**Project Planning Phase**

**Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)**

|  |  |
| --- | --- |
| Date | 1st, April 2025 |
| Team ID | SWTID1742493942 |
| Project Name | Connectify - Social Media App |
| Maximum Marks | 2 Marks |

**5. PROJECT PLANNING & SCHEDULING**

* 1. **Project Planning**

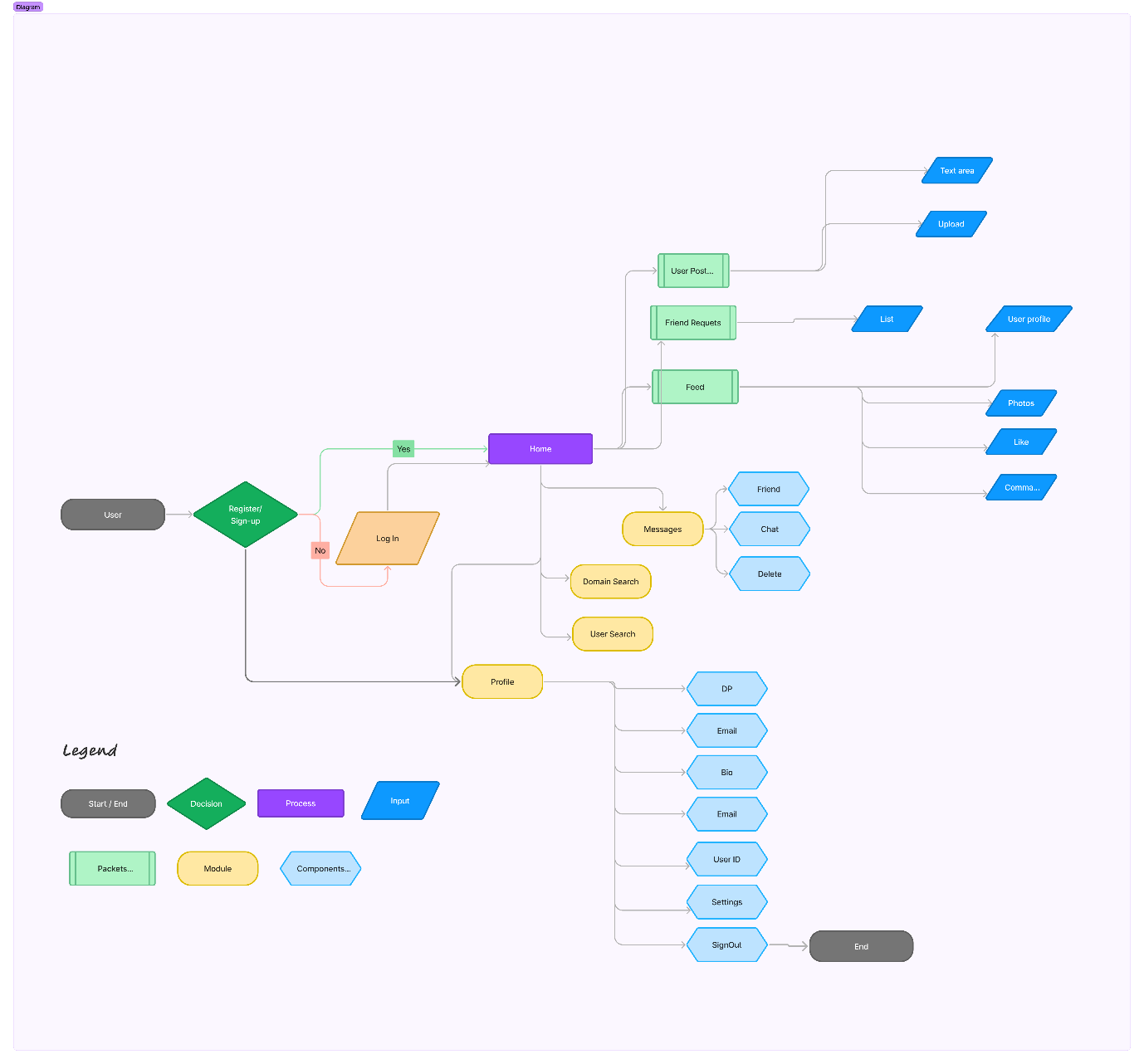
The development of the Social Media Web Application was planned by dividing the work into clear feature-based phases. The major focus areas included.

**Phases of Development**

* **Requirement Analysis**  
  • Identifying core features like user authentication, posting, liking, commenting, and reminders  
  • Defining the tech stack: MERN (MongoDB, Express.js, React.js, Node.js) with localStorage for storing reminders  
  • Planning basic access control and routing for protected components
* **UI/UX Design**  
  • Creating wireframes for pages like login/signup, feed, profile, and reminder panel  
  • Designing a responsive user interface for desktop and mobile using CSS Flexbox and Grid  
  • Focusing on user-friendly navigation, post layout, and interactive icons
* **Backend Development**  
  • Setting up MongoDB collections for users, posts, comments, and likes  
  • Developing RESTful APIs using Express.js to handle CRUD operations for user and post data  
  • Implementing JWT-based authentication for secure user sessions  
  • No cloud storage: file uploads handled in a basic local environment.
* **Frontend Development**  
  • Building reusable React components for login, register, feed, profile, reminder panel, and post creation  
  • Connecting the frontend to the backend using fetch to consume APIs  
  • Using localStorage to persist reminder data and maintain small local states  
  • Managing global state with Context API.
* **Testing & Debugging**  
  • Unit testing of backend routes and user authentication APIs using tools like Postman  
  • Testing React components and form validation manually  
  • Verifying responsiveness across browsers and devices  
  • Ensuring localStorage data is preserved after refresh
* **Deployment & Hosting**  
  • Deploying the frontend using Netlify or Vercel  
  • Hosting the backend server on Render or Railway  
  • Connecting frontend and backend using deployed API endpoints  
  • Configuring custom domain and enabling SSL/HTTPS
* **Maintenance & Feedback Integration**  
  • Regularly collecting user feedback to improve UX  
  • Adding enhancements like emoji support, dark mode, etc.  
  • Monitoring logs for any runtime issues and fixing bugs  
  • Optimizing performance, especially related to localStorage and feed rendering.

**Estimated Timeline:**

| Phase | Duration | Status |
| --- | --- | --- |
| Requirement Analysis | March31-April1 | Completed |
| UI/UX Design | April 2 -April 4 | Completed |
| Backend Development | April 5 – April 6 | Completed |
| Frontend Development | April 7 -April 9 | Completed |
| Testing & Debugging | April 10 – April 11 | Completed |
| Deployment | April 12 | Completed |
| Feedback & Maintenance | April 13 -April 14 | Completed |



**User Flow**