



University of the West of Scotland

CREATIVE COMPUTING TECHNOLOGIES

GAME DESIGN DOCUMENT (GDD)

SORTICA - ALGORITHMIC INVENTORY
SORTER

Frank Antonio Oldfield Montilla

SEPTEMBER 11, 2025

Key Notes

- These days, GDD's tend to be shorter, not longer. Challenge yourself to explain features concisely, use visuals, and leave some parts open to the team, sometimes less details are better, especially if the scope of your game is large (No one wants to read a 100 pages long GDD).
- Pictures are worth a thousand words.
- Make this as you visualize your game: Even if it doesn't feel like you'll be able to gather a full team to take on all the necessary tasks, plan as if you have a full team, afterwards you can scale it to size and prioritize elements. If you feel like you want a feature like that in your game, put it in here.
- If one of the sections doesn't make sense for your game, feel free to delete it from the final document or use "Not applicable".
- Make sure the final version is coherent; re-read it several times and send it to friends first. If other people understand it with a simple read, then you're doing great.
- Have fun!

Contents

1	Introduction	4
1.1	Scope of the document	4
1.2	Elevator pitch	4
2	Game Overview	4
2.1	Game concept	4
2.2	Audience	5
2.3	Genre	5
2.4	Setting	6
2.5	Game structure	6
2.6	Player	7
2.7	Game flow summary	7
2.8	Look & Feel •	8
3	Gameplay	8
3.1	Objectives	8
3.2	Progression	8
3.2.1	Challenge structure	8
3.3	Play flow	8
3.4	Difficulty	9
4	Mechanics	9
4.1	Rules	9
4.2	Game universe	9
4.3	Physics	9
4.4	Input Queue	10
4.5	Output Queue	10
4.6	Schedule History	10
4.7	Buffer Desk	10
4.8	Economy	10
4.9	Character movement	11
4.10	Player interaction	11
4.10.1	Game menus •	12
4.10.2	Saving	12
4.10.3	Game options •	12
4.11	Assets •	12
4.11.1	Player/Characters	12
4.11.2	Sortica Headquarters	13
4.11.3	Sorting Street	13
4.11.4	Levels	13

5 Graphics and audio	14
5.1 Visual system	14
5.1.1 Player camera	15
5.1.2 Landscape	15
5.2 Interface	16
5.2.1 Game HUD	16
5.3 Audio system	16
5.3.1 Game music ●	16
5.3.2 Audio look & feel	16
6 Story and narrative	16
6.1 Backstory	16
6.2 Main plot	17
6.2.1 Plot progression	17
6.3 Cutscenes	17
7 Characters	17
7.1 Main character - Sortie	17
7.1.1 Backstory	17
7.1.2 Personality	17
7.1.3 Appearance	17
7.1.4 Abilities	18
7.1.5 Relationships	18
7.2 Supporting character - Rollie/Roller/Scrollie	18
7.2.1 Backstory	18
7.2.2 Personality	18
7.2.3 Appearance	18
7.2.4 Abilities	19
7.2.5 Relationships	19
8 Game world	19
8.1 Look & Feel of the world ●	19
8.2 Locations	20
8.2.1 Sorting Street	20
8.2.2 Sortica Headquarters	20
8.3 Levels	21
8.3.1 Tutorial level	21
8.3.2 Main levels	22

1 Introduction

Important! Some terms like queue and scheduling algorithm, among others, have been used in an arbitrary way to make this document more understandable.

A GitHub repo is already up with information regarding the development of this project. <https://github.com/frankoldfield/algorithmic-inventory-sorter>.

There is also a website available with information about this game and the development process. <https://frankoldfield.github.io/sortica/>

1.1 Scope of the document

This document is intended for anyone involved in the production of the video game or any related products such as research papers or podcast episodes. Specifically, researchers at the Creative Computing Technologies lab at University of The West of Scotland. It will also be part of the handoff documentation, for anyone that might be involved in the development of this game in the future. Or annexed as part of a research publication.

1.2 Elevator pitch

Sortica - Algorithmic Inventory Sorter is an immersive Serious Game (SG) where users will have to solve 10-20 minute levels in simulated scenarios while learning about scheduling algorithms. It will be intended to be used as a learning tool in undergraduate courses, alternative to traditional methods like lectures or problem solving. By using novel immersive technologies, we aim to have a higher engagement and knowledge retention for students with or without additional educational needs.

2 Game Overview

2.1 Game concept

The player will play as a robot, part of a restoration crew, in an enterprise called Sortica, whose main goal is the restoration of the planet. Particularly, their sorting crew's goal is the complete restoration of Sorting Street, home to several local businesses. To achieve this goal the players will have to solve puzzle-like simulations in different settings, (e.g., a restaurant and a factory). Every time the player completes the puzzle, that level will be considered completed, and the corresponding building will be completely restored.

The main objective is that the players learn about queuing algorithms while enjoying the game and showing an improved learning progress compared to more traditional means. For this, it is important that the **level of difficulty is well suited and not arbitrary for each user**.

