STUDENT ACTIVITY POINTS MANAGEMENT

Software Requirements Specification

GROUP - 9

Project lead: Veda Vijayshankar

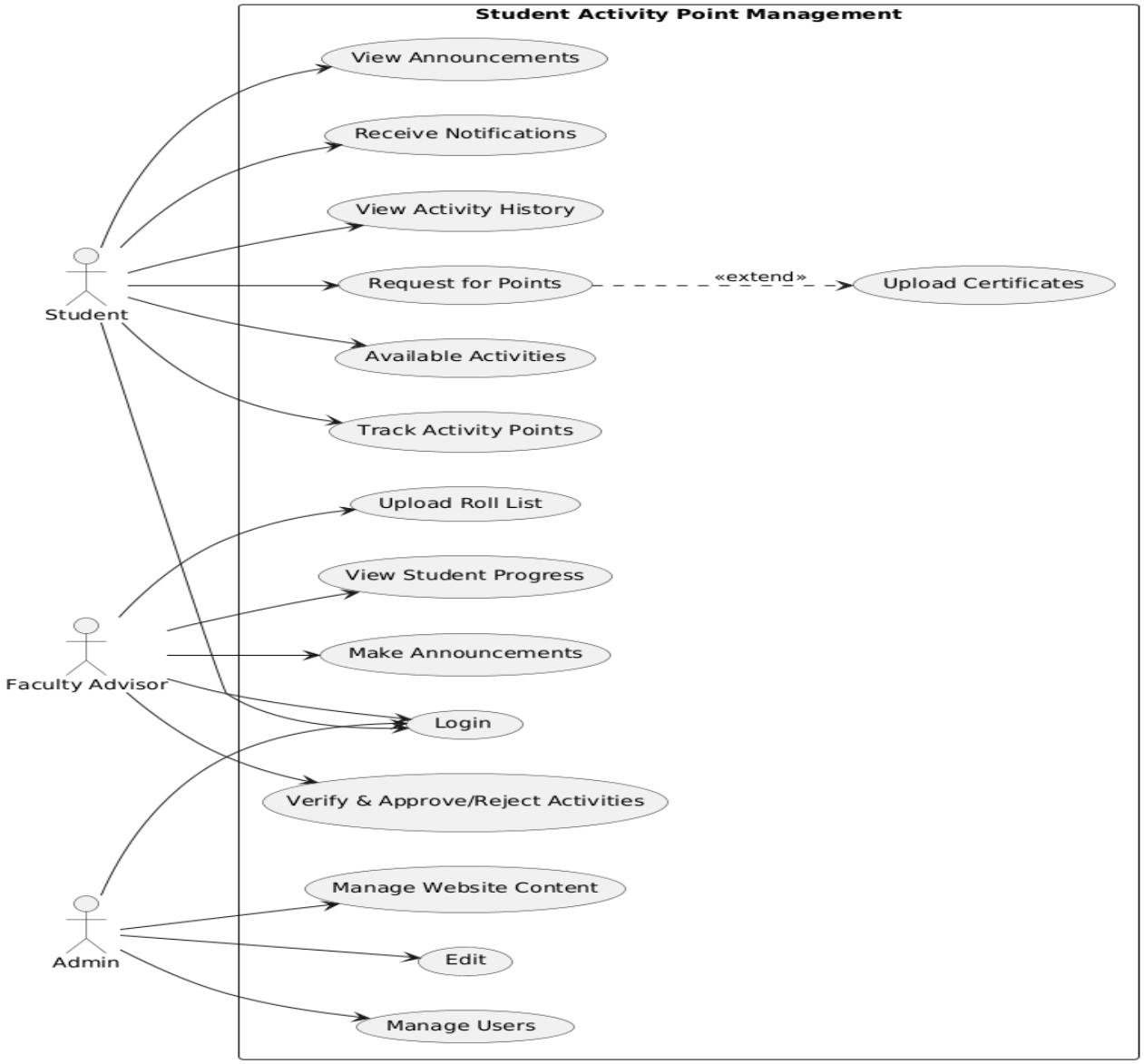
Group Members	Roll Number
Keerthana Lekshminarayanan	B220038CS
Panguluri Tharun Chowdary	B221097CS
Veda Vijayshankar	B220584CS

