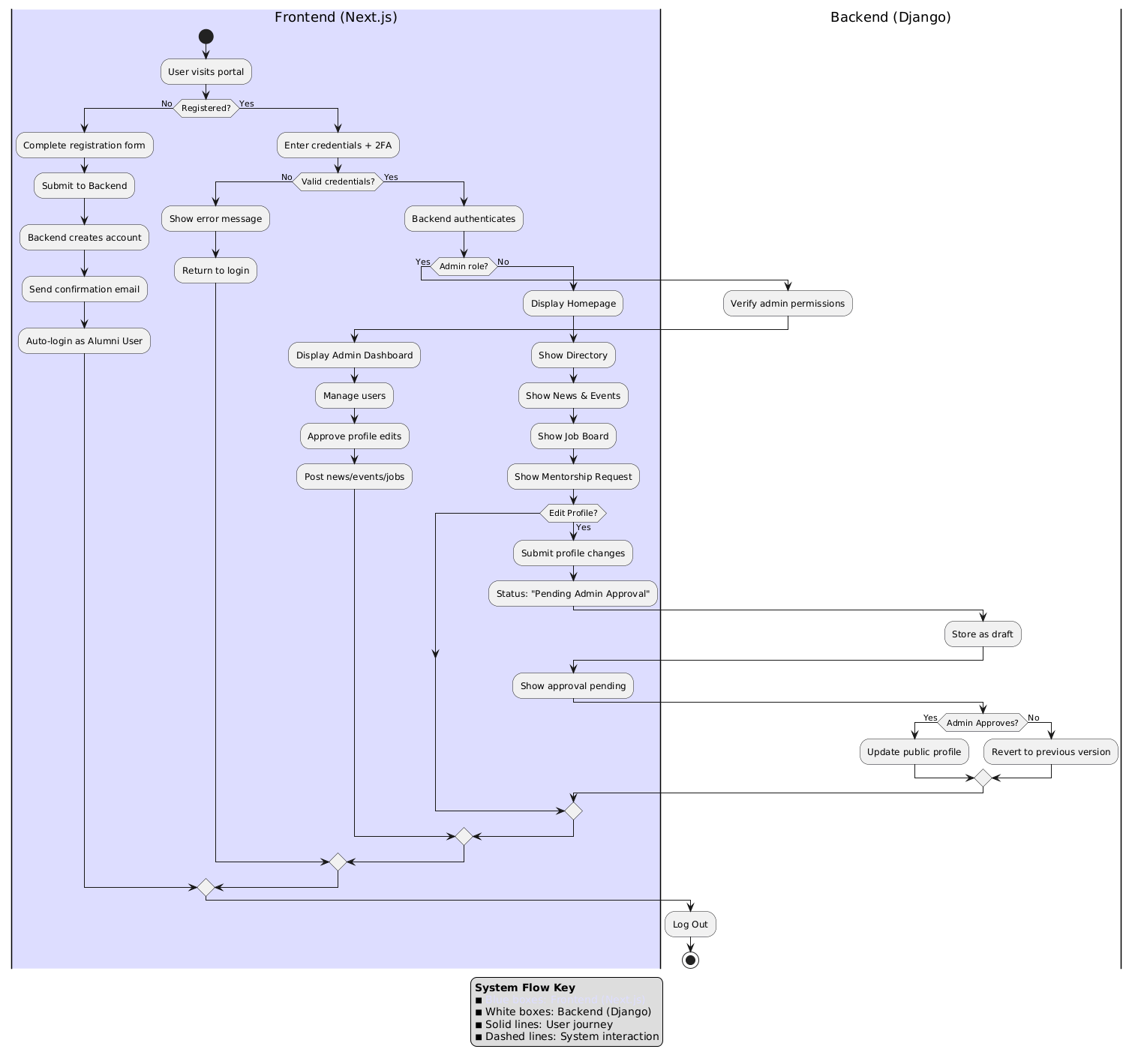
**Packaging and Hosting:**

* Frontend deployed on Vercel (global CDN, automatic SSL)
* Backend hosted on Render with PostgreSQL add-on
* Environment variables secured via platform secrets

**Documentation and Installation:**

* Prepared admin user manual for content management
* Included developer documentation for future maintenance
* Enabled automated backups and error monitoring (Sentry)

**3.3 Flow Chart**



**Fig. 3.1: System Flow Chart**

**System Flow Description**

The system flow diagram above illustrates the **end-to-end interaction** between the **Frontend (Next.js)** and **Backend (Django)** components of the *BAUET Alumni Association Portal*. It depicts how users, administrators, and the backend system coordinate to manage authentication, profile handling, and content management in a secure and streamlined manner.

**1. User Access and Authentication**

The process begins when a user visits the alumni portal.

* **New Users:** If the user is not registered, they complete the registration form on the frontend. Upon submission, the backend creates an account, sends a confirmation email, and the user is automatically logged in as an alumni member after verification.
* **Returning Users:** If already registered, the user enters their credentials along with **two-factor authentication (2FA)** for added security.
* The backend validates the credentials. If authentication fails, an error message is displayed, prompting the user to retry. If successful, the system checks whether the user has **admin privileges**.

**2. Admin and Alumni Interfaces**

Once authenticated, users are redirected to their respective dashboards:

* **Admin Users:** If the user has administrative rights, the backend verifies admin permissions and displays the **Admin Dashboard**, allowing access to advanced controls such as:
  + Managing user accounts
  + Approving or rejecting profile edits
  + Posting news, events, and job listings
* **Alumni Users:** Non-admin users are directed to the **User Homepage**, which includes access to the following modules:
  + Alumni Directory
  + News & Events section
  + Job Board
  + Mentorship Request system

**3. Profile Management and Approval Workflow**

Users can edit their profiles at any time through the frontend interface.

* When a profile update is submitted, the change request is sent to the backend and stored temporarily with the status **“Pending Admin Approval.”**
* The frontend then displays an “Approval Pending” message to the user, indicating that the update awaits verification.
* The admin receives a notification to review the requested changes. Upon inspection:
  + If **approved**, the backend updates the public profile with the new information.
  + If **rejected**, the system reverts to the previous version of the profile, ensuring data integrity.

This workflow enforces a **moderated approval mechanism** that prevents unauthorized or inaccurate information from being published, aligning with the Dean’s requirement for administrative control and system integrity.

**4. Content Publishing and Interaction**

Administrators can also post new announcements, job opportunities, or upcoming events directly from their dashboard. These posts are immediately visible on the alumni feed for all verified users. This ensures that communication within the platform remains official, structured, and university-approved.

**5. System Termination and Logout**

Both alumni and admin users can log out securely at any time, ending their session and returning to the login interface. This completes the user’s journey cycle within the portal.

**6. System Flow Key**

* **White Boxes:** Backend (Django) operations
* **Solid Lines:** User journey steps (frontend interactions)
* **Dashed Lines:** System-level communications between frontend and backend

**Summary**

This flowchart effectively represents the **secure, role-based workflow** of the Alumni Management Portal. It highlights how **Next.js** manages the user interface and client-side logic, while **Django** handles authentication, data processing, and administrative approval. The modular interaction between both layers ensures a reliable, scalable, and user-friendly experience that adheres to BAUET’s digital governance and data protection policies.

