

## Joint Report: Group 10

Agile development is an iterative process that places an emphasis on collaboration and regular communication, in addition to dynamic and effective teamwork. Our project was the design and creation of a java-based UWE accommodation system program with a GUI interface. Achieving a project of such scope required the team to implement a variety of Agile principles and methodologies to achieve our objectives, such as effective team communication, SOLID principles and sprint cycles.

A core component of Agile Development is effective communication. With this in mind, the team endeavoured to prioritise this element throughout the entire project, allowing us to exchange ideas and information, maintain our project schedule and support each other with producing work to the best of our abilities. In addition to implementing Agile development methodologies, one of the team's main goals was to develop a project that adheres to the SOLID principles of software design. One reason for this is because our program may need to be extended and modified by collaborative teams in the future, which requires a codebase that is easy to understand, flexible and extensible (*Krishna, 2023*).

One way we have achieved this goal is by creating classes that implement the Single Responsibility Principle. This means that each class in our program has only one responsibility and one reason to change, making them easier to maintain and modify. A second way the team has followed the SOLID principles is evidenced by our UML class diagram, which highlights our implementation of the Open-Closed Principle. This is achieved by structuring the program so that derived classes create objects inherited from the base classes. For example, the class 'newLease' obtains its properties from 'createLease'.

## Joint Report: Group 10

Alongside effective communication and SOLID principles, another key element of Agile development is a well-considered product backlog, which makes iteration planning easier and allows the entire team to meet expectations. Whilst producing the UWE accommodation system, our team followed the below backlog;

### **Weeks 1-2: Team Setup and Project Kickoff**

- Identify team members' strengths and assign initial roles.
- Hold a kickoff meeting to discuss deliverables and roadmap.

### **Weeks 3-4: Use Case Diagram**

- Gather requirements for the system.
- Collaboratively create a UML Use Case Diagram.

### **Weeks 5-6: Class Diagram**

- Develop a detailed UML Class Diagram based on the Use Case Diagram.
- Identify key relationships and attributes.

### **Weeks 7-8: Sequence Diagram**

- Select two critical use cases from the Use Case Diagram.
- Develop a UML Sequence Diagram based on the use cases.

### **Weeks 9-10: Joint Report**

- Write a joint report on Agile development.
- Include sections on team communication, project backlogs, and SOLID principles.

### **Weeks 11-12: Implementation**

- Implement the business logic.
- Design and develop the JavaFX GUI, following the Class Diagram.

## Joint Report: Group 10

### **Weeks 13-14: Testing**

- Develop a test strategy for the system.
- Create test cases for numerous use cases and additional validation.
- Rigorously test the system, including incorrect data scenarios.

### **Weeks 15-16: Individual Report and Joint Report**

- Document individual contributions to the project.

Building directly upon our project backlog, the team also utilised sprint cycles to achieve the required amount of work for each product addition. A summary of our associated sprint cycles is below;

### **SPRINT 1 (Weeks 3-4): Use Case Diagram**

- Task: Collaboratively create a UML Use Case Diagram.
- Goal: Complete the Use Case Diagram deliverable.

### **SPRINT 2 (Weeks 5-6): Class Diagram**

- Task: Develop a detailed UML Class Diagram based on the Use Case Diagram.
- Goal: Establish a solid foundation for the system's structure.

### **SPRINT 3-4 (Weeks 7-8): Sequence Diagrams**

- Task: Develop UML Sequence Diagrams for two critical use cases.
- Goal: Ensure a clear understanding of the system's interactions.

### **SPRINT 5-6 (Weeks 9-10): Joint Report**

- Task: Write a joint report on Agile development.

## Joint Report: Group 10

- Goal: Communicate the team's approach to Agile development.

### **SPRINT 7-8 (Weeks 11-12): Implementation**

- Task: Implement the business logic and Java classes.
- Task: Design and develop the JavaFX GUI.
- Goal: Have a functional prototype ready for testing.

### **SPRINT 9-10 (Weeks 13-14): Testing**

- Task: Develop a test strategy and test use cases.
- Goal: Ensure the system meets the specified requirements and is robust.

### **SPRINT 11-12 (Weeks 15-16): Individual Report**

- Task: Document individual contributions.
- Task: Write a reflective report on skills and knowledge.
- Goal: Summarise the project and reflect on the development process.

The principles and tools that Agile provides were instrumental in the production of our UWE accommodation system. By utilising core elements such as continuous iterations of thoroughly communicated collaborative work, adherence to SOLID principles of software design and a well-defined project backlog, our team was able to continuously develop an agile software project and correctly achieve our objectives.

References:

## Joint Report: Group 10

- Krishna, A. (2023). *What Is SOLID? Principles for Better Software Design*. [online] freeCodeCamp. Available at:  
<https://www.freecodecamp.org/news/solid-principles-for-better-software-design/>  
[Accessed 18 Mar. 2024].