Deliverable 1 - Proposal
CSCE 5430 (Spring 2024)

I) Project Details

 # Project Title : Team Chat App

 # Group Name : Group-7

 # Group Members :

1. Shivanandha reddy vasudevula (11709232)
2. Srichandan Kota (11711406)
3. Kantumutchu Dinesh(11638076)
4. Gana Deekshith (11614888)
5. Sandeep Chowdary Ari(11700315)
6. Bhanu Prasad Krishna Murthy(11654250)
7. Swapna Sonti (11653211)
8. Venkata sai shankar koppula(11692147)

II) Description

Our project is focused on creating a team chat application utilizing Next.js 13, React, Socket.io, Prisma, Tailwind, and MySQL to offer a comprehensive suite of features for enhanced team collaboration. The application will support server creation, real-time messaging, media attachments, and various communication channels, coupled with advanced user management capabilities. Emphasizing a user-friendly and customizable interface, it will incorporate dark and light modes, secure authentication, and a scalable architecture.

 # Development Environment

- Frontend: Built with React for dynamic UIs, emphasizing component-based architecture.

- Backend: Utilizes Node.js and Next.js 13 for server-side logic, enhancing performance and SEO.

- Real-time Communication: Socket.io for instant messaging and live updates.

- Database: MySQL for data management, with Prisma as the ORM for efficient queries.

- Styling: Tailwind CSS for customizable, utility-first design.

- Collaboration Tools: Git and GitHub for version control, Google Meet for communication, and Trello for task management.

 # Key Features

- Server Management: Facilitates server creation for team organization.

- Messaging: Supports real-time messaging with edit/delete functionalities.

- Multimedia Sharing: Enables sharing of files, images, and videos.

- Diverse Communication Channels: Offers text, video, and audio channels.

- Private Conversations: Supports one-on-one and group chats.

- User Management: Advanced features for inviting users and managing roles.

- Customization: Customizable themes with dark and light modes.

 # Testing and Maintenance

- Testing Strategy: Incorporates unit, integration, and security testing, alongside user acceptance testing, to ensure reliability and performance.

- Maintenance Plan: Includes bug tracking, user feedback loops, and regular documentation and training updates to keep the application current and aligned with user needs.

For the development and deployment of the described team chat application, leveraging a combination of cloud-based services and development tools will provide the most efficient, scalable, and flexible platform. Given the technologies (Next.js 13, React, Socket.io, Prisma, Tailwind, and MySQL) and the project requirements (scalability, real-time communication, and security), the following platforms and services are recommended:

 # Development and Collaboration Platforms

- GitHub: For version control and collaboration. GitHub offers powerful CI/CD pipelines through GitHub Actions, facilitating automated testing and deployment processes.

- Visual Studio Code (VS Code): A versatile code editor with extensive support for JavaScript, React, and Node.js development. It offers real-time collaboration features through Live Share, enhancing team productivity.

- Trello : For project management and task tracking, allowing the team to organize development sprints, track progress, and prioritize features and bugs.

 # Database Hosting

- PlanetScale: A serverless database platform that supports MySQL and is built for scalability and ease of use. It offers automatic branching and CI/CD integration, which can be beneficial for agile development workflows.

III) Project Timeline

Our project roadmap details a structured timeline for developing an advanced team chat application, integrating a blend of cutting-edge technologies and functionalities aimed at optimizing team collaboration.

 # Phase 1: Initiation and Planning (Jan 23 - Feb 5, 2024)

- Objectives: Outline project scope with features like real-time messaging, file sharing, and advanced user management.

- Team Formation: Assign roles within development, UX/UI design, QA, and project management.

- Tools Setup: Utilize GitHub for version control and Jira for project management.

- Key Deliverables: Complete a project proposal and establish a risk management framework.

- Deadline: February 5, 2024.

