**Project Catalyst - Software Requirements Specification (SRS)**

**Table of Contents**

1. Introduction

1.1 Purpose

1.2 Scope

1.3 Definitions, Acronyms, and Abbreviations

1. System Overview

2.1 System Architecture

2.2 Users

1. Functional Requirements

3.1 User Registration and Authentication

3.2 Student Features

3.2.1 View Available Projects

3.2.2 Propose a Project

3.2.3 Check Proposal Status

3.3 Faculty Member Features

3.3.1 Add Project Ideas

3.3.2 Approve/Reject Student Proposals

3.3.3 Mark Projects as Unavailable

1. Non-Functional Requirements

4.1 Performance

4.2 Security

4.3 Usability

1. Constraints
2. Project Timeline
3. Conclusion

**1. Introduction**

**1.1 Purpose**

The "Project Catalyst" is a web-based platform designed to facilitate the sharing, approval, and management of project ideas for students and faculty members within an educational institution.

**1.2 Scope**

1.2.1 In-Scope

* User registration and authentication.
* Students can view available project ideas.
* Students can propose project ideas.
* Students can check the status of their proposed projects.
* Faculty members can add, edit, or delete project ideas.
* Faculty members can approve or reject student project proposals.
* Faculty members can mark projects as unavailable.

1.2.2 Out-of-Scope

* Advanced project collaboration features (e.g., team formation).
* Financial transactions.
* Integration with external systems.

**1.3 Definitions, Acronyms, and Abbreviations**

* SRS: Software Requirements Specification
* HTML: HyperText Markup Language
* CSS: Cascading Style Sheets
* PHP: Hypertext Preprocessor
* Bootstrap: A front-end framework for web development

**2. System Overview**

**2.1 System Architecture**

The system will consist of a front-end developed using HTML, CSS, and Bootstrap. The back-end will use PHP for server-side processing and interact with a MySQL database for data storage.

**2.2 Users**

* **Students:** Users who can browse, propose, and track project ideas.
* **Faculty Members:** Users with additional privileges to manage project ideas and approve student proposals.

**3. Functional Requirements**

**3.1 User Registration and Authentication**

* Users must register with their name, email, and password.
* Passwords should be securely hashed and stored.
* Registered users must log in to access the system.

**3.2 Student Features**

3.2.1 View Available Projects

* Students can view a list of available project ideas.
* Projects should display the title, description, faculty member, a request button and status (available or unavailable).

3.2.2 Propose a Project

* Students can select an available project and propose it.
* The request will be sent to corresponding faculty member for their approval.

3.2.3 Check Proposal Status

* Students can check the status of their project proposals (approved, rejected, or pending).
* Feedback or comments from faculty members should be visible for rejected proposals.

**3.3 Faculty Member Features**

3.3.1 Add Project Ideas

* Faculty members can add new project ideas, including a title, description, and status (available).

3.3.2 Approve/Reject Student Proposals

* Faculty members can review and approve or reject student project proposals.
* They can provide feedback or comments when rejecting proposals.

3.3.3 Mark Projects as Unavailable

* Faculty members can mark approved projects as unavailable.
* Unavailable projects should no longer be visible to students.

**4. Non-Functional Requirements**

**4.1 Performance**

* The system should handle a large number of concurrent users without significant performance degradation.
* Response times should be within acceptable limits.

**4.2 Security**

* User data, especially passwords, should be stored securely and transmitted over HTTPS.
* Authentication and authorization mechanisms should be robust to prevent unauthorized access.

**4.3 Usability**

* The user interface should be intuitive and responsive, compatible with different devices and browsers.

**5. Constraints**

* The project must adhere to all relevant data protection and privacy regulations.
* Development should use HTML, CSS, PHP, Bootstrap, and MySQL as specified.

**6. Project Timeline**

30th September 2023

**7. Conclusion**

This Software Requirements Specification outlines the essential features and functionality of the "Project Catalyst" project. It will serve as a foundation for the development team to design and implement the system according to the specified requirements.

Top of Form