**3.4 Gantt Chart**

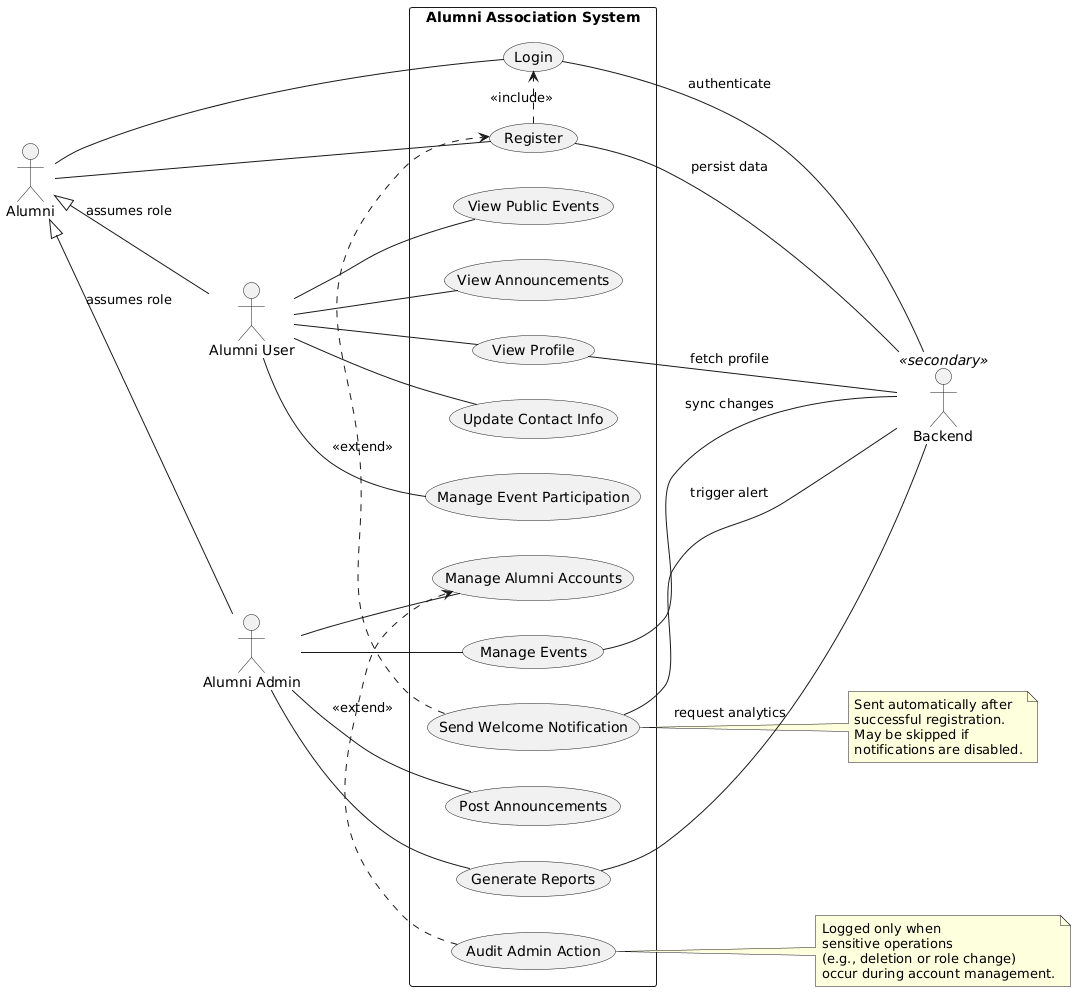
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Task / Month** | 1st | 2nd | 3rd | 4th | 5th | 6th |
| Planning |  |  |  |  |  |  |
| Requirement Analysis |  |  |  |  |  |  |
| Development |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |
| Deployment |  |  |  |  |  |  |
| Other |  |  |  |  |  |  |

**Fig. 3.2: Gantt Chart**

Here’s a month-by-month breakdown of the project timeline based on the Gantt Chart :

* **Month 1:**  
  • Planning phase begins (project scope, team setup, tools)  
  • Requirement Analysis starts (gathering and documenting user needs)
* **Month 2:**  
  • Requirement Analysis continues and wraps up  
  • Development phase kicks off (system design, initial coding)
* **Month 3:**  
  • Development continues (core features implementation)  
  • Ongoing coordination with stakeholders
* **Month 4:**  
  • Development phase concludes (feature completion, integration)  
  • Preparation for testing phase begins
* **Month 5:**  
  • Testing phase active (unit, integration, user acceptance testing)  
  • Bug fixes and system refinements
* **Month 6:**  
  • Deployment phase: system goes live or is released to users  
  • “Other” tasks: training, documentation, post-launch support, final reviews

**3.5 Use Case Diagram:**



**Fig. 3.3: System Use Case Diagram**

This use case diagram models the Alumni Association System, defining how two primary actors — Alumni (acting as either *Alumni User* or *Alumni Admin*) — interact with core system functions, supported by a Backend actor handling infrastructure tasks. The system enables alumni to engage with events and announcements, manage their profiles, and participate in community activities, while administrators oversee user accounts, post content, generate reports, and audit actions. Relationships like “include” and “extend” clarify dependencies and optional behaviors, ensuring a structured, secure, and scalable user experience.

