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

The goal of our project is to develop a team chat application that offers an extensive feature set for improved team collaboration. We will achieve this by utilizing Next.js 13, React, Socket.io, Prisma, Tailwind, and MySQL. Along with sophisticated user management features, the application will enable media attachments, multiple communication channels, server construction, and real-time messaging. It will include a scalable architecture, secure authentication, and both dark and light modes, with an emphasis on an intuitive and adaptable user interface.

 # Development Environment

- Frontend: Component-based architecture is prioritized with React-built dynamic user interfaces.

- Backend: Improves SEO and performance by using server-side logic using Node.js and Next.js 13.

- Real-time Communication: For live updates and immediate messaging, use Socket.io.

- Database: Prisma serves as the ORM for effective querying, and MySQL is used for data administration.

- Style: For adaptable, utility-first design, use Tailwind CSS.

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

 # Key Features

- Server Management: Assists in setting up servers for group management.

- Messaging: Offers edit and delete features for real-time messaging.

- Multimedia Sharing: Facilitates the exchange of documents, pictures, and videos.

Provides Text, Video, and Audio Channels for a Variety of Communication.

- Private Chats: Facilitates individual and group discussions.

- User Management: Powerful tools for assigning responsibilities and sending invitations to users.

- Customization: Dark and light mode themes that can be changed.

 # Testing and Maintenance

- Testing Strategy: To guarantee performance and dependability, unit, integration, and security testing are combined with user acceptability testing.

- Maintenance Plan: To maintain the program up to date and in line with user needs, it includes bug tracking, user feedback loops, and frequent documentation and training updates.

The most effective, scalable, and adaptable platform for the creation and implementation of the team chat application as described will come from utilizing a combination of cloud-based services and development tools. The following platforms and services are suggested in light of the technologies (Next.js 13, React, Socket.io, Prisma, Tailwind, and MySQL) and project requirements (scalability, real-time communication, and security):

 # Development and Collaboration Platforms

- GitHub: For teamwork and version control. Through GitHub Actions, GitHub provides robust CI/CD pipelines that streamline deployment and testing procedures.

- Visual Studio Code (VS Code): A feature-rich code editor that offers comprehensive support for Node.js, React, and JavaScript programming. Through Live Share, it provides real-time collaboration tools that increase team productivity.

- Trello: This project management and task tracking tool helps the team plan development sprints, monitor advancement, and rank features and defects.

 # Database Hosting

- PlanetScale: A user-friendly, scalable serverless database platform that supports MySQL. It provides CI/CD integration and automated branching, which are advantageous for agile development processes.

III) Project Timeline

Our project plan outlines a methodical schedule for creating a sophisticated team chat application that combines a variety of state-of-the-art features and technology to maximize teamwork.

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

- Goals: Describe the functionality of the project, including file sharing, real-time messaging, and sophisticated user management.

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

- Tool Setup: Use Jira for project management and GitHub for version control.

Principal Outcomes: Finalize a project proposal and create a framework for risk management.

- February 5, 2024 is the deadline.

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

- System Design: For flexibility and upkeep, design a scalable system with a microservices approach.

Database & Tech Stack: Select MySQL for storing data, Next.js and React for front-end development, Socket.io for instantaneous communication, and Prisma for object-relation mapping.

- Security Planning: Use HTTPS for encrypted data transport and JWT for safe authentication.

- February 26, 2024 is the deadline.

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

- Messaging & File Sharing: Provide smooth group and one-on-one chat features, along with file type support.

- User Management: Use Prisma and MySQL to create detailed user profiles, friend lists, and group memberships.

- UI Customization: For a customized user experience, use Tailwind CSS to enable the dark and bright modes.

Closing date: March 18, 2024.

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

- Communication Channels: Use WebRTC to integrate audio and video call features for peer-to-peer connections that are direct.

- Server & Channel Management: Put in place systems for setting up and overseeing specific areas for communication.

- Collaboration Tools: To improve team cooperation, provide task management systems and shared calendars.

Closing date: April 8, 2024.

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

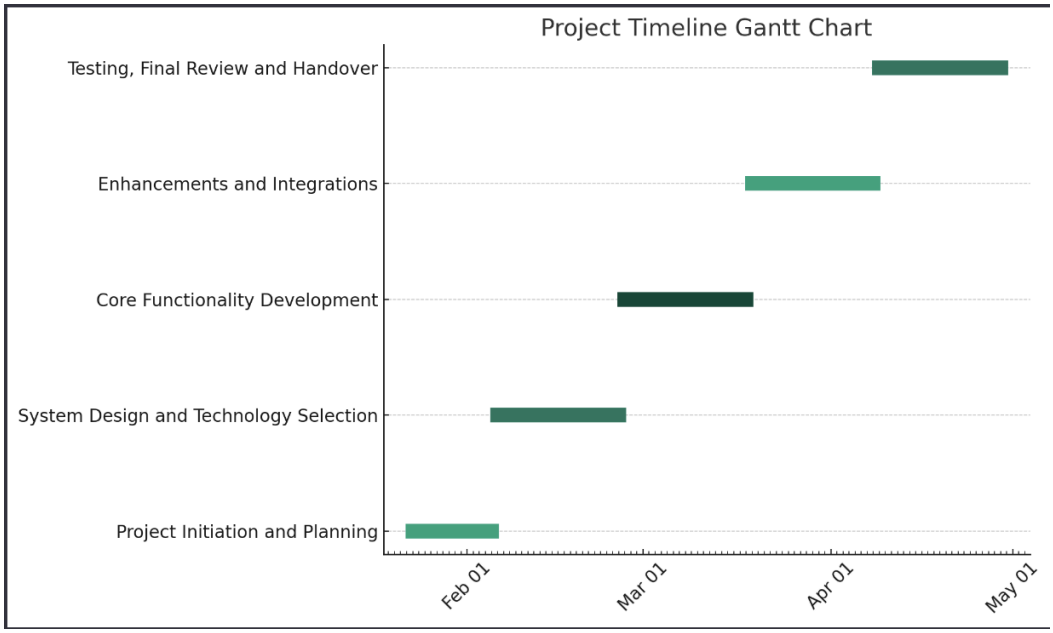
- Testing: Conduct unit, integration, and end-to-end testing using Jest and Cypress to ensure functionality and reliability.

- 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: Our team will carry out comprehensive weekly assessments that center on the alignment between the project's current progress and the originally established scope in order to prevent scope creep. The goal of these assessments is to identify any scope expansions or deviations early on.

- Dynamic Reevaluation Techniques: As part of our plan, we'll establish a methodical monthly review procedure to appraise and critically analyze any suggested modifications or enlargements of the project's purview. This will entail a thorough examination of their possible effects on the project schedule and resource distribution, with prompt notification of any modifications to the project plan to all relevant parties.

- Sturdy Contingency Measures: We'll set up a strict change control procedure to efficiently handle scope modifications. This procedure will be created to carefully assess and authorize any changes to the project's scope, based on clearly defined standards for decision-making. Furthermore, we will specifically budget for and include a particular contingency buffer in our project timeline to account for unanticipated changes in scope.

 # Risk 2: Unforeseen Technical Challenges and Delays

- Proactive Monitoring System: To record and carefully monitor any technical issues as they arise, our team intends to implement an advanced issue tracking system. Regular technical review meetings will also be planned in order to thoroughly evaluate the project's advancement and spot any possible technological impediments.

- Strategic Reevaluation Process: Technical audits designed to evaluate the efficacy of our selected technology stack and identify areas for optimization will be triggered by significant project milestones. Our updated risk assessments will be influenced by the information gathered from these audits, as well as our continuous evaluation of emerging technological trends and first-hand project experiences.

- Thorough Contingency Planning: We will create a vast knowledge base and documentation to address technical problems efficiently and guarantee prompt resolution of typical technical difficulties. For further assistance in overcoming important technological obstacles, consider forming strategic connections with technology providers or participating in pertinent technical groups. In addition, our contingency plan has an adaptable resource allocation strategy that permits the addition of outside knowledge or the reallocation of in-house personnel to handle urgent technical problems as required.

 # Risk 3: Security Vulnerabilities and Data Privacy Issues

- Continuous Security Monitoring: We'll use cutting-edge continuous monitoring technologies to protect against security threats. Real-time anomaly, vulnerability, and potential security breach detection will be made possible by these tools. The security posture of our application will be further strengthened by routine security audits that include vulnerability scanning and penetration testing.

- Ongoing Risk Reevaluation: A new security risk assessment will be required following any significant changes to the project's scope or technology, or after each major project milestone. In order to modify our security procedures and strategies appropriately, it is imperative that we stay up to date on the most recent security threats and vulnerabilities that may affect our project.

- Comprehensive Contingency Framework: The foundation of our contingency measures will be a thorough incident response plan that lays out precise steps for dealing with security problems in an efficient manner. Strong data backup and recovery procedures will be added to this strategy, and they will be regularly tested to guarantee that there will be as little data loss as possible in the case of a breach. Rapid response and mitigation efforts for major security threats will be made possible by collaboration with external security professionals or services. Ultimately, the cornerstone of our efforts to safeguard sensitive data will be the implementation of stringent access restrictions and the guarantee of data encryption while it is in transit and at rest.

By adopting these complex tactics, we hope to proactively manage the project's major risks and open the door for a smooth transition towards the accomplishment of our goals.

Even the risk management matrix was examined in order to improve work prioritization.

- **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 **#**

 # 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.

To become expert in the technologies they have been assigned, each member will investigate them, and they will work together to make executive choices on the project's course. Regular team meetings will be planned to promote information exchange and cooperative tutorial sessions to guarantee team proficiency with the chosen 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	Next.js 13	oversaw the front-end development process, planned testing techniques, started the documentation structure, oversaw project planning, organized team responsibilities, and created preliminary presentation outlines.
Swapna Sonti	Implementation Lead for Back end	React	contributed to the first UI mock-ups and conducted research for a tutorial on React components for team training.

Sandeep Chowdary Ari	System Administrator Lead	socket.io	examined real-time communication protocols and, as a demonstration, created a simple chat server with socket.io.
Venkata Sai Shankar Koppula	Design Lead, Demo and presentation Lead	Tailwind CSS	Tailwind was used to develop the project's design system, and responsive design templates were produced for the first deliverable.
Shivanandha Reddy Vasudevula	System Administrator Lead	MySQL (Planetscale)	created a development database with test data and drafted the first database structure for early testing.
Gana Deekshith	Configuration Management Lead	Clerk	created a methodology for incorporating authentication into the project, investigated Clerk documentation, and outlined the authentication flow.
Kantumutchu Dinesh	Configuration Management Lead	Shadcn UI	created a collection of UI components using Shadcn UI and helped with front-end element design.
Bhanu Prasad Krishna Murthy	Documentation Lead	Prisma	created standards for database interaction using Prisma and documented the original database models and linkages.

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