Florida International University
School of Computing and Information Sciences

Software Engineering Focus

Final Deliverable

Project Title: AR-VR-VE For Computer Science Education 2.0

Team Members:
Carlos Martinez, Mairim Barrios, Fidel Hernandez, Samira Tellez, Arelys Alvarez

Product Owner(s):
Dr. Francisco Ortega

Mentor(s):
Dr. Francisco Ortega

Instructor:
Dr.Francisco Ortega & Dr.Masoud Sadjadi

Abstract

This technical documents presents the planning process, design, system architecture, system requirements, resources, implementation and validation that the research and implementation project "AR-VR-VE for Computer Science Education 1.0" underwent, under the supervision of the product owner Dr. Francisco Ortega.

Table of Contents

INTRODUCTION

Current System

The education system plays a key role in this issue of including more people into computer science and the tech industry in general. In that sense, our project has been developed, showing very basic programming concepts in an approachable way. Our team has focused on the specific aspects that ensure the project's very first objective is achieved.

The previous project implemented with this overall goal is not considered a previous version of our current project because if was developed with other perspectives. For instances, CodeAdventure is a video game made in Unity3D engine that implements VR technology.

The environment of the game is developed inside jail. The character is a robot that is inside and the only way it has to become free is by selecting the right instructions. Then we can say it ensures that players use a logic reasoning to play. The features as colors and objects used to create the jail scene were very realistic.

We can conclude that CodeAdventure as the rest of the projects implemented last course were improved versions of their respective previous systems. Each of them had a different but not less valuable contribution and will definitely continue to be improved by the current and futures versions because of the results achieved.

Purpose of New System

AR-VR-VE for Computer Science Education 1.0 aims at attracting non-Computer Science students to the field by creating a simple and enjoyable game that teaches them the fundamental concepts of computer programming.

The objective of the game is to have a character navigate a series of mazes. To accomplish this task the user is given a series of commands and they need to choose which ones they are going to use and in which order. These instructions are then parsed and if the user chooses the right sequence of commands, the character navigates the maze successfully.

USER STORIES

The following section provides the detailed user stories that were implemented in this iteration of the AR-VR-VE For Computer Science Education. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories

User Stories: User Interface Mairim Barrios

As a User I would like to easily navigate through the user interface and add instructions to the solution panel so that I can develop a solution to submit later.

Acceptance Criteria:

1. Controls are laid out in an uniform manner.
2. When clicking a control/instruction, the same title pops up in the solution panel.

As a User I would like to submit my solution by clicking on the submit button so that I can move my character.

Acceptance Criteria:

1. Submit button adds the commands to a command list

As a User I would like to delete instructions from the solution panel so that I can remove those instructions I believe unnecessary.

Acceptance Criteria:

1. When the use hovers over one of the solutions in the command panel, a delete icon appears.
2. If the user clicks on the delete icon for a command, that command is removed from the solution panel.
3. Once a command has been removed, the commands below it are pushed up one position in the solution panel.

4. The command that was also removed from the CommandList<List> in the script that processes the commands.

As a User I would like to have a clear button to remove all instructions at once so that I can empty my slate and begin fresh.

Acceptance Criteria:

1. When the user clicks on the trash icon, all instructions are removed from the solution panel.
2. When the user clicks on the trash icon, instructions saved in the CommandList<List> in the scripts are also removed.

As a User I would like to be walked through the game before starting so that I can get a clear understanding of how to succeed in the game.

Acceptance Criteria:

1. Before the first level commences, the user see various tool tips that indicate for what each control is used, as well as what is expected from him/her.

As a Developer I would like to disable the submit & clear buttons once the character starts executing commands until it finishes.

Acceptance Criteria

1. Once the “Submit” icon has been clicked, the instructions are sent for execution and all of the command controls, the submit button and the clear button are disabled.
2. Once the character has executed its last command, the controls, submit button and clear button become enabled again.

As a player I would like to see the time I have been taking for my solution, displayed at the top of the screen.

Acceptance Criteria

1. Time does not begin counting until all the initial tips have been displayed.

As a User I would like to click on the repeat button and specify the amount of times I would like a command to be repeated.

Acceptance Criteria

1. When the user clicks on the repeat button, a small panel pops up in the middle of the screen with the commands and an input field next to each one.
2. The user can enter how many times they would like that command to be repeated in that section.

3. When the user clicks the “confirm” button within the panel, the command and the amount of times it should be repeated show up in the solution panel.

Carlos Martinez

As a user I would like there to be a pause menu so that I can take a break from the game whenever I desire.

Acceptance Criteria:

1. There is a button that pauses the game
2. Player can exit game by pressing the quit button
3. Player can return to game by pressing either ‘resume game’ or with a button from the keyboard.

As a user I would like to move my character in the basic cardinal directions so that I could properly play the game.

Acceptance Criteria:

1. Basic up, down, left, right commands work properly
2. Character moves accordingly/smoothly to commands

As a developer I would like to have it so that the character when moving around does so smoothly so that the user never notices any jerky movement.

Acceptance Criteria:

1. Character moves smoothly through maze
2. Character turns smoothly

As a user I would like there to be a way for I can put my commands in a loop so that I can use less commands to reach the end of the level

Acceptance Criteria:

1. Basic commands working alongside the loop commands
2. Basic commands work in the loop and get done multiple times

As a user I would like to be able to issue multiple while loop commands so if I need to use more than one while loop to solve a problem I can do so with ease.

Acceptance Criteria:

1. Multiple while loops in one series of commands
2. Each loop is able to perform a different series of commands.

Pending User Stories

As a user I would like my character to polymorph into different objects so that they can solve interesting weight based puzzles

Acceptance Criteria:

1. Player has a button named polymorph that they can press
2. Player can choose any item to polymorph into
3. Player can change their weight property whenever they choose a specific item.

As a user I would like there to be a system implemented where I can store commands in a method so I can use them instead of repeating a lengthy series of commands

Acceptance Criteria:

1. User can store series of commands
2. User can name stored series of commands
3. Can input basic commands and loop commands into stored commands

(The script for it is complete. The UI has not yet been updated to incorporate it)

Project Plan

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings

are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

Unity:	Game engine
C#:	Primary programming language used
Game Sparks:	Gaming server used
Github:	Used for version control

Sprints Plan

Sprint 1 Planning Meeting Minutes

Attendees: Arelys Alvarez
Fidel Hernandez
Samira Tellez
Carlos Martinez

Mairim Barrios

Start time: <7:00pm>
End time: <7:30pm>

- All Team members will start getting familiarized with Unity Game Engine, C# and Git.
- All team members will review previous documentation.

Sprint 2 Planning Meeting Minutes

Attendees: Arelys Alvarez
Fidel Hernandez
Samira Tellez
Carlos Martinez
Mairim Barrios

Start time: <7:00pm>
End time: <7:37pm>

After discussion, the velocity of the team were estimated to be <80hrs>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story <1: As a user I want a menu system so that I can select whether I want to begin the game or exit > 8hrs
- User Story <2: As a player I want to be able to enter my information so that I can store my scores in the game> 8hrs
- User Story <3: As a player I want the first level to be easy in difficulty so that I can understand the concept of the game.> 8hrs
- User Story <4: As a user I want the game to walk me through the different tools and mechanics that I can use to advance to the next level. > 8hrs
- User Story <5: As a user I want to see the different instructions available for each of the levels> 6hrs
- User Story <6: As a user I want to know if the instructions selected are the correct ones so that I know if I pass or fail the level > 8hrs

- User Story <7: As a character I want to move in the direction specified by the user's commands so that I can get closer towards the exit > 8hrs
- User Story <8: As a developer I want to research the most optimal ways of moving a character on top of a layer of blocks so that my character moves from beginning to end > 10hrs
- User Story <9: As a user I want to be able to see >
- User Story <10: As a developer I want to research ways of designing three dimensional paths so that the player sees the illusion of a maze > 8hrs
- User Story <11: As a developer I want to research the ways of implementing a drag and drop functionality so that the player can drag the instructions into the command holder > 10hrs
-

The team members indicated their willingness to work on the following user stories.

- <Mairim Barrios>
  - User Story <5: As a user I want to see the different instructions available for each of the levels> 6hrs
  - User Story <6: As a user I want to know if the instructions selected are the correct ones so that I know if I pass or fail the level > 8hrs

- <Carlos Martinez>
  - User Story <1: As a user I want a menu system so that I can select whether I want to begin the game or exit > 8hrs
  - User Story <7: As a character I want to move in the direction specified by the user's commands so that I can get closer towards the exit > 8hrs

Sprint 3 Planning Meeting Minutes

Attendees: Arelys Alvarez
Fidel Hernandez
Samira Tellez
Carlos Martinez
Mairim Barrios

Start time: <1:35pm>
End time: <2:00pm>

After discussion, the velocity of the team were estimated to be <84pts>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story <787: Add and adjust an additional camera into the main screen.>
- User Story <789: Basic Character Movement>
- User Story <801: Spawn user instructions upon button click>
- User Story <791: Multiple Loops>
- User Story <799: Basic Movement with Methods>

The team members indicated their willingness to work on the following user stories.

- <Mairim Barrios>
  - User Story <787:Add and adjust an additional camera into the main screen.>
  - User Story <801: Spawn user instructions upon button click>

- <Carlos Martinez>
  - User Story <789: Basic Character Movement>
  - User Story <791: Multiple Loops>
  - User Story <799: Basic Movement with Methods>

Sprint 4 Planning Meeting Minutes:

Attendees: Arelys Alvarez
Fidel Hernandez
Samira Tellez
Carlos Martinez

Mairim Barrios

Start time: <1:30pm>
End time: <1:45pm>

After discussion, the velocity of the team were estimated to be <80>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story <808:Rearrange UI for Level 2 and add new functionality for new difficulty>
- User Story <809: Create scripts to parse more complex user instructions>
- User Story <800: Methods with Loop>

The team members indicated their willingness to work on the following user stories.

- <Mairim Barrios>
  - User Story <808:Rearrange UI for Level 2 and add new functionality for new difficulty>
  - User Story <809: Create scripts to parse more complex user instructions>

- <Carlos Martinez>
  - User Story <800: Methods with Loop>

Sprint 5 Planning Meeting Minutes:

Attendees: Arelys Alvarez
Fidel Hernandez
Samira Tellez
Carlos Martinez
Mairim Barrios

Start time: <1:30pm>
End time: <1:45pm>

After discussion, the velocity of the team were estimated to be <80>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story <810:Add editing tools for solution>
- User Story<813: Work character movement with illusion>
- User Story<814: Object Oriented based movement>

The team members indicated their willingness to work on the following user stories.

- <Mairim Barrios>
  - User Story <User Story <810:Add editing tools for solution>

- <Carlos Martinez>
  - User Story<813: Work character movement with illusion>
  - User Story<814: Object Oriented based movement>

Sprint 6 Planning Meeting Minutes:

Attendees: Arelys Alvarez
Fidel Hernandez
Samira Tellez
Carlos Martinez
Mairim Barrios

Start time: <1:30pm>
End time: <1:45pm>

After discussion, the velocity of the team were estimated to be <80>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story <816:Create the UI for levels 3 and 4 >

The team members indicated their willingness to work on the following user stories.

- <Mairim Barrios>
  - User Story <816:Create the UI for levels 3 and 4 >

- <Carlos Martinez>
  - User Story <822:Add finishing touches to animation>
  - User Story <823:Continue working on movement(turning)>

SYSTEM DESIGN

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. Finally, the design patterns used in the project are discussed.

Architectural Patterns

The system was structured in a **MVC** architecture. Where the **M**odels are the game objects such as the player, the maze, the solution panel and the controls. The Unity scene that shows the user interface was treated as the **V**iew, and lastly, the C# scripts that allowed for communication between the Unity Scene and the backend, was treated as the **C**ontroller. This structure separates the business layer, the user interface and the database logic from each other.

System and Subsystem Decomposition

Models

Player: Character that performs the commands

SolutionPanel: Holds the list of commands , loops , editing buttons and scroll rect

Commands (Buttons): Specify the instructions that the player will execute

Controllers

SubmitButtonController : When the submit button within the solution panel is clicked it calls the submitButtonController, which handles the processing of the commands and its execution.

SolutionPanelController: Controls the display of deleteButtons upon hovering of the mouse over a command. Enables/Disables the submit/clear/command buttons while the player is executing instructions.

