Caretaker Chaos:
Game Proposal

Milestone 4

Alexandra Colf, Christian Barbu, Luka Demetrashvili

I. Milestone 1

1. Game Description

1.1. Elevator Pitch

Caretaker Chaos is a fast-paced life simulation game where your mission is simple: keep grandma safe from the chaos that unfolds around her. With fires to extinguish, slippery floors to clean, and all sorts of unexpected hazards, you'll need quick reflexes to protect her from every danger that occurs.

1.2. General Idea

After finding out that their grandma has died, the player uses a device to go back in time and save them. The game is going to be fast-paced, and the player has to keep the grandparent alive for the whole duration of multiple levels. Players interact with the environment through subtle interventions while managing grandma's physical and mental state parameters, creating a unique blend of strategy and puzzle-solving that challenges both reflexes and planning. Grandma is going to be an AI that goes on with her day-to-day life, cooking, going to the bathroom etc, while the player has to make sure she manages to achieve these tasks successfully.

Main Character (Grandma Betty)

1.3. Story Overview

The game starts with the player receiving some devastating news: their grandmother has passed away in her home. While searching her apartment for clues about what happened, the player finds a time travel device, which they decide to use to prevent Grandma from dying. As the player travels back in time, they learn that they can manipulate the past and alter specific events. Their goal is to eliminate every hazard that could lead to Grandma's death. While trying to prevent a tragedy from happening, the player must ensure that Grandma doesn't notice anything. If she witnesses any of the changes made by the player, her mental state will deteriorate, which could lead to her death. Throughout the game, the player has the ability to investigate the apartment and uncover hidden truths about Grandma.

After helping Grandma survive for three days (or, in the functional minimum, one day), the player returns to the present moment. The player discovers that every change made in the past alters the present in unexpected ways. They discover that the past cannot be changed without creating a paradox. In the end, the player must face the heartbreaking truth that they cannot change the past and save Grandma and must let her go. If we have enough time, we plan to include a final level where the player has the chance to talk to their grandma one last time, say goodbye to her, and uncover the whole story behind her life and death.

1.4. Game Mechanics

The game features different mechanics for the present and the past.

Past Mechanics
In the past, the player's goal is to keep the Grandma AI safe and healthy by preventing accidents before they happen. Hazards will randomly appear in the house and the player must resolve them without being noticed by Grandma.

Examples of player tasks:

- **Obstacle Removal:** Remove any items that the Grandma could trip over.
- **Floor Water Cleanup:** Clean up spills before Grandma slips on them.
- **Object Repairer:** Fix broken objects that could harm Grandma and make sure she can successfully complete her routine.

Present Mechanics

In the present, the game focuses on investigation. The player explores Grandma's apartment, trying to determine the hidden truth about Grandma's life and her death and discovering that their actions in the past have changed the present.

At the end of the game, the player will have the opportunity to speak with Grandma one last time and uncover the full truth.

1.5. Relatedness to the theme "Time Travel"

The game mechanics for our game were inspired by the theme. The player is able to switch between the present and the past, using a time machine to alter past events in an attempt to prevent the Grandma from dying and change the future. Every action from the past has direct consequences in the present.

1.6. Tools and Assets

For the technical foundation of our game, we have selected the Unity engine. For the visuals, we opted for a cozy and inviting art style, primarily utilizing the Cozy Interior Asset Pack to create our environment. The main character, the grandma, will be created by us, ensuring a unique design with the necessary animations. We are also planning to potentially introduce another character, the grandfather, in the future, if time permits, which would lead to interesting emergent behavior from the AI in the house.

Initial furniture sketches

1.7. Inspiration and feedback

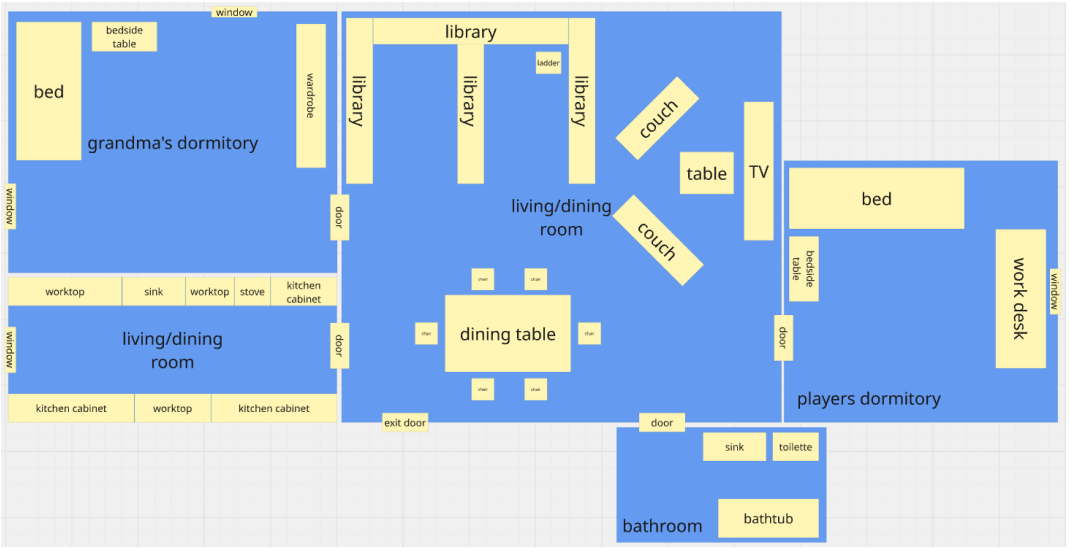
After reviewing the feedback from Milestone 0, we realized that the timeline system we initially proposed had some significant risks. It could become too complex, confusing, and ultimately unenjoyable to play or develop. Additionally, we shifted away from the stealth and puzzle game concept we initially envisioned. We first

decided to drop the stealth aspect, as it didn't fit well with the puzzle focus and would likely make the gameplay frustrating.

