vCIS 350 – (Team) Project #5 – **Due Sunday 10/30 before 11:55pm**

Your team's next big goal is to create a playable game prototype that meets the design objective below with Unity and C# that you can test at the final in-class playtesting scheduled for November 21st. It needs to meet the main requirements for what counts as a playable game prototype (listed at the bottom of this document) by when you submit it with Project 6 on November 20th.

As a team, you can choose any of the following as your team's design objective or theme:

Steps to Complete

Continue with your same teams. Pick a design objective from the list above, and work as a team to answer any questions under that design objective.

Deliverable: write which design objective or theme your team chose and your answers to any questions under that design objective or theme above.

Persuasive Game to Reduce CO2 Emissions:

What is a specific behavior change players can make in the real world that will reduce their carbon footprint (CO2 emissions)? Brainstorm a few possibilities and choose one to focus on.

- **Make a negative association for the player with gas guzzling cars causing them to have the desire to reduce their use of such vehicles in any way they see fit.**

Develop a Persuasive Game that will persuade players to make that specific CO2-reducing behavior change in their real lives.

- **The game will have gameplay where you target CO2 emitting cars as the enemy causing a negative association with such cars and try to protect/avoid the good electric cars with your tank.**

What emotions will best motivate players to make this behavior change that will reduce their carbon footprint?

- **Subtle guilt in regards to the CO2 emitting "bad" cars driving around as the player targets them.**
- **Awareness to the problem of CO2 emissions due to the exaggerated/on-the-nose aspects of the game**

Framing is setting the assumed criteria for what counts as good. Complete the following sentence at least 3-5 times: [The action or behavior change] is good because [a reason]. (e.g. Bicycling instead of taking a car is good because it is better exercise.)

- **Reducing CO2 emissions is good because it prevents unwelcomed climate change**
- **Reducing CO2 emissions is good because it makes the air cleaner for all people to breathe**
- **Reducing CO2 emissions is good because high levels of CO2 can have cascading effects ranging from rising water levels to creature habitat destruction**

Intended Experience or Desired Outcomes:
Given your stated design objective or theme, what are the emotions you want players to feel when they play your game? Other than enjoyment and an optimal level of challenge, how do you want players to feel while playing your game?

Get specific about how you want players to feel. If it helps, you can use metaphors, but then ask yourselves how you want that experience to make the player feel. Avoid broad terms or synonyms for enjoyment like "satisfied" or "rewarded". Avoid game mechanics, player actions, feedback, or other ways the game will provide the intended experience.

If your design goal is an outcome other than an emotional experience, what is the desired impact on the thoughts, feelings, knowledge, beliefs, or behavior of players after they play your game?

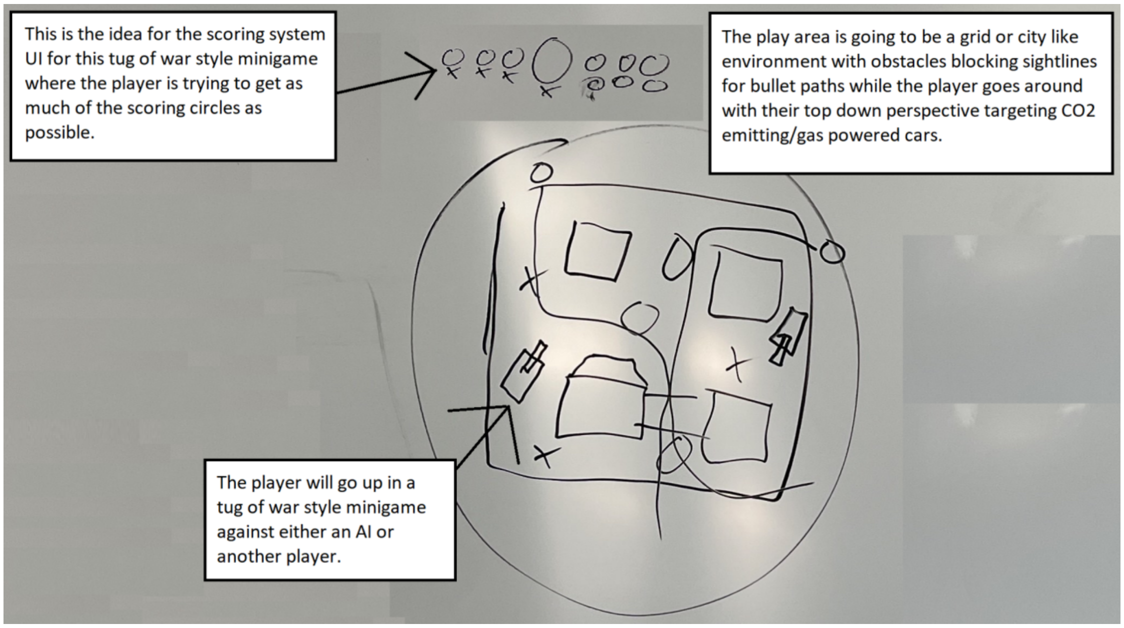
Deliverable: a bullet-point list of about 3-7 specific emotions or feelings other than enjoyment or optimal challenge you want players to feel when they play your game.

