Video Game Design Specification

Parchments of Fate

Version 1.0

**Team:** Team POF

**Authors: A**bundo, Aaron Brent

**A**llosada, William Aristotle

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author(s)** |
| 02/16/20 | 1.0 | First draft | Abundo A.B.  Allosada W.A. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1. Overview 5](#_Toc392058027)

[1.1 Purpose of this document 5](#_Toc392058028)

[1.2 Scope of Description 5](#_Toc392058029)

[1.3 Definitions 6](#_Toc392058030)

[1.4 Referenced Documents 7](#_Toc392058031)

[2. Functional Overview 8](#_Toc392058032)

[2.1 Use Case List 8](#_Toc392058033)

[3. FUNCTIONAL DETAILS 9](#_Toc392058048)

[3.1 Use Case Diagrams 9](#_Toc392058049)

[3.1.1 Use Case Diagram for AI Controller 9](#_Toc392058050)

[3.1.2 Use Case Diagram for Menu Controller 10](#_Toc392058051)

[3.1.3 Use Case Diagram for Player Controller 11](#_Toc392058052)

[3.2 Robustness Diagram 12](#_Toc392058054)

[3.2.1 Player Controller Robustness Diagram 12](#_Toc392058055)

[3.2.2 AI Controller Robustness Diagram 13](#_Toc392058056)

[3.2.3 Menu Controller Robustness Diagram 14](#_Toc392058057)

[3.3 Class Diagram 15](#_Toc392058058)

[3.3.1 Main Controller Class Description 15](#_Toc392058059)

[3.3.2 AI Controller Class Description 16](#_Toc392058060)

[3.3.3 Menu Controller Class Description 16](#_Toc392058061)

[3.3.4 Player Controller Class Description 16](#_Toc392058062)

[3.4 Sequence Diagrams 17](#_Toc392058065)

[3.4.1 AI Movement Control Sequence 17](#_Toc392058066)

[3.4.2 Menu Control Sequence 18](#_Toc392058067)

[3.4.3 Player Attack Control Sequence 19](#_Toc392058068)

[3.4.4 Player Control Sequence 20](#_Toc392058069)

[3.4.5 PlayerMovement Sequence 21](#_Toc392058070)

[3.5 Activity Diagrams 22](#_Toc392058071)

[3.5.1 AI Character Activity Diagram 22](#_Toc392058072)

[3.5.2 Menu Control Activity Diagram 23](#_Toc392058073)

[3.5.3 Player Character Activity Diagram 24](#_Toc392058074)

# Overview

## Purpose of this document

There are countless top down arcade games available to be played. One of the best ones is known as Journey of the Prairie King (Is a minigame from Stardew Valley). For the purpose of this assignment, I will always refer to the game as Stardew minigame).

* [Stardew Valley](https://www.stardewvalley.net/)

You will learn more about this game during the course of this HW1 task. Indeed, your primary goal is to determine (a) the core functionality that must be provided by your system and (b) the numerous possible extensions. This document available today is minimal since the primary task of each team is to fill in the requisite details as required to complete the assignment.

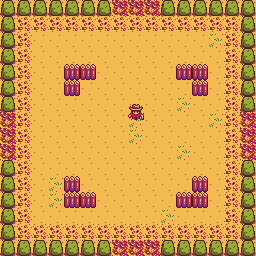


Figure 1: Sample Stardew Minigame layout

## Scope of Description

Parchments of Fate is a game where the player fights off hordes of enemies in an increasing difficulty and numbers. The player will move around a specified location where enemies will spawn around the area. This is a single-player video game where the character you play is the protagonist. The specific rules of the game shall be made at a later date. To keep the game as simple as possible there will not be a network version of the game. The single-player shall use a mouse and keyboard as his/her primary controlling tool.

