3.3 Design Goals

The design of the internship management system aims to achieve the following objectives:

Maximize System Usability: Ensure that the platform is user-friendly and intuitive for students, faculty, and company representatives.

Ensure Data Security: Implement robust security measures to protect sensitive internship-related data and ensure compliance with privacy regulations.

Optimize System Performance: Ensure that the system can handle a large number of concurrent users and data transactions efficiently.

Enhance Scalability: Design the system to accommodate future growth in the number of users and functionalities.

Improve Reliability: Ensure the system operates consistently and accurately, with minimal downtime and errors.

Maintainability: Develop the system with a modular architecture to facilitate easy updates, bug fixes, and feature enhancements.

Facilitate Seamless Communication: Enable effective communication and collaboration between students, universities, and host companies through integrated messaging and notification systems.

Streamline Administrative Tasks: Automate and simplify internship management tasks to reduce manual work and improve efficiency.

3.4 Current Software Architecture

Currently, St. Mary's University’s internship management operates through a manual and disparate system, characterized by the following components and interactions:

Orientation Sessions: Conducted manually to provide students with information about the internship process.

Student Internship Search: Students independently search for internships, often leading to mismatches and delays.

Host Company Assignment: The Career Office assigns companies to students if they fail to secure one within a set timeframe.

Internship Documentation and Communication: Communication and documentation are handled through physical letters and manual transmission of records.

Oversight and Monitoring: The university relies on manual check-ins and progress reports to monitor internships.

Evaluation and Assessment: Manual submission and evaluation of internship reports by faculty members.

Limitations of the Current Architecture

Fragmented Processes: The lack of a centralized system leads to inefficiencies and potential errors in communication and documentation.

Manual Workload: Significant manual effort is required for managing internships, leading to delays and administrative burden.

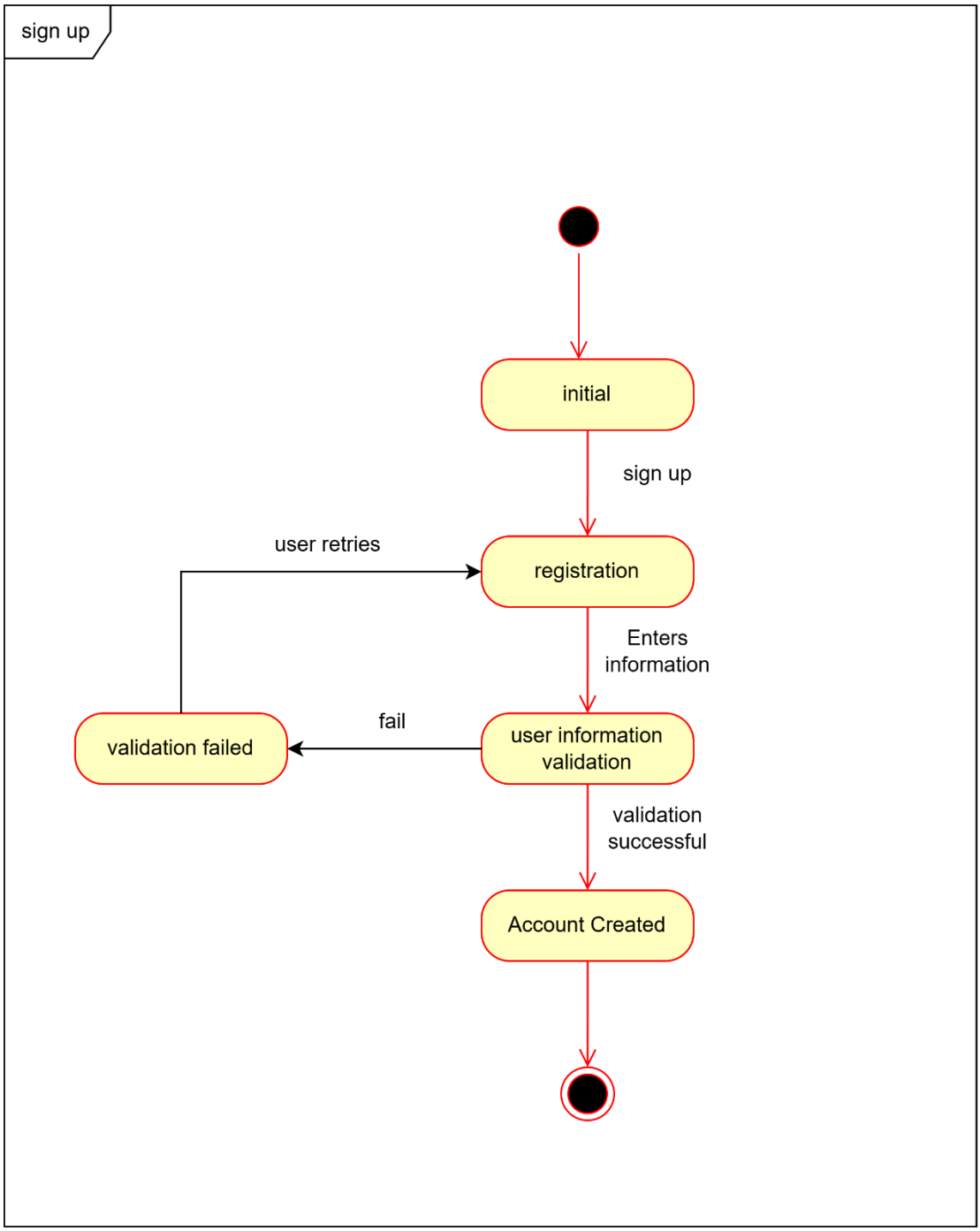
Random Assignments: Students are sometimes placed in internships that do not align with their interests or career goals.

