**CS 4850 Form1: Analysis of Project and Team Plan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Name:** | | | Project 4B: AI and Chess Variant | | | |
| **Team Members:** | | | Ryan Deem, Bram Altunian, Jonathan Parks, Michelle Campbell | | | |
| **Description:** | | | Develop a chess playing application, with a GUI that displays the board and allows the human player to play against a distributed AI model. All moves must be consistent with the rules of the game. | | | |
| **Components Needed** | | | | **Skills Required** | **Team Members** | |
| **Name** | **Skillset** |
|  | *Describe this component of the end system* | | *What is needed in order to build this component* | *Who* | *What skills do they currently have, and what skills will they need to develop* |
| **1.** | Create the environment: an 8x8 chessboard | | Arrays | All | Data Structures |
| **2.** | Create an agent: an AI “player”: using a distributed AI model | | Evaluation of current position and potential “safe” moves;  Maybe Minimax, but currently unsure if this is something we can use for the current program derivations | All | Java or Python, depending on whether the AI is more difficult, or the GUI is. |
| **3.** | Human-player: input moves from a human player interacting with the agent | | Evaluate move legality based on set chess game rule constraints | All | Dependent on the language we end up working with based on #2 |
| **4.** | Create GUI interface | | Create GUI board, pieces, and user input | All | Python libraries or using C# with Visual Studio |
| **5.** |  | |  |  |  |
| **6.** |  | |  |  |  |
| *Add additional components or sub-components as needed.* | | | | | |