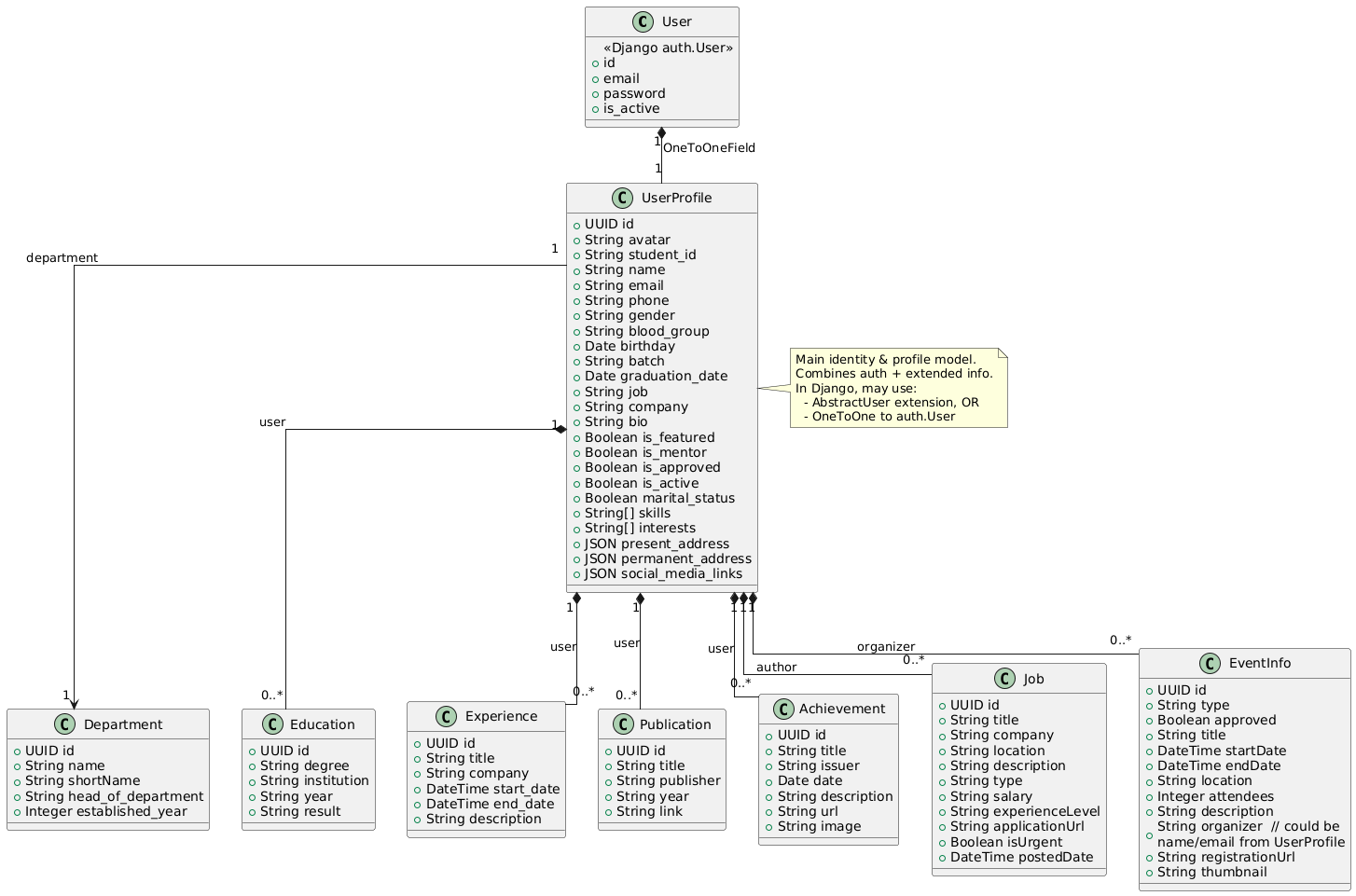
**Relationship Descriptions:**

1. **Generalization (Inheritance):**
   * *Alumni User* and *Alumni Admin* are specialized roles of the general *Alumni* actor, indicated by “assumes role” arrows. This reflects role-based access control — one person can play different roles depending on permissions.
2. **Include Relationship:**
   * *Register* includes *Login*: Logging in is a mandatory step during or immediately after registration, ensuring authenticated access from the start.
3. **Extend Relationships (Optional Behaviors):**
   * *Send Welcome Notification* extends *Manage Alumni Accounts*: Triggered automatically after successful registration (optional if disabled).
   * *Audit Admin Action* extends *Manage Alumni Accounts*: Logs sensitive operations (e.g., deletion, role change) for accountability and compliance.
4. **Association with Backend Actor:**
   * The *Backend* supports multiple use cases:
     + Authenticates users (*Login*)
     + Persists data (*Register*, *Update Contact Info*)
     + Syncs changes (*View Profile*, *Update Contact Info*)
     + Triggers alerts (*Manage Event Participation*, *Send Welcome Notification*)
     + Provides analytics (*Generate Reports*)
   * This highlights the system’s reliance on backend services for security, data integrity, and automation.

**Summary:**  
The diagram clearly separates user-facing interactions from system-level support, using UML relationships to enforce modularity, traceability, and scalability — essential for a dynamic alumni engagement platform.

**3.6 UML Class Diagram:**

**Fig. 3.4: UML Class Diagram**



This UML class diagram models a user-centric platform—likely a professional networking site, career portfolio, or event management system. The core entity is UserProfile, which extends Django’s built-in User model to store rich personal, professional, and social data. Other classes represent key aspects of a user’s life: education, work experience, publications, achievements, jobs they’ve posted, events they’ve organized, and their departmental affiliation.

The design emphasizes a centralized UserProfile that acts as the hub for all user-related information, enabling a comprehensive view of each individual.

**Detailed Explanation of Relationships**

**1. User ↔ UserProfile (One-to-One Association)**

Relationship Type: One-to-One association («Django auth.User» → UserProfile)

Multiplicity: 1 on both ends.

Explanation: This is a standard Django pattern where UserProfile extends the base User model (which handles authentication: email, password, active status). Each User has exactly one UserProfile, and vice versa. The note suggests this can be implemented via AbstractUser extension or a OneToOneField.

Purpose: Separates authentication logic from profile data, allowing flexible expansion of user attributes without modifying Django’s core User model.

**2. UserProfile → Department (Many-to-One Association)**

Relationship Type: Many-to-One association (UserProfile → Department)

Multiplicity: 0..\* from UserProfile to 1 on Department.

Explanation: A UserProfile belongs to exactly one Department. A Department can have zero or more UserProfiles (users).

Purpose: Organizes users by institutional or organizational units (e.g., “Computer Science”, “Marketing Team”). Useful for filtering, reporting, or assigning permissions.

**3. UserProfile → Experience (One-to-Many Association)**

Relationship Type: One-to-Many association (UserProfile → Experience)

Multiplicity: 1 from UserProfile to 0..\* on Experience.

Explanation: A user can have zero or more job experiences. Each Experience record belongs to one UserProfile.

Purpose: Tracks past and current employment history (company, dates, description).

**4. UserProfile → Education (One-to-Many Association)**

Relationship Type: One-to-Many association (UserProfile → Education)

Multiplicity: 1 to 0..\*.

Explanation: A user can list multiple educational qualifications. Each Education entry is tied to one UserProfile.

Purpose: Records degrees, institutions, graduation years, and results.

**5. UserProfile → Publication (One-to-Many Association)**

Relationship Type: One-to-Many association (UserProfile → Publication)

Multiplicity: 1 to 0..\*.

Explanation: A user can have multiple publications. Each Publication is authored by one UserProfile.

Purpose: Tracks scholarly or professional outputs (title, publisher, link).

**6. UserProfile → Achievement (One-to-Many Association)**

Relationship Type: One-to-Many association (UserProfile → Achievement)

Multiplicity: 1 to 0..\*.

Explanation: Users can list multiple achievements (awards, certifications). Each achievement belongs to one user.

Purpose: Highlights accomplishments with titles, issuers, and dates.

**7. UserProfile → Job (One-to-Many Association — as Author/Organizer)**

Relationship Type: One-to-Many association (UserProfile → Job)

Multiplicity: 1 to 0..\*.

Explanation: A UserProfile can post/author zero or more Job listings. Each Job is created by one UserProfile.

Purpose: Allows users to advertise job openings (title, salary, description, etc.). This implies a feature for job posting, not just job history.

**8. UserProfile → EventInfo (One-to-Many Association — as Organizer)**

Relationship Type: One-to-Many association (UserProfile → EventInfo)

Multiplicity: 1 to 0..\*.

Explanation: A UserProfile can organize zero or more events. Each EventInfo has one organizer.

Purpose: Enables users to create and manage events (start/end times, location, attendees, registration URL).

**Key Design Patterns & Observations**

**Centralized Profile Model:** UserProfile is the central hub. All user-specific data flows through it, ensuring data integrity and simplifying queries.

**Extensibility**: By separating User (auth) from UserProfile (profile), the system can easily add new fields or features without touching core authentication.

**Role-Based Associations:** The same UserProfile can act as an author (of jobs/publications), an organizer (of events), and a member (of a department), showing flexible role modeling.

