SE 3XA3: Software Requirements Specification
Specification
Zombie Survival Kit

Group 6ix

Brian Jonatan, jonatans
Mohammad Hussain, hussam17
Shivaansh Prasann, prasanns

November 28, 2018

Contents

List of Tables

List of Figures

Table 1: **Revision History**

Date	Version	Notes
October 4, 2018	1.0	Added first changes to requirements doc
October 5, 2018	1.1	Initial Commit
November 28, 2018	2.0	Updated for Final copy

1 Project Drivers

1.1 The Purpose of the Project

~~Creating an FPS game is not an easy task, and many aspiring game developers pick up a game engine but are unable to finish their projects due to the huge learning curve.~~ The purpose of Zombie Survival Kit is to create an easier experience for aspiring game developers who want to develop a zombie survival game, by laying the base foundation for developing such a game. The base foundation includes all the functional requirements in section 2.2. By having the foundation completed, game developers can spend less time on developing the functionalities of the game and delve deeper into their own design and development decisions (E.g. Incentive of the game, level creation, story, etc.)

Zombie Survival Kit aims to support aspiring game developers by providing a fully customizable and easy to use starting ground for them to create the FPS game they dream of. Zombie Survival Kit shall also introduce video game players to game development.

1.2 The Stakeholders

1.2.1 The Client

The clients for Zombie Survival Kit ~~as of this moment~~ are ~~the group members of Team 6~~ Group 6ix, as well as the instructor and teaching staff of SFWR ENG 3XA3.

1.2.2 The Customers

The customers for Zombie Survival Kit ~~would~~ are be aspiring game developers.
Game developers would be using Zombie Survival Kit as a starting point to create their own First Person Shooter games.

1.2.3 Other Stakeholders

Other stakeholders include ~~video game players~~ users (1.4) and testers 1.4.
Game players will be using Zombie Survival Kit not only as an entertainment

product, but also as a tool to explore game development.
Testers will be given a survey to complete, and the survey will ask questions relating to the functional and non-functional requirements.

1.3 Mandated Constraints

1. **Input Constraints**: Only a keyboard and a mouse are to be used for input.

2. **Space Constraints**: Zombie Survival Kit shall not use more than 1 GB of space on the computer's hard drive.

3. **Pricing Constraints**: Zombie Survival Kit shall not cost more than $0.00

1.4 Naming Conventions and Terminology

1. **FPS**: Frames per second, a unit of measurement used to determine the smoothness of game visuals.
2. **Unity**: A game development framework which uses C# as the primary programming language and allows us to integrate media and animations into a project to create an interactive computer game.
3. **Users**: Developers and video game players who use Zombie Survival Kit.
4. **Developers**: Programmers who will use Zombie Survival Kit to make their own first person shooter game.
5. ~~Players: People who will play games made using Zombie Survival Kit.~~
5. **Player**: Refers to the in-game player object that the users can control through mouse and keyboard inputs.
6. **GPU**: A piece of hardware on a computer used to render graphics on a computer.
7. **VRAM**: The amount of memory available to the GPU.
8. **Testers**: Refers to those testing the functional and non-functional requirements of Zombie Survival Kit. 9. **LMB**: Refers to left mouse button 10. **'inventory' key**: Refers to 'I' key on the keyboard 11. **'interact' key**: Refers to 'E' key on the keyboard 12. **'unequip all' key**: Refers to 'U' key on the keyboard 13. **'reload' key**: Refers to 'R' key on the keyboard

1.5   Relevant Facts and Assumptions

1. The computer should have a discrete GPU with at least 1 GB of GDDR5 VRAM.
2. The user should be at least 12 years old.
3. The computer should be running the Windows operating system, at least Windows 8.
4. The user has a keyboard and a mouse connected to computer.
5. The user has previous experience of playing PC games.
6. User's will be using Unity to gain access to Zombie Survival Kit

2   Functional Requirements

2.1   The Scope of the Work and the Product

The original developer's work will be used as ~~a~~ ground zero for the project to replicate some basic functionalities from each aspect ~~they~~ worked on. Coding style will be heavily improved, and the scripts will be changed to use a MVC (Model-View-Controller) model. Extra features, ~~items, or NPCs~~ and items will be added into ~~each part of~~ the game in order to make it more full/complete. ~~In general, the game will consist of a player (first-person perspective) dropped into an open world with various terrains that has different types of zombies that they must defeat to stay alive and get better gear, weapons, consumable items, and more.~~ Zombie Survival Kit will consist of the functional requirements listed below (2.2).

