**PROJECT PHASE 1 PLANNING**

**1. Clarity of Purpose**

**Objective:**

* **Our application** is a **service quote platform** designed to help users find the best quotes for services like plumbing, electrical work, snow clearing, car repairs, etc.
* **Primary Problem**: It solves the problem of users searching for reliable service providers and getting competitive quotes by centralizing service offerings and quotes in one platform.
* **Main Function**: The platform allows users to post their service needs, and professionals can submit quotes. Users then select the best quote, confirming the deal directly with the provider. This removes the need for users to individually search for service providers, making it a more efficient process.

**Category Specification:**

* **SaaS (Software as a Service)**: **Our app** offers a utility for managing service needs and getting quotes. This falls under the SaaS category because it provides users a platform to manage tasks and obtain competitive service quotes.

**2. User System Design**

**User Interactions:**

* **User to Client Side (React app)**:
  + Users can visit the homepage, view categories of services, and create posts for their service needs.
  + On each service post page, users can input details about the service they require (e.g., plumbing, electrical work).
* **Client to Server Communications**:
  + The React app sends POST requests to the server to create new service posts and fetch quotes.
  + Server-side logic will handle creating the post in the database and sending it to relevant professionals.
* **Server to Database**:
  + The server interacts with the database to store and retrieve information like user profiles, service posts, quotes, and provider details.

**User Interaction Diagram (Conceptual Example):**

1. **Homepage**:
   * Users visit the homepage and see available service categories (plumbing, electrical, etc.).
   * Upon selecting a category, users are shown a list of service posts.
2. **Service Post Page**:
   * Users create a service post (enter details about their requirements).
   * The client app sends the post to the server, which updates the database.
3. **Quote Page**:
   * Users can browse quotes from professionals who responded to their service post.
   * The client app allows the user to select a quote and confirm the service deal.

**3. Feature Justification**

**Justification of Key Features:**

* **User Registration/Login**:
  + Ensures that users can track their service posts, quotes, and history in a secure and personalized manner. It allows for a tailored experience, such as viewing past service needs and preferred service providers.
* **Service Post Submission**:
  + The core feature of the platform, allowing users to list their needs for various services like plumbing or electrical work.
  + It helps users communicate their requirements clearly to service providers without the need to search around for individual providers.
* **Quote Submission by Professionals**:
  + Service providers can send quotes, helping users easily compare different options and select the best offer, saving time and effort in searching for reliable providers.
* **Selection and Confirmation of Deals**:
  + After reviewing quotes, users can confirm the deal with the service provider. This process streamlines the entire service hiring process, making it easier to settle on the right provider.

**4. Feasibility**

**Milestones for the Project:**

* **Phase 1 (Week 1-3)**:
  + Research and planning, finalize project scope and design.
  + Develop front-end (React app) with basic UI/UX and service posting functionality.
* **Phase 2 (Week 4-6)**:
  + Develop server-side logic (Node.js/Express), user authentication, and database setup (MongoDB).
  + Implement the ability to submit quotes and manage service posts.
* **Phase 3 (Week 7-9)**:
  + Testing, debugging, and integration of the quote submission and deal confirmation features.
  + Refine UI and user flow.
* **Phase 4 (Week 10-12)**:
  + Final testing, documentation, and preparing for deployment.
  + Deployment to production environment.

**Challenges and Limitations:**

* **Time Management**:
  + The backend and database setup might take more time than anticipated, especially when implementing user authentication and managing quotes.
  + We will prioritize milestones to ensure consistent progress.
* **Server-Side and Database Complexity**:
  + Handling dynamic relationships between users, service posts, and quotes may introduce complexity, but we plan to break the work into manageable chunks.

**References:**

* **YouTube Tutorials**:
  + "React for Beginners" (for front-end development)
  + "Building REST APIs with Node.js" (for server-side)
  + "MongoDB for Beginners" (for database setup)
* **Documentation**:
  + React Documentation
  + Node.js Documentation
  + [MongoDB Documentation](https://www.mongodb.com/docs/)