

Student Name: DEVASRI S

Roll number: 7376221CS132

Seat No: 513

Project ID: 16

Project Title: Internship course exemption

Stack:

Components	Tech stack
Front End	HTML, CSS, JavaScript
Back End	Django
Database	MySQL
API	openAPI, SOAP API

Introduction

1. Problem statement:

Develop a structured process to exempt students with significant prior work experience from the mandatory 90-day internship course. This will prevent redundancy, optimize resource allocation, and enhance student engagement by acknowledging their professional expertise and enabling them to focus on more advanced, career-relevant learning opportunities.

2. Purpose:

The purpose of this initiative is to establish a formalized process for granting exemptions from the 90-day internship course to students with substantial prior work experience. This process aims to prevent redundant learning, improve resource allocation, and boost student motivation by recognizing their professional background. By doing so, the initiative ensures that all students engage in educational experiences that are most relevant and beneficial to their career development

3. Scope:

The scope of this initiative includes the development and implementation of a structured exemption process for the 90-day internship course. It involves setting clear criteria for significant prior work experience, creating an application procedure, and forming an evaluation panel consisting of faculty and industry professionals. Additionally, the scope encompasses providing alternative assignments or advanced projects for exempted students. This ensures that the process is fair, transparent, and enhances the educational experience for all students, particularly those with substantial industry experience.

System Overview:

1. Criteria Development:

- Define clear, measurable criteria for what constitutes significant prior work experience. Criteria will include duration of employment, relevance of job role, and nature of tasks performed.

2. Application Submission:

- Create an online portal where students can submit their exemption applications. The portal will allow for the upload of supporting documents such as resumes, job descriptions, letters of recommendation, and performance reviews.

3. Evaluation Panel:

- Establish a panel composed of faculty members and industry professionals to review and assess the applications. The panel will evaluate the submitted documents against the established criteria.

4. Decision and Notification:

- Implement a system for the panel to record their decisions. Successful applicants will receive notifications of their exemption status along with details on alternative assignments or projects.

5. Alternative Assignments:

- Develop a set of advanced assignments or projects tailored to the career goals of exempted students. These assignments will ensure continued professional growth and learning.

6. Tracking and Reporting:

- Incorporate a tracking system to monitor the progress and outcomes of exempted students. This will include feedback mechanisms to assess the effectiveness of the exemption process and alternative assignments.

2.1 Features in Dashboard:

• User Authentication:

- **Login/Logout:** Secure login and logout functionality for students, faculty, and administrators.
- **Registration:** New user registration for students and faculty.
- **Password Management:** Password reset and recovery options.

• Application Submission:

- **Form Submission:** Students can fill out and submit internship exemption application forms.
- **Document Upload:** Ability to upload necessary supporting documents.
- **Save Draft:** Save applications as drafts before final submission.

• Application Management:

- **View Submissions:** Students can view the status of their submitted applications.
- **Edit Applications:** Option to edit applications that are in draft or pending status.

- **Track Status:** Real-time updates on application status (pending, approved, rejected).
- **Review and Approval:**
 - **Review Interface:** Faculty and administrators have a dedicated interface to review applications.
 - **Feedback and Comments:** Provide feedback and comments on applications.
 - **Approval/Rejection:** Approve or reject applications with reasons and feedback.
- **Notification System:**
 - **Email Notifications:** Automatic email notifications to students on application status changes.
 - **In-Dashboard Alerts:** Real-time notifications within the dashboard for updates and changes.
- **Alternative Assignment Management:**
 - **Assign Projects:** Assign alternative projects or assignments to students who are exempted.
 - **Track Progress:** Monitor the progress of assigned projects.
 - **Submit Reports:** Students can submit progress reports and final submissions for alternative assignments.
- **Admin Dashboard:**
 - **Overview Statistics:** Summary statistics of total applications, approvals, rejections, and pending submissions.
 - **User Management:** Manage student and faculty accounts, roles, and permissions.
 - **Report Generation:** Generate detailed reports on application statuses, user activity, and system usage.
- **Data Management:**
 - **Database Management:** Efficient storage and retrieval of application data.
 - **Backup and Recovery:** Regular backups and recovery options for data integrity and security.
 - **Search and Filter:** Advanced search and filtering options for applications and user data.
- **API Integration:**
 - **OpenAPI Documentation:** Detailed API documentation for developers.
 - **External Integration:** Integration with external systems using SOAP API and other web services.
- **User Interface:**
 - **Responsive Design:** Ensures compatibility with various devices (desktops, tablets, mobile phones).

