PROJECT

SUB:SOFTWARE ENGINEERING

TITLE:WONDERING IN THE WOODS GAME DEVELOPMENT

GROUP:

1. NAVEEN KUMAR MARRI(L30080893)
2. GOPI KRISHNA PERLA(L30081625)
3. VYSHNAVI DUPPEKAR(L30087742)
4. SYED ZEESAN NUSRATHI(L30079935)

**Table of Contents**

[1. Introduction 3](#_Toc148738049)

[Purpose 3](#_Toc148738050)

[Game specification 3](#_Toc148738051)

[2. Process model 3](#_Toc148738052)

[3. Use case 4](#_Toc148738053)

[Use case 1 – Grade K-2 4](#_Toc148738054)

[Use case 2 – Grade 3-5 4](#_Toc148738055)

[Use case 3 – Grade 6-8 students 5](#_Toc148738056)

[4. UML model 5](#_Toc148738057)

[Use case diagram 5](#_Toc148738058)

[Deployment diagram 6](#_Toc148738059)

[Class diagram 7](#_Toc148738060)

[State diagram 7](#_Toc148738061)

[Activity diagram 8](#_Toc148738062)

[5. Personas 9](#_Toc148738063)

[6. GUI Mock-up 10](#_Toc148738064)

[Home screen Grade K-2 10](#_Toc148738065)

[Home screen Grade 3-5 11](#_Toc148738066)

[Home screen Grades 6-8 12](#_Toc148738067)

[7. Testing strategy 13](#_Toc148738068)

[8. User manual 14](#_Toc148738069)

[9. References 25](#_Toc148738070)

# Introduction

## Purpose

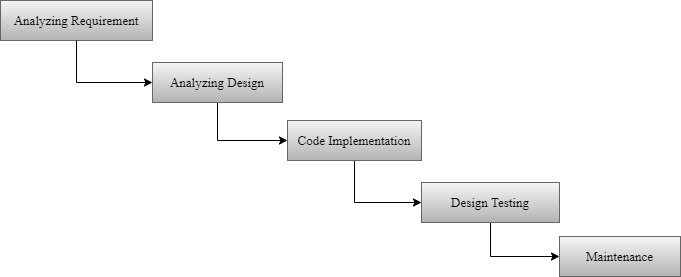
The report describes Wandering in the Woods game development which happened using Java swing programming language. The reason behind the game development is to help the K-8 students learn about computation, math concepts, computer science, and computational thinking. The game has been designed in three levels with simple visualization to make the students see the choices of development. Each level has its challenges, to make the students work with local and big data and gather some ideas related to big data, graph generation, and so on. Finally, the game is played with a pair of students over the screen.

## Game specification

The Wandering in the Wood game was initially designed to be played by two players in level 1. The game consists of three stages the first stage is for the K-2 students and is played with two players. Initially, the forest was set in a square shape where two players could not see each other. In any case, when the player meets in the same grid, the game ends. The player needs to play until they don’t meet each other. After completing level 1, the next level will be played which is for grade 3-5 students. The specified feature in this level is that students can select their grid and decide their starting time. In this level, the students can view their total runs without any disturbance, time taken, and overall game score after the game ends. In the second level, the students have no other option where they can select the size of the grid and can simulate the game. The final stage is for grade 6-8 students which has been designed more complex than the other two versions. In this stage up to 8 players can play the game which helps them to determine various information and explore different protocols. With the Java swing, the best gaming environment has been designed to enhance the player's interest which tends to have several features to develop a successful gaming environment.

# Process model

The Waterfall model has been used for the Wandering in the Woods game, which offers creating the program using a simple method. Figure 1, describes the flow of the diagram which has been made using Java Swing, where all the gaming processes are done in a sequential flow to make successful development without making each stage of development overlap. Each of the stages of the gaming development is based on the given requirements, which helps the students to understand each stage of the gaming process.



**Figure 1 – Flow diagram for Wandering in the Woods game**

# Use case

## Use case 1 – Grade K-2

**Table 1 – Use case for Grade K-2 students**

|  |  |
| --- | --- |
| **Primary actor** | Player |
| **Precondition** | The game created for two students |
| **Description** | The first level has been created for K-2 students using the Java program. Here the student needs to enter several columns and the grids. After the students select the column and grid details, the student can start the game by clicking on the ok button. Each of the players will be in the opposite direction, after the game starts the program counts each of the player's moves and displays the count after the program ends. |
| **Acceptance criteria** | Level 1 is for K-2 students and can be played using two players. |

## Use case 2 – Grade 3-5

**Table 2 – Use case for Grade K-2 students**

|  |  |
| --- | --- |
| **Primary actor** | Player |
| **Precondition** | Four players have been involved in the level 2 stage |
| **Description** | The simulation was created for Grades 3-5, where four players were involved in the defined stage. This stage allows students to select of total number of rows, columns, and the grid. After the game gets started by the player, each of their moves gets counted. When the player meets each other, the game ends. The average move made between the players gets counted along with displaying the total number of moves made by each of the players. Even the player who wins the game without meeting another player gets displayed. |
| **Acceptance criteria** | The total number of players for this game increased. |

## Use case 3 – Grade 6-8 students

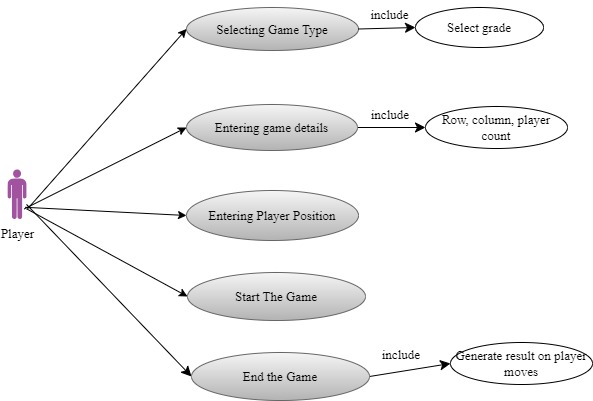
**Table 3 – Use case for Grade K-2 students**

|  |  |
| --- | --- |
| Primary actor | Player |
| Precondition | The game created for 8 players |
| Description | In stage 3, the Wandering in the Wood game was created for 8 players to start the game by the players, they need to select the grade which they need to play followed by entering the total number of rows, columns, gird, and player count. Longer time to play the game where the number of moves taken by each player along with the average count taken among the players gets counted and displayed. After the game ends, the result would be displayed stating the average move made, the player who wins the game without meeting other players. |
| Acceptance criteria | This level has been designed by increasing the player count to play the game. |

# UML model

## Use case diagram

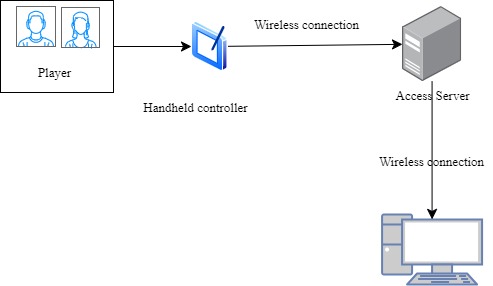
The developed use case diagram in Figure 2 describes the information that has been gathered with consideration to all the internal and external influences of the Wandering in the Woods game [1]. Each use case specifies a different action of the designed game where the players need to start the game by selecting the game type, selecting row and column details, and selecting the player position. After selecting all the details, the player can start the game.



**Figure 2 – Use case diagram**

## Deployment diagram

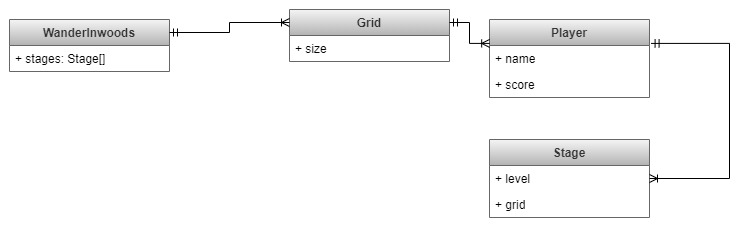
Servers, interface and different application been used in the Wandering in the Woods game, where the server used for the software designing plays an important role in the game and further offers system control and overall controls of the players used for the game to perform their task in user-friendly way.



**Figure 3 – Deployment diagram**

## Class diagram

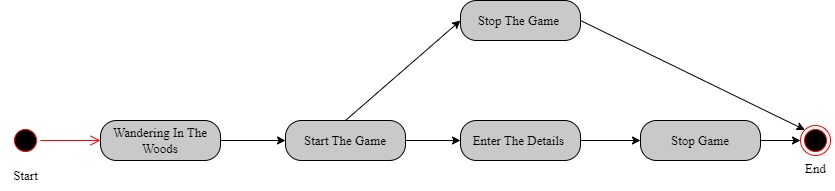
This diagram is the unified modeling language that clearly describes the entire structure of the system describing their class, attributes, operation, and the relationship among the objects. This diagram when used for the Wandering in the Woods project represents the various classes used for developing the program. Each grade class specified in Figure 4 has several rows, columns, and player counts.



**Figure 4 – Class diagram**

## State diagram

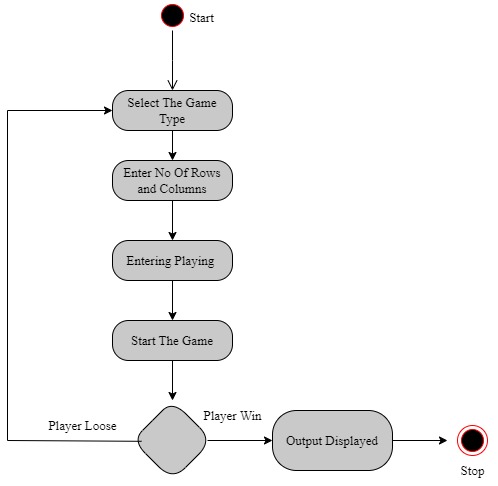
This diagram has also been defined as the state machine diagram which is one of the types of behavioral diagrams in UML showing the transition between several objects [2]. Using this diagram for the Wandering in the Woods game would help in virtually determining its operation. From the developed design, initially, the player would start the game and specify the number of rows, columns, grid, and player. After the necessary selection, the game starts and effectively ends by displaying the output.



**Figure 5 – State diagram**

## Activity diagram

The activity diagram for the Wandering in the Woods game would define the dynamic aspects of the system [3]. All the processes done by the player from the initial start of the process to the result have been clearly described. By starting the game, the player needs to specify the details and play the game. At the final stage when the game ends, the result will be generated stating the number of moves, average moves, and players with no meeting. If any player needs to start the game again, it can be done after the existing game is completed.



