

Software Requirement Specification for TAC Portal

Name	Kavin Kumar P
Roll no	7376212CT124
Seat no	265
Project ID	53
Problem Statement	Tac Portal

1. Introduction

1.1. Purpose:

The purpose of this document is to present a detailed description of the Tac portal. 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 Project:

- This software system will serve as a portal for the Technical Approval Committee (TAC), enabling students to submit their projects and receive their rewards. From an administrative perspective, this system will provide a comprehensive **analytical dashboard for project oversight**.

- Administrators have the ability to approve or reject projects. Once a project is approved, students can schedule an appointment using their accepted PTAC ID. The system will calculate the number of days between the approval date and the current date. If this duration is less than 30 days, a warning will be displayed indicating that students can claim only 30% of their rewards, which is contingent on the number of days since approval.

2. System Overview:

2.1. Users:

1. Students:

They have the ability to submit applications for TAC approval, upload relevant project documents, monitor the status of their application, schedule appointments following approval, and review their TAC interaction history.

2. Admins:

Review submitted TAC applications, approve or reject applications (with remarks), manage appointments, schedule meetings, and access analytical dashboards for project oversight

2.2. Features:

1. Login and registration:

Students can register for an account or login with their existing account

2. TAC Application Submission:

Students can input relevant details regarding their project application including project title, description, objectives, and any

necessary attachments. Upon completion, the application is submitted to the admin interface for review and further processing

3. Application Status:

Students can view the current status of their application and also see the history logs in the option Activity

4. Appointment Booking:

Student with approved TAC ID can request for Project review after completion of 30 days

5. Admin Access:

Admin can view all submitted TAC applications in a category of either software or hardware, view application details, approve or reject the application with suitable remarks, schedule meetings.

6. Admin's Analytical Dashboard:

Admin can view the number of applications by category, appointments request and also see the latest log of applications

```

    usecaseDiagram
        actor Actor
        participant DB as DB
        participant System
        participant Form as Tac Request Form
        participant Activity as Activity of student
        participant Review as Book an review appointment
        participant Approved as Approved tac ID
        participant Duration as Duration (current date - Date of approval)
        participant Eligible30 as Eligible 30% or(accoding to days), and show warning
        participant Eligible100 as eligible for 100% (no warning)
        participant NameRoll as Name, Roll
        participant SkillSet as Skill set
        participant UI as UI
        participant Frontend as Frontend
        participant Backend as Backend
        participant Components as list of components
        participant Yes as Yes
        participant Excel as Excel sheet for proof

        Actor --> Validates{Validates account}
        Validates --> DB : Fetch data and compare
        Validates --> Invalid[Invalid Details] : NO
        Validates --> System : yes
        System --> Form
        System --> Activity
        System --> Review
        Review --> Approved
        Review --> Duration
        Duration --> Eligible30 : if day >= 30
        Eligible30 --> Eligible100 : True
        Eligible30 --> Eligible30 : False
        Form --> NameRoll : 1
        Form --> SkillSet : 1
        Form --> UI : 1
        Form --> Frontend : 1
        Form --> Backend : 1
        NameRoll --> SkillSet : Hardware
        SkillSet --> UI : Software
        SkillSet --> Components : Hardware
        Components --> Yes
        Yes --> Excel
        DB --> DB : Fetch Data
    
```