If your design goal goes beyond an emotional experience, instead list 3-7 specific player thoughts, feelings, knowledge, beliefs, or behavior other than enjoyment or optimal challenge that you want your game to have an impact on.

- **Desire to reduce CO2 emotions**
- **Frantic stress from competition**
- **Pride from a overcoming opponent**
- **Frustration from getting set back**

1) Brainstorm design ideas with your team. Use sketches to communicate your design ideas. This is a team project, so share your ideas freely among your team.

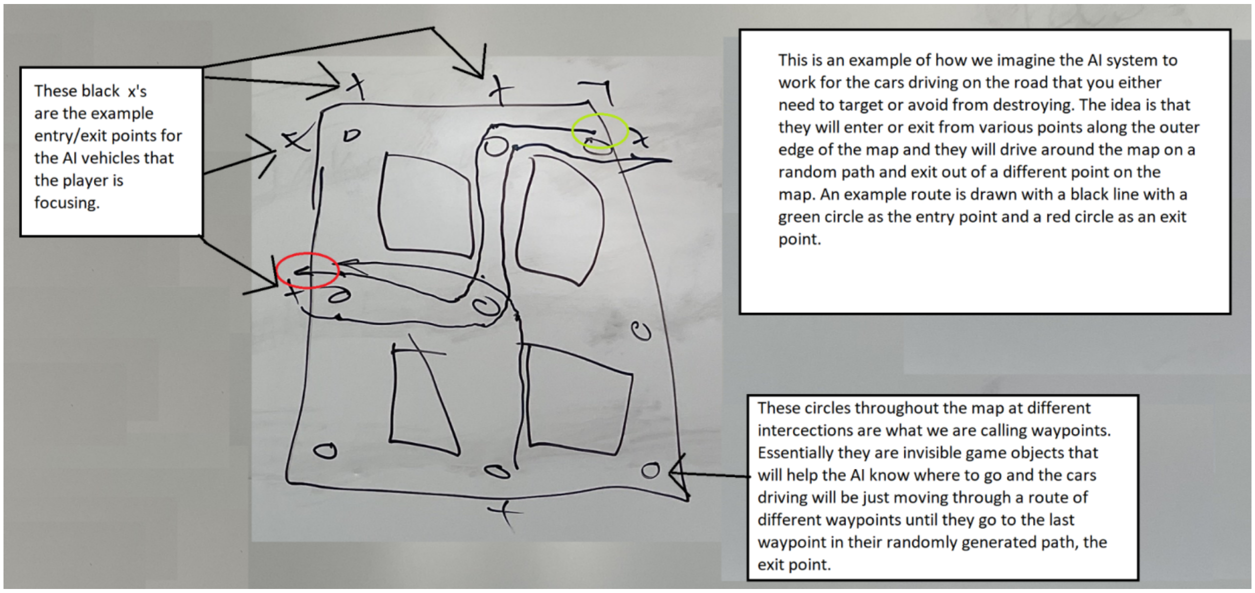
   Deliverable: include at least 3-5 hand-drawn sketches (at least one from each team member) visually communicating a design idea. It does not need to be beautiful fine art – it just needs to visually communicate an idea for the game's design.