- **Intuitive Navigation:** Easy-to-use navigation for accessing different features of the dashboard.
- **Customization:** Personalize dashboard settings and preferences.

Functional Requirements

User Authentication

- **Login/Logout:**
 - Users must be able to securely log in and log out of the system.
 - The system should validate user credentials against stored data.
- **Registration:**
 - Students and faculty should be able to register for an account by providing necessary information.
 - The system should send a confirmation email to the user to complete the registration process.
- **Password Management:**
 - Users must be able to reset their passwords if forgotten.
 - The system should send a password reset link to the registered email address.

Application Submission

- **Form Submission:**
 - Students must be able to fill out and submit internship exemption application forms.
 - The form should include fields for personal information, internship details, and reasons for exemption.
- **Document Upload:**
 - Students must be able to upload supporting documents such as internship offer letters, previous work experience, etc.
- **Save Draft:**
 - Students should be able to save their applications as drafts and complete them later before final submission.

Application Management

- **View Submissions:**
 - Students must be able to view the status of their submitted applications (e.g., pending, approved, rejected).
- **Edit Applications:**
 - Students should be able to edit applications that are in draft or pending status.
- **Track Status:**
 - Students must receive real-time updates on the status of their applications.

Review and Approval

- **Review Interface:**
 - Faculty and administrators must have a dedicated interface to review submitted applications.
- **Feedback and Comments:**
 - Faculty should be able to provide feedback and comments on applications.
- **Approval/Rejection:**

- Faculty must be able to approve or reject applications, with the option to provide reasons and feedback.

Notification System

- **Email Notifications:**
 - The system should send automatic email notifications to students when the status of their application changes.
- **In-Dashboard Alerts:**
 - Students, faculty, and administrators should receive real-time notifications within the dashboard for updates and changes.

Alternative Assignment Management

- **Assign Projects:**
 - Faculty should be able to assign alternative projects or assignments to students who receive exemptions.
- **Track Progress:**
 - Faculty and students must be able to monitor the progress of assigned projects.
- **Submit Reports:**
 - Students should be able to submit progress reports and final submissions for their alternative assignments through the dashboard.

Admin Dashboard

- **Overview Statistics:**
 - Administrators should have access to summary statistics of total applications, approvals, rejections, and pending submissions.
- **User Management:**
 - Administrators must be able to manage student and faculty accounts, including roles and permissions.
- **Report Generation:**
 - The system should generate detailed reports on application statuses, user activity, and system usage.

Data Management

- **Database Management:**
 - The system must efficiently store and retrieve application data.
- **Backup and Recovery:**
 - Regular backups must be performed, and recovery options should be available for data integrity and security.
- **Search and Filter:**
 - Users should be able to perform advanced search and filtering on applications and user data.

API Integration

- **OpenAPI Documentation:**
 - The system should provide detailed API documentation for developers.
- **External Integration:**

- The system must integrate with external systems using SOAP API and other web services, if necessary.

User Interface

- **Responsive Design:**
 - The dashboard must be compatible with various devices, including desktops, tablets, and mobile phones.
- **Intuitive Navigation:**
 - The user interface should be easy to navigate, with clear access to different features of the dashboard.
- **Customization:**
 - Users should be able to personalize dashboard settings and preferences

Non-Functional Requirements

Performance

- **Response Time:**
 - The system should have a response time of less than 2 seconds for any user request.
- **Scalability:**
 - The system must be able to handle an increasing number of users and applications without performance degradation.
- **Throughput:**
 - The system should support at least 1000 concurrent users with consistent performance.

