CHAPTER 3

Proposed System

3.1 Overview

The ASTU AlumniLink project is envisioned as an innovative web-based platform designed to enhance networking, mentorship, and job opportunities for the community associated with Adama Science and Technology University (ASTU). Its primary purpose is to create a centralized hub that facilitates meaningful interactions among users, aiming to foster connections between students, alumni, faculty, and prospective employers. By providing tailored features, the platform seeks to encourage networking, enhance career development, and promote alumni engagement, allowing alumni to actively participate in the university community by sharing their expertise and guiding current students.

The system distinguishes between four main user groups—students, alumni, faculty, and companies, each with specific access rights and functionalities. Students will access mentorship opportunities and job listings, internship opportunity while alumni can offer guidance and post events. Faculty will facilitate connections and provide resources, and companies will recruit directly from the ASTU community. Key functionalities of the platform include user registration, mentorship matching, job and internship listings, discussion forums, and event management.

This initiative addresses existing challenges in traditional networking methods, which often prove inefficient and fragmented. The current career services at ASTU lack structured communication channels to effectively connect students with alumni and employers, leading to missed opportunities. By creating a dedicated space for continuous interaction, the ASTU AlumniLink platform aims to provide targeted resources that cater to the university community's needs, thereby fostering a supportive ecosystem for professional growth.

The expected outcomes of the project include increased student engagement with alumni and employers, enhanced access to mentorship and job opportunities, and a collaborative environment that promotes lifelong learning. Ultimately, the ASTU AlumniLink project is a strategic initiative designed to transform career development and networking within the university, enriching the educational experience and fostering valuable professional connections.

3.2 Functional Requirements

The ASTU AlumniLink platform includes several key functionalities designed to enhance user interaction and streamline networking, mentorship, and job opportunities. Below is a detailed yet concise overview of the functional requirements:

★ User Registration and Profile Management:
The system will allow different user types—students, alumni, faculty, and companies—to create and manage their profiles. Each user type will have unique registration forms tailored to their specific needs and access rights.

★ Mentorship Matching:
The platform will feature a mentorship matching system that connects students with alumni based on shared interests, fields of study, and professional goals. This functionality will facilitate meaningful mentor-mentee relationships.

★ Job and Internship Listings:
The system will enable companies to post job vacancies and internship opportunities. Students will have access to these listings, allowing them to apply directly through the platform.

★ Discussion Forums:
Users will have the ability to participate in discussion forums where they can share insights, ask questions, and engage with peers and mentors on various topics related to career development and industry trends.

★ Event Management:
The platform will support event creation and management, allowing users to organize networking events, workshops, and seminars. Users can RSVP to events and receive notifications about upcoming opportunities.

★ Ratings and Feedback:
After attending events or engaging with mentors, users will have the option to provide ratings and feedback. This feature will help improve the platform's offerings and enhance user experiences.

★ Communication Tools:
The system will include built-in messaging tools that allow users to communicate directly with one another, facilitating networking and collaboration.

★ Analytics and Reporting:
The platform will provide analytics tools for administrators to monitor user engagement, job placement success, and overall platform effectiveness, enabling continuous improvement.

3.3 Non-Functional Requirements

The non-functional requirements of the ASTU AlumniLink platform ensure it operates efficiently, securely, and user-friendly. Here’s a detailed yet concise overview:

★ Usability:
The platform will feature an intuitive interface characterized by a clean and user-friendly design that facilitates smooth navigation and registration processes. To support new users, comprehensive help resources will be provided, including user manuals and FAQs, alongside interactive tutorials that guide users through the platform's functionalities.

★ Performance:
The system is designed to achieve a response time of under two seconds for all user interactions,

ensuring a seamless and smooth experience. Additionally, it will be capable of supporting a significant number of concurrent users without performance degradation, particularly during peak usage times, to maintain service quality.

★ Accessibility:

The ASTU AlumniLink platform will be accessible on a variety of devices, including desktops, tablets, and smartphones, allowing users to engage with the platform from any location. Compliance with WCAG (Web Content Accessibility Guidelines) will be prioritized, ensuring that the platform is usable for individuals with disabilities and promoting inclusivity.

★ Compatibility:

The system will provide cross-browser support, functioning seamlessly across major web browsers such as Chrome, Firefox, Safari, and Edge without compatibility issues. Additionally, it will support various operating systems, including Windows, macOS, and Linux, ensuring broad access for all users.

★ Security:

Ensuring user data protection is a top priority; therefore, all user data will be encrypted, and secure authentication protocols will be implemented to safeguard user privacy. Regular security audits will be conducted to identify and address potential vulnerabilities, maintaining a secure environment for users.

★ Documentation:

Comprehensive documentation will be meticulously maintained to cover all system functionalities, including detailed user guides and design documents. This documentation will serve as a valuable resource for future reference and training, and will be maintained with version control to ensure that all updates are tracked and accessible.

★ Reliability:

The platform will strive to achieve an uptime guarantee of 99.9%, ensuring that users can consistently access services without interruption. In the event of issues, the system will provide clear error messages and recovery options, empowering users to navigate problems effectively and maintain trust in the platform's reliability.

★ Error Handling and Extreme Conditions:

The system will incorporate robust error management mechanisms designed to detect and handle errors gracefully, thereby minimizing user frustration. Additionally, the platform will be designed for scalability, accommodating future growth in user numbers and feature enhancements without significant downtime, ensuring it can adapt to the evolving needs of its community.

3.4. System model

This section presents the system model for the ASTU Alumnilink system, illustrating its structure, functions, behavior, and interactions among various components. The model encompasses conceptual, logical, and physical aspects, providing a comprehensive view of how the system operates.

3.4.1. Scenario

Scenario 1: Alumni Registration

Use Case Name: Alumni Registration
Participating Actors: Alumni, System Administrator (Admin)

Entry Condition:

- The alumni user accesses the Alumni Link website to register.

Flow of Events

1. The alumni clicks on the "Register" link on the homepage.
2. The system displays the alumni registration form.
3. The alumni fills in the required information
4. The alumni clicks the "Register" button.
5. The system validates the input and displays a success message indicating that the registration is submitted for review.
6. The system sends a confirmation email to the alumni, notifying them that their registration is under review.
7. The system administrator reviews the registration details.
   - ★ If Accepted:
     - □ The Admin approves the registration.
     - □ The system sends an acceptance email to the alumni, providing them with access details.
   - ★ If Rejected:
     - □ The Admin rejects the registration.
     - □ The system sends a rejection email to the alumni, explaining the reason for rejection.

Alternate Case:

If the alumni makes an error while filling out the form (e.g., missing required fields, invalid email format) and clicks the "Register" button, the system displays an error message indicating the specific issues to correct.

Exit Condition

- After completing the registration process (either successfully with acceptance or after an error correction), the alumni exits the system.

Scenario 2: Student Registration

Use Case Name: Current Student Registration

Participating Actors: Current Student, System

Entry Condition:

- The current student accesses the Alumni Link website to register.

Flow of Events

1. The current student clicks on the "Register" link on the homepage.
2. The system displays the student registration form.
3. The current student fills in the required information
4. The current student clicks the "Register" button.
5. The system checks that the email provided matches the format of a university email (e.g., username@astu.edu.et).
6. The system verifies the provided email against the student registrar database to ensure the student is currently registered at ASTU.
    - ★ If Verified:
        - □ The system displays a success message indicating that registration is successful.
        - □ The system sends a confirmation email to the student’s university email, providing access details.
    - ★ If Not Verified:
        - □ The system displays an error message indicating that the email does not match the university records or that the student is not currently enrolled.

Alternate Case

If the current student makes an error while filling out the form (e.g., missing required fields, invalid email format) and clicks the "Register" button, the system displays an error message indicating the specific issues to correct.

Exit Condition

After completing the registration process (either successfully or after correcting input errors), the current student exits the system.

Scenario 3: Faculty registration

Use Case Name: Faculty Registration

Participating Actors: Faculty, System Administrator (Admin), System

Entry Condition:

The faculty member accesses the Alumni Link website to register.

Flow of Events

1. The faculty clicks on the "Register" link on the homepage.
2. The system displays the faculty registration form.
3. The faculty fills in the required information (e.g., name, department, email).
4. The faculty clicks the "Register" button.
5. The system checks that the email provided is in a valid format.
6. The system verifies the faculty member's details against the faculty database to ensure they are an active faculty member at ASTU.
    - ★ If Verified:
        - ❑ The system sends the registration details to the Admin for review.
        - ❑ The Admin reviews the registration details.
            - ★ If Approved:
                - ❑ The Admin approves the registration.
                - ❑ The system sends an acceptance email to the faculty member, providing access details.
            - ★ If Rejected:
                - ❑ The Admin rejects the registration.
                - ❑ The system sends a rejection email to the faculty member, explaining the reason for rejection.
    - ★ If Not Verified:
        - ❑ The system displays an error message indicating that the faculty member does not match any records in the faculty database.

Alternate Case

If the faculty makes an error while filling out the form (e.g., missing required fields, invalid email format) and clicks the "Register" button, the system displays an error message indicating the specific issues to correct.

Exit Condition

After completing the registration process (either successfully with approval, after receiving a rejection, or after correcting input errors), the faculty member exits the system.

Scenario 4: Company registration

Use Case Name: Company Registration

Participating Actors: Company Representative, System Administrator (Admin), System

Entry Condition:

The company representative accesses the Alumni Link website to register.

Flow of Events

1. The company representative clicks on the "Register" link on the homepage.
2. The system displays the company registration form.
3. The company representative fills in the required information, including:
   - ☐ Company Name
   - ☐ Contact Information (email, phone number)
   - ☐ Mission Statement
   - ☐ Vision Statement
   - ☐ Letters or documents indicating their intent to collaborate with the university, alumni, and students (e.g., internship proposals).
4. The company representative clicks the "Register" button.
5. The system checks that all required fields are completed and that the provided email is in a valid format.
6. The system sends the registration details to the Admin for review.
7. The Admin reviews the submitted information.
   - ★ If Approved:
     - ☐ The Admin approves the registration.
     - ☐ The system sends an acceptance email to the company representative, providing access details and confirming their registration.
   - ★ If Rejected:
     - ☐ The Admin rejects the registration.
     - ☐ The system sends a rejection email to the company representative, explaining the reason for rejection.

Alternate Case

If the company representative makes an error while filling out the form (e.g., missing required fields, invalid email format) and clicks the "Register" button, the system displays an error message indicating the specific issues to correct.

Exit Condition

After completing the registration process (either successfully with approval, after receiving a rejection, or after correcting input errors), the company representative exits the system.

Scenario : Create event

Use Case Name: Create Event

Participating Actors: Alumni, Company Representative, Faculty, System Administrator (Admin), System

Entry Condition:

The user (Alumni, Company, Faculty, or Student Leader) is logged into their dashboard on the Alumni Link website.

Flow of Events

1. The user clicks on the "Create Event" link on their dashboard.
2. The system displays the event creation form.
3. The user fills in the required informationThe user clicks the "Submit" button.
4. The system checks that all required fields are completed and that the date and time are valid.
5. The system sends the event details to the Admin for review.
6. The Admin reviews the submitted event information.
   - ★ If Approved:
     - □ The Admin approves the event.
     - □ The system sends a confirmation email to the user, notifying them of the approval and providing event details.
   - ★ If Rejected:
     - □ The Admin rejects the event.
     - □ The system sends a rejection email to the user, explaining the reason for rejection.

Alternate Case

If the user makes an error while filling out the form (e.g., missing required fields, invalid date/time) and clicks the "Submit" button, the system displays an error message indicating the specific issues to correct.

Exit Condition

After completing the event creation process (either successfully with approval, after receiving a rejection, or after correcting input errors), the user exits the system.

Scenario 5:Participation online event

Use Case Name: Participate in Online Event

Participating Actors: Alumni, Company Representative, Faculty, Student Leaders, System

Entry Condition:

The user (Alumni, Company, Faculty, or Student Leader) is logged into their dashboard on the Alumni Link website, and the event is scheduled to start.

Flow of Events

1. The user navigates to the "Events" section on their dashboard.
2. The system displays a list of upcoming events, including the online discussion forum.
3. The user selects the relevant online event from the list.
4. The system checks if the current time matches the scheduled start time of the event.
    - ★ If the Event is Live:
        - □ The system provides a "Join Now" button.
        - □ The user clicks the "Join Now" button.
        - □ The system redirects the user to the online discussion forum platform The user participates in the discussion, engaging with other attendees and speakers.
    - ★ If the Event is Not Live:
        - □ The system displays a message indicating that the event has not yet started or is not currently active.

Alternate Case

If the user attempts to join the event before the scheduled start time, the system displays a message indicating that the event will begin shortly and provides the scheduled start time.

Exit Condition

After the event concludes, the user exits the online discussion forum and returns to the Alumni Link website.

Scenario 6: Add connections and message

Use Case Name: Add Connections / Add Friends

Participating Actors: Student, Faculty, Alumni, System

Entry Condition:

The user (Student, Faculty, or Alumni) is logged into their dashboard on the Alumni Link website.

Flow of Events

1. The user navigates to the "Connections" or "Friends" section in their dashboard.
2. The system displays a list of suggested connections based on the user's profile, interests, and existing connections.
3. The user identifies a person they wish to connect with and clicks the "Add Connection" or "Send Friend Request" button next to that person's name.
4. The system sends a connection request to the selected user.
5. The selected user receives a notification of the incoming connection request.
   - ★ If the Selected User Accepts the Request:
     - □ The system updates both users' connection status to "Connected."
     - □ The system sends a confirmation notification to both users indicating the successful connection.
   - ★ If the Selected User Declines the Request:
     - □ The system notifies the original user that their connection request was declined.
   - ★ If the Selected User Ignores the Request:
     - □ The request remains pending until the selected user decides to respond.
6. Once connected, users can communicate with each other through:
   - ★ Direct Messaging:
     - □ The user navigates to the "Messages" section or uses a messaging feature.
     - □ The user selects the connected friend from their contact list.
     - □ The user composes and sends a message.
     - □ The system delivers the message to the recipient.
     - □ The recipient receives a notification of the new message.
   - ★ Discussion Forums or Group Chats:
     - □ Users can participate in public or private discussions related to events or topics of interest.

