Vision Document

Introduction

For this project we are looking to build a computerized version of the game called ‘Quoridor.’ The game can have up to four players, of which at least one must be a human player and the others will be CPU. The game will allow for the manipulation of the pawn and fence pieces on the board. The user will have the option of playing the game in two difficulty settings for their convenience. Furthermore, the progress of the game will be recorded so that the user can resume the saved game at a later time. Finally, the game will be able to determine a winner.

Problem Statement and Product Features

We are looking to create a computerized version of a classic board game and therefore, our digital game would bring all the perks and features that cannot be seen in a physical version. Misplacing a few key pieces can hinder a board game, therefore, users need not worry about that on our computerized version.Furthermore, users can save a game and resume it later without having to worry about the physical mess. The biggest constraint for the physical board game is that the players have to be in close proximity to each other. However, we plan to expand our game to a network so that users can play against each other over LAN/WAN. Another problem we often find is that the abilities of a skilled player are often not matched because of the physical constraint. Our game will have two difficulty settings (easy and hard) which will make the game much more challenging for veteren players, while going easy on newbies.
Last but not least, our game will accommodate those with colour vision deficiency so that all players can equally enjoy the game.

Key Stakeholder	Description	Needs & Scope
Player	The user/CPU	● Game should not lag or crash. ● Game can be opened/closed any instance. ● Game can be saved and resumed at a later time.
Software Architects	The initial builders of the software (Group 2)	● The program should achieve all its goals. ● Ability to easily amend the program for bugs. ● Program should be comprehensible by other programmers to collaborate.
Collaborators	Other programmers who did not initially work on the software but may do so in the future.	● Ability to comprehend the program and its goals with ease. ● Ability to easily amend updates or features. ● Ability to easily remove bugs.