 # Phase 2: System Design and Tech Selection (Feb 6 - Feb 26, 2024)

- System Design: Architect a scalable system using a microservices approach for flexibility and maintenance.

- Database & Tech Stack: Opt for MySQL for data storage, Next.js and React for frontend development, Socket.io for real-time communication, and Prisma for ORM.

- Security Planning: Implement JWT for secure authentication and HTTPS for encrypted data transmission.

- Deadline: February 26, 2024.

 # Phase 3: Core Functionality Development (Feb 27 - Mar 18, 2024)

- Messaging & File Sharing: Develop seamless one-on-one and group chat functionalities, including support for various file types.

- User Management: Create comprehensive user profiles, friend lists, and group memberships with Prisma and MySQL.

- UI Customization: Enable dark and light modes with Tailwind CSS for a personalized user experience.

- Deadline: March 18, 2024.

 # Phase 4: Enhancements and Integrations (Mar 19 - Apr 8, 2024)

- Communication Channels: Integrate video and audio call functionalities using WebRTC for direct peer-to-peer connections.

- Server & Channel Management: Implement tools for creating and managing dedicated communication spaces.

- Collaboration Tools: Add shared calendars and task management features for enhanced team coordination.

- Deadline: April 8, 2024.

 # Phase 5: Testing, Final Review, and Handover (Apr 9 - Apr 29, 2024)

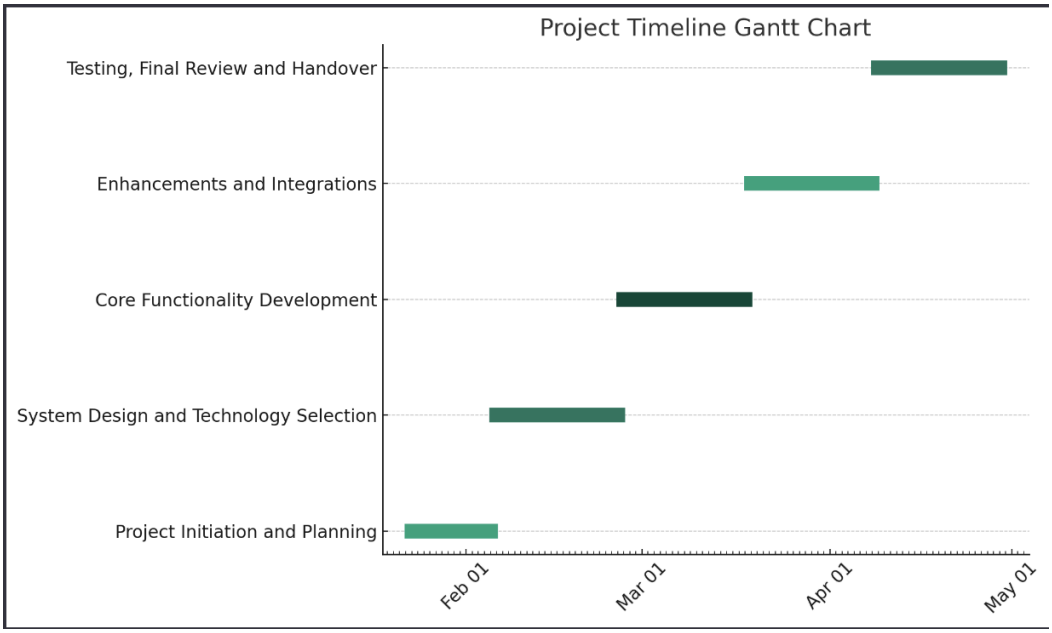
- Testing: Conduct unit, integration, and end-to-end testing using Jest and Cypress to ensure functionality and reliability.

- Security Audits: Perform thorough security assessments to patch vulnerabilities and ensure data protection.

- User Feedback: Facilitate user testing sessions for real-world usability insights and final adjustments.

- Deadline: April 29, 2024.