**Figure 6 – Activity diagram**

# Personas

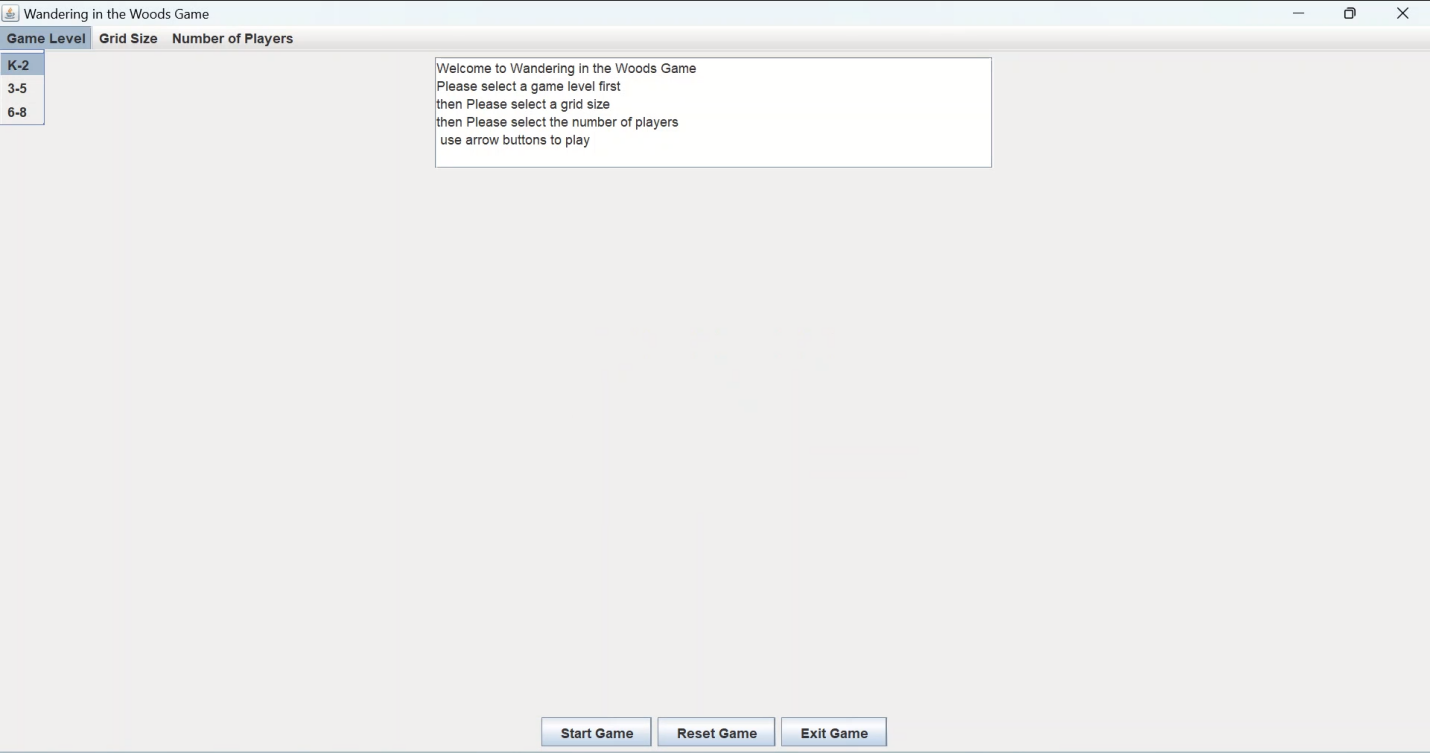
The designed Wandering in the Woods game has a maximum of 8 players to play the game. Based on the level the total number of players to play the game gets changed. Initially for the Grade K-2, the game design allows two players to play the game; further grade 3-5 allows 4 players to play the game and finally Grade 6-8 allow 8 players to play a game at a time. Based on the selection of level, the determined players would be allowed to play a game. **For example:** when the player selects level 2, then they are allowed to enter the total number of players and, the total number of rows and columns. Finally, after specifying the game, the start button gets selected to start the game.

# GUI Mock-up

The developed GUI Mock-up happened with consideration to Wandering in the Woods gaming environment. The designed first window would make the players initially select the grades followed by allowing the players to select on total number of rows and columns that are to be generated. After selecting the gaming details, the player needs to start the game. After the game ends, the output would be generated stating the total moves made by each player, and the average moves among the players. Based on the result generated from the game, the size of the window would vary.

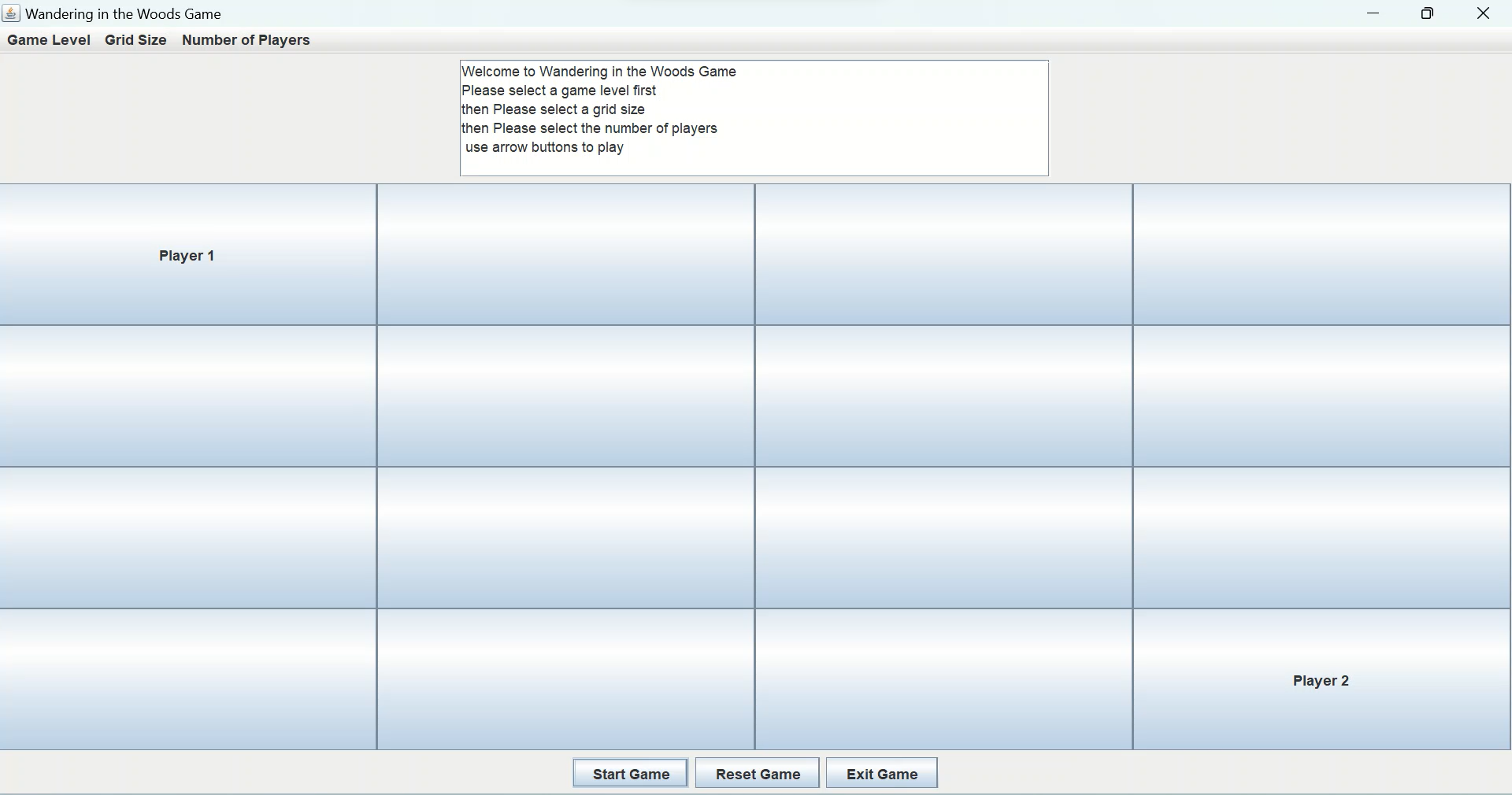
## Home screen Grade K-2

Figure 7 defines the information stated on the home screen, where players will be allowed to select the desired grade they need to play.



**Figure 7 – Selecting Level**

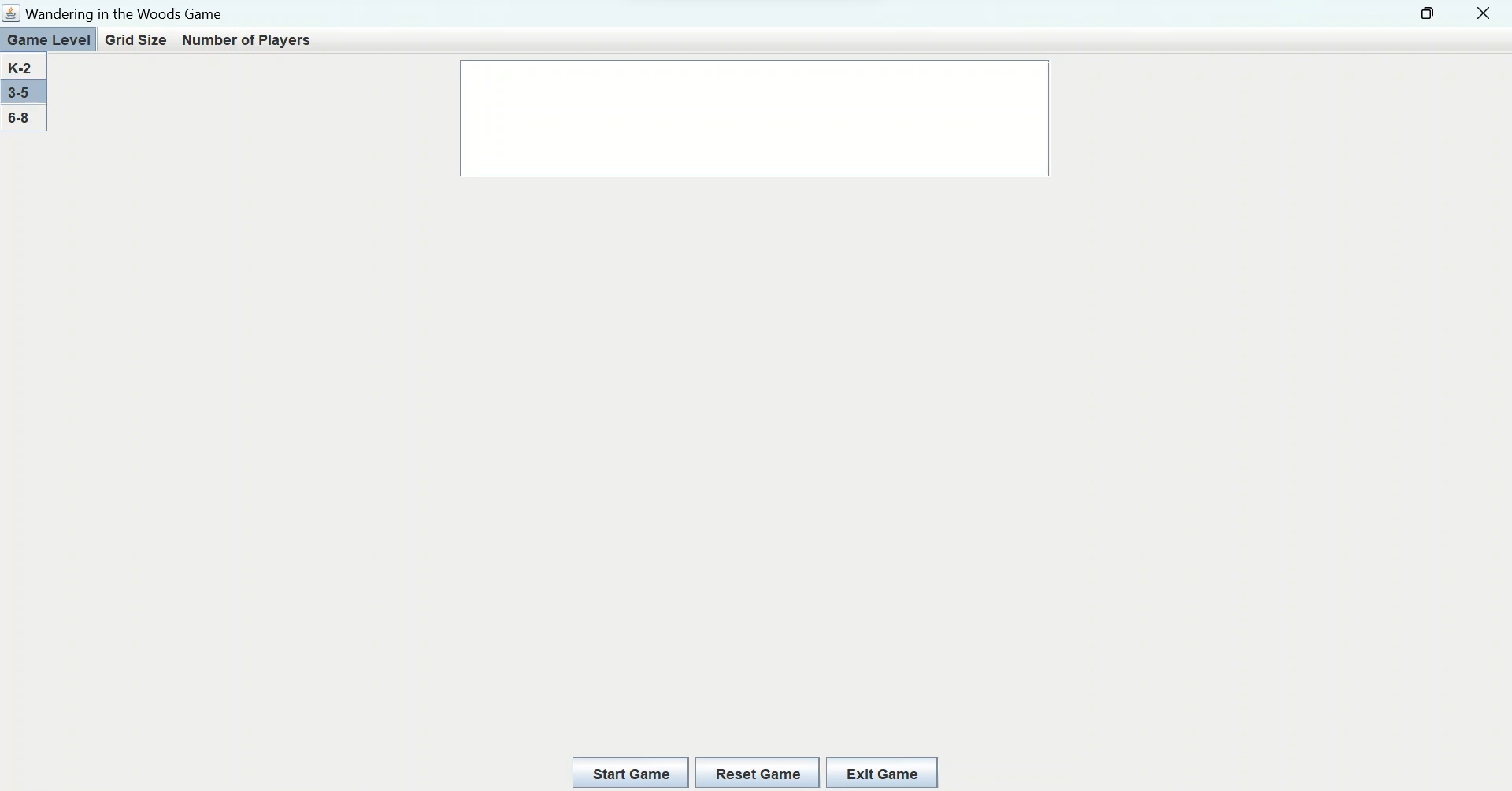
The below figure describes the details of the game the player needs to specify to start the game. In any case when invalid details have been specified by the player then an error message gets displayed where the user corrects moves during the game and the result gets displayed as the final output.