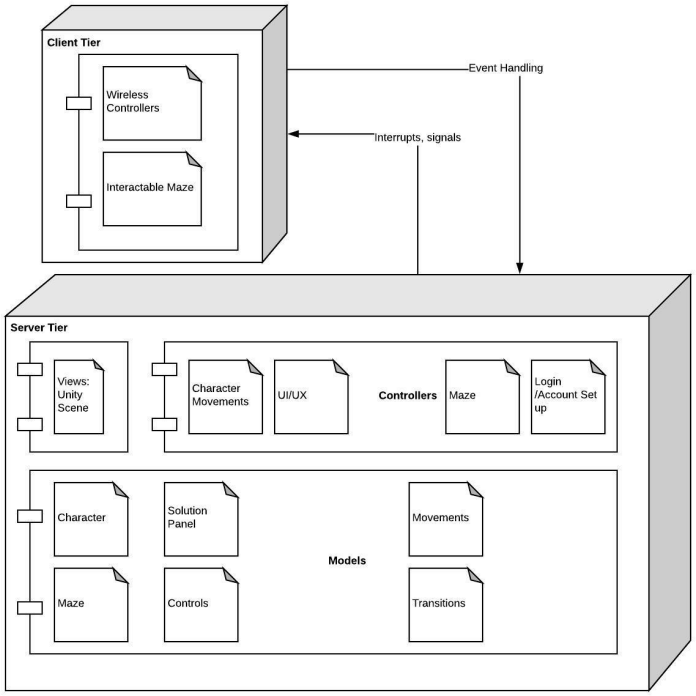
PlayCommandController: Handles the turning and movement of the character according to the input into the solutionPanel.

BlockRotatorController: Handles the movement of the blocks.

Views

The single Unity Scene that contains the maze, player movement and solution panel is considered the view in this system.

Deployment Diagram



System Validation

Unit Test

- Test case ID: 1
- Description/Summary of Test: Verify that by clicking a command control, the correct command pops up in the solution panel.
- Pre-condition: User clicks a command.
- Expected Results: Solution panel shows the same command being added.
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 2
- Description/Summary of Test: Verify that the delete button shows up when hovering over a command.
- Pre-condition: Player hovers over a command.
- Expected Results: Delete button shows up
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 3
- Description/Summary of Test: Verify that the delete button leaves when hovering away from a command.
- Pre-condition: Player's mouse is currently over a command.
- Expected Results: The delete button leaves.
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 4
- Description/Summary of Test: Verify that all commands are deleted upon clicking the trash icon.

- Pre-condition: There are commands in the solution panel.
- Expected Results: All commands are deleted from the solution panel.
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 5
- Description/Summary of Test: Verify that the commandList on the backend is cleared when the trash icon is cleared.
- Pre-condition: The trash icon has been clicked.
- Expected Results: The commandList holding the commands deletes all of its items.
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 6
- Description/Summary of Test: Verify that the timer is paused when the game starts until the tool tips are finished.
- Pre-condition: The player has started playing.
- Expected Results: The timer begins after the last tool tip.
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 7
- Description/Summary of Test: Verify that the commands are added to the commandList on the backend when the user clicks on Submit.
- Pre-condition: There are commands in the commandPanel.
- Expected Results: The commandList gets populated when the player clicks on submit button.
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 8
- Description/Summary of Test: Verfiy that a panel with the repeat options pops up upon clicking the repeat button.
- Pre-condition: Player clicks on repeat button.
- Expected Results: A pop up panel shows up showing the optional commands and an input field to specify the amount of times it can be repeated and the order.
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 9
- Description/Summary of Test: Verify that after clicking the confirm button on the repeat panel, the solution panel is filled accordingly.
- Pre-condition: Player clicks the confirm button on the repeat panel.
- Expected Results: Solution panel gets populated
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 10
- Description/Summary of Test:
- Pre-condition:
- Expected Results:
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 10
- Description/Summary of Test:
- Pre-condition:
- Expected Results:
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 11
- Description/Summary of Test:
- Pre-condition:
- Expected Results:
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 12
- Description/Summary of Test:
- Pre-condition:
- Expected Results:
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 13
- Description/Summary of Test: Verify that after pressing pause button, pause menu appears
- Pre-condition: Player has started the game and presed pause button
- Expected Results: After pushing pause button pause menu appears
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 14
- Description/Summary of Test: Verify that after inputting the “Turn Left” command the character turns left
- Pre-condition: Player has begun the game and issued a “Turn Left” command
- Expected Results: Character turns left
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 15
- Description/Summary of Test: Verify that after inputting the “Turn Right” command the character turns left
- Pre-condition: Player has begun the game and issued a “Turn Right” command
- Expected Results: Character turns right
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 16
- Description/Summary of Test: Verify that after inputting the “Move Forward” command the character moves forward
- Pre-condition: Player has begun the game and issued a “Move Forward” command
- Expected Results: Character moves forward
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 17
- Description/Summary of Test: Verify that player can issue a series of commands and the character will do them in sequence
- Pre-condition: Player has begun the game and issued a series of commands
- Expected Results: Character executes series of commands successfully
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 18
- Description/Summary of Test: Verify that player can issue a series of commands in a loop and that the player will execute that loop a certain number of times
- Pre-condition: Player has begun the game and issued a series of commands in a loop
- Expected Results: Character executes series of loop commands successfully
- Actual Result: Status (Fail/Pass): Passed

Unit Test

- Test case ID: 19
- Description/Summary of Test: Verify that player can issue multiple different series of commands in different loops and that the player will execute those loops a certain number of times in sequential order
- Pre-condition: Player has begun the game and issued a series of commands in multiple different loops
- Expected Results: Character executes series of loop commands successfully
- Actual Result: Status (Fail/Pass): Passed

GLOSSARY

1. Maze : The path which the player will walk on to get the key.
2. Solution Panel: The display that shows the commands the player wishes to execute.
3. Repeat Panel: A pop up form in which the player can choose which commands he wishes to repeat instead of selecting the same command multiple times.

APPENDIX

Appendix A - UML Diagrams

The UML diagram provided for the User Interface is that of the entire UI system. For UML diagrams broken down into smaller user stories, please refer to the DOCUMENTS folder under each of the sprints.

Fig 1. Use Case Diagram: Add/Edit instruction to and from the solution panel.

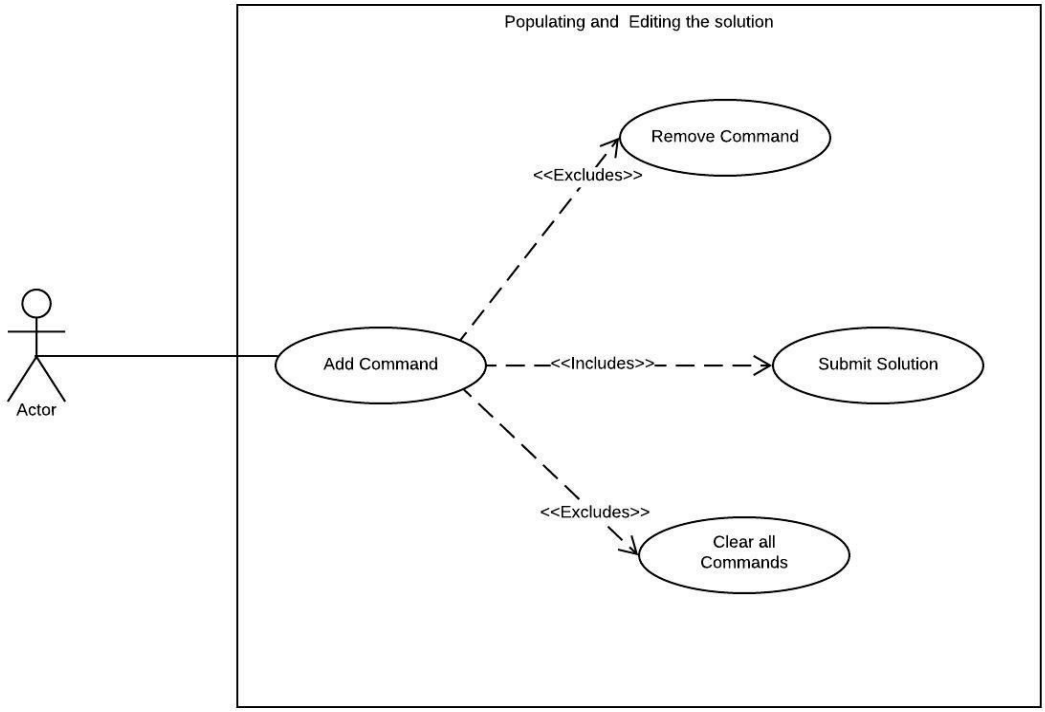

Populating and Editing the solution
Remove Command
<<Excludes>>
Actor
Add Command
<<Includes>>
Submit Solution
<<Excludes>>
Clear all
Commands

Fig 2. Sequence Diagram: Add/Edit instructions to and from the solution panel.

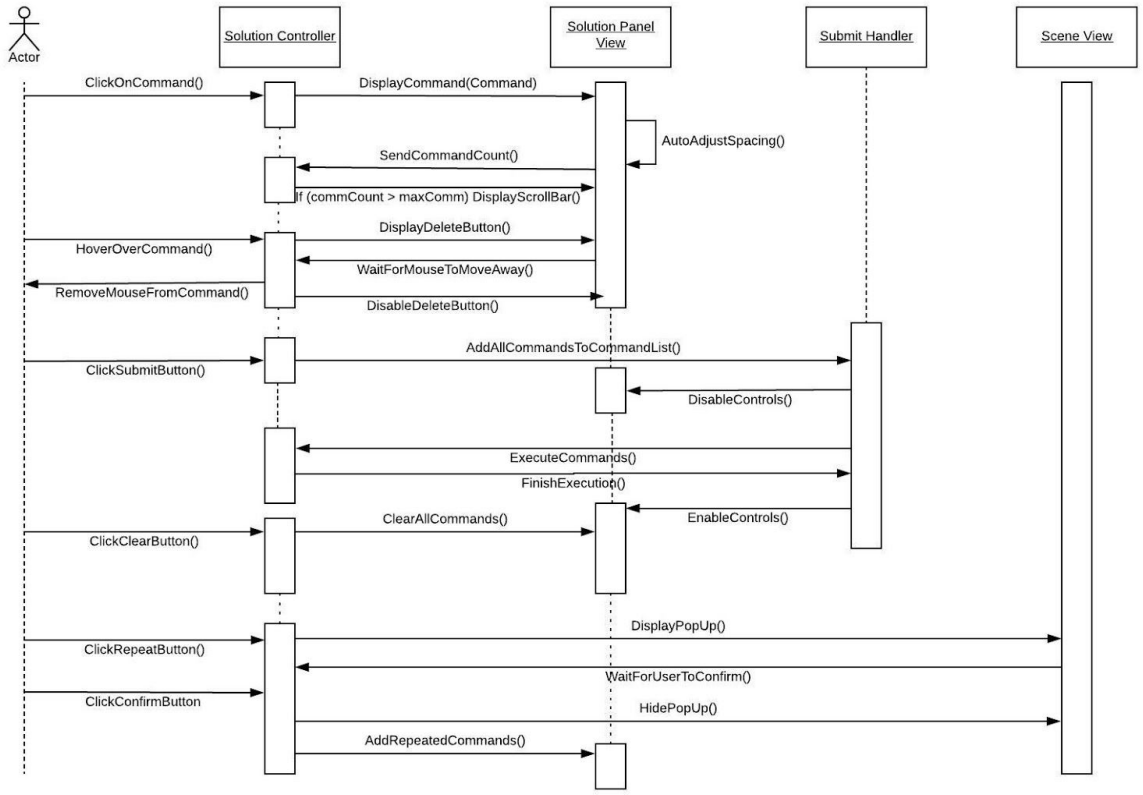
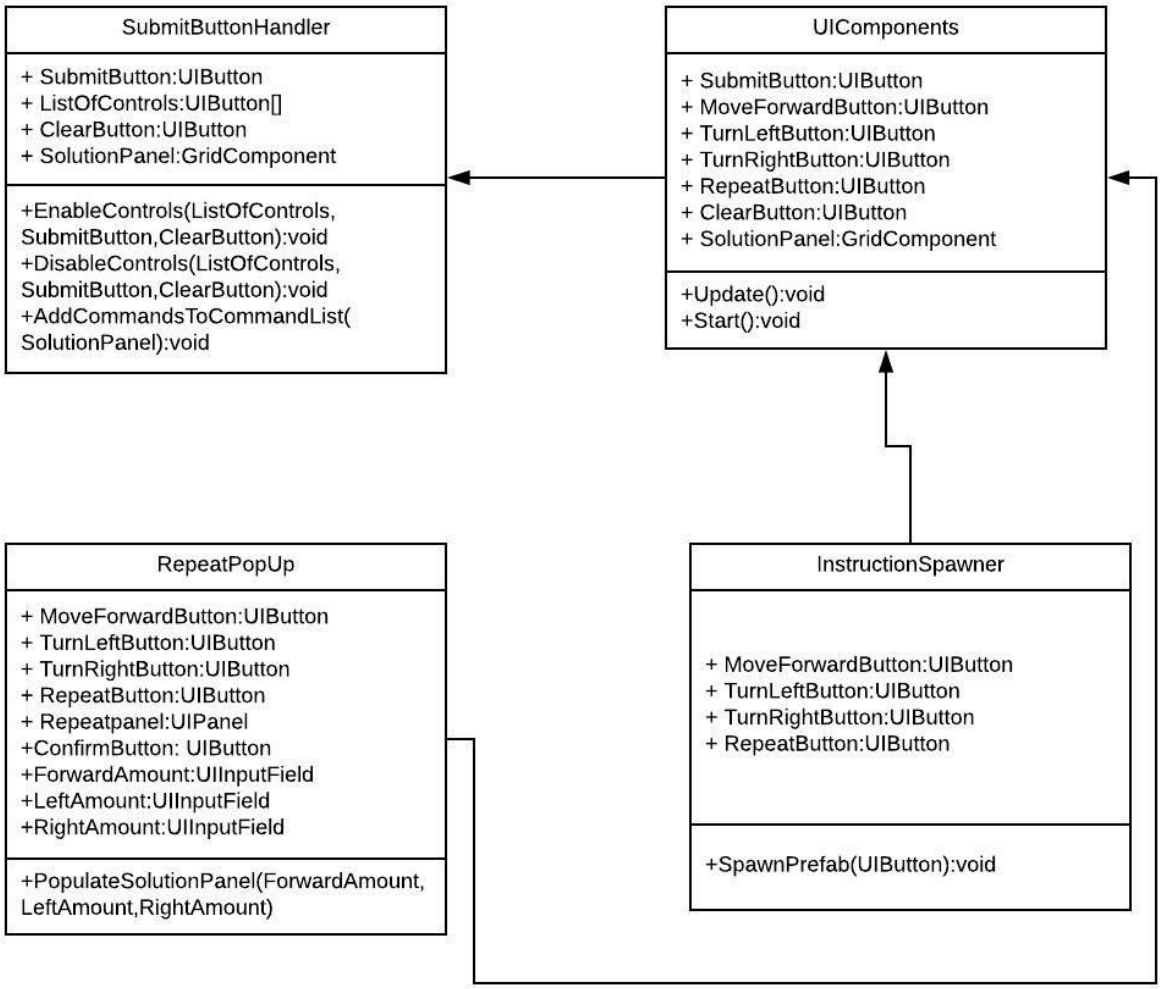


