Wandering in the Woods Game: Agile Project Documentation

Rachael Morrison, Benjamin Ayirifa, Shivasai Priyatham Kota, and Siva Prasad Polisetty

College of Aviation, Science, and Technology, Lewis University

FA22-CPSC-60500-004 Software Engineering

Dr. Fadi I. Wedyan

December 16, 2022

Table of Contents

[1.0 Introduction 2](#_heading=h.30j0zll)

[1.1 Purpose 2](#_heading=h.1fob9te)

[1.2](#_heading=h.3znysh7) Description 2

[2.0 Process Model](#_heading=h.2et92p0) 3

[3.0 Use Cases](#_heading=h.tyjcwt) 4

[3.1 Use Case 1:](#_heading=h.3dy6vkm) 4

[3.2 Use Case 2:](#_heading=h.1t3h5sf) 4

[3.3 Use Case 3:](#_heading=h.4d34og8) 4

[3.4 Use Case 4:](#_heading=h.2s8eyo1) 4

[3.5 Use Case 5:](#_heading=h.17dp8vu) 5

[3.6 Use case 6:](#_heading=h.3rdcrjn) 5

[4.0 UML Models](#_heading=h.26in1rg) 6

[4.1 Use Case Diagram](#_heading=h.lnxbz9) 6

[4.2 Class Diagram](#_heading=h.1ksv4uv) 7

[4.3 Activity Diagram](#_heading=h.2jxsxqh) 8

5[.0 UI Mock-up](#_heading=h.1y810tw) 9

6[.0](#_heading=h.4i7ojhp) User Guide 10

# Introduction

## Purpose

This document describes the software design of the Wandering in the Woods game. The program is intended to kelp K-8 students learn about computation, computational thinking, math concepts, and computer science. After being distributed to teachers, the game can be utilized in a class setting for educational purposes. The three-level design allows students to work with more difficult concepts as they progress through their school years.

## Description

Wandering in the Woods is a game where people are “lost in the woods” where the woods are represented by a rectangular grid. The woods are dense, and the people can’t see or hear each other until they are in the same cell of the grid. There are three levels for students to try as they progress through their school years.

Grades K-2 Level: The grids are always square, there are always two people, and they start out in diagonally opposite corners of the grid. They wander about randomly, and each move is counted, with a counter for each person. Run time is displayed in the upper right corner. Music plays as cartoon characters wander in the woods. When the people bump into each other, the statistics from the wandering are displayed and the player is given the option to restart or return to the main menu. When the game is reset, the players return to their starting points and the game starts up again. Average and best run times are saved to be displayed every time the level ends.

Grades 3-5 Level: The game is set up the same as the K-2 level. However, students can now set up the size of a grid, which can be rectangular (instead of just square). There can be 2, 3, or 4 people, and students can place them wherever on their grid. If the game is played with 3 or 4 players, if two find each other, they will be moving together until they find the third player. In the same way, if the third player is found (assuming we have 4 players), the three players will move together until the fourth player is found

 Grades 6-8 Level: The game is set up the same as the 3-5 level, but 6-8 students will be challenged to run experiments to determine how the average run varies with the size and shape of the grids. They will also be able to explore different protocols for wandering, and to decide which is the best way to wander if you want to shorten the time it takes to meet up. When this level is reset, the player has the option to change the grid size, the number of players on the board, and the wandering protocol.

# Process Model

The software development team will use the evolutionary process model as show in Figure 1. The model the team selected is based on the desire to allow for rapid prototyping and iteration of design. The model is based on agile and spiral model principals to encapsulate feedback and risk management into the decisions making for continuous evolution of the game.

