BANNARI AMMAN INSTITUTE OF TECHNOLOGY
Autonomous Institution Affiliated to Anna University - Chennai, Accredited by NAAC with A+ Grade
Sathyamangalam - 638401 Erode District, Tamil Nadu, India

SOFTWARE REQUIREMENT SPECIFICATION FOR STUDENT ACTIVITY LOG

NAME	SRINIDHI P
ROLL NO	7376221CD147
DEPARTMENT	COMPUTER SCIENCE AND DESIGN
PROJECT ID	3
TABLE NO	61
PROBLEM STATEMENT	STUDENT ACTIVITY LOG

Implementation Timeline

Phase	Deadline	Status	Notes
Stage 1	24/07/2024	Under review	Planning and Requirement Gathering
Stage 2		Not started	Design and UI/UX Prototyping
Stage 3		Not started	Database Design and Implementation
Stage 4		Not started	Backend Development
Stage 5		Not started	Integration and Testing
Stage 6		Not started	Deployment

1. INTRODUCTION

1.1. Purpose

The purpose of this document is to present a detailed description of the Student Activity Log. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli.

1.2. Scope of the Project

The goal of the Student Activity Log project is to provide a thorough system for monitoring involvement in activities and attendance by students. The system will be used by students to sign in and out of different activities like Placements and Special labs, recording their participation information. Reviewing these logs, approving or rejecting check-ins and check-outs, and providing comments for any denials will be within the power of faculty members. Accurate tracking of student engagement will be made possible via the technology, enabling improved support and monitoring. The entire data collection will be managed by an administrative role that has authority to view, manage student activity logs.

2. SYSTEM OVERVIEW

2.1. Users

1.Students

Students will use the system to check in and check out of activities. This feature records the time and details of their participation, creating a digital log of their activity.

2.Faculty

Faculty members will have the ability to review student check-ins and check-outs. They can approve or reject these entries, and if an entry is rejected, faculty must provide a reason or remark. This allows for accountability and ensures that only legitimate activities are recorded.

3.Admin

Administrators have comprehensive access to all data within the system. They can view, manage student activity logs, facilitating oversight and strategic planning.

2.2. Features

1. Login

Students can login into the portal with their respective college mail id.

2. Student Check-In/Check-Out:

Simple and intuitive interface for students to log their participation in activities like Placement and Special Labs and time-stamped entries for accurate tracking of activity duration.

3. Faculty Review and Approval:

Faculty access to view student check-in/check-out logs and option to approve or reject entries with a mandatory remarks field for rejections.

4. Administrative Oversight:

Comprehensive dashboard for administrators to view all activity logs.

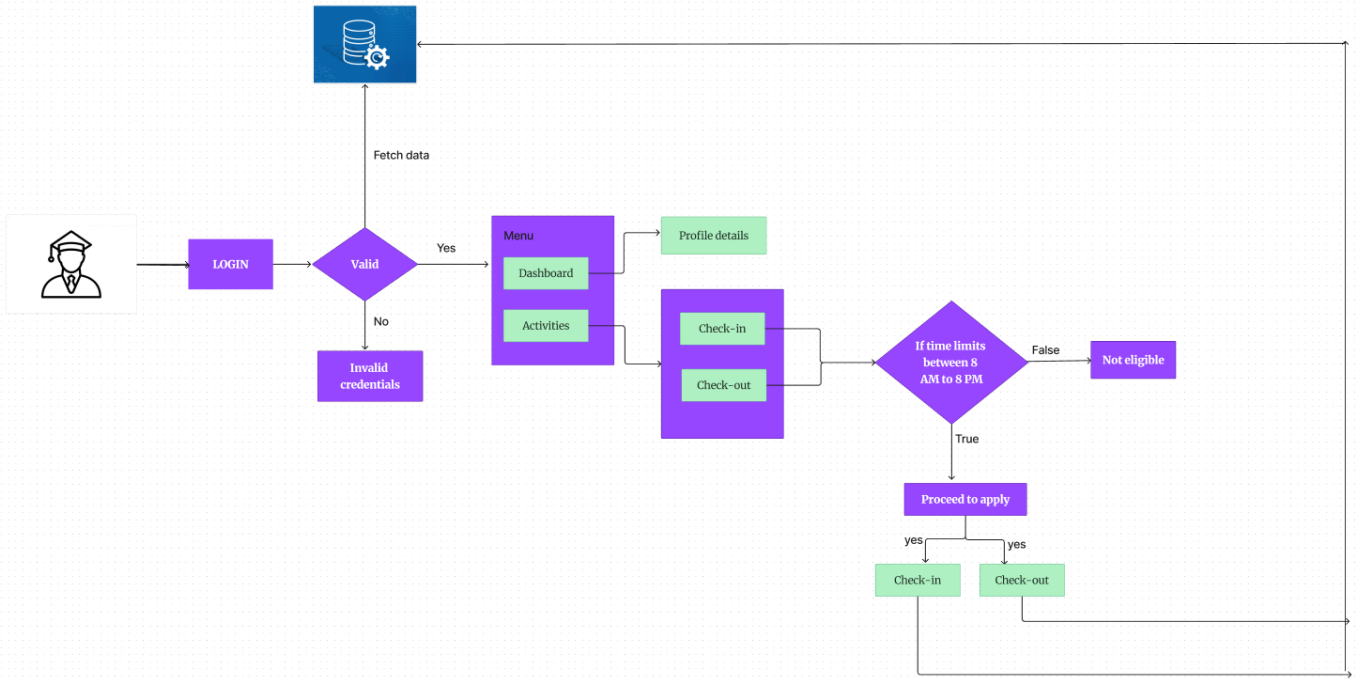
5. Data Management:

Secure storage of all logs with backup and recovery options and compliance with data privacy regulations to protect student information.

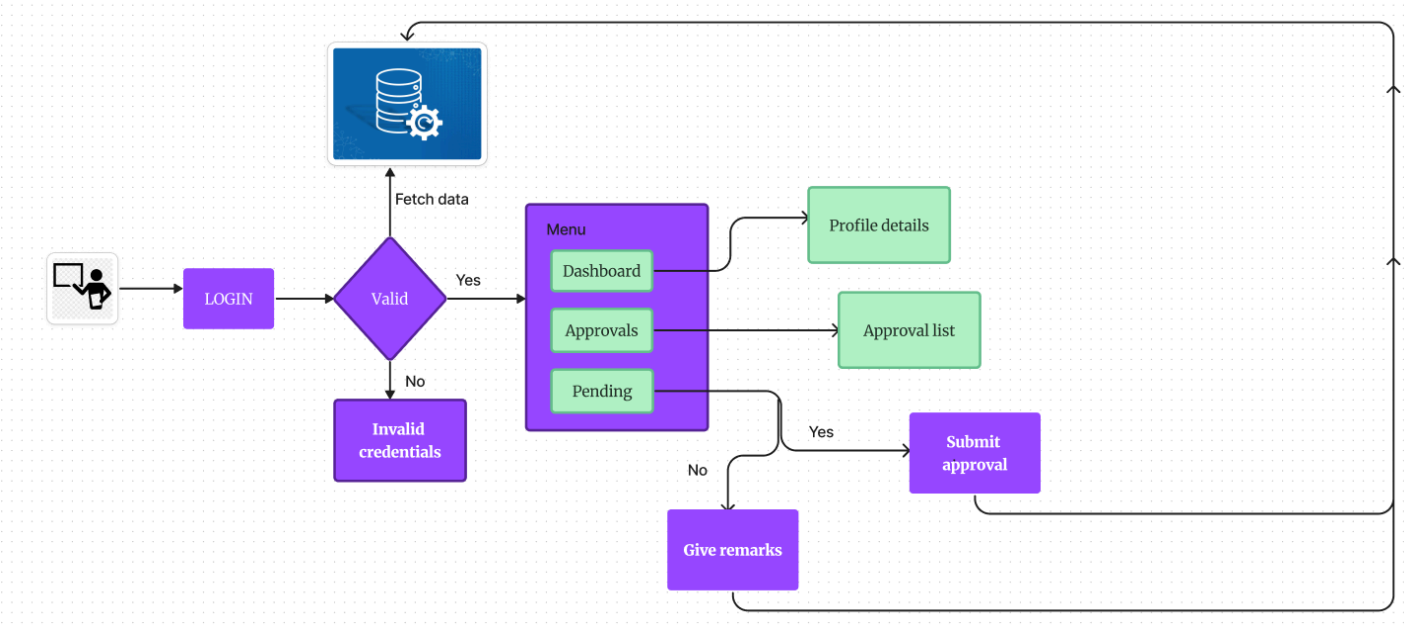
6. User-Friendly Interface:

Responsive design for accessibility on various devices like your desktops,phones and tablets.