Alternate Case

- □ If the user tries to send a connection request to someone they are already connected with, the system displays a message indicating that they are already friends.

- □ If the selected user cannot be found (e.g., due to a typo), the system displays an error message indicating that no matching user was found.

Exit Condition

After sending a connection request or communicating with a friend, the user exits the "Connections" or "Messages" section and returns to their dashboard.

Scenario 7: Search for Alumni

Use Case Name: Search for Alumni

Participating Actors: Student, System

Entry Condition:

The student is logged into their dashboard on the Alumni Link website.

Flow of Events

1. The student navigates to the "Alumni Search" section on their dashboard.
2. The system displays a search bar with options to filter by name or department.
3. The student enters the name of the alumni they wish to find or selects their department from a dropdown menu.
4. The student clicks the "Search" button.
5. The system processes the request and searches for matching alumni in the database.
    - ★ If Matches Are Found:
        - □ The system displays a list of alumni that match the search criteria, including their names, graduation years, and current positions or companies.
        - □ The student can click on an alumni's name to view their profile and see additional details (e.g., contact information, career history).
    - ★ If No Matches Are Found:
        - □ The system displays a message indicating that no matching alumni were found.

Exit Condition

After completing the search (whether successful or not), the student can either return to their dashboard or explore other sections of the Alumni Link website.

Scenario 8: Alumni Review

Use Case Name: Admin Review

Participating Actors: System Administrator (Admin)

Entry Condition:

The admin is logged into the admin dashboard on the Alumni Link website.

Flow of Events

1. The admin navigates to the "Review Submissions" section on their dashboard.
2. The system displays a list of pending submissions
3. The admin selects a submission to review (e.g., an event request).
4. The system displays the details of the selected submission
5. The admin evaluates the submission
6. The admin can take one of the following actions:
    - ★ If Approved:
        - □ The admin clicks the "Approve" button.
        - □ The system updates the submission status to "Approved."
        - □ The system sends a confirmation notification to the requesting user, informing them of the approval.
    - ★ If Rejected:
        - □ The admin clicks the "Reject" button.
        - □ The system prompts the admin to enter a reason for rejection.
        - □ The system sends a rejection notification to the requesting user, including the reason for rejection.
    - ★ If Further Information is Needed:
        - □ The admin clicks the "Request More Information" button.
        - □ The system sends a notification to the requesting user asking for additional details.

Alternate Case

If the admin tries to approve a submission that is already approved or rejected, the system displays a message indicating the current status of the submission.

Exit Condition

After completing the review process for one or more submissions, the admin can return to the "Review Submissions" section or log out of the admin dashboard.

Scenario 9: Mentor Match

Use Case Name: Mentorship Match

Participating Actors: Student, Alumni, Faculty

Entry Condition:

The user (Student, Alumni, or Faculty) is logged into their dashboard on the Alumni Link website.

Flow of Events

➔ Initiating a Mentorship Match

1. User Navigation:

    - ☐ The student navigates to the "Mentorship" section on their dashboard.
    - ☐ The system displays options for finding a mentor or becoming a mentor.

2. Finding a Mentor:

    - ☐ The student selects the option to "Find a Mentor."
    - ☐ The system prompts the student to fill out any additional preferences for mentorship, such as:
        - ☐ Desired skills to learn
        - ☐ Availability for meetings
    - ☐ The student submits their mentorship preferences.

3. Matching Process:

    - ☐ The system processes the submitted preferences and searches for potential mentors (alumni or faculty) based on the areas of expertise listed in their profiles.
    - ☐ The system generates a list of potential mentors who match the student's interests and availability.

4. Displaying Matches:

    - ☐ The system displays the list of matched mentors, including their profiles, areas of expertise, and availability for meetings.
    - ☐ The student reviews the profiles and selects a mentor to connect with.

5. Connecting with a Mentor:

    - ☐ The student clicks the "Connect" button next to the chosen mentor's profile.
    - ☐ The system sends a connection request to the selected mentor.

6. Mentor Response:

    - ☐ The chosen mentor receives a notification of the connection request.
    - ☐ If the Mentor Accepts:

        - □ The system updates both users' statuses to "Connected."
        - □ The system sends a confirmation notification to both the student and the mentor with next steps for scheduling meetings.
    - □ If the Mentor Declines:
        - □ The system notifies the student that the request was declined, and the student can choose another mentor from the list.

Exit Condition

After completing the mentorship matching process, whether by finding a mentor or becoming a mentor, users can return to their dashboards or log out of the system.

Scenario 10: User feedback

Use Case Name: User Feedback Submission
Participating Actors: Student, Alumni, Faculty, Company Representative, System

Entry Condition:

The user (Student, Alumni, Faculty, or Company Representative) is logged into their dashboard on the Alumni Link website.

Flow of Events

➔ Submitting Feedback

1. User Navigation:

    - □ The user navigates to the "Feedback" section on their dashboard.
    - □ The system displays options for submitting feedback

2. Filling Out the Feedback Form:

    - □ The user selects a category for their feedback and is directed to a feedback form.
    - □ The user fills in the required fields, including:
        - □ Subject of Feedback
        - □ Detailed description
        - □ Rating (e.g., 1 to 5 stars)
    - □ The user submits the feedback form.

3. System Processing:

- □ The system processes the feedback submission and stores it in the database.
- □ The user receives a confirmation notification that their feedback has been submitted successfully.

➔ Admin Review of Feedback

4. Admin Actions:

- □ The admin navigates to the "User Feedback" section on their dashboard.
- □ The system displays a list of all submitted feedback, categorized by type.
- □ The admin reviews feedback, can filter by category, and can respond to feedback as necessary.

➔ Viewing Feedback

6. User Viewing Feedback:
- □ Users can navigate to the "My Feedback" section to view feedback they have submitted.
- □ The system displays a list of their feedback entries, showing the status (e.g., Pending, Addressed) and any responses from the admin.

Exit Condition

After submitting feedback or reviewing feedback, users can return to their dashboards or log out of the system.

Scenario 11: Resource request

Use Case Name: Resource Request from University

Participating Actors: Alumni, System Administrator (Admin)

Entry Condition:

The alumni user is logged into their dashboard on the Alumni Link website.

Flow of Events

➔ Submitting a Resource Request

1. Alumni Navigation:

- □ The alumni navigates to the "Resource Request" section on their dashboard.
- □ The system displays an option to submit a new resource request.

2. Filling Out the Resource Request Form:

- □ The alumni clicks on "Request Resources" and is presented with a form.

- □ The alumni fills in the required information
- □ The alumni submits the request form.

3. System Processing:

- □ The system processes the resource request and stores it in the database.
- □ The alumni receives a confirmation notification that their request has been submitted successfully.

➔ Admin Review of Resource Requests

4. Admin Actions:

- □ The admin navigates to the "Resource Requests" section on their dashboard.
- □ The system displays a list of pending resource requests submitted by alumni.
- □ The admin selects a request to review.

5. Reviewing the Request:

- □ The admin views the details of the request, including the project description, requested resources, and justification.
- □ The admin evaluates the request based on:
    - □ Relevance to university resources
    - □ Feasibility and availability of requested resources
    - □ Impact on the alumni's research

6. Decision Making:

- □ The admin decides to either approve or reject the request:
    - ➢ If Approved:
        - □ The admin clicks the "Approve" button.
        - □ The system updates the request status to "Approved."
        - □ A notification is sent to the alumni informing them of the approval and any next steps.
    - ➢ If Rejected:
        - □ The admin clicks the "Reject" button.
        - □ The system prompts the admin to enter a reason for rejection.
        - □ A notification is sent to the alumni explaining the reason for rejection.

Exit Condition:

After processing the resource request, whether by approval or rejection, the admin can return to the "Resource Requests" section or log out of the system.

Scenario 12: Profile management

Use Case Name: User Profile Management
Participating Actors: Student, Alumni, Faculty

Entry Condition:

The user (Student, Alumni, or Faculty) is logged into their dashboard on the Alumni Link website.

Flow of Events

➔ Accessing the Profile Management Section

1. User Navigation:
   - □ The user navigates to the "Profile" section on their dashboard.
   - □ The system displays the user's current profile information

➔ Updating Profile Information

2. Editing Profile:
   - □ The user clicks on the "Edit Profile" button.
   - □ The system allows the user to modify various fields
   - □ The user makes the desired changes and clicks the "Save Changes" button.

3. System Processing:
   - □ The system validates the updated information
   - □ If validation is successful, the system updates the user's profile in the database and displays a confirmation message.

➔ Viewing Profile

4. Viewing Profile:
   - □ The user can view their updated profile by navigating back to the "Profile" section.
   - □ The system displays the updated profile information, allowing the user to confirm that all

5. Profile Deletion (Optional):
   - □ The user may have the option to delete their profile.
   - □ If the user selects "Delete Profile," the system prompts for confirmation.
   - □ Upon confirmation, the system deletes the user's profile and sends a notification about the deletion.

Exit Condition

After managing their profile, the user can return to their dashboard or log out of the system.

3.4.2. Use case model

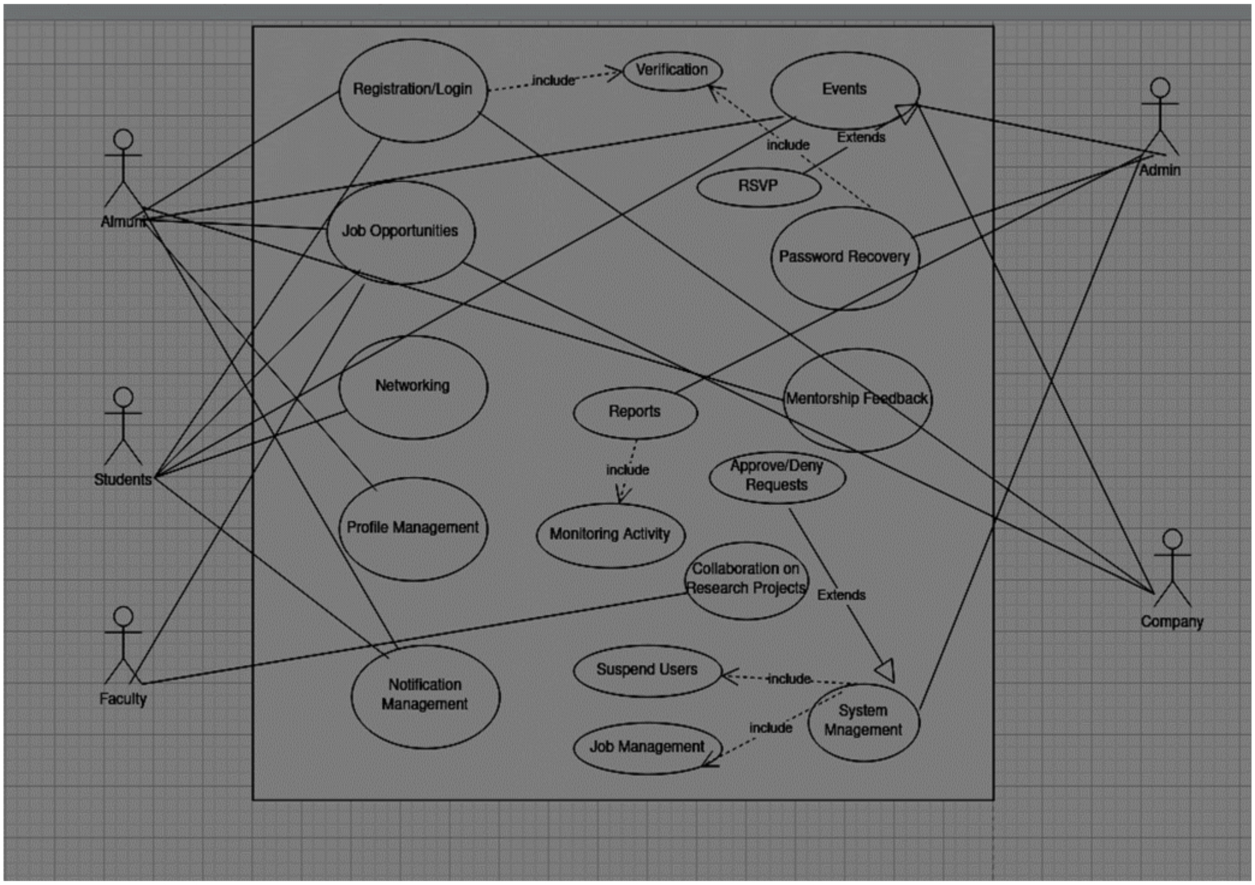


Figure: Use Case Diagram

Use Case Description

Use Cases	Description

User Registration	- Actors: Alumni, Students, Faculty, Companies - Preconditions: User is on the registration page. - Postconditions: User account is created, and a confirmation email is sent. - Main Flow: 1.User fills in the registration form with required information (name, email, password, role). 2. User submits the form. 3. System validates the input. 4. System creates a new user account in the database. 5. System sends a confirmation email to the user.

User Login	- Actors: Alumni, Students, Faculty, Companies - Preconditions: User has a registered account. - Postconditions: User is logged in and directed to their dashboard. - Main Flow: 1. User enters their email and password on the login page. 2. User submits the login form. 3. System validates the credentials. 4. If valid, the system logs the user in and redirects them to their dashboard. 5. If invalid, the system displays an error message.
Profile Management	- Actors: Alumni, Students, Faculty, Companies - Preconditions: User is logged in. - Postconditions: User profile is updated in the database. - Main Flow: 1. User navigates to the profile management section. 2. User edits their profile information (bio, skills, work experience). 3. User submits the changes. 4. System validates the input.

