Wandering in the Woods Game

Software Engineering, Fall 2022-0004

Group 3

Nithish Reddy Pinninti

Navya

Vamshidhar

Ajay

Akshitha

Table of Contents

[1.0 Introduction 2](#_30j0zll)

[1.1 Purpose 2](#_1fob9te)

[1.2 The](#_3znysh7) Game 2

[2.0](#_2et92p0) Agile Software Development

2.1 Principles of Agile Methodology  [3](#_2et92p0)

[3.0 Use Cases 4](#_tyjcwt)

[3.1 Use Case 1: C](#_3dy6vkm)reating the main Screen 4

[3.2 Use Case 2: Plot a K2 grade Block 4](#_1t3h5sf)

[3.3 Use Case 3: P](#_4d34og8)lot a K3-5 grade Block 4

[3.4 Use Case 4: Pl](#_2s8eyo1)ot a K6-8 grade Block 5

[3.5 Use Case 5:](#_17dp8vu) Edit Game 5

[4.0 UML Model 6](#_26in1rg)

[4.1 Use Case Diagram 6](#_lnxbz9)

[4.2 Deployment Diagram 6](#_35nkun2)

[4.3 Class Diagram 7](#_1ksv4uv)

[4.4 State Diagram 10](#_44sinio)

[4.5 Activity Diagram 11](#_2jxsxqh)

[8.0 Testing Strategy 15](#_4i7ojhp)

[8.1 Unit Testing 15](#_1ci93xb)

[8.2 Integration Testing 16](#_2bn6wsx)

[8.3 Usability Testing 17](#_1pxezwc)

[8.4 Validation Testing 17](#_49x2ik5)

# Introduction

## Purpose

The game Wandering in the Woods is a simple simulation in which two figures on a screen go around at random.The characters move across the screen until they collide or find a common ground. The basic wandering in the woods game consists of two players and a predefined square screen playing environment. The game is about finding each other while moving. It has three stages in the game. The first stage is for K-2 students who can play the game with the visuals while playing and also background music which makes the K-2 students interesting.

The second stage of the game is for grades 3-5 which is slightly more challenging where they can make choices, see the data and meet additional challenges. If the game is played with more than 2 players in the game if the first player finds the second player then both the players will wander to find the third player likewise if the third player found all the three will move to find the fourth player when all the players are found the game ends.

Moving to grade 6-8 in this case the game becomes more challenging with increasing difficulty students can choose the grid size and number of players which make it more complex. For the grade 6-8 has all the controls of grade 3-5 and they will be able to explore different ways to wander, find the best way to meet and shortest way to finish the game.

## The Game

As the game progresses the first screen will be asking please select the grade you want to play. It displays all the three levels that are K-2, grade 3-5 and grade 6-8 from which the user picks one of the levels to start. After he chooses the level or grade it will prompt to select the grid size to play, then the user selects the grid size and next the system asks for the number of players to choose from 2 to 4.

After a selection of a number of players the game starts and players move from different corners to find each other once they find each other the happy music plays and the game ends. Same way if they are playing with more than 2 players if any one finds the second person they will move along to find others. After the end of the game it goes back to the first page so that it becomes easy to restart the game again.

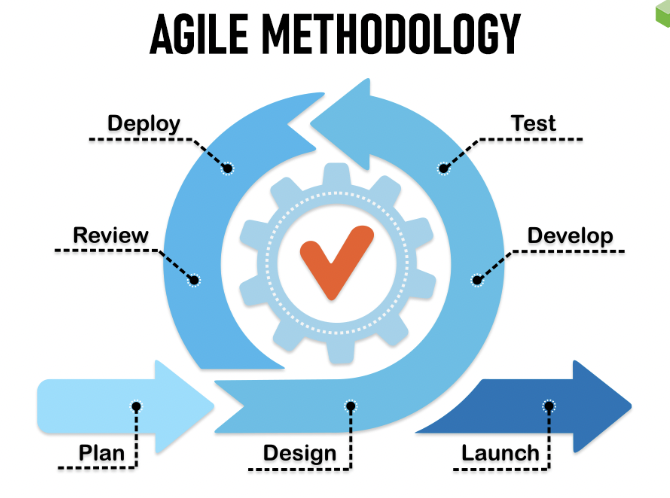
# Process Model:

# 2.1 Agile Software Development:

Agile software development is a type of development process where the process is divided into smaller units and each function is completed separately. The "Waterfall 4 model" was replaced with the agile development technique. Because the waterfall process takes so long, lengthy items are delivered. And if further standards are introduced, the outcome will be a flawed product.