Learning Objectives

The general learning objectives (*LO*) of this serious game are:

- *LO1*: After playing the game, the learner will be able to explain and apply the concept of Scheduling Algorithm.
- *LO2*: After playing the game, the learner will be able to explain and apply the concept of Queue.
- *LO3*: After playing the game, the learner will be able to explain and apply the concept of Stack.
- *LO4*: After playing the game, the learner will be able to explain and apply the concept of Multi-Level Queue (MLQ).
- *LO5*: After playing the game, the learner will be able to explain and apply the concept of Priority.
- *LO6*: After playing the game, the learner will be able to explain and apply First-in-First-Out (FIFO) algorithms.
- *LO7*: After playing the game, the learner will be able to explain and apply Last-in-First-Out (LIFO) algorithms.
- *LO8*: After playing the game, the learner will be able to explain and apply FIFO algorithms involving multiple queues with different priorities.
- *LO9*: After playing the game, the learner will be able to explain and apply Round-Robin algorithms.
- *LO10*: While playing the game, the learner will be able to demonstrate problem-solving skills using interactive, hands-on gameplay mechanics.
- *LO11*: While playing the game, the learner will engage with accessibility features that support diverse visual, auditory, and motor needs while completing in-game tasks.

2.2 Audience

The people who'll play the game are undergraduate students from degrees like computer science. Some will play this game as a part of a test study, to figure out the impact on their knowledge, these participants will also be subject to a stealth game-based assessment. Others will play this game as part of the learning process in lessons related to queuing algorithms. They will be 18 years or older.

2.3 Genre

This is a Serious Game, specifically, a simulator where players will have to solve puzzles.

2.4 Setting

Sortica takes place in a post apocalyptic world, after a climate disaster, but some special chapters may take place in other fictional environments, this can act as a way of implementing new levels, without them being too repetitive, special chapters can show the perspective of other restoration crews destined in other areas of the planet. Having a setting this typical helps link the game world with the main goal.

2.5 Game structure

Sortica will initially have 5 levels, one will act as a tutorial, and the other 4 will be completed linearly, each time a level is completed, the corresponding building will be restored, and the next level will be unlocked.

- **1st Level - Tutorial:** In this level, the player will have to use a FIFO algorithm.
 - Concepts: FIFO, Queue, Scheduling Algorithm.
 - Learning Objectives: *LO1, LO2, LO6, LO10, LO11.*
- **2nd Level - Factory:** In this level, the player will have to use a LIFO algorithm.
 - Concepts: LIFO, Stack, Queue, Scheduling Algorithm.
 - Learning Objectives: *LO1, LO2, LO3, LO7, LO10, LO11.*
- **3rd Level - Pastry Shop:** In this level, the player will have to use a FIFO algorithm, some items will have already been queued, and the player will have to fix the order by using the Schedule History Board.
 - Concepts: FIFO, Queue, Scheduling algorithm.
 - Learning Objectives: *LO1, LO2, LO6, LO10, LO11.*
- **4th Level - Warehouse:** In this level, the queue will have two levels, and the player will have to use a priority based FIFO algorithm, the premium user level of the queue will have priority over the regular user level.
 - Concepts: FIFO, Queue, Scheduling algorithm, Priority, Multi-Level Queue.
 - Learning Objectives: *LO1, LO2, LO4, LO5, LO6, LO8, LO10, LO11.*
- **5th Level - Restaurant:** In this level, the queue will have three levels, and the player will have to use a Round-Robin algorithm, each level will represent a different type of plate (Appetizer, Main course, Dessert) and will have different priorities.
 - Concepts: FIFO, Queue, Scheduling algorithm, Priority, Multi-Level Queue, Round-Robin.
 - Learning Objectives: *LO1, LO2, LO4, LO5, LO6, LO8, LO9, LO10, LO11.*

2.6 Player

The game will be single player. The player will control a character whose goal is the restoration of Sorting Street. While doing this, they will have to move through levels, completing the proposed puzzles.

2.7 Game flow summary

As previously mentioned, the game will revolve mainly around Sorting Street and the different levels. Sortica Headquarters will act as a lobby/menu. In Figure 1 we have a flowchart describing the main gameloop.

