Ctrl Alt Clean GDD

1.0. High Level Concept/ Design

1.1. Concept statement

Fun and disastrous room cleaning simulator.

1.2. Genre(s)

Simulator, Action, Casual, FPP

1.3. Target audience

7+

1.4. Unique Selling Points - Features

Room encounters/events and items randomness.
Storyline unlocks Endless Mode of each level.
Difficulty progression - more tools, encounters and items to clean.

2.0. Product Design

2.1. Player Experience and Game POV

The Player starts in the messy room. Player is informed about the fact that a significant character is coming to visit unannounced and the goal is to clean the room before they come.
Time pressure intensifies when the Player encounters random events during the time he is cleaning the room. (e.g. a window opens up through the gust of wind and everything on the desk below the window is being blown away on the floor). Player is also grabbing items and putting them away.
There are multiple different cleaning tools that the Player can use during his cleaning time.

2.2. Visual and Audio Style

3D low poly with flat textures. Chill music at the beginning, stressful and more intense in the end.

2.3. Game World Fiction

Current times.

2.4. Monetization

Steam purchases.

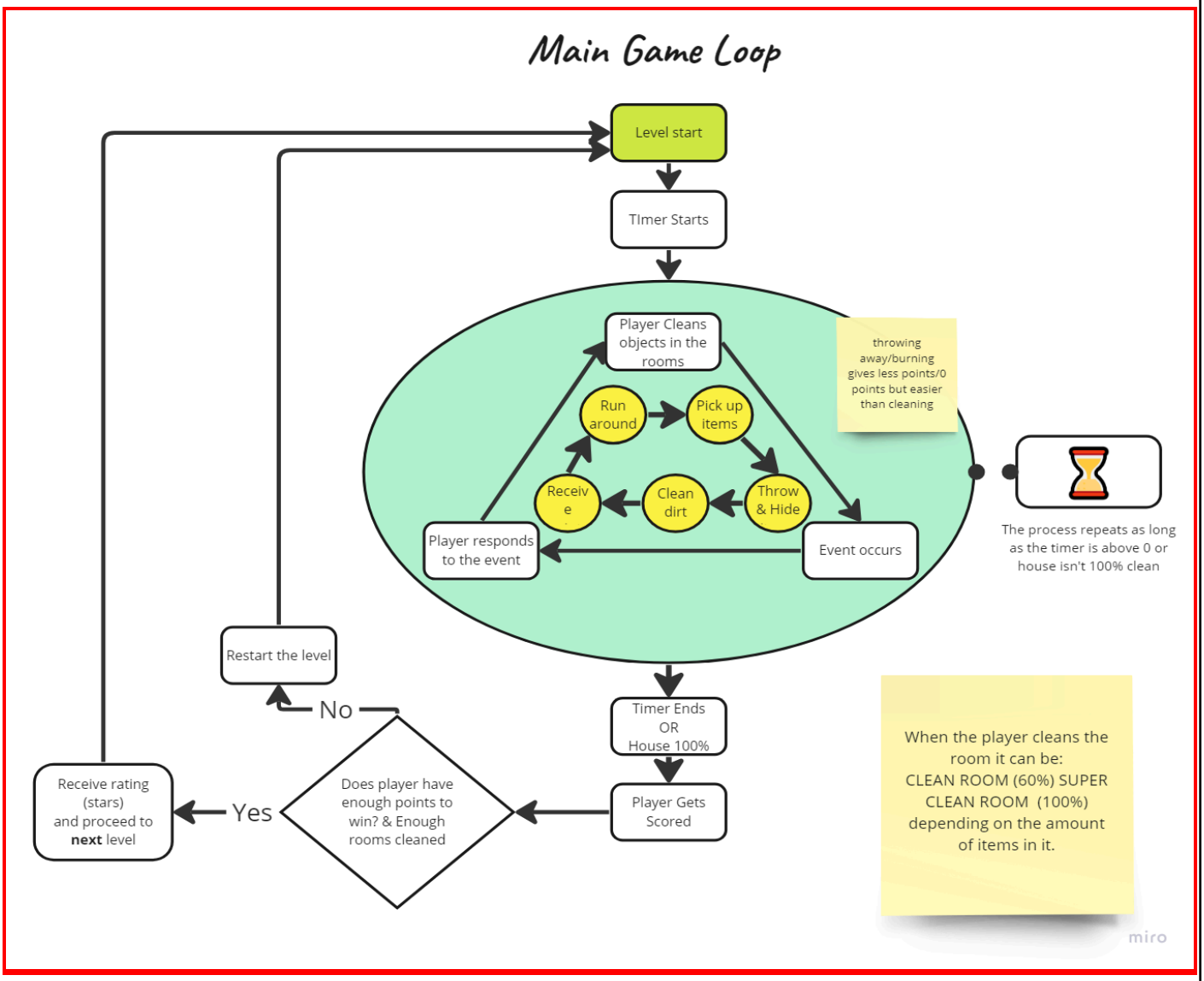
2.5. Platform(s), Technology, and Scope (brief)