We also recognized that, given the team's skills, designing engaging single-player puzzles would be difficult, especially since they would need a strong story to feel enjoyable, which isn't our main strength. Based on the feedback and recommendations, we decided that it would be better to create a faster-paced game with mechanics that are fun to play and easy to understand. This decision felt right, as a lot of feedback emphasized the importance of a simplified timeline, a clear core gameplay loop, and well-defined goals.

As a result, we decided to focus more on implementing strong core mechanics and the AI system rather than a complex storyline. For the gameplay loop, we drew inspiration from fast-paced household management games, where players must complete numerous tasks within a limited time frame. Examples include Nanny Mania, where players clean the house and care for a baby. For the AI, we drew strong inspiration from *The Sims*, known for its interesting and dynamic AI behaviors that create emergent situations during gameplay. The final genre we are aiming for is a strategy and simulation game.

First Sketch of a Top-Down Apartment:



2. Technical Achievement

Our game aims to implement a moderately complex AI system. The desired effect is to achieve AI behavior that is somewhat realistic in certain ways, not fully predictable by players, yet still easy to understand. The AI should easily adapt to any environment in which it's placed, and adding new objects for the AI to interact with should be

straightforward. For this, the final decision was to create a utility AI, which is closely related to behavior-with-activation-level action selection methods commonly used in autonomous robotics. The main inspiration for this is the AI used in the popular game series *The Sims*. The main goal will be to reverse-engineer the Sims AI and reimplement it in our game, despite the limited availability of detailed implementation information on the web.

AI agents in games can be implemented in different ways. Common approaches include finite state machines, behavior trees, and influence maps. However, we found that utility-based AI would be the best fit for our game, as it offers important benefits. Like other methods, it decides the next action based on the agent's internal state. The main loop will be:

1. Create a queue with possible actions.

2. If the queue isn't empty, pop the next one off and perform it.

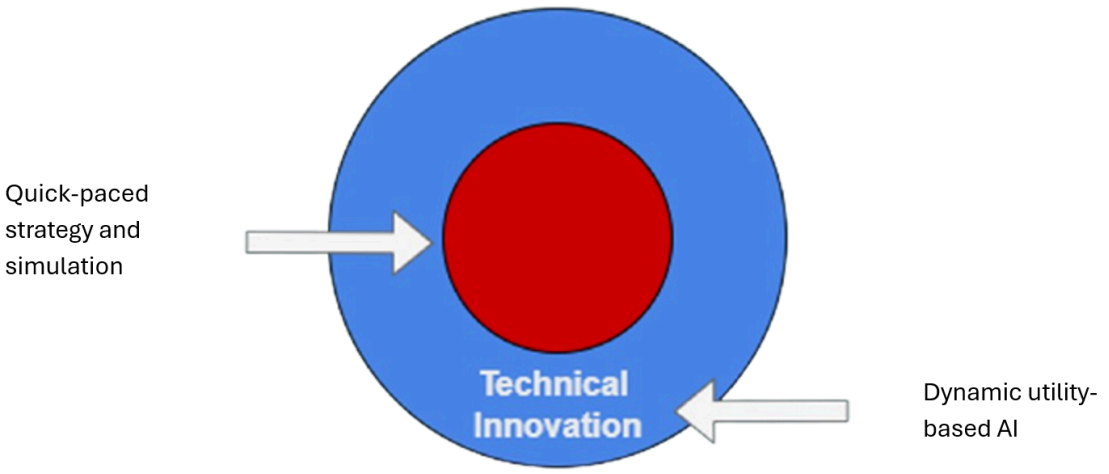
3. If you run out of actions, perform action selection based on the internal state.

Action selection works in the following way: each object in the scene advertises how it affects the agent's internal state. A simple example: if the agent's hunger level is low, it looks around at all objects and decides its next action. It will then see that a fridge advertises the following effect: if the agent interacts with it, it refills hunger (for example, +10 hunger). The agent then creates a table of all possible changes to its internal state and chooses the best one. There are several important aspects to consider. For example, if an agent's fun level is low but their hunger level is also low, it should prioritize filling their hunger first.

Implementing the AI this way allows us to easily extend the level with more possible actions for it to perform. We simply assign data to the new object, describing what it advertises to the agent. Let's say we add a TV, then it would advertise +20 to fun for the agent. The discovery of available actions is decoupled from choosing among them. Each object fully controls what it advertises, so it's easy to enable or disable actions based on the object's state. For example, a working kitchen might advertise "prepare food" by default; however, if the AI agent burns it, the object would stop advertising that value entirely. More information can be found in the following article: http://robert.zubek.net/publications/Needs-based-AI-draft.pdf

Another feature we plan to include is dynamic difficulty adjustment. If the player is doing too well and the agent's sanity and health are high, then the AI will look for more dangerous ways to lower them fast, increasing the difficulty.