	5. System updates the user profile in the database.
Job Posting	- Actors: Alumni, Companies - Preconditions: User is logged in and has the role of Alumni or Company. - Postconditions: Job posting is created and submitted for approval. - Main Flow: 1. User navigates to the job posting section. 2. User fills in the job details (title, description, requirements). 3. User submits the job posting. 4. System stores the job details and marks it as pending approval.
Job Application	- Actors: Students, Alumni - Preconditions: User is logged in and has found a job listing. - Postconditions: Application is submitted and stored in the database. - Main Flow: 1. User views job listings and selects a job. 2. User clicks on “Apply” and fills in the application form. 3. User submits the application.

	4. System validates the application and stores it in the database.
Event Creation	. - Actors: Alumni, Faculty - Preconditions: User is logged in. - Postconditions: Event is created and submitted for approval. - Main Flow: 1. User navigates to the event creation section. 2. User fills in event details (title, date, location, description). 3. User submits the event for approval. 4. System stores the event details and marks it as pending approval.
Networking (Connection Requests)	- Actors: Alumni, Students - Preconditions: User is logged in. - Postconditions: Connection request is sent or accepted. - Main Flow: 1. User searches for other users to connect with. 2. User sends a connection request. 3. If the other user accepts, the connection is established. 4. System updates the connection status in the database.

Admin User Management	- Actors: Admin - Preconditions: Admin is logged in. - Postconditions: User accounts are verified, approved, or suspended. - Main Flow: 1. Admin navigates to the user management section. 2. Admin reviews pending user registrations. 3. Admin approves or denies user accounts, providing reasons for denial if applicable. 4. System updates the user status in the database.
Admin Job Management	- Actors: Admin - Preconditions: Admin is logged in. - Postconditions: Job postings are approved or removed. - Main Flow: 1. Admin navigates to the job management section. 2. Admin reviews pending job postings. 3. Admin approves or denies job postings, providing reasons if applicable. 4. System updates the job status in the database.

Admin Event Management	. - Actors: Admin - Preconditions: Admin is logged in. - Postconditions: Events are approved, denied, or removed. - Main Flow: 1. Admin navigates to the event management section. 2. Admin reviews pending events. 3. Admin approves or denies events, providing reasons if applicable. 4. System updates the event status in the database.

3.5. Object Model

3.5.1. Data Dictionary

The data dictionary is used to define classes in the system and the member of class like attribute, operation and description about the classes

Classes	Attributes	Operations	Describtion

User	FName(str), U_id(int), password(str), email(str), phone(str), userType(str), bio(text), ProfilePicture(str)	login()
		Register()
		updateProfile()
		viewProfile()
			Used to login, register and Displays the user's profile information.

jobListing	Job_Id(int), title(str), description(text), company(str), location(str), postingDate(Date)	createJobListing(), viewJobListing(), Apply(), updateJobListing(), deleteJobListing()	This class encapsulates the essential attributes and operations related to job postings, enabling users to view, apply for, and manage job listings effectively.

Event	Event_ID(int), title(str), date(Date), description(text)	createEvent()
		viewEvent()
		cancelEvent()
			represents events on the ASTU AlumniLink platform, including workshops and seminars, and facilitates user engagement.

Resource	Resource_ID(int), type(str), link(str), description(text)	shareResource()
		viewResource()
		deleteResource()
			represents shared educational materials on the ASTU AlumniLink platform, facilitating access to valuable learning resources.

Discussion	Discussion_ID(int), topic(str), message(text)	createDiscussion()
		postMessage()
		viewMessage()
			allow users to engage in dialogue, share experiences, and post messages within discussion forums.

Mentorship	Mentorship_ID(int), mentor_ID(int), mentee-ID(int), startDate(Date), endDate(Date), goals(text)	createMentorship()
		viewMentorship()
		updateMentorship()
		endMentprship()
			manages mentor-mentee relationships, facilitating connections and support for personal and professional development.

notification	Notification_ID(int), user_ID(int), message(text), date(Date), status(str)	createNotification()
		viewNotification()
		markAsRead()
		deleteNotifications()
			manages alerts and updates for users, keeping them informed about relevant activities and changes within the platform.

feedback	Feedback_ID(int), user_ID(int), event_ID(int), rating(int), comments(text)	submitFeedback()
		viewFeedback()
		updateFeedback()
		deleteFeedback()
			collects and manages user feedback on events and resources, helping to improve future offerings.

Admin	Admin_ID(int),
	Username(str), password(str), email(str), role(str)
		Login()
		manageUsers()
		createEvent()
		manageJobListings()
		sendNotifications()
		manageContent()
		manageFeedback()
			manages user accounts, oversees content, generates reports, and communicates updates on the ASTU AlumniLink platform, ensuring smooth operations and enhancing user experience.

Table :- Data dictionary

3.5.2. Class diagram

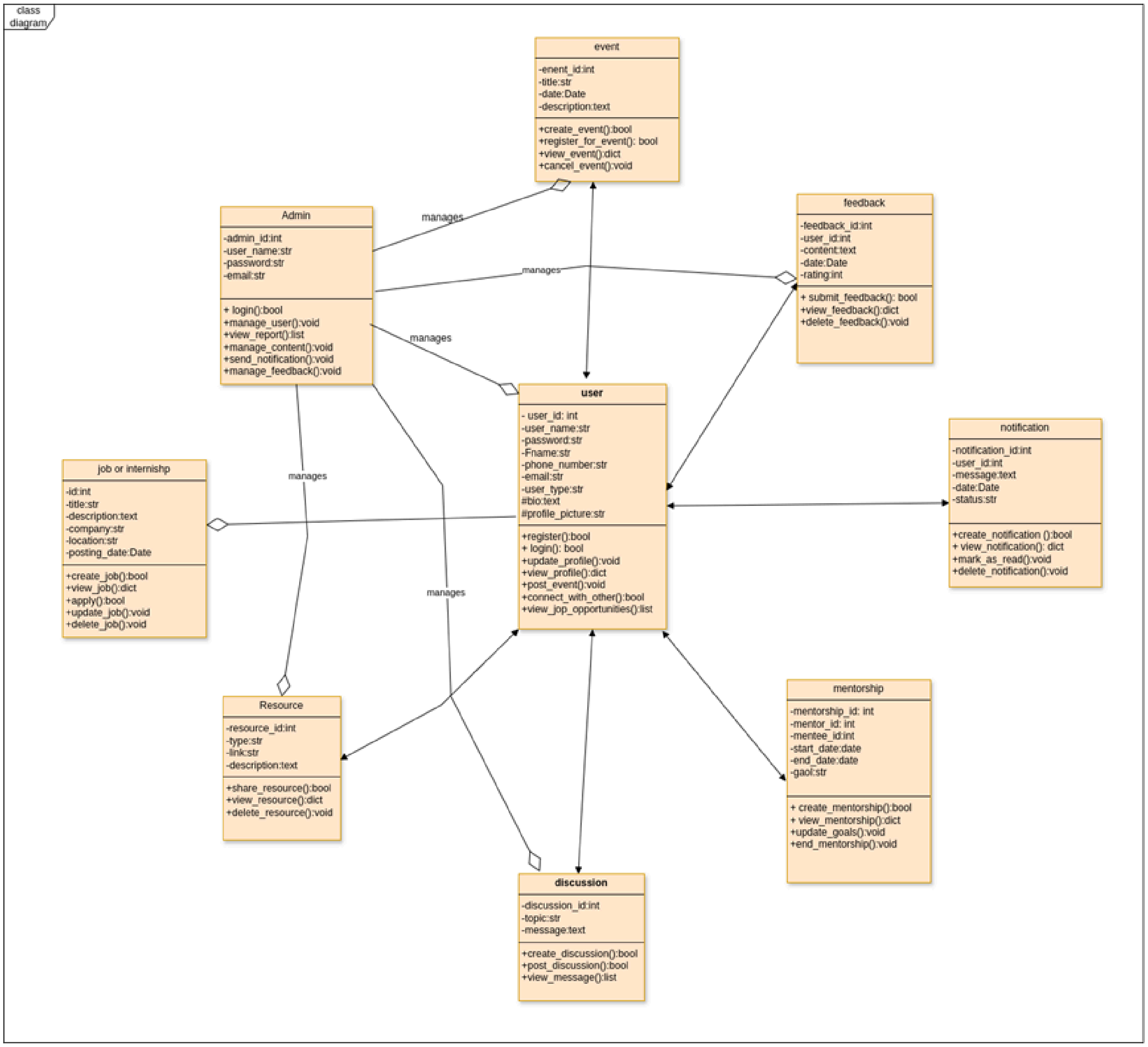


Figure : Class diagram

3.5.3. Dynamic model

The dynamic model delineates the behavior and interactions of various components or features of the project. Additionally, it illustrates the interactions and state transitions within the system. It focuses on the behavior of the system and how it responds to events or changes in state. This can include:

- □ State Changes: How an object transitions from one state to another.

- ❑ Events: Actions or occurrences that trigger changes.
- ❑ Interactions: How different parts of the system communicate with each other.

Here's a conceptual breakdown for ASTU ALUMNILINK:

1. User Signup

- Initial State: User is on the signup page.
- Event: User fills out the registration form (e.g., name, email, password, role).
- Action:
  - ❑ Validate user inputs.
  - ❑ Create a new user record in the database.
  - ❑ Send confirmation email (if applicable).
- State Change:
  - ❑ If validation succeeds: Redirect user to the login page with a success message.
  - ❑ If validation fails: Display error message (e.g., "Email already exists").
- Final State: User is either redirected to the login page or remains on the signup page with an error message.

2. User Login

- ❖ Initial State: User is on the login page
- ❖ Event: User enters email and password and submits the form.
- ❖ Action:
  - ❑ Authenticate user credentials.
  - ❑ Retrieve user data from the database.
- ❖ State Change:
  - ❑ If credentials are valid: Redirect user to their dashboard.
  - ❑ If credentials are invalid: Display error message (e.g., "Incorrect email or password").
- ❖ Final State: User is either on the dashboard or remains on the login page with an error message.

3. User Profile Edit

- ❖ Initial State: User is on their profile page.
- ❖ Event: User clicks the "Edit Profile" button.
- ❖ Action:
  - ❑ Load current profile details into editable form fields.
  - ❑ User updates information such as name, bio, skills, contact details, or profile picture.
- ❖ State Change:
  - ★ If the user submits valid updates:
    - ❑ Save changes to the database.

        - Refresh the profile page with updated information.
        - Display a success message (e.g., "Profile updated successfully!").
    - If the user submits invalid data:
        - Display an error message (e.g., "Invalid email format" or "Required fields are missing").
    - If the user cancels editing:
        - Discard changes and return to the profile view without any updates.
- Final State:

Profile is either updated and displayed with the new details, or remains unchanged with the original data.

4. Sending a Connection Request

- Initial State: User is on dashboard profiles.
- Event: User clicks the "Connect" button on a target profile.
- Action:
    - Save the connection request in the database.
    - Notify the recipient of the request.
- State Change:
    - Connection request is set to Pending.
    - The sender sees a "Request Sent" status on the profile.
- Final State: The connection request is in Pending state until the recipient responds.

5. Accepting a Connection Request

- Initial State: User views pending connection requests.
- Event: User clicks "Accept" on a connection request.
- Action:
    - Update the connection status to Accepted in the database.
    - Notify the sender that the request was accepted.
- State Change:
    - Connection status changes to Connected for both users.
    - The sender and recipient can now interact on the platform.
- Final State: Connection established between the two users.

6. Messaging

- Initial State: User is on the messaging page or viewing their inbox.
- Event: User types a message and clicks "Send."
- Action:

    - □ Validate the message content.
    - □ Save the message to the database.
    - □ Notify the recipient of the new message.
- ❖ State Change:
    - □ If message is sent successfully: Update the conversation thread.
    - □ If message fails: Display error message (e.g., "Message not sent").
- ❖ Final State: Message is either delivered to the recipient or remains unsent.

7. Posting a Job

- ❖ Initial State: User (e.g., company or faculty) is on the job posting page.
- ❖ Event: User fills out job details and clicks "Post Job."
- ❖ Action:
    - □ Validate the job post details.
    - □ Save the job post in the database.
    - □ Notify relevant users about the new job posting.
- ❖ State Change:
    - □ If validation succeeds: Job post is set to Published.
    - □ If validation fails: Display error message (e.g., "Required fields missing").
- ❖ Final State: Job is either published or remains in draft state.

8. Mentorship Program

- ❖ Initial State: User (e.g., student or alumni) is on the mentorship page.
- ❖ Event:
    - □ For Mentees: User dashboard available mentors and sends a mentorship request.
    - □ For Mentors: User sets availability and preferences for mentoring.
- ❖ Action:
    - ★ Mentee:
        - □ Select a mentor and send a request.
        - □ Save the mentorship request in the database.
        - □ Notify the mentor of the request.
    - ★ Mentor:
        - □ Update availability and preferences in the database.
- ❖ State Change:
    - ★ If request is accepted by mentor:
        - □ Mentorship status changes to Active.
        - □ Both mentor and mentee can communicate and set schedules.
    - ★ If request is declined:
        - □ Notify the mentee and return to the mentorship page.

- ❖ Final State:
  - □ Mentorship is either active (mentor-mentee pair established) or remains in pending/declined status.