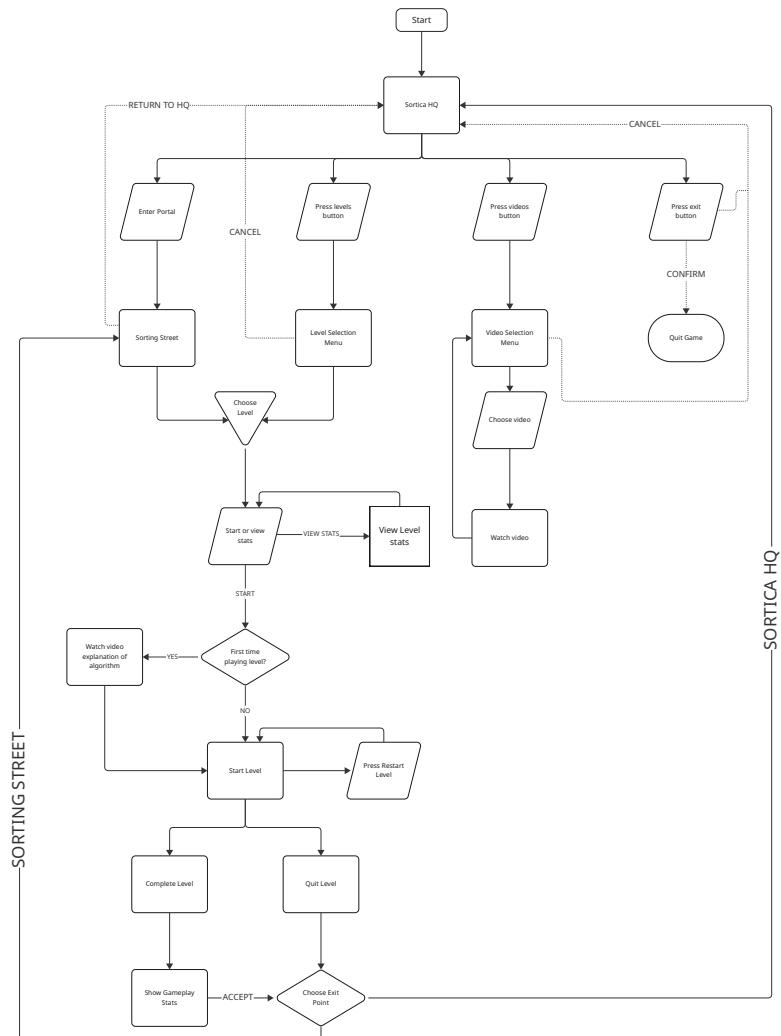


Figure 1: Game flowchart

2.8 Look & Feel •

We want the game to be fairly simple, the goal is to make it feel a bit like Portal but less technical/cold, with a warmer feel. We don't want the game to be distracting. **I think in this matter, Luke's guidelines might help me.**

The game will have two main locations, Sorting Street, where the levels are located, and Sortica Headquarters, which will act as a base/main menu.

3 Gameplay

3.1 Objectives

The main objective for the player is beating all the levels and completing the reconstruction of Sorting Street. Secondary objective is completing all the levels in the hardest difficulty.

3.2 Progression

The goal in each level is to optimize scheduling processes, in each level they will have one or more incoming queues of objects, a set of instructions to sort the objects on their way out, and a set of outgoing queues, once the player has successfully sorted these objects, they will have optimized the scheduling process in that level. Completing a level will unlock the next one.

3.2.1 Challenge structure

The first level after the tutorial will be a basic level, one queue and FIFO algorithm. After the tutorial, each level will add a layer of difficulty progressively by introducing new concepts the user will have to apply in addition of the previous ones.

- **1st Level - Tutorial:** FIFO, Input queue, Output queue, Schedule History, Buffer Desk, Queue.
- **2nd Level - Factory:** LIFO, Stack.
- **3rd Level - Pastry Shop:** Reordering.
- **4th Level - Warehouse:** Priority, Two-level queue.
- **5th Level - Restaurant:** Three-level queue, Round-Robin.

3.3 Play flow

After selecting a level, if it is not their first time playing it, the level will start right away. Otherwise, the player will have to view a short video explaining the queuing algorithm they will be using beforehand. After viewing this short video, the level will start. Once the level starts, they will have to apply said scheduling algorithm in order to solve the puzzle in that level. After completing the level, the stats from their gameplay will be

shown to them. Finally, they will return to where they choose, Sorting Street or Sortica Headquarters.

3.4 Difficulty

Initially, a difficulty setting for each level was going to be implemented. But it would be redundant, if we already have a learning curve produced by increasing difficulty among different levels, it becomes unnecessary, and it would probably make the game feel more repetitive and less immersive. The difficulty of the levels will be adjusted with both the number of queues and the scheduling algorithm to be used.

4 Mechanics

4.1 Rules

- The player can't modify any assets from the game, they can only move the objects it has to queue and interact with other assets, in a determined way.
- When an object arrives from an input queue, the player must grab it, and then move it to either an output queue, or to an Buffer Desk.
- If the player thinks he has made a mistake, he can fix it by reordering the small items displayed on the scheduled history board, these represent the items he has already put on an output queue.

4.2 Game universe

Each level, Sorting Street, and Sortica Headquarters, will be different scenes, managed by a master script, when the player performs an action that will take him to another location, the master script will be in charge of saving the state of the current scene, load the new scene, and start it.

The game will keep all the items to be sorted out of sight, and will provide them to the player through the input queue. Once the player puts an item on the output queue, its identifier will be added to a data structure that will store only the order of the items, the items will then be deleted. Every time the player modifies the order using the schedule history board, the game will modify the data structure that stores the order to reflect the output queue correctly. The game will check if the items on the output queue are in correct order every time the user either reorders them using the board or adds an item to the output queue. In case the order is correct, the level will have been completed.

4.3 Physics

Realistic physics. The only remarkable thing to mention is that the user levitates (in reality it's just that his character is floating, it can't fly or move vertically if it doesn't jump), this is only for aesthetic purposes, this way, we can make the player smaller and more appealing, without its camera being too low.

4.4 Input Queue

This asset will provide the player with the items they will have to interact with, the items will not necessarily come in order. When the player grabs an item, the next one will be dispatched by the Input Queue.

4.5 Output Queue

The Output Queue is where players will place items they think are already in the correct order. After placing an item on the Output Queue, said item will disappear. After disappearing, the item will be represented in the Schedule History.

4.6 Schedule History

This asset allows the player to visualize the order they have introduced the objects into the output queue in, and change it after making any mistakes. For example, in Figure 2, the player would have to select the item number 2, and then the item number 1, the two would swap spots, and the order would then be correct.

