The goal of my project is to improve the state of AI in Massively Multiplayer Online Role Playing Games (MMORPGs, or just MMOs). Currently, most AI in MMOs could be described in pseudo pseudocode as follows: “Stand still or walk along a predetermined path until a Player Character (PC) gets within a certain range. At that point, check the PCs faction/reputation to determine if you want to attack. If you do, check your level against theirs to determine if you should attack. If so, run at the PC and attack until either you or the PC is dead.” There are a few extra complexities that can be introduced, such as the NPC running away once it is low on health, but the general algorithm is very simple, and in my opinion, very unsatisfying. This simplistic AI sets off the following chain of consequences: since the AI is so simple and repetitive, players can kill hundreds of NPCs an hour. Since game developers and players don't want the game to be finished in only a few days, players must kill many thousands of these dumb NPCs to reach the 'end content'. This has led to the gameplay of MMOs being described as 'grinding'. Therefore, one of the motivations of this project is to make MMOs more fun.

A second motivation for this project is as a research environment for cognitive science and artificial intelligence in general. While there have been great advances with respect to algorithms used for learning, planning, and knowledge representation, they have typically been used for very specialized domains instead of being combined and integrated into a generally behaving agent (for example, during my psychology degree I developed a neural network for predicting the outcomes of NHL games). One of the reasons for this is the lack of suitable environments for AI agents to interact with and behave in. There are essentially only two options: developing programs that behave in the physical world (robotics), or developing programs that behave in a complex virtual environment (e.g. A video game). While I believe that robotics holds great promise, it is currently very expensive even for hardware that is laughably rudimentary by biological standards, and perhaps more importantly, it forces the researcher to solve many extraordinarily difficult problems involving perception and motor control before the agent can even accomplish the most basic tasks. Using a virtual environment, on the other hand, allows the researcher to bypass those problems and immediately begin developing and testing algorithms focused on behaviour and cognition.

The MMO that I have chosen as a research vehicle is EverQuest, for several reasons. First, an open source server emulator exists that has been in use for over a decade. Second, I already own the game client and have experience with playing it. And lastly, since the game was released in 1999, the resources required to run the server and client are much less than a more modern game, freeing up more computational cycles for use on AI algorithms. I plan on starting with a stripped down version of the original game, by using only animals as NPCs (as opposed to including 'smarter' races such as goblins and orcs, which can use magic and items). The first (and possibly only, depending on how much I can accomplish) objective will be to emulate a rule of animal behaviour found in nature: never ignore a person. The animal may flee, or freeze, or attack, depending on the species and circumstances, but an animal never simply stands there and allows a person to run up to it (the Dodo is an unfortunate counterexample, but also an evolutionary anomaly!). The current state of MMO AI, where if an NPC chooses not to 'aggro' it will remain standing or walking as if the PC isn't there, is highly unrealistic and easily defeated.

Although I know you would prefer if we worked in groups of two, I was hoping to work on my own if possible. I only have two courses this semester, so I believe I can handle the extra workload, and I think it would be unlikely that someone else in the class possesses one of the few EverQuest game clients that are compatible with the server emulator.