3. Big Idea Bullseye



The core idea of our game is a strategy and simulation experience where the player must protect the grandparent from harm. The gameplay focuses on observing her behavior, predicting her next moves, and strategically intervening to keep her safe, while being hidden from her line of sight. The main technical innovation supporting this concept is a dynamic utility-based AI system that drives the grandma's realistic, unpredictable, and adaptive behavior within the level.

4. Development Schedule

4.1. Layered Development Plan:

Functional Minimum

- Basic AI movement for the grandma character
- Grandma movement animations
- Basic scene design
- One death condition for grandma
- One corresponding player solution to prevent Grandma's death

Low Target

- Three possible death conditions for Grandma
- Player solutions for each death scenario

- Display showing grandma’s planned path to help the player strategize
- One fully playable level
- Level timer

Desirable Target

- Two fully designed and playable levels
- Grandma's sanity bar is affected by witnessing player actions
- Display for grandma’s stats (sanity, health)
- Visual display for grandma’s line of sight (seeing ray)
- AI behavior tuning

High Target

- Intro and ending cutscenes
- Main game menu
- Level selection screen
- Event-based audio and music transitions
- Three completed levels with progressive difficulty
- Bug fixing and performance optimization
- Final polish on visuals, UI
- Story integration between levels
- Dialogue or environmental storytelling elements

Extras

- Add grandpa AI
- Add cat/dog AI
- Neighbor characters delivering packages
- Additional levels beyond the initial plan
- New game mechanics
- A present-day interactive level where the player pieces together the events that led to their current situation and can interact with the grandma

4.2. Timeline:

Date	Milestone	Task	Assigned person	Expected Hours
15.10-22.10	Game Idea	Brainstorming	All	10
		Presentation	Alexandra	3
22.10-29.10		Choose Assets	All	1
		Presentation	Christian	3
		Documentation	All	15
29.10-05.11	Refined Game Proposal	Brainstorming	All	20
05.11-12.11		Sketch grandma	Alexandra	2
		Create Prototype	All	3
		Presentation	Luka	3
		UI, Level Design	Alexandra	8
	Prototype	Mechanics	Christian, Luka	16
12.11-19.11		Mechanics	All	30
19.11-26.11		Grandma animations	Alexandra	8
		UI	Luka, Christian	16
		Documentation	All	15
		Presentation	Alexandra	3
26.11-03.12		Grandma animations	Alexandra	8
	Interim Demo	Bug fixing, final touches	All	20
03.12-10.12		Mechanics	All	24
10.12-17.12		Mechanics	All	24
17.12-24.12		UI	All	24
24.12-07.01		Story Integration	Christian	8
		Presentation	Christian	3
		Documentation	All	15
		Mechanics	Alexandra	8
		Add Audio	Luka	8
07.01-14.01	Alpha Release	Bug fixing, final touches	All	24
14.01-21.01		Presentation	Luka	3
		Documentation	All	15
21.01-28.01	Playtesting	Bug fixing, final touches		24
		Bug fixing	All	24
		Trailer Video	All	10
		Documentation	All	15
28.01-04.02	Final release	Presentation	All	3

5. Assessment

The game is planned to be a casual, quick-play experience where players can easily start playing and get into it right away. The main focus will be on gameplay, and using the AI agent as the grandma will provide a different experience each time, as her

behavior will dynamically adjust based on the player's actions. Since the player's tasks are simple and straightforward, the game should be suitable for casual players without much prior gaming experience. Navigation and interaction will be kept simple, mainly limited to mouse movement.

The game aims to entertain all age groups. Although we plan to include a tutorial, it is recommended that the player is familiar with basic pick-and-place and drag-and-drop mechanics. The game simulates a cozy household with an elderly person living in it. Due to medical issues, she has difficulty taking care of herself, and it is up to the player to ensure her safety. The game aims to highlight the experiences of many young people who care for the elderly, whether in the healthcare system or at home for family members. Despite the emotional weight of the topic, the game is meant to feel lighthearted and to show that caring for another person doesn't have to feel like a burden.

To evaluate whether we succeeded in creating this experience, we will assess the game based on several criteria:

1. Are the game mechanics fun?
2. Does the player stay engaged during the level?
3. Does grandma's behavior feel natural and alive?
4. Does her behavior avoid feeling too random or unfair?
5. Does the AI avoid feeling too simple or "stupid"?
6. Are the game's visuals and audio appealing?
7. Does the game have good replayability?

II. Milestone 2

1. Game Rules

Introduction

For the sake of simplicity, we decided to realize a turn-based approach. The core mechanics for the game remain the same as for the real game, however it is important to mention that the real game is not going to be turn-based. To stay as close to the original idea as possible, we have chosen the following game sketch and rules for the prototype:

We use a room layout, with which the player and grandma can interact. Movement can be done vertically, horizontally and diagonally.

In the board game version of our game, one player takes the role of Grandma, who is trying to get herself killed, while the other player takes the role of the Caretaker, whose goal is to keep Grandma safe by removing hazards from her path. The game is turn-based.