2.1.1   The Context of the Work

~~The Zombie Survival Kit open source project has all the individual components of a zombie survival game coded, but none of it is put together. The creator of the project developed the logic and UI for individual aspects such as a systems for combat, inventory, items, day/night, and enemy AI but there is no playable environment with all these parts working together. What Group 6 plans to do is re-work this game from the ground up and develop it how it was meant to be, as a finished playable product.~~

2.2   Functional Requirements

F1. The player must be able to move forward, left, down, and right using the WASD keys on the keyboard.
Fit Criterion:
The user will observe that the player is moving by having static objects in the scene (E.g. trees) showing that the distance of the static objects relative to the player changes.

F2. The player must be able to look in all directions by moving their mouse.
Fit Criterion:
The user will observe that the player is moving by having static objects in the scene (E.g. trees) showing that the location of the static objects on the user's screen changes when the user moves the mouse around once the scene has started.

F3. Zombie enemies must walk back and forth between random points in their 'spawn circle', which imitates them walking around.
Fit Criterion:
The user will observe that the zombies never leave the general area from which they are instantiated.

F4. Zombie enemies must start attacking the player when they come within a certain radius.
Fit Criterion:
The user will observe that once the player is at attacking distance from the zombie, the zombie's attack animation will begin and the player's health will deteriorate (which will be indicated by the player UI.

F5. Zombie enemies must follow the player if ~~they start running away while the zombie is attacking them~~ the player is at a certain distance from the zombie.
Fit Criterion:
The user will observe that even as the player moves away from the zombie, the zombie will continue to travel towards the player.

F6. If the player runs past a certain radius from the zombie's original spawn location, the zombie must return to their circle.
Fit Criterion:
The user will observe that once the player moves at a distance far enough

from te zombie's original spawn location, the zombie will stop following the player and travel back to their spawn location.

F7. Different types of zombies must have different statistics (health, damage, and attack speed). Different types of zombies will be distinguishable by their different appearances.
Fit Criterion:
The user will observe that some zombies will attack faster, where each attack deteriorates the player's health at a smaller scale (E.g. by a factor of 5 of 100 health points on the Player UI), and can kill a zombie in less hits (E.g. 3 hits from the player to kill the zombie). The user will also observe that some zombies will attack slower, where each attack deteriorates the player's health at a larger scale (E.g. by a factor of 10 of 100 health points on the Player UI), and is harder to kill (E.g. 5 hits from the player to kill the zombie).

F8. Zombies must have randomly generated chance of dropping items the player can ~~use~~ equip or consume when killed.
Fit Criterion:
The user will observe that once a zombie dies, an item will appear on the location of the zombie's death.

~~9. Dropped items must be automatically deleted if not picked up within a certain time~~

F9. The player must be able to pick up items they are looking at with the 'interact' key.
Fit Criterion:
The user will observe that if the 'interact' key is inputted when the player is close enough to an item, the item will disappear from the user's screen. Any item picked up will be stored in the player's inventory.

F10. The player must be able to access their inventory by pressing the 'inventory' key, and will have a maximum space of 20 items.
Fit Criterion:
The user will observe that if the 'inventory' key is inputted, the Inventory UI will appear on the user's screen.

F11. The player must be able to equip, consume, ~~delete, and move around~~

and drop items in their Inventory UI using the ~~mouse~~ LMB.
Fit Criterion:
The user will observe that if the LMB is used on an item icon, the item will be used. If the item is an equipment item, the item will appear on the Equipment UI; if the item is a consumable item, any deteriorated health will fill up and appear on the Player UI. If the LMB is used on the remove button, the item will dissappear from the Inventory UI and will appear in front of the player.

F12. The player must be able to unequip items in their Equipment UI using the LMB or the 'unequip all' key.
Fit Criterion:
The user will observe that if the LMB is used on the remove button in the Equipment UI, that item is removed and will reappear in the Inventory UI; if the 'unequip all' key is used, all equipped items are removed and will reappear in the Inventory UI. If there are no more open slots left in the inventory, the removed equipment items will appear in front of the player.

F13. The player must be able to fire/use ~~their equipped weapon~~ an equipped gun or axe by pressing the left mouse button (LMB).
Fit Criterion:
The user will hear gunshot or an axe slicing through the air when a gun or axe is equipped after pressing the LMB.

F14. If the player has a firearm equipped, ~~they must be able to aim down sights using the right mouse button (RMB)~~ the user can reload the gun using the 'reload' key. The mazimum amount of bullets will be 7.
Fit Criterion:
The user will observe that, if the number of bullets left in the gun is less than 7 (indidcated by the Player UI) and the gun is reloaded, the number of bullets in the gun will return to 7. If the number of bullets left in the gun is 7, no extra bullets are added into the gun.

