Software Requirements Specification

for

Agile Dungeon

Version 1.1e

Prepared by Jaeke Barkin, Kenneth Lu Diaz, John Lemme, Oliver McLaughlin, Noah Shelton, Carl Stoker, Alfred Timperley

URI CSC 305 Group H

2021-07-02

Table of Contents

Table of Contents ii

Revision History iii

1. Introduction 1

1.1. Purpose 1

1.2. Document Conventions 1

1.3. Project Scope 1

1.4. References 1

2. Overall Description 1

2.1. Product Perspective 1

2.2. User Classes and Characteristics 1

2.3. Operating Environment 2

2.4. Design and Implementation Constraints 2

2.5. Assumptions and Dependencies 2

3. System Features 2

3.1. Playable Character 2

3.1.1. Description 2

3.1.2. Stimulus/Response Sequences 2

3.1.3. Functional Requirements 2

3.2. Enemy 3

3.2.1. Description 3

3.2.2. Stimulus/Response Sequences 3

3.2.3. Functional Requirements 3

3.3. Merchant 3

3.3.1. Description 3

3.3.2. Stimulus/Response Sequences 3

3.3.3. Functional Requirements 4

3.4. Map 4

3.4.1. Description 4

3.4.2. Stimulus/Response Sequences 4

3.4.3. Functional Requirements 4

3.5. Movement 5

3.5.1. Description 5

3.5.2. Stimulus/Response Sequences 5

3.5.3. Functional Requirements 5

3.6. Item 5

3.6.1. Description 5

3.6.2. Stimulus/Response Sequences 5

3.6.3. Functional Requirements 6

3.7. Weapon (Item) 6

3.7.1. Description 6

3.7.2. Stimulus/Response Sequences 6

3.7.3. Functional Requirements 6

3.8. Attack (Action) 7

3.8.1. Description 7

3.8.2. Stimulus/Response Sequences 7

3.8.3. Functional Requirements 7

4. Data Requirements 7

4.1. Logical Data Model 7

4.2. Data Dictionary 7

4.3. Reports 8

4.4. Data Acquisition, Integrity, Retention, and Disposal 8

5. External Interface Requirements 8

5.1. User Interfaces 8

5.2. Software Interfaces 8

5.3. Hardware Interfaces 9

5.4. Communications Interfaces 9

6. Quality Attributes 9

6.1. Usability 9

6.2. Performance 9

6.3. Security 9

6.4. Safety 9

6.5. [Others as relevant] 10

7. Internationalization and Localization Requirements 10

8. Other Requirements 10

Appendix A: Glossary 10

Appendix B: Analysis Models 10

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Carl Stoker | 2021-06-09 | Initial Version. Filled in all requirements for submission. | 0.1a |
| Oliver McLaughlin | 2021-06-11 | Changed / added to Carl’s work, fixed some issues in section 3. | 0.1b |
| Alfred Timperley | 2021-06-11 | Changed / Added to Carl’s work, fixed formatting issue, update table of contents | 0.1c |
| Carl Stoker | 2021-06-11 | Cleaned up version for submission to Brightspace. | 1.0 |
| Oliver McLaughlin, Alfred Timperley, et al. | 2021-07-02 | Added content to 3.x sections | 1.1d |
| Carl Stoker | 2021-07-02 | Filled in missing 1.x sections. Fixed formatting and layout issues in 3.x sections | 1.1e |
| Kenneth Lu Diaz | 2021-07-26 | Added content to sections 5.x and 6.x | 2.0 |
| Oliver McLaughlin | 2021-07-29 | Cleaned up; Worked on section 4.x | 2.1 |

# Introduction

## Purpose

The purpose of this document is to inform users about the specifications of our single player action-adventure game. The target audience of this document is developers, project managers, marketing staff, testers, and documentation writers.

## Document Conventions

This document adheres to standard Software Requirement Specification conventions. It does not contain any project-specific typographical conventions or standards.

## Project Scope

The software’s purpose is to be an entertaining and fun top-down action-adventure game. Its main features will include multiple areas, enemies, items, and an inventory system. The player will traverse a randomly generated dungeon, finding loot and slaying enemies.

## References

This Software Requirements Specification does not refer to any external sources.

# Overall Description

This project is a top-down action-adventure game in the same vein as the old-school Legend of Zelda games. The game will be run on desktop computers and its target audience is English-speaking players. The constraints for this project will be the requirement that the computer have a version of python running. Its main dependency is the pygame engine.

