

Final Project

As the culmination of our Agile Development course, your team of five will complete a final project to be presented on the last day of class, Thursday 30 January at 7p.

Description

- The content and theme of your project are completely open-ended, giving your team creative freedom. Just make sure you follow the below guidelines.
- Your project must be a collaborative effort among your team.
- You will be creating a console application in C# using Rider, xUnit, Git, and Trello.
- Your program will take input at the command line via the args in your

```
1 public static void Main(string[] args)
```

 **See "How To Pass In Command Line Arguments" for details on how command line arguments work**

- Your program can output either via the console (Console.WriteLine()). In other words, your application does NOT need a user interface.
 - Each team member must take on each role (Scrum Master, Developer).
 - Jon and Tom will be the Product Owners (who tell you to do whatever you want)
 - Rotate the roles of Scrum Master through your sprints.
 - You are all responsible for the quality of your code and product, and will not have an explicit QA team.
 - Each team member MUST implement code and unit tests and refactor.
 - You are all developers and writing code, and that code includes unit tests.
 - Your team's work will be evaluated based on how well you incorporate Agile practices, teamwork, and technical implementation.
 - We will be working on the project all class on Tuesday 28 January and Wednesday 29 January.
-

Requirements

The project requirements are expressed below as user stories (because - on theme):

1. Create A Simple Spec Document Describing the Features

- As a team, we want to write a specification for our project so that we can define its scope, purpose, and high-level features.
- There is no format for the spec doc, just clearly describe the project that you are building out.

2. Write User Stories

- As a team, we want to break down the project specification into actionable user stories so that we can clearly understand what needs to be implemented.

3. Create Your Backlog in Trello

- As a team, we want to organize our user stories into a backlog in Trello so that we can track progress and prioritize tasks.

4. Size Your User Stories

- As a team, we want to assign sizes to our user stories (e.g., Fibonacci points) so that we can estimate the effort required to complete each task.

5. Refine Your Backlog

- As a team, we want to regularly refine our backlog in Trello so that it remains up-to-date and accurately reflects our project's priorities.

6. Sprint Planning and Execution

- As a team, we want to plan and execute a series of sprints, including holding standups and sprint retrospectives, so that we can iteratively implement our project.

7. Test-Driven Development

- As developers, we want to write unit tests before implementing the corresponding functionality so that we can ensure our code meets the requirements and is reliable.

8. Refactoring

- As developers, we want to refactor our code during development so that it remains clean, efficient, and easy to maintain.

9. Version Control with GitHub

- As a team, we want to use GitHub for version control, including working in our own branches and merging back into main, so that we can collaborate effectively and manage our codebase.
- Use pull requests to merge your code into main.

10. Running the Project

- You should be able to run the project by cloning the repository, going into the project directory, and then using `dotnet run` in a console.
- The code must compile without requiring any additional steps to setup

Deliverables

Zip up and turn in on Canvas:

1. Project specification document.
2. Snapshots of your Trello board (screencaps work), showing:
 1. Your initial backlog including sizing
 2. Before and after for each sprint
3. A text file with the link to your GitHub repository with your code and unit tests.
 - Your git history and pull requests will show how your application was developed, so write good comments and descriptions!


Presentation

On the day of the final (Thursday 30 January), each group will present their projects. You may choose to present however you like. A Powerpoint, a PDF, a series of dance numbers.

Your presentation should follow this basic structure:

1. Tell us what you made and why you made it
2. Demonstrate your project (connect to the projection screen in the classroom)
3. A brief explanation of exactly what each group member contributed
4. What you learned during the project
5. What challenges you struggled with
6. What you'd do differently in the future

Be prepared to take questions from both the instructors and other students.

 **We're aiming for each presentation to be 10 minutes or less, including questions.**