CM50109 Software Engineering

Sprint Documentation

Daniel Edwards

Yuru Luo

Jake Turner

Luke Worgan

Shuai Cheng

Dominika Nadia Wojtczak

git repository: https://github.com/Daniel-C-Edwards/Software-Engineering-Prince-Game/tree/delivery

Sprint 1: 06/11/2019 – 12/11/2019

Sprint Overview

The team should generate user stories to help guide the process over future sprints, and then determine a number of tests to check that the product meets the user stories.

The development team should research programming languages and environments for the production of the game, and decide on a language/environment by the end of the sprint. If time permits, they should begin looking at procedural level development using the chosen language. Whilst the artwork developers should present three storylines to the team in which the team can vote on, and towards the end of the sprint begin artwork development for the chosen storyline. The scrum leader will begin sound development and perform general admin work.

Sprint Review

The team generated a number of user stories at the end of the sprint, along with several tests to verify that these have been achieved by the developed system. They also decided upon a programming language (C#) and environment (Unity) to produce the game with. They also voted on the final storyline, choosing the prince idea but incorporating ideas from the mystery idea.

The artwork developers began development of artwork towards the end of the sprint.

The scrum leader has begun sound development.

Team Meetings

Meeting identifier: 01

Meeting date: 06/11/19

Meeting minutes: 180 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai

Meeting actions:

- Three main ideas for the story line were chosen and presented to the team: Prince idea, Mystery idea and the Bear idea.
- As a team created a mind map on A2 paper for each idea, trying to further develop each idea.
- As a team began to brainstorm on A2 paper the functionality of each idea.
- Presented the three ideas to the customer during the customer meeting.

Task assignments:

- Scrum leader: begin sound production, sprint reviews and other admin.
- Developers: research development languages/environments.
- Artwork developers: research methods of creating assets and artwork.

Meeting identifier: 02

Meeting date: 12/11/19

Meeting minutes: 10 minutes

Meeting attendees: Dominika, Yuru, Daniel

Meeting actions:

- Voted on a storyline (Prince idea chosen).
- Voted on the programming language to be used (C# in Unity).
- Discussed organization of sprints, deciding to work with weekly sprints, starting on the Wednesday of each week (due to correlation with the customer meetings), and ending on the following Tuesday.
- Discussed user stories and arranged for a member of the team to begin writing these up.

Task assignments:

- Scrum leader: continue sound production, sprint reviews and other admin.
- Developers: begin development in Unity of procedural level generation, and write user stories/tests for user stories.
- Artwork developers: begin artwork for player and enemies, produce final storyline draft.

Backlog

Backlog identifier: 01

Completed Backlog tasks: N/A

New Backlog tasks: N/A

Exception handling

One member of the team could not make it to meeting 01 due to car issues. To overcome this, photos of all mind maps were sent to the missing team member, asking for their input, along with a short summary to bring them up to speed with everything discussed, and their tasks for the sprint.

Three members of the team could not make it to meeting 02 due to illness, therefore a meeting was conducted using group text message to ensure the important points could be discussed and decisions could be made as a group. Although this method of communication is not as effective as face to face, it was the only method available to the whole team.

Product documents

Customer meeting and analysis:

The three storylines were presented to the customer; they seemed excited for all three ideas, but ultimately stated that the decision on which storyline to continue with was up to the team.

The team decided that due to the views of the customer they would put the choice to a vote.

User stories:

As a user I want to be able to...

1. Complete the game, so that I can win the game.
2. Fail the game, so that I can lose the game.
3. Understand the storyline of the game so that I can feel more invested into the game.
4. Easily understand the objectives of the game so that I can choose how to invest my time in the particular activities.
5. Restart the game after failing without restarting the application, so that I can save time when restarting the game.
6. Navigate the game using the arrow keys and the mouse only so that it is easy to play.
7. Play against a computer (bot) so that the game is more interactive.
8. Easily understand the combat system so that I can focus my attention lese where in the game.
9. Know how to exit the level to move to the next level so that I don't waste time stuck in one level.
10. Play the game offline so that I can play it anywhere.
11. Navigate a main menu screen before going into the game, so that the game feels more structured.
12. Hear sound effects when I perform actions in the game so that I feel more motivated when playing the game.
13. Hear background music within the game so that the game is more atmospheric.
14. Collect items which affect my player so that I can be highly motivated in the game.
15. Easily identify which body parts I've already collected, and those I am yet to collect, so that I can focus my attention on other parts of the game.
16. Quantify my performance during the game so that I can replay the game again and again aiming to beat my previous scores.

User story tests:

1. Check the user can complete the game.
2. Check the user can fail the game.
3. Check whether a user who is unfamiliar with the system can understand the storyline throughout the game.
4. Check whether a user who is unfamiliar with the system can easily understand the objectives of the game.
5. Check the user can restart the game after failing, without restarting the application.
6. Check the user can complete the game using the arrow keys and the mouse only.
7. Check that the user plays against a computer at some point during the game.
8. Check whether a user who is unfamiliar with the system can easily understand the combat system.
9. Check whether a user who is unfamiliar with the system can work out how to exit the level to progress to the next level.
10. Check that a user can play the game offline.
11. Check the user can navigate a main menu screen upon opening the application.
12. Check the user can hear sound effect when performing actions, assuming they have a working method for audio output.
13. Check the user can hear background music in the game, assuming they have a working method for audio output.

14. Check the user can collect items during the game which affect their player.
15. Check whether a user who is unfamiliar with the system can easily identify which body parts they have collected so far, and which ones still need to be collected.
16. Check that the game contains a method of quantifying the player's performance during the game, and that the player's performance is presented to them using some variable.

Requirements use cases:

No requirements use cases have been developed so far.

Use case tests:

No use case tests have been developed so far.

CRC cards:

No CRC cards have been developed so far.

Design use cases:

No design use cases have been developed so far.

User interface design:

No user interface design has occurred so far.

Sprint 2: 13/11/2019 – 19/11/2019

Sprint Overview:

The team should derive a number of use cases from the user stories that were developed at the end of sprint 1. These use cases should provide a more formal, extensive version of the user stories.

The development team should continue work on procedural level development, whilst the artwork developers continue with artwork development for the chosen storyline with the intent to finish the player and enemy's artwork by the end of the sprint. The scrum leader will continue with sound development and general admin work.

Sprint Review:

The team developed a number of use cases derived from the user stories developed in sprint 1.