Grandma's turn
Grandma first chooses a task to fulfill (Sleep, Eat, Go to the Bathroom, Shower, or Watch TV). She then rolls three dice and moves that many tiles. If Grandma reaches a hazard (like a banana or a puddle), she loses Health because she slips. If Grandma notices an object disappear or move, she loses Sanity.

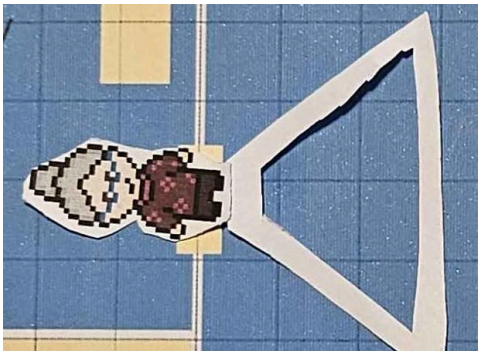
Caretaker's Turn
The Caretaker rolls two dice. Based on the result, they can:

- Remove a puddle if the roll is greater than six
- Move a banana peel up to the number of tiles equal to the dice total

Winning Conditions

- Grandma wins if either her Sanity or Health bar reaches zero
- The Caretaker wins if they manage to keep Grandma alive for twelve turns

Grandma's line of sight will be shown as a trapezoid. Objects inside the trapezoid are seen by Grandma.



Hazards are always **spawned** in a way so that she will not directly encounter them in the next turn. This way, the player will have the chance to move the object to a different location, without being seen.

Turn-based Rules

We will be using different dice for Grandma's turns and for the player's turn. The player will use 2x6 and the grandma will use 3x6.

We have decided on the following objects to "spawn"

- the banana behind the grandma, when grandma rolls a 7
- a new water spill randomly on the grid, when grandma rolls a 10

Grandma has both health and sanity. For simplicity we keep the max value to three.

- Sanity decreases by 1 if the hazard disappears, after she saw it once (but does not actively see it moving)
- Sanity decreases by 2 if the hazard disappears, after she sees it actively moving

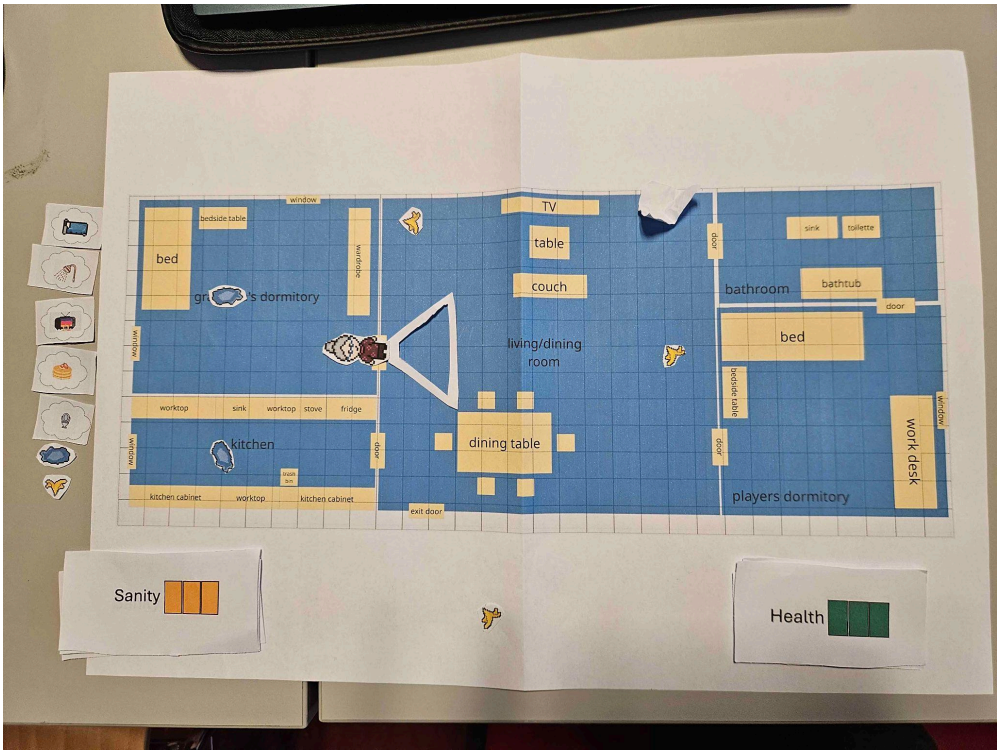
Additionally, grandma can not see actions done by the player, while she is doing an action, so the player may be incentivised to move things if grandma is currently doing something.

When grandma sees a banana or a water puddle, she will add it to her task list to clean that area after finishing her current task. If the hazard is still there when she reaches it, she will lose one health because she will slip on it. Both bananas and water puddles are always spawned out of her line of sight so that the player has time to clean it, without her noticing it.