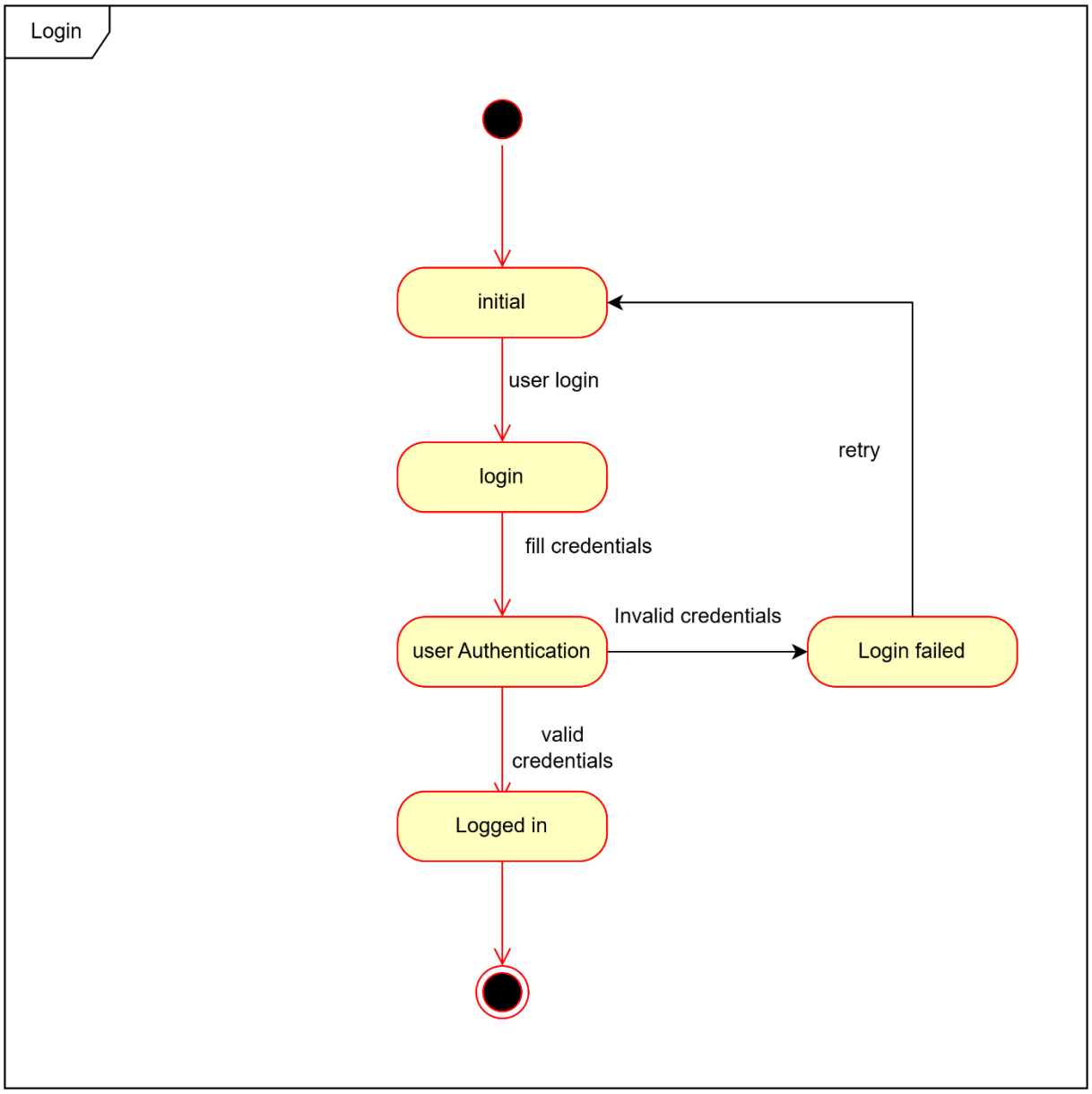
Data Inconsistencies: Manual transmission of records increases the risk of data loss, errors, and inconsistencies.