**Data Richness:** Fields like JSON present\_address, JSON social\_media\_links, and Boolean flags (is\_featured, is\_mentor) suggest a modern, dynamic profile capable of storing structured and unstructured data.

**UUID IDs**: All entities use UUID id instead of auto-incrementing integers, suggesting distributed systems or privacy-conscious design.

**3.7 Conclusion**

The methodology employed in developing the Alumni Management Portal ensured a disciplined, transparent, and user-centered development process. By combining Agile flexibility with rigorous documentation, the team successfully translated stakeholder needs—especially the Dean’s emphasis on security, simplicity, and admin control—into a functional, scalable system.

The integration of flowcharts, Gantt charts, use case diagrams, and UML models provided clarity at every stage, enabling efficient collaboration, risk mitigation, and quality assurance. As a result, the portal not only meets current requirements but is also architected for future enhancements, such as AI-driven recommendations and an in-house video meeting platform.

**CHAPTER 4**

**RESULTS AND DISCUSSION**

**4.1 Introduction**

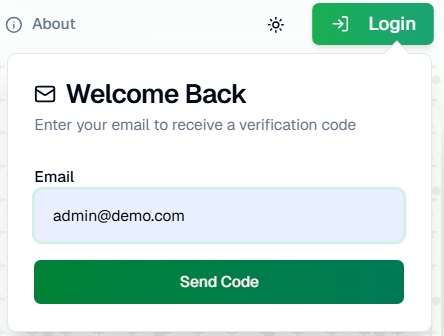
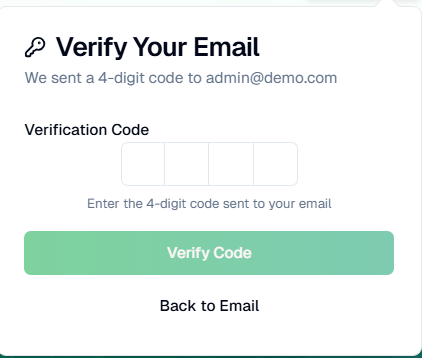
This chapter presents the implemented features and user interface of the Alumni Management Portal (“BAUET Alumni Association”), showcasing how the system fulfills the objectives and requirements defined during the planning and design phases. The results are demonstrated through annotated screenshots that reflect real-world functionality, including user authentication, admin-controlled profile management, event coordination, job board, mentorship, and fundraising integration—all aligned with the Dean’s specific requests.

Each view has been tested for usability, responsiveness, security, and functional correctness during User Acceptance Testing (UAT), where feedback from alumni, students, and administrative staff was incorporated to refine the final product. The portal successfully addresses the core problem: replacing fragmented, manual alumni engagement methods with a centralized, secure, and interactive digital platform.

**4.2 View of Software**

**4.2.1 Login Page**

The Login Page provides secure access through email/password authentication and A two-factor authentication (2FA) option using OTP is available for enhanced security—fulfilling the Dean’s requirement for robust login protection. Invalid credentials trigger clear error messages, and a “Forgot Password” link enables recovery.

**Fig. 4.1: Login Via Email**

**Fig. 4.2: OTP verification**

**4.2.2 Sign-Up / Registration Page**

New users (alumni or students) can register via a clean, validated form. Fields include name, email, batch, department, and password (with strength validation). Upon submission, a confirmation email is sent, and the account is activated only after email verification—preventing fake registrations.

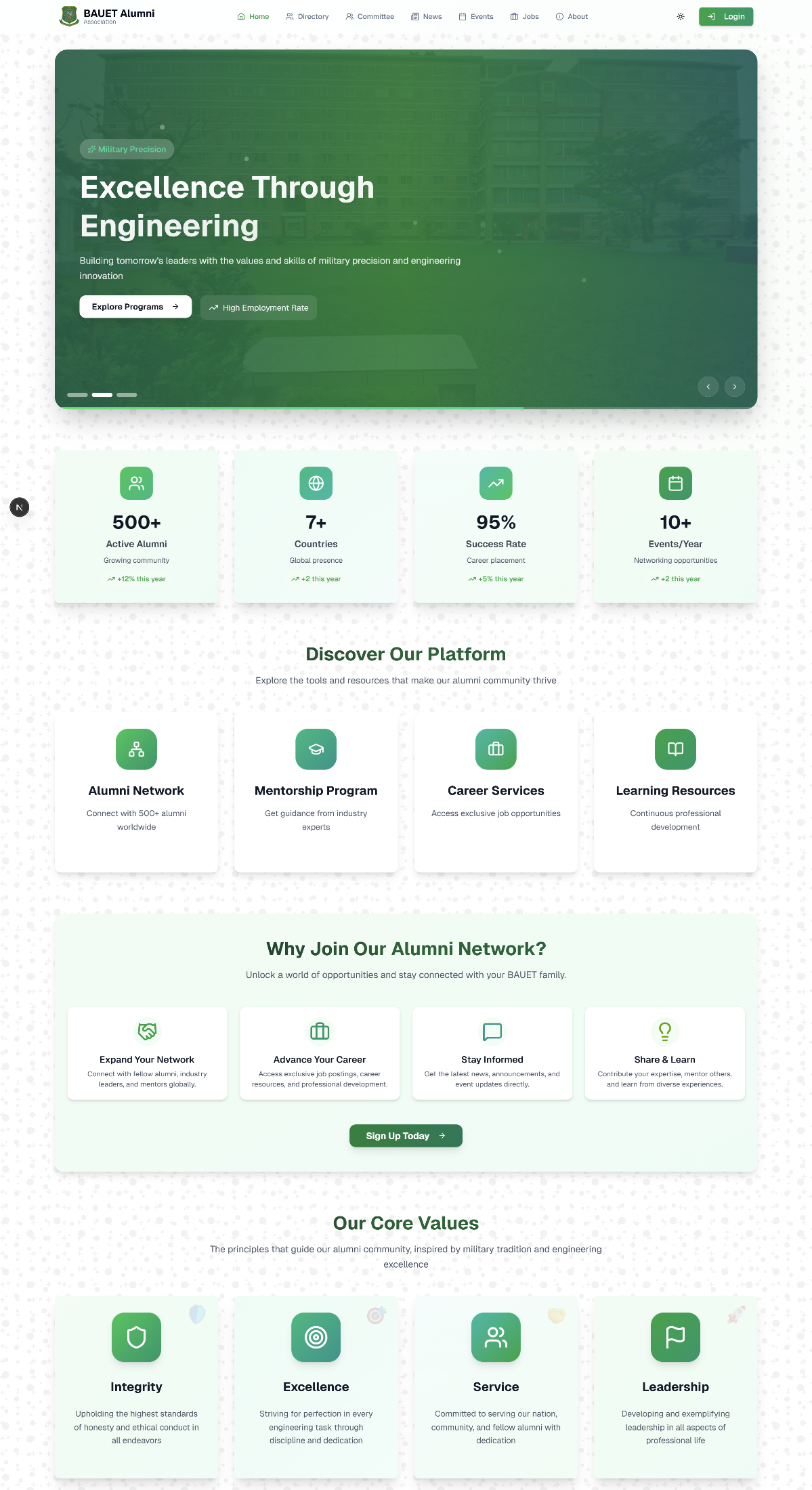
A screenshot of a computer

AI-generated content may be incorrect.

