**CampusSink** - Project Charter Document.

# Purpose:

The objective of this project is to create a centralized web-based system that displays all the student assignments, seminars, guest lectures, and project reviews organized by branch and year. Teachers will have full control over adding, editing, and managing assignments and deadlines. Students will only be able to view tasks, seminars, and projects relevant to their year and branch.

# Objectives:

* **Role-based Access**: Provide different levels of access for teachers and students.
  + **Teachers**: Can view and manage all assignments, seminars, and projects. Can edit deadlines and track submissions
  + **Students**: Can view assignments, seminars, and projects for their branch and year.
* **Assignment Management**: Teachers can upload, edit, and delete assignments, project reviews, and seminars.
* **Deadlines**: Display upcoming deadlines for assignments, seminars, and projects in a user- friendly format and the user will be notified of the deadlines.
* **Categorization**: Organize the assignments based on the student's branch and academic year.

**Pestle Analysis**:

|  |  |
| --- | --- |
| **Factor** | **Impact on the Exam Evaluation System** |
| Political | No impact |

|  |  |
| --- | --- |
| Economic | The cost of developing, maintaining, and hosting the portal can impact project sustainability. |
| Social | Some teachers and students may resist the move to digital platforms, particularly in more traditional or under-resourced institutions.  Training and support will be essential.. |
| Technological | Using technologies like React.js, Node.js, PostgreSQL, and cloud hosting (AWS) will allow for scalability and flexibility.  With sensitive data involved, strong encryption and secure access control measures are necessary to protect against data breaches and cyberattacks. |
| Environmental | A digital platform for assignments will significantly reduce the use of paper, which contributes positively to the environment by minimizing the carbon footprint associated with traditional education systems. |
| Legal | Teachers may want to protect their content (e.g., assignments, materials) from being reused or plagiarized. Clear policies on intellectual property will be necessary. |

# Opportunity based / Demand based:

The **portal** is **opportunity-based**, utilizing advancements in technology to streamline assignment management, improve accessibility, and replace inefficient paper-based systems, offering modern solutions to educational institutions.

# Business / High-Level Features:

Teacher Features:

* Add, edit, and delete assignments, seminars, and project reviews.
* View all assignments for different branches and years.
* Set and manage deadlines.
* Monitor assignment submission status.

Student Features:

* View only their branch and year's assignments, seminars, and projects.
* See upcoming deadlines and submission status.

# Technical Requirements:

Frontend:

* **React.js** for building the web interface.
* **CSS** or **Bootstrap** for a responsive design.

Backend:

* Node.js with **Express.js** for server-side logic and API handling.

Database:

* **PostgreSQL** for structured data storage to manage user accounts, assignments, and deadlines.

Hosting & Deployment:

* **AWS Cloud** or **Vercel** for hosting the web application and backend server.

# Security Considerations:

* **OAuth** for user authentication and ensuring secure login.
* **End-to-end encryption** to protect sensitive student and teacher data, particularly around assignment submissions and deadlines.

**Risk Analysis:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk Category** | **Description** | **Impact** | **Mitigation Strategy** |
| **1. Technical Risk** | The scope of the project may change due to changing requirements or technical  challenges. | High | Clearly define and freeze requirements in the planning phase. |
| **2. Cost Risk** | The project may face unexpected cost overruns due to unforeseen design, development, or operational  needs. | High | - Establish detailed cost estimates with contingency buffers. |
| **3. Schedule Risk** | There may be delays in activities such as regulatory approval, detailed design, and integration of features, which could push back  project timelines. | Medium | * Use a detailed project plan with adequate buffer time for each task. * Regular progress tracking and early identification of potential bottlenecks. |
| **4. Budget Risk** | * The project budget may be underestimated or insufficient, leading to financial strain. * Scope creep could add   unanticipated expenses. | High | * Establish a realistic budget with allowances for scope changes. * Monitor scope closely to prevent scope creep and use change control processes to manage budgetary   impact. |
| **5. Business Risk** | Delays in management approvals or inputs from key stakeholders (teachers, institutions) may result in the loss of market timing, affecting project  profitability. | Medium | * Ensure timely communication with stakeholders. * Set firm deadlines for decision- making, approvals, and input from business entities. |
| **6. Infrastructure**  **Risk** | - Lack of adequate  infrastructure, such as cloud | Medium | - Plan infrastructure needs early and  work closely with suppliers (cloud |