Agile methodology, however, encourages responsive behavior in which the customer and the developer are linked, and the development team responds quickly to requests for adjustments or the addition of new features to the program. Agile methodology reduces development time while increasing product quality. Below the Figure.1 is a diagram of the process of agile development.

The teamwork component of the agile development process is key. All of the jobs are broken up into their appropriate teams and completed in order. A high-quality product is ultimately produced.



**Fig 1: Agile Methodology Development**

2.2 Principles of Agile Methodology:

* Customer satisfaction is our first focus, and we achieve this through timely and consistent delivery of high-quality software.
* Even late in the development process, accept changing needs. Agile methodologies harness change for the benefit of the customer's competitiveness.
* Deliver working software frequently, preferably in shorter time frames of a few weeks.
* Throughout the project, businesspeople and developers must collaborate everyday.
* Build initiatives around motivated people. Trust them to complete the task and provide them with the environment and assistance they require.
* A face-to-face talk is the most effective and efficient way to communicate information to and within a development team.

2.3 The Game Process Model

* The process model for the "Lost in the Woods" game describes the steps involved in playing the game. Here is a summary of the process model for the game:
* The game starts with the players in diagonally opposite corners of the grid. The grid is displayed on the canvas in the GUI, and the players' symbols are placed on the grid to indicate their starting positions.
* Player 1 enters their move by clicking a button in the GUI and selecting a direction (up, down, left, or right). The game updates the grid to reflect the player's move and redraws the grid on the canvas.
* The game checks if player 1 has found player 2. If player 1 has found player 2, the game ends and a message is displayed on the GUI indicating that player 1 has won.
* If player 1 has not found player 2, it is player 2's turn. Player 2 enters their move in the same way as player 1. The game updates the grid and redraws it on the canvas.
* The game checks if player 2 has found player 1. If player 2 has found player 1, the game ends and a message is displayed on the GUI indicating that player 2 has won.
* If neither player has found the other, the game repeats steps 2 through 5 until one player has found the other.
* This process continues until a player has found other player, at which point the game ends.

# Use Cases

Use cases are a way for locating, outlining, and organizing system needs in system analysis. The use case consists of a number of potential interactions between users and systems in a specific environment that are connected to a specific objective. The procedure generates a document that lists each step a user took to finish an activity.

The following use cases were defined by the requirements for the delivery of the game. Each use case has a list of the name of the use case, primary actors, preconditions, description, and acceptance criteria.

**Description**: A user creates a new game of "Lost in the Woods" by starting the game and setting up the grid and GUI.

**Preconditions:** The game has been installed and the user has opened it.

**Main Flow:** The user clicks the "New game" button in the GUI.The game creates a new grid and displays it on the canvas.The game places the players' symbols on the grid to indicate their starting positions. The game displays the buttons and labels needed for the players to enter their moves and for the game to display messages.

**Postconditions:** A new game of "Lost in the Woods" has been created and is ready for the players to start playing.

## Use Case 1: Creating the main screen

**Primary Actor:** Student/Player/User

**Preconditions:** The game has been installed and the user has opened it.

**Description:** A user creates a new game of "Lost in the Woods" by starting the game and setting up the grid and GUI.

The user clicks the "New game" button in the GUI. The game creates a new grid and displays it on the canvas.The game places the players' symbols on the grid to indicate their starting positions. The game displays the buttons and labels needed for the players to enter their moves and for the game to display messages.

**Acceptance Criteria:** I can choose which grade to play and with the number players to play along with the grid size

**Post Condition:**  A new game of "Lost in the Woods" has been created and is ready for the players to start playing

## Use Case 2: Plot a K2 grade block

**Primary Actor:** Student,Player/User

**Preconditions:** A game has been designed where one select the plotted K2 grade block

**Description:** As a player I want to be able to choose/ select a K2 grade block. I should be able to click on the block to select the K2 grade. I should be able to play with 2 players initially and after the game finishes a happy music should be played and a deadly music should be played if the players are moving in the opposite direction

**Acceptance Criteria:** I can select and choose the players while choosing the grade blocks

Use Case 3: Create a K3-5 grade block

**Primary Actor:** Student/User/Player

**Preconditions:** The game has been designed to hold the different grade blocks data, a K2 grade block has been created

**Description:** As a player I want to be able to choose/select the grade blocks. Here I want to select a K3-5 grade block. I should be able to click on the block to select the K3-5 grade block. I should be able to choose more than 2. If the first player finds the second player a happy music must be played. 2 players who found each other should find the next player. When all the players find each other a happy music must be played. If the players are moving towards the opposite direction a deadly music must be played. After finishing the game a home page music must be played. The number of players playing must be flexible to choose in K3-5 grade block.