The team also organized new meeting times for the future weeks, increasing the number of short 30 minute meetings from once per sprint to twice per sprint; in addition to the 3 hour lab session.

Developers had trouble sharing Unity projects through GitHub. Lead developer has made significant progress with procedural multi-level generation. Artwork development is coming along for the players and the enemies but is still incomplete.

The artwork for the player and enemies has been added to the project backlog 02. The team has decided to continue with this work in the next sprint.

Team Meetings:

Meeting identifier: 03

Meeting date: 13/11/19

Meeting minutes: 180 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Discussed changes of roles based on implementation of Unity and experience with this language.
- Presented revised final storyline to the customer.
- Further development of procedural multi-level generation.
- Discussed the current backlog and decided to continue with the artwork development in the next sprint.
- Discussed the development of user stories into use cases and assigned a member of the team to write these up.

Task assignments:

- Scrum leader: finalize sound production, set up GitHub, sprint reviews and other admin.
- Developers: continue development of multi-level generation, upload projects to GitHub, and develop user stories into use cases.

- Artwork developers: continue work on assets and artwork and design a card game for battle between bot and player.

Meeting identifier: 04

Meeting date: 19/11/19

Meeting minutes: 10 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Organized new short meeting times (Tuesday 11:15 1W 2.03 and Friday 10:15 library).
- Increased the number of meetings per sprint.
- Discussed issues with members of the team getting the Unity project from GitHub to run on their own computers.

Task assignments:

- Scrum leader: sound production, sprint reviews and other admin.
- Developers: fix issues with GitHub and sharing Unity projects.
- Artwork developers: finalize artwork for player and enemies.

Backlog:

Backlog identifier: 02

Completed Backlog tasks: N/A

New Backlog tasks: artwork for player and enemies

Exception handling:

During this sprint several team members could not make all the meetings due to sickness, and other unavoidable duties. So, meeting 04 was conducted via group text message, as no other alternative method was possible.

Due to this occurring during the previous sprint as well, new meeting times were arranged for future sprints, along with an additional meeting per sprint to ensure that the whole team is communicating more effectively.

Product documents:

Customer meeting and analysis:

The final storyline was presented to the customer along with a discussion on some of the functionality the customer can expect in future sprints. The customer was very happy with the storyline chosen and the proposed functionality.

User stories:

No new additions/alterations since sprint 1.

User story tests:

No new additions/alterations since sprint 1.

Requirements use cases:

Use case number relates directly back to the user story that it was derived from.

1 Use Case: Completing the game

1.1 Description:

The user wants to be able to complete the game.

1.2 Level: User goal.

1.3 Primary actors:

The user.

1.4 Additional/Supporting Actors:

Witch boss character.

1.5 Frequency of use:

Once per game.

1.6 Triggers:

The player defeats the end witch boss.

1.7 Stakeholders:

The customer.

1.8 Preconditions:

The user must have reached the final level of the game, collected all body parts, and defeated the final boss character (the witch).

1.9 Postconditions:

The end game scene will execute, and the user will be informed by text that they have completed the game. They will be given the option to either restart the game or exit the application.

1.10 Main success scenario:

1. User collects final body part from final level
2. User defeats the end game boss (the witch)
3. User is presented with the game complete scene

4. User either chooses to ‘restart’ the game from the beginning or ‘quit’ the application

1.11 Extensions:

1.11.1 Exceptions:

1. User ‘dies’ in the game through a battle with a monster or the witch, and never makes it to the end game screen.

2. User never collects all the body parts in the game.

1.11.2 Handling exceptions:

1. User is taken to the game over screen instead, and is given the same options, to quit the application or restart from the beginning.

2. User is prevented from completing a level until he has collected the body part present on that level.

1.12 Special requirements:

1.12.1 Usability/Accessibility:

1. User must be able to read English.

2. User must be able to operate a mouse and keyboard.

2 Use Case: Fail the game

2.1 Description:

The user wants to be able to fail the game.

2.2 Level: User goal.

2.3 Primary actors:

The user.

2.4 Additional/Supporting Actors:

Witch boss character or monsters.

2.5 Frequency of use:

Once per game.

2.6 Triggers:

The player loses a battle against a monster or the witch and has ran out of lives.

2.7 Stakeholders:

The customer.

2.8 Preconditions:

The player moves into the same space as a monster triggering a battle scene. The player must have only one life left before they enter battle with a monster, they must lose this battle, resulting in 0 lives left.

Or the player collects all of the body parts present in the game and initiates battle against the witch boss character. The player can enter battle with any number of lives against the witch and lose.

2.9 Postconditions:

The game over scene will execute, and the user will be informed by text that they have lost the game. They will be given the option to either restart the game or exit the application.

2.10 Main success scenario:

1. User collects final body part from final level
2. User loses battle against the end game boss (the witch)
3. User is presented with the game over scene
4. User either chooses to ‘restart’ the game from the beginning or ‘quit’ the application

Or

1. User enters battle against monster by sharing the same space as the monster
2. User loses battle against the monster
3. User is out of lives
4. User is presented with the game over scene
5. User either chooses to ‘restart’ the game from the beginning or ‘quit’ the application

2.11 Extensions:

2.11.1 Exceptions:

1. User never collects the final body part and initiates combat against the witch.

2. User completes the game.

3. User never shares the same space as monsters in the game.

2.11.2 Handling exceptions:

1. User will not be able to continue the level until they have collected the body part present on that level.

2. User greeted with game complete scene instead and is given the same options, to quit the application or restart from the beginning.

3. This exception should be allowed; the user may navigate through the game successfully evading monsters.

2.12 Special requirements:

2.12.1 Usability/Accessibility:

1. User must be able to read English.
2. User must be able to operate a mouse and keyboard.

3 Use Case: Understanding storyline

3.1 Description:

The user wants to be able to understand the storyline of the game.

3.2 Level: User goal.

3.3 Primary actors:

The user.

3.4 Additional/Supporting Actors:

N/A

3.5 Frequency of use:

Several times throughout the game.

3.6 Triggers:

The user begins the game, when the user collects body parts and also when the user defeats the witch.

3.7 Stakeholders:

The customer.

3.8 Preconditions:

The user begins the game by selecting start game on the menu screen. The user collects a body part in a level.

3.9 Postconditions:

The user will be returned to the main game scene to continue.

