BANNARI AMMAN
INSTITUTE OF TECHNOLOGY
An Autonomous Institution, Affiliated to Anna University,
Approved by AICTE, Accredited by NAAC with 'A+' Grade
Stay Ahead

Student Name: SAMYUKTHA S

Seat No: 375

Project ID: 15

Project title: PRODUCT DEVELOPMENT

Technical Components

Front End	▪ React (JS Library for building user interfaces)
Back End	▪ Node.js with Express.js
Database	▪ MongoDB(NOSQL Database)
API	▪ OpenAPI

PROBLEM STATEMENT:

The task is to create a portal for product development that facilitates key stages like problem statement approval, BoM submission, reviews, student member assignment, uniqueness checking, documentation, and reward points. The portal should include forms, workflows, task tools, plagiarism checks, document management, and enable expert committee approval and weekly review updation to industry experts, admin, and faculty.

I. Introduction

1. PURPOSE:

The purpose of this document is to present a detailed description of the Product Development portal. It will explain the purpose and features of the system, the interfaces of the system, and system function, the constraints under which it is operated.

2. SCOPE OF PROJECT:

- This software system will serve as a portal for the Product Development, enabling faculty to propose their own problem statement or industry given problem statement and get approval from expert comitte. From an administrative perspective, this system will provide a comprehensive analytical dashboard for problem statement oversight.
- Expert comitte have the ability to approve or reject projects based on the uniqueness of the problem statement. Once a project is approved, students will receive a mail regarding the availability of problem statement. Intrested students will apply for problem statement and selected through interview process. Once the student are selected then the project work will be continued.

II. SYSTEM OVERVIEW:

1.Users:

i. Faculty:
They have the ability to propose problem statement for expert comitte approval, by uploading faculty details, problem statement title , descriptions, domain, category, incharge faculty and roles required. And they also monitor the review updates and schedule interview for selecting students for problem statement.

ii. Student:
Apply for problem statement and if selected through interview process then the project work will be started. And review updation is given to faculty.

iii. Expert committee:
Approve the problem statement proposed by the faculty based of uniqueness and if it is rejected then remarks will be sent to correct the mistake.

iv. Admin:
Have access to add or remove the problem statement and monitor the process of registered problem statement.

2. Features:

i. Login and registration:
Students, faculty, expert committee, industry expert and admin can login with their existing account

ii. Propose problem statement:
Faculty can apply for problem statement by giving revelant input like faculty details, domain, category, problem statement description, roles required and faculty in charge.