Platform: PC, **Engine**: Unity, **Graphics**: 3D
Team: 6 people: (roles and names)
Scope:

- Project will take 5 months to make (May- beginning October)
- Project is divided into the following stages:
    - Greybox - all the must have mechanics implemented in a space of rigid bodies without graphics and lighting
    - Alpha - version of the game with complete level layouts and more mechanics implemented to see how the mood and feel of the game cooperates with simple graphics (biggest part of development)
    - Beta - all graphics included and all the major mechanics included. Little to no development on new features.
- Small game with a few levels that provides approx ~1-2h of playtime on average.
- As an extension of the levels there is an endless mode
- Simple amount of assets and tools used to minimize the amount of work spent on processing assets

3.0. Detailed Game Mechanics & Designs

3.1. Core Loops / Play Flow



3.2. Objectives and Progression

Player controls a character in the first person view. Viewmodel is a hand or both hands carrying, holding and using items. The Gamer cleans the mess in his room.
Short-term goal is to solve a cleaning conundrum: e.g. Do I throw the socks under the bed to hide them, or do I roll them together and put them in a drawer? Former will give you less points but you will save time, which is limited per level and the latter will give you more points, but you will lose more time while being so careful.
In the end a combination of those choices will award the Gamer points, which if there is enough of, will award bronze, silver or gold medal after completing a level. After each level a new mechanic will unlock: be it a tool, event or an asset to clean.

3.2.1 Story Mode / Tutorial

Purpose of story mode is to introduce the player to the mechanics of the game throughout 4 levels

3.2.1.1. Level 1

Tools: Wipe
Events: Earthquake
Assets: MoSCoW
- Exit Doors, Window, Bed, Wardrobe, Nail, Picture, Ball, Basket, Plant, Hoodie, Chair, Desk, Computer, Shelf, Ceiling Lamp, Plate, Banana peel, Desk Lamp, Vase

3.2.1.2. Level 2

Tools: TBD
Events: TBD
Assets: TBD
- TBD
- TBD

3.2.1.3. Level 3

Tools: TBD
Events: TBD
Assets: TBD
- TBD
- TBD

3.2.1.4. Level 4

Tools: TBD
Events: TBD
Assets: TBD
- TBD
- TBD

3.2.2. Endless mode

Endless mode is the second game mode along the story mode and Endless mode is unlocked after finishing the whole story mode (4 levels).
Player has all the rooms unlocked and all the tools at his disposal. In other words, all levels that are prepared for the game are used in the Endless mode.

3.2.2.1 General descriptions

As the name suggests, the endless mode can be played ‘forever’ and does not have limited content. The player can keep playing it as long as they want to and will always have new things to look forward to, in contrast to the story mode, which has a limited amount of levels.

Endless mode consists of a series of rooms to clean in a sequence, which are randomly generated. Upon cleaning one room, the player can move onto next time. The player keeps moving into another room as long as they have time to do so. The moment the time runs out the player loses and their total score is shown.

Main objective of Endless mode is to progress through the levels and earn as many points as possible throughout the process.

The rooms become more and more complex as the player progresses through them. At each consecutive level there are more encounters and things to tidy up.
The variables in difficulty of the room are as follows:

- number of random encounters spawned,
- number of cleanable objects spawned.
- number of sortable objects spawned
- number of trash spawned.

The player is eventually overburdened by things to clean and runs out of time.
The score and time UI is displayed and the leaderboard.

3.2.2.2 Endless mode structure

The endless mode is a series of rooms appearing one after another, when the previous room gets cleaned. Rooms are generated based on different layouts from a general pool of rooms created by Maciej.