3.10 Main success scenario:

1. User enters the text-based cut scenes
2. User chooses to continue on each text scene
3. Once all cut scenes have been navigated, the user is returned to the main game

3.11 Extensions:

3.11.1 Exceptions:

1. N/A

3.11.2 Handling exceptions:

1. N/A

3.12 Special requirements:

3.12.1 Usability/Accessibility:

1. User must be able to read English.

2. User must be able to operate a mouse and keyboard.

4 Use Case: Understanding objectives

4.1 Description:

The user wants to be able to easily understand the objectives of the game.

4.2 Level: User goal.

4.3 Primary actors:

The user.

4.4 Additional/Supporting Actors:

Witch boss character.

4.5 Frequency of use:

Once per game.

4.6 Triggers:

The user begins the game.

4.7 Stakeholders:

The customer.

4.8 Preconditions:

The user chooses to start the game from the main menu scene.

4.9 Postconditions:

The user is taken to the main game scene.

4.10 Main success scenario:

1. User selects start game from the main menu scene.
2. User navigates through the cut scenes which explain the objectives of the game.
3. User enters main game.

4.11 Extensions:

4.11.1 Exceptions:

1. N/A

4.11.2 Handling exceptions:

1. N/A

4.12 Special requirements:

4.12.1 Usability/Accessibility:

1. User must be able to read English.

2. User must be able to operate a mouse and keyboard.

5 Use Case: Restarting the game

5.1 Description:

The user wants to be able to restart the game after failing without restarting the whole application.

5.2 Level: User goal.

5.3 Primary actors:

The user.

5.4 Additional/Supporting Actors:

N/A

5.5 Frequency of use:

Once per game.

5.6 Triggers:

The user selects restart game by clicking the restart game button.

5.7 Stakeholders:

The customer.

5.8 Preconditions:

The user must have failed the game by either running out of lives, or by losing battle against the witch.

5.9 Postconditions:

The user is taken back to the start of the game, everything is reset, and they start again.

5.10 Main success scenario:

1. User selects restart game by clicking the restart game button.
2. User is taken back to the beginning of the game where everything in the game is reset.
3. User replays game.

5.11 Extensions:

5.11.1 Exceptions:

1. N/A

5.11.2 Handling exceptions:

1. N/A

5.12 Special requirements:

5.12.1 Usability/Accessibility:

1. User must be able to read English.

2. User must be able to operate a mouse and keyboard.

8 Use Case: Understanding the combat system

8.1 Description:

The user wants to be able to easily understand the combat system.

8.2 Level: User goal.

8.3 Primary actors:

The user.

8.4 Additional/Supporting Actors:

Monsters or the witch.

8.5 Frequency of use:

Throughout the game.

8.6 Triggers:

The user selects the explain combat option in the corner of the combat screen.

8.7 Stakeholders:

The customer.

8.8 Preconditions:

The user enters combat against a monster or the witch.

8.9 Postconditions:

The user returns to the combat scene.

8.10 Main success scenario:

1. User enters combat with either the witch or a monster.
2. User chooses to view the combat guide using the button in the top right of the scene.
3. User is taken to the combat guide scene.
4. User chooses to return to the combat.
5. User continues combat.

8.11 Extensions:

8.11.1 Exceptions:

1. The player forgets how to use the combat system.

8.11.2 Handling exceptions:

1. The player can return to the combat explained scene as many times as they want.

8.12 Special requirements:

8.12.1 Usability/Accessibility:

1. User must be able to read English.

2. User must be able to operate a mouse and keyboard.

12 Use Case: Actions with sound effects

12.1 Description:

The user wants to be able to hear sound effects when they perform actions in the game.

12.2 Level: User goal.

12.3 Primary actors:

The user.

12.4 Additional/Supporting Actors:

Monsters or the witch.

12.5 Frequency of use:

Throughout the game.

12.6 Triggers:

The user performs an action in the game such as attacking a monster or collecting a body part/other item in the game.

12.7 Stakeholders:

The customer.

12.8 Preconditions:

The user is in the main game scene.

12.9 Postconditions:

The sound effect plays and the user can hear if they have sound enabled on their computer, along with a device to output the sound.

12.10 Main success scenario:

1. User performs an action in the main game scene.
2. The sound effect for the action plays.
3. The user continues game play.

12.11 Extensions:

12.11.1 Exceptions:

1. N/A

12.11.2 Handling exceptions:

1. N/A

12.12 Special requirements:

12.12.1 Usability/Accessibility:

1. User must be able to operate a mouse and keyboard.

2. User must have sound enabled on their computer along with a device to output sound.

14 Use Case: Collecting items

14.1 Description:

The user wants to be able to collect items which affect their player.

14.2 Level: User goal.

14.3 Primary actors:

The user.

14.4 Additional/Supporting Actors:

N/A

14.5 Frequency of use:

Throughout the game.

14.6 Triggers:

The user moves into the same space as an item in the game.

14.7 Stakeholders:

The customer.

14.8 Preconditions:

The user is in the main game scene.

14.9 Postconditions:

The effect of the item is applied to the user's player.

14.10 Main success scenario:

1. User moves their player onto the same space as an item in the game.
2. The effect of the item is applied to the user's player.
3. The user continues to play the game.

14.11 Extensions:

14.11.1 Exceptions:

1. N/A

14.11.2 Handling exceptions:

1. N/A

14.12 Special requirements:

14.12.1 Usability/Accessibility:

1. User must be able to operate keyboard.

16 Use Case: Quantifying performance

16.1 Description:

The user wants to be able to quantify their performance in the game.

16.2 Primary actors:

The user.

16.3 Additional/Supporting Actors:

Monsters.

16.4 Frequency of use:

Throughout the game.

16.5 Triggers:

The user completes the game by defeating the witch and is taken to the game complete scene.

16.6 Stakeholders:

The customer.

16.7 Preconditions:

The user is in the final level of the main game.

16.8 Postconditions:

The user is in the game complete scene, where the number of monsters killed is listed and acts as a score keeper for the user.

16.9 Main success scenario:

1. User defeats the witch.
2. User is taken to game complete scene.
3. User can identify how many monsters they've killed which acts as a way of quantifying their performance in the game.