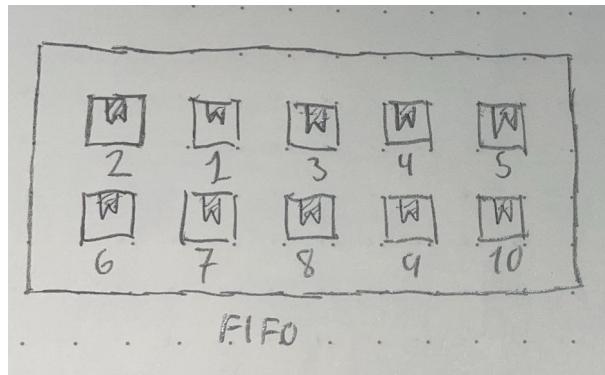


Figure 2: Schedule history board

4.7 Buffer Desk

This desk acts as a buffer, it allows the player to store objects and reorder them without putting them into the output queue, for example, in a LIFO scheduling algorithm, the player will have to put every object on this desk, and then start putting them into the output queue in reverse order.

4.8 Economy

Initially, we had thought of implementing an economy, to make completing levels more appealing, but users would only be able to buy cosmetic items, we think that this could prompt the users to engage in the game in a more repetitive way, that could halt the learning process. Because of this, we have decided not to implement an economy system.

4.9 Character movement

The player can move around every location, some will have visible barriers (Every level, and Sortica Headquarters), and others will have invisible barriers (Sorting Street). The default movement type will be Continuous movement.

4.10 Player interaction

The player can interact with the helper NPC at all times. In Sorting Street, the player will be able to select the level they want to play, or read information about it. During levels, they will be able to interact with the objects they have to sort, with the queues, and with other elements such as the Buffer Desk and the queuing history board.

The actions the player can perform are:

- **Interact:** By interacting with an element, the player doesn't alter the game world, but triggers a series of events.
 - **Start level:** By interacting with the building, the player can start the level.
 - **Restart level:** By interacting with the restart button, the player can restart the level.
 - **Exit level:** By interacting with the exit button, the player can exit the level.
 - **Talk with Helper NPC:** By interacting with the helper NPC, the player will receive verbal instructions.
 - **Toggle settings:** By interacting with a setting button, the player can turn on or off, the selected feature.
 - **Select Schedule History Board item:** By interacting with an item displayed on the Schedule History Board, the player will have now selected it, after selecting it, he can then interact with other position in the Schedule History Board, if the other position is empty, the item will be placed in said position, if it is not empty, then the two items will swap positions.
- **Grab:** The player can grab the items he will have to sort. He can only grab an object if he is not currently carrying one.
 - **Remove from input queue:** By grabbing an item currently placed in the input queue, the player will now have it on him, and the input queue will dispatch the next object.
 - **Remove from Buffer Desk:** By grabbing an item currently placed in the Buffer Desk, the player will now have it on him, and the spot where the item was placed will now be free.
- **Release:** The player can release items that he is currently holding, by doing this, the item will be placed on the position where the user is pointing.
 - **Add to output queue:** By releasing an item on top of the output queue, the item will be fed into the queue, making it disappear. Also, the Schedule History Board will now show a small item representing the item the player added to the output queue and its position in it.

- **Add to Buffer Desk:** By releasing an item on top of a free spot in the Buffer Desk, the item will now be placed in that free spot.
- **Walk or run:** The standard moving setting is walking, but the player can run to move faster.

4.10.1 Game menus •

The menus will be developed as diegetic interfaces, settings will be accessible in Sortica Headquarters. Controls and the option to exit a level will be present in every level. **Still need to add sketches**

4.10.2 Saving

Every time the player completes or exits a level the game will save automatically.

4.10.3 Game options •

The player will be able to use any of the implemented accessibility measures, they will also be allowed to change sensibility and sound volume. The player will be allowed to change the controller bindings to allow for better accessibility. **Still need discussing and linking with accessibility guidelines.**

Visual accessibility features

- Disable complex models (everything turns into basic models)
- Colour filters for colour blindness

Auditory accessibility features

- Disable sounds
- Sound queues signaling next step

Spatial accessibility features

Motor accessibility features

- Controls personalization.

4.11 Assets •

Still need to discuss sounds. Still need to attach sketches.

4.11.1 Player/Characters

- Player model
- Player textures
- Player animation/movement
- Helper model
- Helper texture
- Helper animation/movement
- Chef model
- Chef texture
- Chef animation/movement

4.11.2 Sortica Headquarters

- Room model
- Terminal model
- Terminal texture
- Panel model
- Panel texture
- Portal model
- Portal texture
- Wall/Flooring/Ceiling textures

4.11.3 Sorting Street

- Street texture
- Sky texture
- Factory model
- Factory texture
- Pastry shop model
- Pastry shop texture
- Warehouse model
- Warehouse texture
- Restaurant model
- Restaurant texture
- Fence model
- Fence texture
- Bush model
- Bush texture

4.11.4 Levels

Test

- Simple Queue model
- Simple Queue texture
- Simple box model
- Simple box texture
- Simple board model
- Simple board texture
- Simple table model
- Simple table texture

Factory

- Factory Queue texture
- Metal rods model
- Metal rods material
- Factory board texture