The levels consist of cleanable objects on a specific layout. There is trash spawned and sortable objects. There are also random encounters in the room.

Levels are chosen by random encounters and objects are also randomly placed and generated.
After cleaning 60% of the room, the door to another level opens. The player may decide to go through them and start cleaning another room or continue cleaning the current one.
After 80% of completion of cleaning the room the player gets bonus time and extra points. After finishing the room fully, i.e. cleaning the 100% of the room, the player gets even more bonus time and points.

When the player gets 60%, they can leave the room by interacting with the door behind them.
After interacting with the door, an animation plays, they walk through and they cannot go back.

3.2.2.3 Finishing endless mode

When the player is eventually overburdened by things to clean and runs out of time, so the timer hits 0:00, the player loses. Then, the score and time UI is displayed and the leaderboard shown.

Afterwards, the player is sent back to the main menu, where the player can spend the currency earned during the endless run and story mode toward the upgrades to his tools, thus making his runs in endless mode easier and longer.

3.3. Game Mechanics

3.3.1. Movement

Player uses prebound ‘WSAD’ movement to walk around the space.
Player rotates the view using the mouse.
Player is able to crouch using prebound ‘Ctrl’.

3.3.2. Camera

Camera with slight ‘fish-eye’ effect to make the rooms look bigger and stylize the game a little bit.

3.3.3. Tools

- Wipe

Piece of cloth that cleans objects. Cloth gets dirty after collecting dust and requires being washed in the sink or submerged in the water basket to refreshen. Otherwise if the cloth is dirty it has a percentage chance of making the object even more dirty.
Pseudocode

- Vacuum

Vacuum collects trash and items in a cone shaped area. The longer it is used, the harder it is to control it. Its recoil increases through time. It can suck in trash and sortable items, like socks etc.

- Trash Bag

A few trash bags are available at the start of the level and they replenish in new spawned rooms. The User can collect trash into the trash bag and toss the bag out the window. Trash bag fils up to 100% which is indicated by the UI only, no trash bag 3D model is used.

3.3.4. Encounters

- Earthquake

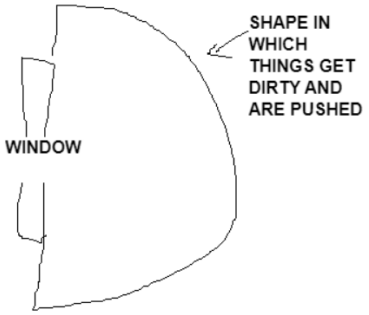
Prerequisite: None.
Camera shakes and paintings drop from the wall as well as other dynamic objects (e.g. bottles socks).
Destructible items, if they fall, they crash on the floor?
Wardrobes open and items from them move, if they fall they fall. Pictures and posters drop from the walls

The player gets points for putting stuff up again on the wall/in correct places/ throwing away broken stuff.

- Wind

Prerequisite: At least one window is fully open.
If at least one window is opened, then the window can occur in the encounters. When the event occurs, all of the windows in the room slam open!

Wind moves dynamic objects that are close to the window. The wind pushes the items away from the window. In addition to that, all objects that are close to the window are made dirty (dusty). Like follows:



The encounter happens every 20 seconds, meaning every 20 seconds all windows slam open, objects get dirty and are pushed away unless the player stops it.
To stop the encounter from re-happening the player has to close all of the windows.

- Smoke

Prerequisite: Closed window.
Smoke comes out of whatever and causes vision impairment and covers the top of the room. You need to crouch to see properly. You move slower during the event. The smoke makes objects at a certain height dirty. If they are cleaned while the smoke is present, then after **5 seconds** they become dirty again.
You have to open the window to clear out the room of smoke.

If you open the window fully it clears out in 15 seconds.
Every additional window you open fully cuts down the time needed to clear out smoke in half.

If you open the window half-way, it clears out in 30 seconds. Every additional window opened half-way cuts down the time needed to clear out smoke in 1/3rd

- Short Circuit

Prerequisite: None.
Short circuit obscures the vision in the room (by making everything dark) in addition to that it forbids the player from using the vacuum.
The player can solve the short circuit by interacting with the light switch/ fuse box. It works like hold to clean mechanic. After That everything goes back to normal.