F15. The environment must slowly go through a day and night cycle.
Fit Criterion:
The user will observe that the lighting of the scene changes as time passes.

F16. The player must lose health when hit by a zombie.

Fit Criterion:
The user will observe the player's health deteriorating indicated by the Player UI.

F17. Zombies must lose health when hit by the player by using the LMB once the player is within attacking distance from the zombie.
Fit Criterion:
The user will hear an attack sound when a valid attack has landed on the zombie.

F18. Players or zombies must die when they reach 0 health.
Fit Criterion:
If the player dies, a black screen will appear with the words "Game Over" indicating the player has died. When a zombie dies, the zombie undegoes a death animation before disappearing from the scene.

19. Upon player death, the game must reset

F19. An Inventory UI will give the user a visual representation of the player's inventory and will be accessable by pressing the 'inventory' key.
Fit Criterion:
The user will observe that once the 'inventory' key, the Inventory UI will be located at the top bottom right corner of their screen.

F20. An Equipment UI will give the user a visual representation of the player's equipped items and will be appear with the Inventory UI by pressing the 'inventory' key.
Fit Criterion:
The user will observe that once the 'inventory' key, the Equipment UI will be located at the top left corner of their screen.

F21. A Player UI will give the user a visual representation of the player's health and remaining bullets if a gun is equipped to the player. The Player UI will also consist of a reticle.
Fit Criterion:
The user will observe that the health and bullets will be located at the bottom left corner of their screen, while the reticle will be in the middle of the screen.

3 Non-functional Requirements

All Non-Functional Requirements will be validated through testers.

3.1 Look and Feel Requirements

NF1. The graphics of the game shall look very simplistic

NF2. The input commands from the keyboard and mouse shall appear easy to use

NF3. When playing the game and inputting commands through the mouse and/or keyboard, the game shall feel responsive.

3.2 Usability and Humanity Requirements

NF4. Zombie Survival kit shall be usable on a computer or laptop

NF5. Zombie Survival kit shall require minimal learning for command inputs through the keyboard and mouse

NF6. Zombie Survival kit is best used by users who speak English as the instructions on how to play the game will be provided in English.

3.3 Performance Requirements

NF7. When the user loads the executable to start the game, the time it takes to load the entire game shall not exceed more than 20 seconds.

NF8. The game shall run at a video refresh rate of 60 frames per second on computers with a GPU with at least 1GB of GDDR5 VRAM.

NF9. The time it takes the user to close the game shall not exceed more than 5 seconds.

NF10. Zombie Survival Kit will not take up more than 1gb of ram to run properly.

3.4   Safety Critical Requirements

NF11. If the amount of free space on the computer's hard drive required to download is lacking, the game shall not download and install onto the computer.

3.5   Precision Requirements

NF12. The collisions on the zombie that indicate the range in which a player's attack is successful shall ~~be very close to the actual animation of the zombie on screen.~~ cohere with the dimensions of the zombie. Assuming that the player is within attacking distance, if reticle is not on any part of the zombie, any attacks from the player to the zombie should not be valid.

NF13. The day and night system shall use the computer's system time to keep track of how much time has passed during the "day" and "night. Once the specified time has been reached, "day" will transition to "night", and "night" will transition to "day".

NF14. The range in which the player can pick up ~~a droppable~~ an item is very close to the animation of the ~~droppable~~ item; the distance in which an item can be picked up by the player will be intuitive to the user. ~~The item must also be very close to the center of the player's point of view.~~ The reticle must be located on the item.

3.6   Reliability and Availability Requirements

NF15. All user inputted commands through the keyboard or mouse should only produce that key's function and not a different key's function.
E.g. Pressing the "w" key on the keyboard to move the player forward will not cause the player to move backwards; which is inputted by pressing the "s" key on the keyboard.

NF16. Zombie Survival Kit will always be available for use so long as the computer or laptop is turned on

3.7 Operational and Environmental Requirements

3.7.1 Expected Physical Environment

NF17. The user is expected to use the application indoors on a computer or laptop.

NF18. Zombie Survival Kit shall use no more than 10 GB of space on the computer's hard drive.

NF19. The computer should have a discrete GPU with at least 1GB of GDDR5 VRAM for Zombie Survival Kit to run optimally.

3.7.2 Expected Technological Environment

The computer should be running with a Windows 8 or higher operating system.