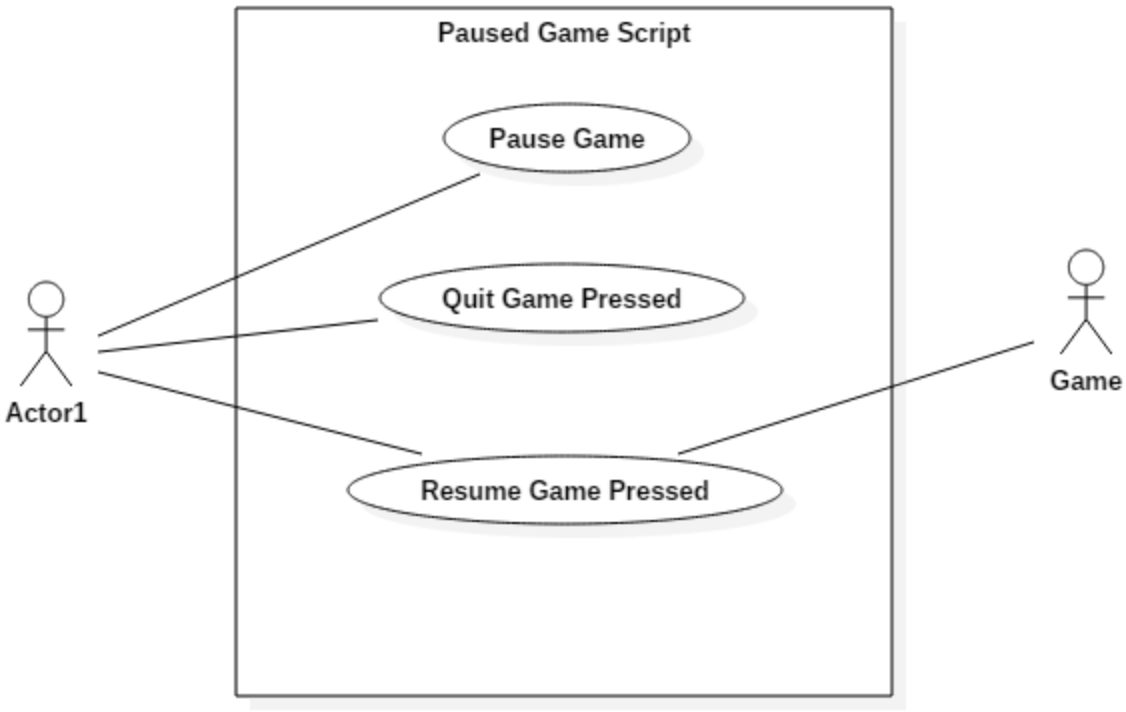
Fig 3. Sequence Diagram: Add/Edit instructions to and from the solution panel.



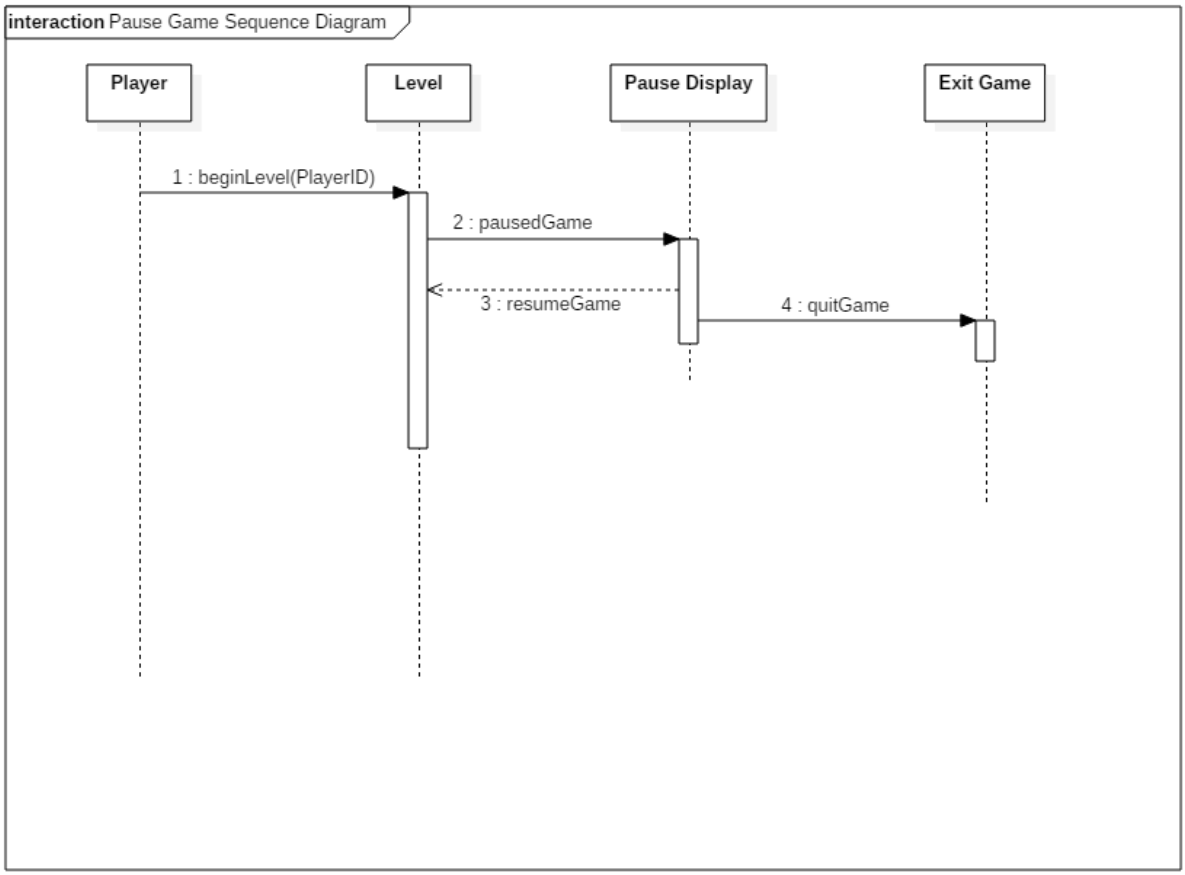
User Story <788>

As a user I would like there to be a pause menu so that I can take a break from the game whenever I desire.