- Bird Flying across the room

Prerequisite: Open window, fully open.
An animation plays with a bird flying around the room. The bird drops items that it flies through onto the ground (possibly breaking them). If it flies through non-dynamic items, it just makes them really dirty.
The player has no way of counterplaying this encounter, it just happens and takes like 10-15 seconds to complete the flight. After that the encounter stops and the room state is like it was, with the exception of items affected (made dirty and dropped) by the bird.

3.3.5. Special Mechanics / Other mechanics

- Drink Soda/Can

Player can drink soda cans in different colours. Upon interacting with the can, the player gets either a positive or a negative effect. There is 70% chance for a positive and 30% chance for a negative effect.

The colour of the can (different textures) do not change how the can works. All types of cans have the same influence.

Possible outcomes of the soda drinking are as follows:
Positive soda drinking outcomes (70% chance):
- Add 20 seconds to timer
- Movement speed & hold to clean boost for 20 seconds
- Highlight all dirty items for 20 seconds

Negative soda drinking outcomes (30% chance):
- Invert key binds for 30 seconds
- Slow player movement speed and hold to clean for 20 seconds
- Dizzy screen for 30 seconds

3.3.6. Special Objects

~~- Bed (scrapped)~~

~~**Tags:** Static / Interactable / Event~~
- ~~Player can trigger a minigame after clicking on the unmade bed.~~
- ~~Triggers 2D 'stretch the crumbled duvet' mini game.~~

- Wardrobe / Drawers

Tags: Static / Interactable / Event
- Player may put things like clothing and socks into the wardrobe or drawers
- Wardrobe doors are openable
- Drawers are openable
- During an Earthquake event Wardrobe can open and spill its contents.

- Chair

Tags: Dynamic
- Player can move chair
- Player can stand up on the chair to reach higher places

- Socks

Tags: Dynamic / Interactable / Event
- Player can put socks into drawers, hide them under the bed, or toss it to garbage or out the window
- Player can match socks together and put them away as a pair

- Smoke alarm

Tags: Static / Cleanable / Event
- Player can clean the smoke alarm
- Smoke alarm will beep with an annoying sound and request battery replacement
- Player will be able to replace batteries in the smoke alarm

- Ball

Tags: Dynamic / Interactable / Event
- Player can accidentally kick the ball if approached too quickly
- If the ball is kicked it will bound around the room and cause damage. (Leave imprints on the walls, destroy plants, destroy vases)
- Player may carefully approach the ball and move it or hide it somewhere so he does not kick it by accident

- Nails and Pictures / Frames / Posters

Tags: Static (Nail) / Dynamic (Picture), Interactable, Event
- Player can hang pictures up on the nail
- Pictures will fall from the nail during an Earthquake event

- Flower Pots and Plants

Tags: Dynamic / Interactable / Destructible / Event
- Player may water the plant to make it look presentable
- Player may accidentally knock over the plant, which if dropped on the floor may crash and spill soil on the ground.
- Plants on higher shelves have percentage of dropping down during Earthquake event

- Exit Doors

Tags: Structure / Interactable
- Player can Quit the level when clicking and holding the doors.

- Window

Tags: Structure / Interactable / Event
- Player can open the window and throw things out of the window
- Opened window can trigger wind event
- Closed window can trigger wind effect, it slams open.

3.3.7. Regular Objects

- Desk - Static / Wipeable
- Computer Screen - Dynamic / Wipeable / Destructible
- Keyboard - Dynamic / Wipeable / Destructible
- Mouse - Dynamic / Wipeable / Destructible
- PC Unit - Static / Wipeable / Destructible
- Shelf - Static / Wipeable
- Ceiling Lamp - Dynamic / Wipeable
- Plate - Dynamic / Wipeable
- Banana peel - Dynamic / Tossable
- Desk Lamp - Dynamic / Wipeable / Destructible
- Vase - Dynamic / Wipeable / Destructible

3.3.8. Medals

3.3.9. Achievements

3.3.10. Procedural Room Generation with Randomly Placed Objects

The goal is to implement a system in Unity where a room is procedurally generated and populated with various objects (e.g., bottles, trash, books) placed randomly.