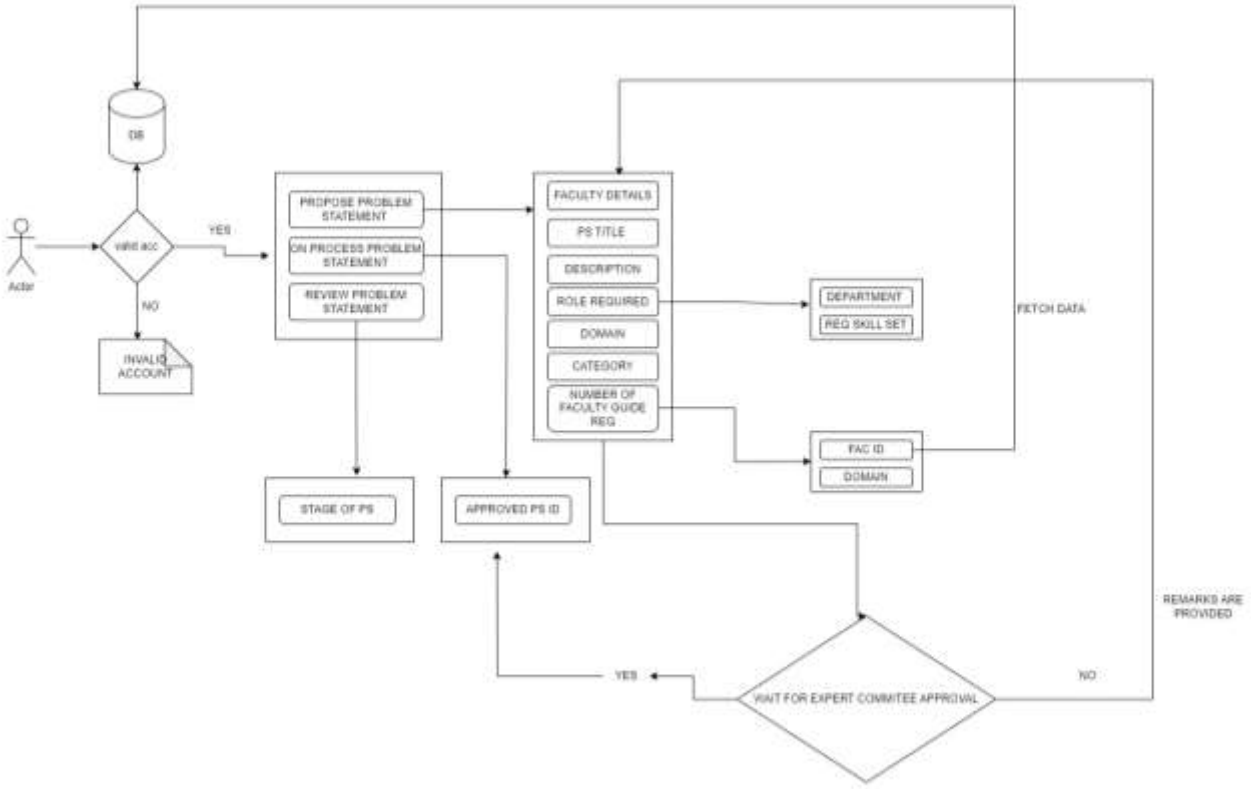
iii. Approval status:
If the proposed problem statement is unique then it will be approved by expert committee or remarks will be shared so that faculty can make changes in the problem statement.

iv. Student selection:
For the approved problem statement interested students will register and faculty will arrange for an interview and students will be selected through an interview process and skill set which they acquire.

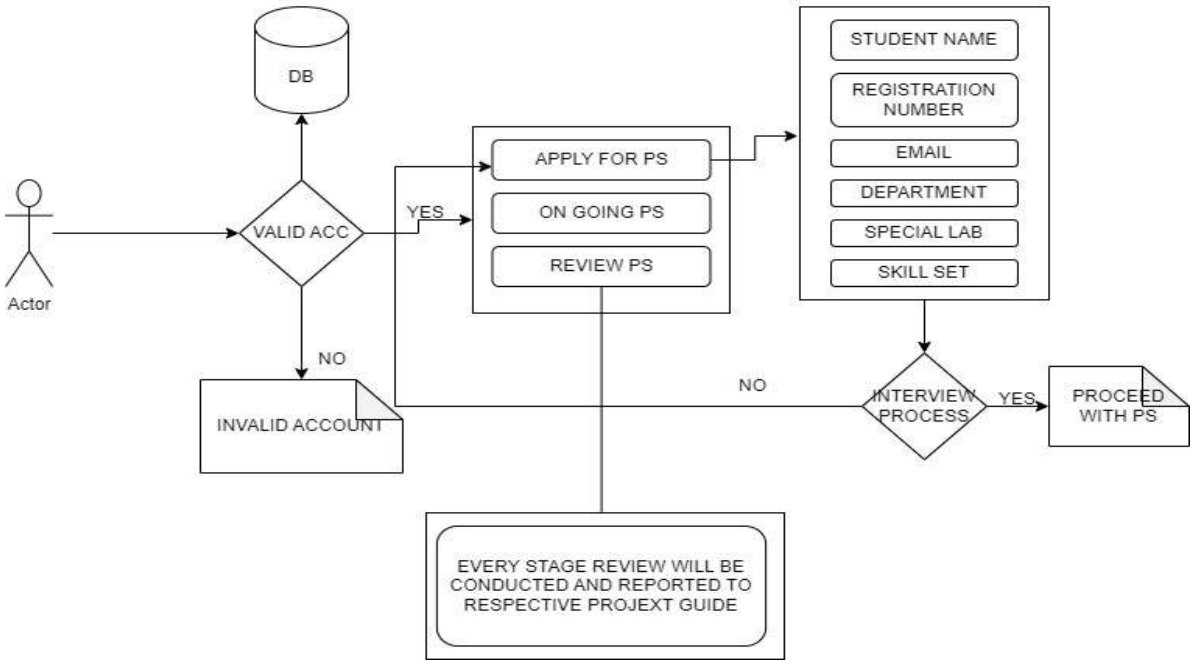
v. Review updation:
Once the project has been started there will be weekely review conducted and students will update it in website and the updates will be received to faculty,admin and industry expert committee.