**Acceptance Criteria:** I can be able to choose the number of players while choosing the grade blocks

## Use case 4: Plot a K5-8 grade block

**Primary Actor:** Student/ User/ Player

**Preconditions:** The game has been designed to hold the different grade blocks data. A K2 grade block and a K3-5 grade block has been created

**Description:** As a player I want to be able to choose the number of players along the grid size. I want some happy music played when the players find each other, I want deadly music to be played when the players are running in the opposite direction. And a home page music when the game is finished as in when all the players find each other

**Acceptance Criteria:** I can be able to choose the number of players and the size of the grid

3.4 Use case: Edit Game

**Primary Actor:** Player/ User/ Student

**Preconditions:** The game has been designed to hold the different grade blocks data. A K2 grade block, K3-5 grade block and a k6-8 grade blocks are created

**Description:**  As a user I want to be able to rechoose the blocks, the number of players, the grid size and the music that has to be changed

**Acceptance Criteria:** I can make changes to the number of players and the grid size.

# UML Model

Different levels of detail are represented by systems in Unified Modeling Language (UML) models. While other models provide more detail, others represent a system at a higher, more abstract level. UML models include one or more diagrams that depict a particular view of a system together with model features like actors, use cases, classes, and packages. A model may also include additional, more comprehensive models.

## Use Case Diagram

Use case diagrams are used to compile a system's requirements, taking into account both internal and external factors. Most of these needs are for the design. Therefore, use cases are created and actors are identified when a system is evaluated to gather its functionality.

Below figure.1 displays the use cases included in the use case diagram. The user is the only player who interfaces with the Game Designer System. The user is able to choose the number of players, and the size of the grid. Among the use cases the user is able to change the type of music, the players and the size of the grids based on the created grade blocks. Each of the created use cases can be extended by the Edit Game use case. This Edit game use case will help the user to modify any changes to the different use cases.