Revision History

Version	Date	Description of Changes
1.0	25-02-2025	SRS DOCUMENT

CONTENTS

USE CASE DIAGRAM



FUNCTIONAL REQUIREMENTS

1.1 USER REQUIREMENTS

F1: Students and faculty advisors must be able to log in using Google Authentication.

F2: Students should be able to request activity points by uploading certificates as proof of completed activities.

- Students must be able to enter activity details, including name, type, date, and organizer to request activity points.
- Duplicate activity submissions are prevented.

F3: The system must display a student's activity history, showing all previously submitted activities along with their status (Pending, Approved, or Rejected).
F4: Automated notifications must be sent to students regarding approvals, rejections and new announcements.
F5: Students should be able to view announcements posted by faculty advisors.
F6: A list of available activities that students can participate in to earn points must be accessible.
F7: Faculty advisors must have role-based access to the system.
F8: Faculty advisors should be able to verify, approve, or reject student activity submissions.
F9: They must be able to track student progress, including earned points, pending activities, and history.
F10: Faculty advisors should be allowed to publish announcements related to activities, deadlines, or updates.

- FA must provide title and body of announcements.
- Both FA and student can view announcements.

F11: Automated notifications must be sent to faculty advisors regarding new activity submissions or pending approvals.
F12: Admin must have exclusive access to manage user accounts of students and faculty advisors.
F13: The admin should have the authority to create, update, and modify platform guidelines.
F14: The admin should be responsible for uploading and managing the roll list for faculty advisors.
F15: Admin must be able to add or remove students as well as faculty advisors from the database when they join or leave the institution.
F16: Students should have access to their activity history, including status updates (Approved, Pending, Rejected).

1.2 SYSTEM REQUIREMENTS

F16: Reports on student progress and activity participation should be generated by the system.
F17: The system should validate activity submissions and allow students to upload proof in PDF format.
F18: Student records and uploaded certificates must be stored securely in the database.
F19: A secure authentication mechanism must be in place for student, faculty, and admin

logins.
F20: Faculty advisors should have access to create and publish announcements that will be visible to students.
F21: The platform must display a list of available activities that students can participate in which can be modified, added or deleted by Admins. F34: A categorized list of upcoming activities (mandatory or optional) should be displayed.
F22: The system should display total activity points along with a category-wise breakdown for students.
F23: Student details, including Name, Roll Number, and Department, should be maintained in the system.
F24: Faculty information, such as Name, Email, and Department, must be stored securely.

NON FUNCTIONAL REQUIREMENTS

4.1 PERFORMANCE REQUIREMENTS

- The system should handle at least 50 concurrent users smoothly.
- The dashboard should load within 3-5 seconds under normal conditions. Overall, the performance of the system will be fast and accurate.

4.2 SAFETY AND SECURITY REQUIREMENTS

1. User Identity Authentication: The system should require authentication using Spring Security with unique usernames and passwords for each user, preventing unauthorized access to sensitive data.
2. Data Privacy: Data should be accessible only to authorized users.
3. Data Encryption: All sensitive data should be encrypted. For eg:- using BCrypt for password hashing.

- These security measures ensure that the system meets industry standards for privacy, data protection, and secure access.

