Biweekly Report

Name: Jingxiao Han

Period: April 7 - April 20, 2025

Project: Smart Maintenance Platform for Aero Engine Industrial Equipment

Week 3: Development Environment Setup and Test Planning (April 7-13, 2025)

During this week, I focused on establishing the development infrastructure and creating comprehensive test plans for our predictive maintenance platform. My specific contributions included:

1. **Development Environment Setup (April 4-7)**
    - Configured and deployed the complete development environment for the team
    - Created CI/CD pipelines for automated testing and deployment
    - Documented the environment setup process for team reference
2. **Test Plan Development (April 7-11)**
    - Drafted comprehensive test strategy document
    - Created detailed test plans for all system modules
    - Designed integration testing approach

Time spent: 16 hours

Week 4: Interface Design and Backend Development Preparation (April 14-20, 2025)

This week was dedicated to designing user interfaces and preparing for backend development:

1. **Interface Design (April 17-20)**
    - Designed user interface mockups for core system modules
    - Established UI component library and design system
    - Designed responsive layouts for multi-device support
2. **Backend Service Planning (April 14-16)**
    - Defined API specifications and endpoints
    - Planned database schema and relationships
    - Established backend architecture patterns

Time spent: 14 hours

Completed WBS Items

- **2.2 Test Plan Development** (WBS Item, 12 hours) - 100% Complete
- **4.1 Development Environment Setup** (WBS Item, 4 hours) - 100% Complete
- **3.3 Interface Design** (WBS Item, 24 hours) - 70% Complete

Challenges & Solutions

The main challenge I encountered was establishing a development environment that supports both web application development and machine learning model integration. This required balancing different technology stacks and ensuring compatibility across components. I addressed this by:

1. Creating a modular architecture with clear separation between frontend, backend, and ML components
2. Implementing containerization to isolate dependencies and ensure consistency
3. Establishing well-defined APIs for communication between different system components

Next Steps

1. Complete the remaining interface designs for report generation and data simulation
2. Begin implementation of Non-Core Function Development with Unit Test
3. Collaborate with Xuanhe Yang on integrating ML models with the frontend

Total Hours Worked

Total hours for this reporting period: 30 hours