CIS 350 – Team 1 Project #3: Midterm Playtesting Report and Midterm Game

Time to run the playtesting for your team's arcade-style mini-game, take observation notes, write up your report, and make any last changes based on what you learned from the playtesting.

It's time to prepare for your team presentation and in-class game showcase Wednesday 10/12 during class time.

Your goal is to get your team's first mini-game prototype of the semester to meet all of the main requirements for a playable game and then do what you can to **meet your team's chosen design objective or theme** before presentations and game showcases. Now is the time to do what you can to make your team's mini-game prototype as complete as you can.

Project #3: Playtesting Report (100 points)

- **Design objective or theme your team chose AND how you planned to achieve that design objective or design for this theme:**

Deliverable: State the design objective or theme your team chose for this project. Then, in 1-3 short paragraphs, briefly describe how you planned to achieve this design objective or design your team's game for this theme. Do not assume your audience knows anything about this project. Instead, assume this report will be in a portfolio that will be read by a potential future employer or part of an application to graduate school.

Our design objective was to encourage environmental action.

The game was never meant to make players suddenly decide to be an environmental activist and save the world, but rather serve as a small memory that will surface whenever they see trash on the ground or something that can easily be disposed of. They will then think: *maybe I should pick that up.*

The game features a neat landscape that is constantly being polluted by trash, and this is meant to reflect the state of the real world (but on a smaller level).

- **Intended Experience or Desired Outcomes**

Deliverable: Include your team's bullet-point list of about 3-7 thoughts, feelings, knowledge, beliefs, or behaviors you want your game to give players (other than enjoyment or optimal challenge).

- **Frantic stress due to trying to keep up with trash spawns**
- **Relief from the successful "scoring" of throwing trash into the bin (missing wastes time)**
- **Pride from a completed trash run**
- **Frustration from missing shots/wasting time**
- **Desire to pick up trash**
- **Desire to not litter**

- **Game Development**

Deliverable: In 1-3 short paragraphs, briefly describe how you made your game.

Do not assume your audience knows anything about this project. Be sure to include that you used Unity and C# to develop the game, and used an agile project management approach with weekly sprint planning and sprint retrospectives. You can use "we" to describe what your team did, but do not include team member names in this section and do not break down who did what.

In order to make Trash Pickup Simulator, we used the Unity Game Engine and Visual Studio to produce the game and Github to share code between contributors. Our code was written in C# and we organized tasks that needed to be done using Discord and Google Docs. We utilized the technique of weekly sprint planning and sprint retrospectives to make sure tasks were completed efficiently and everyone was contributing equally. To create in-game models we used Blender & Unity Pro-Builder supplemented with Polybrush and Pro-Grids. Most textures were created in Adobe Photoshop and Illustrator.

Techniques to create game mechanics included using Unity's Character Controller to make a first person player object that could be controlled with the Mouse & WASD, Unity's Nav Mesh System to create a responsive AI enemy that would pathfind to the player's location, and Unity's UI system to create a visually pleasing and responsive HUD.

- **Brief Game Design Document:**

Include your One-Page Game Design Document. If it has changed based on feedback you received, be sure to update it. If it has not changed, you can submit the same one.

Deliverable: A single brief game design document

Brief Game Design Document

9/11/2022
Team 1

This template is loosely based on the Project Design Document on Unity's Create with Code Course, but has been expanded and adapted to this course.

Team Members
Colin Gamagami, John Green, Lucas Johnson, Devun Schneider, Zach Wilson,

Game Design Concept

1 **Player Control**	You control a *Tree Hugger*	in this *3D first person*	game
	where	makes the player	

	WASD, Mouse, E, and LMB	*Move, look, pick up, and throw*

2 Basic Gameplay

During the game, *Trash* appear from *The ground,cars, trashcans randomly*

3 Core Game Mechanic

The goal of the game is to

Discard most amount of trash in certain amount of time

What makes this goal challenging or difficult is

Aiming the trash, accessibility of trash cans

Players have the ability to

Pick up and throw trash, move around, look, choose power/length of throwing trash
Throw away certain items to get a powerup (Speed boost, Time Slowdown, etc.)

And when players use their abilities

When players throw trash into the trash can, they earn points

4 Gameplay Mechanics

As the game progresses, *Time goes down* making it *Player has to strategize which trash they seek*

[*optional*] There will also be

Streak multiplier based on consecutive shots made

5 Win / Loss Conditions

The player will win when

The time runs out and they have completed the set goal for throwing away trash

The player will lose when

The time runs out and they have not met the goal.