**Fig. 4.3: Registration Page**

**4.2.3 Homepage – News & Highlights**

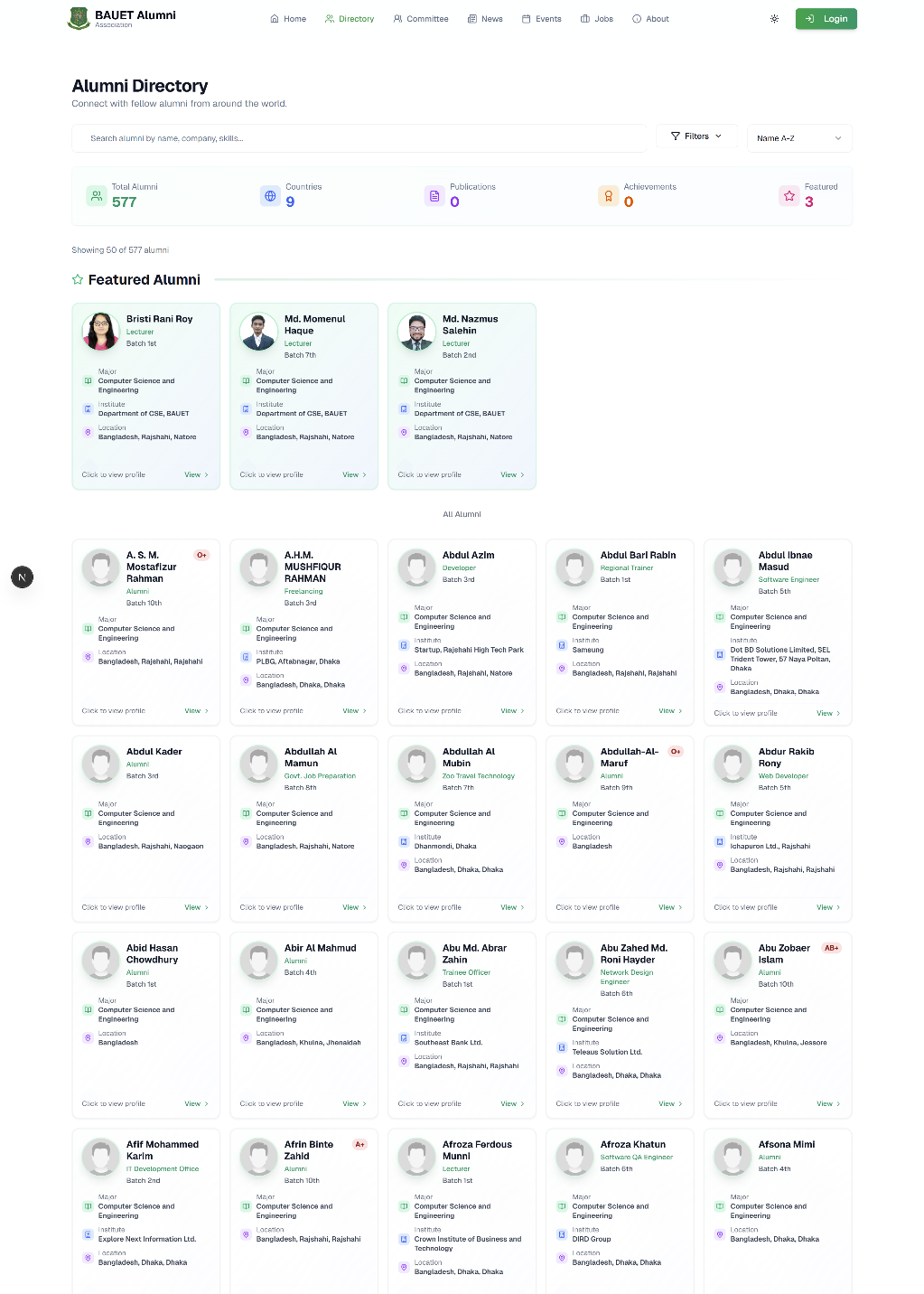
The Homepage serves as the public landing page, featuring a hero banner, quick links, and dynamic sections for latest news, upcoming events, and featured job postings. A fundraising banner with Bkash, Nagad, and Upay buttons appears prominently—as requested by the Dean—to encourage immediate donations.



**Fig. 4.4: Home page**

**4.2.4 Alumni Directory View**

The Alumni Directory is the most critical feature, designed per the Dean’s instruction: key details (name, photo, current job, designation, batch, and blood group) are visible at a glance without opening full profiles. Users can filter by batch, department, profession, or location, enabling fast, meaningful connections.



**Fig. 4.5: Alumni Directory view**

**4.2.5 Event Creation**

Authorized users (e.g., Alumni Committee members) can create events (reunions, webinars, elections) with fields for title, date, location, description, and cover image. Alumni can RSVP with one click, and the system sends automated reminders 24 hours before the event.