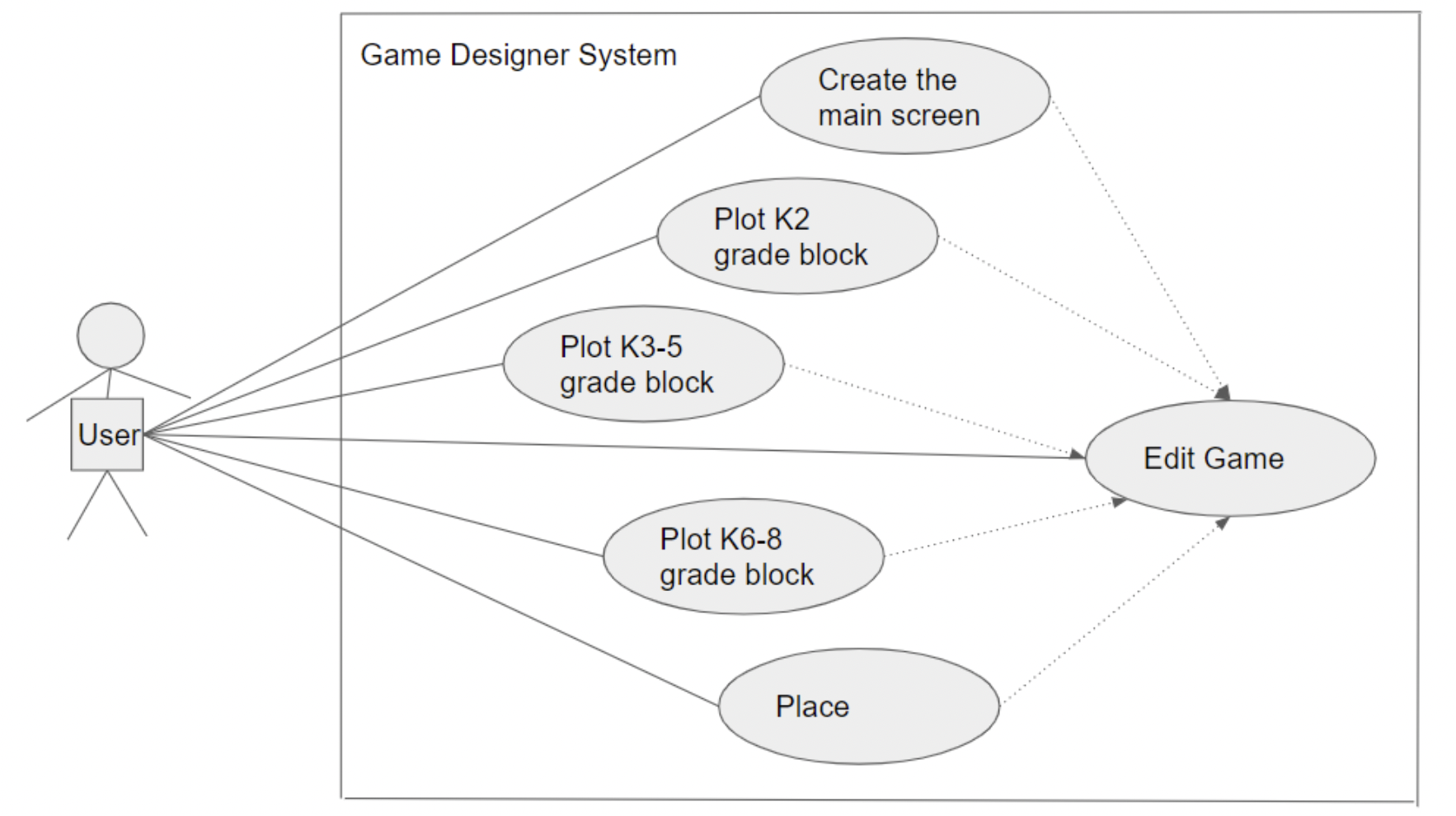


Fig 2: Use Case Diagram

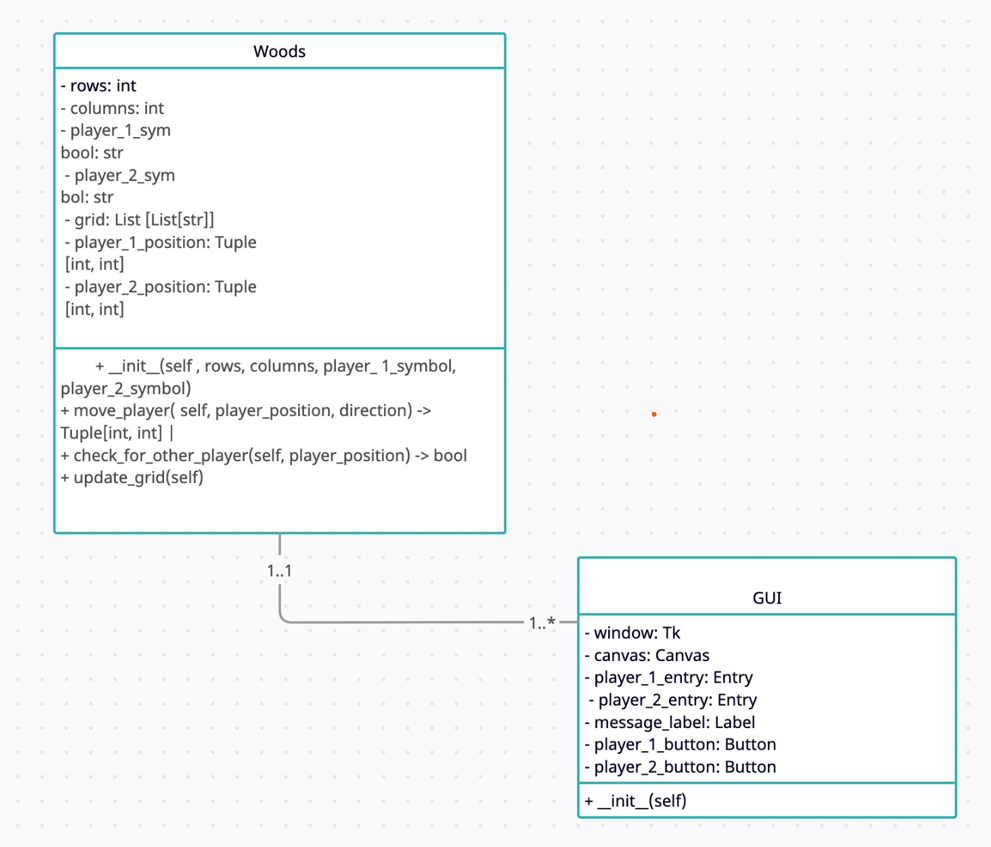
## Deployment Diagram

## 

A deployment diagram identifies the actual hardware that the software system will run on. Additionally, it controls the software's implementation on the underlying hardware. It associates system software components with the hardware that will run them.