- Factory table texture
- Factory furnace model
- Factory furnace texture

Pastry shop

- Pastry Shop Furnace Queue model
- Pastry Shop Furnace Queue texture
- Pastry model

- Pastry texture
- Pastry Shop board texture
- Pastry Shop table texture

Warehouse

- Warehouse Queue texture
- Warehouse box texture
- Warehouse board texture

- Warehouse table texture
- Warehouse truck model
- Warehouse truck texture

Restaurant

- Restaurant Queue texture
- Appetizer Model
- Appetizer texture
- Main Course Model
- Main Course texture
- Appetizer Model

- Appetizer texture
- Restaurant board texture
- Restaurant table texture
- Restaurant door model
- Restaurant door texture

5 Graphics and audio

5.1 Visual system

It is a 3D minimalistic game. One of the reasons why it is minimalistic is to reduce distractions. The shapes and textures will be simple, similar to games like "Fall Guys", "Firewatch", "Outer Wilds" or "Roblox" (See figure 3).



(a) Fall Guys



(b) Firewatch



(c) Outer Wilds



(d) Roblox

Figure 3: Examples of different games.

5.1.1 Player camera

VR First person view.

5.1.2 Landscape

Landscape will only be actually applicable in Sorting Street, it will be an infinite street towards both ends of the street, and a cloudy sky (See Figure 4).

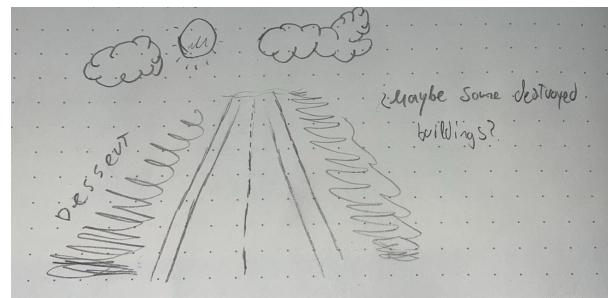


Figure 4: Landscape

5.2 Interface

5.2.1 Game HUD

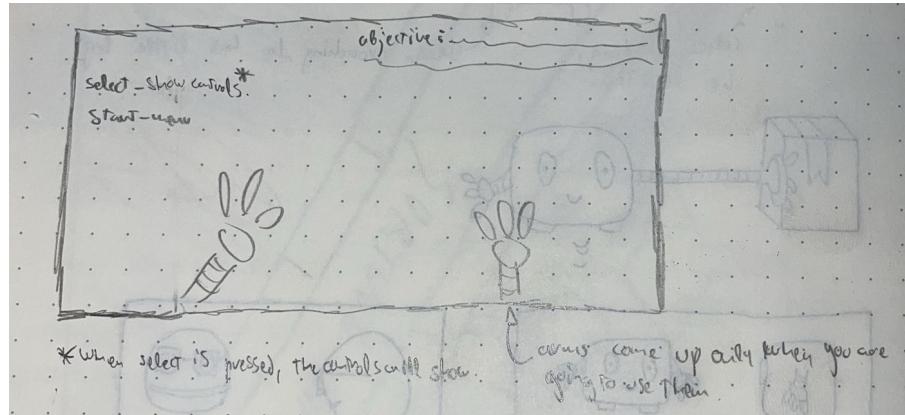


Figure 5: Basic user hud

When pressing select (placeholder) the controls will show on screen. When pressing start (placeholder), the pause menu will pop up. The option to have the controls show on screen is mainly for accessibility purposes, because some people find it harder to remember controls, it is important to facilitate the way they can double check. This way, they will be less distracted from the game.

5.3 Audio system

5.3.1 Game music •

If the game has any music, it will be background, relaxing music, we need to avoid complex music to reduce distractions and disorientation. (**Still need to discuss the use Music Therapy**)

5.3.2 Audio look & feel

We want the audio to feel simple and robotic, yet a bit comical. Again, it can't be distracting, there shouldn't be any strong noises.

6 Story and narrative

6.1 Backstory

This backstory is told by our robot coworker. The world is set in a post apocalyptic setting caused by an environmental disaster. After several years of having to live underground, Sortica INC was founded, with the goal of making the surface habitable again, one of its divisions is the Sorting division, whose objective is making sure every process that takes place on the surface is optimized, to ensure no resources are wasted.

6.2 Main plot

We are part of a restoration crew at Sortica INC, in the Sorting division. Our goal is to complete the restoration of Sorting street, by optimizing the way the restored businesses/buildings manage their processes (applying queuing algorithms). If we are not successful in our duties, we will be considered an unfit model and will be destroyed and replaced by newer models.

6.2.1 Plot progression

The supporting character will inform us on how the completion of each level has impacted the restoration of Sorting Street.

6.3 Cutscenes

We will have cutscenes that explain the algorithms that will be used in each level.

7 Characters

7.1 Main character - Sortie

His full name is Levitating Sorting Restoration Robot version 13.9.2.