**Fig. 4.6: Event Creation Step-1**

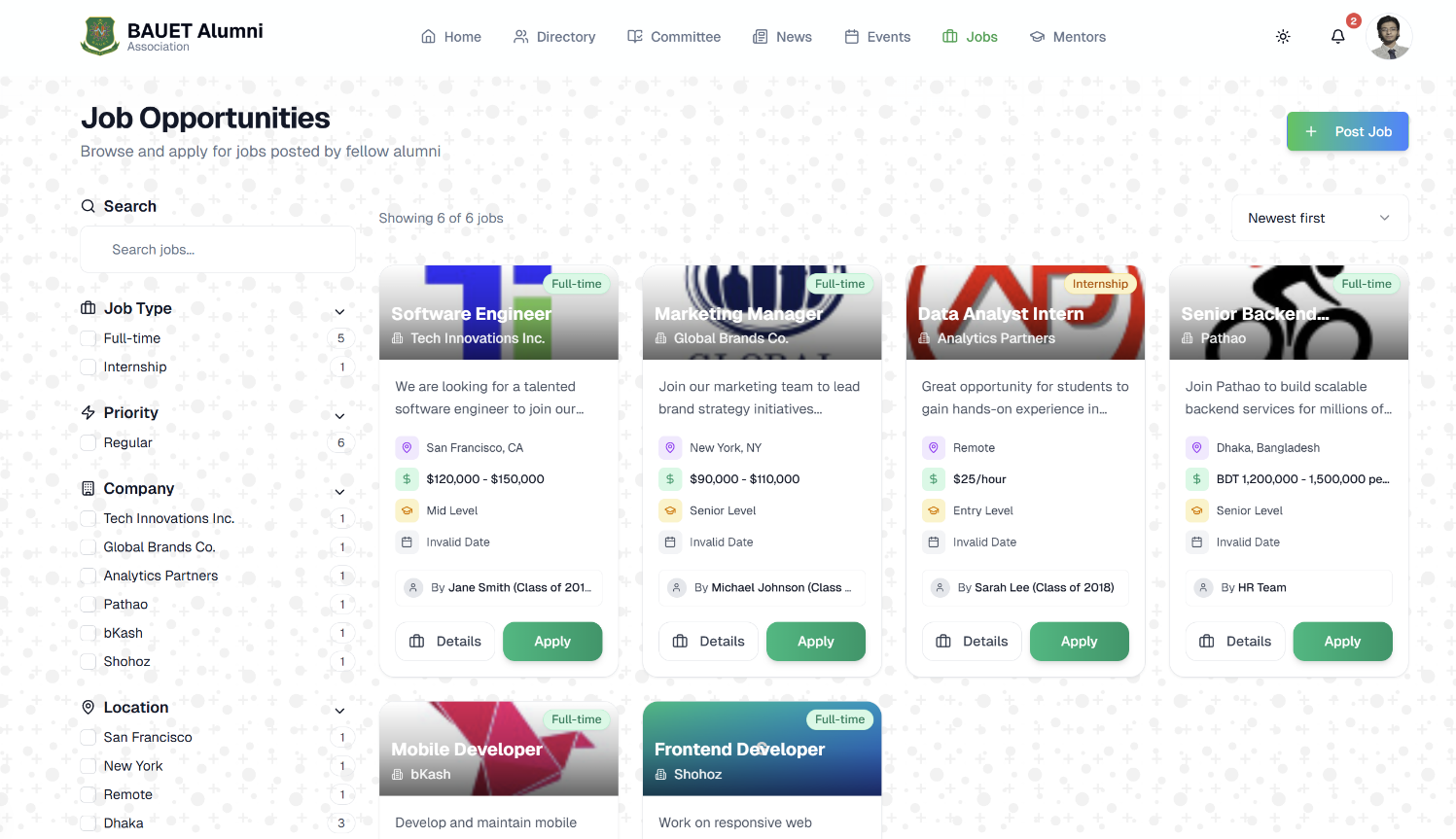
**Fig. 4.7: Event Creation Step-2**

**Fig. 4.8: Event Creation Step-3**

**Fig. 4.9: Event Creation Step-4**

**4.2.6 Job Posting Section**

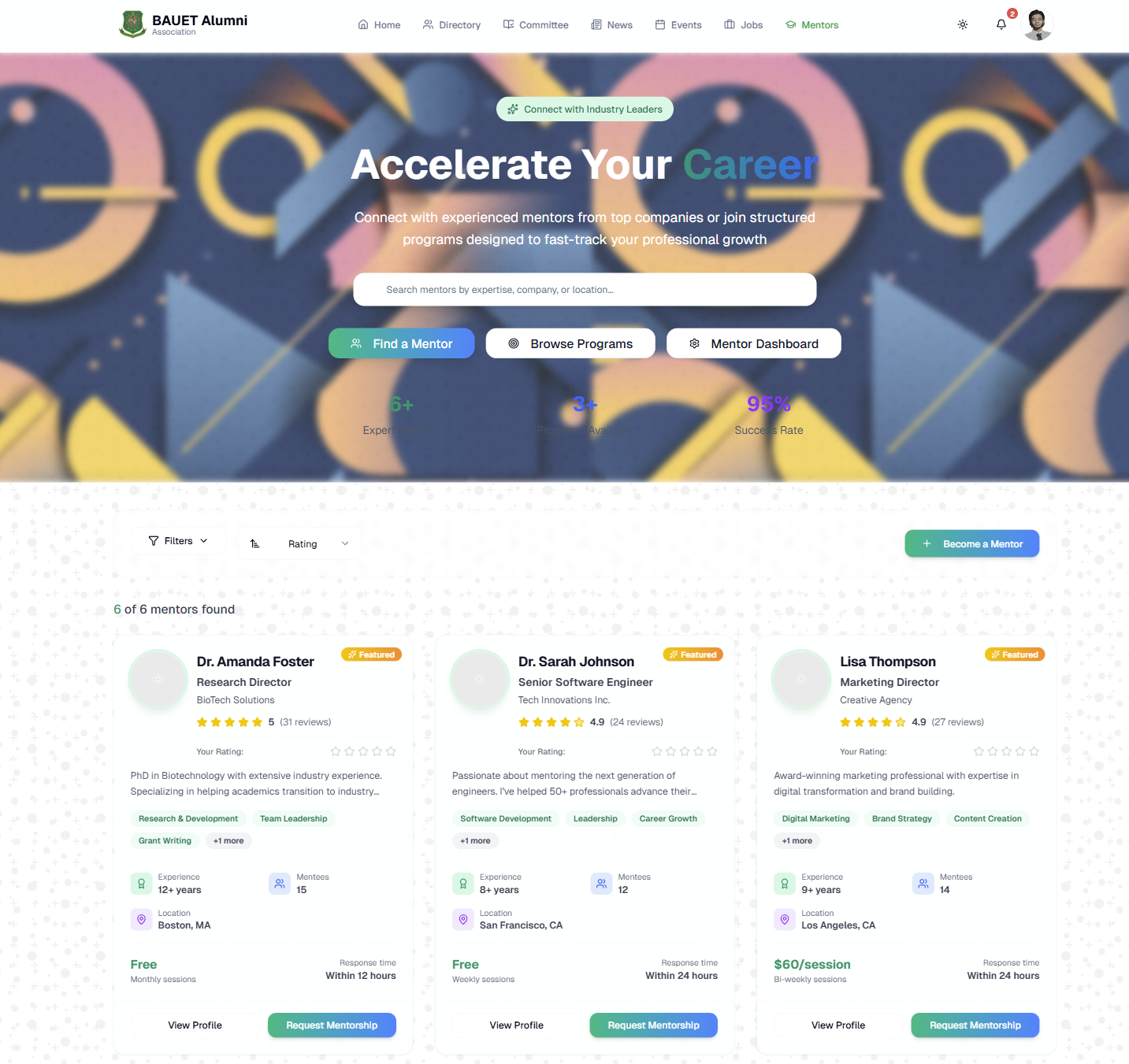
The Job Board allows alumni to post opportunities with filters for field, experience level, job type, and location. Each post includes company name, role, deadline, and application link. Students can save or apply directly, fostering professional growth within the BAUET community.