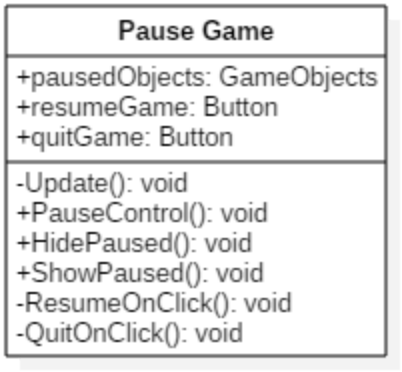
Use Case Diagram



Sequence Diagram



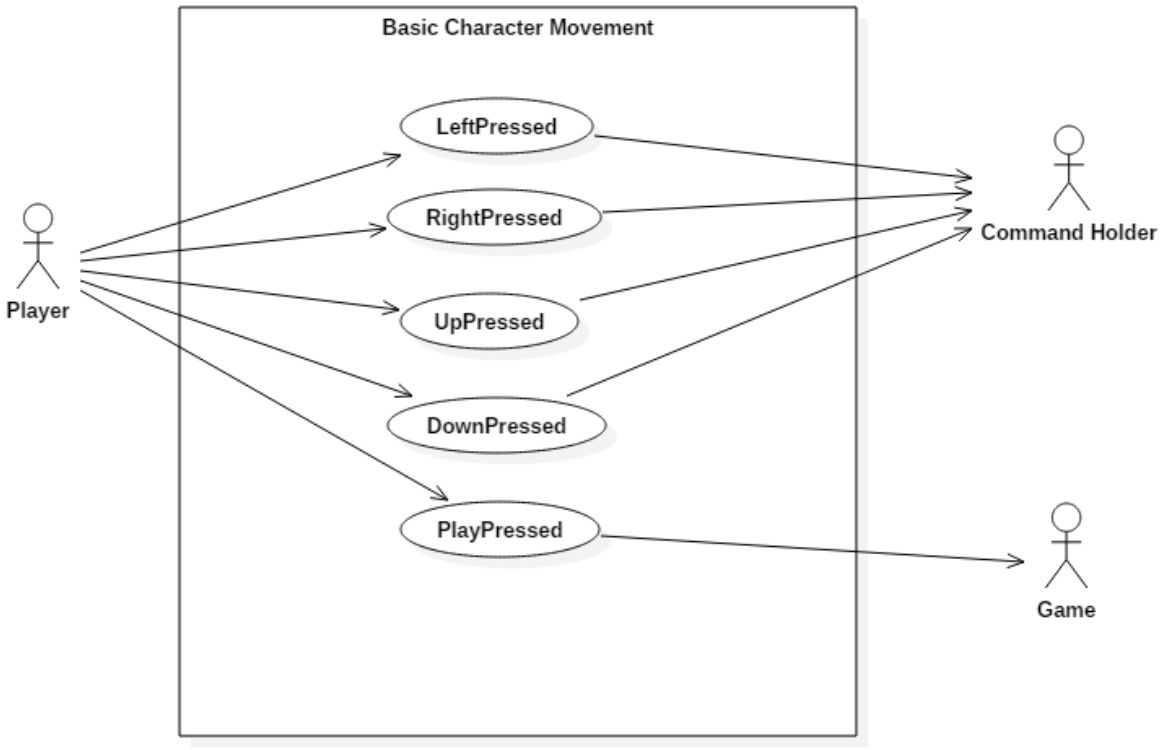
Class Diagram



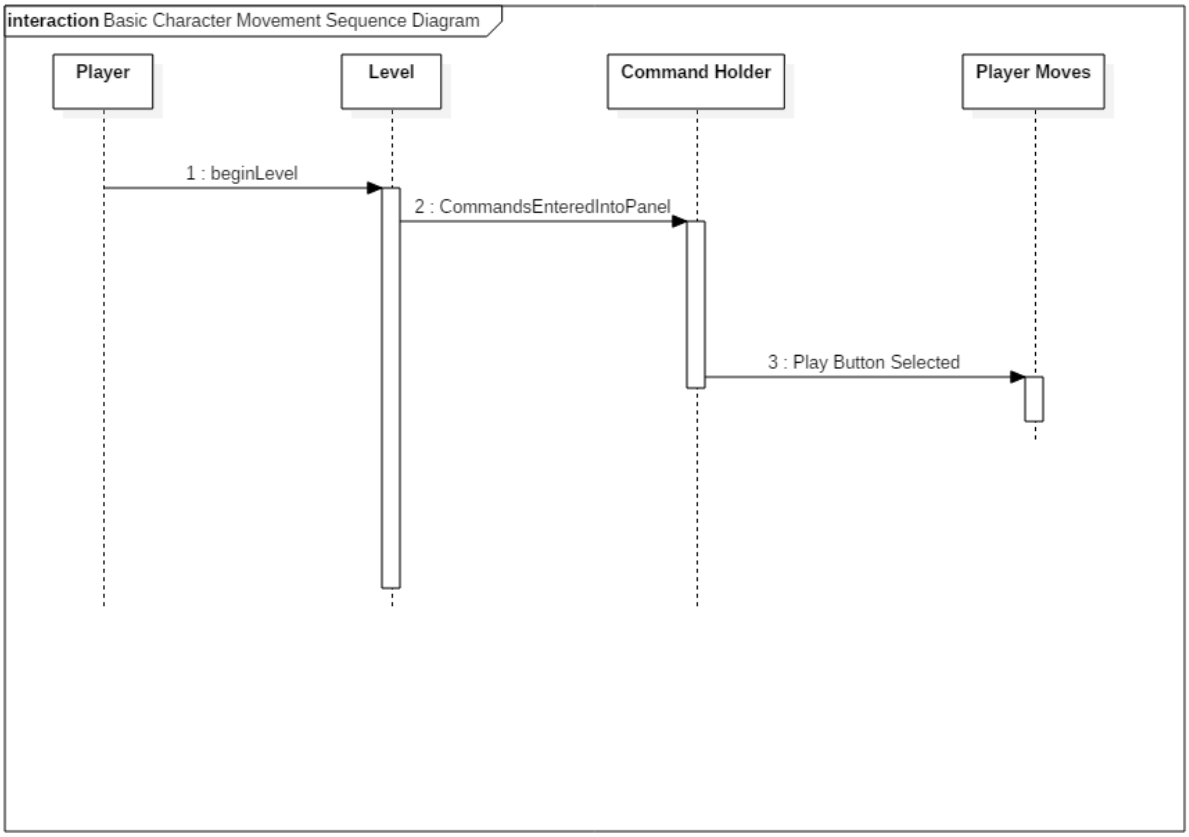
User Story <789>

As a user I would like to move my character in the basic cardinal directions so that I could properly play the game.

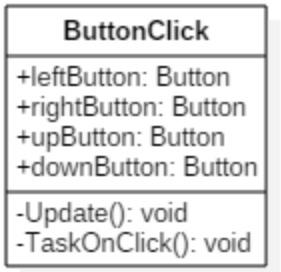
Use Case Diagram



Sequence Diagram



Class Diagram

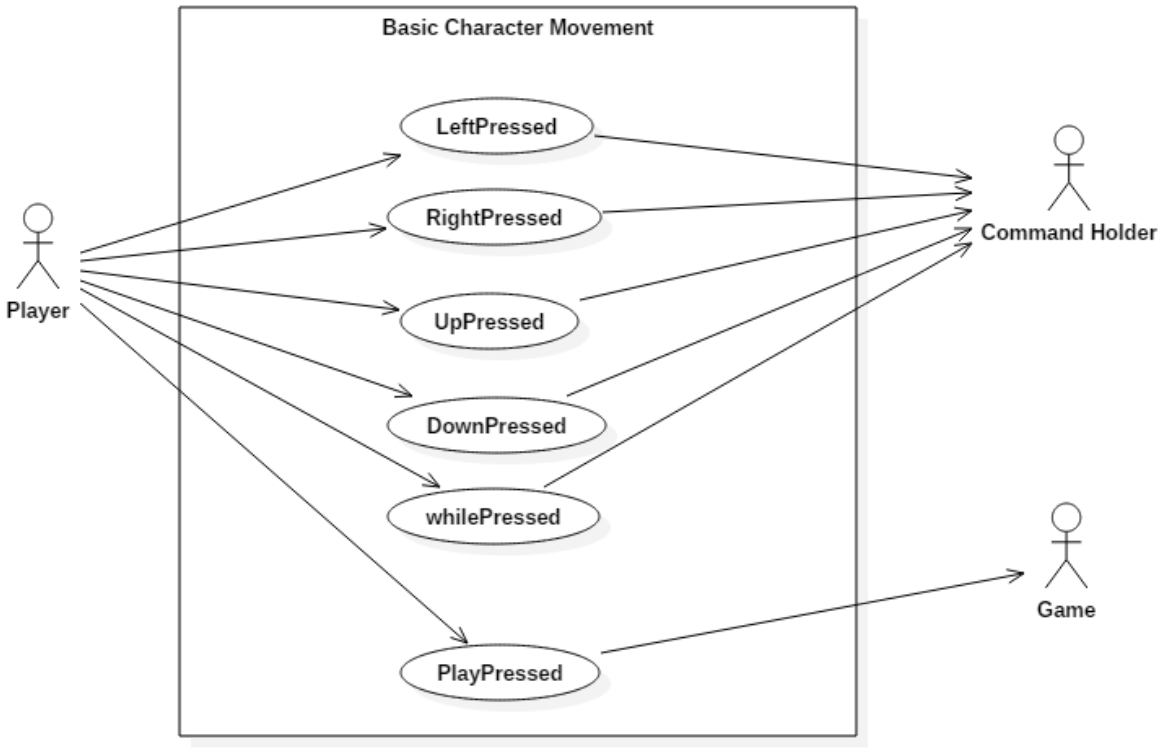


User Story <790>

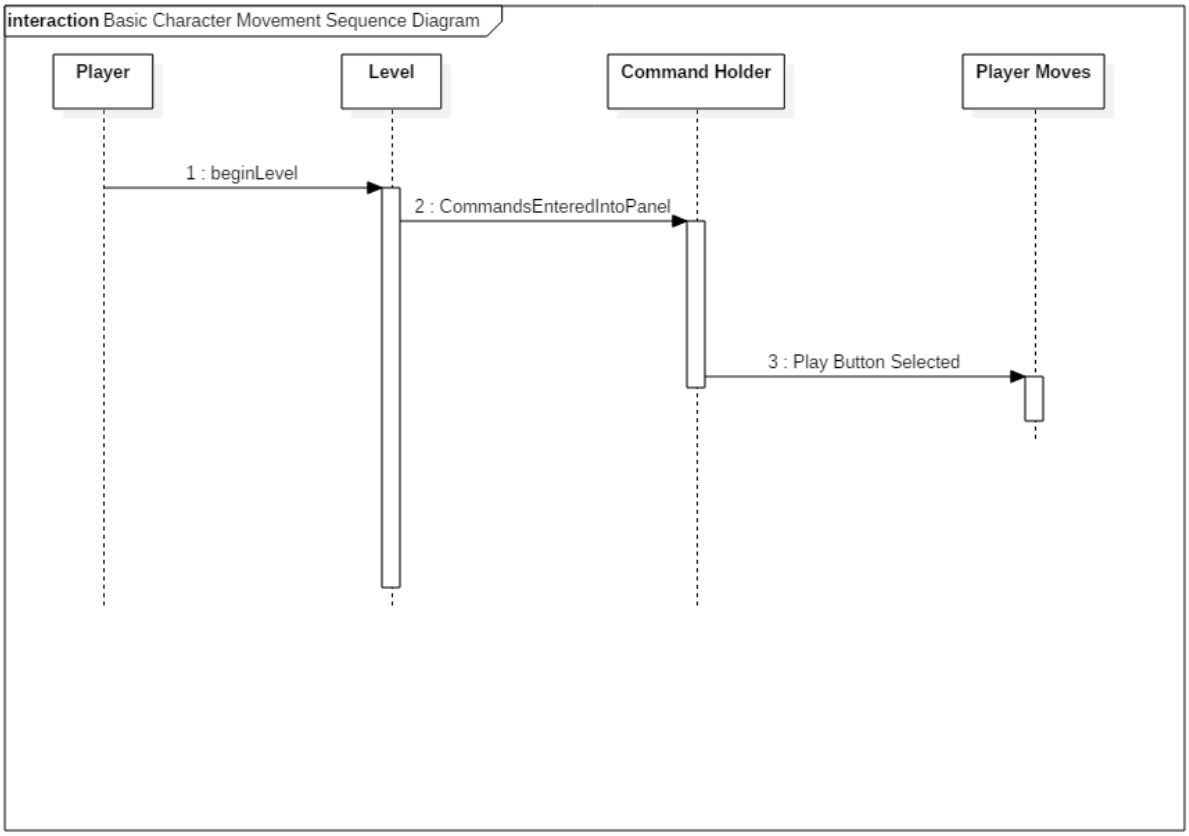
As a user I would like there to be a way for I can put my commands in a loop so that I can use less commands to reach the end of the level

