Wandering in the Woods Game: Agile Project Documentation

by

Erjon Parashumti with Justin Morgan and Marvin Martinez

[**1.0 Introduction 3**](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.wyq3dvrg4n28)

[1.1 Purpose 3](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.7p9xg3jn8voj)

[1.2 Wandering in the Woods 3](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.mh1fdt1vufka)

[**2.0 System Features and Requirements 4**](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.s71if5jf66d5)

[2.1 Functional Requirements 4](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.q2f7pjuzofl2)

[2.2. External Interface Requirements 4](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.4dq8v3g97s1h)

[2.3 System Features 5](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.g06xx91rkgln)

[2.4 Nonfunctional Requirements 5](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.tcwu4oeblpyx)

[**3.0 Process Model 5**](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.kloxjepnmhyr)

[**4.0 Use Cases 6**](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.9ugjvxjpvw1r)

[4.1 Use case 1: Setting Up the Game Environment 6](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.jmvhx4j402b)

[4.2 Use case 2: Playing the Game 7](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.mrgq50yej42i)

[4.3 Use case 3: Analyzing Game Outcomes 7](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.mwza4dhbx7dy)

[4.4 Use case 4: Game Settings and Features for Grades 3-8 7](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.xzs92psck6t9)

[4.5 Use case 5: Restarting and Replaying the Game 8](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.eylz64err5sq)

[**5.0 UML Models 9**](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.tn13o5c0r21)

[5.1 Use Case Diagram 9](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.qnzuqbz3nioc)

[5.2 Deployment Diagram 10](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.kowj68lweh5g)

[5.3 Class Diagram 11](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.r97n0v2gjh81)

[5.4 State Diagram 13](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.tz9e12gwcnie)

[5.5 Activity Diagram 14](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.n0gazidglu6l)

[5.6 Customer Journey Diagram 15](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.g2gv5t2si30)

[5.6 Personas 16](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.vp2faswk2bch)

[5.6 Game UI Visualization 17](https://docs.google.com/document/d/1xCAGuyL89h-OIk2WOQY0ry7CwDjELaMsG9yKm1yiDSs/edit#heading=h.o3g4w1k2xmhe)

1.0 Introduction

1.1 Purpose

This document serves as a comprehensive guide to the software design of the "Wandering in the Woods" game, an educational and interactive game developed as part of our coursework under the guidance of Professor Fadi. Created by Erjon Parashumti, Justin Morgan, and Marvin Martinez, this agile document aims to elucidate the design processes, techniques, and theoretical frameworks underpinning our implementation of the game.

1.2 Wandering in the Woods

The "Wandering in the Woods" game is designed to engage students across various grade levels (K-8) in an immersive learning experience. Through gameplay, students explore mathematical concepts, develop problem-solving skills, and engage in critical thinking. The game is set in a dense forest, represented by a grid where players—lost and unable to see or hear each other—wander randomly until they meet in the same cell.

*For Grades K-2*: The game features a square grid with two players starting at diagonally opposite corners. As they wander randomly, their movements are counted and displayed, accompanied by music and cartoon animations. Upon meeting, the game celebrates the occasion with a joyful graphics display, while statistics are both audibly announced and visually displayed.

*For Grades 3-5*: Students gain the ability to customize the grid size, allowing for rectangular shapes, and choose between 2, 3, or 4 players, placing them at desired starting points. The game supports multiple replays, displaying statistics like the longest run without a meeting, shortest run, and average run. When more than two players are playing, they group, seeking the next player until the 3rd or 4th player is found.

*For Grades 6-8*: Building upon the abilities provided in grades 3-5, students in grades 6-8 are encouraged to conduct experiments to determine how the average run varies with grid size and shape. The students have the opportunity to experiment with various wandering strategies to optimize the time it takes for them to meet.

2.0 System Features and Requirements

2.1 Functional Requirements

The game must run on a system capable of rendering 2D graphics. Support user interaction through a graphical user interface (GUI), allowing for grid customization, player movement, and statistical display.

2.2. External Interface Requirements

The game requires a display capable of visualizing the game's GUI, speakers or headphones for audio feedback, and a keyboard or mouse for user input.

2.3 System Features

Compatible with systems that support Python and the Tkinter library for the GU

2.4 Nonfunctional Requirements

The game should be responsive, with minimal delay in player movement and interaction. Requires a system with at least 2 GB of RAM and a 1.2 GHz processor to ensure smooth gameplay and graphics rendering.

3.0 Process Model

A diagram of software development

Description automatically generated

4.0 Use Cases

4.1 Use case 1: Setting Up the Game Environment

***Primary Actor:*** Student

***Preconditions:*** The game application has been installed and launched successfully. The user peripherals for the system are accessible and functional.

***Description:*** The player (student or educator) initiates the setup process for a new game session. The player selects the desired grid size and shape, which varies based on the grade level.

* For grades K-2, the grid is square-shaped, and the size is preset.
  + There are 2 players for grades K-2.
  + The players are automatically placed at diagonally opposite corners of the grid.
* For grades 3-8, the grid size and shape (square or rectangular) are customizable.
  + For grades 3-5 the students can set up the size of a grid, either square or rectangular spaces.
  + Students have the option between 2, 3, or 4 players.
  + The player positions are customizable.
* For grades 6-8 additional experimental features are available for students.
  + Students have complete control of the protocols and can experiment with the game settings.

***Acceptance Criteria:*** The game accurately configures and displays the selected grid size and shape, the correct number of players, and their positions on the grid as per user inputs.

4.2 Use case 2: Playing the Game

***Primary Actor:*** Student

***Preconditions:*** The game environment is set up correctly with a selected grid size, shape, and number of players. The student has initiated the start of the game.