16.10 Extensions:

16.11.1 Exceptions:

1. User fails the game and never reaches the game complete scene.

2. User never collects all of the body parts.

16.11.2 Handling exceptions:

1. User should be allowed to fail.

2. User will be forced to collect all body parts before moving to the next level.

16.11 Special requirements:

16.12.1 Usability/Accessibility:

1. User must be able to read English.

2. User must be able to operate a mouse and keyboard.

Use case tests:

No use case tests have been developed so far.

CRC cards:

No CRC cards have been developed so far.

Design use cases:

No design use cases have been developed so far.

User interface design:

N/A

Sprint 3: 20/11/2019 – 26/11/2019

Sprint Overview:

The team should produce a number of tests to apply to the use cases that were developed in sprint 2; these will be used in later sprints to ensure that the developed system meets the user's needs.

The development team should complete work on procedural level development, player and bot movement, basic user interface design such as the main menu, and solve the problems occurring with sharing Unity projects through GitHub. The artwork developers should finalize artwork for the player and enemies, present a card game and begin designing artwork for the card game. The scrum leader will continue with sound development and general admin work.

The team should aim to clear the project backlog (02) by the end of this sprint.

Sprint Review:

The team developed a number of use case tests; these were derived from the use cases developed in sprint 2.

The development team faced difficulty with sharing Unity files through GitHub, but devised a new way to collaborate with each other using Unity's inbuilt collaborator tool. They have produced a method of procedural level generation with simple player movement/bot movement mechanics, along with a main menu scene. The artwork team has completed the artwork for the player and the enemies, and has begun designing the artwork for the card game, along with devising the rules for the game. The team is falling behind with the sound design for the game; the scrum leader has directed their focus solely towards the sound design, whilst a member of the development team has shifted their focus to get all of the sprint reviews up to date.

The project backlog 02 has been cleared, but a new item has been added to project backlog 03; the sound development for the game. The team has decided to continue with the work on the sound development into the next sprint.

Team Meetings:

Meeting identifier: 05

Meeting date: 20/11/19

Meeting minutes: 180 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Solved the sharing of Unity projects through GitHub (a new method not using GitHub was chosen).
- Unity's in-built project collaborator was set up between developers to allow developers to work on the same project with ease.
- A simple method of sharing Unity projects via zip files on Google Drive was also devised as a backup if team members encounter future GitHub issues.

- Presented procedural level generation with simple player movement/bot movement to customer, all sprites used were copyright free use graphics as placeholders for our own produced graphics.
- Graphics shared on Google Drive.
- Card game rules devised and presented to team.

Task assignments:

- Scrum leader: finalize sound production.
- Developers: continue development of multi-level generation, assist scrum master with catching up on sprint reviews, and develop a series of tests for the use cases developed last sprint.
- Artwork developers: Begin work on assets and artwork for card game.

Meeting identifier: 06

Meeting date: 22/11/19

Meeting minutes: 30 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Lead developer explained how the current code/Unity project worked.
- Discussed current game functionality.
- Discussed future game functionality.
- Discussed dividing development jobs up.

Task assignments:

- Scrum leader: finalize sound production.
- Developers: continue development of multi-level generation, and assist scrum master with catching up on sprint reviews.
- Artwork developers: Continue work on assets and artwork for card game.

Meeting identifier: 07

Meeting date: 26/11/19

Meeting minutes: 30 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Discussed where everyone is at with their work.
- Discussed and organized each member of the team's assignment for the next sprint.

Task assignments:

- Scrum leader: finalize sound production.
- Developers: finalize development of multi-level generation, and assist scrum master with catching up on sprint reviews.
- Artwork developers: Continue work on assets and artwork for card game.

Backlog:

Backlog identifier: 03

Completed Backlog tasks: player and enemy artwork

New Backlog tasks: sound development

Exception handling:

Scrum leader falling behind on the sprint reviews due to high time demand that sound production is requiring. Therefore, a member of the development team assisted with the sprint reviews to allow the scrum leader to focus on sound development.

Product documents:

Customer meeting and analysis:

The team presented a demo of the game so far, showing the procedural level generation and the basic movement of the player and bots. The customer was happy with the progress being made with the game and the direction the team is heading.

User stories:

No new additions/alterations since sprint 1.

User story tests:

No new additions/alterations since sprint 1.

Requirements use cases:

No new additions/alterations since sprint 2.

Use case tests:

The following table contains the test steps and the expected results of each step for the use cases listed in sprint 2.

Use Case Number	Main Success Scenario Step	Description Of Step	Expected Result
1	1	User collects final body part from final level.	The witch boss appears and takes the user into combat.
	2	User defeats the end game boss (the witch).	User is presented with the game complete scene.
	3	User returns to menu, and can either ‘restart’ the game from the beginning or ‘quit’ the application.	User presented with menu, from there user restarts, the user is taken back into the game, if the user chooses to quit, the application is closed.
	Exception 1	User ‘dies’ in the game through a battle with a monster.	User is taken to the game over screen, and is given the option to return to the menu.
	Exception 2	User ‘dies’ in the game through a battle with the witch.	User is taken to the game defeat screen, and is given the option to return to the menu..

	Exception 3	User does not collect the final body part.	User is prevented from completing a level until he has collected the body part present on that level.
2	1	User collects final body part from final level.	The witch boss appears and takes the user into combat.
	2	User loses battle against the end game boss (the witch).	User is taken to the game defeat screen, and is given the option to return to the menu..
	3	User either chooses to restart the game from the beginning or quit the application.	If user restarts, the user is taken back into the game, if the user chooses to quit, the application is closed.
	1	User enters battle against monster by sharing the same space as the monster; the user has 1 life left before battle.	The user enters battle scene against the monster.
	2	User loses battle against the monster.	User is out of lives. User is presented with the game over scene.
	3	User either chooses to restart the game from the beginning or quit the application.	If user restarts, the user is taken back into the game, if the user chooses to quit, the application is closed.

	Exception 1	User never collects the final body part and attempts to initiate combat against the witch.	User cannot find witch until final body part has been collected.
	Exception 2	User completes the game.	User greeted with game complete scene instead and is given the same option, to return to the menu after victory.
	Exception 3	User never shares the same space as monsters in the game.	User never enters combat until the user fights the witch.
3	1	User selects start game on the main menu scene.	The game scene changes to the first cut scene, with a button which gives the user the option to continue.
	2	User chooses to continue on each text scene excluding the final text scene.	User should be informed of the relevant storyline.
	3	User chooses to continue on final text scene.	User is taken to the first level of the game.

