

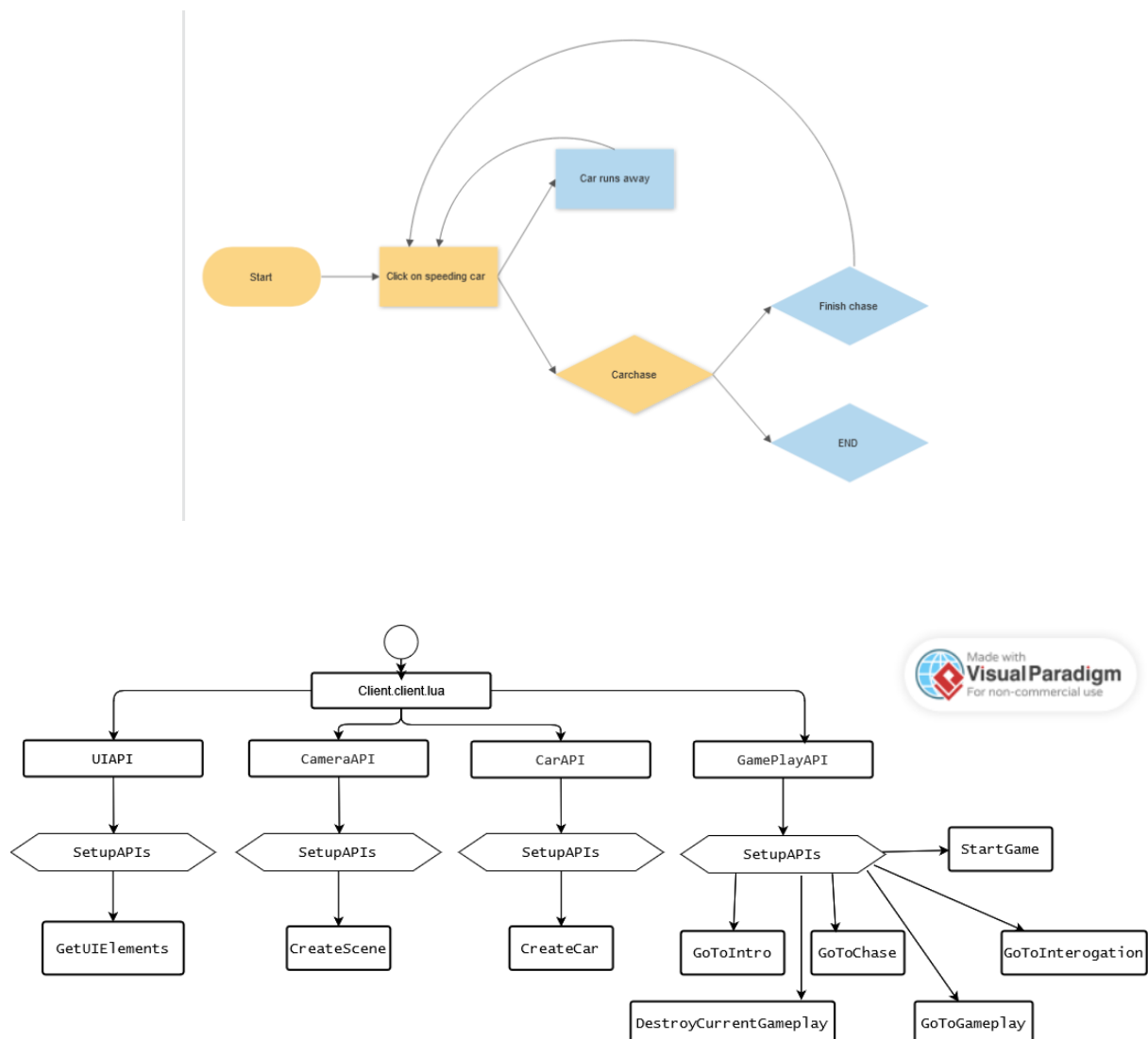
Police-car simulator

Objective:

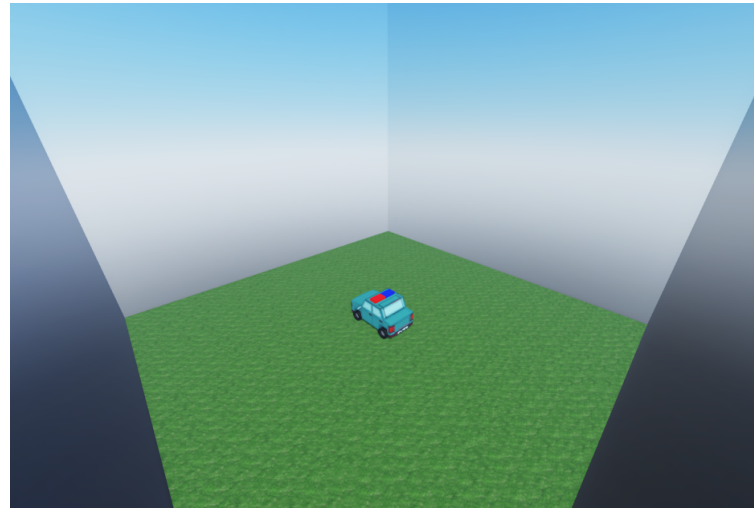
In this police car simulator, your goal is to be a law enforcement officer and maintain order in the city. Hop into your police car and chase down criminals to catch them. Drive skillfully, outmanoeuvre the escaping vehicles, and bring the bad guys to justice. Show off your driving skills and strategic thinking to keep the city safe and peaceful.

WorkFlow:

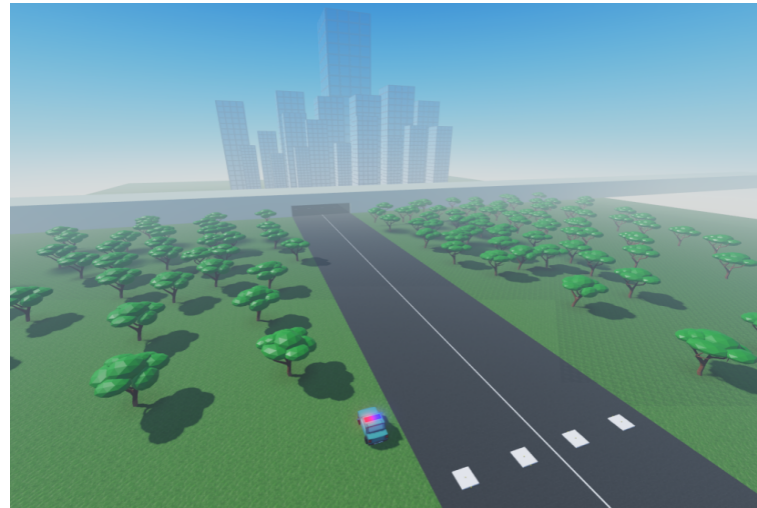
The workflow is represented in the following diagram:



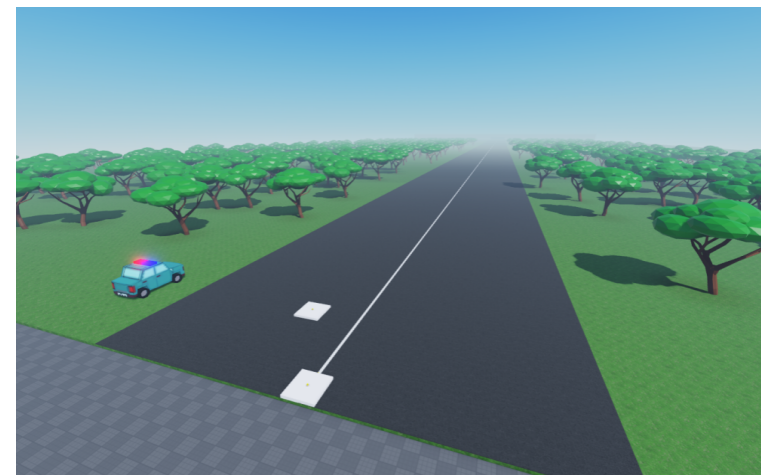
Intro Scene:



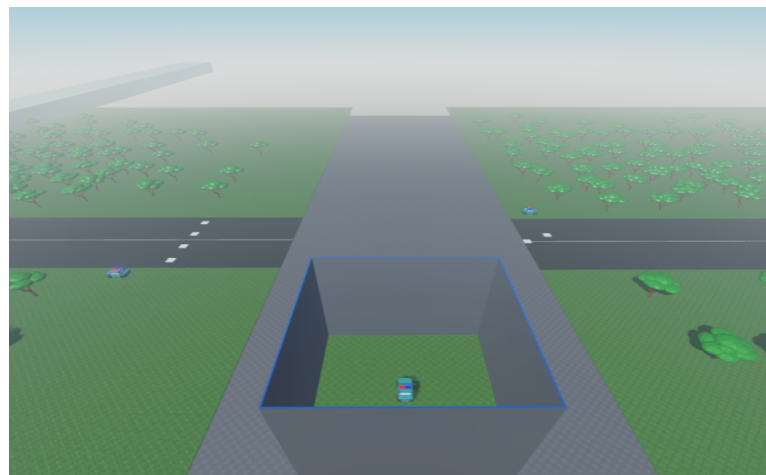
Gameplay Scene:



Chase Scene:



Overview:



User stories:

Personas: Lucas, James, Sophia

Lucas is a 14 years old boy, who plays computer games as a way to spend his free time and to disconnect from reality for a while. He is searching for a game which he could play for short periods of time, which could also help him develop his attention and reaction speed.