3.7.3 Partner Applications

NF20. Scripts shall be coded using Visual Studio 2017

NF21. Objects, player, and zombie sprites shall be taken from Unity's asset store.

NF22. Developers must use Unity 3D v2018 in order to run Zombie Survival Kit.

3.8 Maintainability and Support Requirements

NF23. Zombie Survival Kit is expected to run independently from dataservices, allowing the user to run the game without an internet connection.

NF24. The scripts of Zombie Survival Kit shall incorporate modules for different functions in order to maintain functionality even if one module fails to run properly.

3.9 Security Requirements

NF25. Zombie Survival Kit shall not require any personal information from the user

NF26. Zombie Survival Kit shall require permission to access the user's hard drive to download and install onto the user's computer.

3.10   Cultural Requirements

Zombie Survival Kit shall not incorporate any symbols, sounds, or animations that are offensive to a variety of cultures or political backgrounds.

3.11   Legal Requirements

Zombie Survival Kit is licensed under the MIT license.

3.12   Health and Safety Requirements

NF27. Users with a history of epilepsy or sensitivity to flashing lights should not use this product.

NF28. The product should be used in a well lit environment

NF29. Correct posture should be ensured by the user while sitting on computer

NF30. After every hour of use, a 10 minute break is highly recommended.

NF31. If the user experiences watering of eyes, a sensation of dizziness or nausea, use of the product should be ceased immediately.

NF32. The product should never be used if the user is feeling sleepy.

NF33. The computer being used should be well ventilated and should be cooled adequately.

NF34. The frame rate should be at least 60 FPS at all times to prevent motion sickness, and frame time variance should be checked and maintained at a maximum of 16 ms.

NF35. The age of the user should be at least 12 years.

4 Project Issues

P1. Building the zombie AI so that the actions of the zombie before it notices a player are for it to roam around the playable map, not any farther from the spawn location of the zombie. However, determining how the zombie roams will be hard to implement.

P2. Implementing automated tests for each functional requirement will be difficult to produce, especially when most functional requirements are based on the gameplay of Zombie Survival Kit (which usually requires user inputted commands to make the player move, pick items up, etc.)

4.1 Open Issues

O1. Determining at what range from the player to the zombie that the zombie will begin to chase after the player

O2. Determining the range in which a collision happens between the player and objects in game, as well as the enemy zombie.

O3. Building the zombie AI so that the actions of the zombie before it notices a player are for it to roam around the playable map, not any farther from the spawn location of the zombie.

O4. Determining how long the "day" and "night status of the map will last before it becomes "night" or returns to "day" in the day/night system.

O5. Determining how long it takes for zombie to make a valid attack after the user has entered a specified range from the zombie.

O6. Determining the spawn rate of zombies during the day and during the night.

O7. Determining the depreciation rate of a player's "hunger" in the hunger

system.

O8. Determining the collision aspects of a zombie to indicate a valid player attack through ranged or melee attacks.

O9. Determining how far droppable items drop from the recently killed zombie.

O10. Determining how the rate at which certain items drop

O11. Determining what items can be dropped

O12. Determining how fast "walking" and "running" is in-game.

O13. Determining how fast a zombie will chase a player.

O14. Determining how long the zombie will chase a player before it "loses interest" and begins to roam around.

4.2 Risks

4.2.1 Technical Risks

T1. Zombie Survival Kit may strain the computer hardware, especially the graphics card and cause overheating and/or system crashes.
T2. The hardware on the computer being used to run Zombie Survival Kit may not be powerful enough to maintain the required frame rate of 60 FPS, and cause motion sickness and screen tearing.

4.2.2 Non Technical Risks

NT1. The developers may not be skilled enough to understand the details of the source code and templates used in Zombie Survival Kit.
NT2. The players may not find Zombie Survival Kit as sophisticated as AAA industrial-scale video games and may not find the product enjoyable.
NT3. The developers may want to create a game based on a genre not supported by Zombie Survival Kit.

References

James Robertson and Suzanne Robertson. *Volere Requirements Specification Template.* Atlantic Systems Guild Limited, 16 edition, 2012.

5 Appendix

This section has been added to the Volere template. This is where you can place additional information.

5.1 FAQs

Q: What minimum computer specs will I need to run the game?
A: CPU: Dual-core Intel i5 GPU: GTX 660/960M or equivalent RAM: 8GB
Storage: 10GB

Q: What language/framework is the project coded in?
A: It is made with Unity3D using C#

Q: Will the project be open source and be available for download online?
A: Yes, it will be available via a GitLab link