When the player wins

A victory message will be displayed and their High Score will be saved.

When the player loses

A defeat message will be displayed and the player will be given the option to restart or quit the game.

When the game is over, the player can restart the game or try again from the beginning by

Clicking a Restart or Continue button / Pressing the R key (and a text message tells the player to press the R key to Restart)

6 Sound & Effects

There will be sound effects

- *Win and Lose sound*
- *Successful trash deposit*
- *Powerup gained*
- *Streak gained or lost*

and particle effects

- *Successful trash deposit*
- *Trail of flying trash*

[*optional*] There will also be

7 User Interface

The *score* will *increase* whenever *Trash has been successfully deposited.*

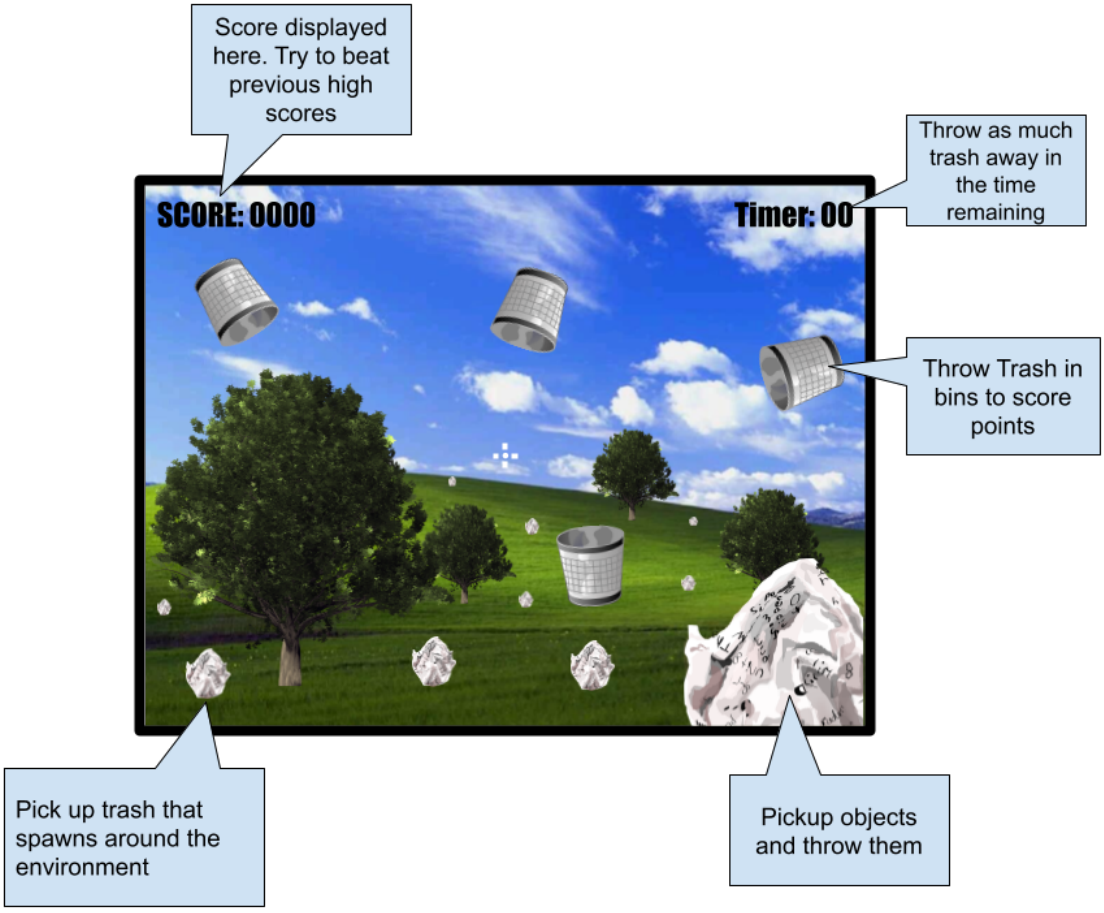
At the start of the game, the title *Trash Pick-Up Simulator* will appear

8 Other Features

Stretch Goal: Upon the win condition player is awarded points that they could spend to get level modifiers (essentially, power-ups that last the entirety of the level), or cosmetic items for player model.
Stretch Goal: Victory screen with visible character model
Stretch Goal: First or third person optional

Game Design Sketch (Annotated with Callouts)

(Also known as a One-Page Game Design Document)



❖ Playtesting Methods

Deliverable: In 1-3 short paragraphs, briefly describe how you playtested your game.

