**Software Engineering**

Reflection:

***1. Introduction***

Our project focused on developing a **newspaper delivery system** to help a newsagent manage their operations, including organising deliveries, tracking inventory, handling customer schedules.

We chose the **Agile methodology**, specifically the **Scrum framework**, because it allowed us to break the project into manageable sprints, adapt to changes, and address challenges collaboratively. This approach ensured we could meet the requirements while improving teamwork and communication.

This assignment highlighted the importance of designing and implementing software in a structured and collaborative manner, especially for non-technical users like newsagents.

***2. Team Dynamics and Roles***

Our team consisted of **Padraig, Ben, Michael, and Liam**, with each member taking on specific responsibilities based on their strengths.

* **Ben**: Focused on SQL setup, ER diagrams, and the Publications class.
* **Liam**: Created completed Deliveries and Invoice as well as the tests for both.
* **Michael**: Created complete Order class with test design and Junit tests. Created UML diagrams. Created group repository.
* **Padraig**: Created complete Customer class with test designs and Junit tests.

Although the team had a slow start, we quickly adapted to Agile practices and improved our collaboration. Responsibilities were distributed by class, with each member developing, testing, and integrating their components. This structure allowed us to work independently while aligning our efforts during sprints.

***3. Sprint/Iteration Summary***

* The work was divided across Four sprints with each consisting of Two Weeks. We updated the work consistently through GitHub, assigning and marking Tasks.
* Sprint 1: We created User stories, acceptance criteria and test design for each entity in the project, started the designing and setting up the project (classes, Sql, etc.)
* Sprint 2: We Implemented the Create Feature for each entity along with implementing Exception handling.
* Sprint 3: We Implemented The Read and Delete Features. Some basic Tidying and optimization.
* Sprint 4: We Implemented the Update Feature, Updated JUnits and tested them against the project. Finalized the Project DEMO.

***4. Agile Practices and Tools***

Several Agile practices and tools contributed to the project’s success:

* **Weekly Stand-ups**: Conducted during lab sessions to discuss progress, blockers, and upcoming tasks.
* **Sprint Planning**: Tasks were divided based on expertise and tracked using a shared task plan.
* **Version Control**: GitHub was used to manage code changes, track progress, and resolve conflicts.
* **Discord**: Used for asynchronous communication and quick issue resolution outside lab sessions.

These practices ensured transparency, adaptability, and effective collaboration throughout the project.

***5. Successes and Strengths***

* Detail areas where the team excelled, such as:
* Motivation and drive to get each part done in a timely manner.
* Task Management, we always made sure we understood which part each of us should be working on, on a weekly basis.
* Weekly meetups, and frequent communication between the group.
* Making sure all scripts were able to work well together and commented.

***6. Challenges Faced***

* We face some challenges working on this project, for example:
* Estimating how long each task should take.
* We had difficulties staying within the scope of the project.
* We had issues with some parts of the code where we had to rework our scripts to be connected to one menu script.

***7. Lessons Learned***

* Our time estimations is something we will improve over time, and is not stressed over.
* We learned the importance of well-developed user stories and how it can affect the entire project and flow of work.
* Doing this project helped us reevaluate the way we will approach similar projects in the future.
* We have learned more about pacing ourselves and taking each step slowly to make sure it is completed before moving to the next step.
* We discovered the level of importance when it comes to reviewing other members’ code and giving feedback.

***8. Conclusion***

* Summarize the project outcomes, including how well the final system met the requirements.
* Reflect on the overall effectiveness of the Agile methodology and what your group might do differently in future projects.
* Optionally, acknowledge the effort and contributions of the entire team.