**Figure 8 – Details of K-2 level game**

## Home screen Grade 3-5

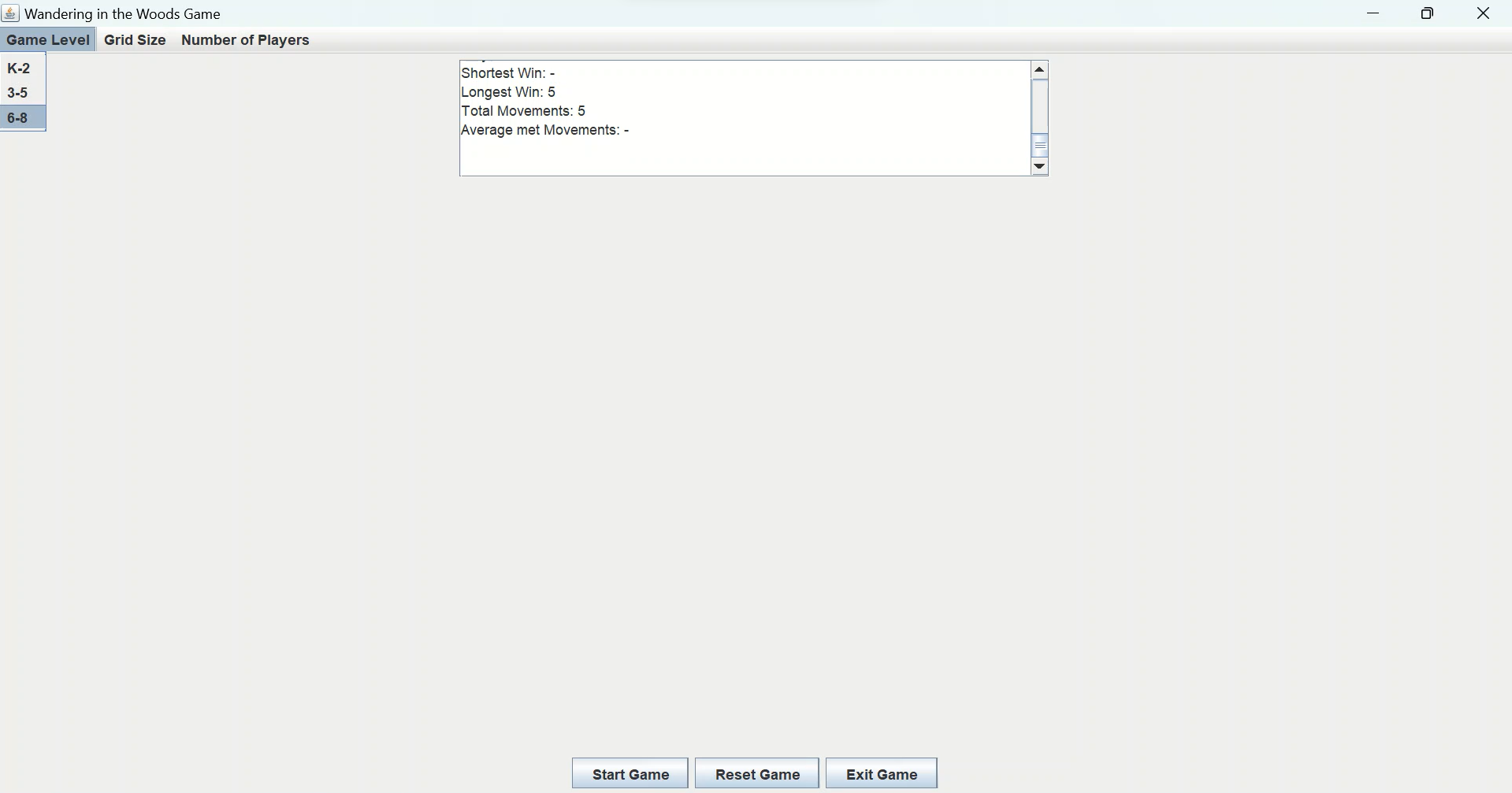
The player in this level needs to select the specified grade followed by selecting the number of players to start the game along with a total number of rows and columns.



**Figure 9 – Grade 3-5 home screen details**

## Home screen Grades 6-8

When the player selects the specified grade, then the total rows, column, and player count to be specified. After entering the details, the player can start the game.



**Figure 10 – Grade 6-8 home screen details**

Figure 11 defines the screen generated for the players to start the game with the specified details on the home screen.



**Figure 11 – Game window**

# Testing strategy

Testing strategy for the Wandering in the Woods gaming software to satisfy all the requirements stated. Effective testing strategy when considered would help in getting the answer to the question of what is to be tested and when the outcome of the testing has been undergone. This strategy is the most commonly used testing technique where the software tester plays a major role in determining suitable testing strategies for the software being developed and helps in finding possible errors and solutions to overcome them. Some of the types of testing strategies that are to be considered for the Wandering in the Woods gaming software are,

**Unit testing**

This testing is determined to be the smallest collection of code where this is one level of testing and been processed together to make the final result. This testing strategy has been determined as White box testing where the testing is determined with consideration to stated gaming requirements [4]. In simple this is determined as the single module, unit, or component with the gaming software being developed.

**Integration testing**

This type of testing strategy would help in constructing the gaming program for wandering in the woods as well as conducting the testing of the software being developed to identify possible errors associated with the interface. With this testing, the defects in the interface get determined along with identifying the interface of the components, where the major objective is to undergo the testing after the unit testing has happened.

**System testing**

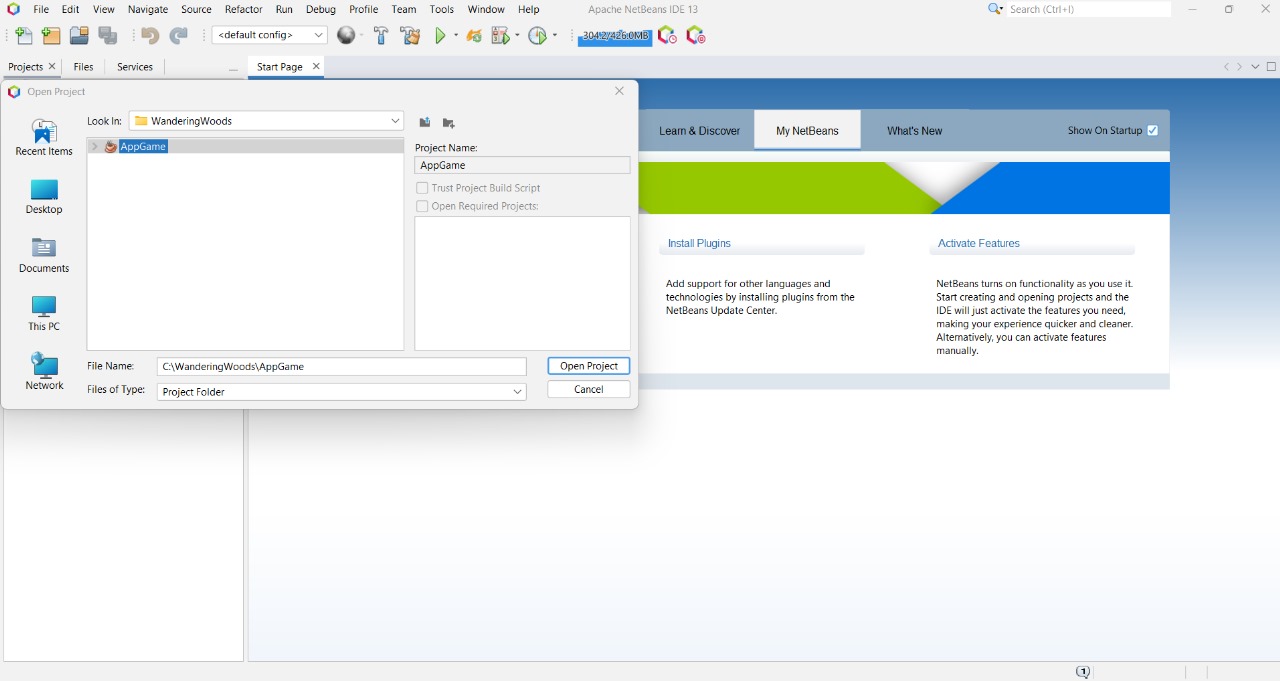
This testing has been processed over the entirely developed Wandering in the Woods gaming software which has further been integrated into the system compliance with all the stated requirements. This testing falls under the scope of Black box testing where the tester doesn’t need any internal knowledge of the concept of the gaming system which includes functional and non-functional requirements [5]. In this case, the specified testing was determined to process a series of tests to evaluate the entire process and determine the integration of every system and its overall functionality.

**Performance testing**

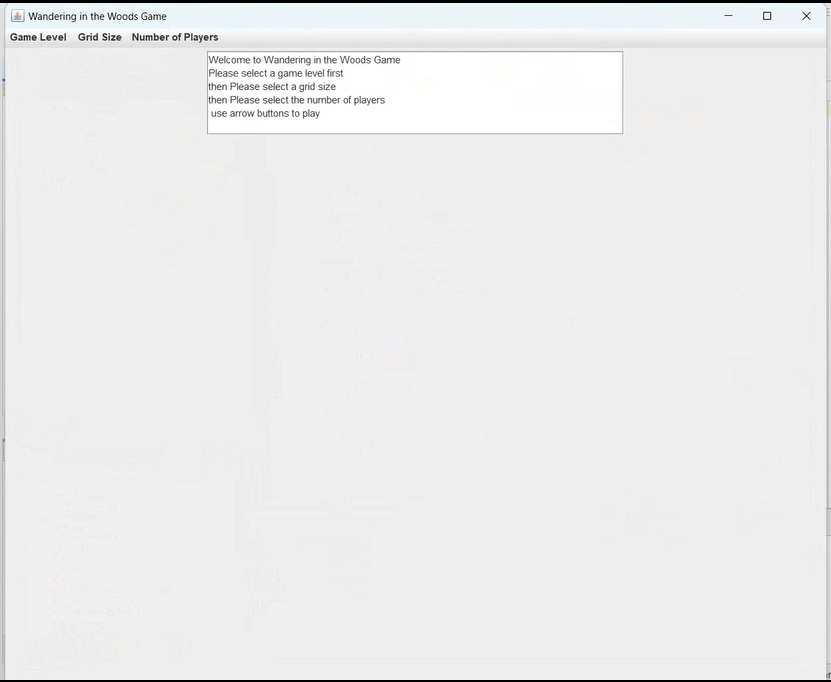
This testing when determined for the Wandering in the Woods gaming software would help in the identification of the overall speed in the gaming performance and further would help in undergoing the testing to improve the system speed, scalability, overall response time, reliability and their performance used for the playing game [6]. The overall response among the players with the developed gaming software gets identified and helps in improving the features to process effective responses.

# User manual

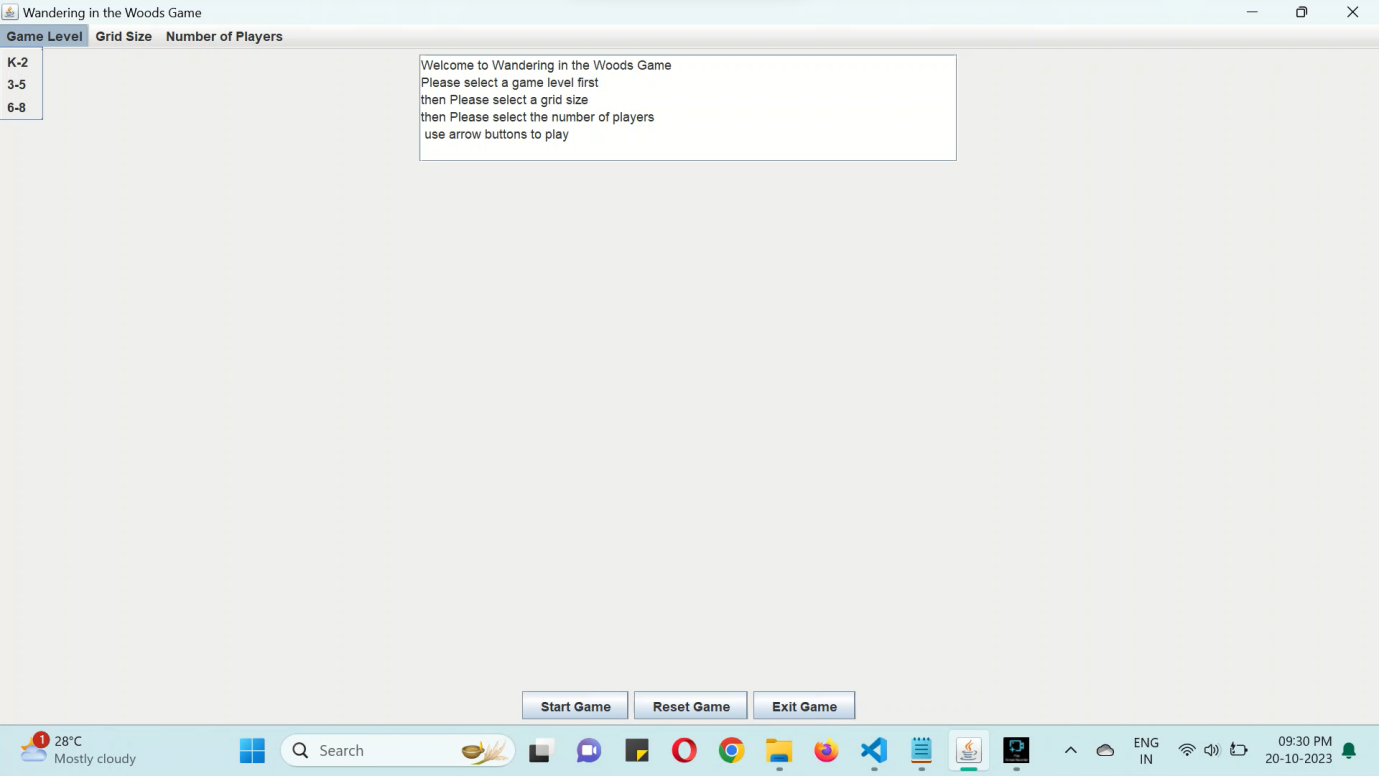
**Step 1 –** The first step is to open the NetBeans tool for importing the Wandering in the Woods App game file.