Do not assume your audience knows anything about this project. Instead, assume this report will be in a portfolio that will be read by a potential future employer.

Be sure to include that you recruited classmates for playtesting, the number of participants in the playtest, that the students played your game and then filled out a questionnaire. Be sure to include a link to your team's playtest questionnaire (or include the text of your questions in an Appendix at the end of your document and refer to it in this section with something like "Please see Appendix A for the full list of questionnaire items."). If you used an iterative design approach of testing the game, making changes based on the testing, and testing it again, be sure to mention that.

For Playtesting we recruited classmates, where we had them play our game for five to eight minutes while talking through their thoughts/questions/concerns about the game as they came across them during their playthrough. After they had completed their play through, we had them ask any remaining questions they had for us and if we had any questions regarding anything they said about the game, we made those queries as well. Finally, we had them fill out a questionnaire with some general questions regarding the game and then several specific questions regarding our game that specifically targeted the emotions that we wanted the player to experience while playing.

❖ Observation Notes

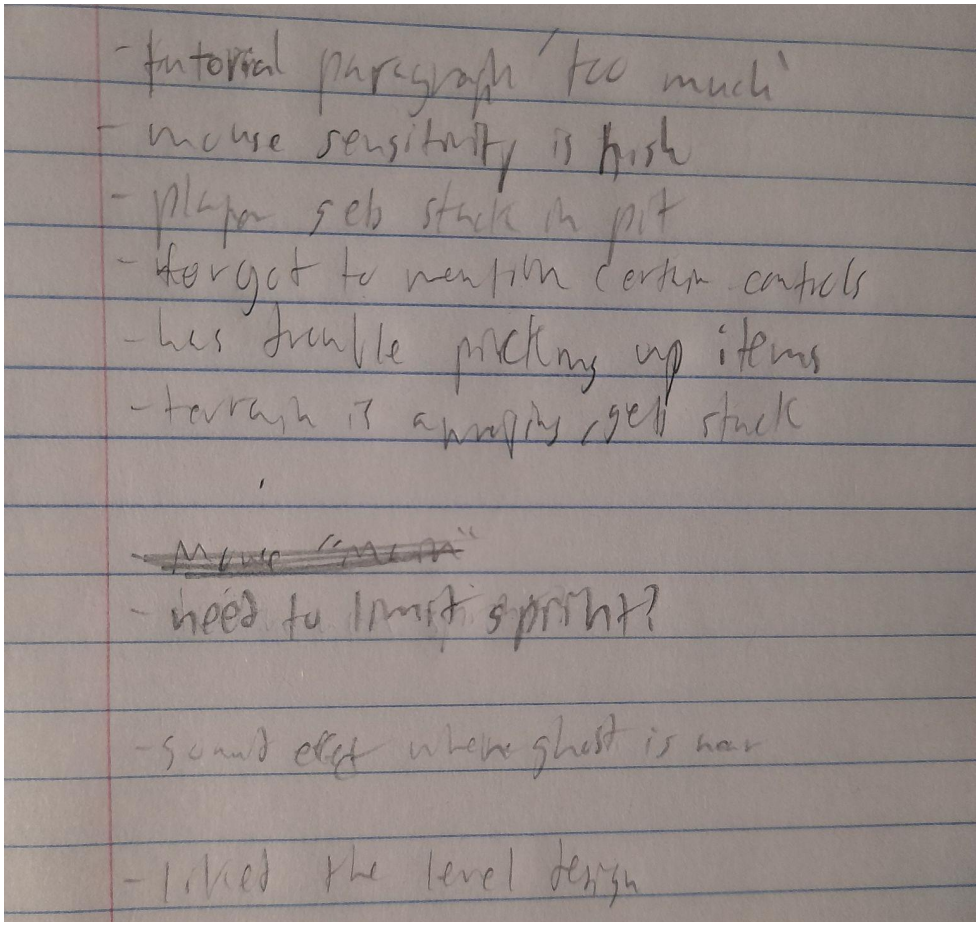
During each playtesting session, one team member at a time will moderate the session and

- ask playtest participants to think out loud,
- remind playtest participants to please think out loud if they go silent.

