STAGE 1: Planning and Requirement Gathering.

Name	KAVIN KUMARAN M
Roll no	7376221CS192
Seat no	244
Project ID	4
Project Title	Hackathon Events Forum (BIT HACK)
Domain	Office Special Labs

TECHNICAL COMPONENTS:

Component	**Tech Stack**
Backend	NODE JS AND EXPRESS JS
Frontend	Angular
Database	MongoDB
API	RESTful services

PROBLEM STATEMENT :

The platform should cater to both organizers and participants, providing features that streamline event management, facilitate team formation, and knowledge sharing.

IMPLEMENTATION TIMELINE:

Phase	**Deadline**	**Status**	**Stage**
Phase 1	03-05-2024	**In Progress**	Planning and requirement gathering
Phase 2		**Not Started**	UI development
Phase 3		**Not Started**	Database design and implementation
Phase 4		**Not Started**	Back-end development
Phase 5		**Not Started**	Testing and implementation
Phase 6		**Not Started**	Deployment

PROJECT-FLOW:

Purpose :

This Software Requirements Specification (SRS) document outlines the requirements for an BITHACK 24- Hackathon Events Forum. The platform likely revolves around fostering a community of hackers, developers, and tech enthusiasts interested in participating in hackathons.

Scope :

1.Event Management:

Implementing features to manage hackathon events, including event listing, registration, scheduling, and coordination with event organizers.

2.User Registration and Profiles:

Allowing users to create accounts, customize profiles, and manage their participation in hackathons.

3.Community Features:

Providing forums, chat rooms, and discussion boards for participants to interact, share ideas, seek collaborators, and receive support from mentors.

4.Project Management:

Offering tools for team formation, project collaboration, version control, and submission of project deliverables.

5.Notification System:

Implementing a notification system to keep users informed about upcoming hackathons, team invitations, event updates, and deadlines.

6.Project Showcase and Judging:

Creating a platform for participants to showcase their projects, receive feedback from peers and judges, and compete for awards or prizes.

SYSTEM OVERVIEW:

User Interface(UI) :

User UI :

Provides functionalities for users to browse event listings, Community Interaction, Team formation, Resource library, Discussion Forum, Notification for team joining request and Private Message System for Teams.

Admin UI :

Enables managing events, updating event details, event collaboration, managing users, and accessing results.

Database :

Stores user information, event details, resources, team data, forum data, notification data and chat data.

Notification System:

Sends emails to users regarding team formations and event registration, deadlines, and other relevant information.

Chat Forum Module:

Facilitates communication among all the Users and Organizers through persistent chat threads.

<u>Data Flow :</u>

Student actions:

- Explore the list of hackathon events available on the platform,view event details, and learn about upcoming hackathons.
- Register to participate in hackathon events.
- Join Teams or create new Teams for the hackathon events.
- Participate In discussion forums and ask queries about events.
- Interacts among participants through the discussion forum.
- Users can send messages to the team members.

Admin Actions:

- ★ Manages Events, including adding, editing, and deleting events.
- ★ Admin can view all the team's information.
- ★ Communication between Participants ,sponsors and stakeholders.
- ★ Access a messaging interface within the website where they can compose emails.
- ★ Admins can view a list of all registered teams
- ★ Applications can be filtered by category (software, hardware).
- ★ Admin can have the access to remove the team in the event.
- ★ Admin can schedule meetings for selected teams.

Key Considerations :

- ➔ Secure user authentication and authorization for access control.
- ➔ Scalable database and infrastructure to handle user growth and event information.

➔ User-friendly interface with intuitive navigation and clear information presentation.
➔ Efficient content delivery to minimize loading times.
➔ Robust project submission and management system with file upload functionalities.
➔ Discussion forum to ensure a safe and productive environment.

Non-Functional Requirements:

- Performance:

Responsiveness : The platform should load pages and content quickly, even on slower internet connections.

Scalability: The platform should be able to handle a growing number of users and courses without performance degradation.

- Security:

User Authentication : Secure password hashing and storage to protect user credentials.

Authorization : Implement role-based access control to restrict unauthorized access to sensitive information.

- Usability:

User Interface (UI): The platform's UI should be intuitive, user-friendly, and visually appealing.