3.5.4. Sequence diagram

1. User Signup

This activity allows new users (alumni, students, or faculty) to create an account on the platform. Users fill out a registration form, which is validated by the system. If the data is valid, the system checks for duplicate entries in the database and registers the user. Upon successful signup, the user is redirected to the login page**.**

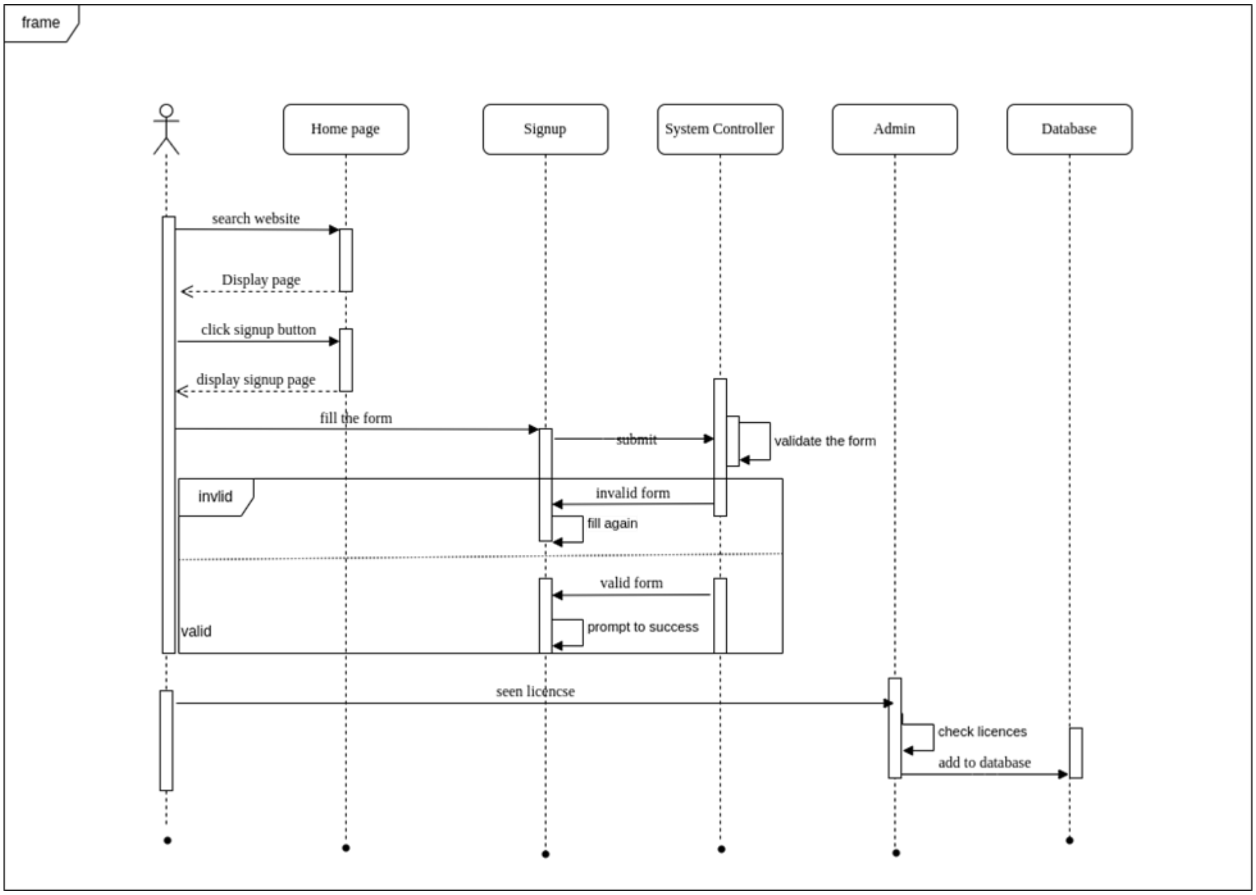


Figure: User signup sequence diagram

2. User Login

This activity handles the process of authenticating users who already have accounts. Users provide their email and password, which are verified against the database. Depending on the result, users are either redirected to their dashboard or shown an error message.

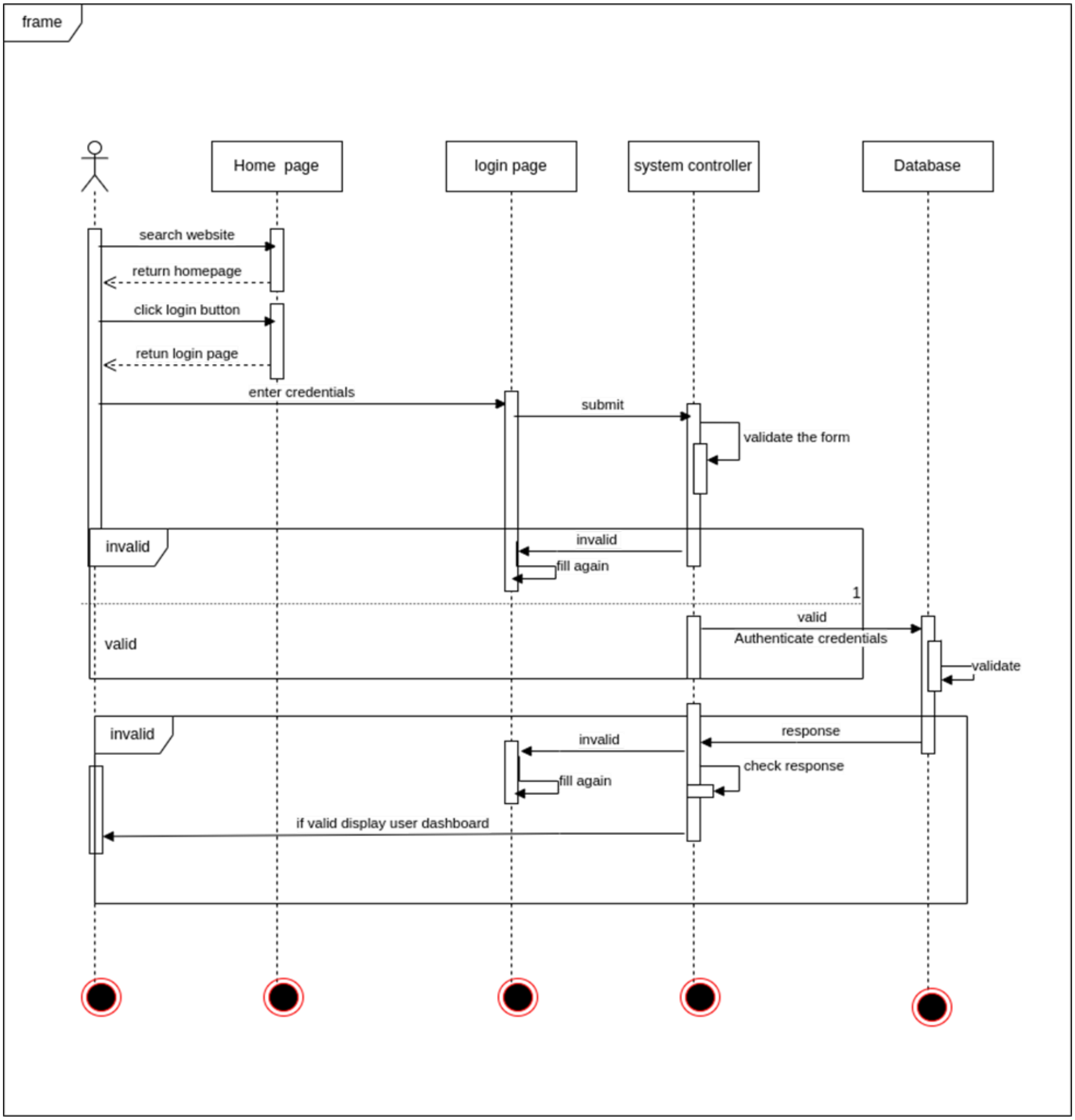


Figure : Login Sequnce diagram

3. User Profile Update

In this activity, users can update their personal information, such as name, email, educational background, work experience, skills, and profile picture. This helps maintain an up-to-date professional profile, making networking and mentorship interactions more effective.

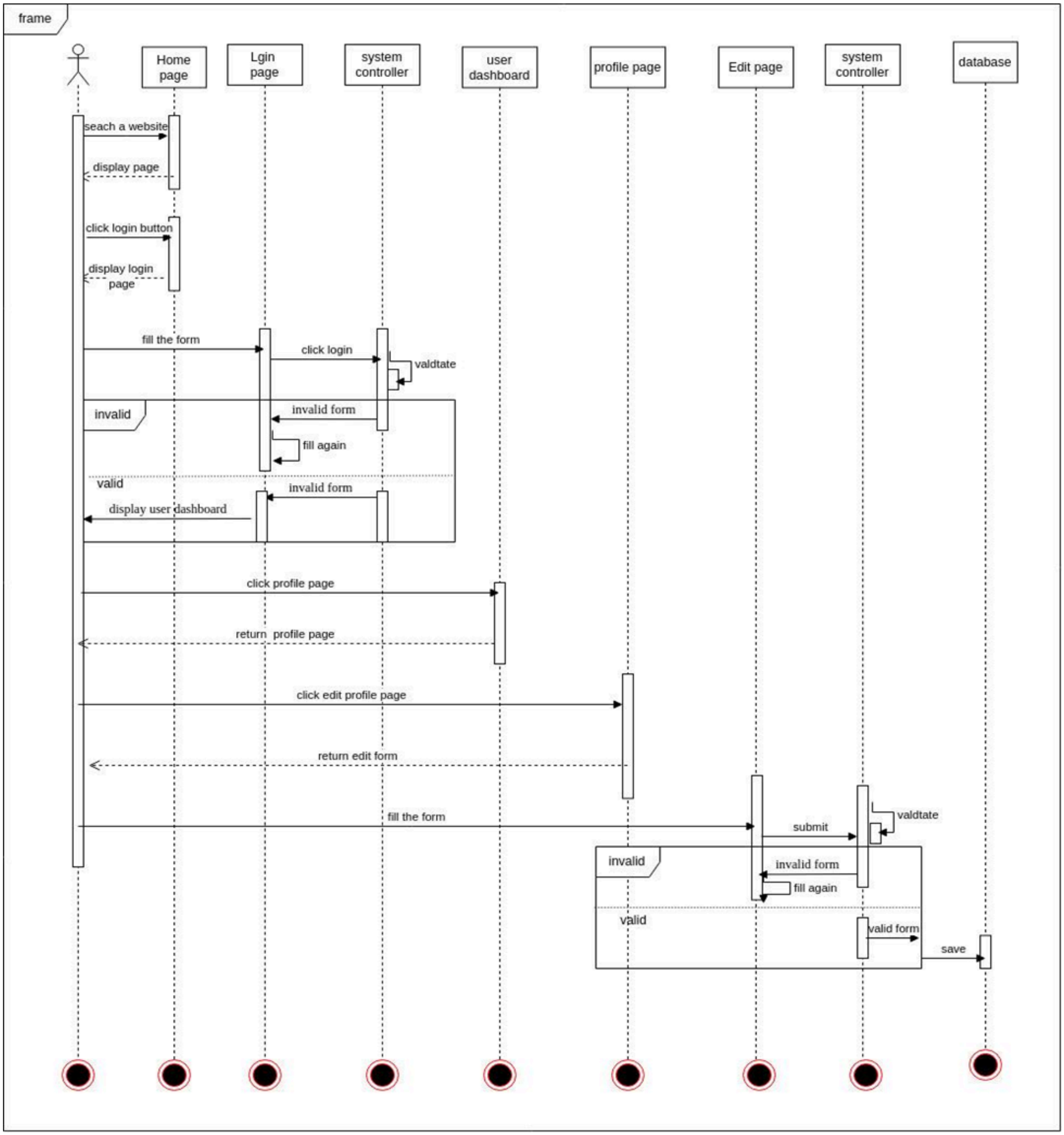


Figure : Edit profile sequence diagram

4. Mentorship Program

In this activity, users (mentees) can request mentorship from alumni or faculty (mentors). The system handles requests and allows mentors to accept or decline. Upon acceptance, a connection is established for further communication and collaboration.

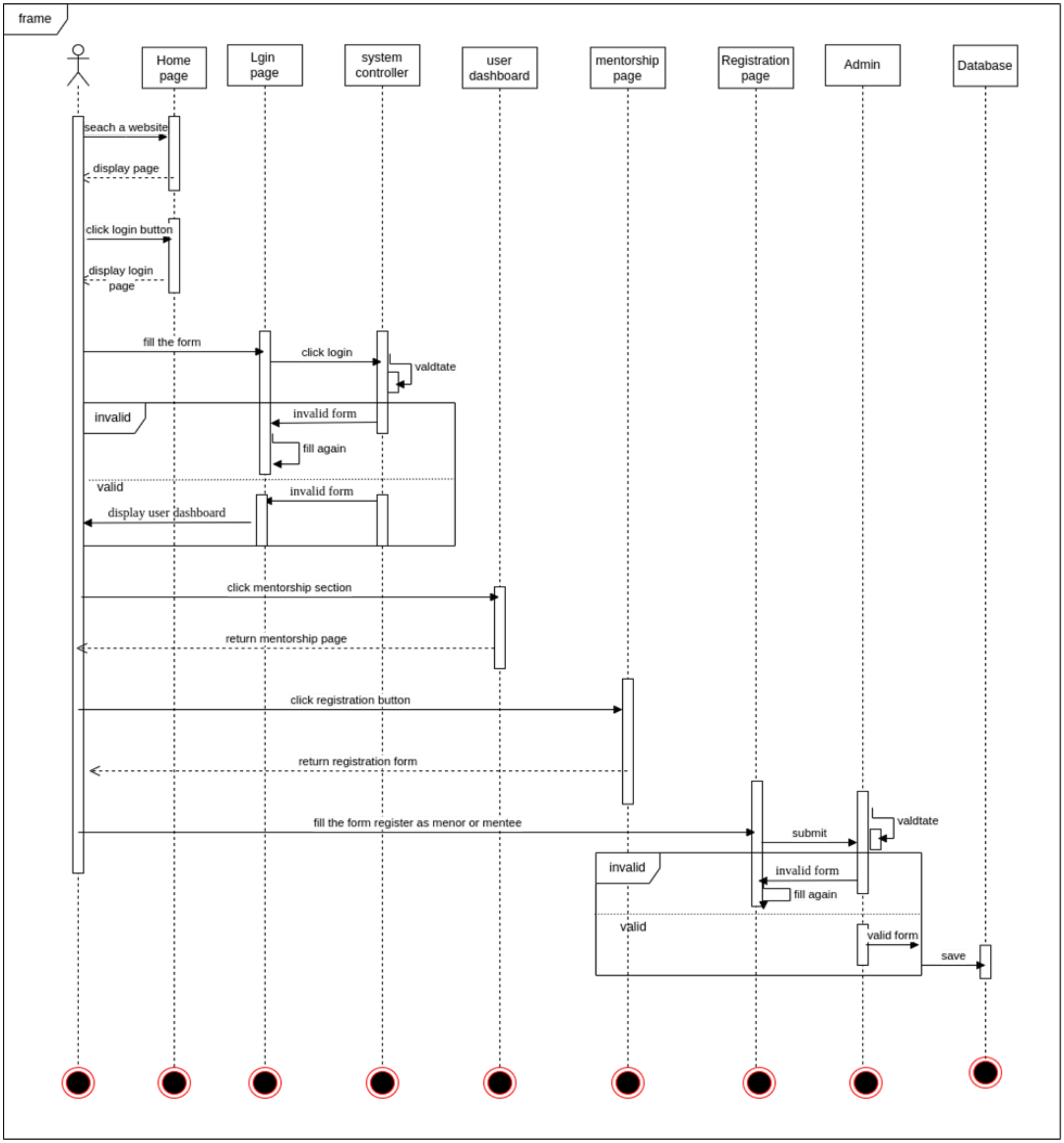


Figure : Mentorship program sequence diagram

5. Event Posting

In this activity, users (such as alumni, faculty, or admins) can post events like webinars, workshops, or job fairs on the platform. The system allows users to fill in event details (title, description, date, time, location, and registration link) and publishes the event for others to view. Users can also register for events and receive updates or reminders.