## Definitions

* **Model** – A schematic description of a system that accounts for its known or inferred properties [http://www.answers.com].
* **View** -A visual representation of a model which might also enable a user to manipulate properties of the model.
* **Game** -A game is an instance of Parchments of Fate.
* **Sandbox** – The location where the player plays.
* **Variation** - A rule designed to be in effect during game play. There are a [number of potential variations](http://www.angelfire.com/wizard/sandshrew777/faq-oxfest.html) that you must consider.
* **Player** - Any user that interacts with the system.
* **System** - Any interactions performed by the application are considered to be performed by the system.
* **Score** - The number of points earned by each player against other players in the game.
* **Submit** – The action of the player indicating that they have placed tiles in a configuration for which the player wishes to receive points.
* **Restart** - The action of the player indicating that they want to restart the game.
* **GUI** – Graphical User Interface

## Referenced Documents

* Start Button   
  [http://pixelartmaker.com/art/397185d2d931676.png]
* HP icon   
  [https://www.shutterstock.com/video/clip-22752997-pixel-art-retro-game-style-red-hearts]
* Slime   
  [http://pixelartmaker.com/art/34e3be4ae06fbef.png]
* Reset Button   
  [http://pixelartmaker.com/art/191cdb8274cfea9]
* Main Heroine   
  [http://pixelartmaker.com/art/f392a4176916b51.png]
* Stardew Minigame Reference [https://stardewvalleywiki.com/Journey\_of\_the\_Prairie\_King]
* Tentacle Monster   
  [https://www.istockphoto.com/vector/vector-pixel-art-cartoon-gm813145798-131647733]
* Tree   
  [https://www.redbubble.com/people/p04i/works/24275043-pixel-art-tree?p=spiral-notebook]
* Grass Fields   
  [https://opengameart.org/forumtopic/grass-help]
* Start and Exit Button   
  [http://pixelartmaker.com/art/c1b594dc5a6c8ce]
* Top down 2d Unity Basics   
  [https://learn.unity.com/tutorial/live-sessions-on-2d-in-unity-4-3]
* Mountains   
  [http://pixelartmaker.com/art/442f9bc1fdcbf54]

# Functional Overview

This section gives a quick peek on each functions of Parchments of Fate. The following figures/diagrams are the functions of Parchments of Fate.

## Use Case List

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Name** | **Description** |
| 1.1.1 | Go forward | Make the Player Character move forward |
| 1.1.2 | Go backward | Make the Player Character move backward |
| 1.1.3 | Go left | Make the Player Character move left |
| 1.1.4 | Go right | Make the Player Character move right |
| 1.2 | Attack | Make the Player Character attack |
| 2.1 | Spawn Enemy | Spawns enemies to attack the Player Character |
| 2.2 | Follow Player Character | Make enemies follow Player Character |
| 3.1 | Start Game | Starts the game session |
| 3.2 | Quit Game | Ends and exits the game |

**Table 2.1 Use Case List**

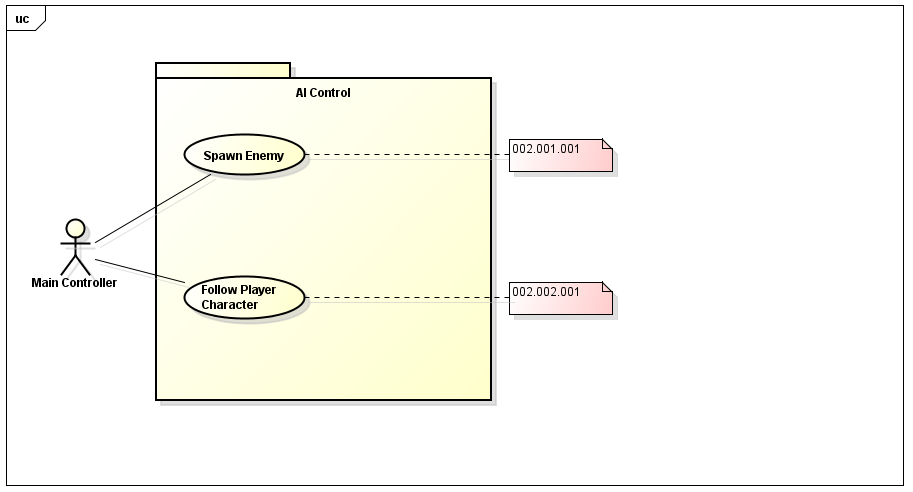
Table 2.1 shows the Parchments of Fate Use Case List. It shows a list of functions that the game does.