|  |  |  |  |
| --- | --- | --- | --- |
|  | services or hosting platforms, could delay project activities such as  system setup or operations. |  | providers).  - Ensure proper allocation of resources for setup and operation  phases. |
| **7. Operational Risk** | Improper execution of deployment, or insufficient resources during critical stages such as testing and launch, could result in system failures or poor  performance. | High | * Implement clear operational guidelines and assign adequate resources for critical phases. * Conduct thorough testing before launch and assign a dedicated support team. |
| **8. Resource Risk** | - Availability of key personnel, such as developers or testers, may become a challenge,  resulting in delays. | Medium | - Secure necessary resources early and maintain backup plans in case of staffing shortages. |
| **9. Supplier Risk** | Third-party suppliers, such as cloud service providers or database service providers, may fail to deliver expected  results or meet deadlines. | Medium | Create clear contracts with performance metrics and penalties for delays or failure to deliver. |

**Stakeholders**:

|  |  |
| --- | --- |
| End User | Students, Teachers (Administrators of  assignments). |
| Sponsors | Educational institutions, academic boards |
| Suppliers | Cloud service providers (AWS, Azure),  database services |
| Frontend developer | Dhiraj Rathod |
| Testers | Shravani Sakore |

|  |  |
| --- | --- |
| Design | Diksha Tiwari |
| Backend developer | Shivshankar Ghyar |
| Consultant | Jayesh Raut |
| Project Lead | Dr.Anuradha Pawar |

**Project ROI (Return on Investment)**:

## Revenue Models:

* + Subscription-based access for institutions to manage and monitor assignments.
  + Pay-per-student or pay-per-year model for accessing assignment management.

## ROI Timeline:

* + Break-even expected within 1-2 years after launching, with adoption in multiple educational institutions.

# Resources & Technologies Needed:

Human Resources:

* **Frontend Developers**: 2 (React.js,CSS).
* **Backend Developers**: 2 (Node.js, Express.js, PostgreSQL).
* **UI/UX Designers**: To design user interfaces that are easy for students and teachers to navigate.
* **Security Experts**: To ensure compliance with data privacy laws.

Technologies & Tools:

* **Cloud Computing**: AWS, Azure for hosting.
* **Database**: PostgreSQL for storing structured data like assignments and user profiles

# Timeline & Milestones:

|  |  |  |
| --- | --- | --- |
|  | **Tasks** | **Timeline** |
| **Phase 1: Planning** | Market research, requirements gathering | 1 week |
| **Phase 2: Design** | UI/UX design, database schema, architecture | 2 week |
| **Phase 3: Development** | Frontend & backend coding, API integration | 3 weeks |
| **Phase 4: Testing** | Testing UI, security audits, bug fixing | 1 week |
| **Phase 5: Deployment** | Deploy to AWS/Vercel, final roll-out | 1 week |

**RACI CHART**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Task/Deliverable** | **Shravani,Shivs**  **hankar** | **Dhiraj** | **Diksha** | **Dr.Anuradha**  **Pawar** | **Jayesh Rout** |
| Coding the Website | R | A | C | I | I |
| Design Implementation | C | A | R | I | I |
| Database & API Setup | R | A | I | I | C |
| Testing & Optimization | R | C | I | I | I |

* **R (Responsible)**: Person who performs the work.
* **A (Accountable)**: Person accountable for task completion.
* **C (Consulted)**: Person whose input is needed before proceeding.
* **I (Informed)**: A person who must be kept updated.