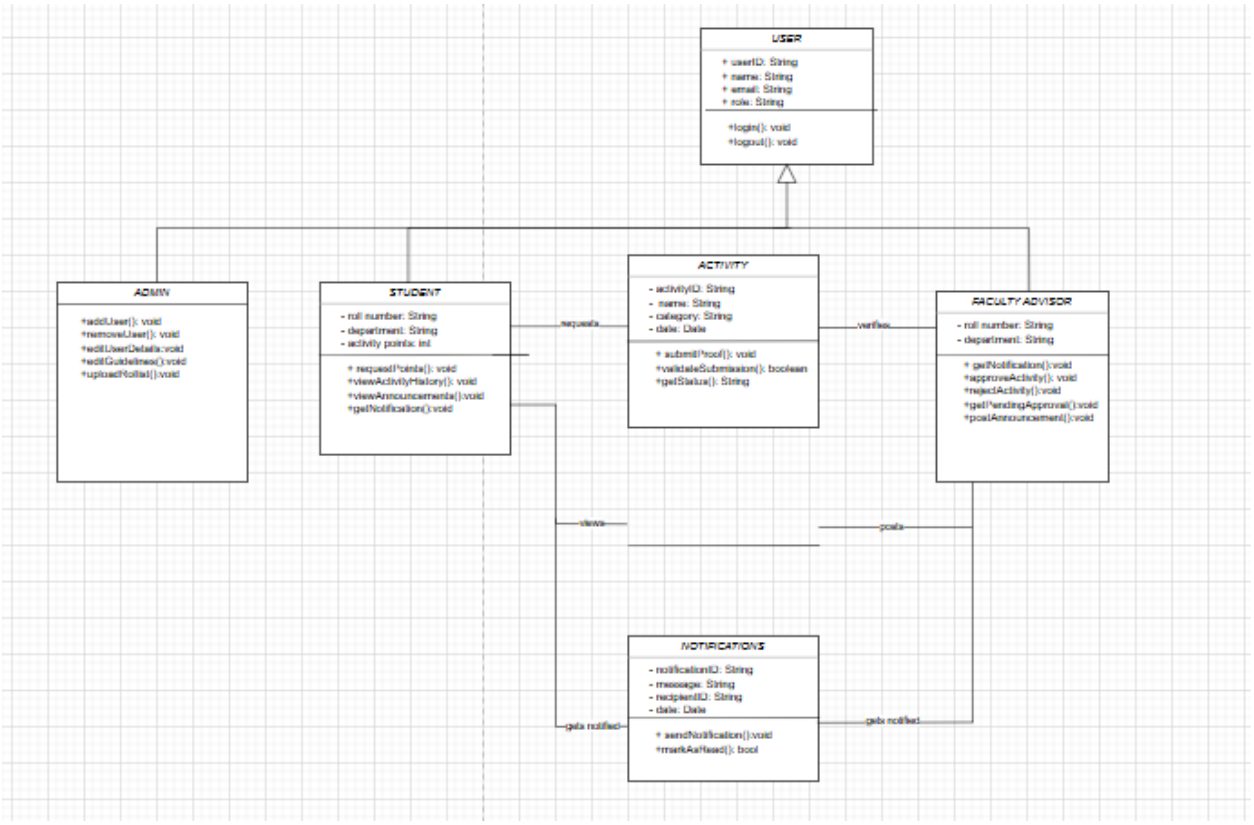
4.3 SOFTWARE QUALITY ATTRIBUTES

1. Reliability
   - The system will not lag and will provide instant and accurate results to all users.
2. Adaptability
   - The system is initially designed to support up to 200 users but should allow future upgrades to accommodate growth.
3. Maintainability
   - The backend code should follow a modular architecture, with separate controllers, services, and repositories for better maintainability.
   - Proper comments and documentation should be included to facilitate future modifications and debugging.
4. Portability
   - The system can be deployed on any machine, ensuring flexibility across different environments.
5. Compliance

- The system should adhere to basic data privacy regulations, ensuring that sensitive data is not publicly exposed.

6. Usability
   - The UI should be fully responsive, ensuring optimal usability across laptops, tablets, and mobile devices.
   - Basic help documentation should be provided to assist users in navigating the system effectively.

CLASS DIAGRAM



DATABASE DESIGN