Steps:

1. Design model for the room/house environment.
2. **Uniform Random Placement**: Use random number generation to determine positions within defined boundaries (room size). Ensure objects do not overlap and are distributed evenly.
3. **Grid-Based Placement**: Align objects to a grid within the room, allowing for precise control over spacing and alignment. This can create a more organized appearance while still maintaining randomness.
4. Implement algorithms or checks to prevent objects from intersecting with each other or with the room's walls. Techniques such as raycasting or spatial partitioning (e.g., using grids or spatial hashing) can help ensure collision-free placement.
5. **Poisson Disk Sampling**: This technique ensures that points (object positions) are evenly distributed, avoiding clustering and gaps. It's particularly useful for natural-looking distributions.
6. Incorporate randomness not only in object placement but also in object attributes such as size, rotation, and type. This adds diversity and realism to the scene.

Useful links:

https://www.youtube.com/watch?v=PhLcNhK9aro&t=3s&ab_channel=TNTC
https://www.youtube.com/watch?v=7WcmyxyFO7o&ab_channel=SebastianLague
http://devmag.org.za/2009/05/03/poisson-disk-sampling/
https://www.youtube.com/watch?v=3WAI47Zm_ks&list=PLuldIT8dkudoNONqbt8GDmMkoFbXfsv9m&index=1&ab_channel=MattMirrorFish

3.4. Scoring System

3.4.1.General Scoring

Player earns points whilst playing the levels in the game. Player earns points through cleaning, throwing/hiding things as well as other things as described in 3.4.2.. Different actions give different amounts of points as described in 3.4.5.
Scoring the player matters for a few reasons:

- Progressing through the levels in story mode and gaining stars in different levels based on the score
- Progressing through the endless mode and getting high score
- Buying upgrades with special currency gained based on points gained in endless mode

As the player is cleaning the room, the total score of the player is increased. When the timer runs out the player's total is shown.

3.4.2. What do we score

Player gains points when performing cleaning actions in the game. Some things in the game make the player LOSE points.
After finishing an action player gets a certain amount of points. Exact numbers are still to be decided.

Here is the list of actions after which player receives points:

- Throwing a dirty (e.g. dirty plate) or unwanted item (e.g. bottle) into the trash bin.
- Matching same coloured socks together.
- Hiding different items inside wardrobes/drawers etc.
- Finishing to clean an object through the "Hold to clean" mechanic
- Cleaning an object with the use of a tool (e.g. vacuuming a dirty object)
- Fixing bad things that got dirty as a result of an encounter (e.g. paintings falling during the earthquake could be put back on the wall to get some points)

Here is the list of actions after which the player loses points (to be considered more)

- For every dirty patch left by the ball on the walls, after the timer runs out, the player gets points subtracted from the total.
-

3.4.3. Scoring in Endless mode

The players get points in the same ways as described previously until time in one of the rooms runs out and the player loses the run. When the player loses the run, the total amount of points is showcased. The more points the player gets, the higher the player is placed on the leaderboards. Also, at the end of the run, the points gained are turned into upgrade points. Upgrade points are a different currency used for upgrading.

The process of turning normal points into upgrade points can be showcased by the following example (NOTE: specific numbers can be subject to change):

1. The player ends the run after failing to clean the 4th room and the timer runs out. The total amount of points gained throughout all 4 rooms is 1555.
2. every 500 points is turned into 1 upgrade point.
3. Player gets 1555 points turned into 3 upgrade points.
4. Player goes back to the upgrade menu and can (but does not have to) spend upgrade points on upgrades.

3.4.4. Scoring in Story mode

Just like in Endless mode the player receives points whilst cleaning the room. The player keeps cleaning the room until the timer runs out. When the timer runs out the player is assessed based on how many items in the room they cleaned (i.e. how clean the room is).

Scoring in Story mode is only for the purpose of familiarizing the player with the scoring system.

In contrast to the Endless mode, the points in the Story mode are not transferred into upgrade points at the end of the level.

3.4.5. Point distribution for cleaning

Different cleaning actions in the game give different amount of points. The exact numbers of points awarded are not yet settled and require testing but it is important to notice that actions harder to do should give more points than the easier ones (e.g. Longer hold to clean should give more points than a short hold to clean).

