**CS 4850 Form2 for Phase 1: Project Plan 10 points (out of 100 for semester)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project Name:** | | | *Name of project: Fuzzy Logic AI Chess Game* | | |
| **Team Members:** | | | *List All:* ***5-7 team members!***  *Paul Neamtu*  *Ryan Long*  *Tim Youngblood*  *Lydia Vaine*  *Souleymane Camara* | | |
| **Description:** | | | *Expanded description of project, what it does, who it serves, how it should operate, what inputs do what*  *This project will be a Chess game variant using Unity with fuzzy logic and working AI opponents. Players will start off with the main GUI that has several options to choose from. These options include Start Game, Quit, Instructions, Credits, and Toggle Sound. If player chooses Start Game whoever is chosen to go first with start. Player can move with the mouse where the chess piece is legal to move via rules. WASD can be used to rotate the camera view. Esc key can be used to bring up a pause menu. The game would continue until either player or AI wins the match and the player is showed an end screen of whether to restart, go to main menu, show move counter, or quit.* | | |
| **Development Phase** | | | **Description of Project/Prototype to this Point**  **Status of each Component** | **Deliverables** | |
| **Deliverable** | **Grade Weight** |
| **1.** | **Sprint 1:** | Creating the chess board and temporary models as place holders for chess pieces.  Researching fuzzy logic and getting used to using it. Creating an early GUI menu set up.  Having basic movement scripts using inheritance.  Start AI scripting.  Create game manager.  Implement Dye  Starting majority of the classes  Implement a rotating camera  Movement of the pieces will use the mouse pressing on a location.  WASD keys for moving the camera around the board with using C to snap the camera to original location | Project Presentation  Ppt to D2L  Prototype 1 | 20 points |
| **Challenges/Risk:**  **Getting used to using Unity engine, github, Trello, and Fuzzy Logic. Using Github for Unity version control could resort to glitches and coding errors. Getting accustomed using C# for those who may have not used it or have not used it in a while.** |
| **2.** | **Phase 2:** | Developing attack scripts with chess rules in such as capturing, different variations of movement two action per turn, the knight can not move directly on top of a piece to capture etc etc. .  Continue working on improving movement scripts  Improving and further developing the GUI  Implement high score and timer.  Continuing working on the AI.  Having the basic game work.  Improving and finishing the classes  Implement movement highlighter | Project Presentation  Ppt to D2L  Prototype 2 | 30 points |
| **Challenges/Risk: High possibility of facing AI glitches. Possible game crashes due to errors in the logic. Pro chess experience being programed into the AI may result as a challenge.** |
| **3.** | **Phase 3:**  **&**  **Complete** | Implement 3D modeled chess pieces where the place holders currently are.  Bug and glitch fixes  Adding animations  Finishing what is left of AI scripting  Polishing graphics and movement  Improved and finished GUI menu with added features  Add additional features we may think of  Add sound  Optional features that could be added:  A player timer for possible speed chess  Player vs player mode  AI vs AI mode  Spectator mode  Background animations  Difficulty selection  Complete:  More polishing if needed as well as bug fixes  Lighting changes if needed  Making sure the code is efficient and readable  Fixing up any possible logic errors and possible lag  Clearing up AI wait times setting a time cap to what is expected. | Project Presentation  Ppt to D2L  Prototype 3  Complete Project to D2L | 40 points |
| **Challenges/Risk:**  Possible bug fixes can lead to new bugs and glitches. Sound quality issues can occur. AI not responding to commands is a possible error. Feature creep can happen if we don’t balance work load correctly and do too much that is not needed. |
| **Challenges/Risk:**  **Game could possible lag or run slow. AI response could delay itself due to a small error. Lighting may not be correct and may look different on a different screen.** |