vi. Admin access:
Admin can view all registed problem statement, on going satus of problem statement and review updations.

FACULTY INTERFACE



STUDENT INTERFACE



3. **Functional Requirements:**
    - **User Management:**
        - User can register and login
        - Admins have access control with an analytical dashboard and dedicated features .
    - **Application for Problem statement:**
        - Faculty can propose problem statement with appropriate details
        - Application contains:
            - Faculty details
            - Domain
            - Category
            - Title
            - Description
            - Faculty in-charge
            - Roles required
    - **Application Status:**
        - Faculty can view the current status of their problem statement
        - If the application is rejected then the remarks is shown
        - Faculty can also able to view the logs of their applications
    - **Appointment Scheduling (After Approval):**
        - Students can register for approved problem statement .
        - Faculty will allot interview schedule for registered students and will be selected according to their skill set
    - **Admin Dashboard:**
        - Admins can view a list of all approved problem statement.
        - Applications can be filtered by domain (software, hardware).
        - Admins can view the details of each problem statement.
    - **Expert committee Dashboard:**
        - Experts can approve or reject applications with suitable remarks.

4. **Non-Functional Requirements:**
    - **Performance:**
      To guarantee effective usability, the system must react to user input in less than two seconds and be able to support at least 100 concurrent users without experiencing appreciable performance deterioration.
    - **Security:**
      Secure authentication methods should be used to limit access to sensitive functionalities to authorized admin users, and user data must be encrypted both during transmission and storage.
    - **Usability:**
      In order to assist users in the event of input errors or system failures, the user interface should be simple to use and intuitive. Clear and concise error messages should also be provided.
    - **Reliability:**
      In order to prevent data loss in the event of system failures or crashes, the system should have a backup and recovery mechanism in place and be available around-the-clock with little downtime.

- **Scalability:**
  The system need to be built to handle growing user counts and data volumes over time, and it needs to be expandable to handle new features and functionalities as needed.

Backend:

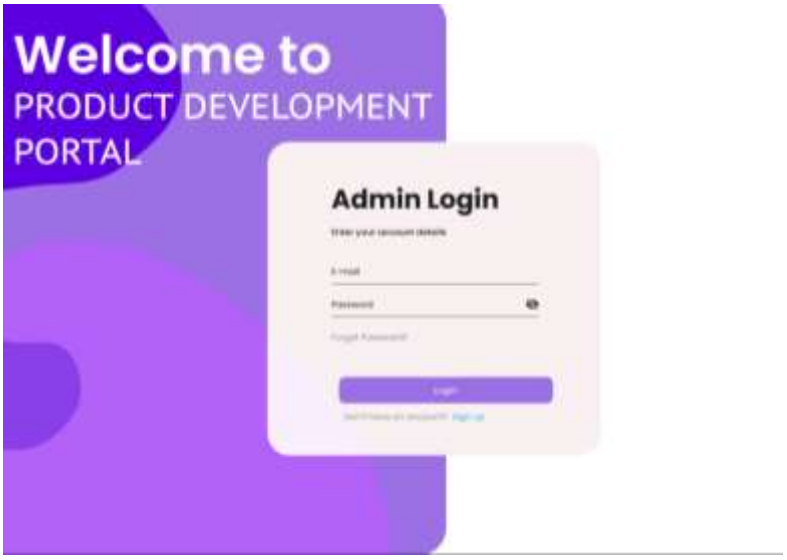
1. **Faculty entity:**

Name	String
Domain	String
Category	String
Title	String
Faculty in-charge	Int
Number of students required	Int
Roles required	String

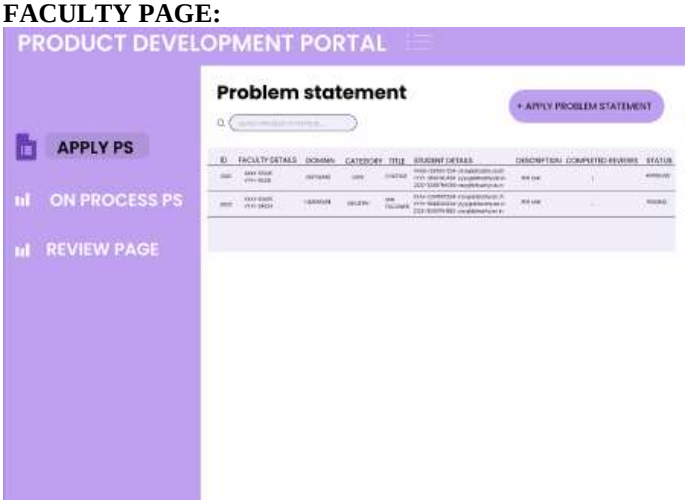
Prototype of the Project:

1. **ROLE SELECTION:**

2. **LOGIN FORM:**

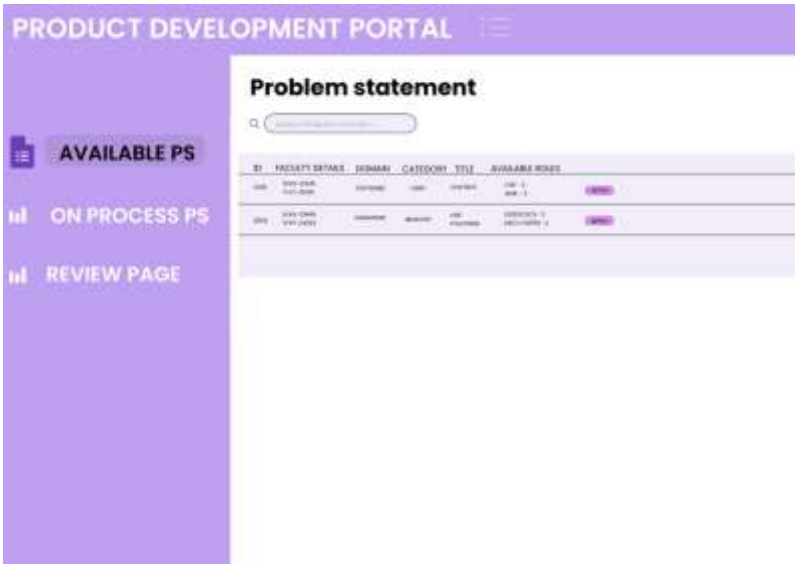


3. **FACULTY PAGE:**

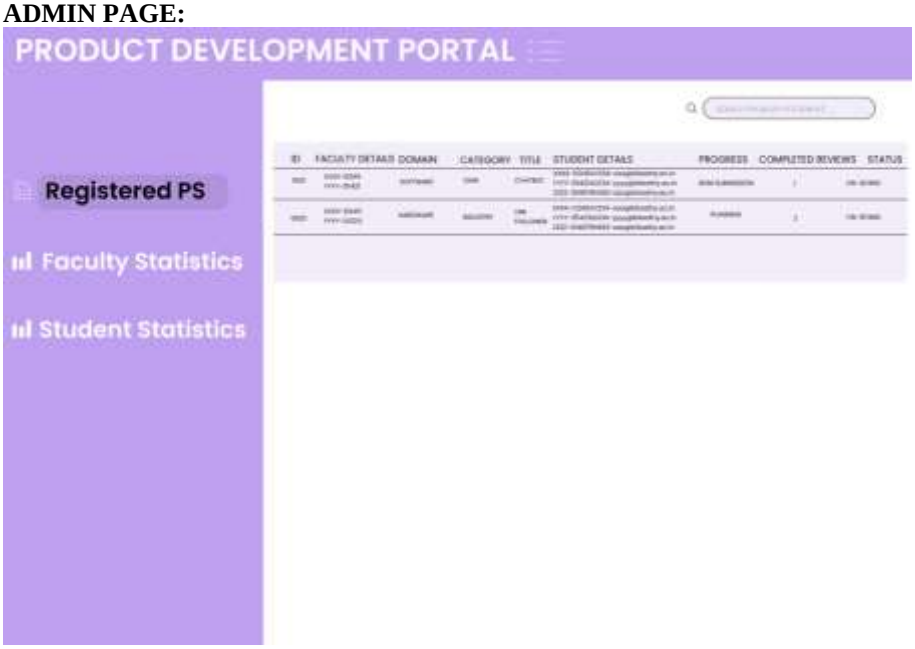


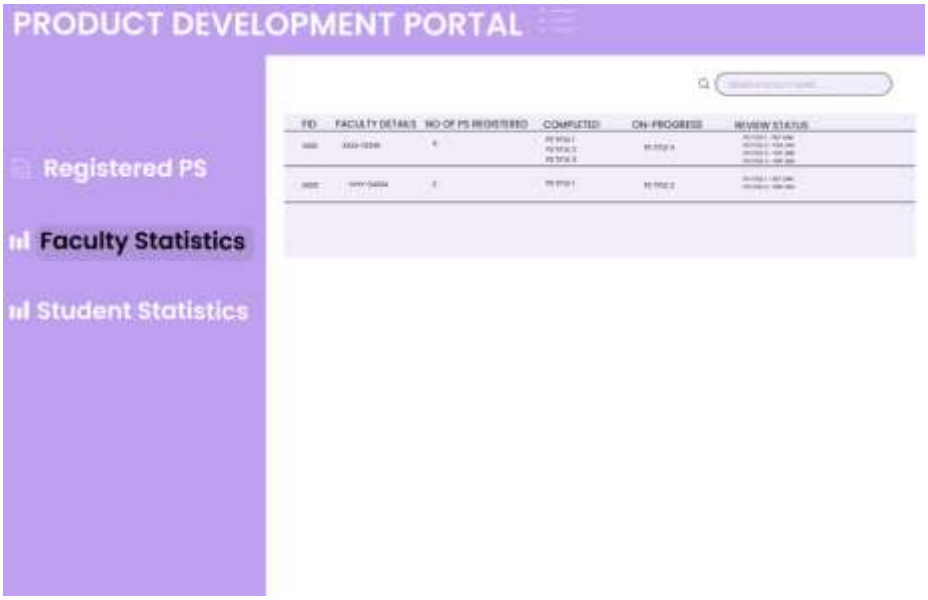
4. **PROBLEM STATEMENT APPLYING PAGE:**

5. **STUDENT PAGE:**



6. ADMIN PAGE:


PRODUCT DEVELOPMENT PORTAL
Registered PS
Faculty Statistics
Student Statistics


PRODUCT DEVELOPMENT PORTAL
Registered PS
Faculty Statistics
Student Statistics