James is a 21 years old young adult, who played a lot of games in his younger years which made him want to develop on his own. He is searching for all the games he could analyse and see what he should focus on in his own development process.

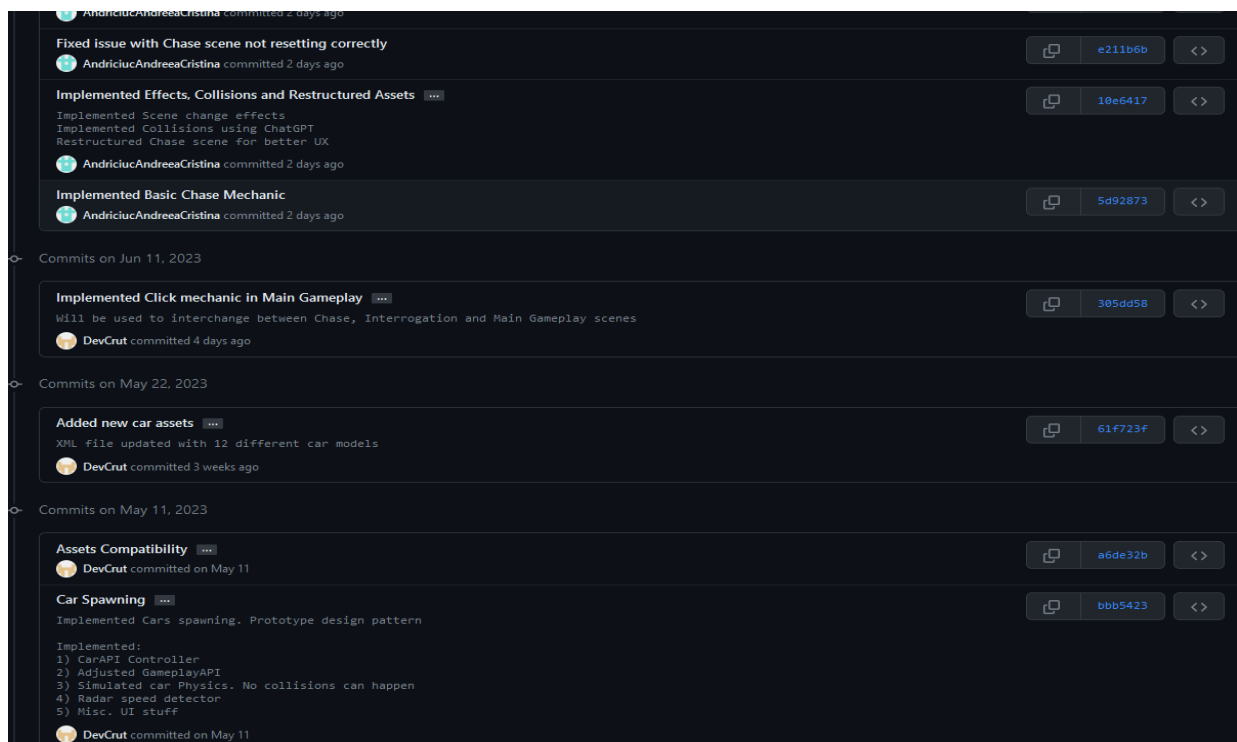
Sophia is an 8 year old teenage girl who wants to play games as a way of being visually engaged since she has a deep passion for art and drawing in general. She is looking for new styles and feels like computer games could inspire her.

"As a [persona], I [want to], [so that]."

- As Lucas, I want the game to not be necessarily challenging, so that I must play it while taking a break from everything else.
- As Lucas, I want my progress/best score to be saved, so that I can keep track of my evolution.
- As Lucas, I want to interact with the game only using the mouse and maybe the space key on the keyboard, so that I can focus on what is happening in the game and not on what keys to press.
- As Lucas, I want to interact as much as possible with police related games, since my father works in this field and I am interested to see if it would be a good fit for me as well.
- As James, I want to be able to repeat a level, even though I have already passed it, so that I could analyse any details.
- As James, I want to be permanently engaged in the story/action of the game, so that I wouldn't get bored.
- As James, I want something with more intricate details, so that I would have more to learn from this experience.
- As James, I want a complex and realistic dialogue system, so that I can feel immersed in the game.
- As Sophia, I want a game with beautiful scenery, so that I can make a painting based on it.
- As Sophia, I want a game with a good colour palette, so that I can learn how to mix colours in my own artwork.