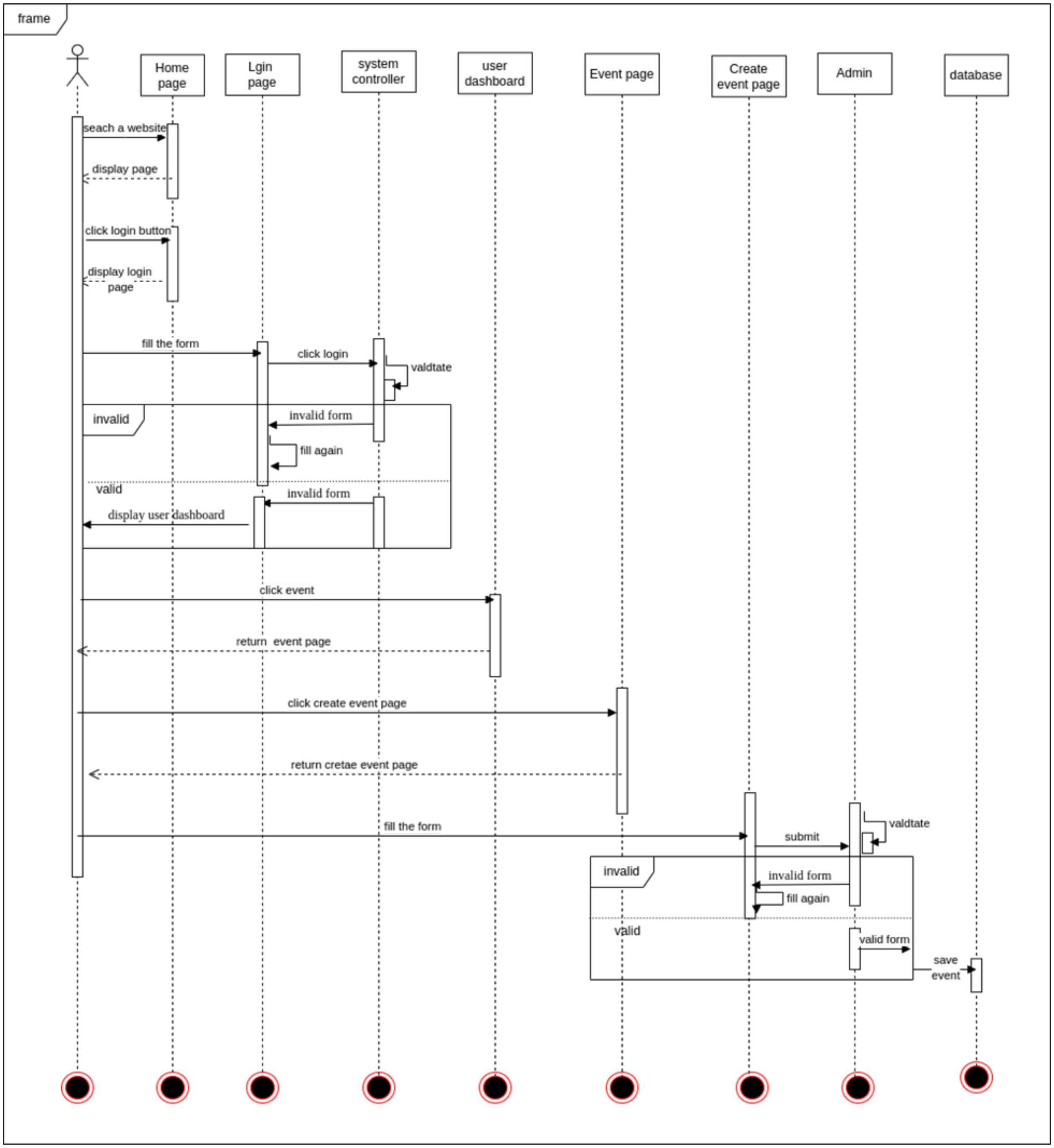


Figure : Create event Sequence Diagram

3.5.5 Activity diagram

An activity diagram describes a system in terms of activities. Activities are states that represent the execution of a set of operations. The completion of these operations triggers a transition to another activity. Activity diagrams are similar to flowchart diagrams in that they can be used to represent control flow and data flow. The following are the system's major activities represented by diagram.

1. Activity Diagram for User Registration

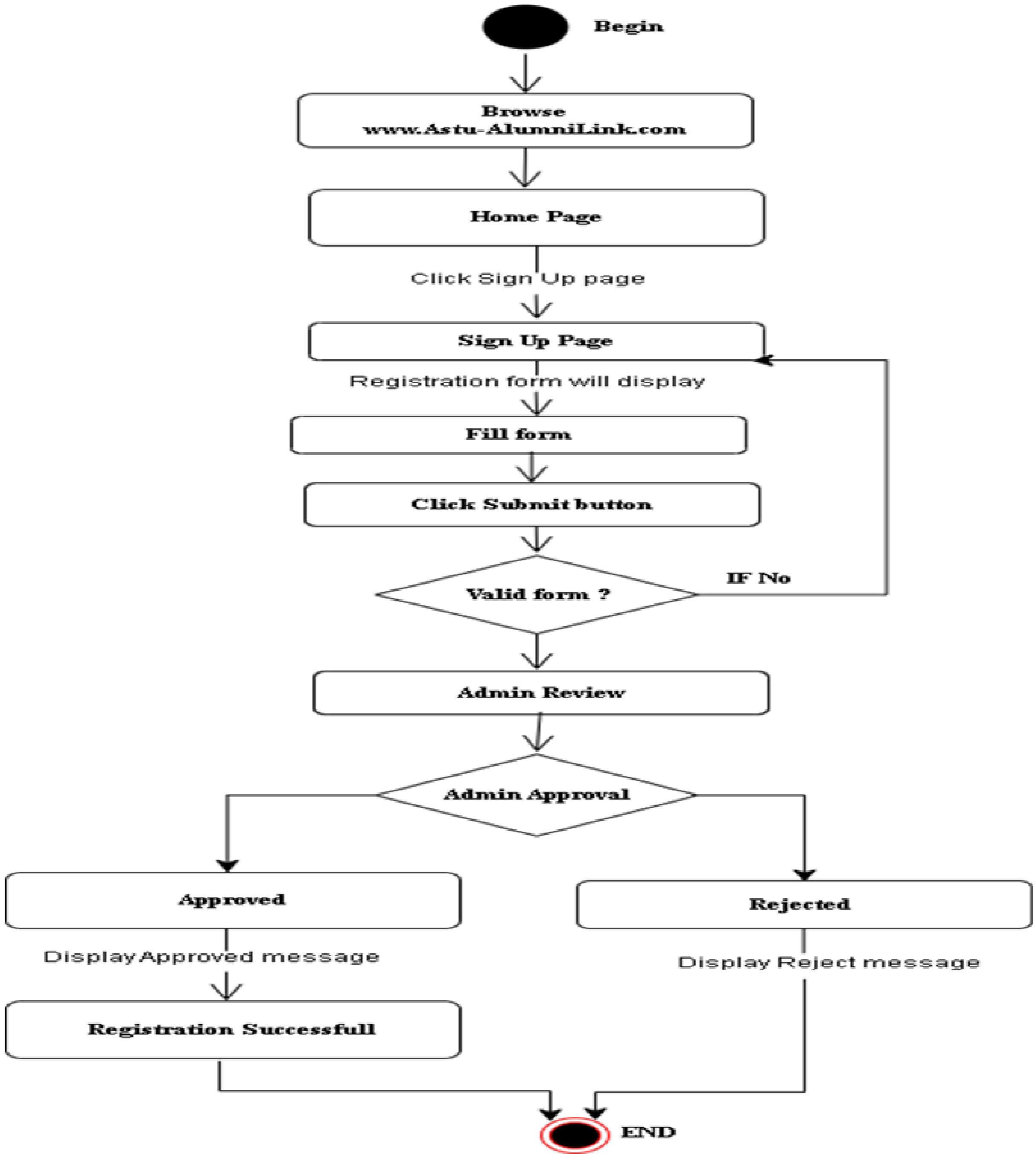
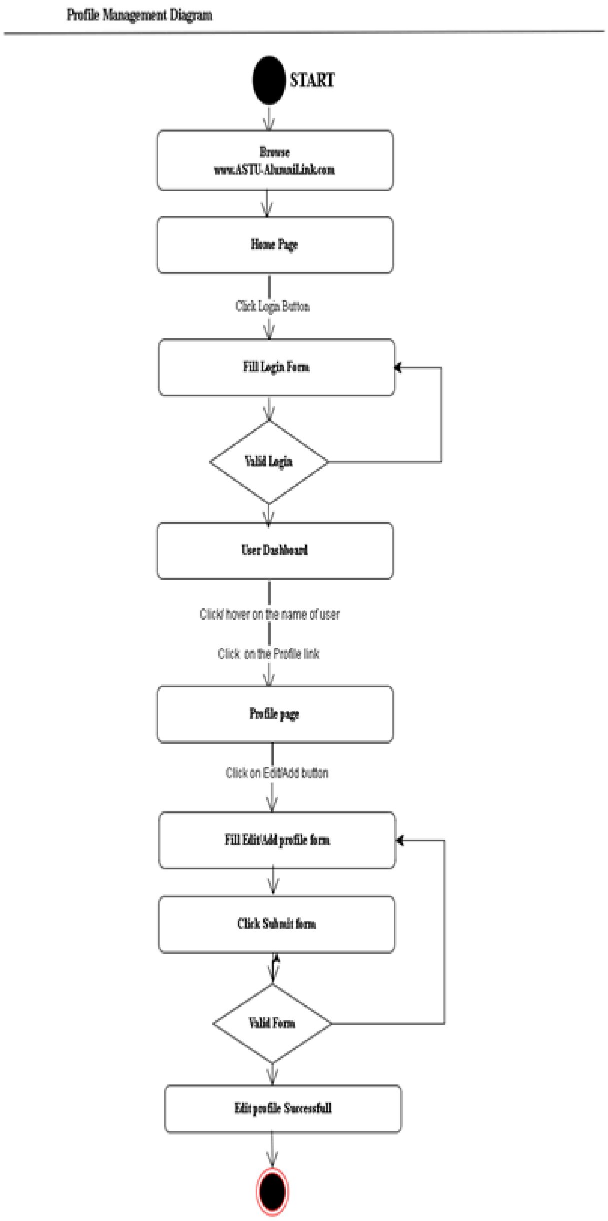
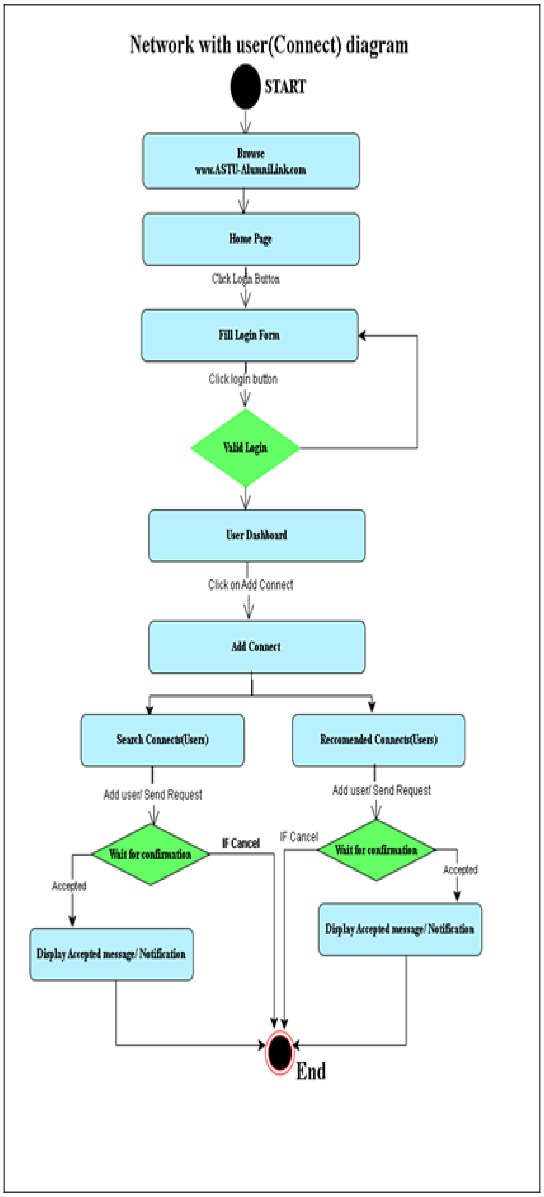