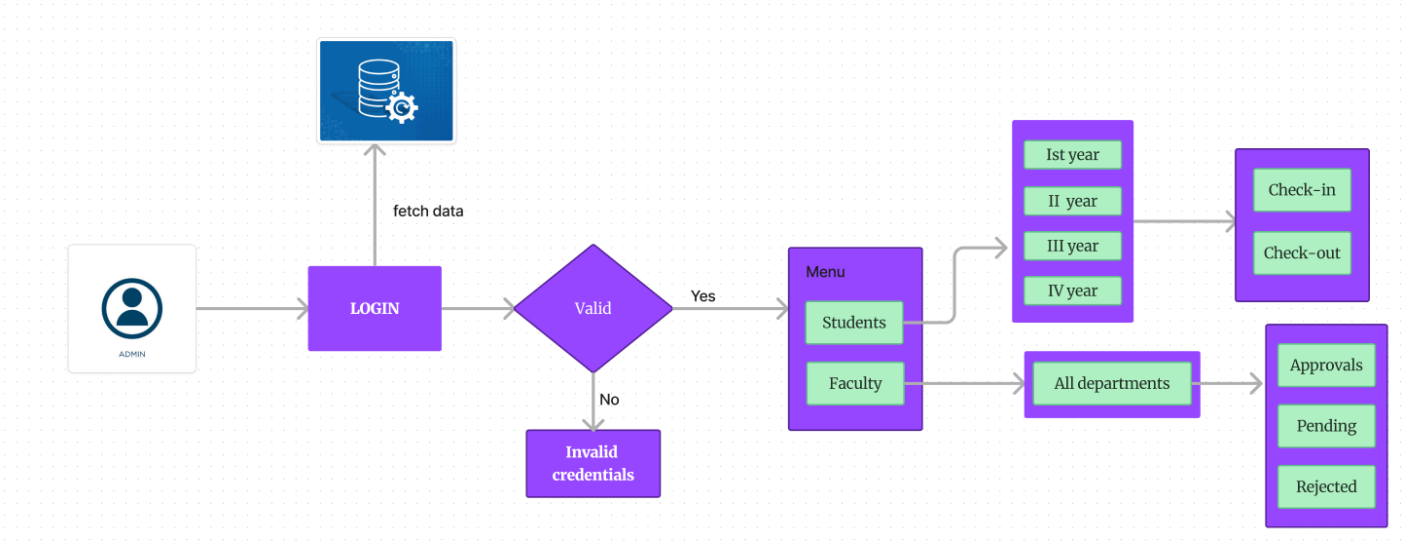
STUDENT INTERFACE:



FACULTY INTERFACE:



ADMIN INTERFACE:



FUNCTIONAL REQUIREMENTS

1.Student Interface:

It must be possible for students to safely check in using their own credentials.A single click must be sufficient for students to sign in and out of activities.Every check-in and check-out time and date must be recorded by the system.The ability for students to see their own activity records and approval statuses is required.

2. Faculty Interface:

Teachers need to be able to use their special login credentials to log in safely.Viewing pending student check-in and check-out entries is a must for faculty.Each entry must be able to be accepted or rejected by faculty.Faculty must be able to comment on items that are rejected and explain why.

3. Administrator Interface:

Using special login credentials, administrators must be able to log in safely.The logs of all student activity must be accessible to administrators.Activity logs must be able to be searched, filtered, and sorted by administrators.

4. User-Friendly Design and Data Management:

Desktop, tablet, and smartphone users must be able to access the system interface.Encryption of all check-in and check-out data is required by the system.

NON-FUNCTIONAL REQUIREMENTS

1. Performance:

At least 40 simultaneous logins must be supported by the system without causing any performance issues.The processes of checking in and out must be finished in a matter of seconds.

2.Scalability:

In order to handle growing user and activity numbers, the system needs to be expandable.

3.Usability:

User-friendly and intuitive system interface design is required. A consistent user experience must be offered by the system on all gadgets, including PCs, tablets, and smartphones.

4.Maintainability:

Easy maintenance and updates are made possible by a modular and well-documented system codebase.

5.Security:

Passwords for users need to be hashed and kept safely.Administrator access requires two-factor authentication, which the system must integrate.

STACK:

FRONT END	HTML, CSS and Javascript
BACK END	Linux with Apache Web Server and PHP with Laravel framework.
DATABASE	MYSQL
API	RESTful API