7.1.1 Backstory

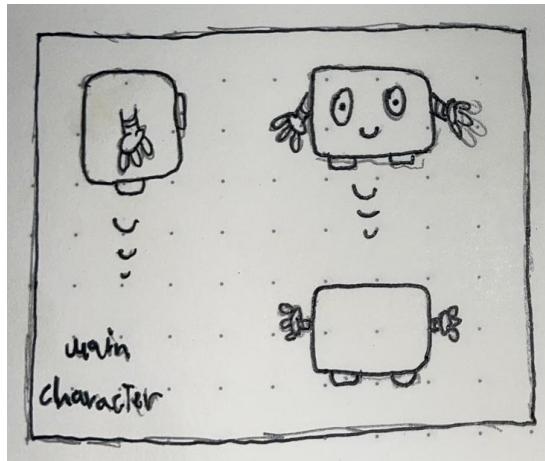
The main character (You) is a robot from a restoration crew, whose goal is to restore the buildings in sorting street.

7.1.2 Personality

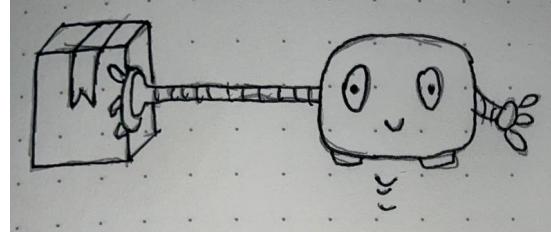
The main character doesn't talk too much, it has a naive personality, and just smiles and nods at everything it is told to do.

7.1.3 Appearance

Small rounded rectangular prism, with stretching arms and little legs. When the character is crouching, it has little legs, when it isn't, it levitates.



(a) Views



(b) Arm extended

Figure 6: Sortie

7.1.4 Abilities

It can levitate, and grab objects from afar with its extendable arms.

7.1.5 Relationships

It is new to the job, other employees treat him just as a regular new employee.

7.2 Supporting character - Rollie/Roller/Scrollie

His full name is Rolling Sorting Restoration Robot version 5.9.

7.2.1 Backstory

The supporting character is in this case an older model of restoration robot, it is now retired of his restoration duties and is dedicated to instructing and teaching newer models.

7.2.2 Personality

It is more apathetic than the main character, it takes his job very seriously.

7.2.3 Appearance

Sphere with eyes, glasses, and mouth.

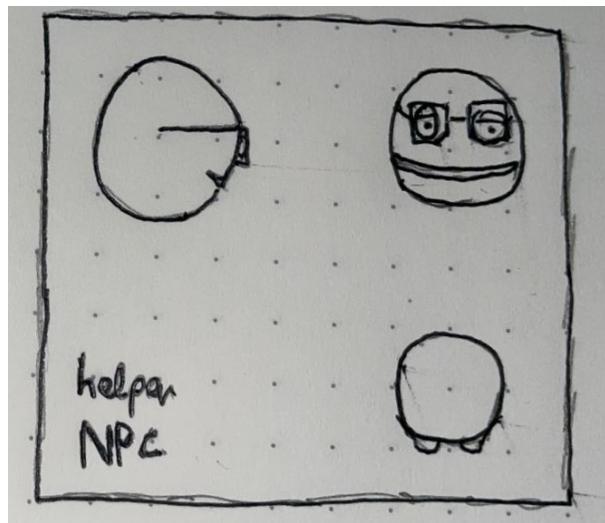


Figure 7: Rollie - Model views

7.2.4 Abilities

It rolls and it teleports, it is omnipresent.

7.2.5 Relationships

It is the supervisor of the main character.

8 Game world

8.1 Look & Feel of the world •

The parts of the world that are out of reach or have not been restored have look old and tore down, the parts of the world where we move around and have restored have to look modern, simple and clean. (Still need to link to accessibility guidelines and test)

8.2 Locations

8.2.1 Sorting Street

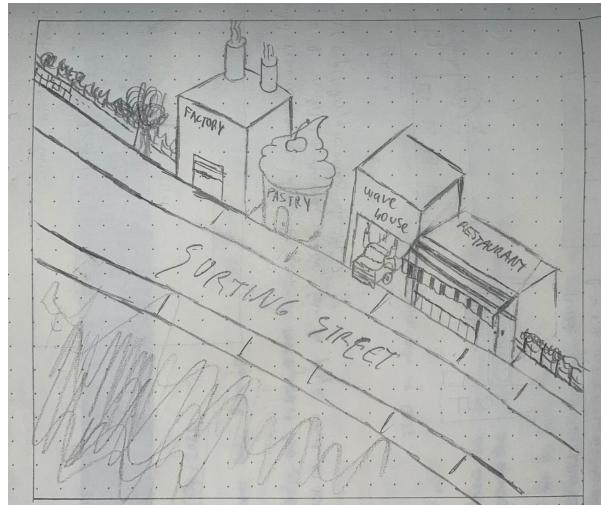


Figure 8: Sorting Street

Connection to the plot

Main location, this is the street the player will have to restore, in this street, each building located in Sorting Street is a level.

8.2.2 Sortica Headquarters

This location acts like a main menu, here, the player can choose between going to Sorting Street, viewing old videos, replay specific levels, or going to the appearance store (TBD).



Figure 9: Sortica Headquarters

Connection to the plot

The headquarters of the restoration company.

8.3 Levels

The levels will follow the same structure, this facilitates design, reproducibility and escalation.