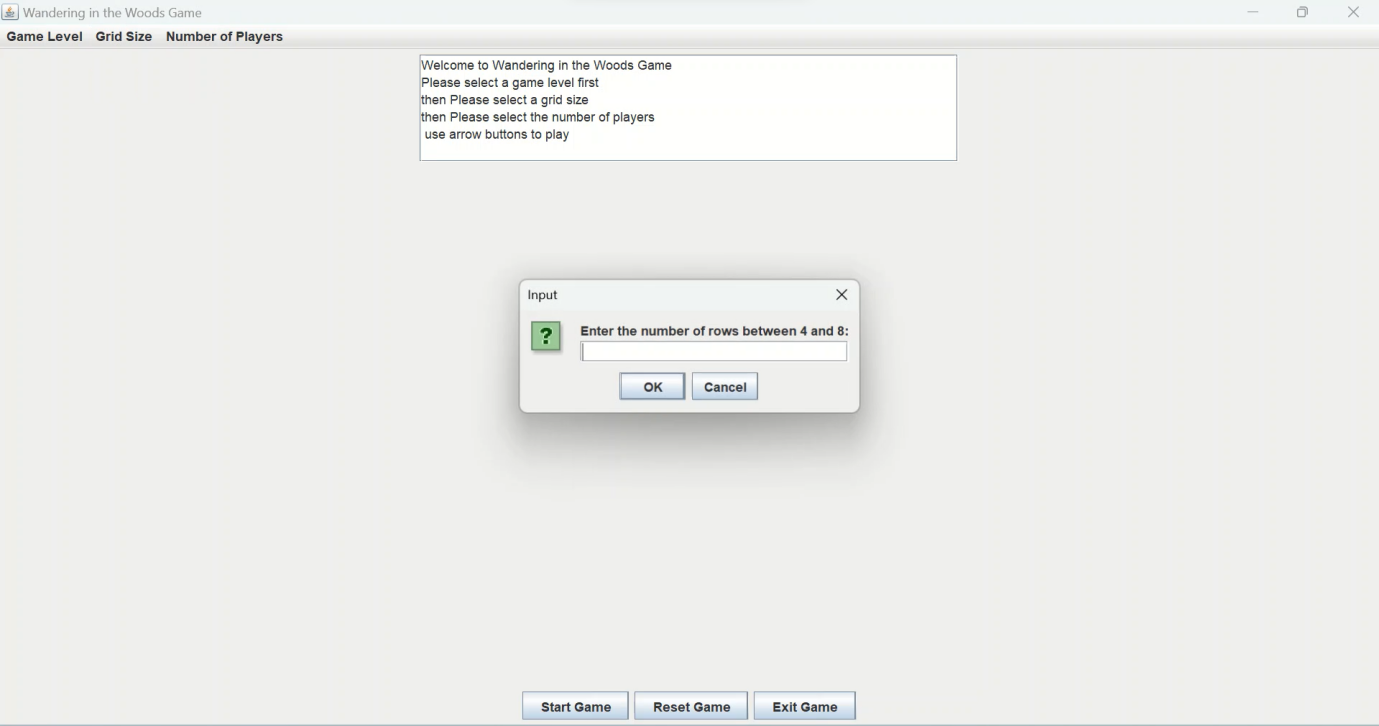
**Step 2 –** After importing the file, the file option to be selected to open the gaming project file.



**Step 3 –** Main method program to be executed for getting the grade selection window followed by selecting the desired grade type (Example: Selecting Grade K-2 option from grade level).



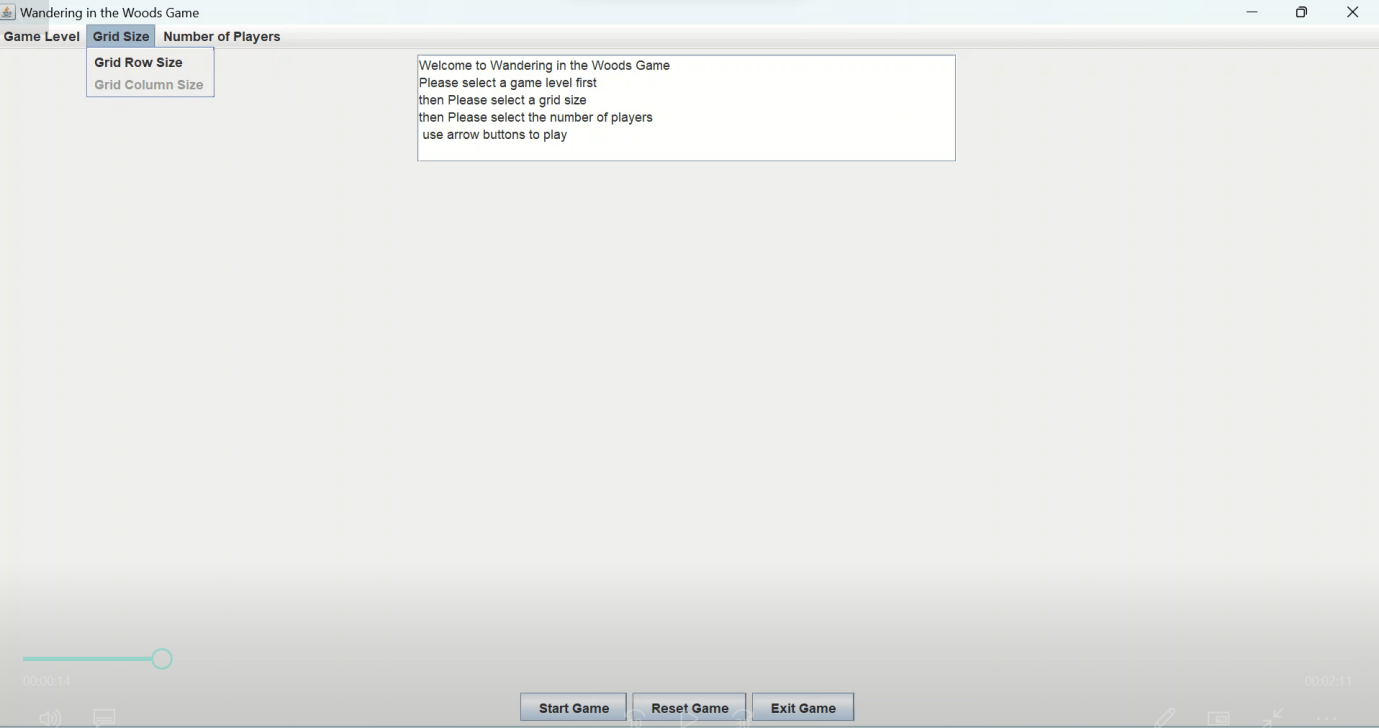
**Step 4 –** After selecting the grade, row for grid size is entered.



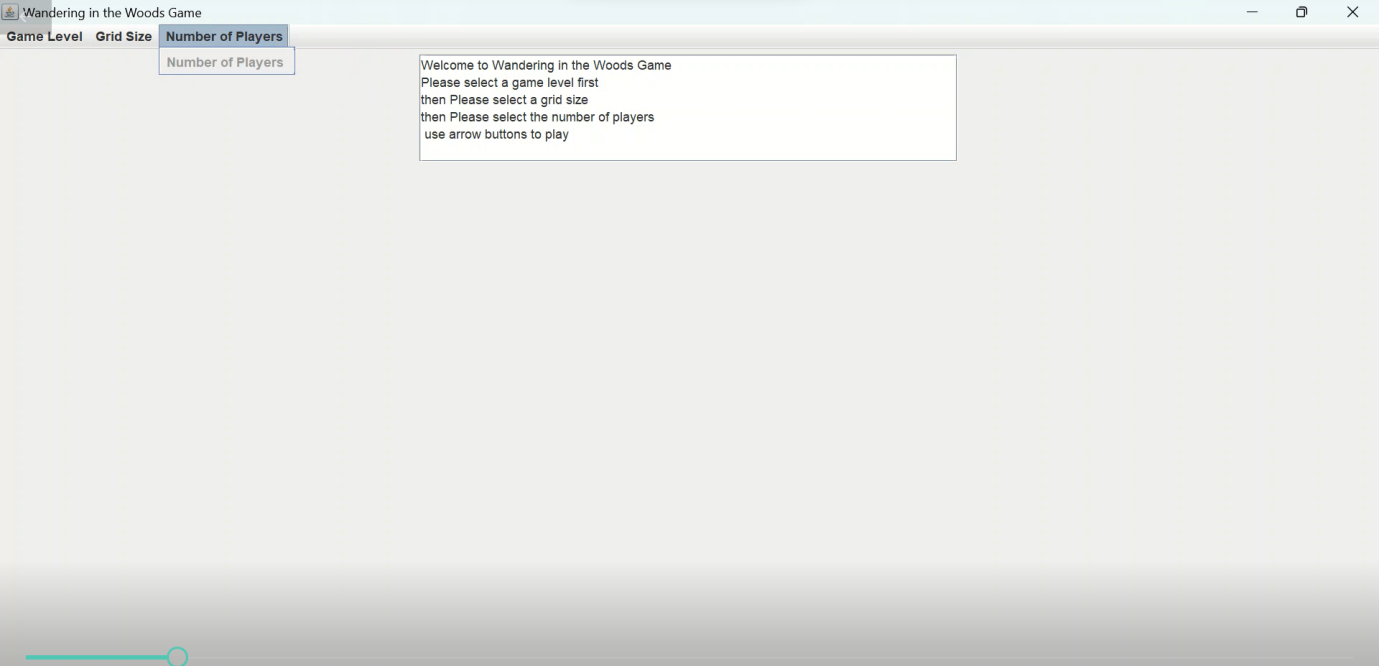
In any case, when the correct details are not specified, error message will be displayed.

**Step 5 –** Column will be disabled since its k-2 we have to keep square and disable column

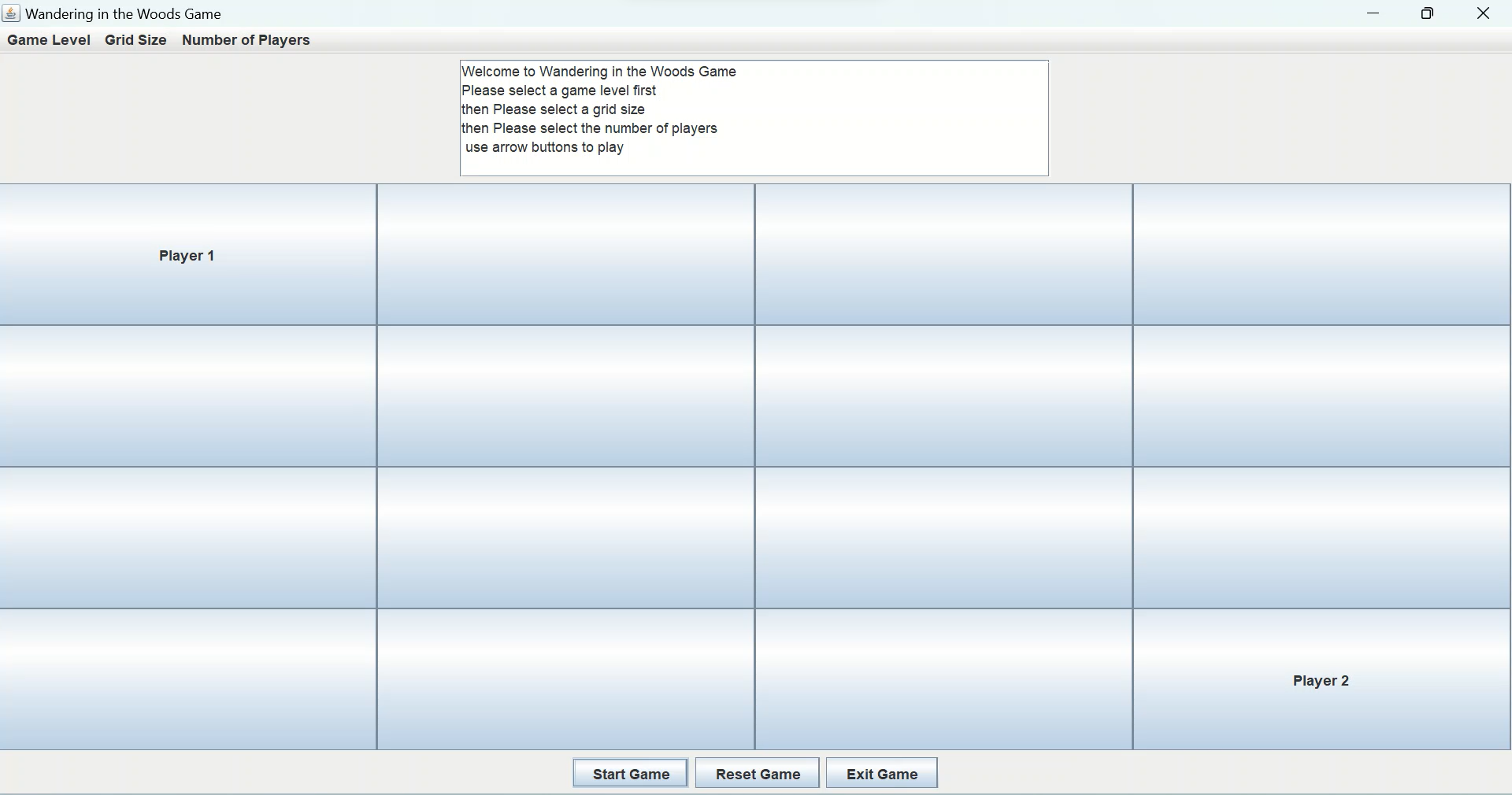
the gaming window will be generated along with selected players.



