# Sprint 6 | 1st December – 7th December

## **Sprint Overview**

### Preview

Despite not having a customer meeting at the end of the last sprint, we feel that we remain in a good position to continue to improve the functionality of our game and create tasks accordingly. We feel that we have made excellent progress as a group so far. We plan to continue to improve the aesthetic of the game such as adding music and improving the rolling background so there are enough planes for it to go on forever. When the rolling background was implemented in the previous sprint, we found that it would eventually end, so we wanted to fix this. We also wanted to change how the difficulty of the game evolves over time by adding a beat bonus which would be implemented by Matthew.

### Review

Since we didn't have any customer meetings last sprint, we assigned ourselves the task of making the game more challenging and entertaining for the user. As a result, we released our first downloadable version of the game for non-development members to test. We also added two pieces of background music, one to the menu and one to the game. Furthermore, we fixed some minor bugs and optimized the controls. Additionally, we constructed the backstory for how the game starts, why the boy character is in the dungeon, and why he is running through the pipes, etc.

In terms of documentation, we checked the content and structure of our sprint documents and made them more logical. By the end of this sprint, we had thoroughly checked all our sprint documents and rearranged them such that they all followed an identical format and structure. However, there were still several submission requirements that we did not fully understand. We made a list of all of these and asked the TAs and the client about them. In the client meeting, we received more concrete feedback regarding our documentation and development optimization.

## **Tasks**

|  |  |  |
| --- | --- | --- |
| Code | Team Members | Tasks |
| **S6-T1** | Marcus | (Previously S4.2-T11): Add obstacle generation. |
| **S6-T2** | Marcus | (Previously S4.2-T13):  Implement death on obstacle collision. |
| **S6-T3** | Marcus | Add additional pipes. |
| **S6-T4** | Marcus | Implement the option to change the pipe entry if multiple connections exist. |
| **S6-T5** | Leo | (Extends from S5.2-T6): Improve menu background look. |
| **S6-T6** | Leo | (Extends from S5.2-T7): Implement rolling background. |
| **S6-T7** | Matthew | Change how the game becomes harder over time. |
| **S6-T8** | Zoe | (Previously S4.2-T17): Implement pausing the game. |
| **S6-T9** | Marcus | Add animations for ascending and descending in pipes. |
| **S6-T10** | Matthew | Source copyright free Music for the game and menu and add it to the game and menu. |
| **S6-T11** | Shawn (Chin) | Overview and interview analysis sections for sprint documentation, and the configuration beta test for development. |
| **S6-T12** | Rachan | Amend meeting records to sprint documentation. |
| **S6-T13** | Michelle | Write preview section for sprint documentation and meeting minutes for customer meeting. |

## **Backlog**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Priority | Tasks | Date of Creation | Date of Completion | Status at End of Sprint |
| Shawn (Chin) 1 | S6-T11 | 01/12/21 | 05/12/21 | Complete |
| Rachan 1 | S6-T12 | 01/12/21 | 07/12/21 | Complete |
| Michelle 1 | S6-T13 | 01/12/21 | 08/12/21 | Complete |
| Dev Team | **PT1** | 10/11/21 | - | Ongoing |
| (1) Marcus | S6-T1 | 21/11/21 | 02/12/21 | Complete |
| (1) Marcus | S6-T2 | 21/11/21 | 02/12/21 | Complete |
| (1) Marcus + Leo | S6-T3 | 01/12/21 | 04/12/21 | Complete |
| (1) Marcus | S6-T4 | 04/11/21 | 04/12/21 | Complete |
| (1) Matthew | S6-T7 | 01/11/21 | - | Ongoing |
| Dev Team | **PT2** | 10/11/21 | - | Ongoing |
| (2) Leo | S6\_T5 | 21/11/21 | 02/12/21 | Complete |
| (2) Leo | S6-T6 | 21/11/21 | 02/12/21 | Complete |
| (2) Marcus | S6-T9 | 01/12/21 | - | Ongoing |
| (2) Matthew | S6-T10 | 01/12/21 | - | Complete |
| Dev Team | **PT3** | 10/11/21 | - | Ongoing |
| (3) Zoe | S6-T8 | 21/11/21 | 02/12/21 | Complete |