## Class Diagram

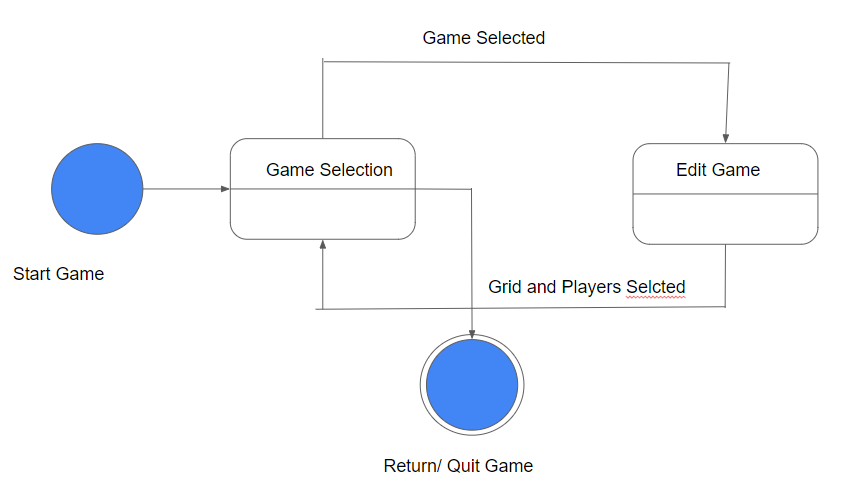
One of the most helpful forms of diagrams in UML are class diagrams, which accurately depict a system's structure by modeling its classes, properties, operations, and relationships among objects. Making these diagrams is easier than it would seem with the help of our UML diagramming tools. You will learn how to comprehend, organize, and produce your own class diagrams using this manual.



## State Diagram

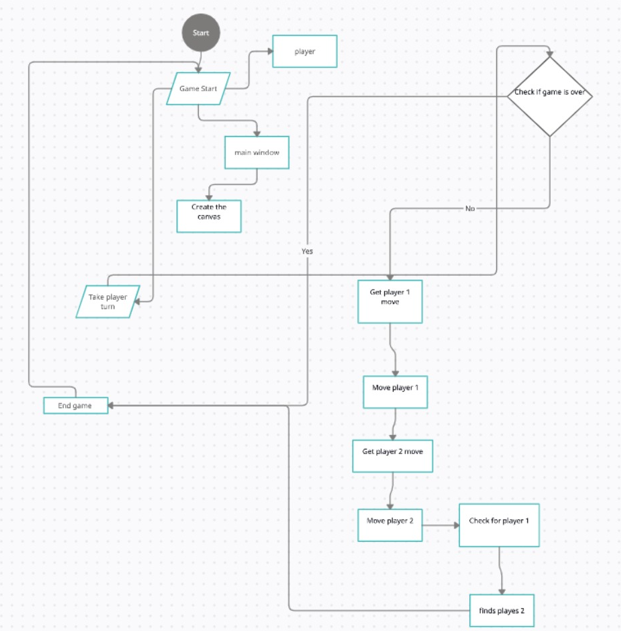
This state diagram shows the different states that the "Lost in the Woods" game can be in, as well as the events that trigger transitions between those states.

The game starts in the "Start" state and then transitions to the "Player 1 turn" state when the "Create game" event is triggered. From the "Player 1 turn" state, the game can either transition to the "Player 2 turn" state if the "Player 1 move" event is triggered,or to the "Game over" state if the "Player 1 win" event is triggered. Similarly, from the "Player 2 turn" state, the game can either transition to the "Player 1 turn" state if the "Player 2 move" event is triggered, or to the "Game over" state if the "Player 2 win" event is triggered.



## Activity Diagram

This activity diagram shows the various steps involved in playing a game of "Lost in the Woods".It starts with the "Create game" activity, which sets up the grid and GUI for the game. The "Player 1 move" and "Player 2 move" activities represent the players taking turns entering their moves. The "Check for win" activity represents the game checking if either player has found the other player.If a player has won, the "Game over" activity is triggered and the game ends. If no player has won, the process repeats with the next player's turn.