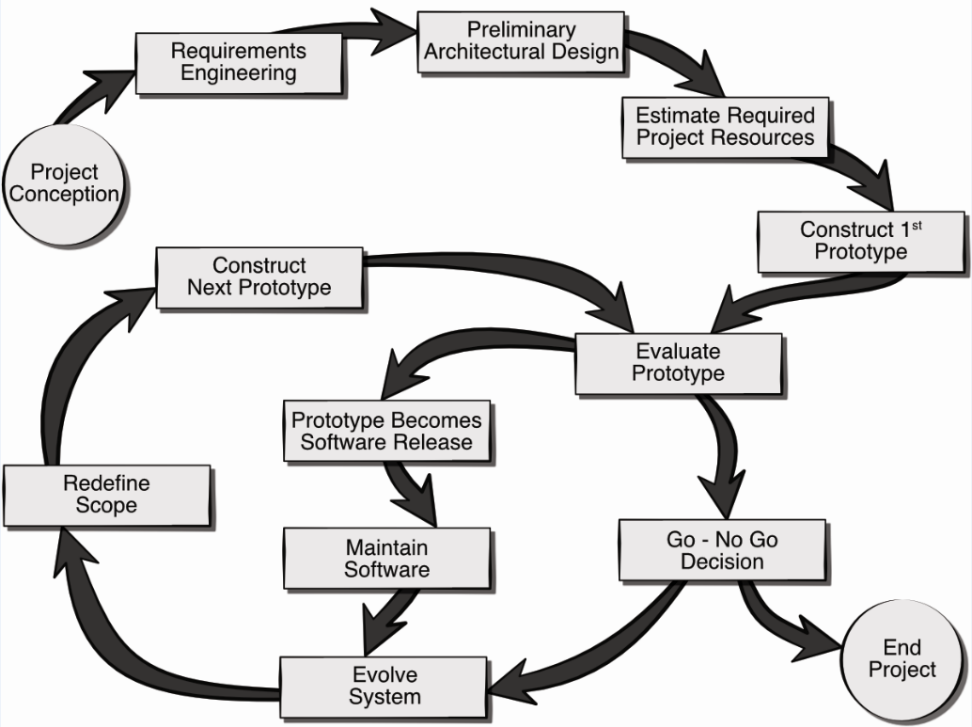


Figure 1

*Figure SEQ Figure \\* ARABIC 1. The AR Room Designer evolutionary process model.*

# Use Cases

The following uses cases were defined by the team as the core system requirements for the delivery of working prototype. Note, this baseline functionality can be easily extended through the inclusion of more use cases as the project progresses. Each use case lists the name of the use case, primary actors, preconditions, description, and acceptance criteria.

## Use Case 1: Select a Level

**Primary Actor:** Student

**Preconditions:** The game was started by running the executable.

**Description:** As a student I want to be able to select the K-2 level, the 3-5 level, or the 6-8 level depending on what grade I am in.

**Acceptance Criteria:** I can click a button which selects the level that I want to play, and I move to the next screen.

## Use Case 2: Select Number of Players

**Primary Actor:** Student

**Preconditions:** I just selected either the 3-5 level or the 6-8 level on the main menu.

**Description:** As a student I want to be able to use the arrow buttons to select either 2, 3, or 4 players to walk around the grid.

**Acceptance Criteria:** I can use the arrow buttons to change the amount of players, and I can click the confirm button to move to the next screen.

## Use Case 3: Set Grid Size

**Primary Actor:** Student

**Preconditions:** The 3-5 level or the 6-8 level was selected and I either just selected the number of players or I just reset the level.

**Description:** As a student I want to be able to use the arrow buttons set the length and width of the grid to a size between 2 and 12.

**Acceptance Criteria:** I can use the arrow buttons to change the length and width of the grid, and I can click the confirm button to move to the next screen.

## Use Case 4: Choose Wandering Protocol

**Primary Actor:** Student

**Preconditions:** The 6-8 level was selected, and I just set the grid size.

**Description:** As a student I want to be able to choose between the ‘random’ and the ‘every other’ wandering protocols using the arrow buttons.

**Acceptance Criteria:** I can use the arrow buttons to choose the wandering protocol

## Use Case 5: Place Players

**Primary Actor:** Student

**Preconditions:** The 3-5 level or the 6-8 level was selected and I either just set the grid size or I chose the wandering protocol.

**Description:** As a student I want to be able to place the players on the grid wherever I want by using the arrow keys to move them and the enter key to confirm their placement.

**Acceptance Criteria:** I can place the players on the grid, which causes the level to automatically start and the players start moving around the grid.

## Use case 6: Reset Game

**Primary Actor:** Student

**Preconditions:** The level started and all the players in the grid have found each other and the game over screen is shown.