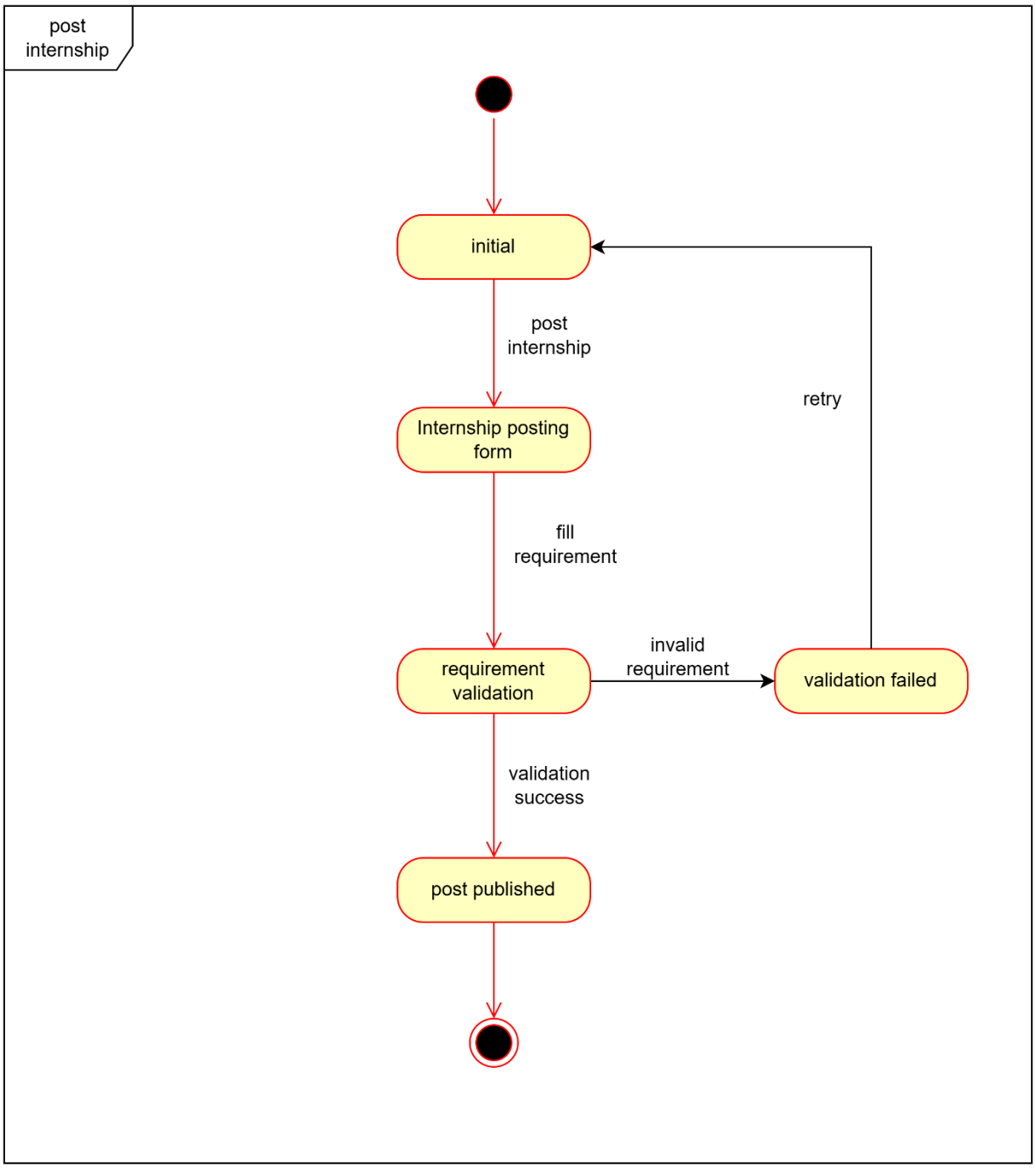
Limited Oversight: Difficulties in monitoring and tracking student progress during internships.