## Product Perspective

This game is an entirely new product which is heavily influenced by classic games such as the Legend of Zelda NES games. This game will be under the umbrella of two-dimensional dungeon crawlers and will be similar to many games in this category.

## User Classes and Characteristics

This product is targeted to the general public. The following are types of user classes that are anticipated to use this product:

1. Casual Gamer: The favored user class and who the product will be designed to primarily accommodate. This user will use our product for their own enjoyment and play the game to maximize their individual fun.
2. Speed Runner: The product will not be designed to accommodate this player however they will not be excluded from the vision. This user will use our product for their own enjoyment and will play the game to maximize their speed of completion.

## Operating Environment

The target environment for this software is any computer running python on a x86\_64 system. However, it may also run (unsupported) on other architectures with pygame support, such as ARM. This product will run locally and not be dependent on geographical location or internet connection of the user to functional correctly.

## Design and Implementation Constraints

The primary constraint will be time, due to this project having to be completed in under ten weeks. Another constraint will be licensing and art, since there are limited pixel art packs available for noncommercial use.

## Assumptions and Dependencies

The primary dependency for this project will be the pygame module. Another dependency for this project will be the non-commercial pixel art we use in the project.

# System Features

## Playable Character

### Description

### High Priority: A playable character with the ability to attack, take damage, pick up new items, and move around the map.

### Stimulus/Response Sequences

**Stimulus:** The user uses the keyboard and mouse to play the game. **Response:** The player moves and interacts according to the player’s inputs.

### Functional Requirements

#### Character can attack enemies

#### Character can take damage from enemies

#### Character can pick up and equip items

## Enemy

### Description

As a game player, I need to have challenging enemies to fight so that the game is more enjoyable with a proper difficulty scale (High priority).

### Stimulus/Response Sequences

**Stimulus:** The player enters a room with enemies. **Response:** The enemy will approach the player.

**Stimulus:** The player is within range of the enemy's attack. **Response:** The enemy will attack the player.

**Stimulus:** The player attacks the enemy. **Response:** The enemy will take damage based on the player’s statistics and equipment.

**Stimulus:** The player defeats the enemy. **Response:** The enemy will drop an item based on random number generation and the type of enemy slain.

### Functional Requirements

#### Enemy knows if the player is in the same room.

#### Enemy can approach the player.

#### Enemy can attack the player.

#### Enemy can drop items.

## Map

### Description

As a game player, I need a playable area to fight enemies and retrieve items, providing an interactive world to explore. Maps must have one or more rooms.

### Stimulus/Response Sequences

**Stimulus:** The player enters a map. **Response:** A network of rooms renders, allowing the player to traverse an area with non-playable characters.

**Stimulus:** The player moves to another room. **Response:** The camera pans to another room stored in the Map.

### Functional Requirements

### 3.3.3.1 Rooms are populated with non-playable characters, along with items

### 3.3.3.2 Rooms are interconnected and the player can freely move between them

3.3.3.3 Maps are randomly generated

## Movement

### Description

The player, enemies, and non-playable characters will all need to be able to move around in various ways to attack, explore, and interact with other entities and the game world.

### Stimulus/Response Sequences

Player:

Stimulus: The user presses any of the W,A,S,or D keys. Response: The player character moves around and can interact with chests, obstacles, doors, enemies, and non-playable characters.

Enemy:

Stimulus: The player comes into a room with enemies. Response: The enemies act on their predefined movement patterns. These may include rigid linear movement, pathfinding, or random movement.

### Functional Requirements

#### The user must be able to move their player using the w/s/a/d keys.

#### The user must be able to collide with things in the game and be met with appropriate behavior (I.e. running through a door moves you to the next room).

#### Enemies must be able to move in either a set path, randomly, or by tracking the player.

#### The user must be able to attack while moving.

## Item

### Description

As a game player, I need to have items to help beat challenging enemies as well as add a variety of ways for the player to defeat enemies.

### Stimulus/Response Sequences

**Stimulus:** The player finds the item, or the player kills an enemy and gets an item. **Response**: The player will be able to pick up the item.

**Stimulus:** The player equips an item. **Response:** Depending on the item category, the player will receive a buff in a respective stat.

**Stimulus:** The player discards or drops an item. **Response:** The item disappears from the player’s inventory and cannot be used unless picked up or bought again.

