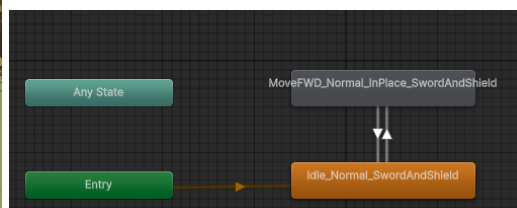
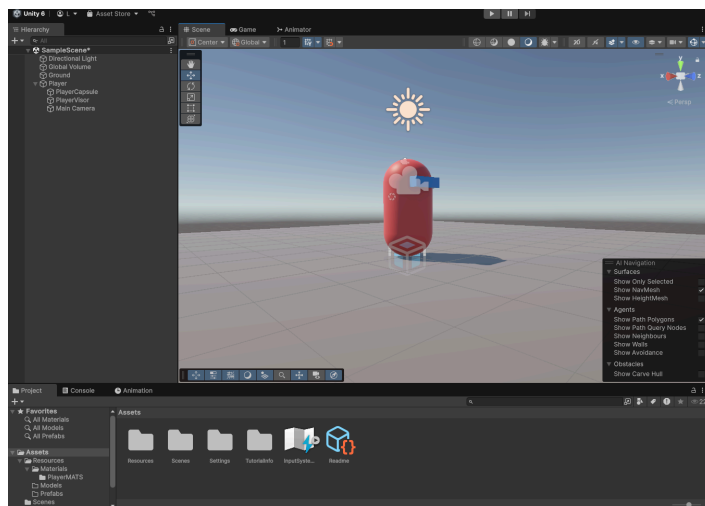


Lucas Ryan Stanczyk Brito – Unity Programmer Task Report

02/06/2025

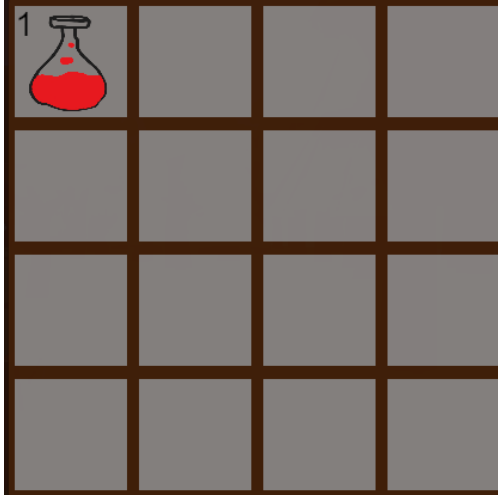
For this assignment, I developed a 3D prototype using Unity 6000.0.35f1 with low-poly assets for style consistency and rapid iteration. My focus was to implement all the required gameplay mechanics, UI systems, and persistence features as requested in the task document.

I began by setting up the core project structure, creating base scripts for character movement and animations. I chose a third-person setup to best demonstrate spatial interaction and camera control. I then implemented the slot-based inventory system using ScriptableObjects to manage item definitions, allowing for modularity and scalability.



Next, I created an interactive world where the player can pick up items using key inputs and store them in the inventory. Items can be dragged, swapped between slots, and used directly by clicking. Health potions, for example, can be consumed to trigger functionality. The system also supports equipping weapons via a dedicated equipment slot, which updates the weapon model on the player.

For UI, I focused on clarity and responsiveness. The inventory displays real-time updates, and hovering over items shows their description with icons and details. A delete zone allows players to discard items by dragging them out of the inventory. The ESC key opens the main menu, and 'I' opens the backpack.



I implemented a persistent save/load system using `PlayerPrefs`, which serializes the inventory state and equipped items. The state is restored automatically on startup.

Depois disso foquei em refinar mecanicas melhorar a UX deixando uma experiencia mais guiada.

Overall, I enjoyed the challenge and iterated quickly. One issue I encountered was with the drag-and-drop behavior initially disrupting layout, which I resolved by switching to a floating icon system. If I had more time, I would polish the visuals further and include more item types. I believe this prototype demonstrates my ability to quickly implement clean, scalable gameplay systems and UI logic within a short deadline.