Development Process:

During our development process, we used Git because it provides a structured way to manage versions of the codebase, allowing us to track changes, create branches for new features or experiments, and easily switch between different versions or commits, along with enabling seamless collaboration among team members, helping to prevent conflicts and providing mechanisms for resolving them when they do occur.



Also, during this process, we used ChatGPT to automatise certain tasks, such as:

Can you implement me a collision detection between two rectangles in LUA?

Certainly! Here's an example of a collision detection function in Lua for two rectangles:

```
lua Copy code

function checkCollision(rect1, rect2)
    -- Check for horizontal overlap
    local rect1_left = rect1.x
    local rect1_right = rect1.x + rect1.width
    local rect2_left = rect2.x
    local rect2_right = rect2.x + rect2.width

    if rect1_right < rect2_left or rect1_left > rect2_right then
        return false -- No horizontal overlap
    end

    -- Check for vertical overlap
    local rect1_top = rect1.y
    local rect1_bottom = rect1.y + rect1.height
    local rect2_top = rect2.y
    local rect2_bottom = rect2.y + rect2.height

    if rect1_bottom < rect2_top or rect1_top > rect2_bottom then
        return false -- No vertical overlap
    end

    return true -- Collision detected
end
```

Regenerate response

We took note of [Lua coding standards](#) and followed through:

Formatting: Indenting uses two spaces

**Naming: Variables holding values or objects are typically lowercase and short;
Using locals rather than globals whenever possible.**

We also provided sets up properties and objects within the "Client" table, which initialises a client-side environment and sets up references to different game elements.

...and others.

```
1  local API_NAME = script.Name;
2
3  local Client = _G;
4  local APIs = Client.APIs;
5  local CurrentAPI = APIs[API_NAME];
6  local Storage = {
7      UI_ELEMENTS = {};
8  };
9
10 function SetupAPIs() -- CORE
11     local function GetUIElements()
12         for i,v in pairs(Client.UI:GetDescendants()) do
13             if v.Name == 'UI_MARKER' and v.ClassName == 'StringValue' then
14                 if Storage.UI_ELEMENTS[v.Value] ~= nil then
15                     warn("DUPLICATE UI MARKER ID=(".. v.Value .."). OVERWRITING");
16                     warn("OLD PARENT: ".. Storage.UI_ELEMENTS[v.Value]:GetFullName());
17                     warn("NEW PARENT: ".. v.Parent:GetFullName()); warn();
18                 end
19                 Storage.UI_ELEMENTS[v.Value] = v.Parent;
20                 v:Destroy();
21             end
22         end
23     end
24     CurrentAPI.GetUI = function(UI_TAG)
25         return Storage.UI_ELEMENTS[UI_TAG];
26     end
27     GetUIElements();
28 end
29
30 return function ()
31     SetupAPIs();
32 end
```

```
2  local Client = _G;
3  Client.Player = game.Players.LocalPlayer;
4  Client.Camera = game.Workspace.CurrentCamera;
5  Client.Mouse = Client.Player:GetMouse();
6  Client.Assets = game.ReplicatedStorage:WaitForChild('Assets');
7  Client.UI = Client.Assets:WaitForChild('UI');
8  Client.UI.Parent = Client.Player.PlayerGui;
9  Client.API = {};
10
```

-- Global table <https://create.roblox.com/docs/reference/engine/globals/LuaG>
-- Player Object
-- Camera
-- Mouse
-- Game assets
-- Interface
-- Reparenting, Players.CharacterAutoloads = false, UI won't load by its
-- Container of function overloads