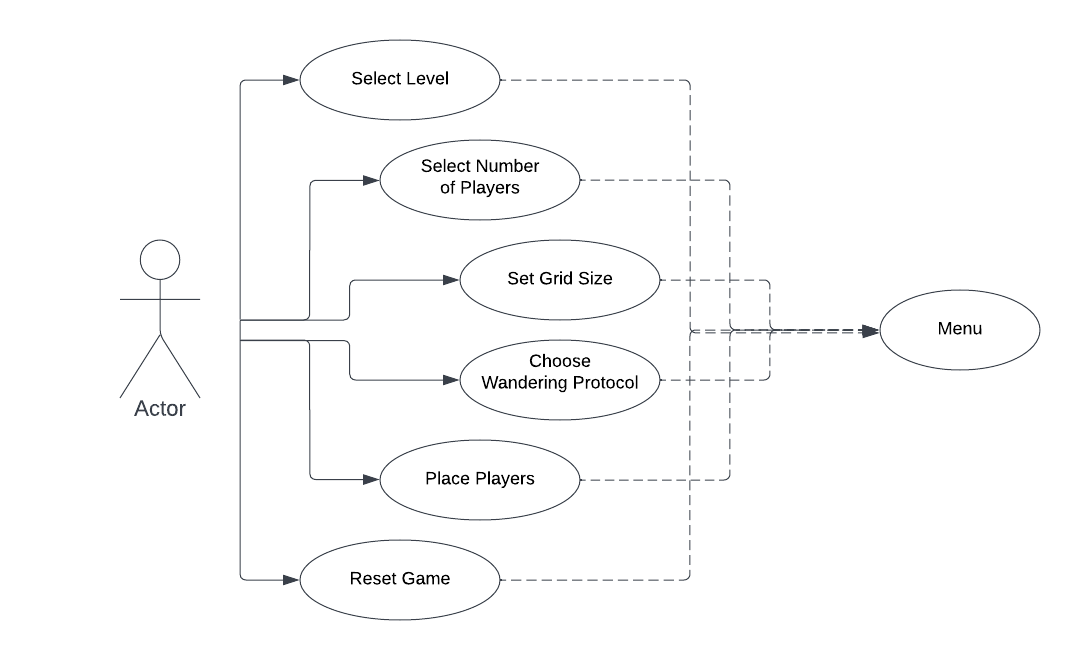
**Description:** As a student I want to be able to reset the level using the reset button in the game over screen so that I can play the same level again.

**Acceptance Criteria:** I can click the reset button which resets the level. If I had selected the 3-5 or 6-8 levels, I am able to change the grid size, wandering protocol, and player placement.

# UML Model

## Use Case Diagram

The six use cases are included in the use case diagram as seen in Figure 2. The only actor I the student, who interacts with several screens from the menu that allow them to pick the level, change game settings, start the game, and reset the game.