For the specified timeline and tasks, here is the respective Gantt Chart:



IV) Risk Management

 # Risk 1: Expanding Project Scope (Scope Creep)

- Detailed Monitoring Approach: To prevent scope creep, our team will conduct thorough weekly reviews focusing on the alignment between current project progress and the initially defined scope. These reviews aim to spot any deviations or expansions in scope early.

- Dynamic Reevaluation Tactics: Our strategy includes setting up a systematic monthly review process to critically assess and evaluate any proposed changes or expansions to the project's scope. This will involve a detailed analysis of their potential impact on both the project timeline and resource allocation, with adjustments to the project plan communicated swiftly to all involved stakeholders.

- Robust Contingency Measures: To manage scope changes effectively, we will establish a rigorous change control process. This process will be designed to meticulously evaluate and approve any modifications to the project's scope, grounded in well-defined decision-making criteria. Additionally, we will earmark a specific contingency buffer within our project timeline and budget, earmarked explicitly for accommodating unforeseen changes in scope.

 # Risk 2: Unforeseen Technical Challenges and Delays

- Proactive Monitoring System: Our team plans to deploy an advanced issue tracking system to log and closely monitor any technical challenges as they emerge. Additionally, regular technical review meetings will be scheduled to diligently assess project progress and identify any potential technical bottlenecks.

- Strategic Reevaluation Process: Key project milestones will trigger technical audits aimed at assessing the effectiveness of our chosen technology stack and pinpointing optimization opportunities. The insights gained from these audits, coupled with our ongoing assessment of current technology trends and firsthand project experiences, will inform our updated risk assessments.

- Comprehensive Contingency Planning: To tackle technical issues effectively, we will curate an extensive knowledge base and documentation, ensuring swift resolution of common technical challenges. Forming strategic alliances with technology vendors or engaging with relevant technical communities will provide additional support for navigating critical technical challenges. Moreover, our contingency plan includes a flexible approach to resource allocation, allowing for the introduction of external expertise or the reallocation of internal resources to address pressing technical issues as needed.

 # Risk 3: Security Vulnerabilities and Data Privacy Issues

- Continuous Security Monitoring: To safeguard against security risks, we will implement state-of-the-art continuous monitoring tools. These tools will be instrumental in detecting any anomalies, vulnerabilities, and potential security breaches in real-time. Regular security audits, including penetration testing and vulnerability scanning, will further bolster our application's security posture.

- Ongoing Risk Reevaluation: Each major project milestone, or after any significant changes to project scope or technology, will prompt an updated security risk assessment. Keeping abreast of the latest security threats and vulnerabilities that could impact our project is crucial for adjusting our security measures and strategies accordingly.

- Elaborate Contingency Framework: A detailed incident response plan will be at the core of our contingency measures, outlining clear procedures for addressing security incidents effectively. This plan will be complemented by robust data backup and recovery processes, which will undergo regular testing to ensure minimal data loss in the event of a breach. Collaboration with external security experts or services will enable rapid response and mitigation efforts for significant security issues. Finally, implementing strict access controls and ensuring the encryption of data, both at rest and in transit, will form the bedrock of our efforts to protect sensitive information.

By embracing these elaborate strategies, we aim to proactively manage the project's top risks, thus paving the way for a seamless progression towards the successful completion of our objectives.

We even analyzed the risk management matrix for better priority solving tasks.

- **High Risk:**

        - Technical Challenges (Probability: Medium, Impact: High)
        - Scope Creep (Probability: Medium, Impact: High)
- **Moderate Risk:**
    - Security and Data Privacy (Probability: Low, Impact: High)

V) Team Members and Roles

Based on the roles and technology assignments provided in the screenshots, here's a description for each team member's role within the project:

 # Srichandan Kota

- Roles: Project Management Lead, Implementation Lead for front end, Testing Lead, Documentation Lead, Demo and presentation Lead

- Technologies: Next.js 13

- Responsibilities: Srichandan is responsible for managing the overall project, leading the implementation of the front end, ensuring the application's features are thoroughly tested, overseeing the creation of comprehensive documentation, and presenting the final product.