4	1	User selects start game on the main menu scene.	The game scene changes to the first cut scene, with a button which gives the user the option to continue.
	2	User chooses to continue on each text scene excluding the final text scene.	User should be informed of the objectives during these cut scenes.
	3	User chooses to continue on final text scene.	User is taken to the first level of the game.
5	1	User completes the game.	User is taken to the game complete scene.
	2	User chooses to restart the game using the restart game button.	The user is taken to the main menu scene.
	3	The user chooses to start the game using the start game button in the main menu scene.	The user is taken to the first cut scene, and the whole game is reset.
8	1	User views the combat instructions from the main menu.	User is taken to instruction screen.

	2	User clicks the button to return to menu.	User is taken back to menu.
	3	User clicks the button to start game.	Game begins as normal.
	4	User enters combat.	Combat begins like normal.
	1	User views the combat instructions from the main menu.	User is taken to instruction screen.
	2	User clicks the button to return to menu.	User is taken back to menu.
	3	User starts the game.	Game begins as normal.
	4	User enters combat with the witch.	Witch battle begins continues as normal.
12	1	User passes through exit of current level.	User is taken to the next level, and a sound effect plays.

14	1	The user moves the player over a health potion.	The user's number of lives increases.
	Exception 1	The Users life is currently max.	Users life doesn't increase above max.
16	1	User defeats the witch after defeating one monster throughout the game.	The user is taken to the game complete scene where it states that the player has defeated two monsters. (The witch is a monster)
	1	User defeats the witch after defeating no monster throughout the game.	The user is taken to the game complete scene where it states that the player has defeated 1 monster.

CRC cards:

No CRC cards have been developed so far.

Design use cases:

No design use cases have been developed so far.

User interface design:

User story number 11 which can be found in sprint document 1, states that the user wants to be able to navigate a main menu screen before entering the game. A simple main menu scene has been created to answer these needs of the user. It gives the user the option to either start the game or quit the game. These options are in the form of buttons, which the user can select using the mouse cursor.

Sprint 4: 27/11/2019 – 03/12/2019

Sprint Overview:

The team should aim to produce a prototype of the game by the end of this sprint, as this is the penultimate sprint, and the final sprint should be used to implement any changes required from testing.

The development team should aim to complete the mechanics for the battles between enemies and the player using a card game mechanic, complete the mechanics for items in the game, complete acquiring body parts mechanics, implement the storyline and produce a prototype.

The artwork developers should complete the graphics for the card game and design the body parts. The scrum leader will begin implementing sound into the game, and begin work on the construction of the: user manual, installation guide and the maintenance guide.

At the end of the sprint the team should get participants to test the game and give feedback that will allow the team to implement any required changes in the following sprint. The team should also perform all the tests listed in the user story tests section of sprint 1, along with all of the tests listed in the use case tests section of sprint 3.

The team should aim to clear project backlog 03 by the end of the sprint.

Sprint Review:

The team produced a prototype of the complete game minus a few elements. The prototype includes the main menu for the game, cut scenes which convey the storyline, along with procedural room generation; it allows the user to navigate the dungeon either interacting with monsters or objects randomly generated within the rooms. The card game for battles between the user and monsters has been fully implemented; including all the graphics. Some sound has begun to be integrated into the game. The missing features include the final boss battle with the witch, the random generation of the body parts throughout the dungeon, the game complete scene, and the ability to die in the game (lose). The artwork team has completed all of the graphics for the game.

Work has begun on the construction of the user manual, installation guide and the maintenance guide as intended.

Testing of the prototype has been performed, and the results documented in the ‘user story tests’ section and the ‘use case tests’ section of this sprint document. Not all tests were passed due to the missing features in the prototype (discussed above).

The team has completed all tasks from project backlog 03. New tasks have been added to project backlog 04. The team has decided to continue with all of these tasks into sprint 5.

The development team also experimented with the use of CRC cards to help with the development of the game.

Team Meetings:

Meeting identifier: 08

Meeting date: 27/11/19

Meeting minutes: 180 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Discussed goals for this sprint.
- Had two different methods of level generation, one where each room is randomly generated and not connected to the previous, and another where you can walk from room to room and return to previous rooms. Due to bugs in the latter method, and the simplistic nature of the first method, the first method was chosen.
- The artwork developers acquired input from the whole team on the ideas they had so far for the card game graphics and body parts.
- No customer meeting.
- Discussed methods of implementing the storyline to the game, text prompts on screen vs finding clues in the game (text prompts chosen).
- Scrum leader gave a demo to the team of the different sorts of sounds he had created and acquired input on which to use.

Task assignments:

- Development team: complete the mechanics for the battles between enemies and the player using a card game mechanic, complete the mechanics for items in the game, complete acquiring body parts mechanics, implement the storyline and produce a prototype.
- Artwork developers: complete the graphics for the card game and design the body parts.
- Scrum leader: begin implementing sound into the game, and begin work on the construction of the: user manual, installation guide and the maintenance guide.

Meeting identifier: 09

Meeting date: 29/11/19

Meeting minutes: 30 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Lead developer shared their recent progression with the project which included a basic scene which allows the user to choose cards and place them on a map for the player versus monster battles.
- Discussed current game functionality, focusing on what is left to do this sprint, the main discussion based around the mechanics of the card game.
- Reorganized the development work for the rest of the sprint.
- The artwork developers shared the artwork they had produced so far and asked for feedback.

Task assignments:

- Development team: complete the mechanics for the battles between enemies and the player using a card game mechanic, complete the mechanics for items in the game, complete acquiring body parts mechanics, implement the storyline and produce a prototype.
- Artwork developers: using feedback from team finalize the graphics.
- Scrum leader: continue implementing sound into the game, and continue to work on the construction of the: user manual, installation guide and the maintenance guide.

Meeting identifier: 10

Meeting date: 03/12/19

Meeting minutes: 30 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Development team gave a demo of the current prototype to the rest of the team, followed by a group discussion on everyone's opinions on the prototype.
- The missing functionality of the game was discussed, and the remaining work redistributed for the final sprint.
- The artwork developers presented their final graphics for the game.
- Testing of the prototype was organized.

Task assignments:

- Development team: complete mechanics for acquiring body parts in the game, and implement the graphics for this.
- Development team: complete mechanics for the witch boss battle and implement graphics for this.
- Development team: Implement game complete end screen.
- Artwork developers: reformat a couple of the graphics for the development team to implement.
- Scrum leader: complete the implementation of sound into the game, and finish work on the construction of the: user manual, installation guide and the maintenance guide.

