Software Requirement Specifications

Name	SRILEKHA V
Roll No	7376221CS317
Seat No	67
Project ID	4
Project Title	Student Ranking Dashboard

TECHNICAL COMPONENTS:

COMPONENT	TECH STACK
Frontend	• HTML • CSS • JAVASCRIPT
Backend	Django
Database	PostgreSQL

Implementation Timeline:

Phase	Description	Deadline	Status
Stage 1	Planning and Requirement gathering		Approved
Stage 2	Design and Prototyping		Not started
Stage 3	DB Designing		Not started
Stage 4	Backend Implementation		Not started
Stage 5	Testing & Implementation		Not started

1. INTRODUCTION:

1.1 Purpose

This document outlines the functionalities and specifications for a student ranking dashboard focused on daily challenges, tasks, and activities. This dashboard will serve as a platform to track student engagement and progress, fostering a healthy competitive environment.

1.2 Scope

This system caters to educational institutions where students participate in daily challenges, tasks, and activities. The dashboard will be accessible to both students and instructors.

2. OVERALL DESCRIPTION

This system has three primary user groups:
Students: Can view their ranking, progress, and details of challenges, tasks, and activities.
Faculty: Can monitor overall student participation, analyze trends, and potentially manage challenges.
Admin: The admin manages the student ranking system: creates user accounts, sets ranking rules, designs challenges, and monitors overall performance.

3. REQUIREMENTS

3.1 Functional Requirements

1. Admin

The admin user serves as the system administrator responsible for maintaining the dashboard andmanaging user roles. Their functionalities might include:

User Management:
Create, edit, and deactivate user accounts for instructors and students.
Assign roles (admin, instructor, student) to user accounts.
Challenge Management:
Create, edit, and delete challenges, tasks, and activities.
Set point values and deadlines for challenges.
System Configuration:
Define the ranking methodology (e.g., total points, fastest completion time).
Manage system settings and configurations.
Reporting (Optional):
Generate reports on overall student participation and performance.
Analyze trends in challenge completion and student engagement.

2. Faculty

This role focuses on monitoring student participation within a specific course or department. Instructors can:

Monitor Student Participation:
View overall participation statistics for their assigned students (e.g., completion rates, average points earned).
Track individual student progress for challenges, tasks, and activities.
Provide Feedback (Optional):
Leave comments or feedback for students on their completed activities (visible to students).
Challenge Management (Optional):
If authorized by the admin, create and manage challenges specifically for their assigned students.

3. Student

Students can access and utilize the core functionalities for tracking their progress:

Login: Authenticate using registered credentials (username/email and password).
Ranking View: See their current ranking based on the defined methodology.
Activity Feed: View details of current and upcoming challenges, tasks, and activities.

Progress Tracking: Track their progress for each completed activity, including points earned and completion time.
Historical Data (Optional): Access a record of past participation, showcasing previous challenges, tasks, activities, points earned, completion times, and rankings (weekly, monthly, etc.).

Additional Considerations:

The user interface should cater to all user groups, providing a clear distinction between functionalities based on their roles.
Security measures like role-based access control (RBAC) should be implemented to restrict access to sensitive information.
Data privacy regulations should be followed to ensure proper handling of student information.

3.2 Non-Functional Requirements

Usability:

The dashboard interface should be intuitive and user-friendly, promoting easy navigation and information access for students.
Data visualizations (rankings, progress charts) should be clear and easy to interpret.

Performance:

The system should efficiently handle concurrent student access and display information promptly.

Security:

User authentication should be implemented with secure protocols to prevent unauthorized access.
Student data (rankings, progress) should be protected.