### Functional Requirements

3.5.3.1 Items change the players base stats (hit points, defense, attack, etc.)

3.5.3.2 When consumable items are used, they apply a stat change on the player for a predetermined lot of time.

3.5.3.3 When equipable items are used, they take up a predetermined slot and cannot be stacked (ex. Can't equip two boots at the same time)

## Weapon

### Description

As a game player, I need weapon items to fight enemies and progress through the game fighting more difficult enemies as the game proceeds.

### Stimulus/Response Sequences

**Stimulus**: Player Picks up weapon **Response:** Weapon is equipped to the player and attack stat changes

**Stimulus:** Player presses attack button **Response:** Weapon is fired.

### Functional Requirements

3.6.3.1 Damage varies between weapons

3.6.3.1 Weapons can be equipped

## Attack (Action)

### Description

As a game player, I need to be able to attack and be attacked so that the game has difficulty and requires skill so that the experience is more rewarding.

### Stimulus/Response Sequences

#### Enemy

**Stimulus:** The enemy has a weapon within range to attack **Response:** The enemy attacks the player with either range or melee attack

#### Player

**Stimulus:** The player inputs an attack. **Response:** The current weapon they are holding is used in an attack in the current direction their mouse is pointing.

#### Functional Requirements

#### 3.7.2.1 Attack has a range of effect.

#### 3.7.2.2. Attack dissipates once it is outside of its range of effect.

#### 3.7.2.3. Attack has a speed.

#### 3.7.2.4 Attack has a hitbox.

# Data Requirements

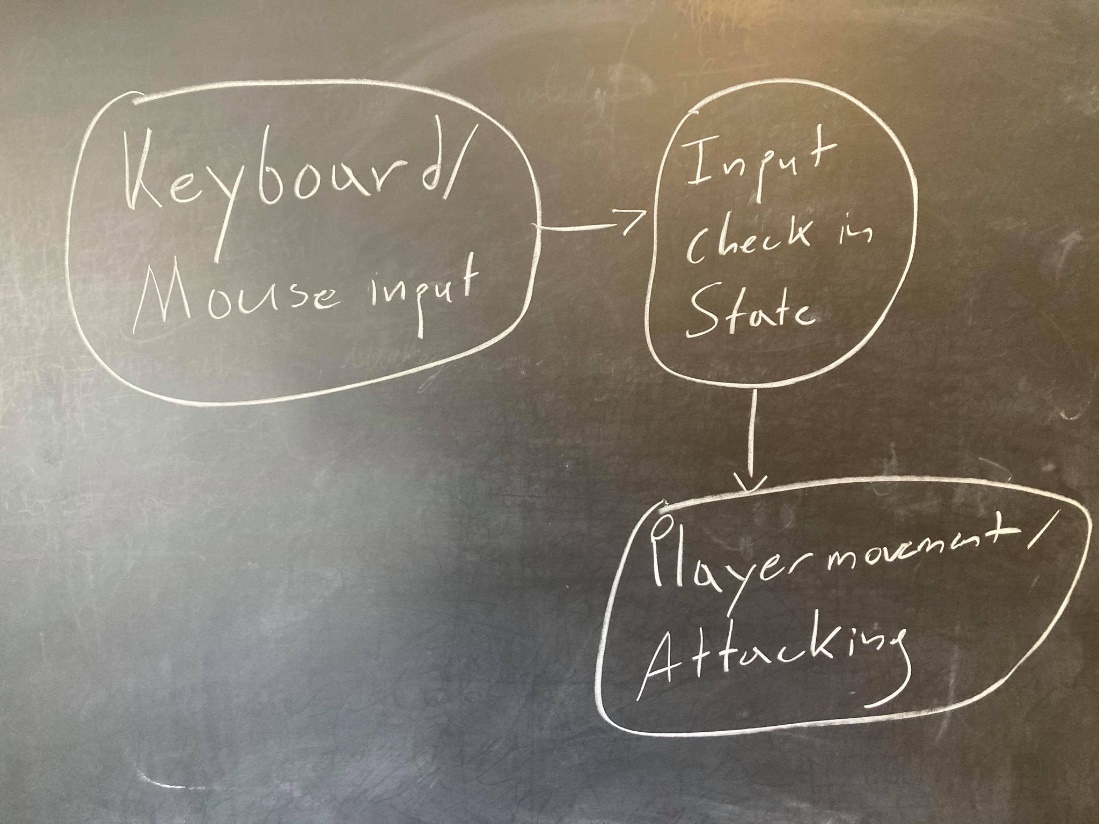
Data inputs:

* User keyboard input
* User mouse input

Data outputs:

* High scores

## Logical Data Model



## Data Dictionary

Not applicable

## Reports

High scores will be recorded in a file located in the project directory.

