Sprint 1 Artifacts:

1. Set Up GitHub Repository
    a. Created a GitHub for Project 3. Invited all of our team members to be collaborators. Added folders to be organized.
2. Set Up Godot
    a. Created an empty Godot project to allow programming to start. Followed a tutorial to correctly initialize it as a 2D project, as our game will be in 2D, not 3D.
3. Integrate Git with Godot
    a. Committed the Godot project to the Git repository so all programmers will use the same project. Ensured every programmer is able to open the Godot project on their machine.
4. Initial Architecture Document
    a. Created the Internstellar Architecture Document to outline the game's structure, systems, and designs
        i. Overview: defined the game as a 2D point-and-click game built with Godot. It will contain scenes, puzzles, and bubble dialogues.
            1. Point-and-click aspect: detecting objects, triggering interactions (inventory, speech bubbles, minigames).
            2. Map navigation aspect: explain how the player navigates between scenes.
            3. Hint system aspect: define logic for detecting incomplete tasks and guiding players using object highlighting or speech bubble hints.
            4. Game story: describe the setting, the intern, and the puzzles.
5. Make at least 3 Architecture Document Diagrams
    a. Created a UML Architecture diagram for Point and Click Functionality. Outlines the high-level functionality and rules that govern how to interact with each type of object.
    b. Created 2 UML Architecture diagrams for Map Navigation. Outlines the high-level functionality and rules that govern how to use the map to navigate through the world.
    c. Created a UML Architecture diagram for Hint Functionality. Outlines the high-level functionality and rules that govern how the hint system will work. This includes how to handle giving hints for multistep problems.
6. Sprint 1 Requirements
    a. Outline deliverables for Sprint 1 in the current Excel sheet based on feedback from the first submission.
7. Wrote the high level story.
    a. Decided to go down the intern route, where the game is focused on a NASA intern's first day on the job.

    b. Brainstormed ideas for the game. Kept a list and voted as a team on which ideas we wanted to keep and which ones we wanted to scrap.
    c. Decided on the following idea:
        i. A NASA team has been researching this UFO in space, when one day they are abducted by aliens. On that same day, an intern was supposed to start working for that team. This intern woke up late, and when they show up, they realize their team is gone. The intern realizes that their team is the only one that can stop the world from being taken over by aliens, so this intern must go after their team by solving various puzzles in order to save the human race.
8. Updated Agile Reference Stories spreadsheet
    a. Update the Agile Reference Stories spreadsheet based on the new developments. We updated the descriptions to match the required word counts. We initially lost a couple of points from this on our first submission.
9. Updated Requirements Stack spreadsheet
    a. Update the Requirements Stack spreadsheet based on the feedback from the sprint 1 submission.
    b. reorganized the spreadsheet to better reflect the logical progression of development. Each requirement was prioritized according to importance and feasibility for the next sprint
10. Sprint 1: GitHub link for Sprint Release
    a. Submitted the GitHub link to Canvas for grading. Ensured all deliverables were ready.
11. Keep GitHub up to date for the sprint 1 submission
    a. Organized the GitHub for timeline clarity for the final sprint submission. Ensured that code, documents, and diagrams are separated from each other to ensure maximum readability.
12. Get familiar with Godot and decide which language to use
    a. Understand Godot's specific Python-like language for development. Watched tutorials and read Godot docs.
13. Sprint 2: Updated Requirements Stack spreadsheet
    a. Updated the spreadsheet for the new sprint. After sprint 1, we got a better idea of how our project was progressing. We updated Sprint 2 requirements in the requirements spreadsheet to reflect these changes.
14. Sprint 2: Next Sprint Requirements Artifacts
    a. Outlined artifacts for the next sprint. We planned ahead by determining what deliverables we want to provide for the sprint 2 release. We documented these updates in our Sprint Artifacts document.