***Description:*** The game autonomously moves players randomly within the grid, applying specific meeting logic based on the number of players.

***Acceptance Criteria:*** The players move randomly within the grid and successfully meet according to the game's logic, with the game accurately tracking and responding to these events.

4.3 Use case 3: Analyzing Game Outcomes

***Primary Actor:*** Student

***Preconditions:*** At least one round has been completed where players have met.

***Description:*** After players meet within the game statistics are displayed to analyze the game's outcomes. This includes the number of steps each player took before meeting. For grades 3-5 and 6-8 more extensive statistics are provided for the students.

***Acceptance Criteria:***  Upon the players meeting, the game provides feedback along with statistics for students to analyze. Statistics include the number of steps taken, the longest run without a meeting, the shortest run, and the average run.

4.4 Use case 4: Game Settings and Features for Grades 3-8

***Primary Actor:*** Student

***Preconditions:*** The game is accessible and ready for customization for players in grades 3-8.

***Description:*** The students engage with the game's settings to tailor the gameplay experience to specific goals or curiosities. This includes customizing aspects of the game such as grid size, shape, and wandering protocols to explore mathematical concepts.

***Acceptance Criteria:*** The students can effectively customize game settings, enabling a tailored gameplay experience that aligns with educational objectives and enhances understanding of mathematical concepts.

4.5 Use case 5: Restarting and Replaying the Game

***Primary Actor:*** Student

***Preconditions:*** A game round has concluded, and players have met within the game.

***Description:*** Players can retain the previous game's settings for a new round or reset them according to new preferences. This feature is important for students in grades 6-8, who may be conducting experiments and want to replay the game multiple times to gather data for statistical analysis.

***Acceptance Criteria:*** The game offers an intuitive and seamless process for restarting, with options to either retain the last game's settings or reset them.

5.0 UML Models

5.1 Use Case Diagram

A diagram of a computer game

Description automatically generated

* Use case 1: Setting Up the Game Environment
* Use case 2: Playing the Game
* Use case 3: Analyzing Game Outcomes
* Use case 4: Game Settings and Features for Grades 3-8
* Use case 5: Restarting and Replaying the Game

5.2 Deployment Diagram

A screen shot of a computer

Description automatically generated

5.3 Class Diagram

A diagram of a software developer

Description automatically generated with medium confidence

**WanderingInTheWoodsGame:**

* *\_\_init\_\_(self, game)*: Initializes the game, and sets up the grid and players.
* *player\_posit(self)*: Updates the grid to show the current player positions.
* *player\_move(self, player)*: Moves a player in a random direction on the grid.
* *meeting\_check(self)*: Checks if the players have met.
* *play\_again(self):* Resets the game to its initial state.
* *start(self)*
* *play\_game(self)*: Main game loop.
* *update\_speed(self, speed)*: Adjusts the speed of player movement.
* *background\_music(self)*: Loads and plays the background music for the game.

**Grade3to5 (inherits from WanderingInTheWoodsGame):**

* *\_\_init\_\_(self, game, width, height, num\_players):* Calls the parent class's constructor and sets additional attributes for grades 3-5.
* *setup\_grid(self):* Sets up the game grid.
* *setup\_players(self):* Sets up the players based on the number of players.
* *player\_posit(self):* Overrides the parent method to update player positions for the custom grid.
* *update\_player\_group(self)*: Updates the player groupings based on current positions.
* *player\_move(self, player)*: Overrides the parent method to move players according to game rules for grades 3-5.
* *meeting\_check(self)*: Checks if all players have met and handle game state.
* *start(self)*: Starts or restarts the game.
* *play\_game(self)*: Overrides the main game loop to handle player movements and checks for meetings.
* *play\_again(self)*: Handles game reset for a new round.

**Grade6to8 (inherits from Grade3to5):**

* \_\_init\_\_(self, game, width, height, num\_players): Inherits the constructor from Grade3to5.
* *player\_move(self, player):* custom movement logic for grades 6-8
* *player\_move\_v2(self, player)*

5.4 State Diagram

A screenshot of a computer screen

Description automatically generated

**Start State:** It's the beginning of the state machine.

**Initial Screen:** The state where the player can start a new game or load an existing one.

**Game Setup:** Where the player selects the grid size, and number of players, and places them on the grid.

**Gameplay:** Represents the state where the game is in progress.

**Players Meet:** Triggered when two players land on the same cell.

**End Game:** The state that represents the end of the game session.

5.5 Activity Diagram

The Activity Diagram in Section 5.5 illustrates the overarching behavior of the "Wandering in the Woods" Game. The process initiates with the user launching the application on their system. The user chooses the desired grade level for gameplay. Following this selection, the user sets various parameters tailored to the grade or based on experimental preferences, after the game begins. The gameplay proceeds through numerous steps, continuously checking for instances where players occupy the same square. Upon fulfilling this condition, the activity concludes.A diagram of a game

Description automatically generated

5.6 Customer Journey Diagram

The illustration below shows the customer journey at a high level. The customer or this case student begins the journey by starting the application. They choose a grade level between k-8. Depending on the grade level of the student parameters might become available for customization. The student then begins the game. The game plays out its game logic and checks for conditions that finish the game. If conditions are met, the journey ends. The line on the far right with one bubble is when a customer begins the application but chooses to not play a game.

A purple and white diagram

Description automatically generated

5.7 Personas

The Wandering in the Woods game targets school-age children from kindergarten to eighth grade, catering to a wide range of educational interests and learning preferences. The game also encourages experimentation and an interactive learning experience.

A group of black text and blue and green images

Description automatically generated with medium confidence

5.6 Game UI Visualization

A screenshot of a computer error

Description automatically generated