Scalability:

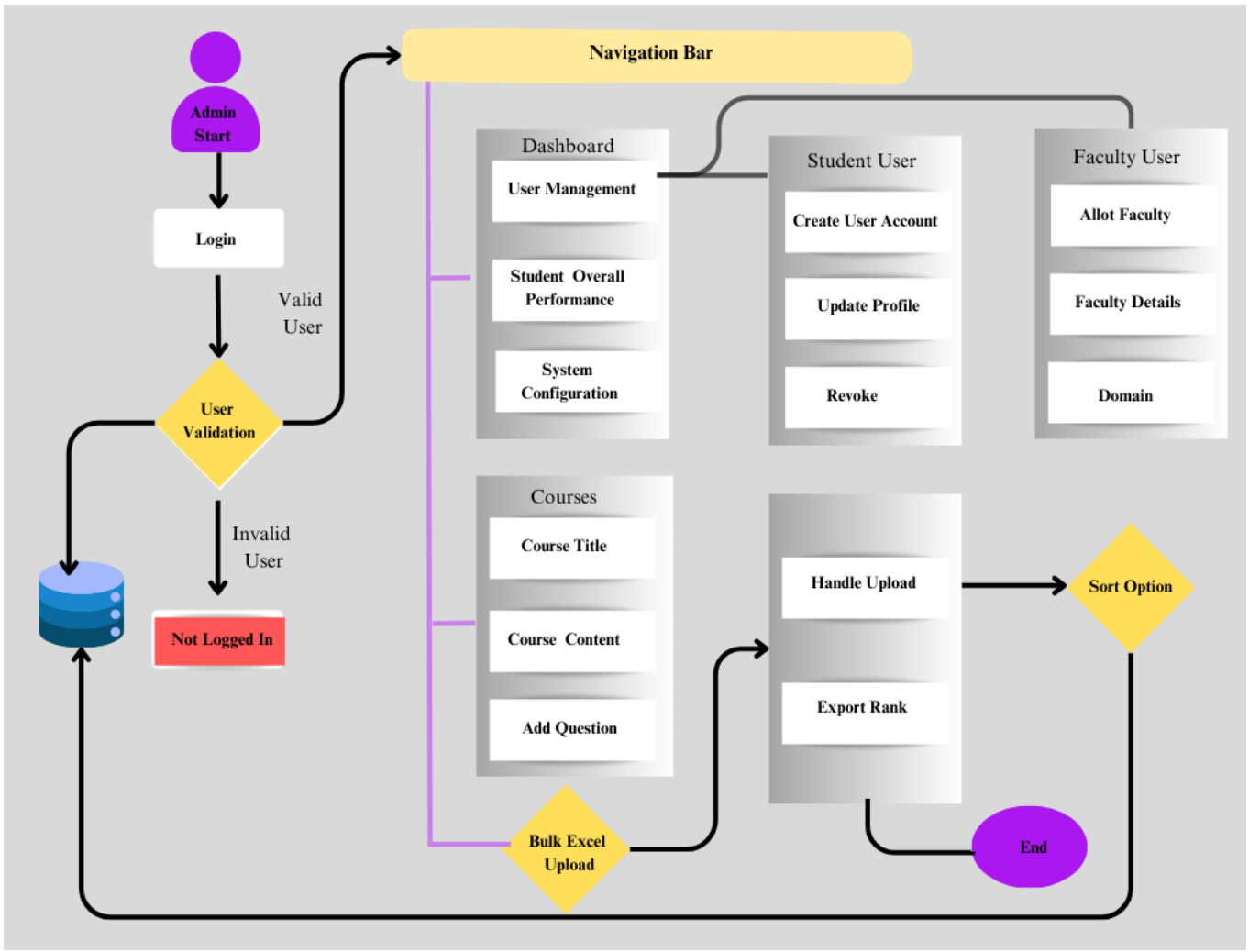
The system should be scalable to accommodate a growing number of students and activities.

