

# Kanban Board Project Week 2 Deliverable

## Kanban Board Project Week 2 Deliverable

Week 2 (August 12 - August 16)

Stage 2: Setup and Basic Implementation

Objective: Establish the development environment and start basic backend and frontend development.

---

### Backend Team:

#### August 12: Development Environment Setup

- **Task 1: Local Development Setup**
  - Set up the local development environment with Node.js and necessary packages.
  - Configure the development environment for MongoDB and Mongoose ORM.
  - Install Postman or similar API testing tools.
  - Test initial setup with a basic "Hello World" API endpoint.
- **Task 2: Cloud Environment Setup- (*Need to check with professor*)**
  - Set up AWS EC2 instances for the backend.
  - Configure AWS S3 for file storage, if needed.
  - Establish a connection between the EC2 instance and MongoDB.
  - Set up security groups and network access control for the EC2 instance.
- **Task 3: Dependency Management**
  - Install and configure backend dependencies (Express, Mongoose, JWT, etc.).
  - Set up environment variables securely using dotenv.

#### August 13: Version Control Setup

- **Task 1: Git Repository Initialization**
  - Initialize separate Git repositories for the backend and frontend codebases.
  - Set up a .gitignore file to exclude unnecessary files and directories.
  - Commit the initial codebase to the main branch.
- **Task 2: Branching Strategy**
  - Define and document the branching strategy (e.g., main, develop, feature branches).
  - Create develop and feature branches for ongoing development.
- **Task 3: CI/CD Pipeline Configuration**
  - Configure a basic CI/CD pipeline using GitLab CI/CD or GitHub Actions.
  - Automate testing and deployment for the backend API.
  - Set up automatic code linting and testing on every push to the develop branch.

## August 14: Backend Framework Setup

- **Task 1: Node.js Project Structure**
    - Set up a clean and modular project structure (e.g., routes, controllers, models, middleware).
    - Create folder structures for different layers like services, utilities, and configurations.
  - **Task 2: Initial API Endpoints**
    - Develop basic CRUD API endpoints for user authentication (register, login, logout).
    - Implement JWT-based authentication and token generation.
    - Develop initial API endpoints for creating and managing projects.
  - **Task 3: Testing API Endpoints**
    - Use Postman or similar tools to test the newly created API endpoints.
    - Validate API responses and error handling mechanisms.
- 

## Frontend Team:

### August 12: Development Environment Setup

- **Task 1: Local Development Setup**
  - Set up the local development environment with React.js and Node.js.
  - Install necessary frontend dependencies (React Router, Redux, etc.).
  - Set up a basic project structure for the React application.
- **Task 2: Cloud Environment Setup- (*Need to check with professor*)**
  - Configure an AWS S3 bucket for static asset hosting, if necessary.
  - Set up AWS CloudFront for content delivery.

### August 13: Version Control Setup

- **Task 1: Git Repository Initialization**
  - Initialize the frontend Git repository.
  - Commit the initial codebase to the main branch.
- **Task 2: Branching Strategy**
  - Follow the branching strategy defined by the backend team.
  - Ensure consistent branch naming conventions across the project.
- **Task 3: CI/CD Pipeline Configuration**
  - Configure a CI/CD pipeline for the frontend application.
  - Set up automated testing, build, and deployment processes.

### August 15-16: Frontend Framework Setup

- **Task 1: React Project Structure**
  - Establish a modular project structure (e.g., components, pages, services).

- Set up routing for different user roles (Admin, Project Manager, Volunteer).
  - **Task 2: Implement Initial Components**
    - Develop the Login component with form validation and integration with the authentication API.
    - Create the Dashboard component with basic navigation and a placeholder for project lists.
    - Implement initial Redux state management for handling user authentication and project data.
  - **Task 3: Connect Frontend to Backend**
    - Integrate frontend components with the backend API for authentication.
    - Fetch and display project data in the Dashboard component.
    - Ensure proper error handling and user feedback mechanisms.
- 

## **UI/UX Team:**

### **August 12: Design Refinement and Iteration**

- **Task 1: Design System Setup**
  - Finalize a design system, including color schemes, typography, and UI components.
  - Create reusable design components for buttons, forms, and modals.
- **Task 2: Design Feedback Loop**
  - Gather feedback from the backend and frontend teams on design feasibility.
  - Iterate on wireframes and mockups based on team input.

### **August 15-16: UI/UX Implementation Support**

- **Task 1: Collaborate with the Frontend Team**
    - Assist the frontend team in implementing the design system within the React project.
    - Ensure that UI components are consistent with the mockups and design guidelines.
  - **Task 2: Usability Testing**
    - Conduct initial usability testing on the Login and Dashboard components.
    - Provide suggestions for improving the user experience based on testing results.
-