Artifact 3 is one of the first Unreal C++ projects I made, and the first proper C++ game I made. In this game, the player must find their way to the end of the maze before any of the AIs. From the player’s perspective it may seem like the AIs are similarly trying to figure their way through the maze, however the AIs are given a list of target point actors at the start of the game and they randomly visit each point. They know exactly where the goal is and how to get there from the start, they are just unlikely to choose to go there. It was developed during my GAM-312 class. Throughout the class we would complete tutorials and then take that information and fold it into our projects. We were required to include certain elements like the player controller, AI controller, and pickup actor, but beyond that we mostly had free rein on our projects.

I chose to include this artifact because I felt I needed at least one C++ project in my ePortfolio, and I am genuinely proud of how this artifact came out. I particularly like my design for the AI because it is generalized enough to serve many possible purposes. The AI works by first looking at all actors in the level for a tag that can be set on each instance of the AI pawn. From there the logic will either work exclusively with the subset of actors with the unique identifying tag, or with all target point actors in the level if no unique subset is specified. While it is designed to move about randomly in this artifact, with a few minor tweaks it can be made into a patrolling NPC, going from point to point in a specific order.

When I made this artifact, I learned how to implement some fundamental elements of Unreal C++ projects, including player controllers, AI controllers, and interactable/pickup objects. I also figured out a clever workaround for adding a maze to an Unreal project because when I initially created the artifact, I did not figure out how to make a maze via an algorithm. Instead I imported a 2D image into Blender, extruded it into 3D, and then imported it into Unreal as an object. Figuring out how to make a random maze was a challenge that I sidestepped when I initially created the artifact, but I gave it another shot when I refined the artifact. Instead of spending my time trying to figure out how to do it by myself, I found a relevant tutorial. I did need to adapt the code from the tutorial because the maze in the tutorial is designed to pop into existence one cell at a time, but I wanted it to appear instantly in its completed state. Adding this random maze generator greatly improved the artifact by making the game more dynamic.