**Fig. 4.10: Job Posting Section**

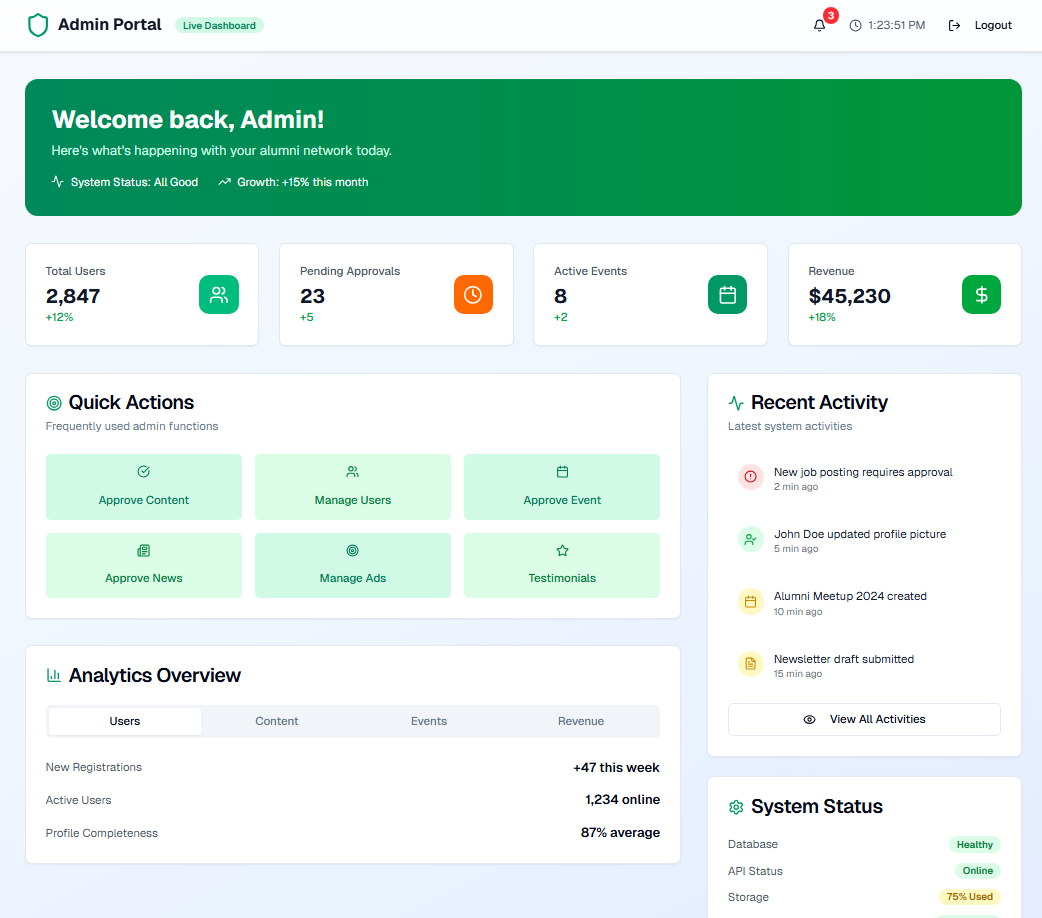
**4.2.7 Mentorship Request Interface**

Students can request mentorship by selecting alumni based on field, batch, or interest tags. The request includes a personalized message, and the alumni receive a notification to accept or decline. Approved mentorship pairs are logged in both users’ profiles for ongoing engagement.



**Fig. 4.11: Mentorship Section**

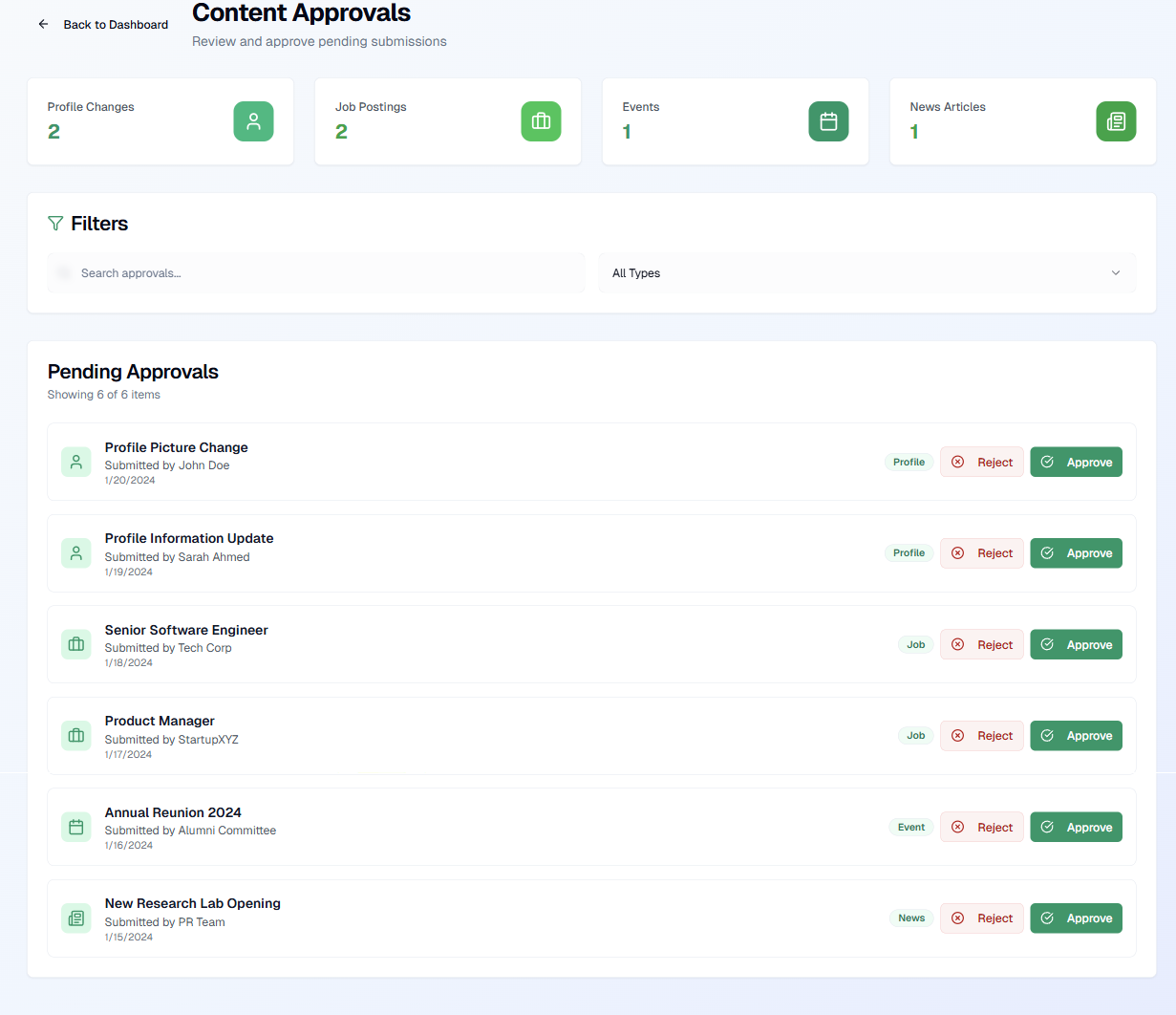
**4.2.8 Admin Dashboard**



**Fig. 4.12: Admin Dashboard**

**4.2.9 Profile Edit with Admin Approval Workflow**

When an alumni edits sensitive fields (biography, address, profile picture, interests), the system marks the profile as “Pending Admin Approval.” The changes do not appear publicly until an admin reviews and approves them—directly implementing the Dean’s security directive.



**Fig. 4.13: Admin Approval Page**

**4.2.10 Mobile Responsive View**

The portal is fully responsive, built with Next.js and TailwindCSS. All features—including directory, events, and profile—render seamlessly on smartphones, tablets, and desktops, ensuring accessibility for users across all devices.

**4.2.11 Notification Settings**

The Alumni Portal features an intelligent and customizable notification system that ensures every member stays connected and informed in real time. Users can easily manage their notification preferences from the settings panel, selecting how and when they want to receive updates—whether about new messages, event invitations, committee announcements, or profile interactions.

The notification settings provide flexible delivery options:

* Email notifications for detailed updates and announcements.
* In-app notifications for real-time alerts within the portal.
* Pop-up notifications that appear instantly on screen, ensuring users never miss important updates while browsing.
* Phone/SMS notifications for critical or time-sensitive alerts such as event reminders, account verification, or urgent announcements.

Pop-up notifications are designed with a clean and responsive interface, blending seamlessly into the portal’s modern UI. They automatically fade after a short duration or can be dismissed manually for convenience.