The diagram illustrates the process of a Tac Request Form. It begins with an Actor interacting with a decision point 'Validates account'. If the account is invalid, the process leads to 'Invalid Details'. If valid, it proceeds to the 'System' block, which contains three sub-processes: 'Tac Request Form', 'Activity of student', and 'Book an review appointment'. The 'Book an review appointment' process leads to 'Approved tac ID' and 'Duration (current date - Date of approval)'. The 'Duration' process leads to a decision 'if day >= 30'. If true, it leads to 'eligible for 100% (no warning)'. If false, it leads to 'Eligible 30% or(accoding to days), and show warning'. The 'Tac Request Form' process leads to a list of five items: '1. name and roll no', '2. Project category', '3. Title', '4.No of students', and '5. Provisional document'. These items are then mapped to specific components: 'Name, Roll' (1), 'Skill set' (2), 'UI' (3), 'Frontend' (4), and 'Backend' (5). The 'Skill set' component is further divided into 'Hardware' and 'Software'. The 'Hardware' component leads to a table of components: 'Mtrs', 'ECE', 'Aero', 'CAD', 'matlab', 'fluid', 'PLC', 'Lab view', 'Propulsi on', 'Robotics', 'pcb', and 'Avionics'. This table leads to 'list of components', which then leads to 'Yes' and finally to 'Excel sheet for proof'.

```
graph LR; A[1. Name, Roll_no, Id  
2. Project category  
3. Title  
4. No. of students  
5. Provisional Attachment] --> B[No of software project  
No of hardware project  
Approve or reject the Project Id  
Show the Appointment]; B -- Reject --> C[Prompt for remarks]; B -- Approve --> D[Store the date when it is approved(ID)]; B --> E[Pick a date for appointment];
```

The flowchart illustrates the project approval process. It begins with a large box containing five input fields: "1. Name, Roll_no, Id", "2. Project category", "3. Title", "4. No. of students", and "5. Provisional Attachment". An arrow points from this box to a central box. This central box contains four sub-sections: "No of software project", "No of hardware project", "Approve or reject the Project Id", and "Show the Appointment". From the "Approve or reject the Project Id" section, two arrows branch out: one labeled "Reject" pointing to a box labeled "Prompt for remarks", and another labeled "Approve" pointing to a box labeled "Store the date when it is approved(ID)". Finally, an arrow points from the "Show the Appointment" section to a box labeled "Pick a date for appointment".

3.1 Functional Requirements:

- **User Management:**
 - Students can register and login.
 - Admins have access control with an analytical dashboard and dedicated features .
- **TAC Application:**
 - Students can submit applications with appropriate details
 - Application form contains:
 - Title of Project
 - Category of the project
 - Number of students involved
 - Provisional document attachment
- **Application Status:**
 - Students can view the current status of their application
 - If the application is rejected then the remarks is shown
 - Students can also see the logs of their applications
- **Appointment Scheduling (After Approval):**
 - Students with approved TACs can request appointments after completion of 30 days
- **Admin Dashboard:**
 - Admins can view a list of all submitted TAC applications.
 - Applications can be filtered by category (software, hardware).
 - Admins can view details of each application.
 - Admins can approve or reject applications with suitable remarks.
 - Admins can schedule meetings for accepted appointments.

- **Analytics Dashboard:**

- Admin can view the number of applications by its category
- Number of appointments is requested based on the category