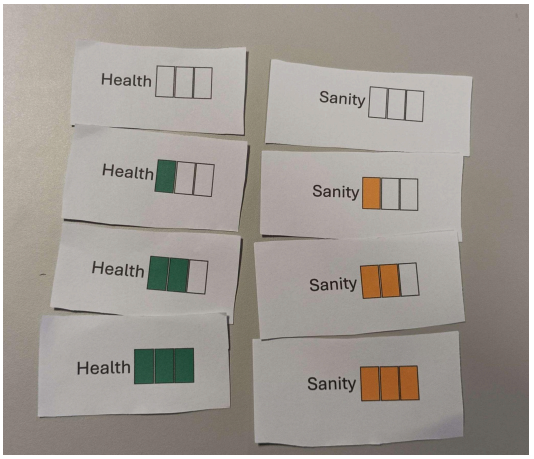
2. Goal of the prototype

The goal of the prototype is to evaluate how entertaining the game is, test the core mechanics we plan to include, identify potential new mechanics, and try to balance the game.

3. Game Board



Board Game Layout



Health and Sanity Bars

Materials

- Game Board (Grandma's house)
- Sanity and Health bars (each with 0-3 bars)
- Grandma character piece
- Grandma's task markers (Sleep, Eat, Go to the Bathroom, Watch TV, Shower)
- Hazard tokens (Trash, Water Puddles)
- Trash can piece
- 3 Dice

4. Evaluation and Takeaways

Creating a physical prototype was challenging, as our game is a fast-paced, real-time experience. The main difficulty was making the prototype enjoyable while maintaining the game's feel. We decided to use a turn-based approach, as it allowed us to better measure and balance the grandma's action and movement speed against the player's possible reactions to save her. At first, it was difficult to make the prototype fun, since the game is asymmetrical, but we eventually came up with more interesting ideas that helped make it engaging. Many of these ideas focused on ensuring that the player's experience felt fair, which will be an important aspect to carry into the final game.

One of our ideas was to make the grandma leave banana peels on the floor, and when she went to clean them up, she would slip. However, how this would work in the game was unclear, so we tested it with the prototype. The results showed that if the grandma immediately went to pick up the banana peel as soon as she saw it, it felt too sudden and unfair, as the player had no time to react. We decided that when the grandma is on her way to her next action, spotting a banana peel will add it to her action queue. She will then go to clean it only after finishing her current action, giving the player enough time to remove it and preventing a direct health loss for the grandma. Instead, if the banana peel is gone by the time she gets there, it will slightly reduce her sanity, since the object she noticed earlier is no longer there.

This change ensures that small player mistakes, such as not cleaning up before Grandma notices something, don't lead to frustrating consequences, like losing health. The banana peels should also spawn behind her, and additional hazards like trash will make the gameplay more dynamic. To support this, we plan to add a pet AI that will occasionally leave trash around the house.

Balancing how far the grandma can see was another key takeaway from the prototype, as it helped us decide on her vision range. Additionally, if the grandma enters a room to perform an action that contains a hazard, the player should have a chance to clean it up before she notices. Otherwise, she would lose too much sanity, since the player would be forced to remove the hazard right in front of her or risk her taking damage after finishing her current task. As a result, we decided that while the grandma is focused on completing an action, she won't notice if the player deals with hazards nearby in the meantime.

Hopefully, these design choices will lead to a smoother and more enjoyable player experience in the final game, just as they did during the prototype testing.

III. Milestone 3: Interim Demo

1. Overview and progress

Our goal for this milestone was to create a playable demo that showcases the basic functionalities of our game. Therefore, we focused on implementing the core mechanics first and left visual and gameplay polishing for later.

Currently, we have completed everything planned for the functional minimum and low target and we have achieved half of the desirable target. We finished the level design for the first level, designed the basic animations for the main character and implemented the core AI mechanics and essential gameplay features. This way, we can test the game effectively and find out what needs to be added or refined.

So far, we are on schedule and feel confident in our progress. We plan to continue working at the same pace for the next milestone. Below, you can see our updated to-do list:

- **Functional Minimum — 100% Complete**
  - ✓ Basic AI movement for the grandma character
  - ✓ Grandma movement animations
  - ✓ Basic scene design
  - ✓ One death condition for grandma
  - ✓ One corresponding player solution to prevent grandma's death
- **Low Target — 100% Complete**
  - ✓ Three possible death conditions for grandma
  - ✓ Player solutions for each death scenario
  - ✓ Display showing grandma's planned path
  - ✓ One fully playable level
  - ✓ Level timer
- **Desirable Target — 60% Complete**
  - ✗ Two fully designed and playable levels *(not done)*
  - ✓ Grandma's sanity bar reacts to player actions
  - ✓ Display for grandma's stats (sanity, health)
  - ✓ Visual display for grandma's line of sight
  - ✗ AI behavior tuning *(not done)*

Gameplay Loop

At this stage, our game begins with a main menu where the player can start the game by pressing the ‘Start’ button. Once the game begins, the player is introduced to the main character’s house (Grandma). As Grandma moves around the environment, various hazards are spawned. These hazards gradually decrease Grandma’s health if the player doesn’t fix them in time.

The player’s objective is to keep Grandma alive until the timer runs out (currently set to three minutes). However, the player must be careful. If Grandma notices the player’s interactions, she will lose sanity, as she cannot understand why objects in her environment are changing.

The current UI displays Grandma’s remaining health and sanity, the active timer, and a task queue showing the actions Grandma plans to take (as shown in the image below). When the player wins or loses, a result screen appears, allowing them to restart the game or return to the main menu.

During the game, the player also has the option to pause the game at any time. The pause menu offers options to adjust audio settings, resume the game, restart the level, or return to the main menu.

