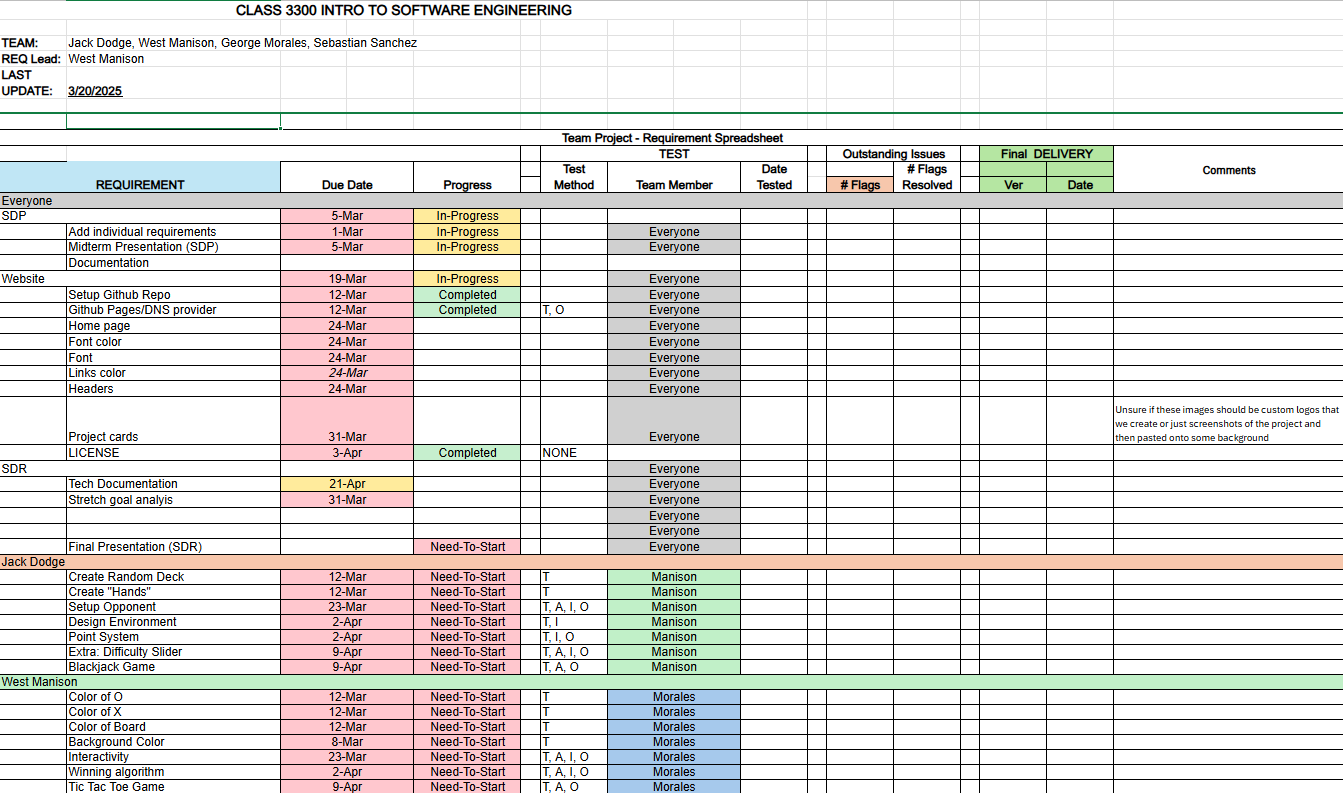
Software Development Schedule

CS 3300-002 Spring 2025

Jack Dodge, West Manison, George Morales, Sebastian Sanchez

# Initial Plan



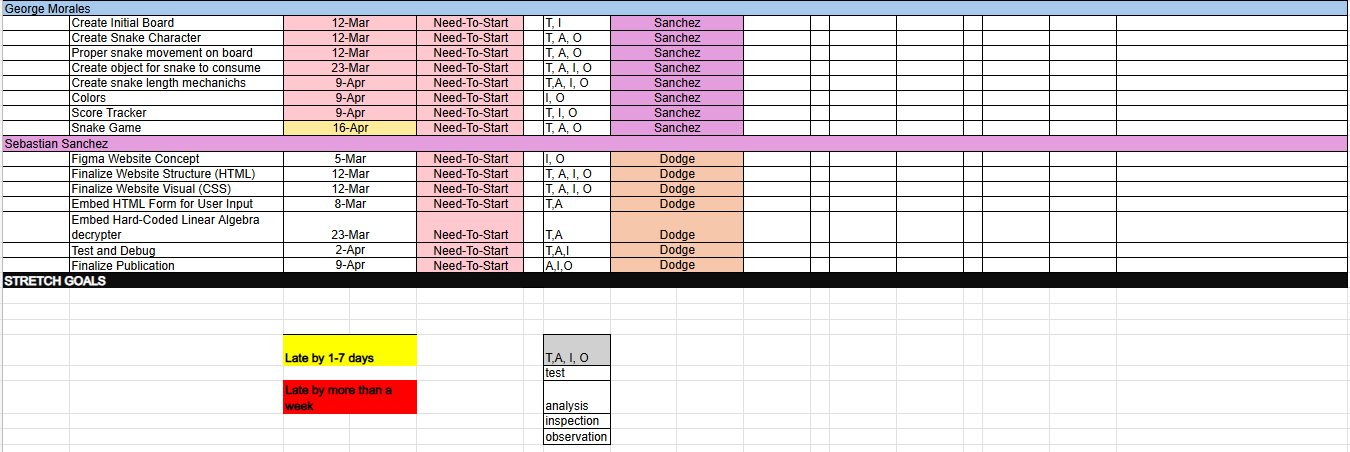
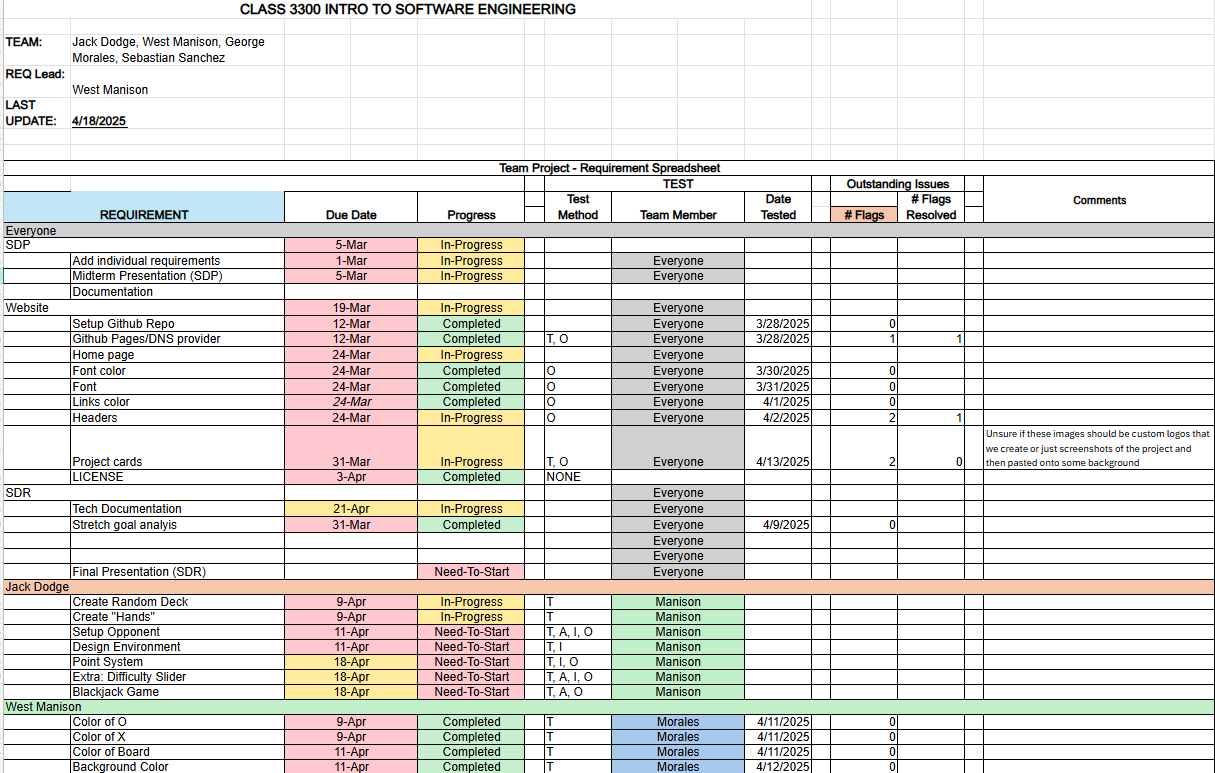
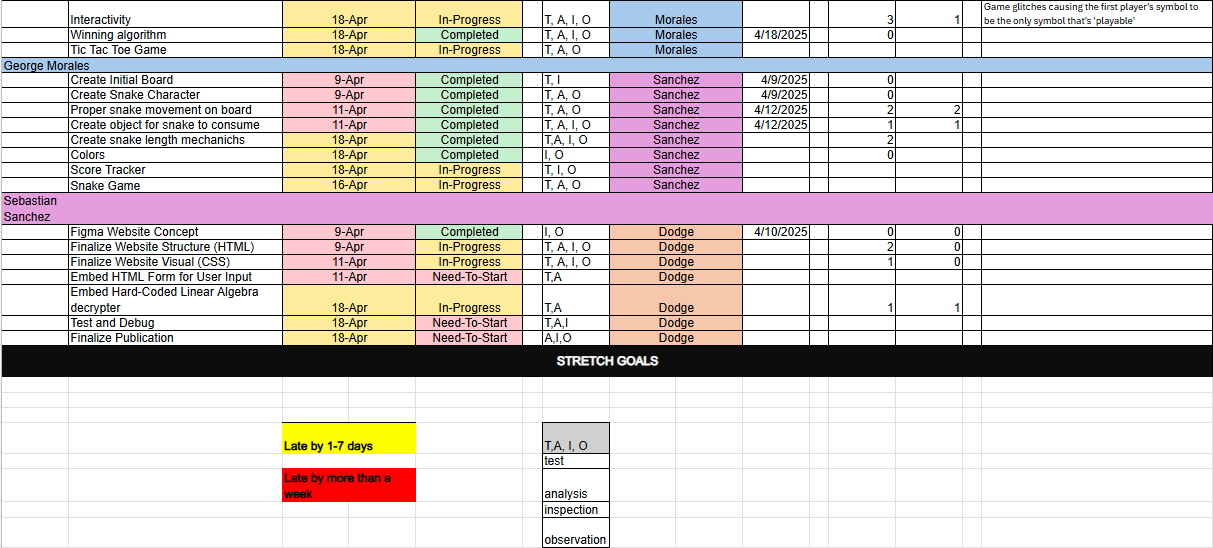
We filled out what we thought to be an extensive list of possible tests/requirements that we would need to complete. We split up our requirements by team members to ensure everyone could quickly see who and what needed the most support; although the team members’ requirements are kept separately, we worked together to come up with a schedule that would work for everyone, trying our best to sync up when work was done.

Figure 1 Initial Requirements Spreadsheet

Since we are developing with Test Driven Development as our primary strategy, it meant sense to us to have our software **and** testing requirements in the same document. This allowed for easy understanding of what tests needed to be done for each piece of software and allowed us to quickly observe at what each team member needed to work on (whether that was developing code or writing the tests for the code to be developed/refactored).

# Continuously Maintained





Above is a snapshot of our requirements spreadsheet two weeks after spring break. The maintaining was the hardest part of the spreadsheet. Ensuring that everyone was going in and editing as they tested and developed was challenging.

# As It Actually Happened

Above is the final requirements spreadsheet. As you can see some things were never updated but for the most part we were able to use this document as a way to keep track of who was doing what and when they were accomplishing it. It was helpful in tracking our progress but also showed our progression away from TDD and into a more agile style of development. We began to build things in mini sprints, asking for testing after code had been developed; this adjusted the way we could’ve used our spreadsheet so we might’ve been limited by the combining our testing and requirements sheet into one.