Figure : Activity Diagram for User Registration

2. Activity Diagram for Profile Management



3. Activity Diagram for Networking User



4. Activity Diagram for Create Event

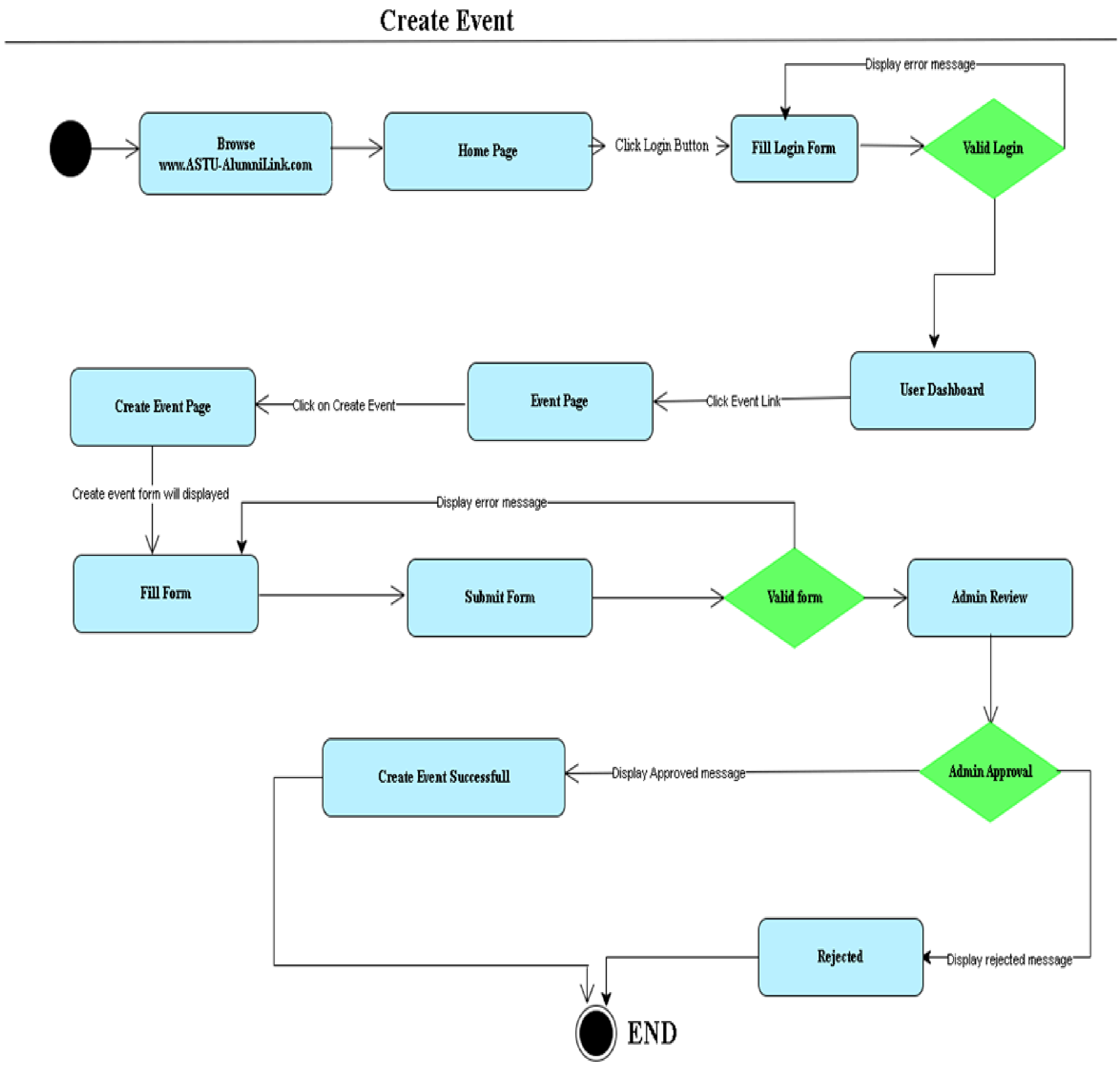


Figure : Activity Diagram for Create Event

5. Activity Diagram of Resource Request

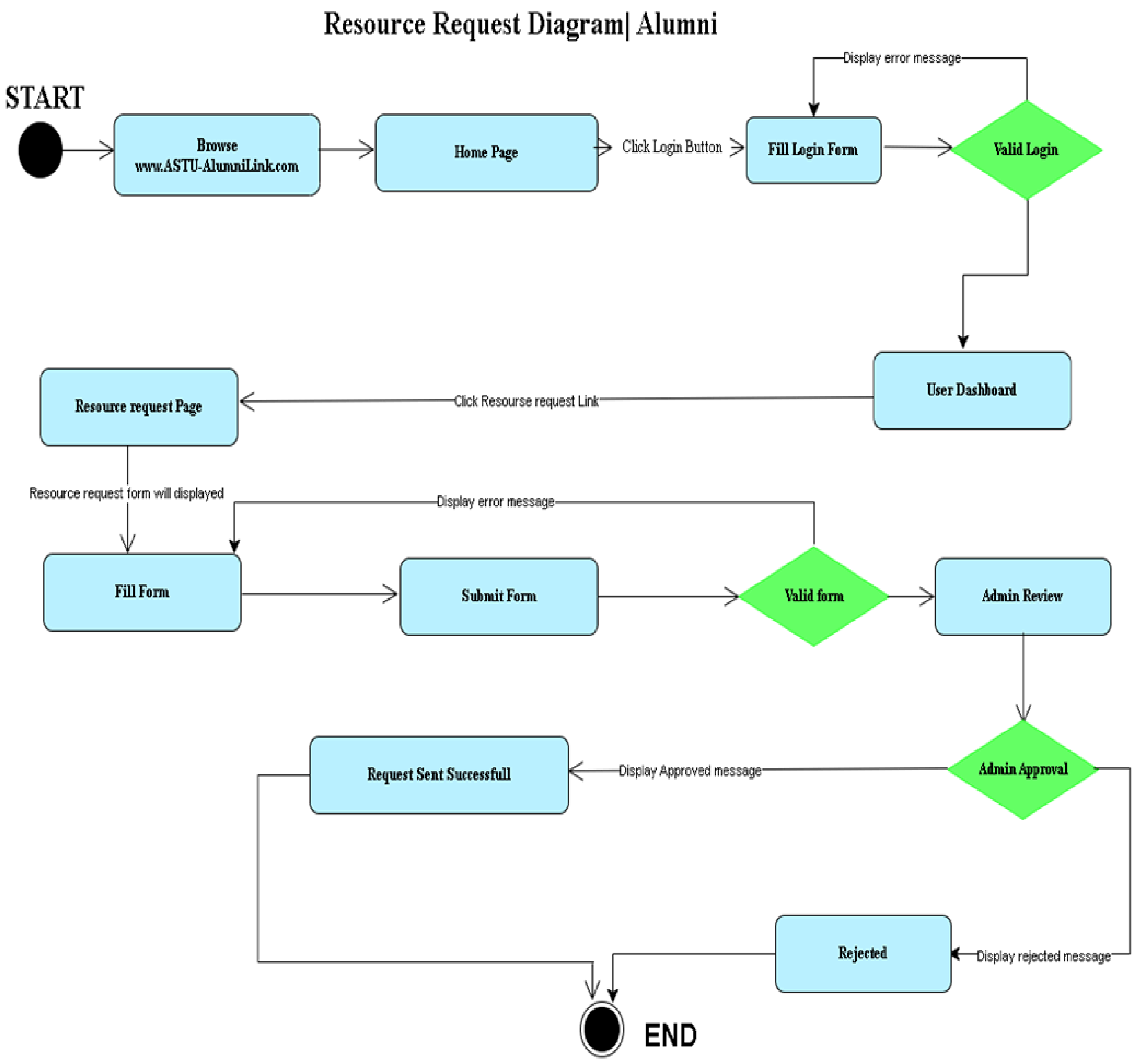


Figure : Resource request Activity diagram

6. Activity Diagram for Message /Communication

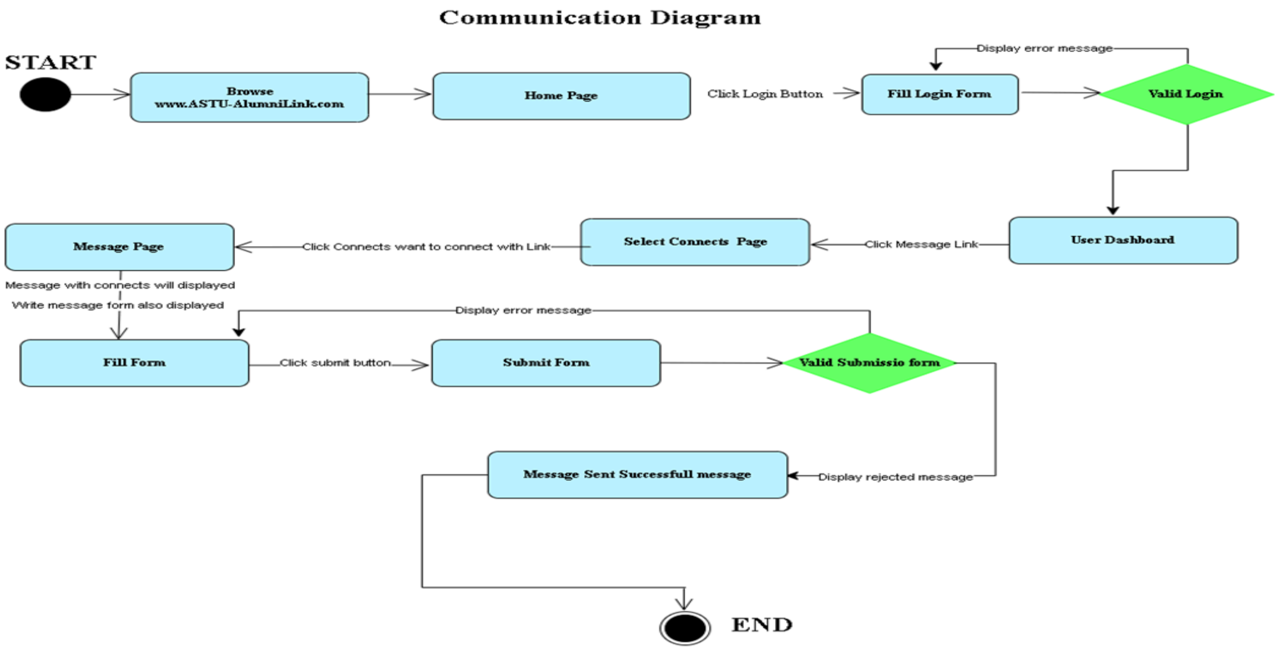


Figure : Activity diagram for message

7. Activity Diagram of Discussion Forum

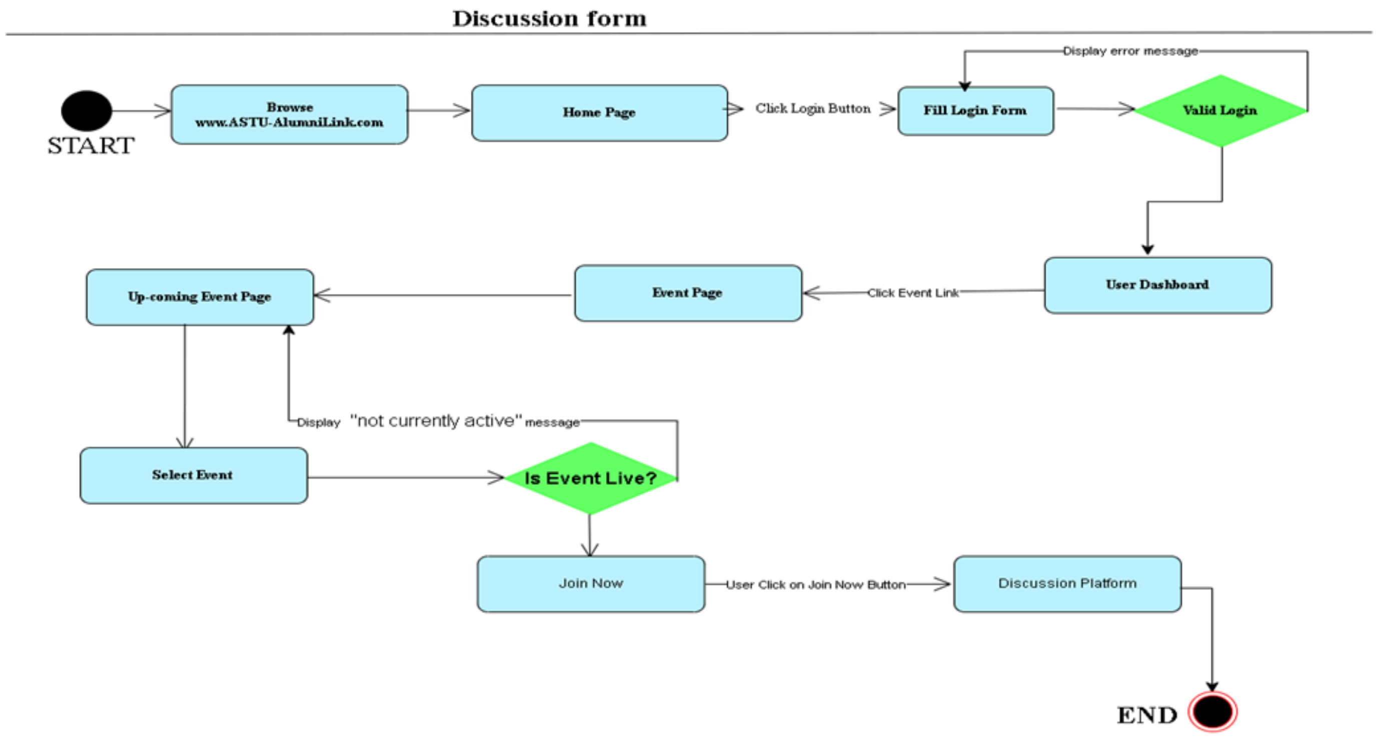


Figure : Discussion activity diagram

8. Activity Diagram of Search Alumni

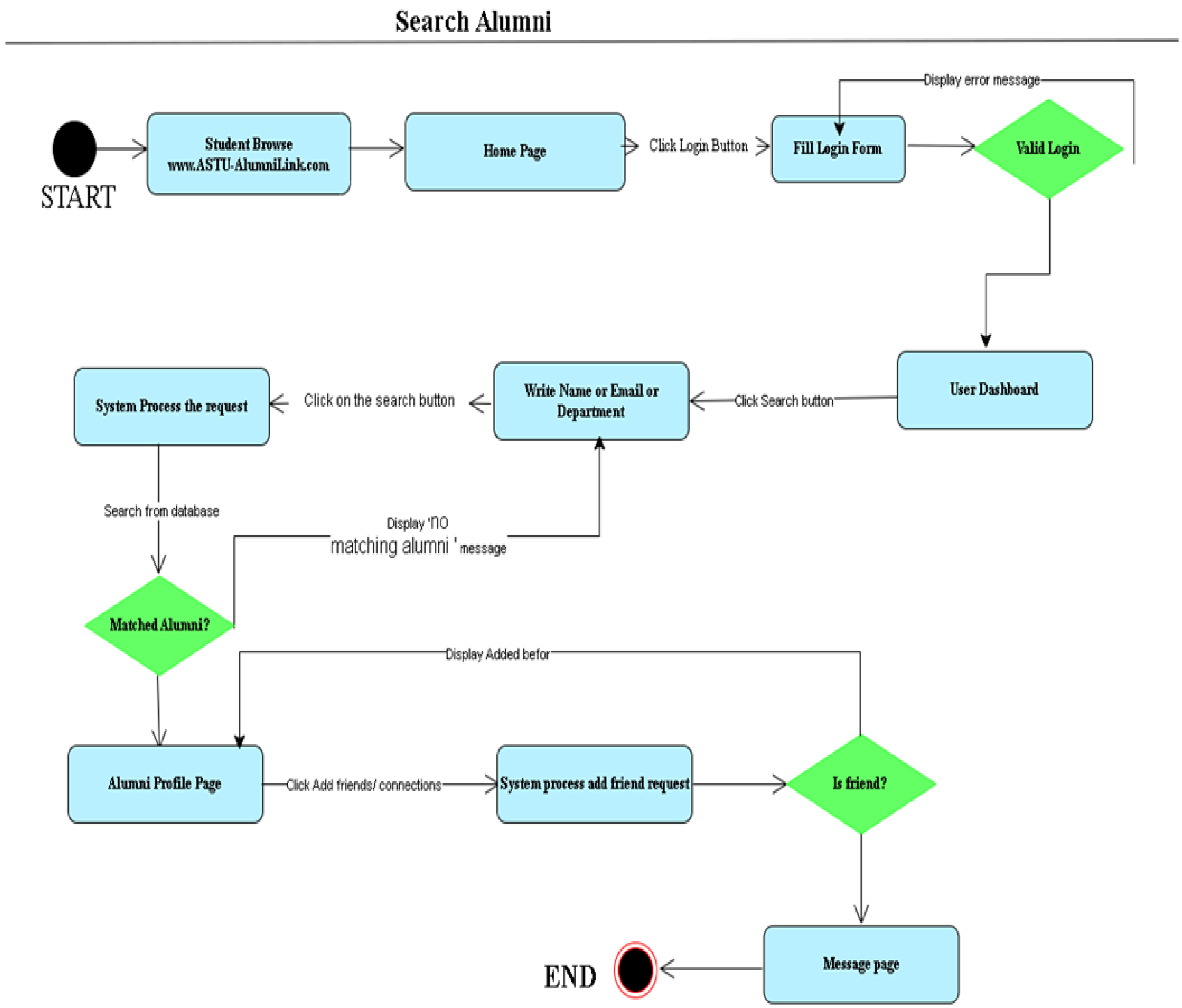

Search Alumni
START
Student Browse www.ASTU-AlumniLink.com
Home Page
Click Login Button
Fill Login Form
Valid Login
Display error message
User Dashboard
Click Search button
Write Name or Email or Department
Click on the search button
System Process the request
Search from database
Matched Alumni?
Display 'no matching alumni ' message
Alumni Profile Page
Click Add friends/ connections