 # Swapna Sonti

- Role: Implementation Lead for front end

- Technology: React

- Responsibilities: Swapna will lead the development of the application's front end using React, focusing on building a dynamic and responsive user interface that integrates seamlessly with the back end.

 # Sandeep Chowdary Ari

- Role: System Administrator Lead

- Technology: socket.io

- Responsibilities: Sandeep will oversee the real-time communication capabilities of the project, ensuring stable and efficient messaging and data transfer over the web sockets using socket.io.

 # Venkata Sai Shankar Koppula

- Role: Design Lead, Demo and presentation Lead

- Technology: Tailwind CSS

- Responsibilities: Venkata Sai will lead the design aspects of the project using Tailwind CSS, crafting an intuitive and attractive user interface, and will co-lead in the demonstration and presentation of the project.

 # Shivanandha Reddy Vasudevula

- Role: System Administrator Lead

- Technology: MySQL (Planetscale)

- Responsibilities: Shivanandha will manage the database aspects, ensuring the data integrity, performance, and scalability of the MySQL database, and will oversee the system's administration.

 # Gana Deekshith

- Role: Configuration Management Lead

- Technology: Clerk

- Responsibilities: Gana will manage the configuration of the project, focusing on implementing Clerk for authentication services, guaranteeing secure access control and user management.

 # Kantumutchu Dinesh

- Role: Configuration Management Lead

- Technology: Chakra UI

- Responsibilities: Dinesh will lead the configuration management for the project, ensuring the UI components are well-integrated and functional across the application with Chakra UI.

 # Bhanu Prasad Krishna Murthy

- Role: Documentation Lead

- Technology: Prisma

- Responsibilities: Bhanu Prasad will be responsible for overseeing the documentation process for the project and managing the interactions between the data model and the database using Prisma ORM for seamless data operations.

Each member will explore their assigned technologies to become proficient and will collaborate to make executive decisions on the project's direction. Team meetings will be scheduled regularly to facilitate knowledge sharing and joint tutorial sessions to ensure team confidence in using the selected technologies.

VI) Member Contribution Table

Below is a structured table detailing each team member's contributions to the deliverable:

Team Member	Role(s)	Technology Assigned	Contributions for Deliverable
Srichandan Kota	Project Management Lead, Implementation Lead for front end, Testing Lead, Documentation Lead, Demo and presentation Lead	Next.js 13	Managed project planning, coordinated team roles, led the development of the frontend, planned testing strategies, began documentation framework, and prepared initial presentation outlines.
Swapna Sonti	Implementation Lead for front end	React	Researched and prepared a tutorial on React components for team training, contributed to initial UI mock-ups.

Sandeep Chowdary Ari	System Administrator Lead	socket.io	Investigated real-time communication protocols, set up a basic chat server using socket.io for demonstration purposes.
Venkata Sai Shankar Koppula	Design Lead, Demo and presentation Lead	Tailwind CSS	Developed the design system for the project using Tailwind, created responsive design templates for the initial deliverable.
Shivanandha Reddy Vasudevula	System Administrator Lead	MySQL (Planetscale)	Drafted the initial database schema, set up a development database with sample data for early-stage testing.
Gana Deekshith	Configuration Management Lead	Clerk	Outlined the authentication flow, researched Clerk documentation, and established a guide for integrating authentication in the project.
Kantumutchu Dinesh	Configuration Management Lead	Shadcn UI	Compiled a UI component library with Shadcn UI, contributed to the design of the frontend elements.
Bhanu Prasad Krishna Murthy	Documentation Lead	Prisma	Documented the initial database models and relationships, prepared guidelines for database interaction using Prisma.

This table represents the foundational efforts and contributions of each team member, setting the stage for the development of the project deliverable.