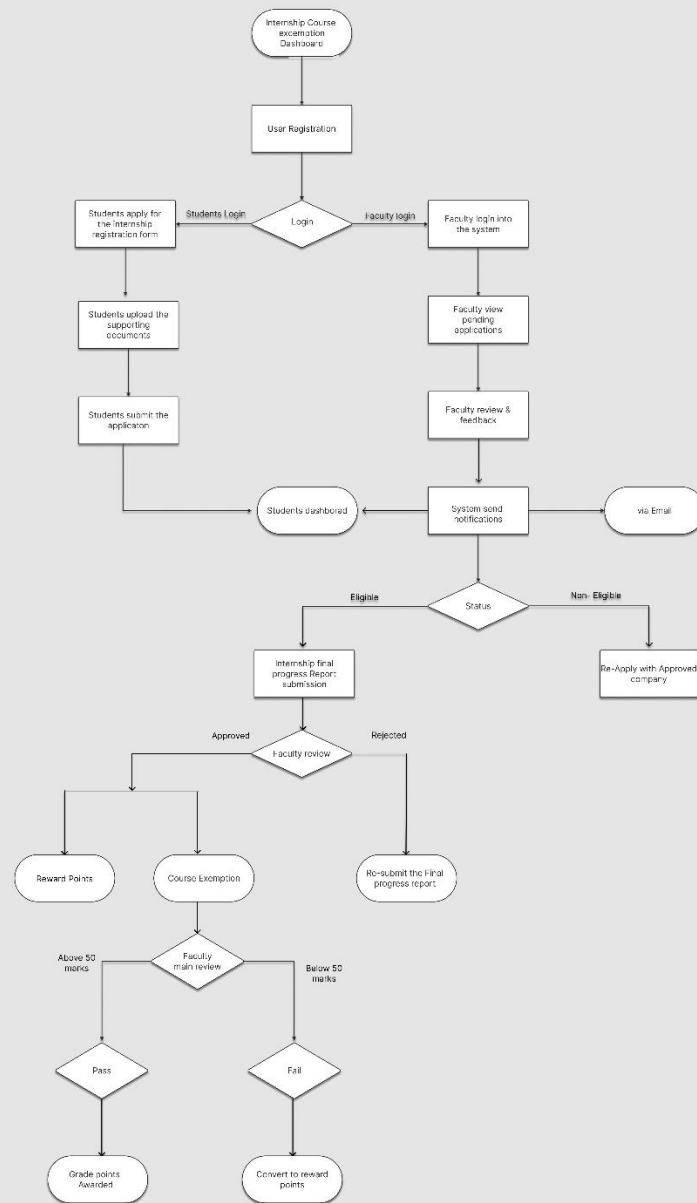
Reliability

- **Uptime:**
 - The system should maintain an uptime of 99.9% to ensure availability for users.
- **Error Handling:**
 - The system must handle errors gracefully and provide meaningful error messages to users.

Security

- **Data Protection:**
 - All user data, including personal and application data, must be stored securely with encryption.
- **Authentication and Authorization:**
 - The system must ensure that only authorized users can access certain functionalities based on their roles.
- **Compliance:**
 - The system must comply with relevant data protection regulations, such as GDPR or CCPA.

Flow chart:



Login Credentials:

Internship Portal

Student login



Faculty login



Faculty Login & Student Login

Faculty Login



BIT INTERNSHIP PORTAL

SIGN IN USING BIT ACCOUNT


Student Login




BIT INTERNSHIP PORTAL


SIGN IN USING BIT ACCOUNT

Student Dashboard:

 Welcome, User103!
7376221CS132

Internship Portal 



 APPLY

Registration form



View Status

Logout