Here is a list of actions along with example points for them:

- Throwing an item into a trash bin **50 pts**
- Matching same coloured socks together. **50 pts**
- Hiding different items inside wardrobes/drawers etc. **100 pts**
- Finishing to clean an object through the “Hold to clean” mechanic **(150-450pts** depending on the size of the object)
- Cleaning an object with the use of a tool (e.g. vacuuming a dirty object) **(50-450pts** depending what other action it is used on)
    - Points differ based on what other action use of the tool supports
    - For example, if the tool is used to vacuum something that should be put in the trash bin, it gives the same amount of points like putting things normally in a trash bin = 50 pts
    - For example, if the tools is used instead of a hold to clean mechanic it would give the amount of points that the hold to clean mechanic would give = 150-450
- Fixing problems that occurred during random Encounters (e.g. paintings falling during the earthquake could be put back on the wall to get some points for resolving the Encounter)
- Cleaning the room (**500pts**) and Spick & Span room (**400pts**)

  NOTE: One item can give points to the player only once. For example if you clean a couch and it becomes dirty again through an event, the couch does not give you anymore points. There should be some kind of a marker, boolean, or whatever to check it.

3.5. Cleaning System

3.5.1.Cleaning in General

Cleaning is a core mechanic of our game. This is the mechanic that is strictly related to how you progress through the levels in story mode and how far you are able to go in the endless mode.

As the player proceeds through the level cleaning the room, the room becomes cleaner (duh). When dirty objects get cleaned/removed/hidden, the room is getting cleaned.

3.5.2.How dirty is room

Each room has 3 states of cleanliness:

- Dirty (less than 60% of dirt is removed)

- Clean (at least 60% of dirt is removed)
- Very Clean (at least 80% of dirt is removed)
- Spick & Span (100% of dirt is removed)

Each object in the room has varying amounts of dirt “in it”, or rather how much dirt it contributes to the overall “dirtiness of the room”. Essentially small items that are easier to clean, such as bottles, contribute less to the overall dirtiness of the room than a big dirty couch. So in this example aif you clean the big couch, you remove 10% of the dirt in the room, whereas one bottle is more like 2%. The exact amount of “dirt” in each item is described in point 3.5.5.

When the player cleans or removes the items that are dirty (have the dirt in them), the room becomes cleaner.

NOTE: Items that are dirty, also involve items that are not actually ‘dirty’ but just make the room dirty, such as empty bottles.

3.5.3.Types of objects to clean

In our game there are 3 types of objects to clean.

1) THINGS TO THROW AWAY:
    a) These are the items that have to be put in the trash bin or be hidden from plain sight (e.g. in a wardrobe/drawer etc). These items include:
        i) Empty bottles
        ii) Destroyed items
        iii) Rotten food
        iv) etc.
2) THINGS TO HIDE
    a) These are the items that cannot be thrown away, but rather have to be put in the drawers/wardrobes or out of plain sight (e.g. under the bed, desk etc.). These items include:
        i) Socks
        ii) Books
        iii) Gym equipment (weights)
        iv) etc.
3) THINGS TO ACTUALLY CLEAN
    a) These are items that cannot be moved, but rather have to be cleaned through the use of ‘hold to clean’ mechanic or with a cleaning tool. These items include:
        i) Carpets
        ii) Paintings
        iii) Shelves
        iv) Beds
        v) Sofas
        vi) etc.m

3.5.4.How much dirt is in an object?

As described in 3.5.2, each object has a certain amount of ‘dirt’ attribute in it. The dirt attribute is the measure of how much the item contributes to the overall dirt level of the room.
The items have the following amount of dirt in them:

Throw away/hide items:

- Empty bottle - 2 dirt
- Destroyed items - 3 dirt
- Book - 3 dirt
- Sock - 1 dirt
- Gym equipment - 3 dirt
- Rotten food - 4 dirt

Hold to clean items:

In hold to clean items, the dirtiness depends on the time needed to clean

- 3 seconds (this is the least time it takes to clean sth) - 1 dirt
- 4 seconds - 2 dirt
- 5 & 6 seconds - 3 dirt
- 7 & 8 seconds - 4 dirt
- 10 seconds (this is the most time it takes to clean sth) - 6 dirt

3.5.5.Cleaning - room setup

When the level starts, the system detects how many dirty objects are in the room.
The system sums up all the 'dirt' values of every object in the room and stores the overall 'dirtiness' level of the room, which is equal to the sum of all dirt values of each item in the room.

(This is important so that we can later increase the room cleanliness by a certain percentage of the whole).