3.2. Non-Functional Requirements:

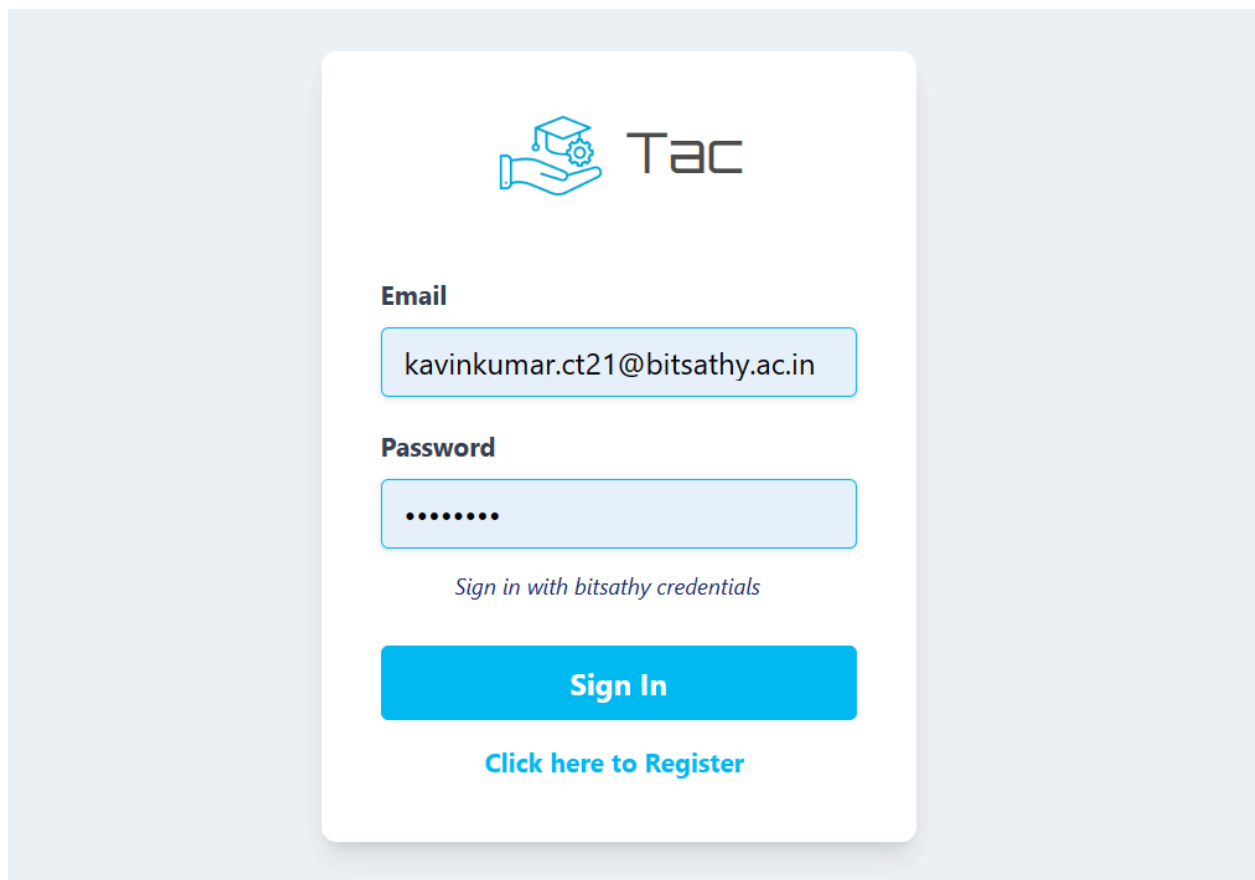
- **Performance:** The system must respond to user actions within 2 seconds to ensure efficient usability and must handle a concurrent user load of at least 100 users without significant performance degradation.
- **Security:** User data must be encrypted during transmission and storage, and access to sensitive functionalities should be restricted to authorized admin users through secure authentication mechanisms.
- **Usability:** The user interface should be intuitive and user-friendly, with clear and concise error messages provided to guide users in case of input errors or system failures.
- **Reliability:** The system should be available 24/7 with minimal downtime and should have a backup and recovery mechanism in place to prevent data loss in case of system failures or crashes.
- **Scalability:** The system should be designed to accommodate an increasing number of users and data volume over time, and it should be scalable to support additional features and functionalities as per future requirements.

Stack:


Front End	Vue Js, Tailwind css
Backend	Node Js, Express
Data Base	MongoDB

Prototype of the Project:

1. Login form



The image shows a login form prototype for a system named 'Tac'. The form is centered on a light blue background. At the top, there is a logo consisting of a blue icon of a hand holding a graduation cap with a gear, followed by the text 'Tac' in a bold, sans-serif font. Below the logo, the form has two input fields. The first is labeled 'Email' and contains the text 'kavinkumar.ct21@bitsathy.ac.in'. The second is labeled 'Password' and contains a series of dots. Below the password field, there is a link that says 'Sign in with bitsathy credentials'. At the bottom of the form, there is a large blue button with the text 'Sign In' in white. Below the button, there is a link that says 'Click here to Register' in blue.



Email

kavinkumar.ct21@bitsathy.ac.in

Password


.....