Backlog:

Backlog identifier: 04

Completed Backlog tasks: sound development

New Backlog tasks: development of mechanics for: acquiring body parts in the game, witch boss battle, end game scene, and ability to lose.

Exception handling:

N/A

Product documents:

Customer meeting and analysis:

Customer meetings did not occur over this sprint due to the UCU strikes.

User stories:

No new additions/alterations since sprint 1.

User story tests:

User story tests performed on the prototype produced in this sprint. As discussed above, the prototype is missing a few elements of functionality; this resulted in the prototype failing many of the tests. These tests allowed the team to see clearly the elements which need to be prioritized in the final sprint, such elements include: the collection of body parts, the witch boss fight, the game complete scene, and the ability for the character to run out of lives and lose.

User story being tested	Test	Result (Pass/Fail)	Changes to implement
1	Check the user can complete the game.	FAIL	Add witch battle, collection of body parts and game complete scene.
2	Check the user can fail the game.	FAIL	Add mechanic for player running out of lives and game over scene.
3	Check whether a user who is unfamiliar with the system can understand the storyline throughout the game.	PASS	

4	Check whether a user who is unfamiliar with the system can easily understand the objectives of the game.	PASS	
5	Check the user can restart the game after failing, without restarting the application.	FAIL	Add mechanic for player running out of lives and game over scene. Include restart game button.
6	Check the user can complete the game using WASD and the mouse only.	FAIL	Add witch battle, collection of body parts and game complete scene.
7	Check that the user plays against a computer at some point during the game.	PASS	
8	Check whether a user who is unfamiliar with the system can easily understand the combat system.	FAIL	Add combat guide to battle scene.
9	Check whether a user who is unfamiliar with the system can work out how to exit the level to progress to the next level.	PASS	
10	Check that a user can play the game offline.	PASS	
11	Check the user can navigate a main menu screen upon opening the application.	PASS	
12	Check the user can hear sound effect when performing actions, assuming they have a working method for audio output.	FAIL	Complete integration of sound.

13	Check the user can hear background music in the game, assuming they have a working method for audio output.	FAIL	Complete integration of sound.
14	Check the user can collect items during the game which affect their player.	PASS	
15	Check whether a user who is unfamiliar with the system can easily identify which body parts they have collected so far, and which ones still need to be collected.	FAIL	Add graphic displaying body parts to the top corner of main game scenes. Add mechanics for collection of body parts.
16	Check that the game contains a method of quantifying the player's performance during the game, and that the player's performance is presented to them using some variable.	FAIL	Add counter to system which keeps track of how many monsters have been defeated.

Requirements use cases:

No new additions/alterations since sprint 2.

Use case tests:

Use case tests performed on the prototype produced in this sprint. As discussed above, the prototype is missing a few elements of functionality; this resulted in the prototype failing many of the tests. These tests allowed the team to see clearly the elements which need to be prioritized in the final sprint, such elements include: the collection of body parts, the witch boss fight, the game complete scene, and the ability for the character to run out of lives and lose.

Use Case Number	Main Success Scenario Step	Description Of Step	Expected Result	Result (Pass or Fail)
1	1	User collects final body part from final level.	The witch boss appears and takes the user into combat.	FAIL
	2	User defeats the end game boss (the witch).	User is presented with the game complete scene.	FAIL
	3	User returns to menu, and can either ‘restart’ the game from the beginning or ‘quit’ the application.	User presented with menu, from there user restarts, the user is taken back into the game, if the user chooses to quit, the application is closed.	FAIL
	Exception 1	User ‘dies’ in the game through a battle with a monster.	User is taken to the game over screen, and is given the option to return to the menu.	FAIL
	Exception 2	User ‘dies’ in the game through a battle with the witch.	User is taken to the game defeat screen, and is given the option to return to the menu.	FAIL
	Exception 3	User does not collect the final body part.	User is prevented from completing a level until he has collected the body part present on that level.	FAIL

2	1	User collects final body part from final level.	The witch boss appears and takes the user into combat.	FAIL
	2	User loses battle against the end game boss (the witch).	User is taken to the game defeat screen, and is given the option to return to the menu.	FAIL
	3	User either chooses to restart the game from the beginning or quit the application.	If user restarts, the user is taken back into the game, if the user chooses to quit, the application is closed.	FAIL
	1	User enters battle against monster by sharing the same space as the monster; the user has 1 life left before battle.	The user enters battle scene against the monster.	FAIL
	2	User loses battle against the monster.	User is out of lives. User is presented with the game over scene.	FAIL
	3	User either chooses to restart the game from the beginning or quit the application.	If user restarts, the user is taken back into the game, if the user chooses to quit, the application is closed.	FAIL

	Exception 1	User never collects the final body part and attempts to initiate combat against the witch.	User cannot find witch until final body part has been collected.	FAIL
	Exception 2	User completes the game.	User greeted with game complete scene instead and is given the same option, to return to the menu after victory.	FAIL
	Exception 3	User never shares the same space as monsters in the game.	User never enters combat until the user fights the witch.	FAIL
3	1	User selects start game on the main menu scene.	The game scene changes to the first cut scene, with a button which gives the user the option to continue.	PASS
	2	User chooses to continue on each text scene excluding the final text scene.	User should be informed of the relevant storyline.	PASS
	3	User chooses to continue on final text scene.	User is taken to the first level of the game.	PASS

4	1	User selects start game on the main menu scene.	The game scene changes to the first cut scene, with a button which gives the user the option to continue.	PASS
	2	User chooses to continue on each text scene excluding the final text scene.	User should be informed of the objectives during these cut scenes.	PASS
	3	User chooses to continue on final text scene.	User is taken to the first level of the game.	PASS
5	1	User completes the game.	User is taken to the game complete scene.	FAIL
	2	User chooses to restart the game using the restart game button.	The user is taken to the main menu scene.	FAIL
	3	The user chooses to start the game using the start game button in the main menu scene.	The user is taken to the first cut scene, and the whole game is reset.	FAIL
8	1	User views the combat instructions from the main menu.	User is taken to instruction screen.	PASS