# FUNCTIONAL DETAILS

## Use Case Diagrams

Parchments of Fate Use Case Diagrams are based on the functional requirement of the game. The Use Case Diagrams of each component (AI controller, Menu controller, and Player Controller) are shown below.

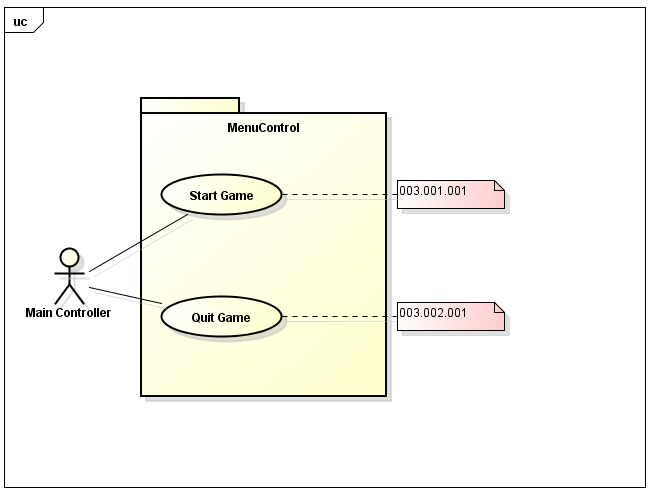
### Use Case Diagram for AI Controller



**Figure 3-1 AI Controller Use Case Diagram**

Figure 3-1 shows the Use Case Diagram for the AI Controller. The use cases are the AI functions that the Enemy spawns and follows the Player Character. The notes refer to the corresponding Requirement that particular Use Case has met.

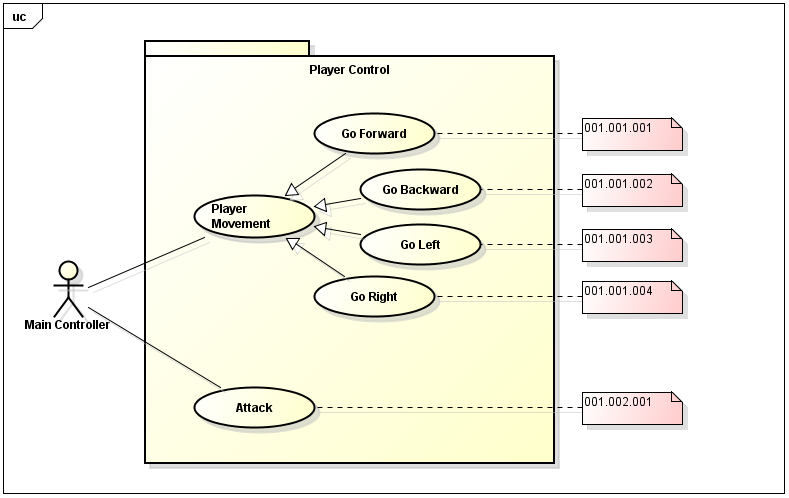
### Use Case Diagram for Menu Controller



**Figure 3-2 Menu Controller Use Case Diagram**

Figure 3-2 shows the Use Case Diagram for the Menu Controller. The use cases are the different menus that are to be selected to Start Game or Quit Game. The notes refer to the corresponding Requirement that particular Use Case has met.