System process add friend request
Is friend?
Display Added befor
Message page
END

Figure : Activity diagram for search alumni

9. Activity Diagram for User Feedback

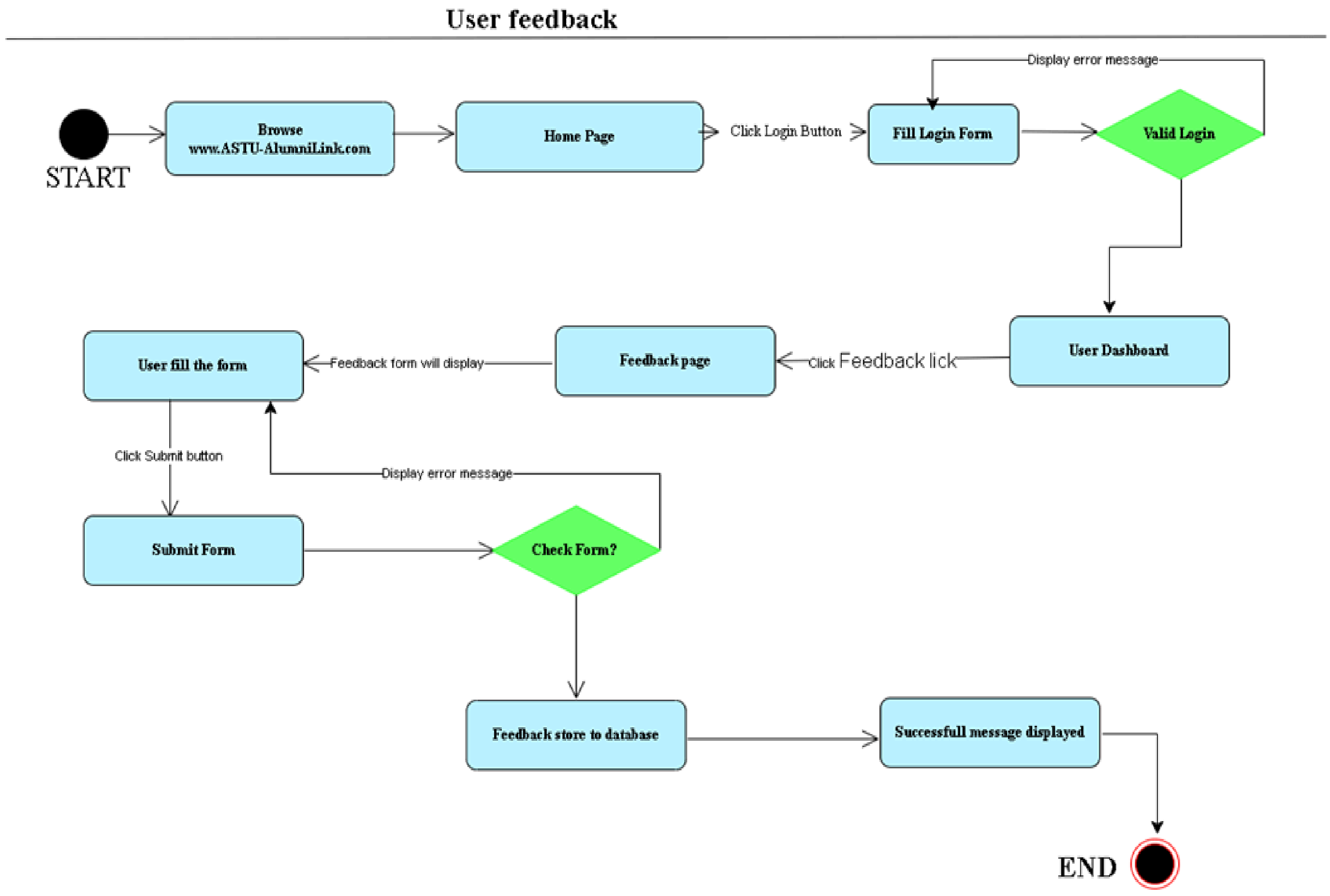


Figure : Activity diagram for user feedback

3.5.6 State chart diagram

A state chart diagram provides a visual representation of the states an object can be in throughout its lifecycle, along with the transitions between those states triggered by events. This type of diagram captures the dynamic behavior of a system by illustrating how an object responds to various inputs and changes over time. Each state reflects a specific condition of the object, while transitions indicate how the object moves from one state to another based on events or conditions. State chart diagrams are particularly useful for modeling complex behaviors, enabling the identification of critical states and transitions that influence the system's functionality. They help in understanding how different components interact and evolve, making them essential for designing robust and responsive systems.