And at least one team member (it can be the same person moderating the session) needs to

- take notes of what they observe, including
    - what players say (player quotes) and
    - what they do (player behaviors)
    - that are related to what went well and what needed improvement.

Deliverable: Take photos of your team's observation notes (you can use the camera on your phone or a scanner) and attach them under the heading Observation Notes. You will need enough notes to use them in your playtesting report below, so **focus on observing and writing notes that will give you the information you will need to write the report below.**



-tutorial paragraph is 'too much' (the paragraph is difficult to read)

-mouse sensitivity is too high

-player gets stuck in pit

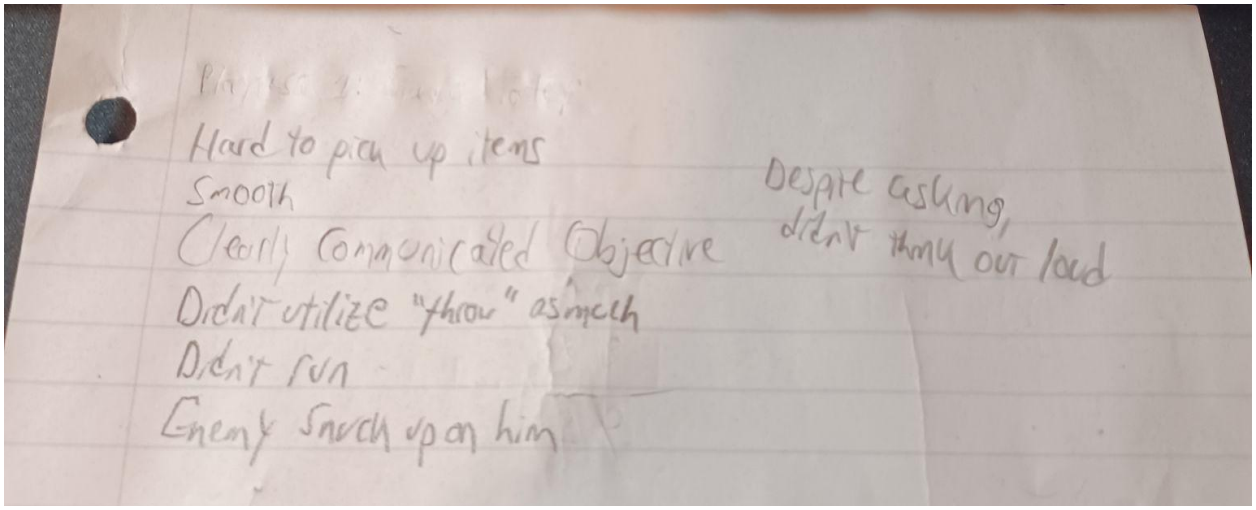
-forgot to mention certain controls

-has trouble picking up items

-need to limit sprint? (sprinting is infinite so there is no need to walk)

-may need sound effect when ghost is near

-enjoyed the level design



- **Summary of Questionnaire Results**

Deliverable: Include a URL link to the summary of your questionnaire results by following the instructions above. Be sure you can see the summary of results at this link even if you are logged out of your google account.

https://docs.google.com/forms/d/19K209o76L2N51p1ekXqVb0uCwnZQzqNAtEI6KWVdFTo/viewanalytics

- **Playtesting Report**

Highlights:

- ☐ **Highlight 1: *Smooth Movement***
  - ☐ **Description:** Some players found the movement really smooth and enjoyed moving around the map
  - ☐ **Quote:** "Oooo, this movement feels so smooth!"
  - ☐ **Action Items:** "None, this is good."
- ☐ **Highlight 2: *Satisfying Aesthetic***
  - ☐ **Description:** Many players enjoyed the aesthetic and environment of our game.
  - ☐ **Quotes:** "I love this environment!" & "These models are really cool!"
  - ☐ **Action Items:** "None, this is good."
- ☐ **Highlight 3: *Enjoyable Enemy***
  - ☐ **Description:** Many players really liked the enemy mechanic including how it functioned and how it looked.
  - ☐ **Quotes:** "Wow, that enemy looks great!" & "It's so cool how it pathfinds around objects!"

- [ ] **Action Items:** “None, this is good.”