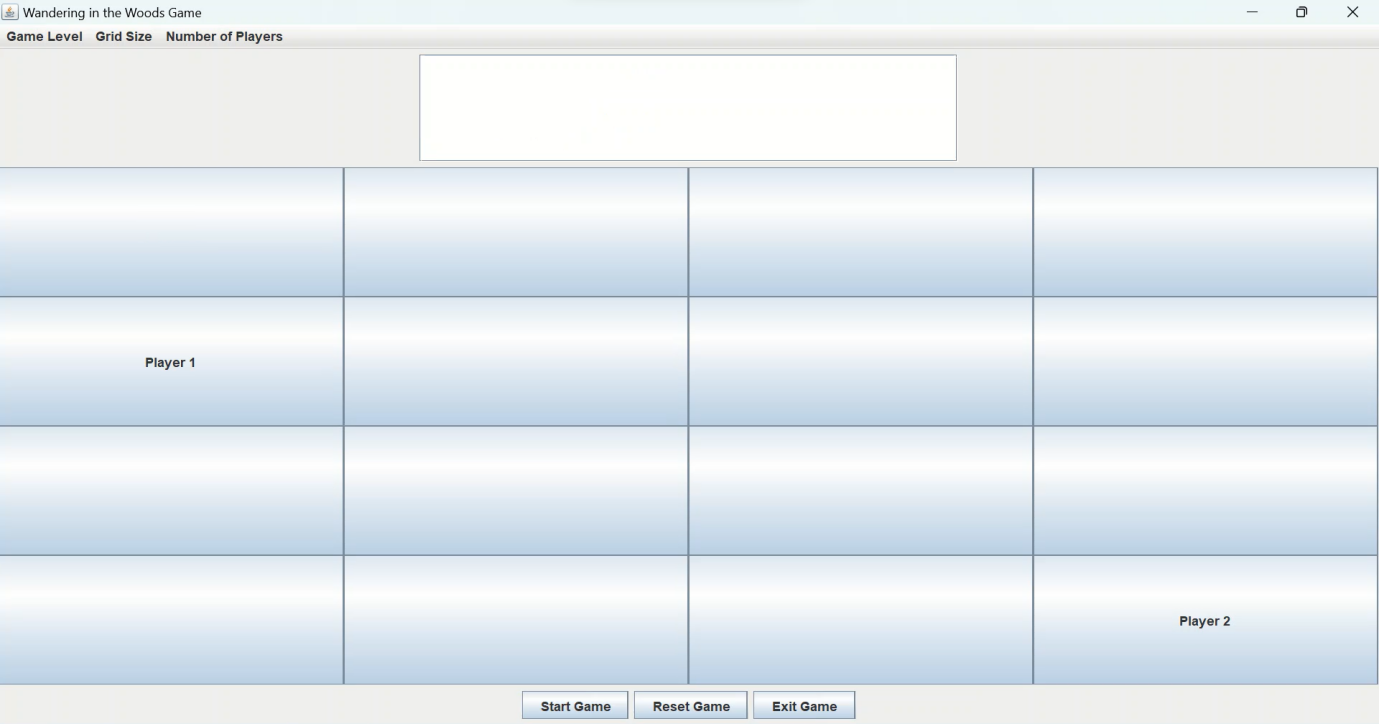
**Step 6 -** Now start the game since player addition also disabled default is 2



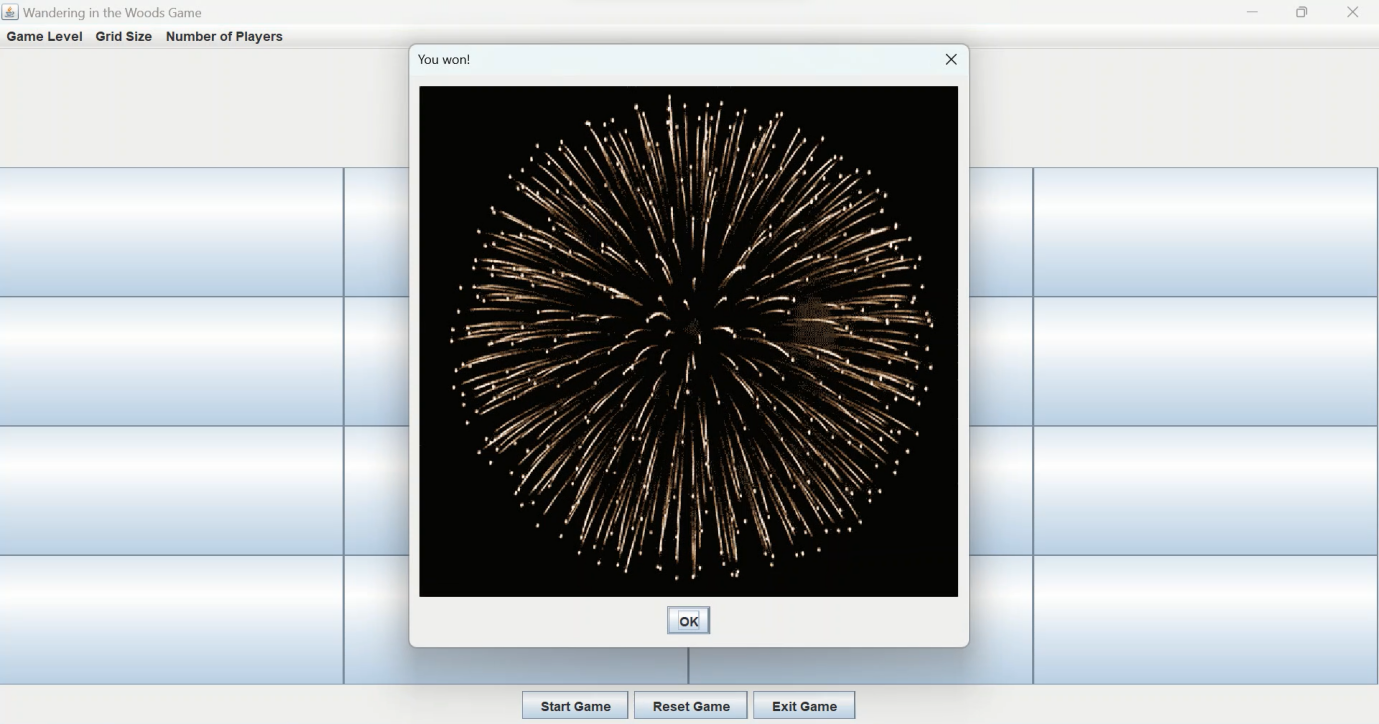
**Step 7 –** Now the game starts with below window with two players.



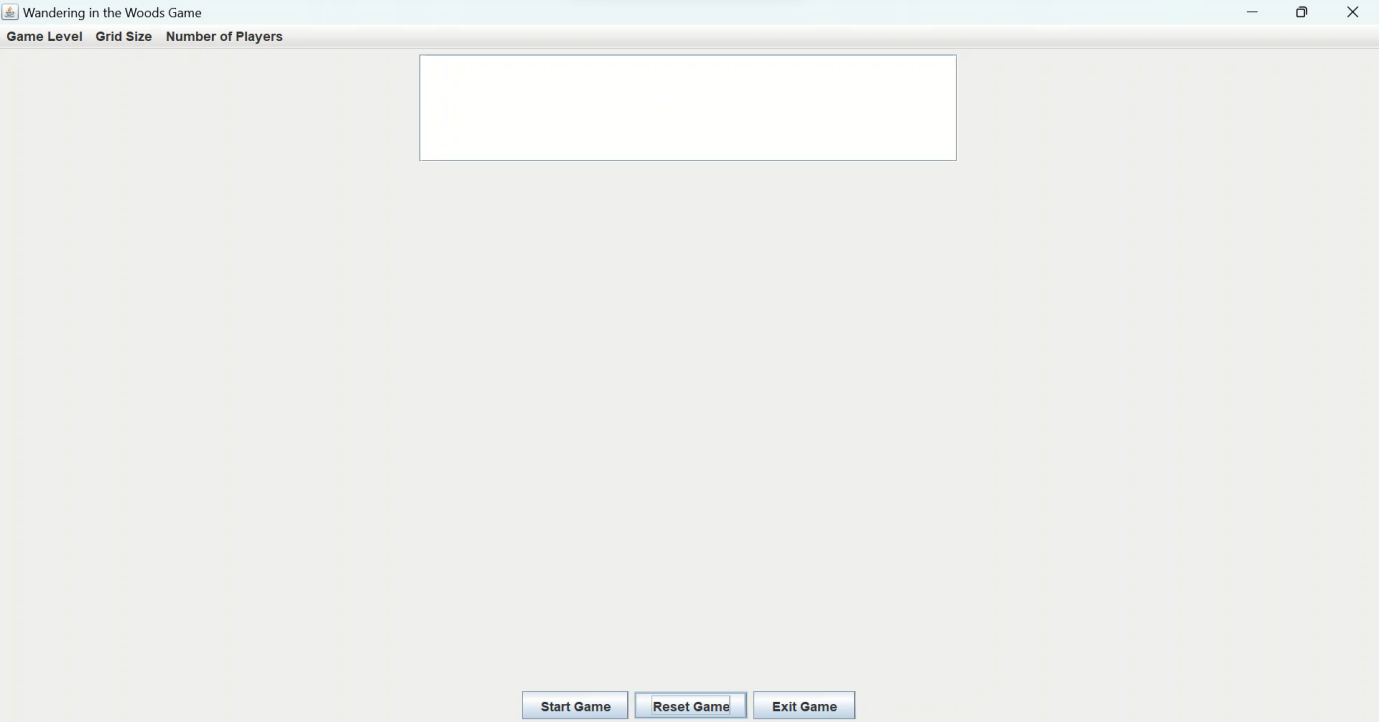
**Step 8 -** Arrow keys are used to move the players from one grid to another.