## Data Acquisition, Integrity, Retention, and Disposal

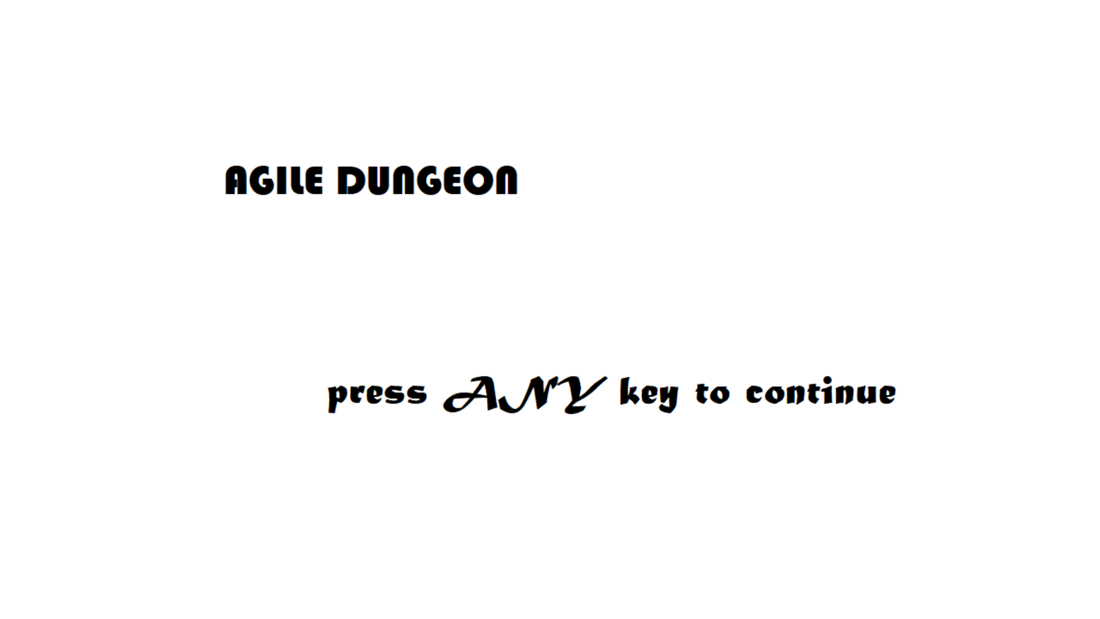
No data other than user input will be collected. No external data will be stored.

# External Interface Requirements

In order to effectively play this game, a user must have access to a monitor, a keyboard, and a mouse or touchpad.

## User Interfaces

The game must utilize the maximum screen size available, in order to make the room and UI components clearly visible. On startup the user is prompted to enter any key in order to start the game.



Upon entering any key, the user must be able to see a screen with his character in a room. The user will also need to see information about the status of the playable character, such as health, current weapon, statistics, inventory, and floor level. All GUI components must utilize the **Macondo** font, retrieved from Google Fonts and stored inside the project repository.



## Hardware Interfaces

The user should only require a keyboard and mouse device to control the playable character on screen.

User Controls:

* Keyboard Keys:
  + W : Moves Character UP
  + S : Moves Character DOWN
  + A : Moves Character LEFT
  + D : Moves Character RIGHT
  + 1 : Uses item in the 1st item slot
  + 2 : Uses item in the 2nditem slot
  + 3 : Uses item in the 3rd item slot
  + 4 : Uses item in the 4th item slot
  + 5 : Uses item in the 5th item slot
* Mouse device:
  + Hover : The user begins to aim their weapon towards the current location of the mouse
  + Left-click : The user fires their current weapon

# Quality Attributes

## Usability

The game must utilize a readable font with characters larger than 12px, for communicating the current state of the playable character. The game must also use simple, common controls for controlling the playable character. If a control does not necessarily follow the previous description, then the monitor must display on screen the keys to press in order to operate the corresponding function. The game should clearly notify the player if they are taking damage or are in danger of losing the game. The game must be system tested for difficulty.