This is the idea for the scoring system UI for this tug of war style minigame where the player is trying to get as much of the scoring circles as possible.

The play area is going to be a grid or city like environment with obstacles blocking sightlines for bullet paths while the player goes around with their top down perspective targeting CO2 emitting/gas powered cars.

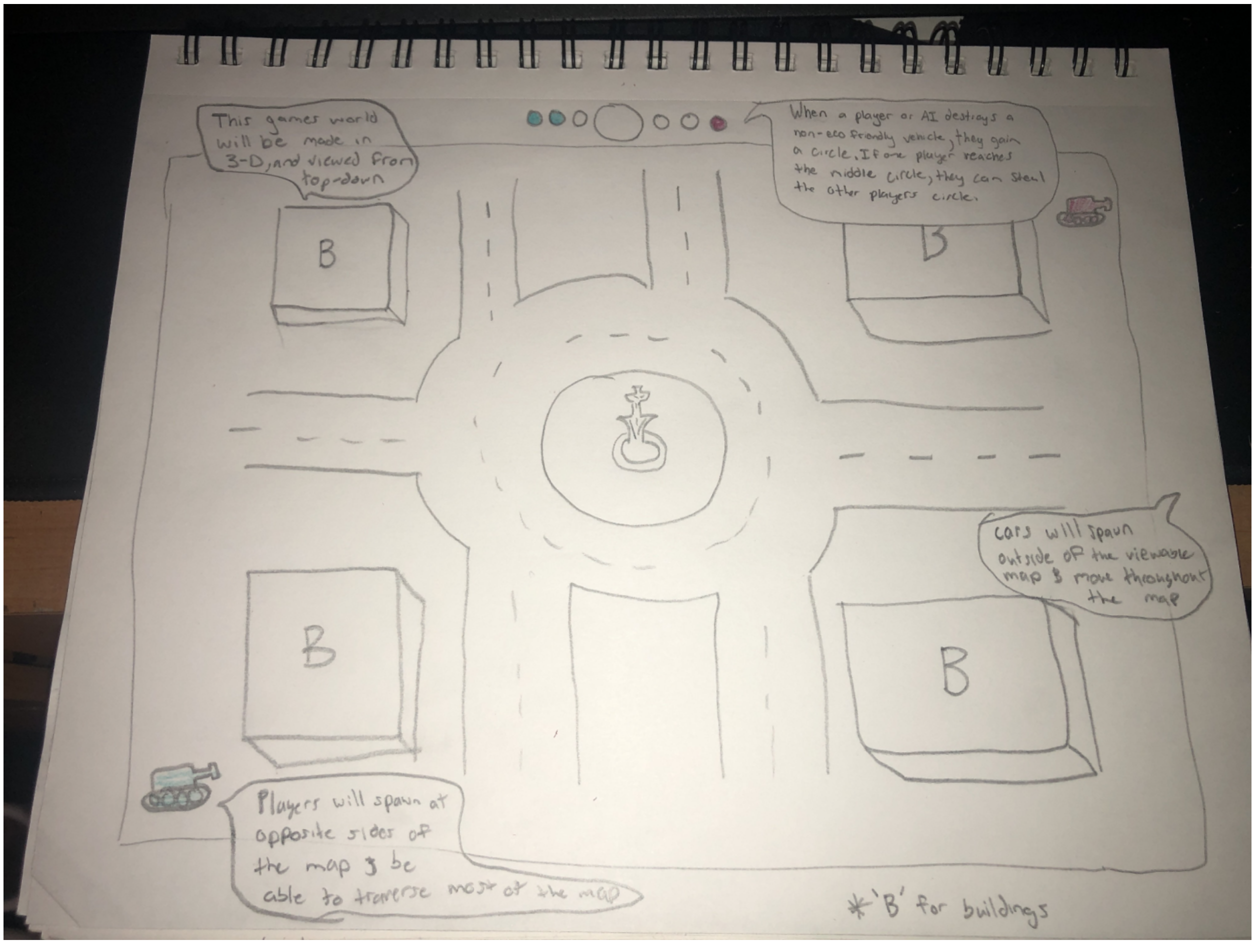
The player will go up in a tug of war style minigame against either an AI or another player.