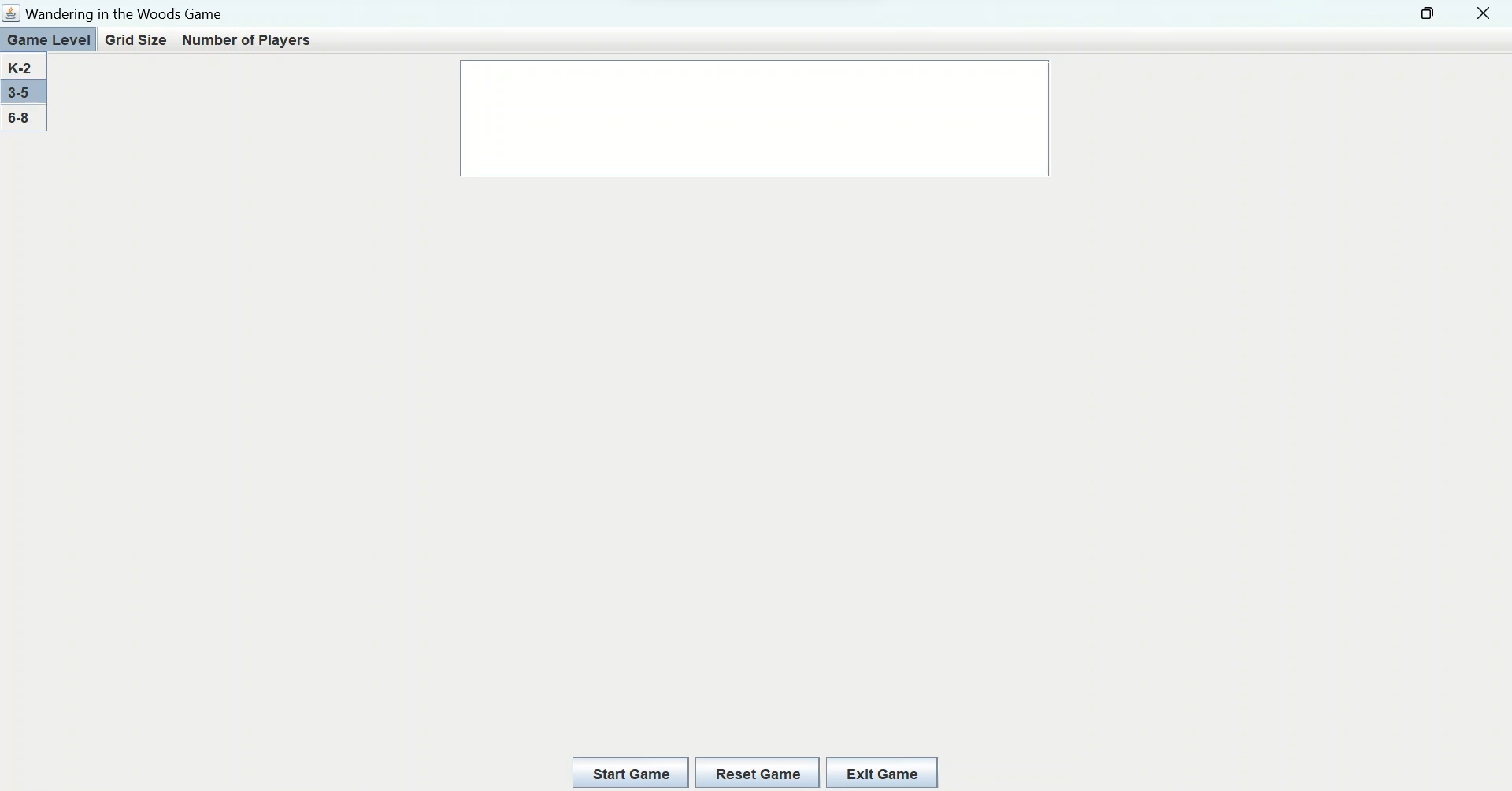
**Step 9-**  When meeting happens between the player results gets displayed and celebration starts with the music



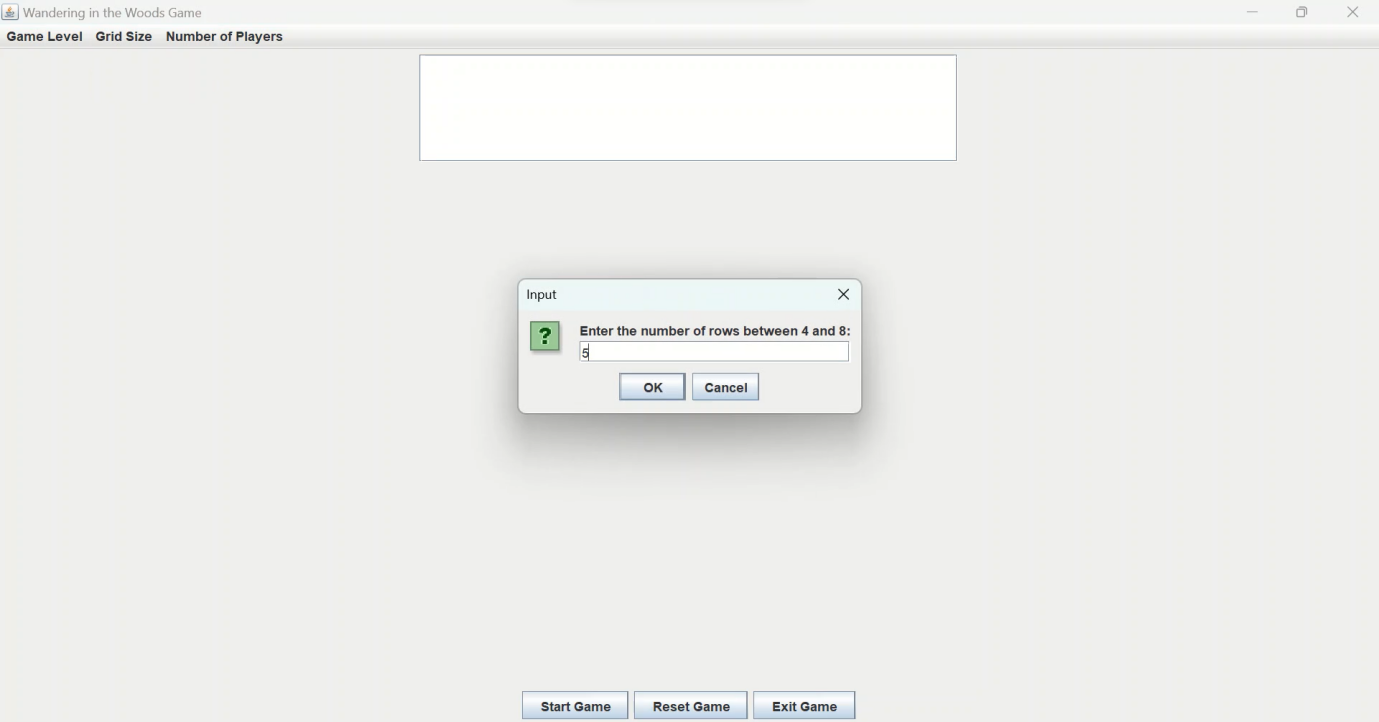
**Step 10 –** In order to move to the next grade level, a reset is required.

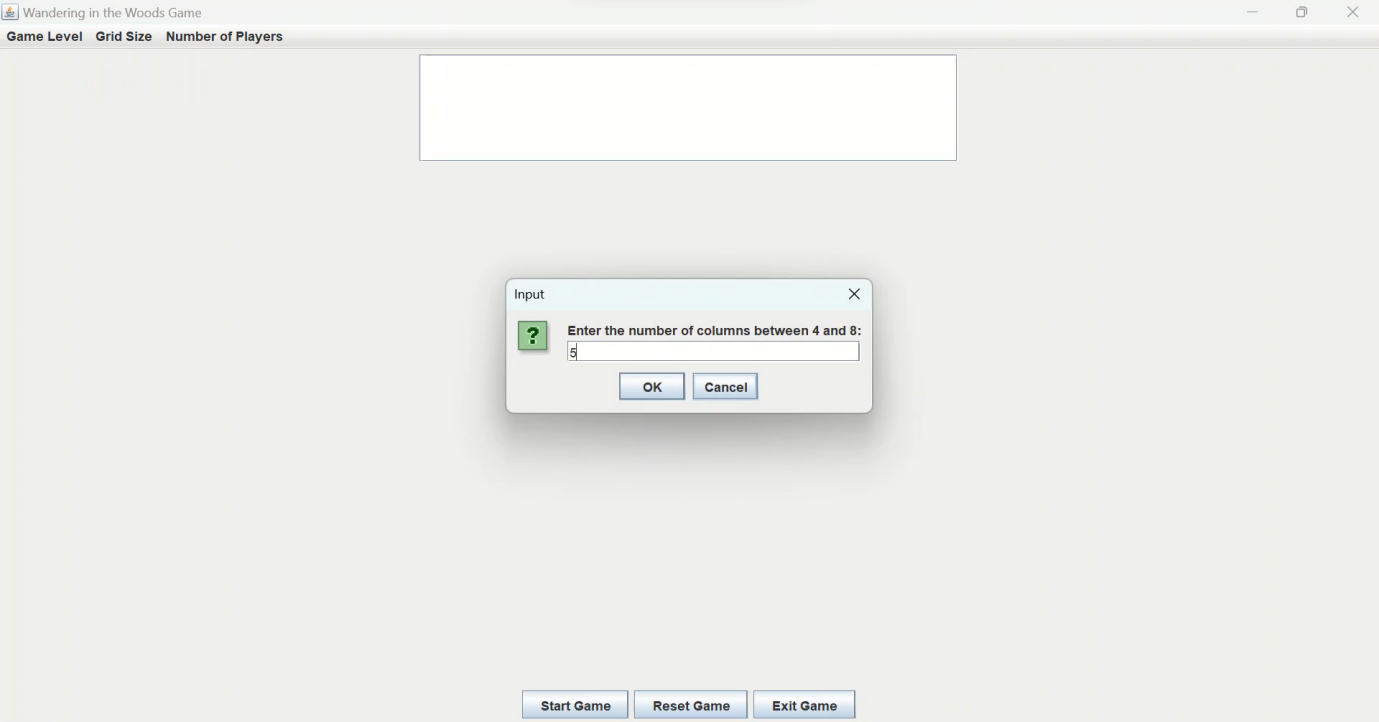


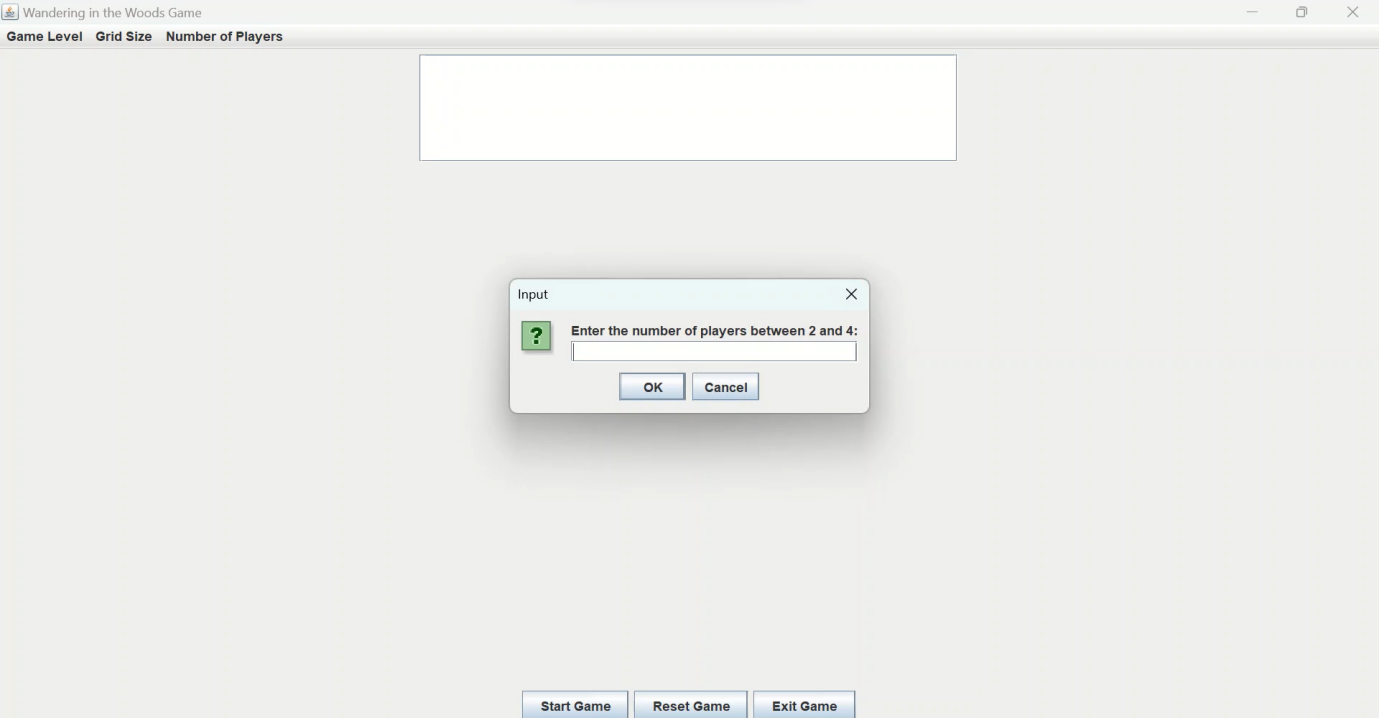
**Step 11** After successfully completing Grade k-2, the next level (Grade 3-5) gets selected.