UI Elements

Pause Screen

2. Key Implementations

Grandma AI Behavior

The game currently uses the AI system described in Milestone 1's technical achievement chapter. This approach proved to be the right choice, as it made it easy to simulate Grandma's simple and realistic daily decision-making based on her available actions. Once an action is selected, Grandma starts moving toward the target location.

The movement, however, turned out to be more challenging than expected. We decided to use Unity's NavMesh system, but instead of the standard 3D NavMesh, we used NavMeshPlus to generate a 2D NavMesh on the XY plane. This only required rotating the navigation surface upward, after which tilemaps, sprites, and 2D colliders were handled automatically. This already allowed the agent to move to any point efficiently while avoiding obstacles.

But we needed Grandma's movement to be strictly 4-directional. In 2D games, this is typically achieved with a grid-based A* system; however, we wanted to keep the benefits of the NavMesh, such as dynamic obstacle avoidance and flexible walkable areas. Designing levels around a strict grid was not ideal either, especially since our assets are not uniformly sized and hazards can appear or move freely at runtime.

To solve this, we implemented a 4-direction movement with the use of Unity's NavMesh to calculate the agent's path:

- The NavMeshAgent is used only for pathfinding, not direct movement.
- The generated NavMesh path is processed by extracting its corner points.
- Each segment between corners is oversampled and converted into Manhattan-style micro-segments, ensuring every step is purely horizontal or vertical.
- If a target point ends up off the NavMesh, we snap it back onto the surface of the NavMesh. Using the surface normal, we push it slightly away from the walls to avoid repeated invalid positions.
- The same process is applied to the agent if it ever falls off the NavMesh: it is warped back to a valid NavMesh surface, and the path is recalculated.

Path in red

Player Camera Movement

The player can move the camera using the WASD keys or by moving the mouse toward the screen edges. The mouse scroll wheel is used to zoom in or out.

Hazards and Solutions

In our current implementation, we have identified three hazards that could lead to Grandma's death. From each of the hazards, she can lose one health point from three

in total. For all three hazards, the player has the option to remove them, which can prevent her death. The three possible hazards are:

- Bananas, which she drops after leaving the fridge or the couch
- Waterspills which emerge after she went to the toilet or after she took a bath in the bathtub
- Fire, which arises after she has cooked in the kitchen

For each of the actions, the player can do the following to prevent her from taking damage:

- Moving the banana to the trash bin, but at the same time avoiding grandma from seeing it being moved

Both the banana and the fire extinguisher have a spring attached to them, so that when pulled too fast or hard, the spring breaks and needs to be pulled again. When spring breaks, a small particle system follows the direction of the break, illustrating that it needs to be pulled again. The parameters of drag and the force at which the spring breaks are parameters that we can later use to balance our game at later stages.

- Cleaning the floor by rapidly moving the mouse over it, also without her seeing it directly

To illustrate the cleaning process, we are using up to four self-made sprites, which depict different stages of cleaning. Additionally, we have added a particle system with self-made soap bubbles to show that the current movement has an effect.

- Using the fire extinguisher and moving it with the spring in an appropriate range near the fire to extinguish it

Design

Since we wanted to get an overall impression of the game's atmosphere and were using online assets for the sprites, we decided to implement the final scene, as shown in the image below. Additionally, we designed the main character (the Grandma) ourselves and have already implemented both the idle and movement animations.

Grandma's House

Grandma's Field Of View

To allow Grandma to detect hazards and the player's actions, we implemented a field of view system, represented by a yellow cone. The field of view defines exactly which objects Grandma can perceive, determining when she reacts to hazards or becomes suspicious of the player. Because we didn't want Grandma to see through walls, we made the field of view rays stop as soon as they hit an obstacle. Since displaying it constantly would clutter the screen and offer little value to the player, we decided to display it only while the player is interacting with objects in the environment. Balancing the range of this field of view has been challenging, as it directly affects the game's difficulty.

Grandma's field of view

Evaluation and Takeaways

We successfully reached the main goals for this milestone and will continue working according to our planned schedule. With the core mechanics now implemented, we can start expanding the game with additional features and more variety in gameplay.

Our biggest technical challenge so far has been implementing the AI movement with strict four-directional motion. Ensuring that Grandma moves only horizontally or vertically while still using the NavMesh system proved difficult, as the agent frequently got stuck or generated invalid paths. After several iterations, we achieved a stable solution, but this area will require additional tuning as development continues.

Moving forward, our next major steps include adding more mechanics for the player, refining and polishing the AI behaviour, and integrating some story elements to the gameplay.

IV. Milestone 4: Alpha Release

1. Overview and progress

Our goal for this milestone was to deliver a working game prototype that can be shared with playtesters in the next milestone. Currently, we have completed everything planned for the desirable target and achieved approximately half of the high target and half of the extra target.

Since we planned a feature freeze at the end of this milestone, we prioritized implementing all core features over polishing. We have successfully implemented the core gameplay mechanics, a tutorial, an introduction, and the game's narrative elements. While we have already conducted some bug testing, we have decided to leave most of the bug fixing and polishing until the next milestone, when we can incorporate feedback from playtesters.