These black x's are the example entry/exit points for the AI vehicles that the player is focusing.

This is an example of how we imagine the AI system to work for the cars driving on the road that you either need to target or avoid from destroying. The idea is that they will enter or exit from various points along the outer edge of the map and they will drive around the map on a random path and exit out of a different point on the map. An example route is drawn with a black line with a green circle as the entry point and a red circle as an exit point.

These circles throughout the map at different intercections are what we are calling waypoints. Essentially they are invisible game objects that will help the AI know where to go and the cars driving will be just moving through a route of different waypoints until they go to the last waypoint in their randomly generated path, the exit point.


This games world will be made in 3-D, and viewed from top-down
When a player or AI destroys a non-eco friendly vehicle, they gain a circle. If one player reaches the middle circle, they can steal the other players circle.
B
B
B
B
cars will spawn outside of the viewable map & move throughout the map
Players will spawn at opposite sides of the map & be able to traverse most of the map
*'B' for buildings

2) **Brief Game Design Document:** Use the attached template to create a Brief Game Design Document for your team's game. Only one game design document is needed for your whole team, but your whole team needs to participate in generating the design. Work together on it, do not just assign it to one person.

   You can use hand-drawn sketches to make the main illustration for your game design concept sketch, but annotate it with callouts in PowerPoint or Google Slides (see the instructions for how to do this in Assignment 1). Show the main thing players will be doing, what makes it challenging.

   Deliverable: A single brief game design document

Brief Game Design Document

mm/dd/yyyy
Team Name

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

Team Members

List all team member's names here

Game Design Concept

1 Player Control

You control a

Tank

in this

Top Down, 2 player competitive area game

where

WASD and spacebar

makes the player

move and shoot

2 Basic Gameplay

During the game,

Cars appear

from

off the screen.

3 Core Game Mechanic

The goal of the game is to

destroy non eco friendly cars quicker than your opponent

What makes this goal challenging or difficult is

this is challenging because the opponent is also destroying cars and players can attack each other to briefly stun their opponent

Players have the ability to

shoot projectiles out of the front of their tank to destroy cars and attack the enemy player. Players must attack non eco friendly cars and avoid environmentally friendly cars.

And when players use their abilities

particle effects will show projectile trails and explosions.

4 Gameplay Mechanics

As the game progresses,

the slider at the top will move in favor of who is winning

making it

show who is closer to winning. When one player moves the slider all the way to the opposite side they will be victorious.

[*optional*] There will also be

cars that move along random paths around the map.

5
Win / Loss Conditions

The player will win when

They have taken over the “tug of war” meter by destroying gas-powered vehicles faster than the AI/other player

The player will lose when

The AI or other player has taken over the “tug of war” meter

When the player wins

A text message will pop-up letting them know they won with the ability to restart the game. A triumphant song will play.

When the player loses

A text message will pop-up letting the player know that they have lost with the ability to restart the game. A sad song will also play.

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

Clicking a Restart button (likely R) they will also be able to go back to the main menu

6
Sound & Effects

There will be sound effects

Tanks shooting, gas-powered cars being damaged, electric cars being damaged, cars being run into, the tank & cars moving

and particle effects

Tank muzzle flash when firing, smoke from the gas-powered cars, confetti or something similar when gas-powered cars are destroyed

[*optional*] There will also be

description of any other expected special effects or animation in the project and what triggers them.

7
User Interface

The	will	whenever
tug of war meter	*move one forward*	*a non eco friendly car is destroyed.*

At the start of the game, the title

Car-nage will appear

8

Other Features

Game will be able to be played in single player (with and AI enemy) or multiplayer with a second user on the same device. (Player 1 - WASD, Player 2 - Arrow Keys)

3) **Project Plan:** Again, your next big goal is to create a playable game by November 20th. Create an overall plan for what needs to be done before November 20th. Create a list of bullet points in two categories: requirements and wishlist. Sort the requirements into a rough order to be completed and put deadlines on each requirement as a team for when they will be done. If you finish the requirements, you can work on the wishlist. These bullet points are features or parts of the game development to be done.