## Performance

The game operates at 60 frames per second (fps), so it should be very unlikely to notice lag between the moment the correct controls are pressed and their corresponding functions appearing on screen

## Security

This program only takes in user input. There are no security requirements.

## Safety

Similarly with security, there are no safety requirements.

## Difficulty

The game should be able to be completed by the average player.

# Internationalization and Localization Requirements

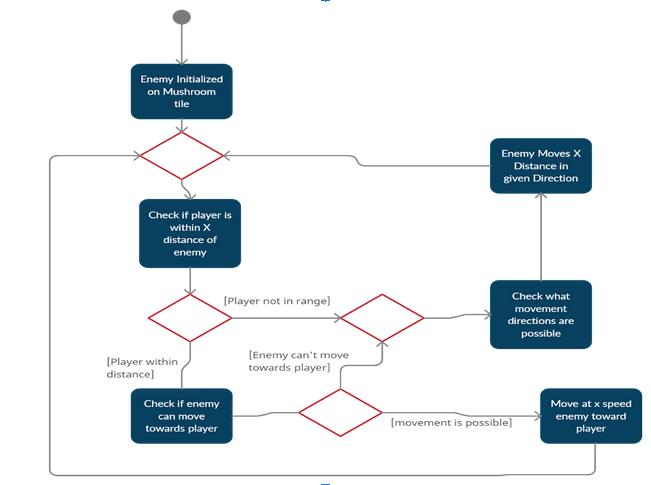
This game is intended for an English-speaking audience. We do not have any localization requirements.

# Appendix A: Glossary

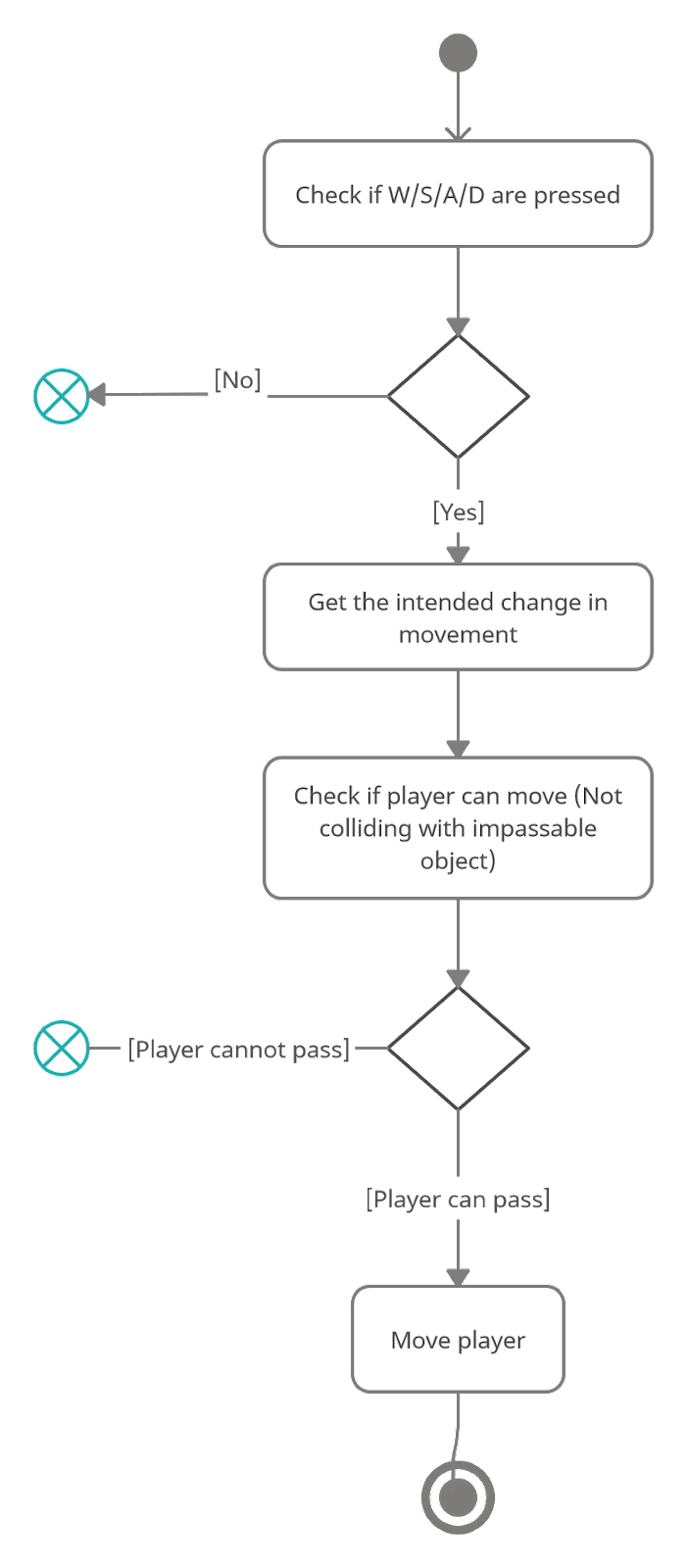
Dungeon Crawler: A type of video game in which the player traverses a “dungeon”. Typically, these games incorporate randomly generated maps.