### Use Case Diagram for Player Controller



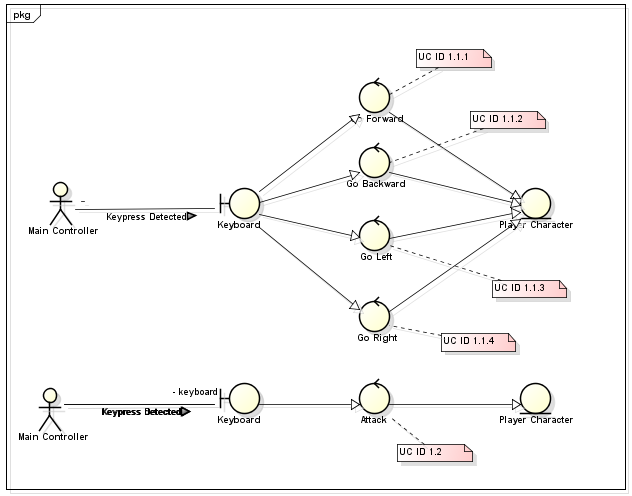
**Figure 3-3 Player Controller Use Case Diagram**

Figure 3-3 shows the Use Case Diagram for the Player Controller. The use cases are the movements and attacks that the Player should do to traverse the whole map. The notes refer to the corresponding Requirement that particular Use Case has met.

## Robustness Diagram

The Robustness Diagram for PlayerController, AIController and MenuController are shown below.

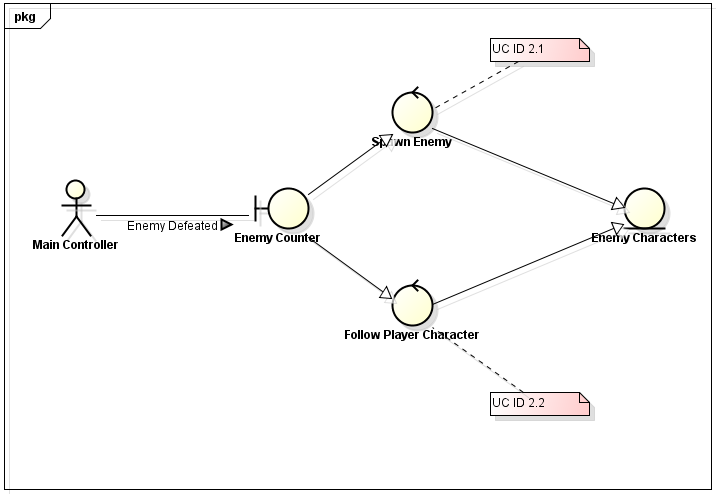
### Player Controller Robustness Diagram



**Figure 3-5 Player Controller Robustness Diagram**

Figure 3-5 shows the Robustness Diagram for the Player Controller with the Keyboard. This shows that boundary between the movements and the main controller is the keyboard. As for setting the Player to attack, Attack button must be pressed. The notes refer to the corresponding Use Case that particular controller represents.

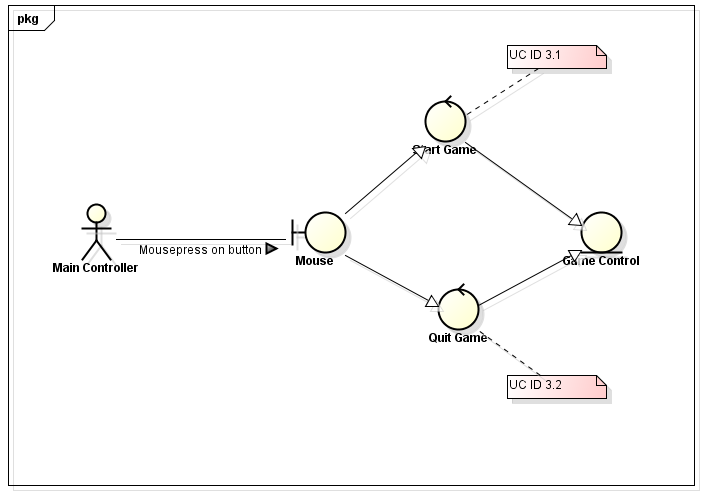
### AI Controller Robustness Diagram



**Figure 3-6 AI Controller Robustness Diagram**

Figure 3-6 shows the Robustness Diagram for the AI Controller. This shows the actions of the Enemy in its AI. The notes refer to the corresponding Use Case that particular controller represents.