Sign in with bitsathy credentials

Sign In

[Click here to Register](#)

2. Register Form

 Tac

Name

Roll no

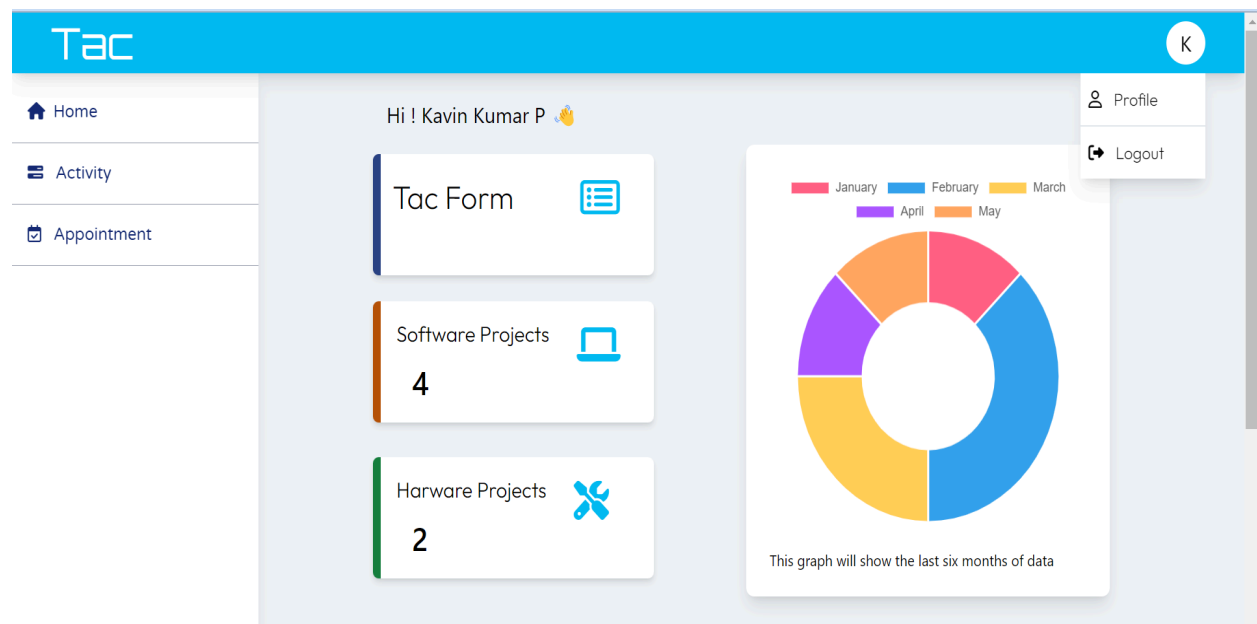
Email

Password

Register

[Click here to login](#)

3. Student's view



4. Tac form:

Name *	Roll No *
<input type="text" value="Name"/>	<input type="text" value="Roll No"/>
Category *	Title *
<input type="text" value="Select Category"/>	<input type="text" value="Title"/>
Number of Students *	PDF Document *
<input type="text" value="1"/>	<div>Choose File No file chosen</div>
Student 1 Name*	
<input type="text" value="Student Name"/>	
Student 1 Roll No *	
<input type="text" value="Roll No"/>	
<input type="button" value="Submit"/>	

5. Activity page:

Your Activity							
ROLL NO	CATEGORY	TITLE	NO OF STUDENTS	STUDENT 1	ROLL NO	PROVISIONAL DOCUMENT	STATUS
7376212CT124	Software	Budget buddy	1	Kavin Kumar p	7376212CT124	7376212CT124-PA-08.02.2023	Initiated
7376212CT124	Software	News Website	1	Kavin Kumar P	7376212CT124	7376212CT124-PA-10.04.2024	Rejected
7376212CT124	Software	Budget buddy	1	Kavin Kumar p	7376212CT124	7376212CT124-PA-08.02.2023	Approved
7376212CT124	Software	News Website	1	Kavin Kumar P	7376212CT124	7376212CT124-PA-10.04.2024	Approved