---

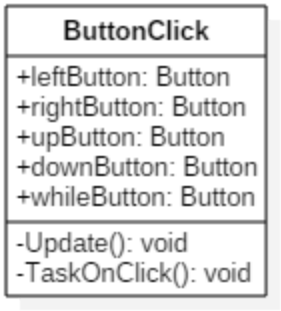
Use Case Diagram



Sequence Diagram



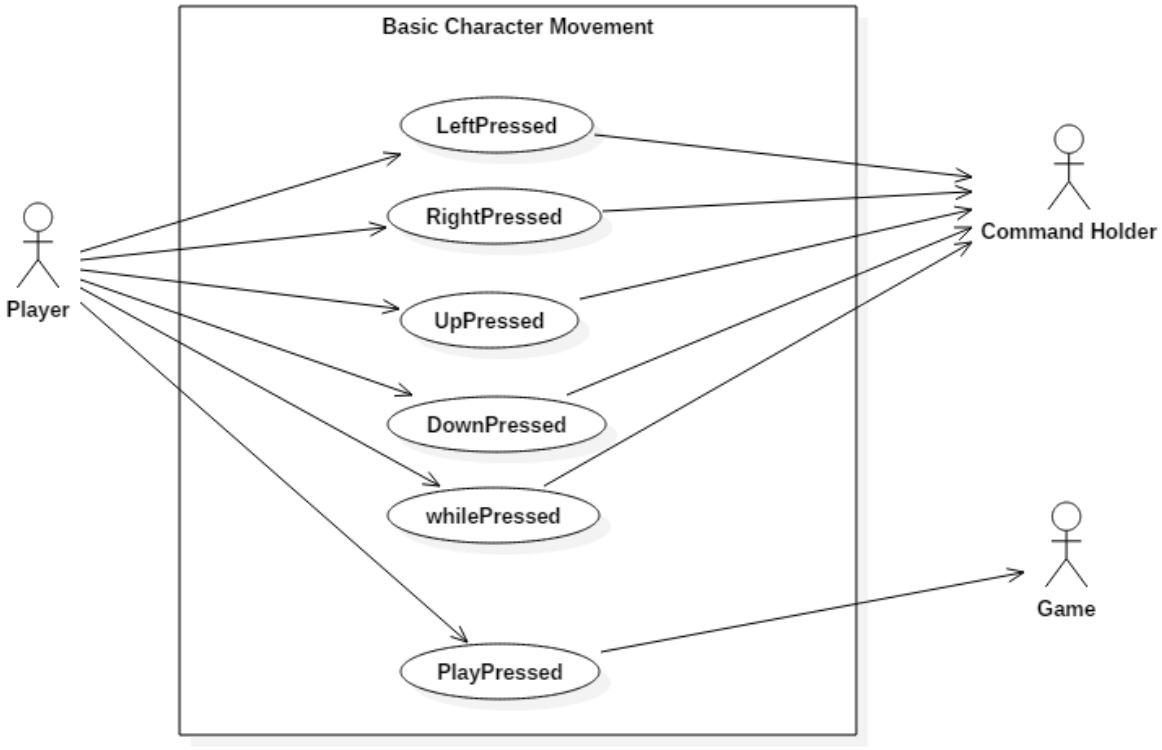
Class Diagram



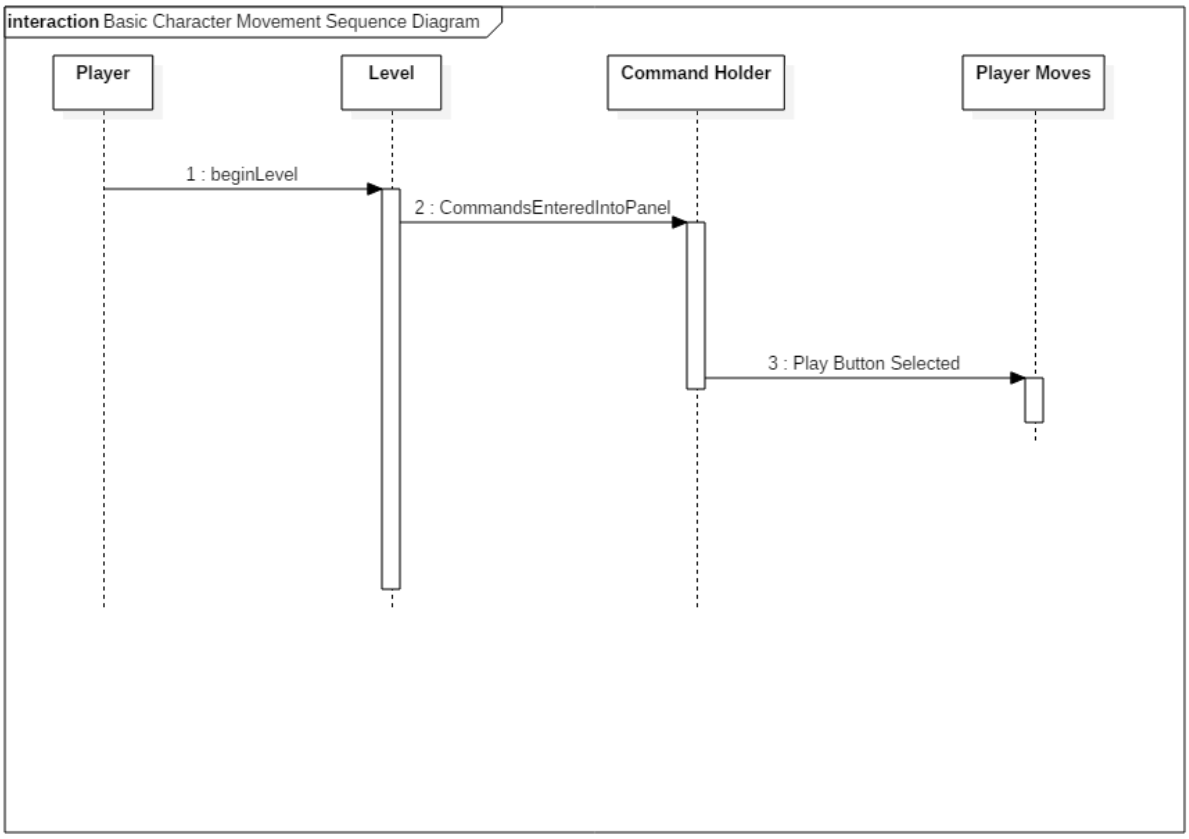
User Story <791>

As a user I would like to be able to issue multiple while loop commands so if I need to use more than one while loop to solve a problem I can do so with ease.

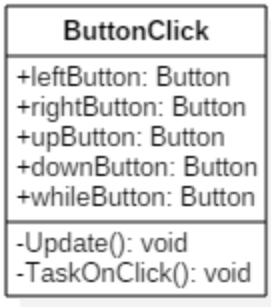
Use Case Diagram



Sequence Diagram



Class Diagram

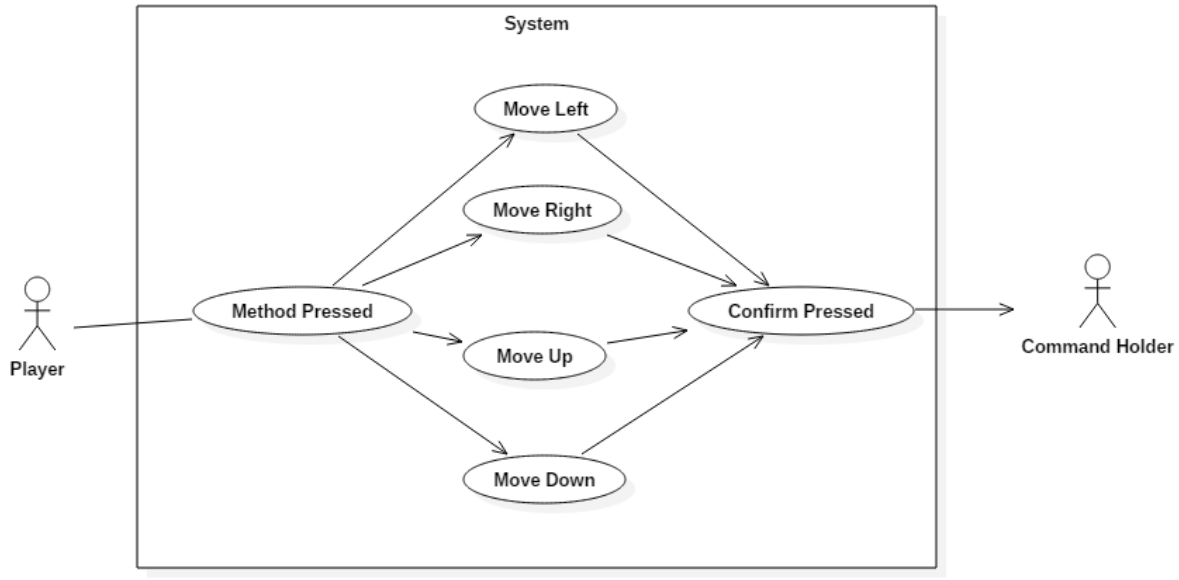


User Story <799>

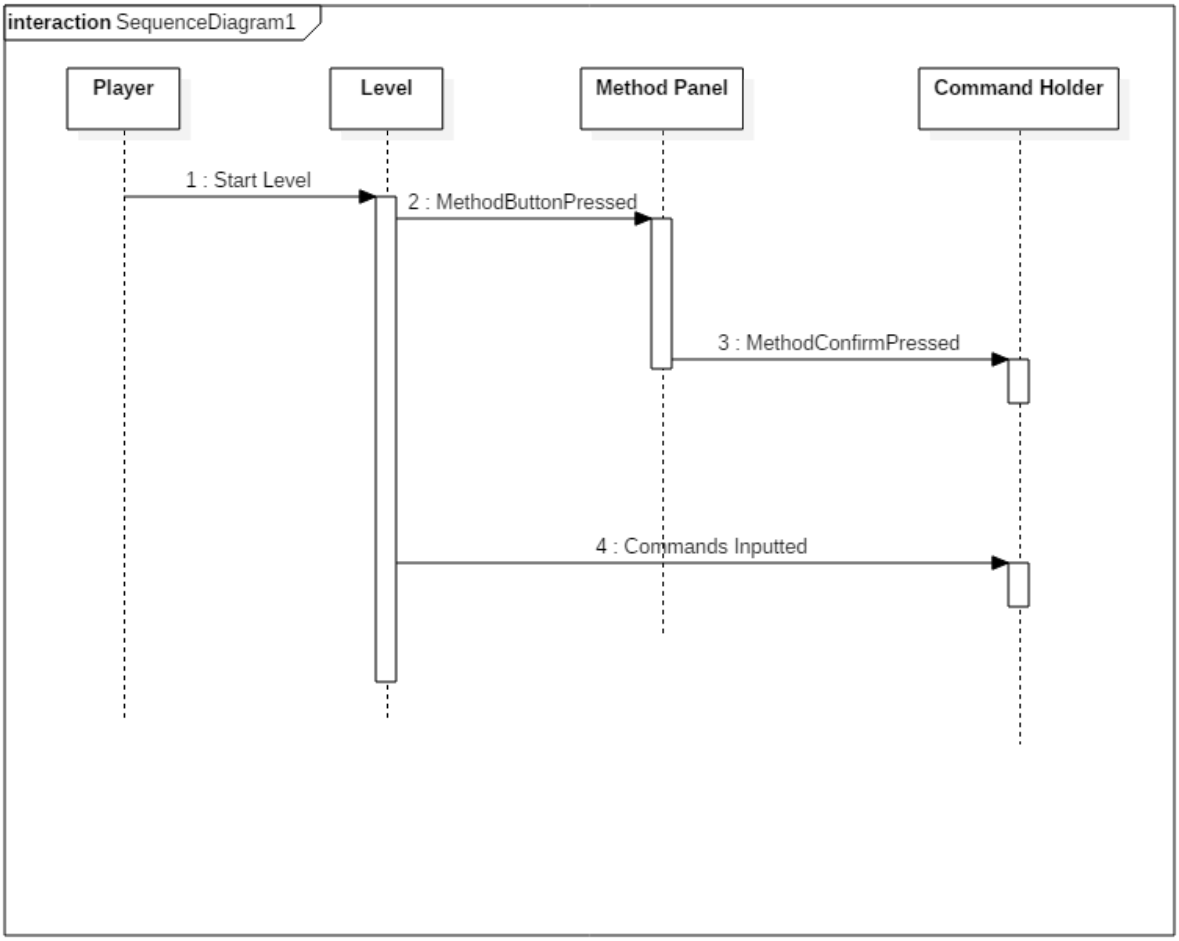
As a user I would like there to be a system implemented where I can store commands in a method so I can use them instead of repeating a lengthy series of commands