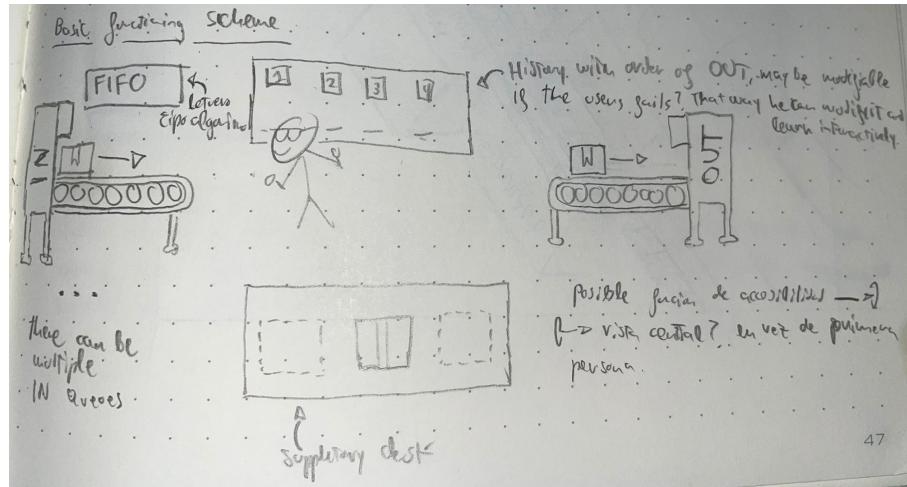


Figure 10: Basic Level Design

8.3.1 Tutorial level

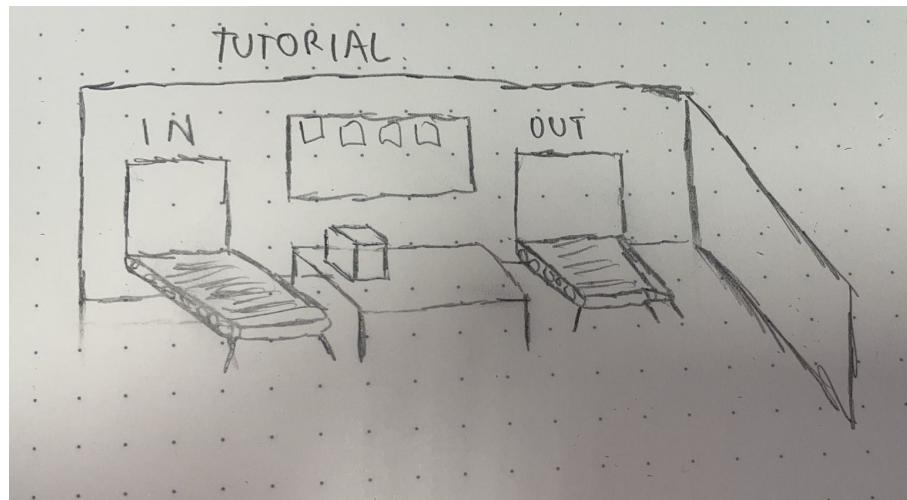


Figure 11: Level 0 - Tutorial

- Queuing Algorithm: FIFO
- Input queues: 1
- Output queues: 1

8.3.2 Main levels

Factory

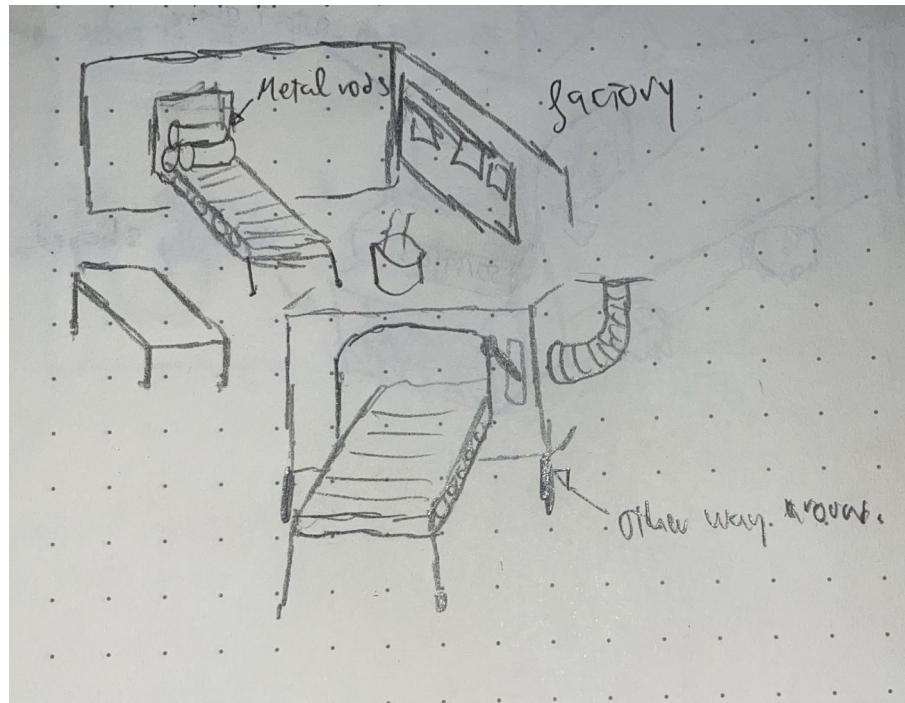


Figure 12: Level 1 - Factory

- Queuing Algorithm: LIFO (Explanation could be they want to retain newer customers with lower fabrication times)
- Input queues: 1
- Output queues: 1