**Step 12 –** Player need to specify, total number of rows, column, and the player count.



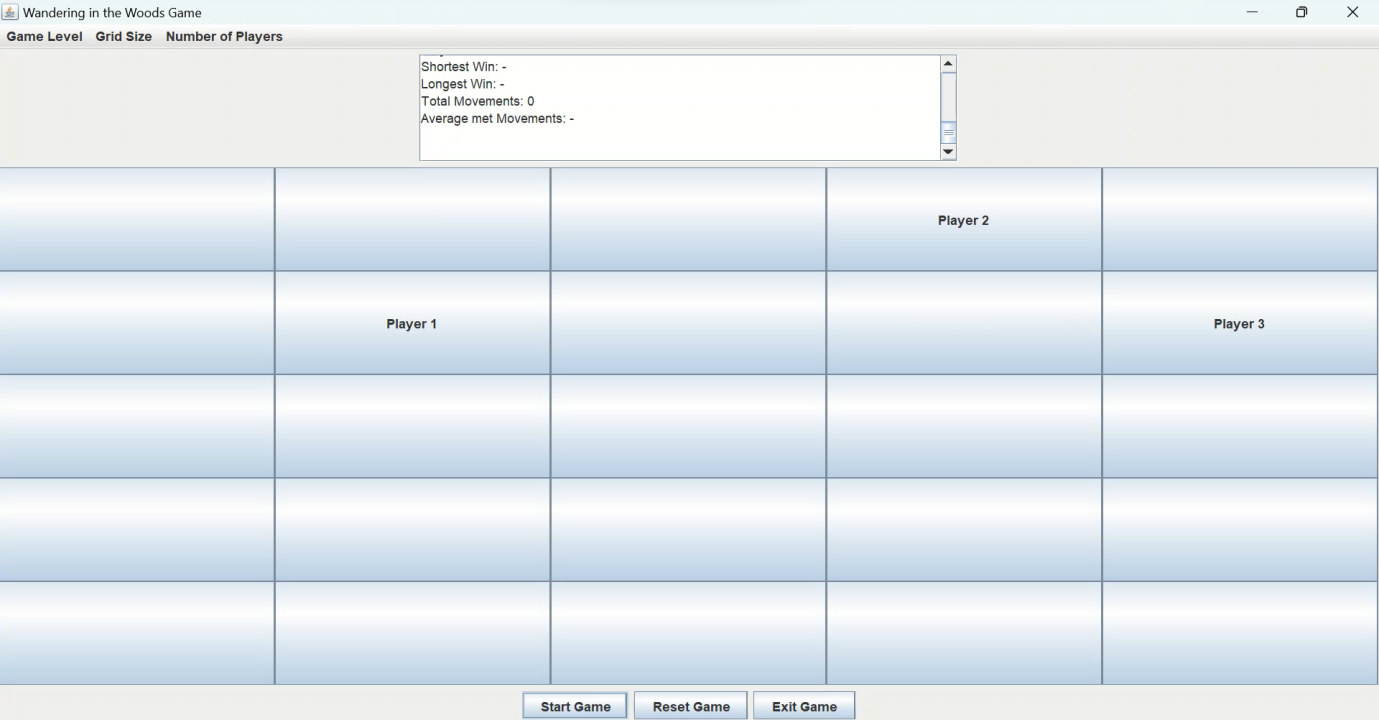




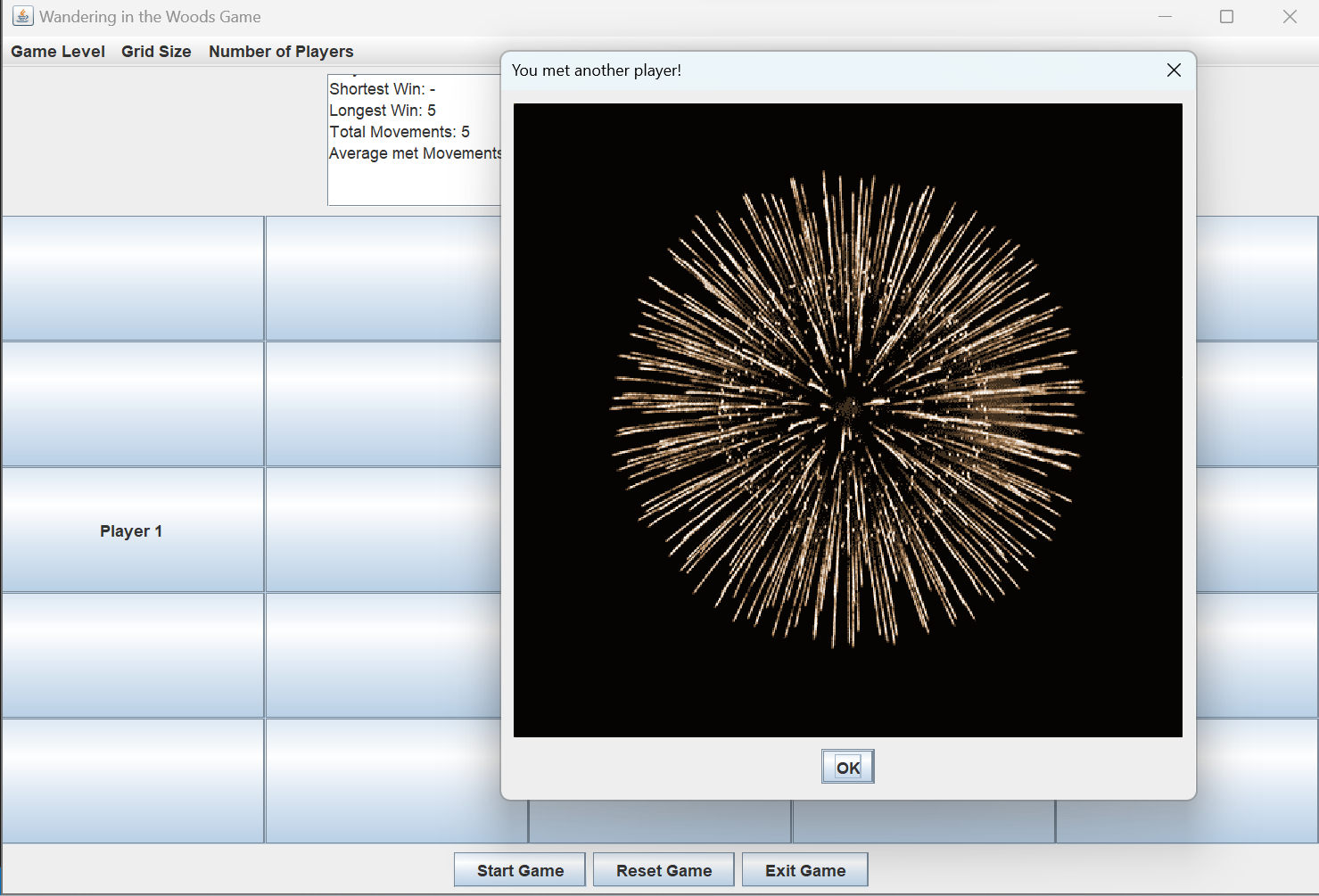
If incorrect number gets specified, error message will be popped up.

**Step 13 –** Based on the total player count, necessary selection the players position to be inserted.

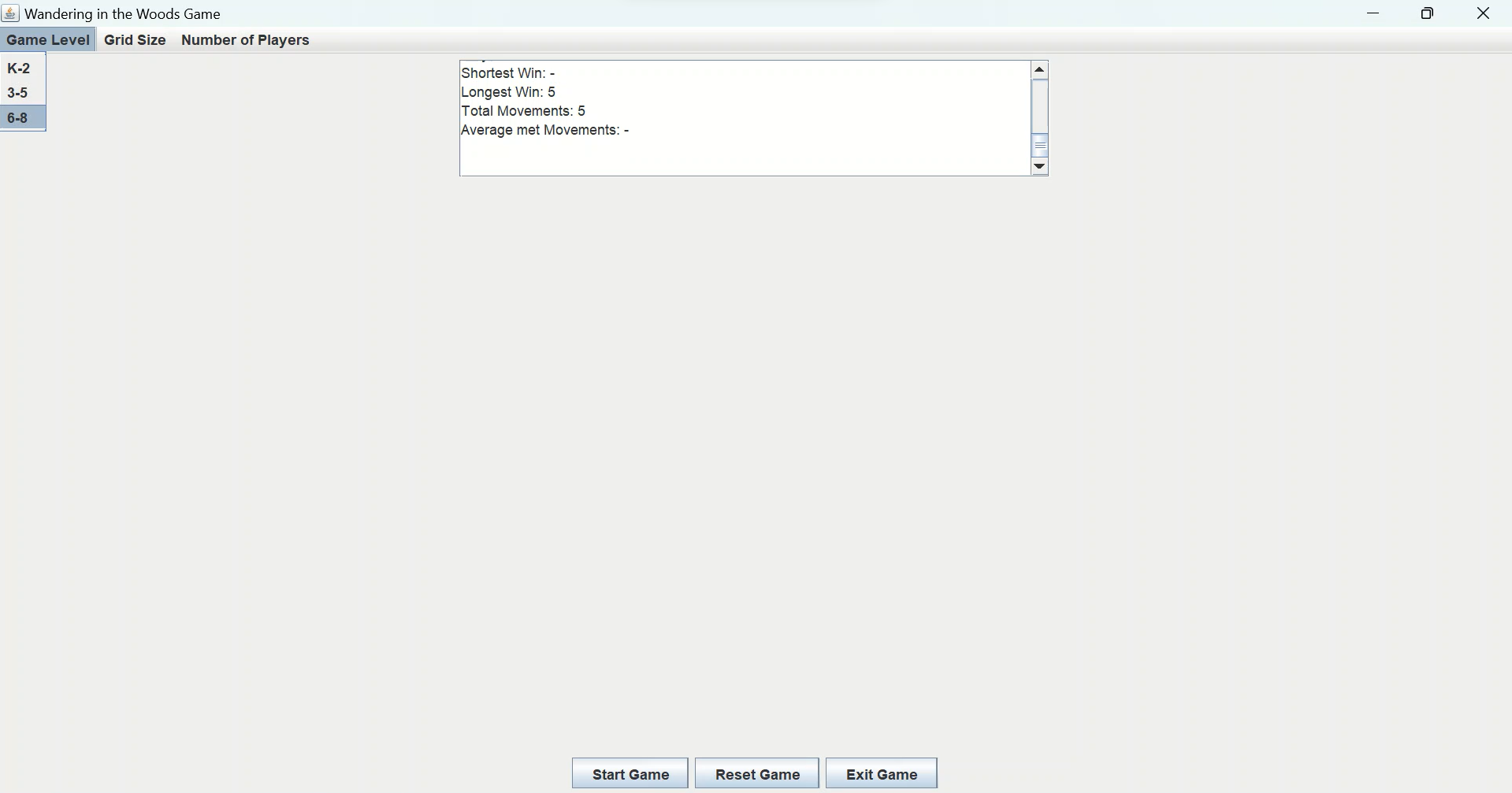
**Step 14 –** Gaming window appear with the selected player position to start the game.



