**PROJECT OVERVIEW**

The project involves creating an Agile Board application with functionalities for managing tasks and users using design patterns like factories and builders. It includes components for data modeling, input/output handling, and utility functions.

**CONTRIBUTIONS**

#### **Pruthvi Niranjan: Backend and Agile Board Logic Developer**

* **Primary Focus:** Development and refinement of the Agile Board's core logic and architecture.
* **Contributions:**
  + **AgileBoard.java:**
    - Designed and implemented methods for managing tasks and user interactions within the board.
    - Integrated task status updates and board display logic.
    - Ensured the board's scalability and flexibility to handle various project sizes.
  + **AgileBoardBuilder.java:**
    - Developed the builder pattern to facilitate the construction of Agile Board instances.
    - Implemented a fluent interface for setting board parameters such as users and initial tasks.
    - Collaborated with other members to ensure seamless integration with the board.

#### **Abhay Deshpande: Factory and Design Pattern Specialist**

* **Primary Focus:** Efficient creation and management of objects using factory patterns.
* **Contributions:**
  + **TaskFactory.java:**
    - Implemented factory methods for creating task objects with different statuses and priorities.
    - Collaborated with the backend developer to integrate factory outputs with the Agile Board logic.
  + **AccountFactory.java:**
    - Developed methods to instantiate various user roles, such as team leads and team members.
    - Ensured that the factory pattern was consistently applied across account creation processes.
  + **Design Patterns:**
    - Provided guidance and documentation on the use of design patterns within the project.
    - Reviewed code for adherence to design principles and suggested improvements.

#### **Anuja Patil : Model and Data Management**

* **Primary Focus:** Data representation and management of user accounts and tasks.
* **Contributions:**
  + **Account.java and Role.java:**
    - Defined data structures for user accounts and roles within the system.
    - Implemented methods for role-based access control and permissions.
    - Ensured data validation and integrity across account objects.
  + **Task.java and TaskStatus.java:**
    - Developed a comprehensive model for task representation, including attributes like title, description, and status.
    - Created enumerations for task statuses to streamline status management and updates.
    - Coordinated with the factory specialist to align task creation with data models.

#### **Ajith Patil: UI/Controller and Utility Developer**

* **Primary Focus:** User interaction interfaces and utility functions for enhanced functionality.
* **Contributions:**
  + **Controller.java:**
    - Developed the main application controller to manage user inputs and command execution.
    - Implemented methods for handling user interactions with the console interface.
    - Coordinated with backend logic to ensure proper command execution and feedback.
  + **ConsoleIO.java and ConsoleError.java:**
    - Designed a facade for console input and output operations to simplify user interactions.
    - Implemented error handling mechanisms to provide meaningful feedback to users.
  + **ArrayUtils.java:**
    - Developed utility functions for array operations, optimizing performance and reducing redundancy.
    - Documented utility functions for easy integration and usage across different modules.