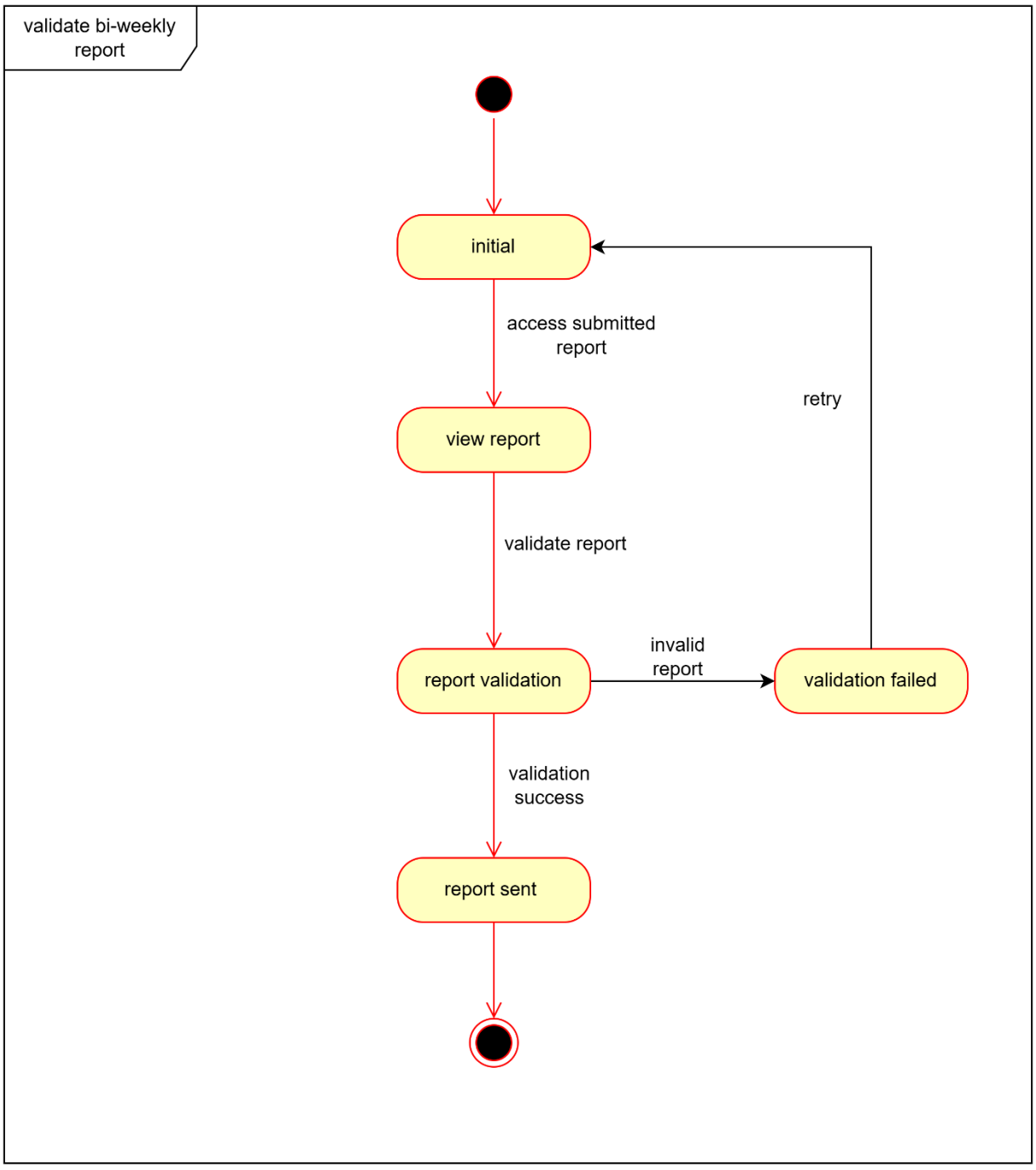
Need for a New Architecture

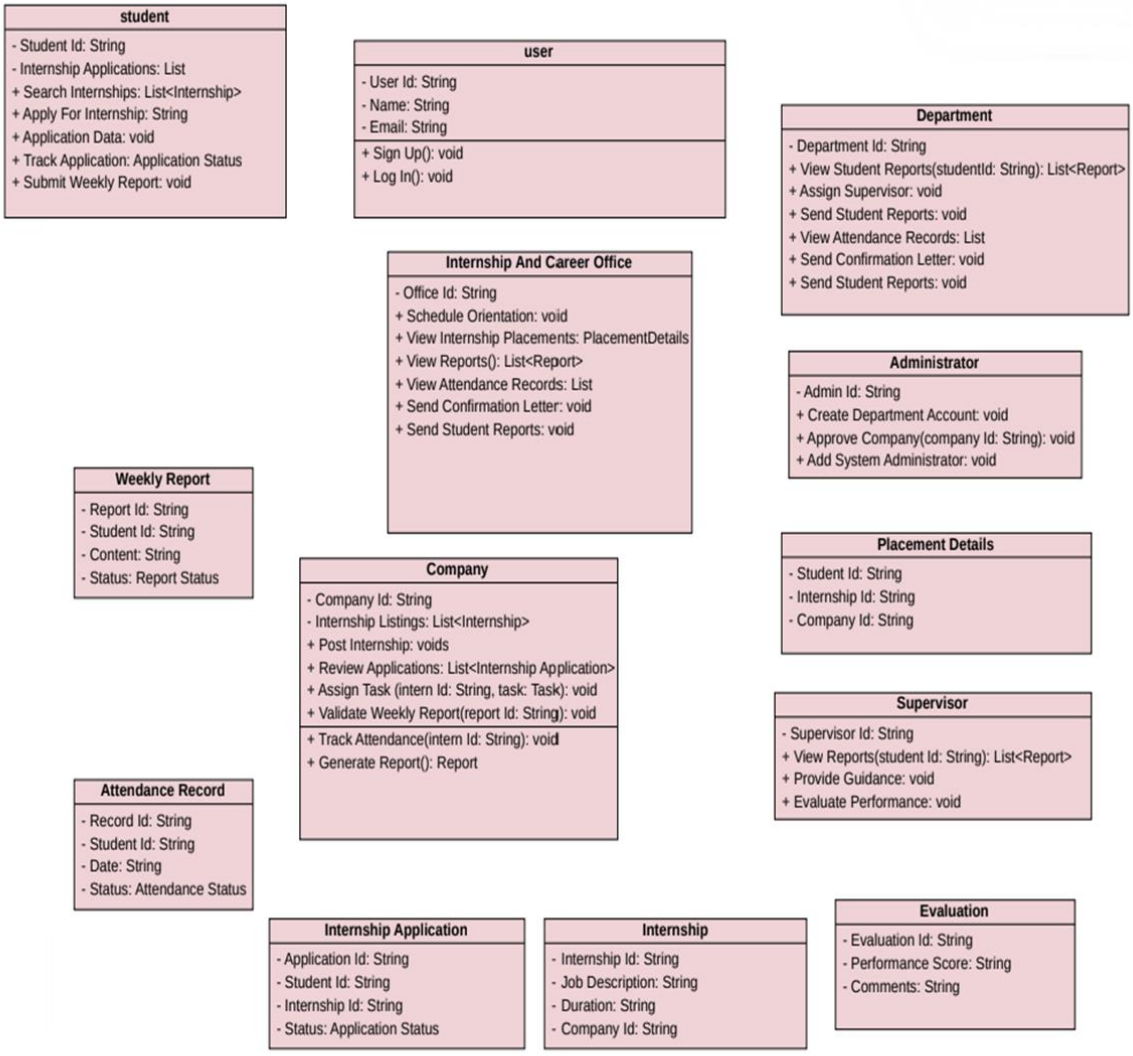
A new, integrated software architecture is needed to address these limitations and provide a more efficient, scalable, and user-friendly platform for managing internships. The proposed system will leverage modern web technologies and design patterns to ensure robust performance, data security, and seamless interaction between all stakeholders.