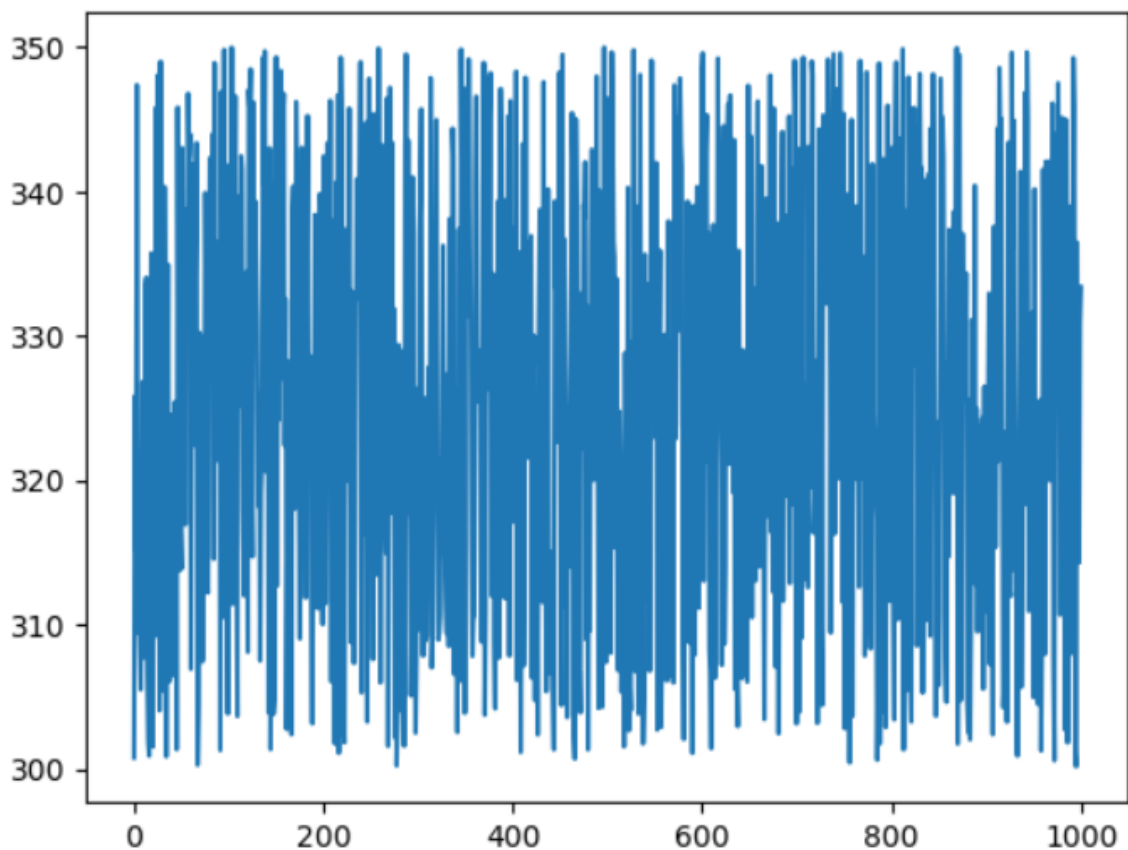
Automated tests:

We checked to see if there were any memory leaks in our game and we came to the conclusion that we have a variation between 300-350 MB of memory occupied when the game is running on 1000 actions.

We created a script in python for plotting the graphic below, using some data we obtained with the following code:

```
local x = {}
for i = 1,1000,1 do
    Client.API.GameplayAPI.GoToGameplay();
    Client.API.GameplayAPI.GoToChase(Client.API.CarApi.CreateCar())
    local mem = game:GetService('Stats').GetTotalMemoryUsageMb()/8
    x[#x + 1] = mem;
end

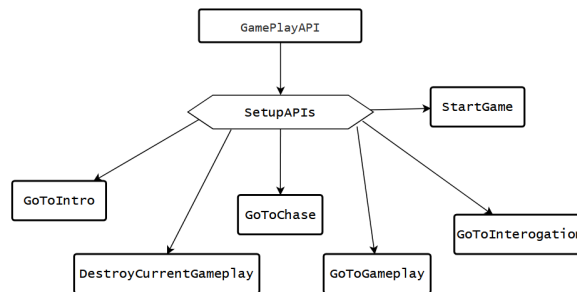
str = "["
for i,v in pairs(x) do
    str = str .. tostring(v)
    if i ~= #x then
        str = str .. ', '
    else
        str = str .. ']'
    end
end
print(str)
```



Design Patterns:

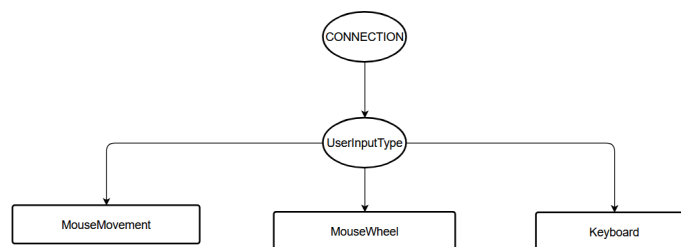
Facade:

Purpose: Provides a simplified interface to a complex subsystem, hiding its complexities and providing a unified entry point.



Observer:

Purpose: Establishes a one-to-many relationship between objects, where changes in the subject are automatically propagated to the observers.



Decorator:

Purpose: Attach additional responsibilities to an object dynamically. Decorators provide a flexible alternative to subclassing for extending functionality.