1. State Chart for Sign up

2. State chart for User Login

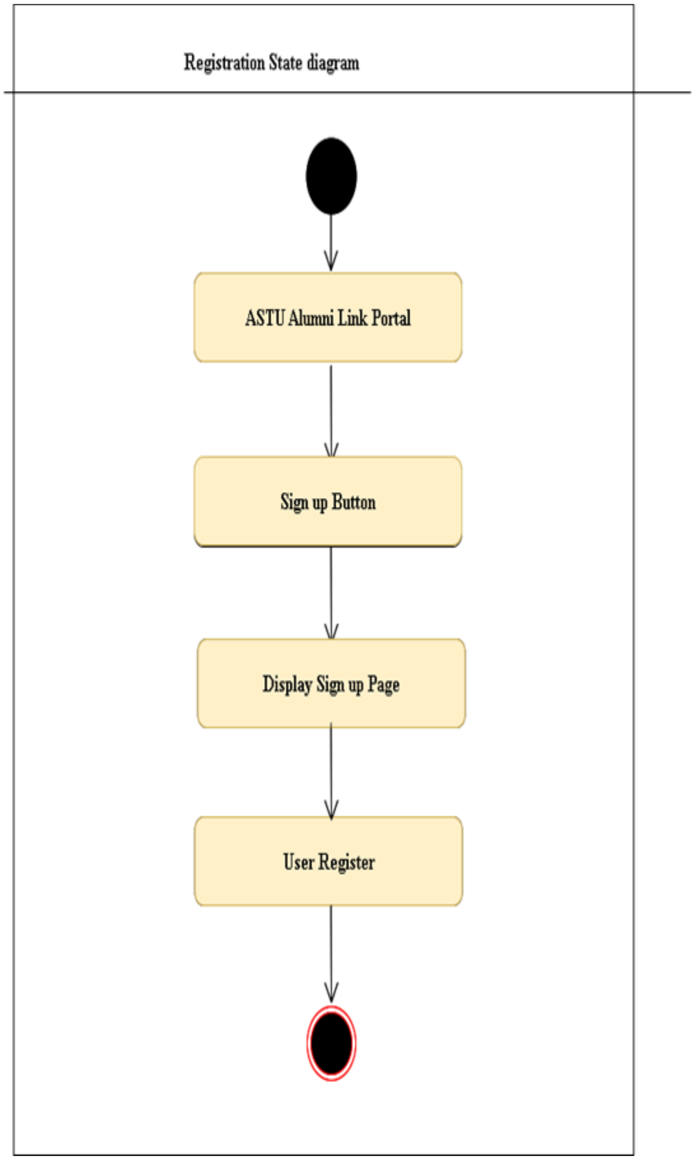


Figure :Registeration statechart Diagram

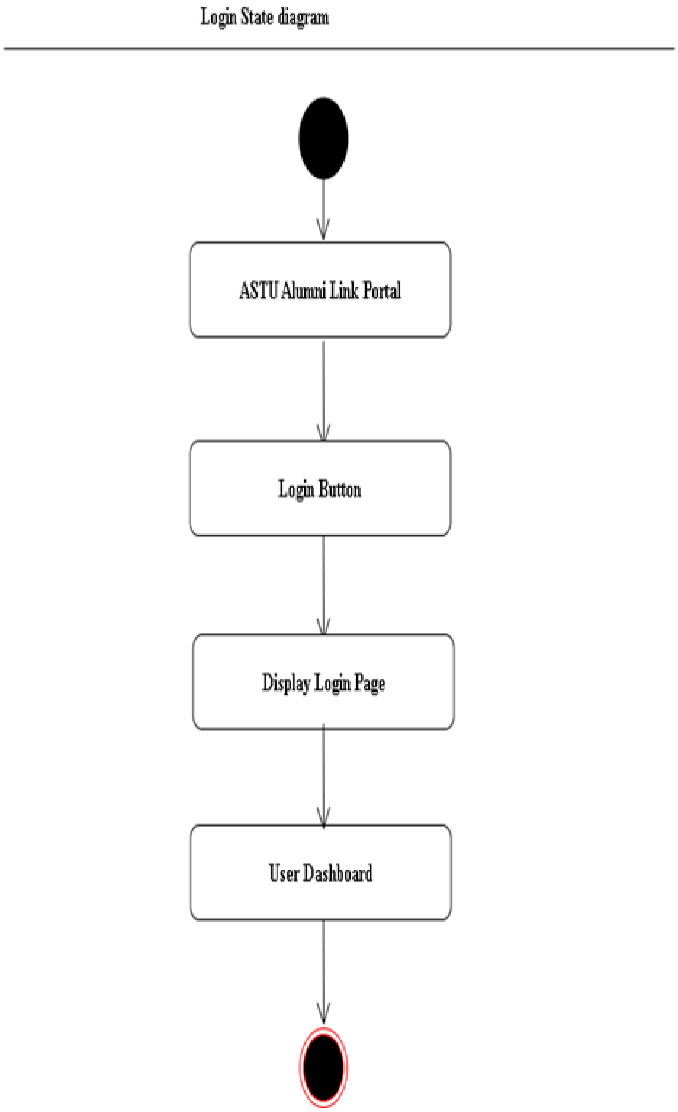


Figure :Registeration statechart diagram

3. State chart Diagram for Create Event

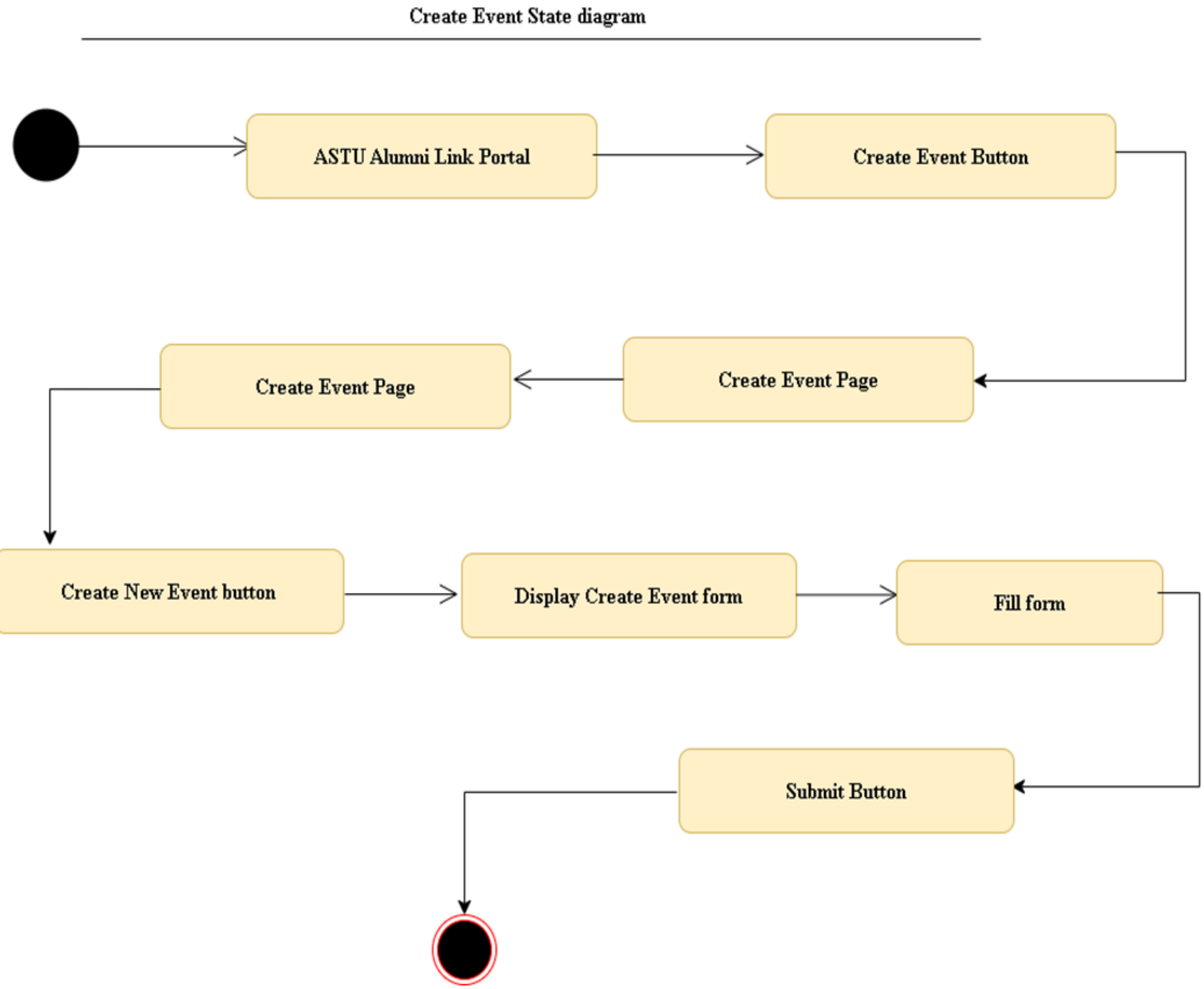


Figure : Create event state diagram

4. State Chart for Discussion Forum

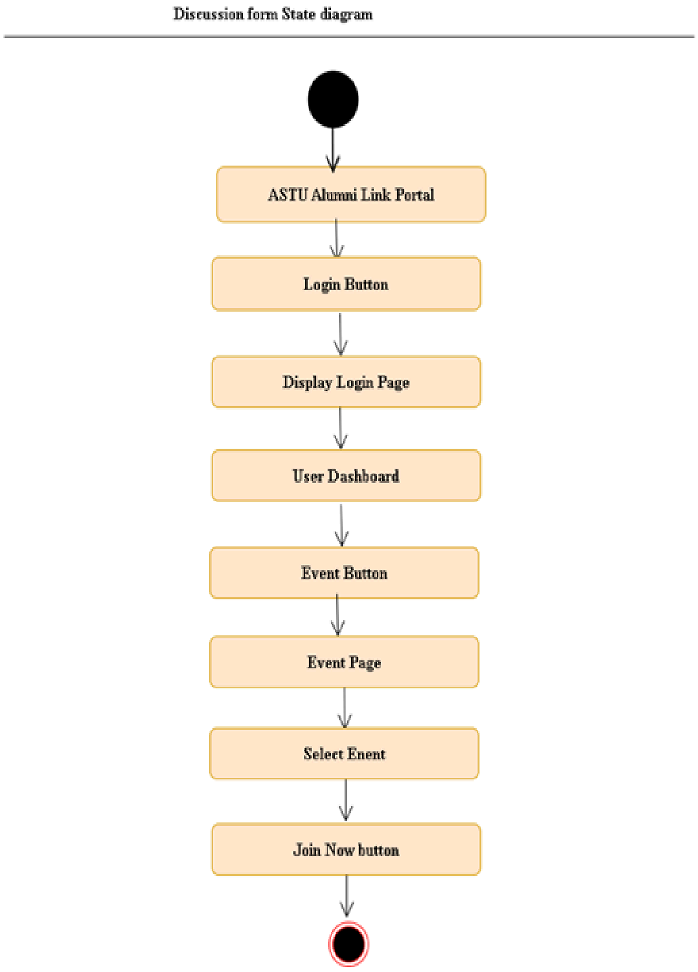


Figure : State chart of discussion forum

5. State chart for Add friend

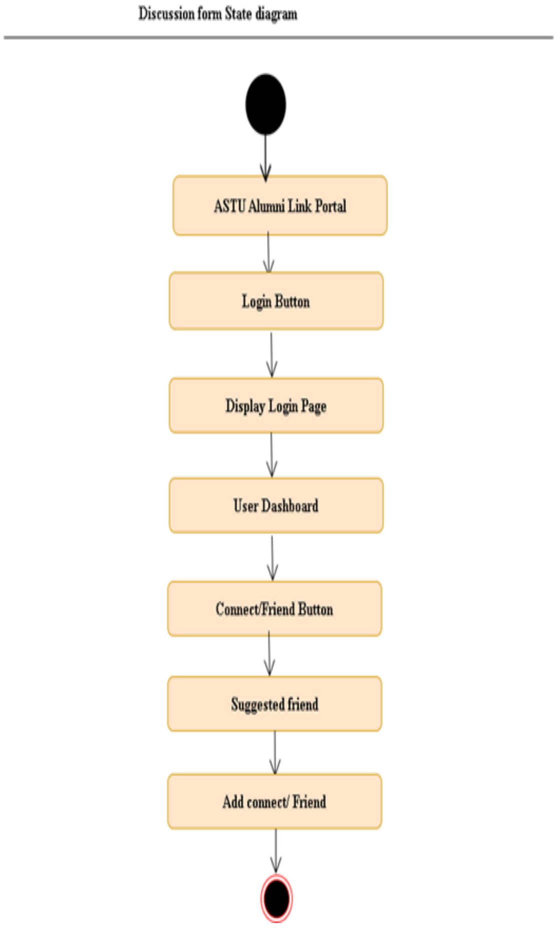


Figure : State chart of Add connects

6. State Diagram for Update Profile

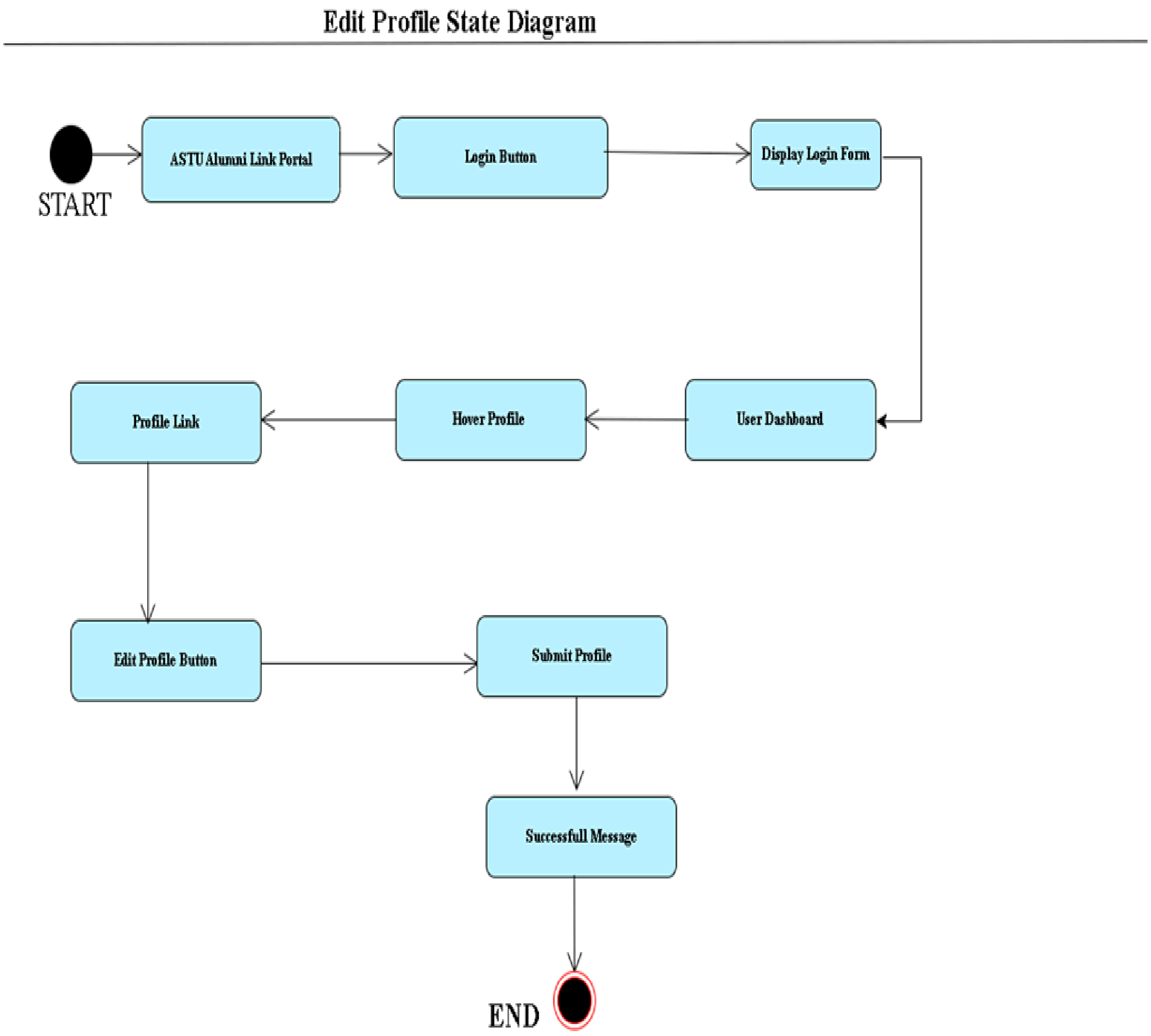


Figure :Edit profile state diagram

7. State Diagram for Search Alumni

Search Alumni State diagram

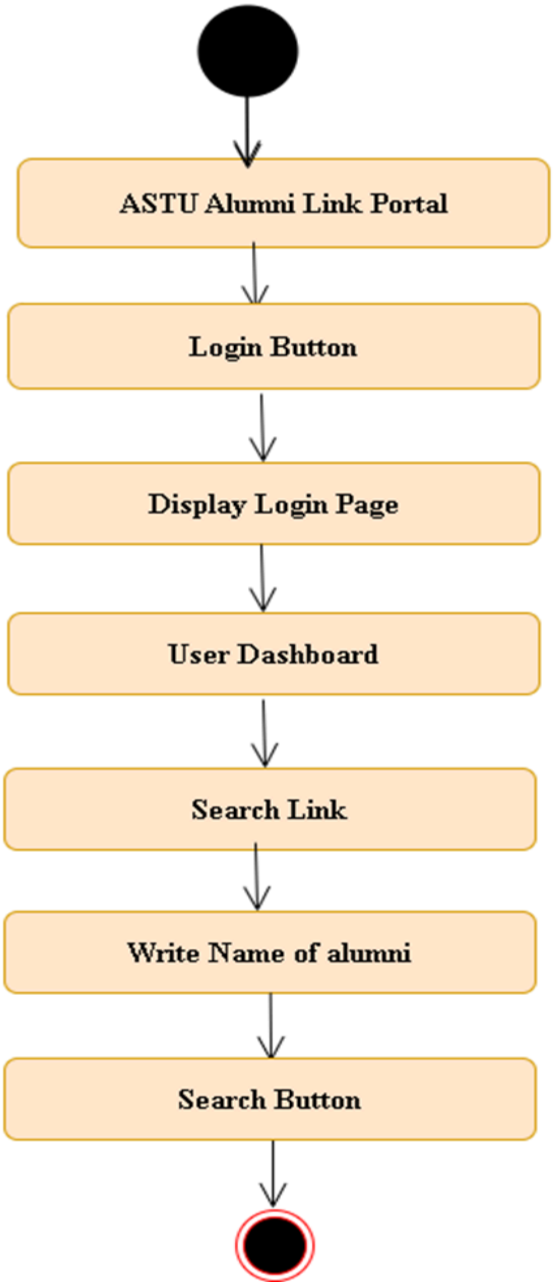


Figure : Search alumni state diagram