Then, the cleanliness level of the room described in 3.5.2. is set at 0%

3.5.6.Cleaning - updating the status

As the player continues the gameplay, things can become dirty again, for example due to events like "smoke". When a certain item becomes dirty, the cleanliness status should be updated. In other words, the cleanliness status of the room should be readjusted to fit the new amount of dirty objects in the room. For example if the player is at 40% and the objects they have already cleaned get dirty again, the cleanliness should be readjusted to (for example) 30%.

Additionally, it may happen that some objects that were not dirty at the start of the level will get dirty, in such case after each encounter that can make things dirty, all dirty items (including those that just got dirty through the encounter) should be again taken into consideration by the system. Their dirt values should be summed up and a new overall dirtiness of the room should be adjusted for that room.

3.6. UI / Screen Flow

Screen Flow Chart, Screen Descriptions, Main Menu Screen, Options Screen, Credits, Game Screen, HUD.

3.7. UI / Screen Flow

Screen Flow Chart, Screen Descriptions, Main Menu Screen, Options Screen, Credits, Game Screen, HUD.

3.8. Saving, Checkpoints, Loading

Game progress and saving game state.

3.9. Easter eggs.

Are there any easter eggs, inside jokes we want to implement?

4.0. Story, Setting and Character

4.1. Story, Setting and Game World

How do game objects and the player's actions form loops? Why is this engaging? How does this support player goals? What emergent results do you expect/hope to see?

In terms of the story, setting, and game world, game objects and player actions form loops by reflecting the progression of the player character's life stages through their surroundings. While the story isn't central to this casual game, the objects and dilemmas encountered in each room still resonate with the player's progression. It is an addition to the gameplay, not the main focus point.

As players clean each room, they encounter objects specific to each stage, such as childhood toys or college textbooks. This loop is engaging because it immerses players in relatable scenarios and nostalgic environments, fostering a deeper connection to the character's journey. It makes the player feel the sense of progression with the player character. It supports player goals by intertwining narrative progression with gameplay mechanics, incentivizing players to explore each stage thoroughly and uncover its story. Also it lets the player have simply more fun when they replay levels, as they uncover new story elements in the game. Emergent results include players feeling emotionally invested in the character's development and forming personal connections to the game world's details.

Story:
Mainly revolves around different stages of the player character's life
Told mainly through simple implicit environmental storytelling or more explicitly in notes

Example level set:
Childhood room -> Bigger, youth room -> Student residence -> Small Flat -> Large house

4.2. Character

How does the player move through the game, literally and figuratively, from tutorial to end? What are their short-term and long-term goals (explicit or implicit)? How do these support the game concept, style, and player-fantasy?

Players move through the game by navigating from one room to another, cleaning messes and unlocking new tools and mechanics along the way.
Their short-term goal is to efficiently clean each room, making decisions about how to manage time and points. Regarding the long-term goals, they aim to progress through the various life stages, experiencing different environments and challenges, such as bigger spaces, more complex room layouts. This supports the game's casual, simulator style by offering a mix of strategic decision-making and immersive, relatable settings, fulfilling the player's fantasy of mastering the art of cleaning amidst life's chaos.

5.0. Levels

3.1. Story, Setting and Game World
How do game objects and the player's actions form loops? Why is this engaging? How does this support player goals? What emergent results do you expect/hope to see?

6.0. Interface and Controls

6.1. HUD

6.1.1 Mess level

Mess level indicates how well we have cleaned the room.

6.2. Controls

7.0. Game Art

7.1. Assets

How do game objects and the player's actions form loops? Why is this engaging? How does this support player goals? What emergent results do you expect/hope to see?

8.0. Pseudocode mechanics

Wiping

```
def IsSuccesful(wipe):
    if wipe's dirtLevel <= 50:
        return True;
     else:
        return True with (100 - (self._dirtLevel / 2))% chance;
        # 100% spoiled wipe will have 50% chance of clearing an object;

def WipeObject(wipe, object):
    if object.isWipeable and object.dirtLevel > 0:
        if IsSuccesful(wipe):
            Add dirtLevel to the wipe based on dirtLevel of the object
            divided by 5 to a maximum of 100;
            Decrease dirtLevel of an object by 30 to a minimum of 0;
            return True;
        else:
            Add dirtLevel to the wipe based on dirtLevel of the object
            divided by 5;
            Increase dirtLevel of an object by 15 to a maximum of a 100;
    return False;
```