Design of the Game:

The purpose of the game is to teach students about probability and statistical analysis in an interactive and engaging way. The game is intended for students in grades K-8.

The functional requirements of the game include the ability to select different modes of play (K-2, 3-5, 6-8), customize the size and shape of the grid, place players on the grid, and track the movement and meeting of players. The K-2 mode features a square grid and two players that start out in diagonally opposite corners. The 3-5 and 6-8 modes allow for rectangular grids and the option to have 2, 3, or 4 players, who can be placed anywhere on the grid. When two players meet in the 3-5 and 6-8 modes, they will move together until they find the third player, and if the third player is found (assuming there are 4 players), the three players will move together until the fourth player is found.

In addition to the functional requirements, the game also has non-functional requirements including good performance, security, and usability. The game should run smoothly and efficiently, and it should be secure to protect users' personal information. It should also be easy for users to understand and navigate.

To help visualize how the game will work, I have created a flowchart that illustrates the steps a user will take to interact with the game. The user will start by selecting the mode of play and customizing the grid and player positions, if applicable. They will then start the game and watch as the players move around the grid randomly. When two players meet, there will be a happy graphics display and statistics from the wandering will be displayed and announced audibly. The game can then be reset and played again. In the 3-5 and 6-8 modes, statistics such as the longest run without meeting, shortest run, and average run will be displayed.

I have also designed the user interface using wireframes or mockups. The interface should be visually appealing and easy to use, with clear buttons and instructions for navigating the game.My development plan includes tasks such as coding the game mechanics, implementing the user interface, and testing the game. I will use python or higher to develop the game, and it will require a compatible hardware and software setup to run.

The game mechanics include rules for movement and finding other players. In the K-2 mode, the players will move randomly around the grid, and when they meet, the game will end and display the statistics. In the 3-5 and 6-8 modes, if two players meet, they will move together until they find the third player, and if the third player is found (assuming there are 4 players), the three players will move together until the fourth player is found.I have identified some potential risks or challenges that may arise during development or deployment of the game, such as difficulty implementing certain features or compatibility issues.I will address these risks as they arise and make any necessary revisions to the design as needed.

Overall, the Wandering in the Woods game in python is an interactive and engaging way for students to learn about probability and statistical analysis. It allows for customization of the grid size and player positions, and it provides a fun and interactive way for students to explore different protocols for wandering and determine which is the most effective at minimizing the time it takes for players to meet up. The game also provides an opportunity for students to develop their critical thinking and problem-solving skills by conducting experiments and analyzing data.

Software Requirements:

* ● Download and install PyCharm website (https://www.jetbrains.com/pycharm/).
* ● Click the "Download" button on the top right of the page.
* ● Select the edition of PyCharm that you want to download. There are two main editions:  
  the Community Edition, which is free and open-source, and the Professional Edition,  
  which is paid and includes additional features.
* ● Select the operating system that you are using (Windows, macOS, or Linux).
* ● Click the "Download" button for your selected edition and operating system. This will  
  start the download of the PyCharm installation file.
* ● Once the download is complete, locate the downloaded file and double-click it to start the installation process. Follow the prompts to install PyCharm on your system.
* ● Once the installation is complete, you can launch PyCharm from the start menu (on  
  Windows) or the Applications folder (on macOS).

# Testing Strategy

Unit Testing:

Unit testing is a type of testing that focuses on individual units or components of the

game, such as functions or classes. For the "Lost in the Woods" game, unit tests could be written to test the functionality of the grid, players, and GUI, as well as the game logic and movement functions.

Integration Testing:

Integration testing is a type of testing that focuses on the interactions between different

units or components of the game. For the "Lost in the Woods" game, integration tests could be written to test the integration of the grid, players, and GUI with the game logic and movement functions.

Usability Testing:

Usability testing is a type of testing that focuses on the user experience of the game.

For the "Lost in the Woods" game, usability tests could be conducted to ensure that the GUI is easy to use and understand, and that the game is enjoyable to play.

Validation Testing:

Validation testing is a type of testing that focuses on the accuracy and correctness of

the game. For the "Lost in the Woods" game, validation tests could be conducted to ensure that the game logic is correct and that the game behaves as expected under different scenarios.