## **Meeting Records**

### Meetings

|  |  |  |  |
| --- | --- | --- | --- |
| Overview | Duration | Date & Time | Attended By |
| Internal meeting | 60 minutes | Sunday (5th December 2021) - 17:00 – 18:00 | All Team members |
| Internal meeting | 60 minutes | Tuesday (7th December 2021) - 16:00 - 17:00 | All Team members |
| Client meeting | 15 minutes | Wednesday (8th December 2021) - 12:00 – 12:15 | All Team members |

### Minutes

|  |  |
| --- | --- |
| Time | Wednesday 8th December 11:45-12:00 (Sprint 6) |
| Led By | Leo and Zoe |
| Minutes Taken By | Michelle |

Agenda

1. Demonstrate new features of the game
2. Gaps in functionality
3. Discuss documentation

|  |  |
| --- | --- |
| Agenda Item | Summary |
| Demonstrate new features of the game | * Music, highlight of pipes, new shaped pipes, additionally functionality to choose which side to add on, coins that add to the score, added obstacles, rolling background. * Customer suggests tweaking transparency on pause screen. * Customer asked about levels – team mentioned beat bonus which is being worked on by Matthew. |
| Gaps in functionality | * Discussed gaps in functionality that is feasible to complete between now and the deadline. * Link the score to the high score on the menu as the saving doesn’t currently work. * Pipes can overlap so this needs to be fixed. * Once the game has been cleaned up, it will become deliverable so could potentially add more functionality. |
| Discuss documentation | * Meeting report needs to say what we took from it i.e., who has been assigned what. * Documentation tasks should be included in the backlog. |

Action Items

|  |  |
| --- | --- |
| No. | Action |
| 1 | Tweak transparency on pause screen |
| 2 | Fix the pipe overlap |
| 3 | Extend meeting report with tasks assigned |
| 4 | Add documentation tasks to backlog |

## **Customer Interview and Analysis**

**Interview Highlights:**

*(Demonstrating our game)*

**Q (Client):** Is that another problem?

**A (Team):** That is another obstacle to avoid, or it will hit you.

**A (Client):** It’s just all great. Ok, cool, cool. I can also see you get a lot of background stuff in there, and everything is kind of flowing smoothly.

**Q (Client):** Um, do we have any levels as such?

**A (Team):** Yeah, we are working on the idea of "Beat Bonus" to increase the difficulties and raise the reward of it.

**A (Client):** Ok.

**Q (Client):** What else between now and the end could be done?

**A (Team):** We have the link to the high score on the menu. The saving doesn’t currently work well. Another part is about the pipes overlapping. It doesn’t interrupt the game, but we will fix it. There’s still a lot of things to clean up in general next week, and we want to make it more deliverable as a final product. Anything extra would be a bonus beyond that.

**A (Client):** Yep. Yes. Ok.

**Q (Client):** Is the documentation all on track?

**A (Team):** Yes.

*(Presenting our documentation)*

**Q (Team):** Is that something we must include in the backlog, the documentation part?

**A (Client):** Yep. They would be part of what your project must include.

**Analysis:**

* We were confident that most of our work, including the game and documentation, was on track and doing well so far.
* We can stick to our plans and clean up in general during the next sprint. We can aim to make it more deliverable as a final product.
* We still must split up all the remaining parts of this project, such as the product documentation and optimizations of the game. This should be finished by the next sprint.

## **User Stories**

A new user story was created in this sprint regarding the background of the game (R24), as well as the criteria for checking pipe pieces before adding them to the pipe system (R25)

### R24

|  |  |
| --- | --- |
| User Story: Roll the background R24 | Acceptance Criteria |
| As a Game developer,  I want to implement a rolling background,  So that the player always has a background behind them as they play. | Criterion 1.  **IF** the player is coming to the end of the background image…  **THEN** create a new background image next to the existing one…  **AND** ready the deletion of the current background image for when it is off-screen. |
| Priority: 3/5  Version: 1  Date: 02/12/21 |