---

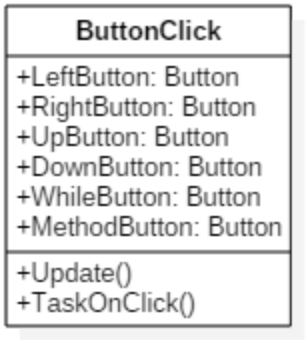
Use Case Diagram



Sequence Diagrams



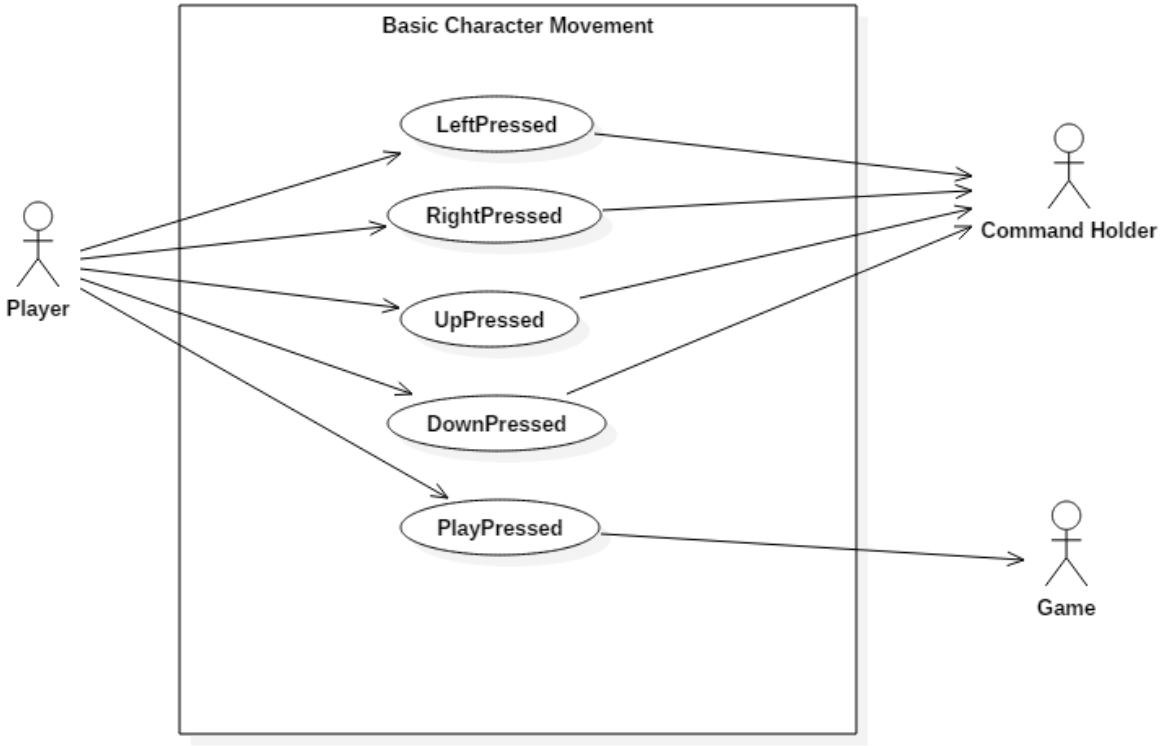
Class Diagram



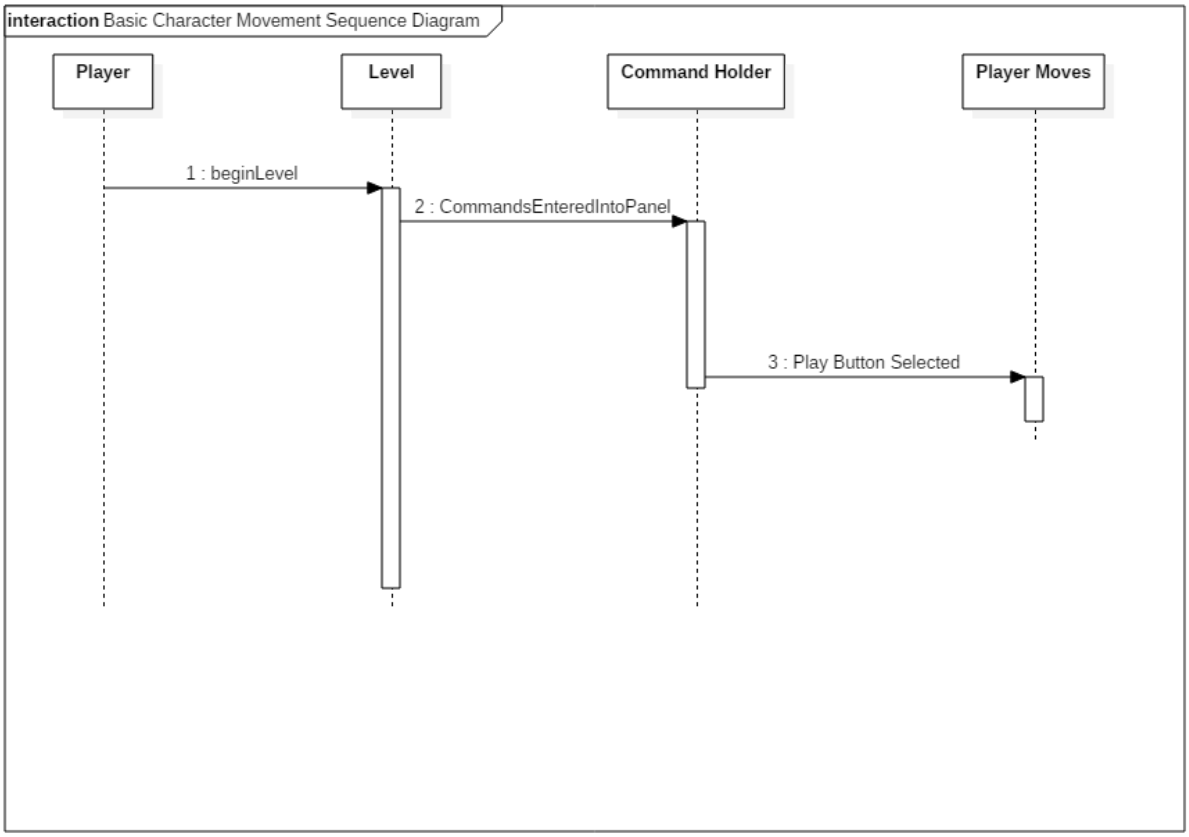
User Story <822>

As a developer I would like to have it so that the character when moving around does so smoothly so that the user never notices any jerky movement.