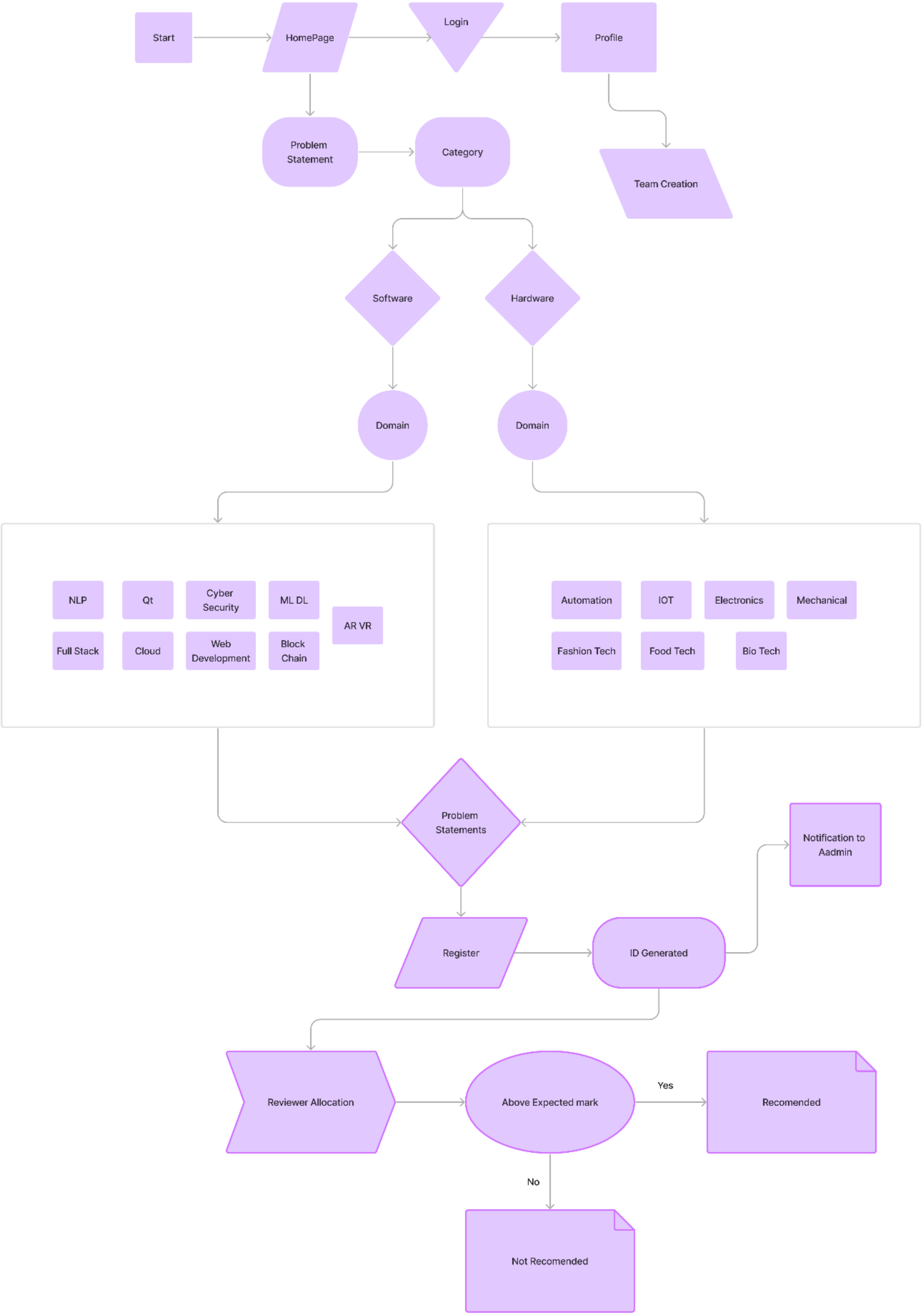
Navigation: Navigation elements should be clear and easy to understand, allowing users to find desired information efficiently.

- 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


Start
HomePage
Login
Profile
Problem Statement
Category
Team Creation
Software
Hardware
Domain
Domain
NLP
Qt
Cyber Security
ML DL
AR VR
Full Stack
Cloud
Web Development
Block Chain
Automation
IOT
Electronics
Mechanical
Fashion Tech
Food Tech
Bio Tech
Problem Statements
Notification to Aadmin
Register
ID Generated
Reviewer Allocation
Above Expected mark
Yes
Recomended
No
Not Recomended