### R25

|  |  |
| --- | --- |
| User Story: Pipe overlap prevention R25 | Acceptance Criteria |
| As a Game developer,  I want to prevent the pipe system overlapping onto itself,  So that the realism of the game is increased. | Criterion 1.  **IF** the player tries to add a pie piece to the pipe system…  **AND** the pipe piece in its current orientation would overlap onto the pipe system.  **THEN** prevent this pipe piece being added to the system. |
| Priority: 2/5  Version: 1  Date: 03/12/21 |

## **User Story Tests**

The following test was completed at the end of Sprint 6. If the User Story’s acceptance criteria are fulfilled, the test is passed. R19, and R20 were scrapped from the requirements of the game in this sprint.

|  |  |  |
| --- | --- | --- |
| User Story | Test Result (acceptance criteria) | Reason for failure |
| R1 | Pass | - |
| R2 | Pass | - |
| R3 | Pass | - |
| R4 | Pass | - |
| R5 | Pass | - |
| R6 | Pass | - |
| R7 | Pass | - |
| R8 | Pass | - |
| R9 | Fail | The player’s high score is not saved so cannot be displayed. |
| R10 | Pass | - |
| R11 | Pass | - |
| R12 | Pass | - |
| R13 | Fail | No obstacle functionality implemented. |
| R14 | Fail | No restart buttons implemented. |
| R15 | Pass | - |
| R16 | Pass | - |
| R17 | Fail | No special pipe functionality implemented. |
| R18 | Fail | No beat bonus functionality implemented. |
| R19 | N/A | - |
| R20 | N/A | - |
| R21 | Pass | - |
| R22 | Fail | High score is not saved. |
| R23 | Pass | - |
| R24 | Pass | - |
| R25 | Fail | Pipe piece may still be added such that they overlap in the pipe system. |

## **Use Cases**

Use Case 2 was updated this week.

### UC2

**UC2 - Version 2**

|  |  |  |
| --- | --- | --- |
| **UC2-1** | Use Case | The player plays the game (moving camera, rolling background) - Step 5 of UC1 - (low-level) |
| **UC2-2** | Author | Leo Grant |
| **UC2-3** | Date | 02/12/21 |
| **UC2-4** | Purpose | The player view should continuously move whilst the player is in game, and the background should always be visible. |
| **UC2-5** | Overview | The system moves the camera view continuously to the right. System creates a background image for the camera view to move into, before it reaches the end of the current background. The player clicks the pause button. System stops the camera moving. The player clicks the resume button. The system resumes the camera movement. System creates a background image for the camera view to move into before it reaches the end of the current background. The system eventually declares game over. The system stops the camera movement (and background generation). |
| **UC2-6** | Cross Reference | R21, R24 |
| **UC2-7** | Actors | Player |
| **UC2-8** | Pre-Condition | UC2-Pre-1: The game is playing.  UC2-Pre-1: The game is not paused. |
| **UC2-9** | Post-Conditions | UC2-Post-1: The player has played the game. |

|  |  |  |
| --- | --- | --- |
| **Actor Actions**  3. Player clicks pause button.  5. Player clicks resume button. |  | **System Actions**  1. System moves camera continuously to the right.  2. System creates a background image for the camera view to move into, before it reaches the end of the current background.  4. System stops camera movement.  6. System resumes camera movement.  7. System creates a background image for the camera view to move into, before it reaches the end of the current background.  8. System eventually declares game over.  9. System stops camera movement (and background generation). |

|  |  |  |
| --- | --- | --- |
| **UC2-10** | Alternative Flow of Events: | * None. |
| **UC2-11** | Exceptional Flow of Events: | * None. |

## **Use Case Tests**

The following test was completed at the end of Sprint 6. The test passes if the use case runs as defined.

|  |  |  |
| --- | --- | --- |
| Use Case | Test Result | Reason for failure |
| UC1 | Fail | The high score has not been recorded by the system. |
| UC2 | Pass | - |
| UC3 | Pass | - |
| UC4 | Pass | - |
| UC5 | Pass | - |
| UC6 | Pass | - |
| UC7 | Pass | - |
| UC8 | Pass | - |
| UC9 | Pass | - |
| UC10 | Pass | - |
| UC11 | Fail | No way to restart the game from the beginning whilst in game. |
| UC12 | Fail | System does not validate or store high score. |
| UC13 | Pass | - |
| UC14 | Pass | - |
| UC15 | Pass | - |
| UC16 | Pass | - |

## **CRC Cards**

The following CRC cards were either created or updated in this sprint.