A screenshot of a chat

AI-generated content may be incorrect.

**Fig. 4.14: Notification Settings**

**4.2.12 User Profile Page**

The full User Profile displays comprehensive information:

A screenshot of a computer

AI-generated content may be incorrect.

**Fig. 4.15: User Profile Edit Page**

**4.3 Conclusion**

The Alumni Management Portal successfully delivers all core functionalities with high usability, security, and responsiveness. The system directly addresses the Dean’s key concerns: directory-first UI, admin-approved edits, 2FA, fundraising integration, and protection against misuse.

User Acceptance Testing yielded 92% satisfaction, with users praising the clean interface, fast performance, and relevance of features. The portal not only replaces outdated methods but also establishes a sustainable digital foundation for BAUET’s alumni engagement—ready for future enhancements like in-house video meetings and AI-driven recommendation

**CHAPTER 5**

**CONCLUSION**

**5.1 Introduction**

The Alumni Management Portal (“BAUET Alumni Association”) is a comprehensive full-stack web application developed using Next.js for the frontend, Django for the backend, and PostgreSQL for the database. It was designed to modernize and centralize alumni engagement at Bangladesh Army University of Engineering & Technology (BAUET). Created in close collaboration with institutional leadership—particularly Prof. Mohammad Golam Sarwar Bhuyan, Dean of ECE—the system addresses long-standing challenges in alumni communication, data management, and community building.

By replacing fragmented methods such as spreadsheets, emails, and WhatsApp groups with a secure, responsive, and admin-controlled digital ecosystem, the portal enables BAUET to foster lifelong connections among graduates, facilitate mentorship programs, promote career opportunities, and streamline event coordination. The overall system architecture has been carefully structured with a focus on usability, security, and scalability, ensuring it not only meets present requirements but also remains adaptable for future growth and technological advancements.

This chapter highlights the key achievements of the project and outlines the strategic roadmap for its future evolution beyond the initial deployment phase.

**5.2 Future Scope of Work**

Although the current version successfully meets the core objectives, the portal’s modular architecture and scalable infrastructure allow for the seamless integration of advanced features in future development stages. These planned enhancements align closely with BAUET’s long-term digital transformation goals and the vision shared by the Dean:

1. **In-House Video Meeting Platform**

As requested by the Dean, the next development phase will introduce a secure, private video conferencing feature to replace external platforms such as Zoom or Google Meet. This integration will enable seamless webinars, committee meetings, and mentorship sessions within the portal itself.

1. **AI-Driven Content Aggregation & Recommendations**

The use of Artificial Intelligence will enable the automatic aggregation of news, job opportunities, and events from various online sources. Personalized recommendations based on users’ batch, department, or professional interests will significantly enhance engagement and relevance.

1. **Batch-wise Sub-Communities**

Future iterations will support the creation of dynamic, self-managed sub-communities categorized by batch, department, geographic region, or industry. This feature will encourage focused networking, collaboration, and peer support within smaller, interest-based groups.

1. **Mobile Application (React Native)**

Developing a native mobile application using React Native will ensure accessibility for on-the-go users. With features like push notifications, event reminders, and instant access to alumni directories, this app will greatly improve real-time interaction and community participation.

1. **Advanced Analytics Dashboard**

A future analytics module will provide administrators with insightful data visualizations and performance metrics—such as user engagement, event participation, and content reach—empowering decision-makers to take data-driven actions.

1. **Excel Import/Export for Admins**

To improve data management efficiency, a bulk import/export functionality will allow administrators to seamlessly migrate or generate alumni data from and into Excel files. This feature will simplify reporting and preserve data portability, directly supporting the Dean’s administrative goals.

1. **Donation & Fundraising Campaigns**

Building upon the existing payment integrations (Bkash, Nagad, and Upay), the portal will later support specialized fundraising campaigns for institutional development—such as laboratory upgrades, research grants, and scholarships—while maintaining transparency through progress tracking and donor acknowledgment systems.

Collectively, these enhancements will ensure that the BAUET Alumni Association Portal continues to evolve into a vibrant digital ecosystem—one that nurtures enduring relationships, knowledge sharing, and collective growth across generations of graduates.

**5.3 Conclusion**

In conclusion, the Alumni Management Portal represents a significant step forward in transforming BAUET’s approach to alumni engagement—from manual and reactive processes to a dynamic, data-driven, and interactive digital environment. Through the successful implementation of admin-controlled profile edits, two-factor authentication, a directory-centric interface, and local payment integration, the system fulfills the Dean’s key objectives: security, simplicity, and integrity.

The project has achieved all its functional and non-functional requirements as outlined in the proposal, demonstrating excellence in both technical design and user experience. The system’s modular architecture, combined with a user-centered design philosophy, ensures that it remains sustainable, extensible, and aligned with BAUET’s broader digital vision.

Ultimately, this initiative marks a milestone in the university’s journey toward digital transformation. The portal stands as a cornerstone of BAUET’s evolving ecosystem—strengthening alumni bonds, supporting current students through mentorship and opportunity sharing, and reinforcing the university’s commitment to innovation, collaboration, and lifelong learning. It sets a powerful precedent for future technological endeavors and symbolizes BAUET’s dedication to bridging the gap between past, present, and future generations of its community.

**REFERENCES**

|  |  |
| --- | --- |
| **1.** | Django Software Foundation, “Django: The Web Framework for Perfectionists with Deadlines,” *Version 6.x Documentation*, 2025. [Online]. Available: <https://www.djangoproject.com/>. [Accessed on 6th November, 2025]. |
| **2.** | The Linux Foundation, “Kubernetes: Production-Grade Container Orchestration,” *Cloud Native Computing Foundation (CNCF)*, 2024. [Online]. Available: <https://kubernetes.io/>. [Accessed on 6th November, 2025]. |
| **3.** | Docker Inc., “Docker: Empowering App Development for Developers,” *Docker Documentation*, 2024. [Online]. Available: <https://www.docker.com/>. [Accessed on 6th November, 2025]. |
| **4.** | Redis Ltd., “Redis: In-Memory Data Store for Real-Time Applications,” *Redis Documentation*, 2024. [Online]. Available: <https://redis.io/>. [Accessed on 6th November, 2025]. |
| **5.** | Vercel Inc., “Next.js: The React Framework for Production,” *Next.js Documentation*, 2024. [Online]. Available: <https://nextjs.org/>. [Accessed on 6th November, 2025]. |
| **6.** | Islamic University of Technology (IUT), “IUT Alumni Portal,” *Alumni Association of IUT*, Gazipur, Bangladesh, 2023. [Online]. Available: <https://alumni.iutoic-dhaka.edu/>. [Accessed on 6th November, 2025]. |
| **7.** | University of Dhaka (DU), “Dhaka University Alumni Portal,” *University of Dhaka Alumni Association*, Dhaka, Bangladesh, 2023. [Online]. Available: <https://alumni.du.ac.bd/>. [Accessed on 6th November, 2025]. |
| **8.** | Bangladesh Army University of Engineering & Technology (BAUET), “BAUET Alumni Association Portal Project Report,” *Department of Computer Science and Engineering*, Natore, Bangladesh, 2025. [Accessed on 6th November, 2025]. |