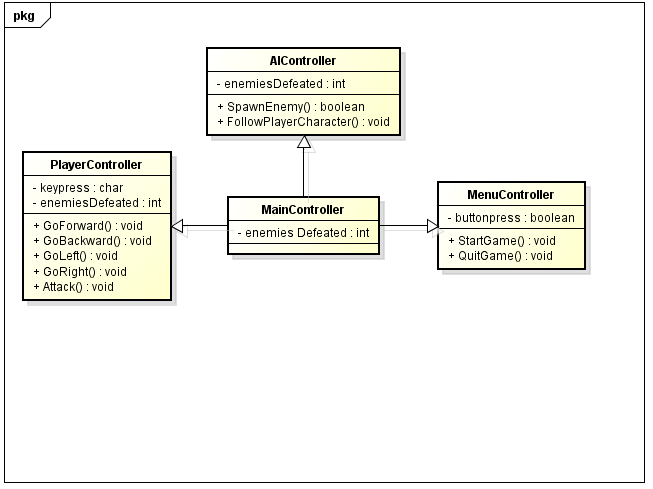
### Menu Controller Robustness Diagram



**Figure 3-7 Menu Controller Robustness Diagram**

Figure 3-7 shows the Robustness Diagram for the Light Indicator. This shows that the mouse pointer interacting the menu buttons of the game. The notes refer to the corresponding Use Case that particular controller represents.

## Class Diagram

The Class Design Diagram for Parchments of fate shows the necessary methods and attributes of the five classes (MainController, AIController, MenuController, and PlayerController) and its association with each other to perform its specific task. Each class represents one major component of Parchments of Fate. 

**Figure 3-8 Parchments of Fate Design Class Diagram**

Figure 3-8 shows the Design Class Diagram for Parchments of Fate. The description of each class will be shown in the subsections 3.3.1 to 3.3.6.

### Main Controller Class Description

The table below are the list of methods in the MainController class and its description.

|  |  |
| --- | --- |
| Methods | Description |

**Table 3.1 MainController Class’s Methods with Description**

### AI Controller Class Description

The table below are the list of methods in the AIController class and its description.

|  |  |
| --- | --- |
| Methods | Description |
| Boolean SpawnEnemy() | This spawn the enemy in the field. |
| Void FollowPlayerCharacter() | This lets the enemy follow the player. |

**Table 3.3 AIController Class’s Methods with Description**

### Menu Controller Class Description

The table below are the list of methods in the MenuController class and its description.

|  |  |
| --- | --- |
| Methods | Description |
| Void Start Game() | This starts the game |
| Void Quit Game() | This quits the game |

**Table 3.5 MenuController Class’s Methods with Description**

### Player Controller Class Description

The table below are the list of methods in the LightIndicator class and its description.

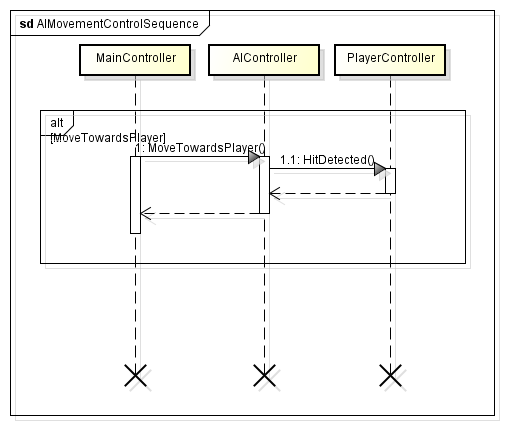
|  |  |
| --- | --- |
| Methods | Description |
| Void GoForward() | Moves Forward |
| Void GoBackward() | Moves Backward |
| Void GoLeft() | Moves Left |
| Void GoRight() | Moves Right |
| Void Attack() | Attacks the enemy |

**Table 3.7 LightIndicator Class’s Methods with Description**

## Sequence Diagrams

The sequence diagram for the AI Movement Control, Menu Control, Player Attack Control, Player Control, and PlayerMovement are shown below. Each function used in each sequence then calls the necessary API’s it needs.