	2	User clicks the button to return to menu.	User is taken back to menu.	FAIL
	3	User clicks the button to start game.	Game begins as normal.	FAIL
	4	User enters combat.	Combat begins like normal.	FAIL
	1	User views the combat instructions from the main menu.	User is taken to instruction screen.	FAIL
	2	User clicks the button to return to menu.	User is taken back to menu.	FAIL
	3	User starts the game.	Game begins as normal.	FAIL
	4	User enters combat with the witch.	Witch battle begins continues as normal.	FAIL
12	1	User passes through exit of current level.	User is taken to the next level, and a sound effect plays.	FAIL

14	1	The user moves the player over a health potion.	The user's number of lives increases.	PASS
	Exception 1	The Users life is currently max.	Users life doesn't increase above max.	PASS
16	1	User defeats the witch after defeating one monster throughout the game.	The user is taken to the game complete scene where it states that the player has defeated two monsters. (The witch is a monster)	FAIL
	1	User defeats the witch after defeating no monster throughout the game.	The user is taken to the game complete scene where it states that the player has defeated 1 monster.	FAIL

CRC cards:

A number of CRC cards were generated by the development team to help understand how each class interacts with each other.

The CRC cards below summarize the main classes used in the game.

Game Manager	
Handles overall game function and flow Persistent object - created at beginning of game and exists throughout Holds all game data needed by other classes Data includes - Player health, Player upgrades, Current enemies, and functional data such as turn delays, and whose turn it is. Handles Scene (level) management Calls enemy movement	Board Manager Battle Manager Player Enemy Loader Upgrade Chest

Board Manager	
Handles to creation and maintenance of the game board the player plays on in the overworld. Procedurally generates the world. Gets level information from Game Manager. Spawns exit and upgrade options for the player. Spawns the player.	Game Manager Upgrade Chest Player Enemy

Player	
Inherits from Moving Object class Handles player movement and interaction with the world. Interacts with Game Manager when the player collects an item or attacks/is attacked by an enemy.	Moving Object (inherits) Game Manager Enemy Upgrade Chest

Enemy	
Inherits from Moving Object Spawned by Board Manager Moves when called by Game Manager Interacts with Game Manager when attacking the player.	MovingObject (inherits) Game Manager Player

The CRC cards below were used to understand how the classes used for the combat scene interact.

Battle Enemy	
Handles enemy action during the battle. Gets information and hands information to the battle manager.	Battle Manager

Draggable	
Class allows drag functionality for the cards in the card game. Handles card score and card effects.	Battle Manager

Drop Zone	
Handles the drop zones for the card game. Changes colour based on the battle manager.	Battle Manager

Design use cases:

No design use cases have been developed so far.

User interface design:

The cutscenes added during this sprint allow the user to easily understand the storyline and the objectives of the game. Thus, the cutscenes added satisfy user story number 3 and 4 which are outlined in sprint 1 documentation.

Sprint 5: 04/12/2019 – 12/12/2019

Sprint Overview:

As this is the final sprint the team should aim to complete all tasks by the end of this sprint. If the team does not manage to complete all tasks by the end of this sprint then due to the continuous integration approach that has been followed throughout the sprints, the team should still have a working system with reduced functionality to deliver to the customer.

The development team should aim to complete the tasks from backlog 4, including the mechanics for the collection of body parts within the game, the witch boss battle scene, the game complete end scene, and the ability to lose. Once these are complete, they should then implement any changes that resulted from the tests performed in sprint 4. They should then redo the testing performed in sprint 4 and check that the system passes all the tests. Finally, they should implement any graphics that have been produced but not implemented yet. If time allows, they should perform a final refactoring of the code before finalizing the product.

The artwork team should assist the scrum leader with the completion of any outstanding product material such as sprint documentation, user manual, installation guide and maintenance guide.

Sprint Review:

The team managed to complete all outstanding tasks. The development team had great success with implementing the final features along with any missing graphics. Although the collection of body parts in the game was not implemented as originally intended, a new solution was devised and implemented which involved an upgrade chest (see the new CRC card in this sprint document), this approach was decided upon due to feedback from testing which indicated that the original idea was a more confusing than this alternative. They then performed a final refactor on the system.

The team then performed the user story tests (outlined in sprint 1 documentation) along with the use case tests (outlined in sprint 3 documentation). The system passed almost all tests successfully, test 15 for the user story test failed, but as stated in the comments an alternative approach was taken which users found less confusing. The artwork team along with the scrum leader, completed the user manual, installation guide and maintenance guide, with assistance from the development team.

The project backlog 04 has been cleared, and there are no outstanding project backlog tasks.

Team Meetings:

Meeting identifier: 11

Meeting date: 05/12/19

Meeting minutes: 90 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Discussed the aims for the sprint.
- Reassessed task assignments for the sprint.

Task assignments:

- Development team: complete mechanics for acquiring body parts in the game, and implement the graphics for this.
- Development team: complete mechanics for the witch boss battle and implement graphics for this.
- Development team: Implement game complete end screen and ability to lose.
- Development team: implement changes that are derived from testing in sprint 4.
- Artwork developers: assist scrum leader.
- Scrum leader: Finish work on the construction of the: user manual, installation guide and the maintenance guide.

Meeting identifier: 12

Meeting date: 12/12/19

Meeting minutes: 180 minutes

Meeting attendees: Jake, Dominika, Yuru, Daniel, Shuai, Luke

Meeting actions:

- Reviewed each sprint document.
- Reviewed user manual, installation guide and the maintenance guide.
- Reviewed the final product (the game).
- Filled in the group questionnaire.

Task assignments:

No task assignments due to the completion of the project.

Backlog:

Backlog identifier: 05

Completed Backlog tasks: development of mechanics for: acquiring body parts in the game, witch boss battle and end game scene.

New Backlog tasks: N/A

Exception handling:

N/A

Product documents:

Customer meeting and analysis:

No customer meeting was arranged this sprint due to the interference of the UCU strikes and the re-scheduling issues that these caused.

User stories:

No new additions/alterations since sprint 1.

User story tests:

User story being tested	Test	Result (Pass/Fail)	Changes to implement	Comments
1	Check the user can complete the game.	PASS		
2	Check the user can fail the game.	PASS		
3	Check whether a user who is unfamiliar with the system can understand the storyline throughout the game.	PASS		Added Instruction screen for mechanics, and popups for story progression.
4	Check whether a user who is unfamiliar with the system can easily understand the objectives of the game.	PASS		Objectives stand out, and instructions for those confused.
5	Check the user can restart the game after failing, without restarting the application.	PASS		Game Fail now goes to the main menu rather than exiting application.