Appendix B: Analysis Models

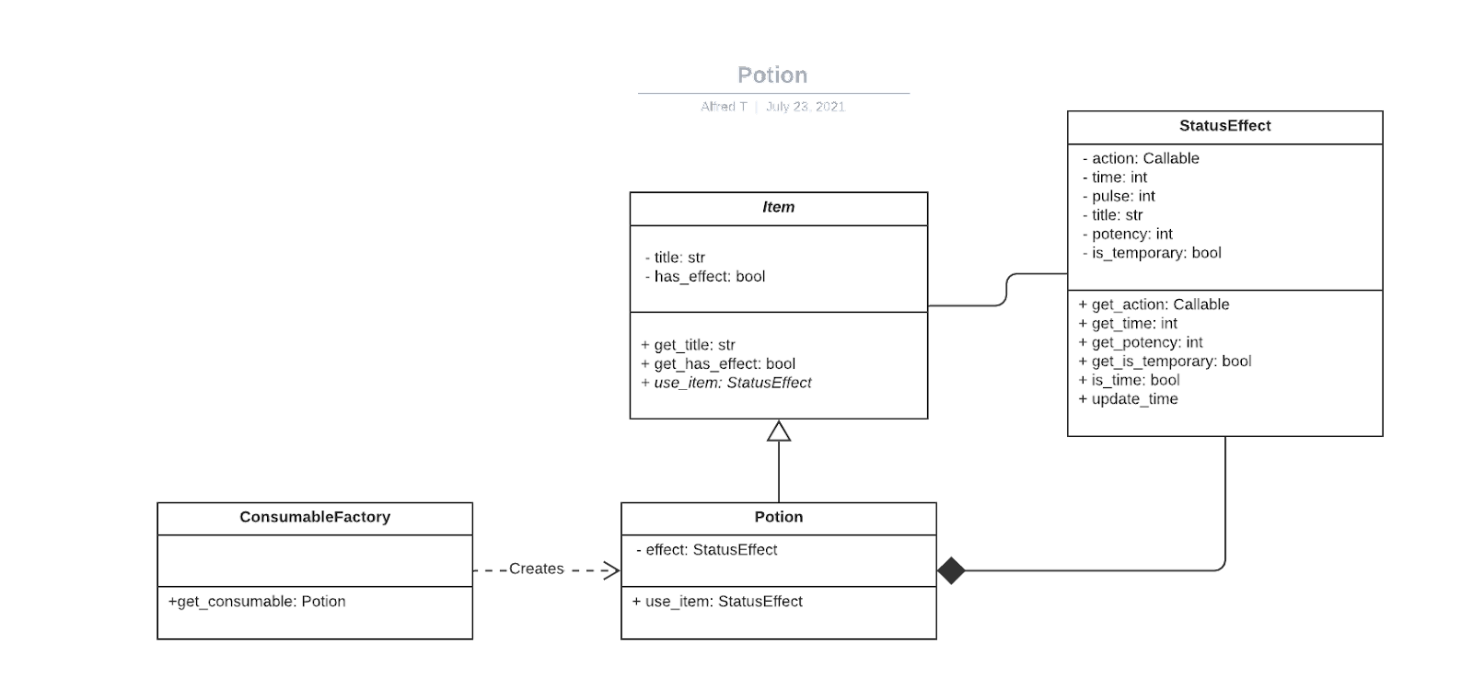
Enemy Movement Diagram



Player Movement Sequence



Potion class diagram



Attacking diagram