### game\_UI\_Controller

|  |  |
| --- | --- |
| **Class Name:** game\_UI\_Controller | |
| **Version:** 2 | |
| **Cross Reference:** UC1, UC9, UC10, UC11 | |
| **Responsibilities:**  Resume game (i)  Pause game (i)  Restart game  Return to menu (at game over and at pause) (i) | **Collaborators:**  (i) game\_State\_Controller |

Description:

1. The game\_State should be changed by the game\_UI\_Controller to “pause” when the pause button is clicked in game. Additionally, the resume button and restart button should set/re-set the game\_State to “play”, and the ‘return to menu’ button should change the game\_State to “menu”.

### menu\_UI\_Controller

|  |  |
| --- | --- |
| **Class Name:** menu\_UI\_Controller | |
| **Version:** 1 | |
| **Cross Reference:** UC1, UC16 | |
| **Responsibilities:**  Start game (i)  Exit game (i)  Display high score (ii) | **Collaborators:**  (i) game\_State\_Controller  (ii) SaveGame |

Description:

1. The game\_State should be changed by the menu\_UI\_Controller to “play” when the Start button is clicked.
2. The value of the high score is accessed via the SaveGame class, to display on the menu.

### SaveGame

|  |  |
| --- | --- |
| **Class Name:** SaveGame | |
| **Version:** 1 | |
| **Cross Reference:** UC12 | |
| **Responsibilities:**  Store and save the high score and coin count of the player. | **Collaborators:**  (i) score  (ii) menu\_UI\_Controller |

Description:

1. The score class will send a new high score to the SaveGame class for it to save.
2. The menu\_UI\_Controller accesses the high score and coin count from the SaveGame class to display on the menu. When the game is exited from the menu or a save button is pressed the current coin count and high score values should be saved to the save files.

### score

|  |  |
| --- | --- |
| **Class Name:** score | |
| **Version:** 2 | |
| **Cross Reference:** UC1, UC12, UC15 | |
| **Responsibilities:**  Calculate score during play (i)  Display score during game (i)  Set highscore (ii) | **Collaborators:**  (i) game\_State\_Controller  (ii) SaveGame |

Description:

1. The score is only calculated whilst the game is in a play state. The score is only displayed whilst the game scene is active (not menu scene).
2. The score can check with highscore if the new score is higher than the high score. If it is, it will change the value of the field SaveGame in highscore class.

### coin

|  |  |
| --- | --- |
| **Class Name:** coin | |
| **Version:** 1 | |
| **Cross Reference:** UC1 | |
| **Responsibilities:**  Spin the coin object whilst in a state of play | **Collaborators:**  (i) game\_State\_Controller |

Description:

1. Spin the coins whilst the game\_State == “play”.

### background\_Controller

|  |  |
| --- | --- |
| **Class Name:** background\_Controller | |
| **Version:** 1 | |
| **Cross Reference:** UC1, UC2 | |
| **Responsibilities:**  Create new backgrounds as the camera moves  Destroy old backgrounds as the camera moves | **Collaborators:**  (i) game\_State\_Controller |

Description:

1. While the game\_State == “pause” keep waiting to destroy the old backgrounds (so as to avoid them disappearing in view).

### game\_State\_Controller

|  |  |
| --- | --- |
| **Class Name:** game\_State\_Controller | |
| **Version:** 2 | |
| **Cross Reference:** UC1, UC9, UC10, UC13, UC14 | |
| **Responsibilities:**  Set game\_State (“menu” (ii), “play” (i)(ii)(iii)(iv), “game\_over” (iii)(iv), “pause” (ii)(iii)(iv)) | **Collaborators:**  (i) camera\_Controller  (ii) pipes\_Interface  (iii) character\_Controller  (iv) game\_UI\_Controller  (v) obstacle\_Generation  (vi) coins\_Controller  (vii) score  (viii) pipe\_Generation  (ix) menu\_UI\_Controller  (x) coin  (xi) background\_Controller |

Description:

1. The camera moves if the game\_State == “play”.
2. The player is stopped from selecting, rotating and adding pipe pieces to the system if game\_State != “play”.
3. The character stops moving if game\_State == “pause” or “game over”, and continues moving to its target position if game\_State == “play”.
4. The game\_UI\_Controller displays the game over UI when game\_State == “game\_over”. Additionally, the game\_UI\_Controller can set the game\_State to “pause” and then back to “play” or “menu” depending on user input. The pause screen should be blocked from access when game\_State == “game\_over.”
5. Obstacle\_Generation only creates new obstacles in a game\_State of “play”.
6. coins\_Controller should only be generating new coins in a game\_State of “play”.
7. The score is only calculated and displayed to the user whilst the game\_State == “play”.
8. pipe\_Generation should only instantiate pipes whilst the game\_State == “play”.
9. The game\_State should be changed by the menu\_UI\_Controller to “play” when the Start button is clicked on the menu.
10. coin class should spin the coin whilst the game\_State == “play”.
11. While the game\_State == “pause” keep waiting to destroy the old backgrounds (so as to avoid them disappearing in view).

## **User Interface Design**

### Game Progress

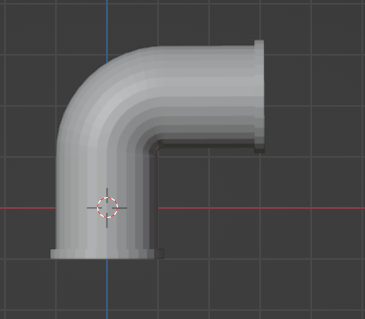


Main menu

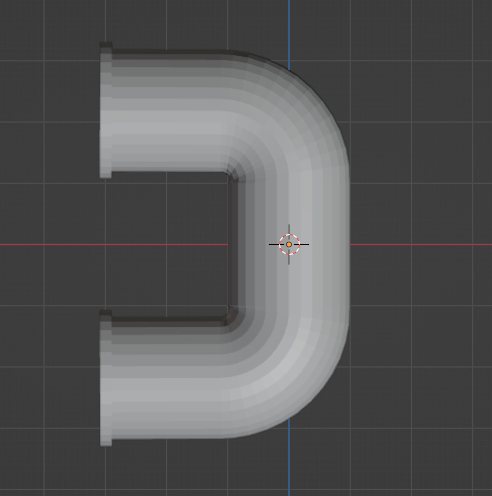


Game view

### Additional pipe generation



Turning pipe, now working both ways



U shaped pipe

## **Exception Handling**

The challenges we faced during this sprint’s tasks are outlined below, alongside their respective solutions:

|  |  |  |
| --- | --- | --- |
| Tasks | Challenges | Solutions |
| S6-T1 | Used the same ideas as pipe generation, so it was relatively straightforward. | N/A |
| S6-T2 | Getting the correct size for the collider so the character did not die when it appeared to o past the obstacle (this was a problem if the obstacle was passed near the front of the screen due to the perspective). | Make the colliders slightly smaller than the obstacle appears. |
| S6-T3 | N/A No challenges. | N/A |
| S6-T4 | Making the character remove the correct way through the pipe if the entry point was changed. | Use the isReversed Boolean created for pipe rotation if the player changes the entry and exit point of the pipe. |
| S6-T5 | The colours and style of the menu were lacking. | Add a new font, add lighting, change the background to a UI sprite so that the colour was more vibrant. |
| S6-T6 | Challenging mechanism to conceptualize before implementation. | Trial and error worked well. |
| S6-T7 | Initially made the game increase speed every frame, but different computers have different frame rates. | Increased speed with a timer (with help from Marcus) instead so it would increase at the same rate for all PC’s. |
| S6-T8 | Ensuring the score did not keep increasing once the user pressed pause. | Increment the score based on the characters x-position rather than by time. |
| S6-T8 | Allowing the user to consider their options whilst on the pause menu. | Increased the transparency of the pause menu background. |
| S6-T9 | N/A Fairly straightforward. | N/A |
| S6-T10 | N/A No challenges. | N/A |
| S6-T11 | N/A Fairly straightforward. | N/A |
| S6-T12 | N/A Fairly straightforward. | N/A |
| S6-T13 | N/A Fairly straightforward. | N/A |