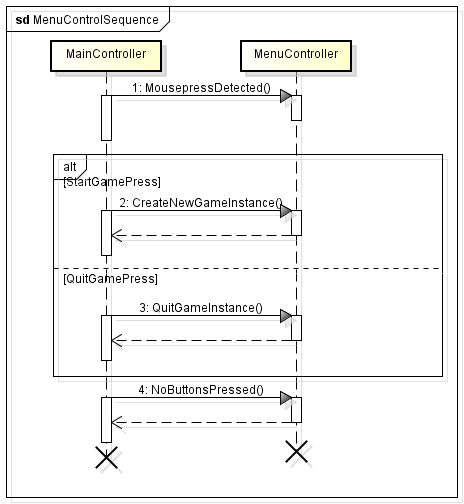
3.4.1 AI Movement Control Sequence



**Figure 3-9 AI Movement Control Sequence Diagram**

Figure 3-9 shows the Design Sequence Diagram for the AI movement of the Enemy. This is the sequence when the Goal Point has been detected.

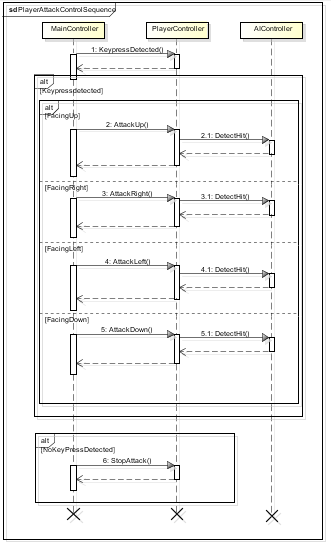
3.4.2 Menu Control Sequence



**Figure 3-10 Menu Control Design Sequence Diagram**

Figure 3-10 shows the Design Sequence Diagram for the Menu of Parchments of Fate.

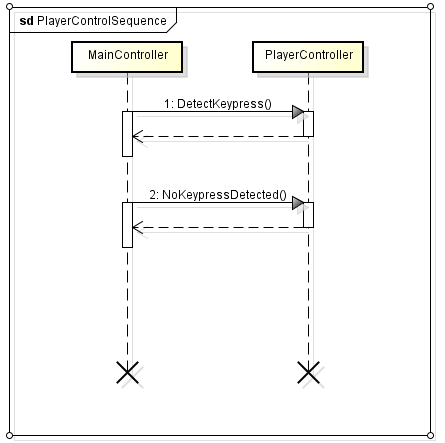
### 3.4.3 Player Attack Control Sequence



**Figure 3-11 Player Attack Control Design Sequence Diagram**

Figure 3-11 shows the Design Sequence Diagram for the Player Attack Control of Parchments of Fate. This is the sequence for determining if the player attacks directionally.

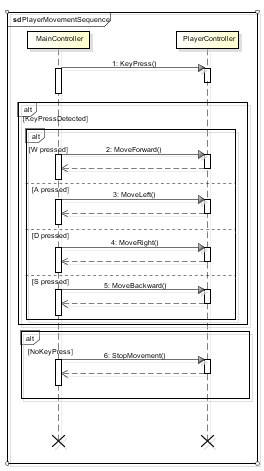
### 3.4.4 Player Control Sequence

****

**Figure 3-12 Player Control Design Sequence Diagram**

Figure 3-12 shows the Design Sequence Diagram for the Player Control of Parchments of Fate. This is the sequence for determining which key button did the player pressed or not.