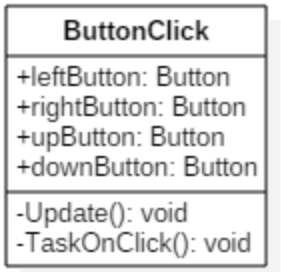
Use Case Diagram



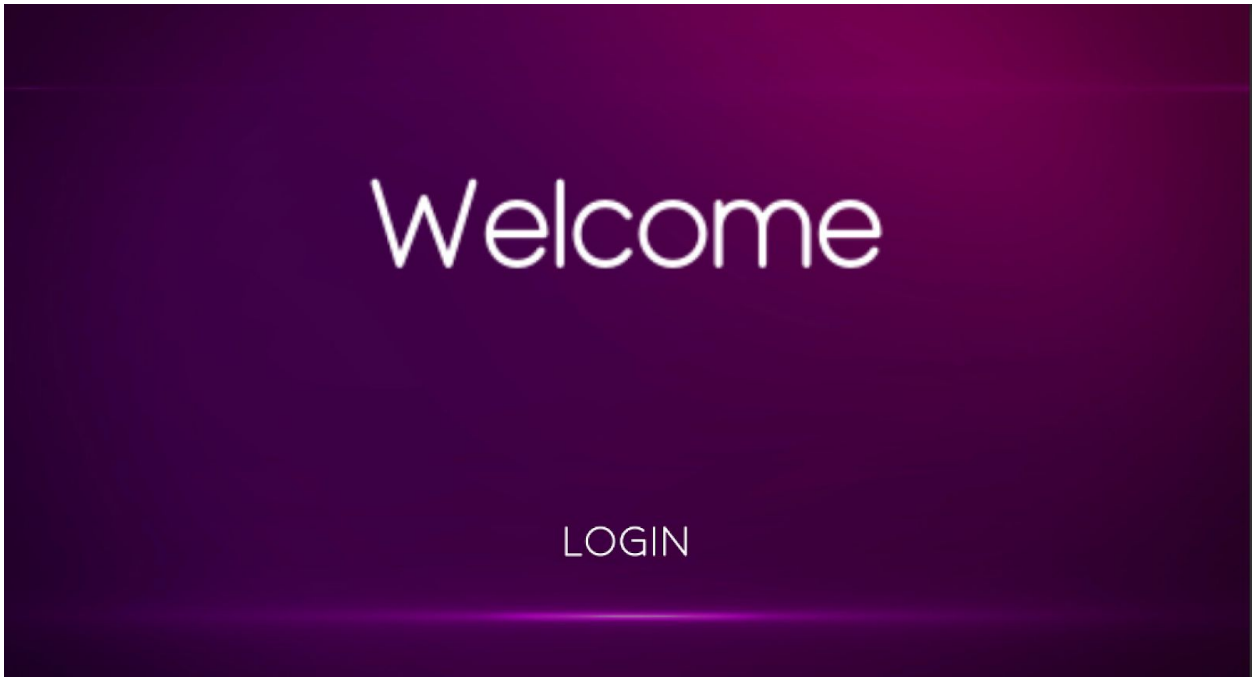
Sequence Diagram

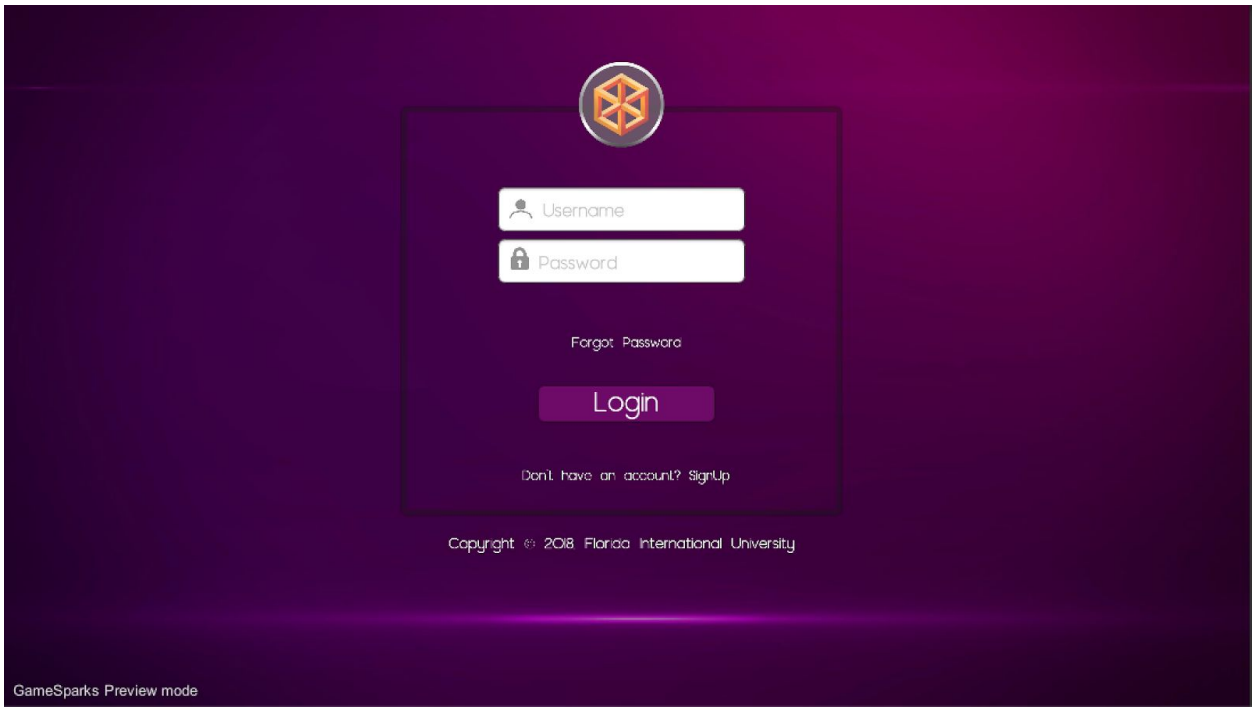


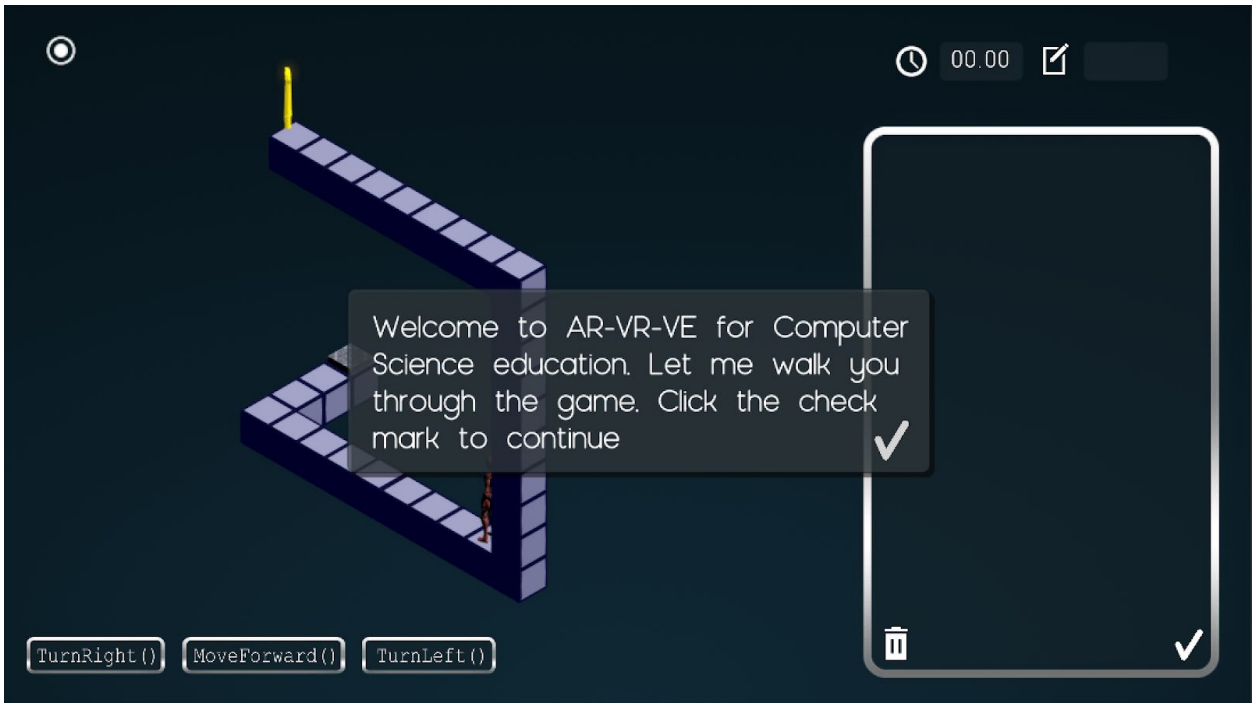
Class Diagram

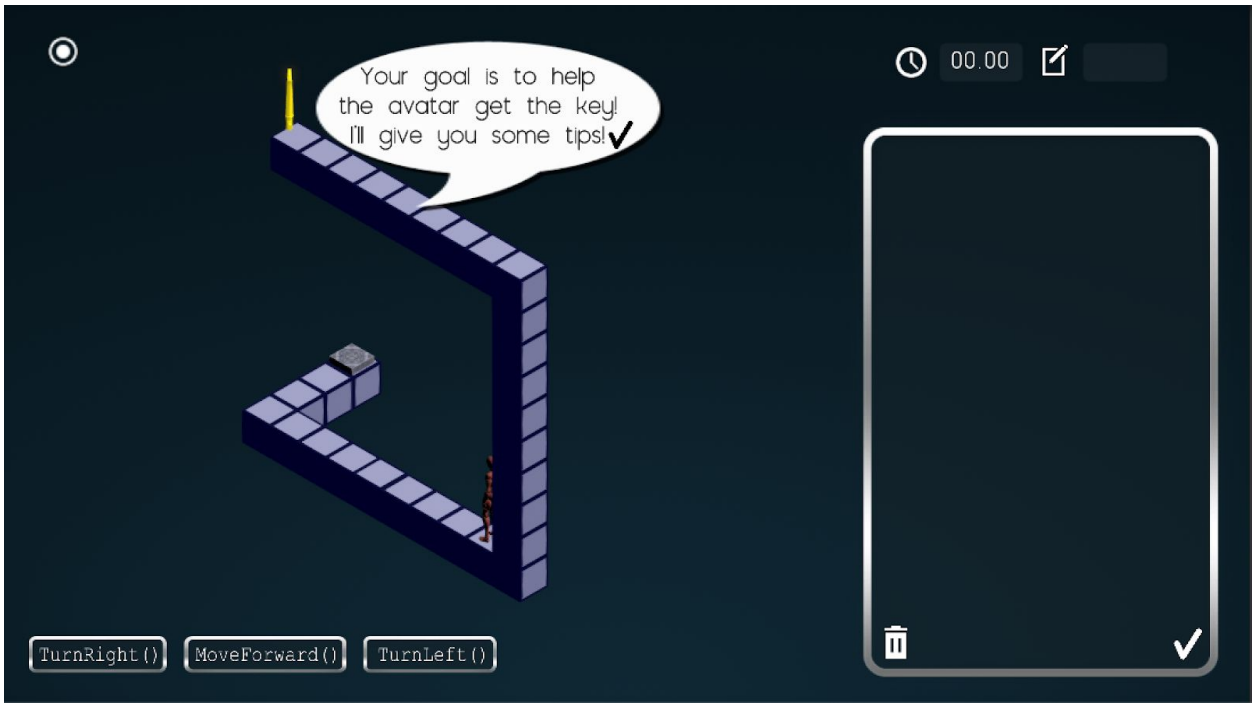


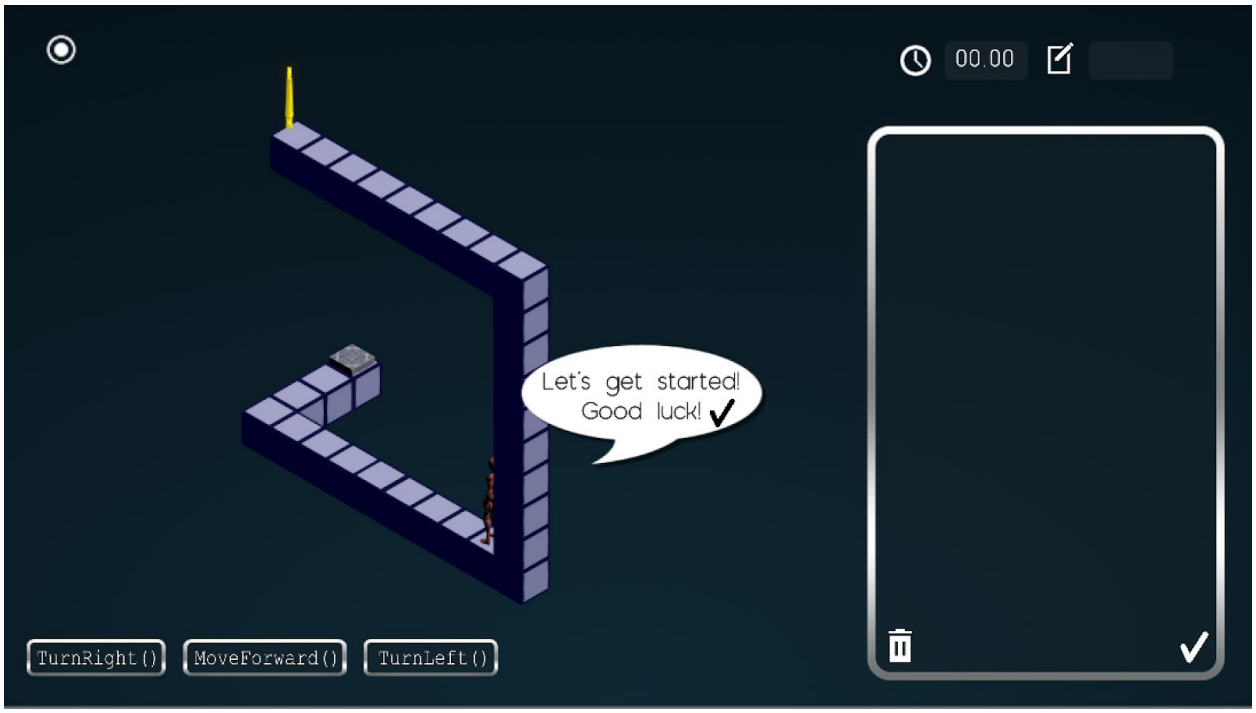
Appendix B - User Interface Design

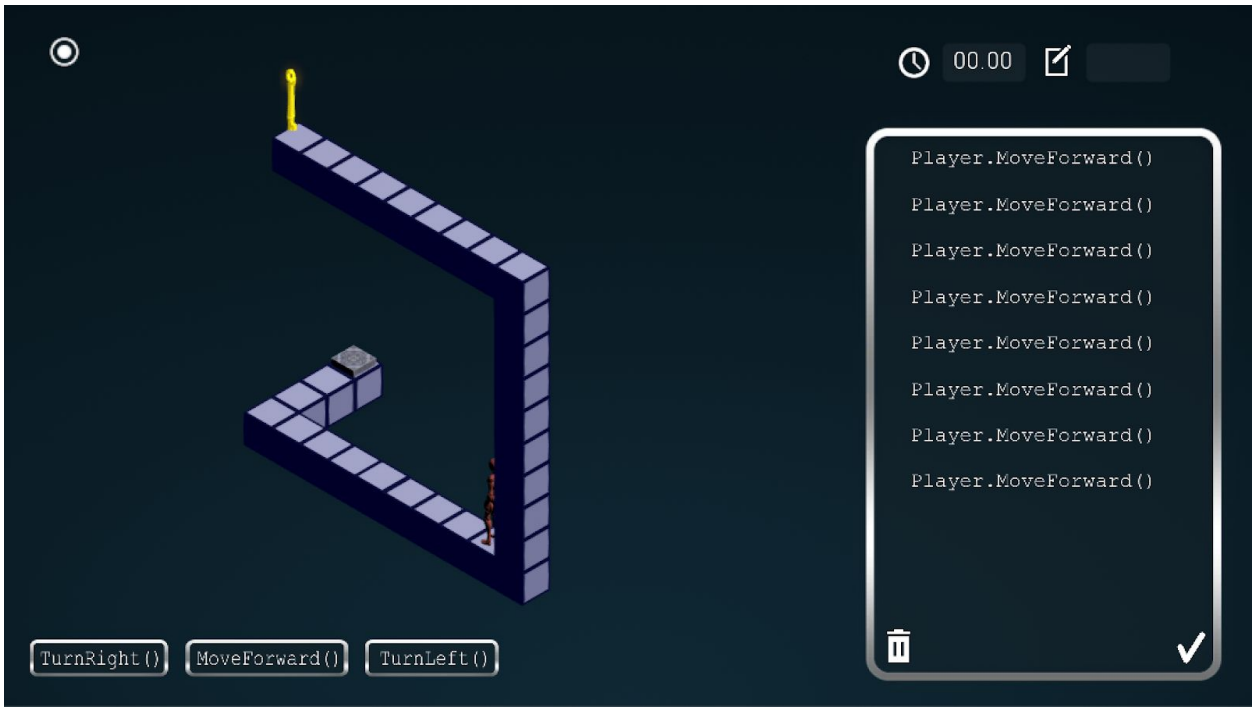


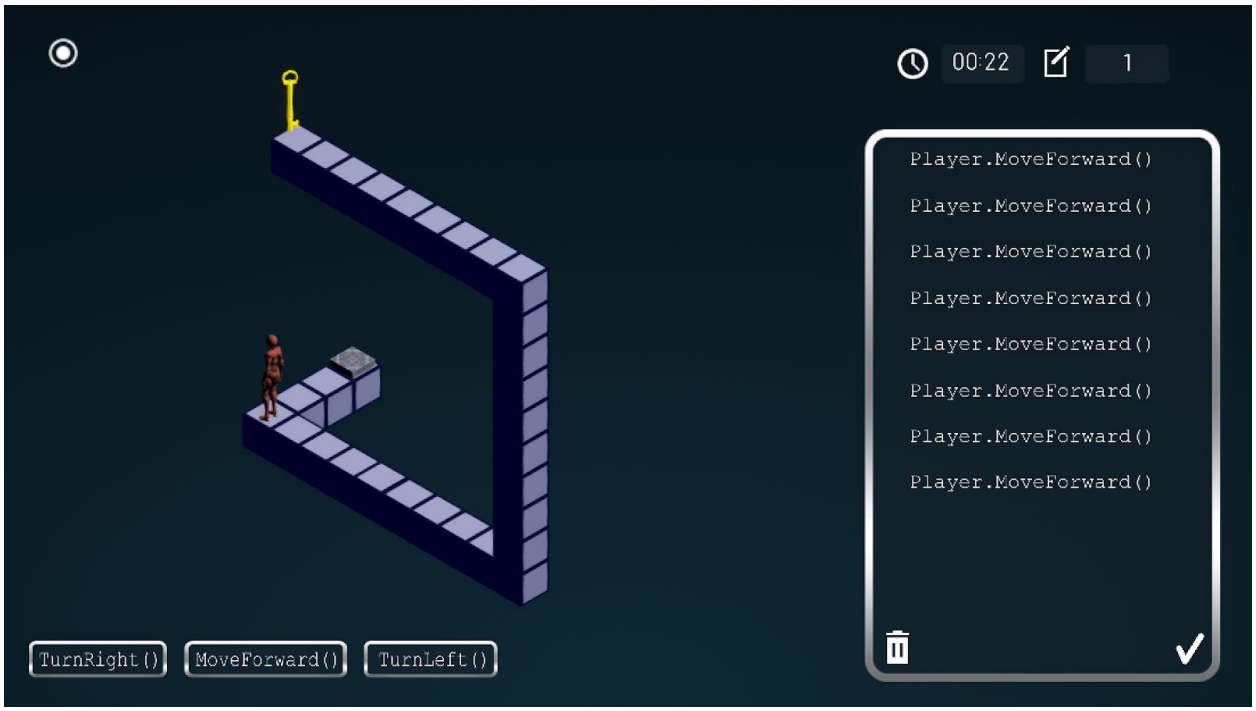

Username
Password
Forgot Password
Login
Don't have an account? SignUp
Copyright © 2018 Florida International University
GameSparks Preview mode