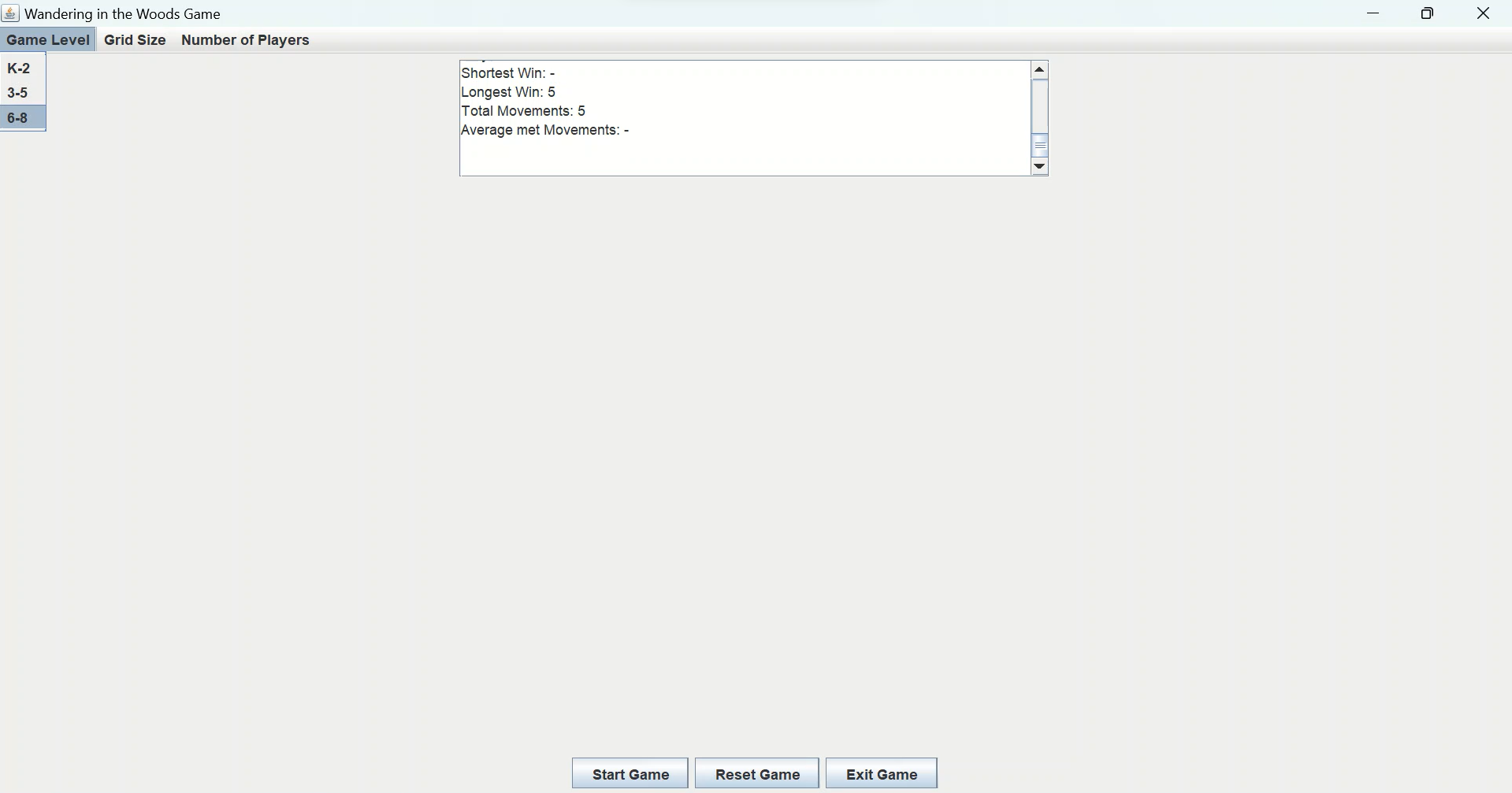
**Step 15 –** When players meet each other, the game would game followed by displaying the output with total moves made by each player and the overall statistics.



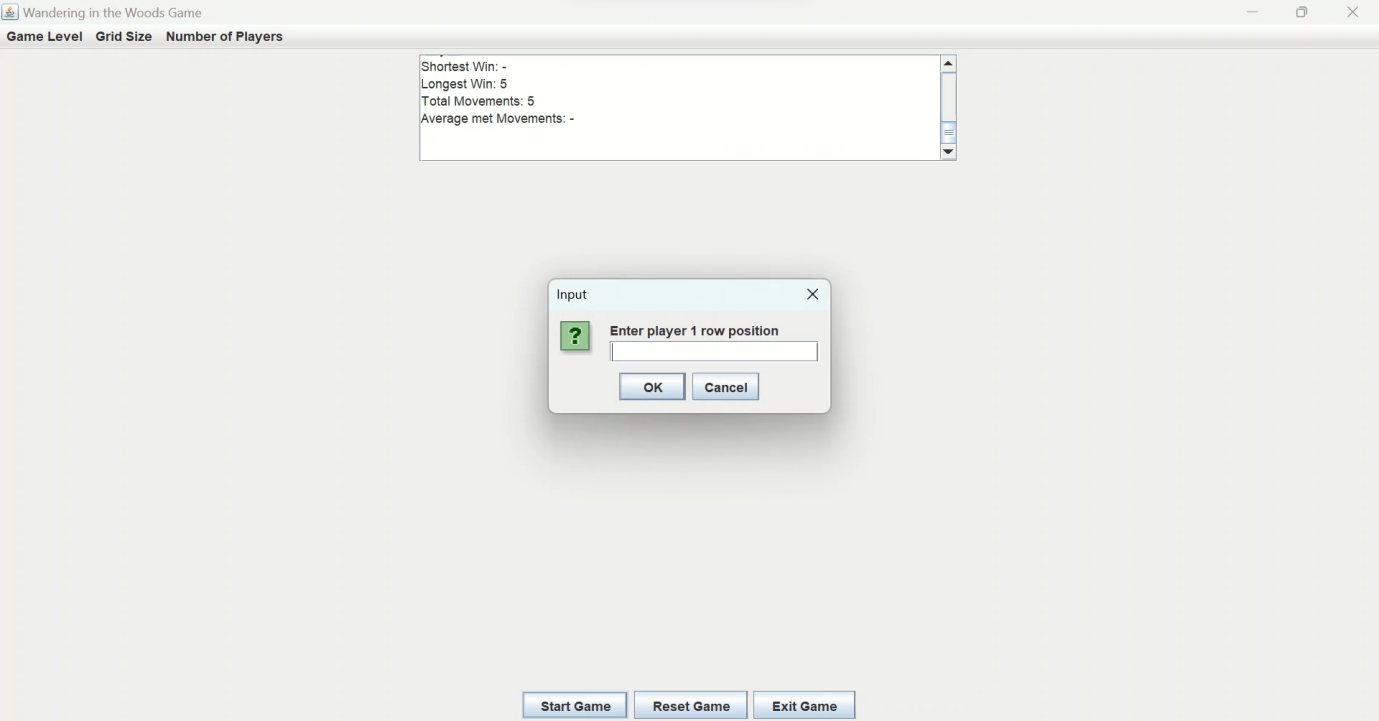
**Step 16 –** After successful completion to grade 3-5, the final level (Grade 6-8) need to get selected.



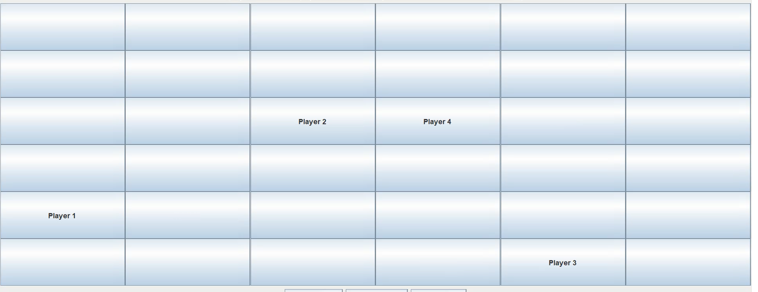
**Step 17 –** With the displayed window, total number of rows, columns, and player count to get specified.



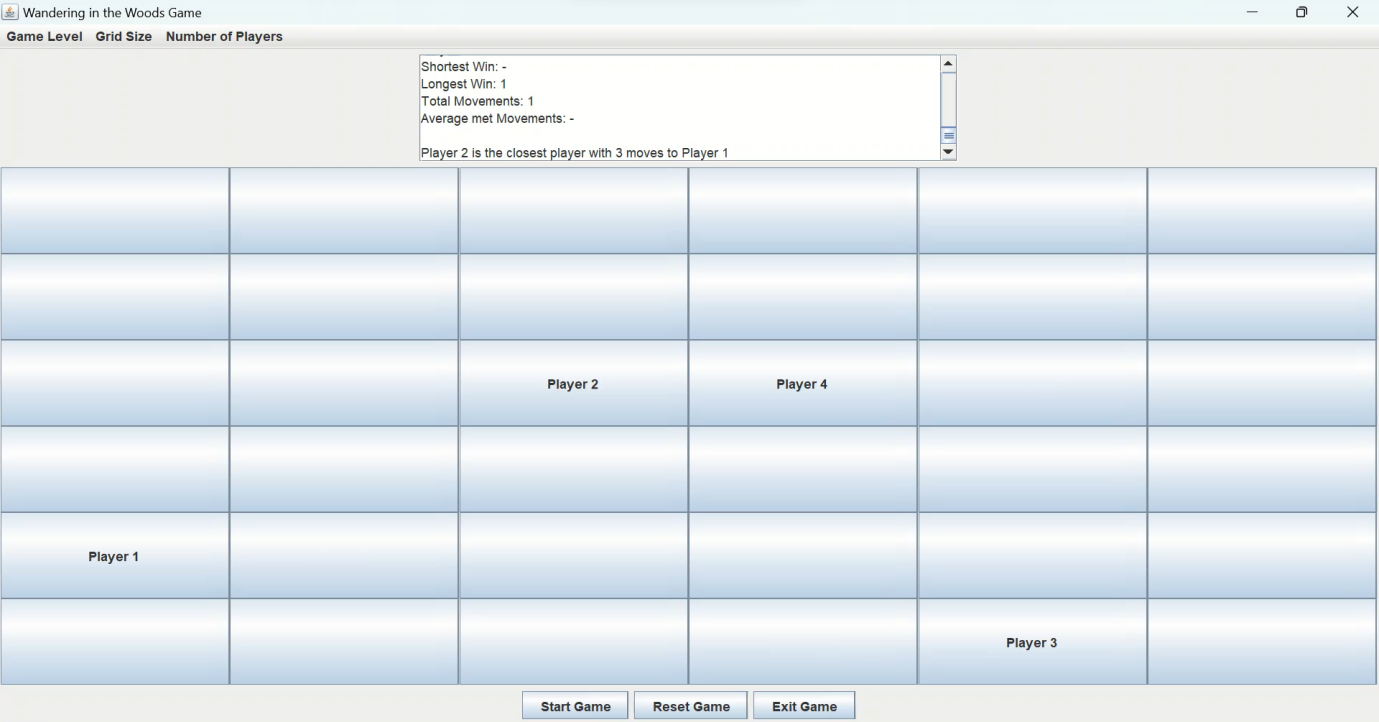
**Step 18** – Later the position of each player to be entered to start the game.



**Step 19** – the gaming window will appear with the stated player count and the selected position.



**Step 20 –** When each of the stated player meet each other in the same grid the game ends followed by generating the output result.



# References

|  |  |
| --- | --- |
| [1] | A. Y. Aleryani, "Comparative Study between Data Flow Diagram and Use case diagram," *International Journal of Scientific and Research Publications,,* vol. 6, no. 3, pp. 124-127, 2016. |
| [2] | H. S. D. Y. W. Bingyang Wei, "From state diagrams to sequence diagrams: a requirements acquisition approach," *International Journal of Computers and Applications,* vol. 41, no. 3, pp. 1-23, 2017. |
| [3] | J. I. A. A. D. T. Tanwir Ahmad, "Model-based testing using UML activity diagrams: A systematic mapping study," *Computer Science Review,* vol. 33, pp. 98-112, 2019. |
| [4] | S. S. Isha, "Software Testing Techniques and Strategies," *International Journal of Engineering Research and Applications,* vol. 4, no. 4, pp. 99-102, 2014. |
| [5] | Sathyavathy, "Review of Software Testing Techniques and Strategies," *International Journal of Research in Advent Technology,* vol. 5, no. 11, pp. 66-70, 2017. |
| [6] | S. Pargaonkar, "A Comprehensive Review of Performance Testing Methodologies and Best Practices: Software Quality Engineering," *International Journal of Science and Research,* vol. 12, no. 8, pp. 2008-2014, 2023. |