Pastry shop

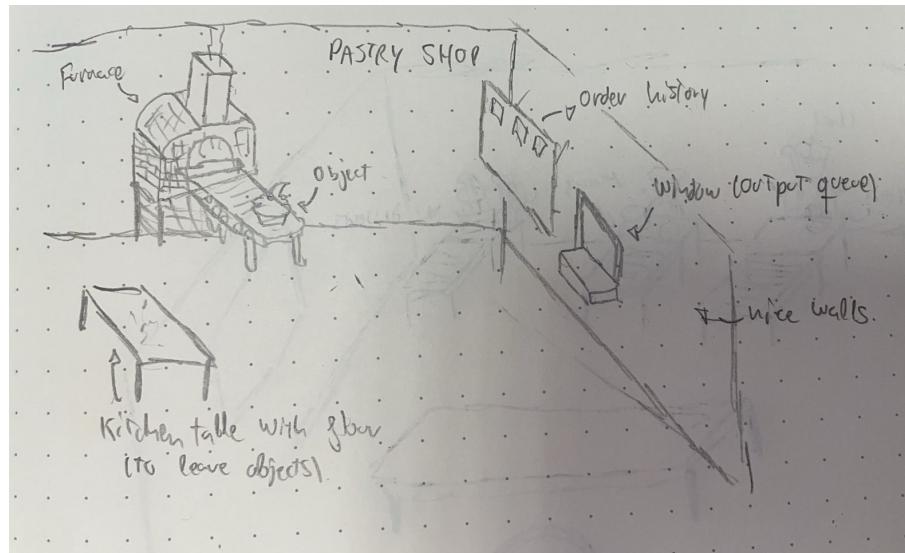


Figure 13: Level 2 - Pastry Shop

- Queuing Algorithm: FIFO
- Input queues: 1
- Output queues: 1

Delivery warehouse

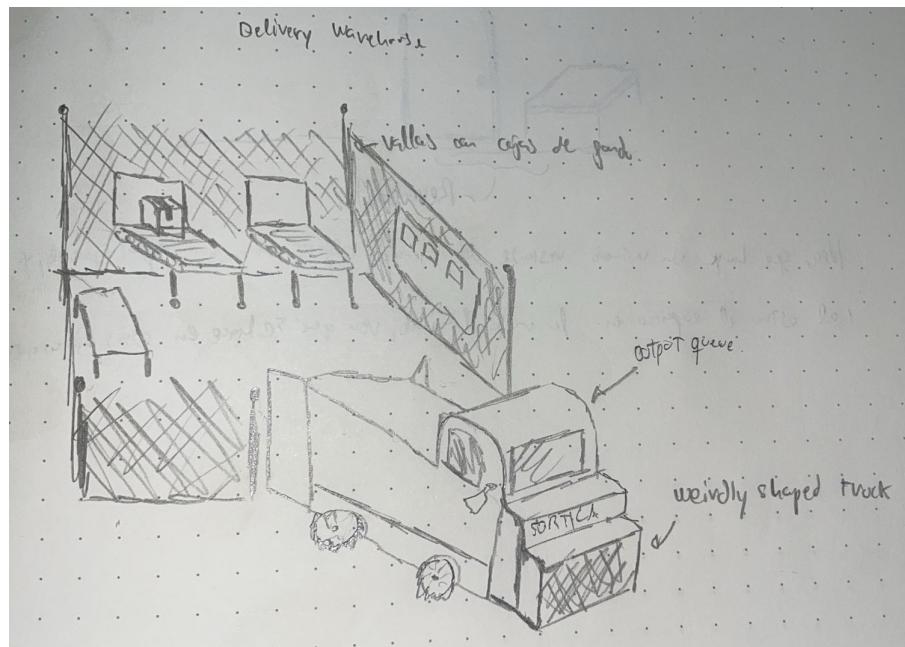


Figure 14: Level 3 - Warehouse

- Queuing Algorithm: FIFO with multiple queues with different priorities.
- Input queues: 2 (Premium user and normal user)
- Output queues: 1

Restaurant

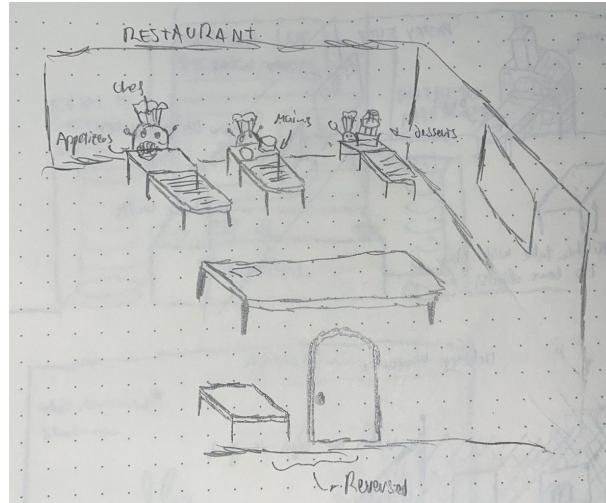


Figure 15: Level 4 - Restaurant

- Queuing Algorithm: FIFO, weighted (time) priority algorithm
- Input queues: 3 (Appetizers, main courses, desserts)
- Output queues: 1 (The harder difficulty can have 2 output queues, one for take-out, and one for dine in)