Below, you can see our updated to-do list:

- **Functional Minimum — 100% Complete**
  - ✓ Basic AI movement for the grandma character
  - ✓ Grandma movement animations
  - ✓ Basic scene design
  - ✓ One death condition for grandma
  - ✓ One corresponding player solution to prevent Grandma's death
- **Low Target — 100% Complete**
  - ✓ Three possible death conditions for Grandma
  - ✓ Player solutions for each death scenario
  - ✓ Display showing grandma's planned path
  - ✓ One fully playable level
  - ✓ Level timer
- **Desirable Target — 100% Complete**
  - ✓ Two fully designed and playable levels
  - ✓ Grandma's sanity bar reacts to player actions
  - ✓ Display for grandma's stats (sanity, health)
  - ✓ Visual display for grandma's line of sight
  - ✓ AI behavior tuning
- **High Target — 60% Complete**
  - ✗ Intro and ending cutscenes (only the intro scene done)
  - ✓ Main game menu
  - ✓ Level selection screen
  - ✗ Event-based audio and music transitions

✓ Three completed levels with progressive difficulty
✗ Bug fixing and performance optimization
✗ Final polish on visuals, UI
✓ Story integration between levels
✓ Dialogue or environmental storytelling elements

- **Extras — 50% Complete**
  ✗ Add grandpa AI
  ✓ Add cat/dog AI
  ✗ Neighbor characters delivering packages
  ✗ Additional levels beyond the initial plan
  ✓ New game mechanics
  ✓ A present-day interactive level where the player pieces together the events that led to their current situation and can interact with the grandma

Gameplay Loop

At this stage, our game begins with a main menu where the player can start the game by pressing the ‘Start’ button or access the available tutorials. After starting the game, the player is presented with an introductory scene in which they knock on a door but receive no response. After they open the door, the player enters Grandma’s house for the first time and discovers her lying dead on the floor, holding a note.

The note provides the first hint toward locating the time travel machine and, when interacted with, takes the player to the level selection screen. From there, the player can choose between the three designed levels. Each level contains a new note that gradually reveals more of the game’s story. At the end of each level, the player is shown either a "Game Over" or a "Game Won" screen, depending on their performance.

Whenever the player encounters a new mechanic, an on-screen explanation is displayed to guide them through it.

2. Key Implementations

Introduction

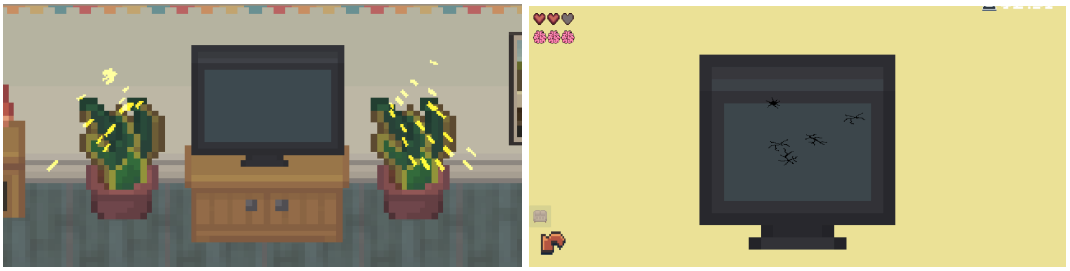
We have restructured the game by adding a new introductory scene. The player is presented with a full-screen view of a door. No matter how many times they knock, nothing happens. To proceed, the player must unlock the door by clicking on the doorknob and then enter the house.

Inside, the player reaches the main scene, which depicts Grandma's house. This scene can be revisited at any time. All the notes collected throughout the levels are displayed in this room. In this scene, the player also discovers Grandma lying dead on the floor, holding a note that provides a hint on how to locate the time machine.

The time machine functions as the level selection screen, allowing the player to choose which level to play in their attempt to save Grandma. Each completed level spawns a new note in the main scene, gradually revealing more of the game's story.

Hazards and Solutions

a) **TV Breakdown -** After Grandma uses the TV and leaves it on, it breaks after 10 seconds, causing sparks to appear. If the player does not repair the TV in time, Grandma will attempt to use it again and take damage. The player can prevent this by interacting with the TV. Clicking on it opens a full-screen canvas displaying the cracks on the screen. To repair the TV, the player must click on and fix all visible cracks.



Broken TV

b) **Fire Spread -** After Grandma uses the oven and forgets to turn it off, a fire starts. We have now updated this feature so that the fire also spreads across the kitchen table. Grandma may die while trying to extinguish the fire. As before, the player needs to select the fire extinguisher to extinguish the fire. Unfortunately, due to the fact that some objects in our scene that are indeed ignitable were too big to use for a single ignitable instance, we were urged to rethink the structure of how neighbouring tiles work. For instance, the kitchen furniture includes multiple objects, but those objects are embedded into the TilemapsRenderer, so we were not able to access individual objects. The solution was to spawn multiple ignitable objects into the ignitable area and connect them via overlapping colliders, allowing us to check their neighbours. We utilized the Hexagonal Circle Packing technique to pack the circles into the rectangle of the TilemapRenderer.

c) **Cat AI -** The third and last level introduces more difficulty by adding a Cat agent that the player has to take care of. The cat uses the same AI system as Grandma, but has its own actions and needs.

i) **Pleasure**

The first need is pleasure. When the cat wants to be petted, it will ask the player by showing a hand sign above it. If the player does not do this, the cat will pee on the ground and leave a puddle that the player has to clean.

ii) **Bladder**

The cat also has a bladder and a litter box that it will use. After using the litter box, it becomes dirty and needs to be cleaned by the player. If the cat uses a dirty litter box, it will leave trash around the house that the player has to clean up.

iii) **Hunger**

The cat also has hunger and will eat from its bowl, which needs to be refilled by the player. If the cat goes a long time without eating, it can starve and die. If Grandma sees a dead cat, she will be heartbroken, suffer two health losses, and lose one sanity.