sign up
initial
sign up
registration
user retries
Enters
information
user information
validation
fail
validation failed
validation
successful
Account Created


Login
initial
user login
login
fill credentials
user Authentication
Invalid credentials
Login failed
retry
valid
credentials
Logged in


post
internship
initial
post
internship
Internship posting
form
fill
requirement
requirement
validation
invalid
requirement
validation failed
retry
validation
success
post published


validate bi-weekly report
initial
access submitted report
view report
validate report
report validation
invalid report
validation failed
retry
validation success
report sent


student
- Student Id: String
- Internship Applications: List
+ Search Internships: List<Internship>
+ Apply For Internship: String
+ Application Data: void
+ Track Application: Application Status
+ Submit Weekly Report: void
user
- User Id: String
- Name: String
- Email: String
+ Sign Up(): void
+ Log In(): void
Department
- Department Id: String
+ View Student Reports(studentId: String): List<Report>
+ Assign Supervisor: void
+ Send Student Reports: void
+ View Attendance Records: List
+ Send Confirmation Letter: void
+ Send Student Reports: void
Internship And Career Office
- Office Id: String
+ Schedule Orientation: void
+ View Internship Placements: PlacementDetails
+ View Reports(): List<Report>
+ View Attendance Records: List
+ Send Confirmation Letter: void
+ Send Student Reports: void
Administrator
- Admin Id: String
+ Create Department Account: void
+ Approve Company(company Id: String): void
+ Add System Administrator: void
Weekly Report
- Report Id: String
- Student Id: String
- Content: String
- Status: Report Status
Company
- Company Id: String
- Internship Listings: List<Internship>
+ Post Internship: voids
+ Review Applications: List<Internship Application>
+ Assign Task (intern Id: String, task: Task): void
+ Validate Weekly Report(report Id: String): void
+ Track Attendance(intern Id: String): void
+ Generate Report(): Report
Placement Details
- Student Id: String
- Internship Id: String
- Company Id: String
Supervisor
- Supervisor Id: String
+ View Reports(student Id: String): List<Report>
+ Provide Guidance: void
+ Evaluate Performance: void
Attendance Record
- Record Id: String
- Student Id: String
- Date: String
- Status: Attendance Status
Internship Application
- Application Id: String
- Student Id: String
- Internship Id: String
- Status: Application Status
Internship
- Internship Id: String
- Job Description: String
- Duration: String
- Company Id: String
Evaluation
- Evaluation Id: String
- Performance Score: String
- Comments: String