Figure

## Class Diagram

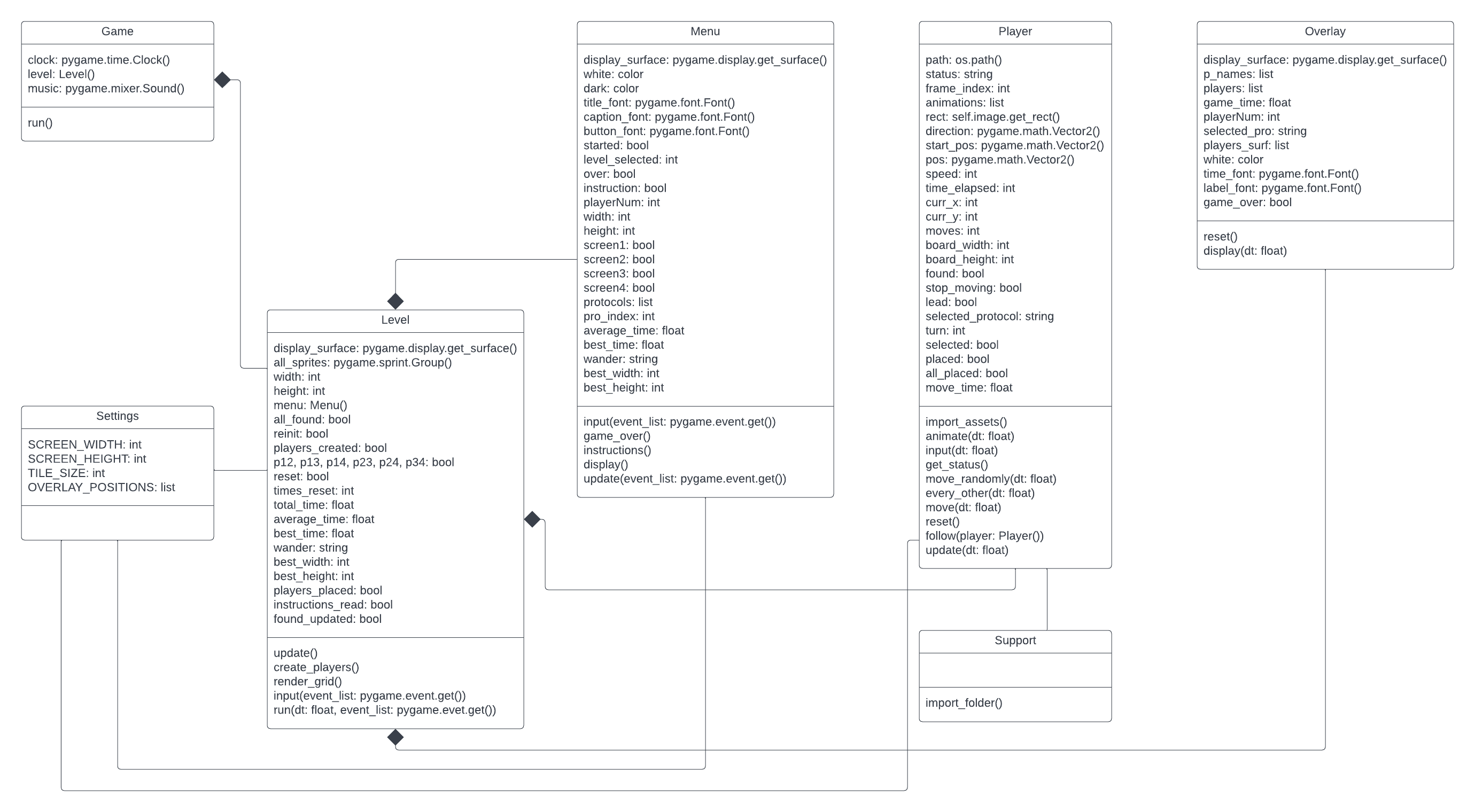
The class diagram for the core system of the Wandering in the Woods game is depicted in Figure 3. The classes in the diagram are described below the figure.

Figure 3

Game: sets up and starts the game

Level: Level within the game that acts as the backbone that initializes and updates the menu, players, and overlay as needed, and keeps track of other game settings as needed

Menu: the menu screens for the game that the user can navigate through

Player: represents a player that will randomly walk around the grid, staying within the grid’s boundaries, until all players are found

Overlay: displays current statistics for the level being played on the right side of the screen

Settings: general settings for screen, tiles, and overlay positions

Support: supports file imports for importing player graphics

## Activity Diagram

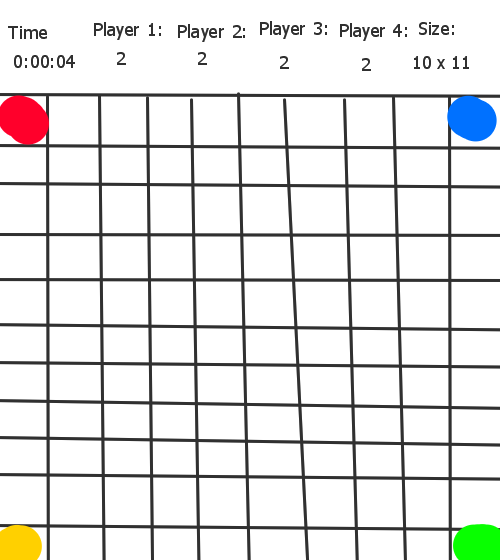
The activity diagram shown in Figure 4 presents a more detailed description of the high-level behavior of the Wandering in the Woods game.



Figure

# UI Mock-up

The initial user interface mock-up for the Wandering in the Woods game is presented in Figure 5. The players on the grid are represented by colored dots The layout was changed to allow for different sized grids within the same screen size, and the overlay was moved to the right side of the screen in the final product. Time, player counters, and other settings are displayed in the overlay.



Figure

# User Manual (For Teachers)

To prepare the game for student use, clone or download the project from <https://github.com/rayemorr/CPSC60500-Project>. Only the executable folder from this project is needed, but please ensure that the contents of this folder stay together for the game to run properly. Once you extract the executable folder from the project, all that is needed to launch the game is to run the ‘Wandering in the Woods.exe’ file. Once the game is launched, either pick the grade level of the students or allow the students to pick the grade level themselves from the main menu. From then on, the game’s instructions will be enough to guide the student through the simulation.