6	Check the user can complete the game using Arrow Keys and the mouse only.	PASS		
7	Check that the user plays against a computer at some point during the game.	PASS		
8	Check whether a user who is unfamiliar with the system can easily understand the combat system.	PASS		Added colours to show effects of card drop, and instruction screen for in detail explanations.
9	Check whether a user who is unfamiliar with the system can work out how to exit the level to progress to the next level.	PASS		
10	Check that a user can play the game offline.	PASS		
11	Check the user can navigate a main menu screen upon opening the application.	PASS		
12	Check the user can hear sound effect when performing actions, assuming they have a working method for audio output.	PASS		

13	Check the user can hear background music in the game, assuming they have a working method for audio output.	PASS		
14	Check the user can collect items during the game which affect their player.	PASS		
15	Check whether a user who is unfamiliar with the system can easily identify which body parts they have collected so far, and which ones still need to be collected.	FAIL		Moved from body part system to upgrades spaced throughout the game which represent the player gaining body parts. This was done based on player feedback, as a less confusing method.
16	Check that the game contains a method of quantifying the player's performance during the game, and that the player's performance is presented to them using some variable.	PASS		

Requirements use cases:

No new additions/alterations since sprint 2.

Use case tests:

Use Case Number	Main Success Scenario Step	Description Of Step	Expected Result	Result (Pass or Fail)
1	1	User collects final body part from final level.	The witch boss appears and takes the user into combat.	PASS
	2	User defeats the end game boss (the witch).	User is presented with the game complete scene.	PASS
	3	User returns to menu, and can either ‘restart’ the game from the beginning or ‘quit’ the application.	User presented with menu, from there user restarts, the user is taken back into the game, if the user chooses to quit, the application is closed.	PASS
	Exception 1	User ‘dies’ in the game through a battle with a monster.	User is taken to the game over screen, and is given the option to return to the menu.	PASS
	Exception 2	User ‘dies’ in the game through a battle with the witch.	User is taken to the game defeat screen, and is given the option to return to the menu..	PASS

	Exception 3	User does not collect the final body part.	User is prevented from completing a level until he has collected the body part present on that level.	PASS
2	1	User collects final body part from final level.	The witch boss appears and takes the user into combat.	PASS
	2	User loses battle against the end game boss (the witch).	User is taken to the game defeat screen, and is given the option to return to the menu.	PASS
	3	User either chooses to restart the game from the beginning or quit the application.	If user restarts, the user is taken back into the game, if the user chooses to quit, the application is closed.	PASS
	1	User enters battle against monster by sharing the same space as the monster; the user has 1 life left before battle.	The user enters battle scene against the monster.	PASS
	2	User loses battle against the monster.	User is out of lives. User is presented with the game over scene.	PASS
	3	User either chooses to restart the game from the beginning or quit the application.	If user restarts, the user is taken back into the game, if the user chooses to quit, the application is closed.	PASS

	Exception 1	User never collects the final body part and attempts to initiate combat against the witch.	User cannot find witch until final body part has been collected.	PASS
	Exception 2	User completes the game.	User greeted with game complete scene instead and is given the same option, to return to the menu after victory.	PASS
	Exception 3	User never shares the same space as monsters in the game.	User never enters combat until the user fights the witch.	PASS
3	1	User selects start game on the main menu scene.	The game scene changes to the first cut scene, with a button which gives the user the option to continue.	PASS
	2	User chooses to continue on each text scene excluding the final text scene.	User should be informed of the relevant storyline.	PASS
	3	User chooses to continue on final text scene.	User is taken to the first level of the game.	PASS

4	1	User selects start game on the main menu scene.	The game scene changes to the first cut scene, with a button which gives the user the option to continue.	PASS
	2	User chooses to continue on each text scene excluding the final text scene.	User should be informed of the objectives during these cut scenes.	PASS
	3	User chooses to continue on final text scene.	User is taken to the first level of the game.	PASS
5	1	User completes the game.	User is taken to the game complete scene.	PASS
	2	User chooses to restart the game using the restart game button.	The user is taken to the main menu scene.	PASS
	3	The user chooses to start the game using the start game button in the main menu scene.	The user is taken to the first cut scene, and the whole game is reset.	PASS
8	1	User views the combat instructions from the main menu.	User is taken to instruction screen.	PASS

	2	User clicks the button to return to menu.	User is taken back to menu.	PASS
	3	User clicks the button to start game.	Game begins as normal.	PASS
	4	User enters combat.	Combat begins like normal.	PASS
	1	User views the combat instructions from the main menu.	User is taken to instruction screen.	PASS
	2	User clicks the button to return to menu.	User is taken back to menu.	PASS
	3	User starts the game.	Game begins as normal.	PASS
	4	User enters combat with the witch.	Witch battle begins continues as normal.	PASS
14	1	The user moves the player over a health potion.	The user’s number of lives increases.	PASS

	Exception 1	The Users life is currently max.	Users life doesn't increase above max.	PASS
16	1	User defeats the witch after defeating one monster throughout the game.	The user is taken to the game complete scene where it states that the player has defeated two monsters. (The witch is a monster)	PASS
	1	User defeats the witch after defeating no monster throughout the game.	The user is taken to the game complete scene where it states that the player has defeated 1 monster.	PASS

CRC cards:

A new CRC card was created to help understand how to introduce an upgrade chest into the system to replace the collection of body parts in the game. This was included to ensure user story number 15 was satisfied without the original collection of body parts idea.

Upgrade Chest	
Handles the player upgrades and ending the level. Spawned by Board Manager. Interacts with Game Manager when triggered by player.	Game Manager

Design use cases:

No design use cases have been developed so far.

User interface design:

User story number 5, which can be found in sprint document 1, states that the user wants to be able to restart the game once failed without restarting the application. The development team has implemented a game over scene along with a game complete scene to answer these needs of the user. This allows the user to either replay the game or quit the application. These options are in the form of buttons, which the user can select using the mouse cursor.

The user story test number 5 was passed successfully; this indicates that the user can restart the game without closing the application.

A number of other UI additions have been made during this final sprint. The introduction of the number of lives illustrated by hearts at the top of the screen allows the user to clearly see how many lives they have left.

The score counter in the game allows the user to easily quantify their performance, which satisfies user story number 16 as stated in sprint 1 documentation.