You can break the requirements down into smaller chunks if that helps, but do **not** assign the work in the overall project plan. The project plan is where you plan as a team what work needs to be done to make the game playable enough to test before November 20th. This is preliminary, so your team can decide to change it as the project evolves, but this is your big-picture plan.

Deliverable: 2 lists of bullet points – requirements (with deadlines for each requirement) and wishlist. The deadlines must show each requirement is planned to be done on or before November 20th.

Requirements:

- Models for the map, cars, tanks, and bullets
- Waypoint system that sends cars in random directions at intersections (always on the right side of the road)
- Spawn randomizer that spawns random cars at random intervals to use the waypoint system
- Tank controls
- Colliders that prevent cars from colliding with each other, only allowing collision with tanks
- AI for enemy tank in single player games
- UI that clearly shows important info (time, score, etc)
- Multiplayer capabilities (local)
- Visually appealing particle effects (explosions, smoke trails, muzzle flash…)
- A game controller that tracks pertinent information, such as the amount of cars on the map to prevent “gridlock”.

Wishlist:

- Multiple maps
- Online multiplayer

4) **Sprint Planning:** Every week in class you will decide as a team how you want to break down, split up and take on the work you will do for that week.

Your team's sprint plan needs to show what each team member is responsible for getting done each week. Aim for small enough chunks of work that they can be done within one week, and a fair distribution of the work. Do not get lost in the details, but be clear and specific enough that the team will know what counts as "done" for that chunk of work.

Let team members pick items from your team's overall project plan. Use your list of requirements as a backlog or list of tasks that need to be done. Let each team member choose the tasks they will take on for the coming week.

Even though each team member has their own chunk of work they are agreeing to finish, you will need to work together to help each other finish the work, teach each other, and guide each other as needed. All parts of the final team project have a shared grade, so each team member is responsible for the success of the project.

Deliverable: For the first week of this new project, a list of each team member's name followed by the tasks they have agreed to get done for that week. E.g. Sally: Build the underwater level. Mark: Get the feedback sound effects and script them to play when the player walks into the trigger. Be sure to create a list like this for each team member, at the beginning of each week.

Week 1 Planning:
Lucas Johnson - Basic Player Movement & Shooting
Zach Wilson - UI & Menu Planning
Devun Schneider - Level Concept Design Sketches
Colin Gamagami - AI Car Movement
John Green - Collect/Make assets (Cars, buildings, etc.)

5) **Your game prototype:** No matter how far along you are in development, I want to see your game prototype. Create a GitHub repo, get the .gitignore file on Canvas, add the .gitignore file to your repo, then add your Unity project folder to the repo.

Deliverable: a URL web address linking to your project repository on GitHub. I will take off points if you do not use a .gitignore file. So please remember that step.

https://github.com/PlatFormPlayZ/CIS-350-Project-2

Submit the deliverables listed above as a **single** .docx or .pdf document on Canvas under Assignments before it is due.

Main Requirements for a Playable Game Prototype <u>(These requirements need to be met for Project 6, for the version due November 20th – they do not need to be done by when Project 5 is due)</u>

1. Tutorial elements that teach the player what controls to use (do **<u>not</u>** assume players know WASD=move or use the mouse to look, but use those standard controls if they make sense for your game)
2. A challenging goal is clearly communicated to the player
3. Achieving the goal of the game has a medium level of difficulty – neither too hard nor too easy (or the game has difficulty settings the player can choose or uses dynamic difficulty adjustment)
4. The game has success and failure conditions, meaning the player can win or lose each time they try.
5. Feedback about whether the player succeeded or failed each try is clearly communicated
6. There is a game loop, meaning the player has the ability to retry or reload the scene without closing and re-opening the game or pressing the Play button in the Unity Editor