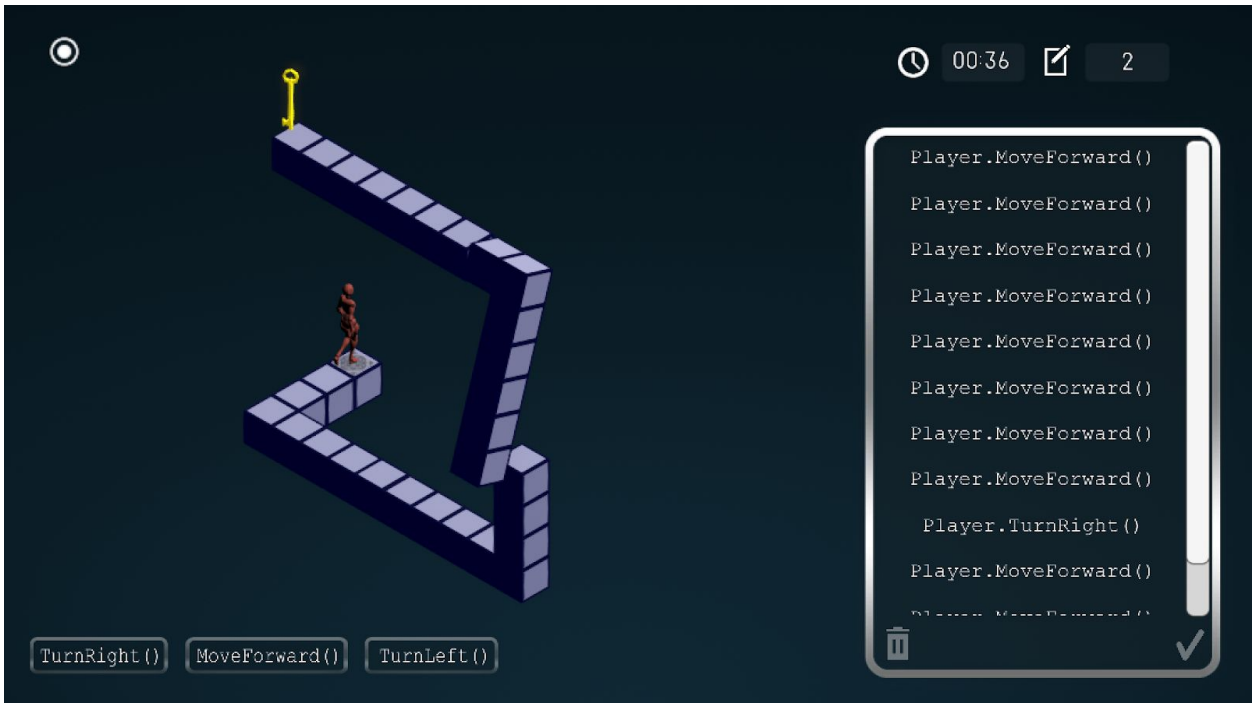

00.00
Welcome to AR-VR-VE for Computer Science education. Let me walk you through the game. Click the check mark to continue
TurnRight()
MoveForward()
TurnLeft()

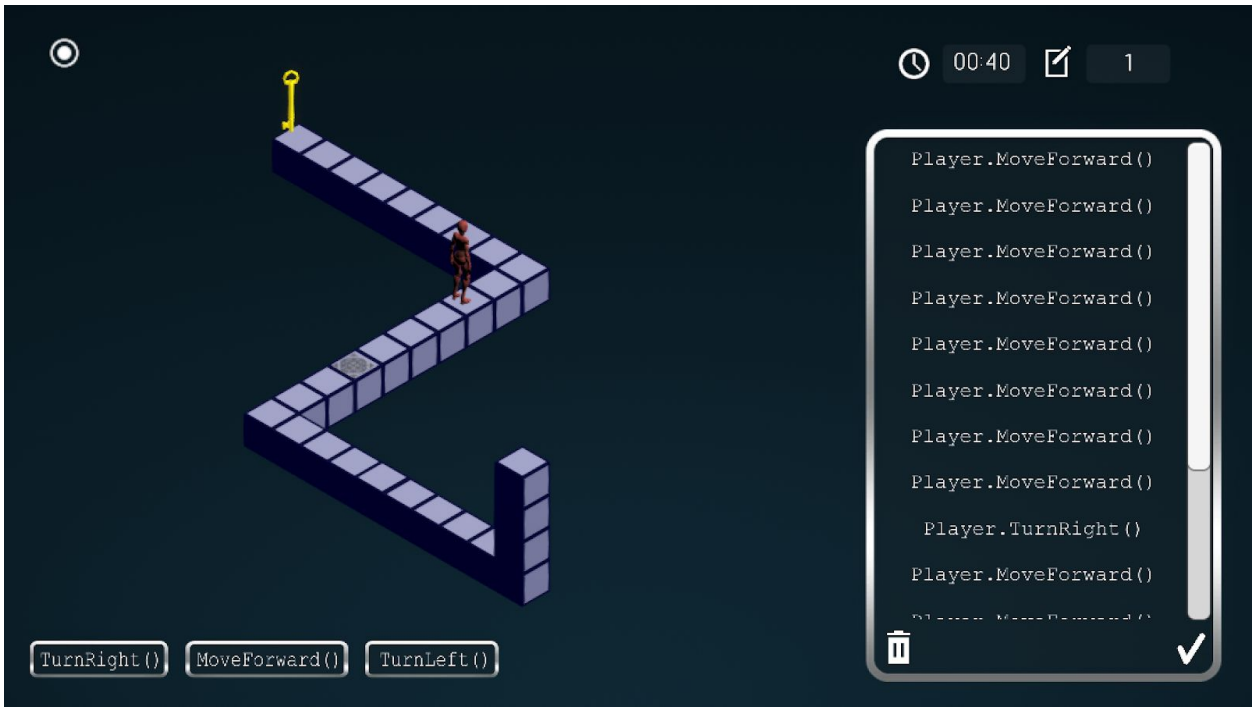

Your goal is to help the avatar get the key! I'll give you some tips!
00.00
TurnRight()
MoveForward()
TurnLeft()


00.00
Let's get started!
Good luck!
TurnRight()
MoveForward()
TurnLeft()


00.00
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
TurnRight()
MoveForward()
TurnLeft()


00:22
1
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
TurnRight()
MoveForward()
TurnLeft()


00:36
2
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.TurnRight()
Player.MoveForward()
TurnRight()
MoveForward()
TurnLeft()


00:40
1
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.TurnRight()
Player.MoveForward()
TurnRight()
MoveForward()
TurnLeft()


01:16
2
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.MoveForward()
Player.TurnRight()
Player.MoveForward()
Congratulations! You have succesfully completed the game! You now understand how a computer digests instructions! Go try a programming class now!
EXIT
TurnRight()
MoveForward()
TurnLeft()

Appendix C - Sprint Review Reports

Sprint 2 Retrospective Meeting Minutes

Attendees: < Mairim Barrios
Fidel Hernandez
Arelys Alvarez
Carlos Martinez
Samira Tellez
>
Start time: <1:00pm>
End time: <1:15pm>

What went wrong?

- Did we do a good job estimating our team's velocity?
    - No, we underestimated the time needed to complete the user stories.
- Did we do a good job estimating the points (time required) for each user story?
    - No, we underestimated the time needed to complete the user stories.
- Did each team member work as scheduled?
    - Yes, everyone worked on their respective user stories every day for the scheduled times.

What went right?

- The team communicated very well and everyone cooperated on the agreed tasks.

How to address the issues in the next sprint?

- How to improve the process?
    - No need for improvements as far as the working process.
- How to improve the product?
    - Everyone has been working on their features very well.

Sprint 3 Retrospective Meeting Minutes

Attendees: < Mairim Barrios
Fidel Hernandez
Arelys Alvarez
Carlos Martinez
Samira Tellez
>
Start time: <9:00pm>
End time: <9:15pm>

What went wrong?

- Did we do a good job estimating our team's velocity?
    - Yes, we were able to complete the user stories indicated for this sprint.
- Did we do a good job estimating the points (time required) for each user story?
    - Yes, we estimated the time needed to complete the user stories correctly.
- Did each team member work as scheduled?
    - Yes, everyone worked on their respective user stories every day for the scheduled times.

What went right?

- The team communicated very well and everyone cooperated on the agreed tasks.

How to address the issues in the next sprint?

- How to improve the process?
    - No need for improvements as far as the working process.
- How to improve the product?
    - Improve the layout of the levels and add user input validation.

Sprint 4 Retrospective Meeting Minutes

Attendees: < Mairim Barrios
Fidel Hernandez

Arelys Alvarez
Carlos Martinez
Samira Tellez
>

Start time: <9:00pm>
End time: <9:15pm>

What went wrong?

- Did we do a good job estimating our team's velocity?
    - Yes, we were able to complete the user stories indicated for this sprint.
- Did we do a good job estimating the points (time required) for each user story?
    - Yes, we estimated the time needed to complete the user stories correctly.
- Did each team member work as scheduled?
    - Yes, everyone worked on their respective user stories every day for the scheduled times.

What went right?

- The team communicated very well and everyone cooperated on the agreed tasks.

How to address the issues in the next sprint?

- How to improve the process?
    - No need for improvements as far as the working process.
- How to improve the product?
    - Improve the layout of the levels.

Sprint 5 Retrospective Meeting Minutes

Attendees: < Mairim Barrios
Fidel Hernandez
Arelys Alvarez
Carlos Martinez
Samira Tellez
>

Start time: <9:00pm>
End time: <9:15pm>

What went wrong?

- Did we do a good job estimating our team's velocity?
    - Yes, we were able to complete the user stories indicated for this sprint.
- Did we do a good job estimating the points (time required) for each user story?
    - Yes, we estimated the time needed to complete the user stories correctly.
- Did each team member work as scheduled?
    - Yes, everyone worked on their respective user stories every day for the scheduled times.

What went right?

- The team communicated very well and everyone cooperated on the agreed tasks.

How to address the issues in the next sprint?

- How to improve the process?
    - No need for improvements as far as the working process.
- How to improve the product?
    - We have continuously worked on improving the user experience.

Sprint 6 Retrospective Meeting Minutes

Attendees: < Mairim Barrios
Fidel Hernandez
Arelys Alvarez
Carlos Martinez
Samira Tellez
>

Start time: <9:00pm>
End time: <9:15pm>

What went wrong?

- Did we do a good job estimating our team's velocity?
    - Yes, we were able to complete the user stories indicated for this sprint.
- Did we do a good job estimating the points (time required) for each user story?
    - Yes, we estimated the time needed to complete the user stories correctly.
- Did each team member work as scheduled?
    - Yes, everyone worked on their respective user stories every day for the scheduled times.

What went right?

- The team communicated very well and everyone cooperated on the agreed tasks.

How to address the issues in the next sprint?

- How to improve the process?
    - No need for improvements as far as the working process.
- How to improve the product?
    - We have continuously worked on improving the user experience.

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents

Installation

- Download the latest version of blender HTTPS://WWW.BLENDER.ORG/DOWNLOAD/
- Download Unity in the download archive HTTPS://UNITY3D.COM/GET-UNITY/DOWNLOAD/ARCHIVE
- Get this version of the project from github and install in a local repository

Maintenance

- No maintenance is necessary

Shortcomings/Wishlist

- Having no previous experience with Unity/Blender, the team took longer than expected to feel fully comfortable with the technologies, which restrained us from going further in difficulty with the game.
- We wish we could have taught more complex principles of computer science through this game.