### 3.4.5 PlayerMovement Sequence

****

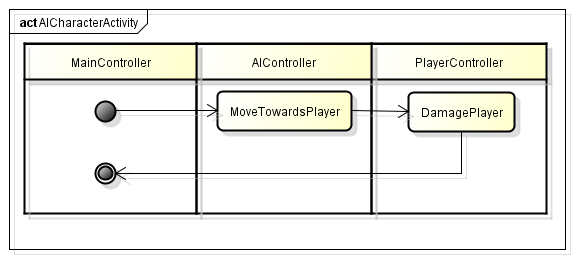
**Figure 3-13 PlayerMovement Design Sequence Diagram**

Figure 3-13 shows the Design Sequence Diagram for the PlayerMovement of Parchments of Fate. This is the sequence for determining which direction the player moves.

## Activity Diagrams

The Activity Diagram for AI Character, Menu Control, Traverse, and Player Character are shown below. These diagrams will show the details of the sequence for each function.

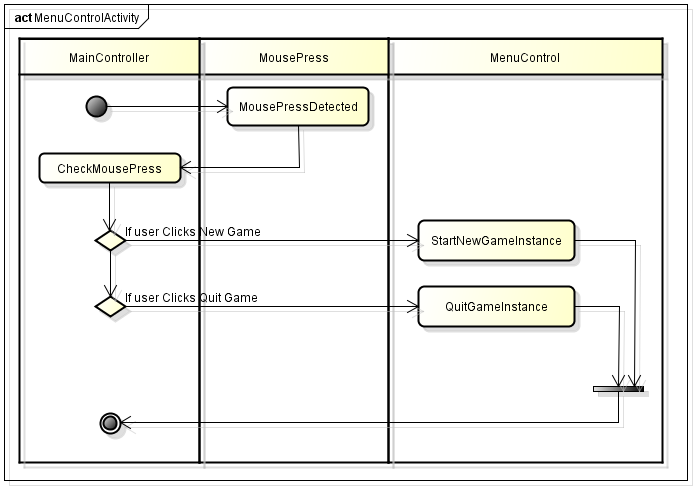
### AI Character Activity Diagram



**Figure 3-14 AI Character Activity Diagram**

Figure 3-14 shows the Activity Diagram for the AI of Parchments of Fate. This shows the corrective actions for the enemy to move towards and damages the player.

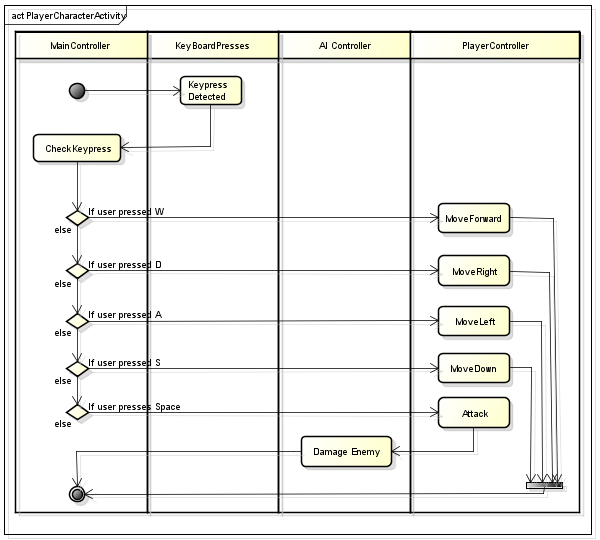
### Menu Control Activity Diagram



**Figure 3-15 Menu Control Activity Diagram**

Figure 3-15 shows the Activity Diagram for the Menu Control of Parchments of Fate. This shows if the mouse press is detected and checks if the user clicks New Game or Quit Game.

3.5.3 Player Character Activity Diagram

****

**Figure 3-16 Player Character Activity Diagram**

Figure 3-19 shows the Activity Diagram for the Player Character of Parchments of Fate. This shows which key did the user pressed in order to execute actions.