6. Appointment Page:

Name *

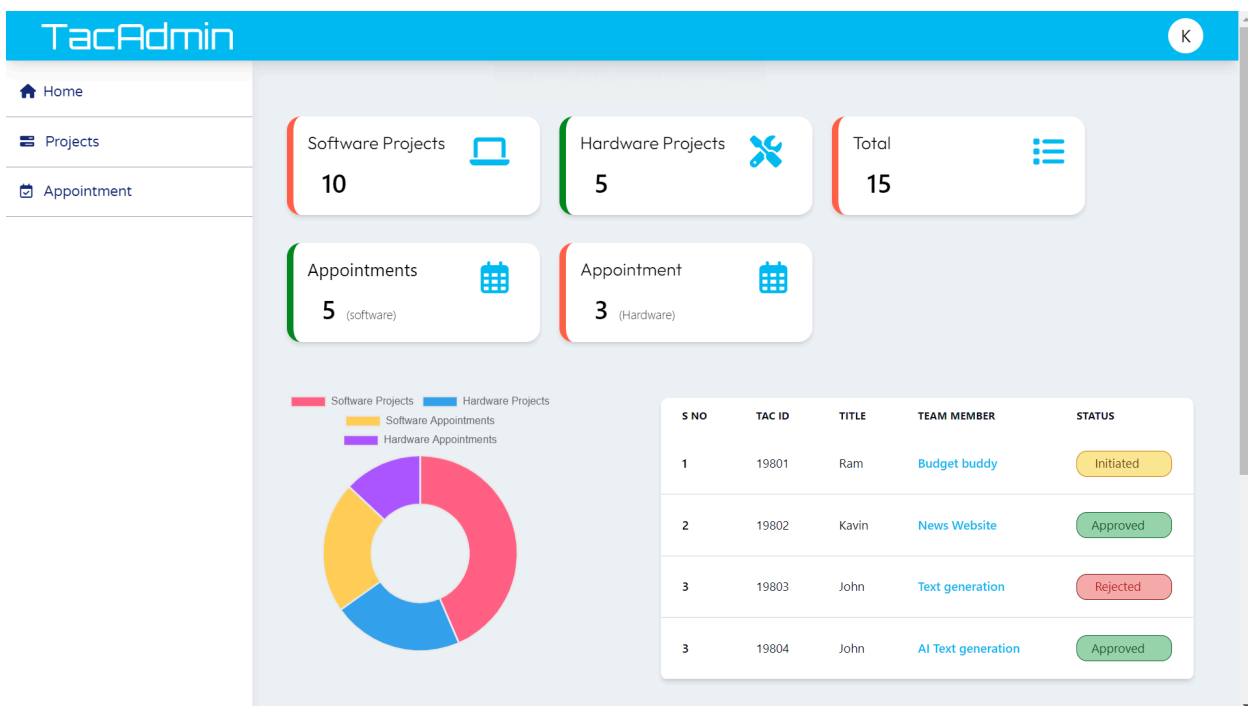
Roll No *

Tac ID *

Appointment date*

Book

7. Admin's View:



8. Approval Page

TAC

S

Home

Projects

Appointment

Requested Application for approval

s.no	Tac Id	Title	Category	No of Students	Team Member	Roll no	List of components	Provisional Pdf	Action
1	19453	Test1	Software	1	John	7376212CE122	-		
2	19454	Test2	Hardware	2	Ram	7376212IT122	No		
3	19455	Test3	Hardware	3	Kiran	7376212IT122	No		
5	19457	Test5	Software	4	kavin	7376212CT124	-		

9. Conform the date

TAC

S

Home

Projects

Appointment

Requested date from Approved tac

s.no	Tac Id	Title	Category	No of Students	Team Member	Roll no	no of days	Booked Date	Preferred data
1	19453	Test1	Software	1	John	7376212CE122	30	25-4-24	
2	19454	Test2	Hardware	2	Ram	7376212IT122	29	25-4-24	
3	19455	Test3	Hardware	3	Kiran	7376212IT122	31	25-4-24	
5	19457	Test5	Software	4	kavin	7376212CT124	10	25-4-24	