Visual Cues

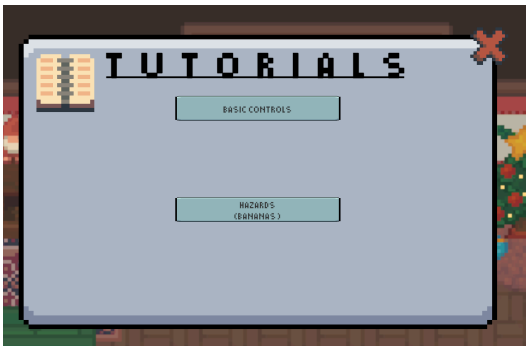
To make it easier for players to understand what is happening, it was essential to add visual cues. For example, when a cat is hungry and waiting in front of its bowl, the player should be notified in a clearer way. Above Grandma's head and the cat's head, indicators will appear.

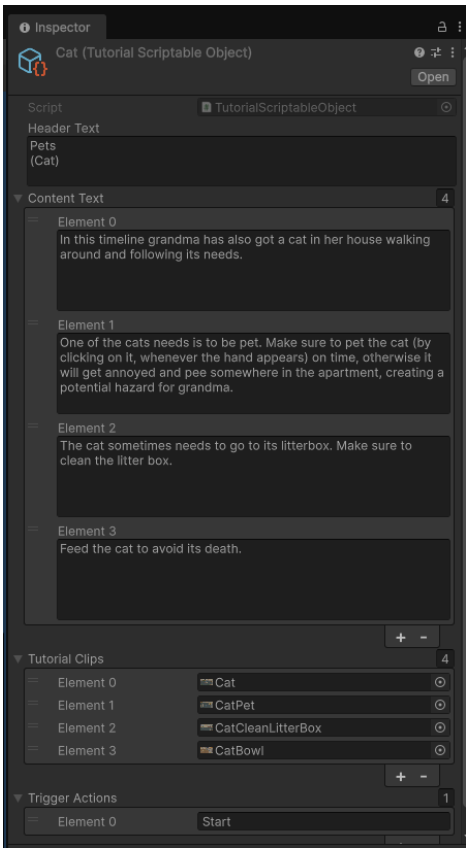
One example is when Grandma goes back to check on a dropped trash can or banana peel, and it has disappeared: she will lose sanity. To visualize this, an exclamation mark appears above her head. When the cat wants to be petted, this is indicated by a hand sign. When the cat uses a dirty litter box and starts leaving trash, an indication that it is stinky appears above its head.

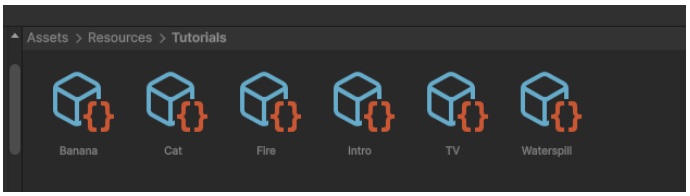
Additionally, when Grandma's sanity gets low, the screen starts to wobble. This indicates that her sanity is low, since having no sanity will start draining her health.

Tutorials

Our goal was to implement a tutorial system that would not overwhelm the player with information. To this end, we created a system that prompts the player the first time they open the game. Additionally, the tutorials are bound to the respective event. When an event occurs, the tutorial starts two seconds later. Tutorials can also be re-read by going back to the main menu. These are added as the player progresses to avoid cognitive overload. For this, we used ScriptableObjects, as they offer scalability.







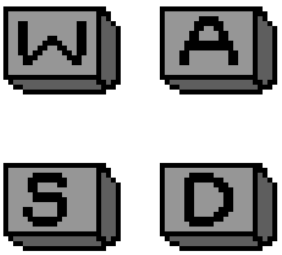
Story Notes

We implemented a note system for the storyline that provides hints about the plot and its connection to the time machine. The player collects the notes in the intro scene, which is set in the present time. We again used ScriptableObjects here to enable us to quickly create notes with minimal effort.

Design Updates
As we added the cat to our game, we also decided to add a litter box sprite, which is not part of the sprite assets we have acquired for this project. So, we decided to create the sprites and animations ourselves. The same accounts for the visuals of the note and the keys for the UI explanation.







3. Evaluation and Takeaways

We successfully achieved the main goals for this milestone and will continue working according to our planned schedule. With all core mechanics now implemented, we are ready to enter the feature freeze phase.

During this milestone, we encountered no major difficulties. Since the core gameplay mechanics were already in place, the most challenging aspect was designing the introductory sequence and effectively integrating the story elements into the game.

Looking ahead, our next major step is to conduct playtesting. This will enable us to gather player feedback and focus on bug fixes and overall game polishing.