Accessibility:

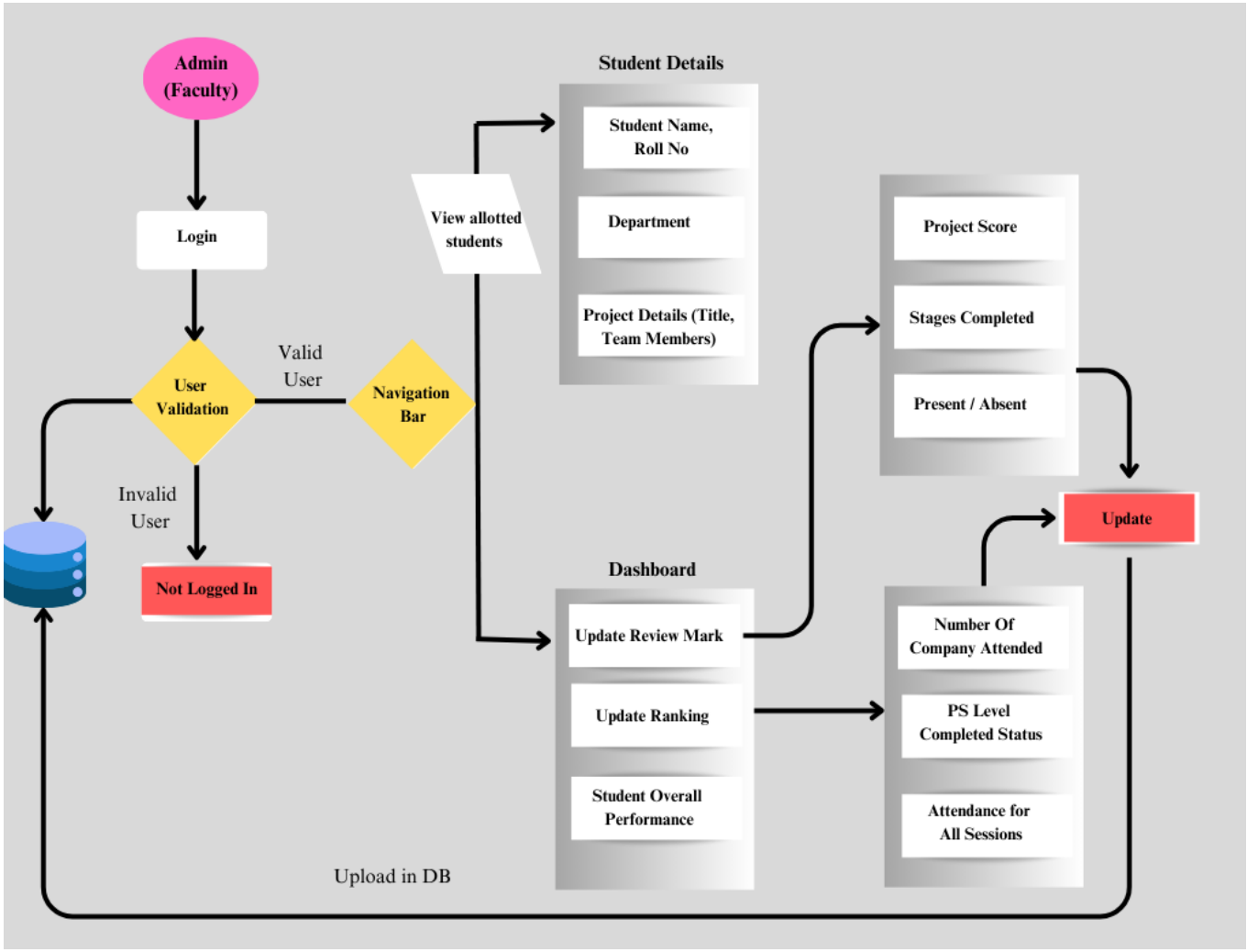
The dashboard should be accessible to students with disabilities by following accessibility best practices.

4. USER FLOW DIAGRAM

4.1 Admin View (Super Admin):



4.2 Faculty(Admin) Interface:



4.3 Student View: