**Sample Solution: Task Management Tool**

**Project Description:**

We are designing a task management tool aimed at helping individuals and small teams organize their work more effectively. The tool will allow users to create tasks, assign deadlines, and track progress with visual indicators.

**Phase 1: Requirements Gathering:**

1. Users can create, edit, and delete tasks.

2. Tasks can have deadlines and priority levels.

3. Users can track task progress visually.

4. A dashboard shows an overview of all tasks and their statuses.

5. Data is saved securely to a cloud database for access from any device.

**Phase 2: System Design:**

**1. Architecture:**

- Frontend: Web app (React.js) for task creation and management.

- Backend: Node.js for handling API requests and task data.

- Database: MongoDB for storing task data.

**2**. **UI Wireframe:**

- A main dashboard showing tasks.

- A task creation form with fields for task name, deadline, and priority.

- A progress tracker using a visual indicator (e.g., a progress bar).

**Phase 3: Development Plan:**

1. Frontend Team: Build UI for task management using React.js.

2. Backend Team: Implement task creation and tracking APIs using Node.js.

3. Database Team: Set up MongoDB to store and retrieve task information.

4. Tech Stack:

- Frontend: React.js

- Backend: Node.js with Express

- Database: MongoDB

**Phase 4: Testing Plan:**

1. Unit Tests: Test the task creation form to ensure data is correctly stored.

2. Integration Tests: Ensure smooth communication between frontend, backend, and database.

3. User Testing: Test app usability and functionality with 5 users for 1 week.

**Phase 5: Maintenance Plan:**

1. Monthly feature updates based on user feedback.

2. Weekly bug fixes and performance optimization.

3. Monitor errors using a tool like Sentry for crash reporting.

This activity encourages hands-on experience with software engineering concepts, simulating real-world software development.