Issues:

- [ ] **Issue 1:** *Input Lag*
  - [ ] **Priority:** 3
  - [ ] **Description**: Some players had some input lag and it was hard to control mouse sensitivity
  - [ ] **Evidence**: Players stated this issue out loud and the issue was visible while watching player engage in the game
  - [ ] **Action Items**: Add a mouse sensitivity slider
- [ ] **Issue 2**: *Tutorial is buggy*
  - [ ] **Priority:** 2
  - [ ] **Description**: Jump, Sprint, Look Around Tutorials not displayed, and it appears when it is not supposed to.
  - [ ] **Evidence**: Bugs were visible during playtest
  - [ ] **Action Items**: Put tutorial into a new scene
- [ ] **Issue 3**: *Can’t Pause the game*
  - [ ] **Priority:** 2
  - [ ] **Description**: The game can not be paused.
  - [ ] **Evidence**: Player asked if the game could be pause and it cannot be
  - [ ] **Action Items**: Add pause menu
- [ ] **Issue 4**: *Hard to pick up objects*
  - [ ] **Priority:** 1
  - [ ] **Description**: Objects are difficult to pick up; requiring the player to look directly at an object and be close enough to pick up.
  - [ ] **Evidence**: Players vocally complained that they were unable to pick up objects smoothly
  - [ ] **Action Items**: Expand pickup range and use a cone instead of a direct raycast
- [ ] **Issue 5**: *Paper object looks like a rock*
  - [ ] **Priority:** 5
  - [ ] **Description**: Paper object is too gray and looks like a rock
  - [ ] **Evidence**: Players passed over the paper object more frequently and often said “Oh, I can pick that up? I thought it was a rock.”
  - [ ] **Action Items**: Make object white to look more like paper
- [ ] **Issue 6**: *Difficulty balancing*
  - [ ] **Priority:** 2
  - [ ] **Description**: Players thought game was too hard
  - [ ] **Evidence**: Long timer and insta-kill trash enemies
  - [ ] **Action Items**: Add decrementing health bar when in proximity to enemies

- **Sprint Retrospective:**

  Zach - Contributed to project document (Done)

  Lucas - Contributed to project document (Done)

  John - Contributed to project document (Done)

  Colin - Contributed to project document (Done)

  Devun - Contributed to project document (Done)

- **Sprint Planning:**

  Zach - Contribute to project document, Fix "Input Lag" issue

  Lucas - Contribute to project document, Fix "Hard to pick up objects" issue

  John - Contribute to project document, Fix "Can't pause the game" and "Difficulty balancing" issues

  Colin - Contribute to project document, Fix "Paper object looks like a rock" issue

  Devun - Contribute to project document, Fix "Tutorial is buggy" issue

- **Completed Plan to Make Changes Based on Playtesting**

Issue:	Priority Rating:	Status:	Deadline:
Hard to pick up objects	1	Completed	10/15
Input Lag	2	Completed	10/15
Can't Pause the game	2	Completed	10/15
Tutorial is buggy	3	Completed	10/15
Paper object looks like a rock	5	Completed	10/15
Difficulty balancing	2	Completed	10/15

- **Team Project Mini-Game Prototype:** I want to see the final version of your team's mini-game prototype.

Submit one GitHub link and one Simmer.io (or itch.io) link.

Be sure your game meets all of the requirements for what counts as a playable game prototype listed in project 2.

Create a GitHub repo, get the .gitignore file on Canvas Modules, add the .gitignore file to your repo, then add your Unity project folder to the repo. Build your game to WebGL and post it on Simmer.io (or a similar website like itch.io).

Deliverables: **A working Simmer.io link** where your game can be played online. Be sure to test your Simmer.io link to make sure your game is playable at that link. **A link to your GitHub repo** with your source files. Along with the URL to your GitHub repo, **write which scene you want me to look at** and **the path to the scene file from your project directory**. I will take off points if you do not use a .gitignore file. So please remember that step.

Git Repo URL - https://github.com/PlatFormPlayZ/CIS-350-Project-1

Simmer URL - https://simmer.io/@Ronis/trash-pick-up-simulator

The scenes in the following directory are part of the game: https://github.com/PlatFormPlayZ/CIS-350-Project-1/tree/main/Trash_Pick-Up_Simulator/Assets/Scenes ignore the unused scenes folder and the sample scene