TANK GAME

THE ATARI CLASSIC: COMBAT

Table of contents:

Executive Summary

Problem

- The problem were trying to solve is boredom

Process

- We will be utilizing the Agile SDLC

---

1. EVERY DAY we'll have our routine stand-up to discuss progress
2. Requirements will be planned and User Stories will be created
3. Each team member will be assigned a task
4. Deliverables depending on the task will be assessed and assigned
5. Everybody will be given testing and security requirements

6. A deadline for the Sprint will be decided

Solution

- We will be recreating (*to the best of our ability*) the Atari classic, Combat

Expected Deliverables

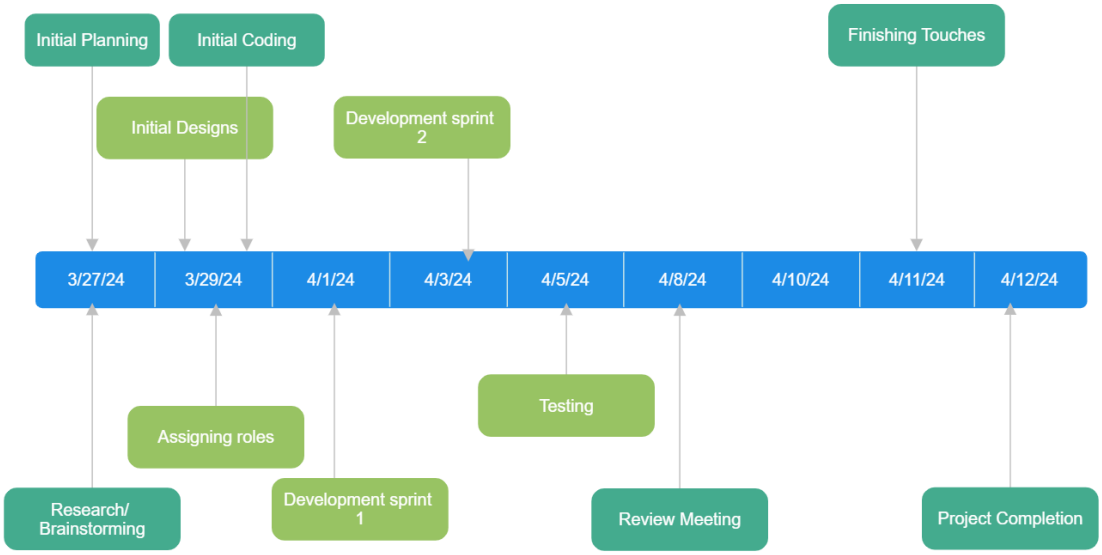
- A game menu to start and exit the game
- 2D graphics
- Two player functionality
- Movable tank sprites
- Hit boxes
- Projectiles
- Scoreboard
- Security regarding the game save state

Project Scope

For this project, we plan to make a simple "Tank Game". Our inspiration is the "Combat" game from Atari. Just like in combat, our tank game will have two sprites. The players (as sprites) will try to shoot each other. They will be able to move around. There will be walls that will act as boundaries for the players.

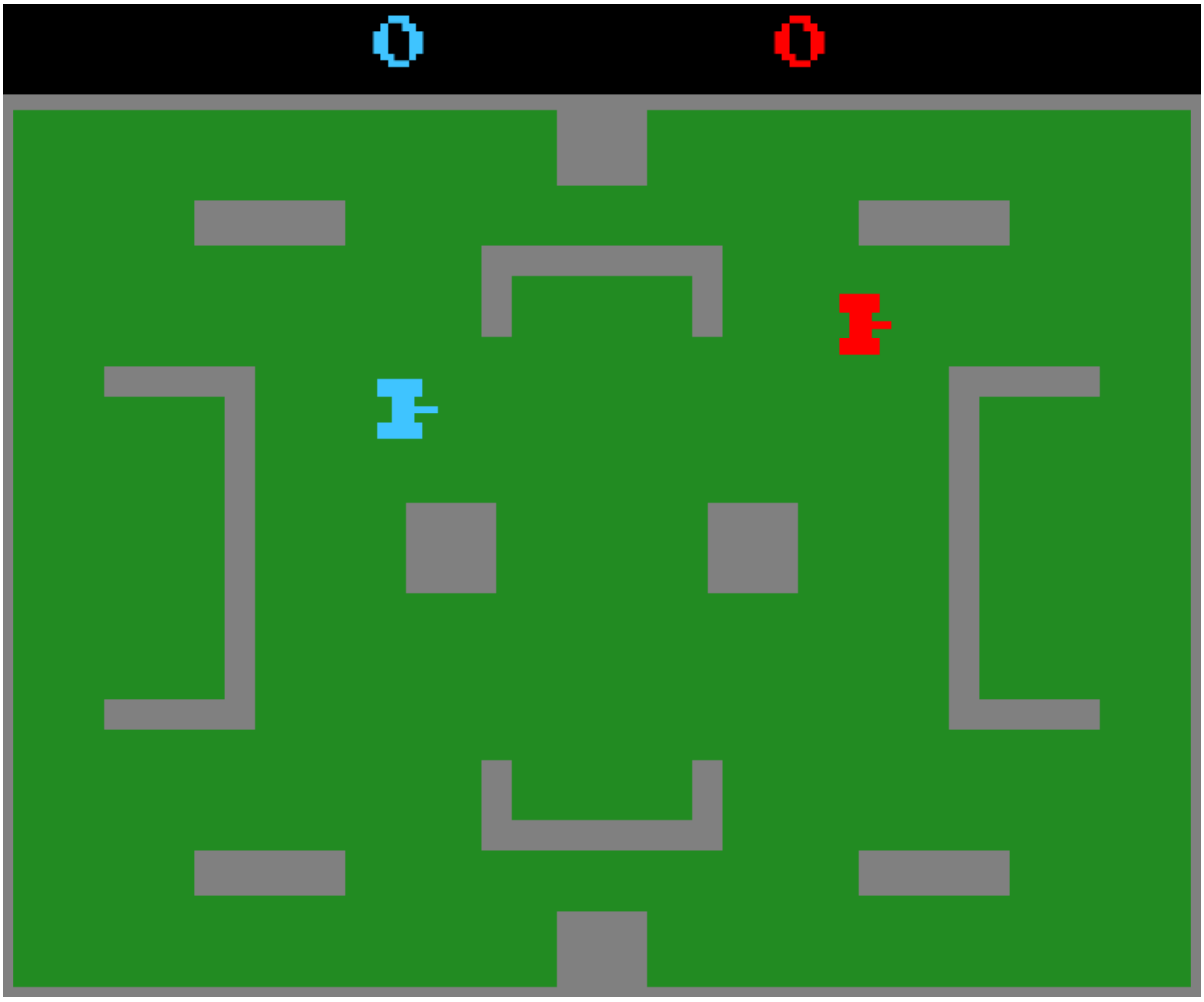
Currently, our scope only includes one level of play. Thus, there will only be one screen that actually includes gameplay. Additionally, we will have a start menu that allows the user to either start the game or exit the program.

PROJECT TIMELINE & MILESTONES



SPRINT

DOCUMENTATION



SPRINT 2

Mead Sprint 2

Did?

The team was tasked with getting familiar with pygame on a basic level. We all created a window and a rectangle. Also, Created documentation and converted it to physical for Ada.

Will do?

I will create the Main.py for the deliverable. Creating a window and clock for our game, as well as starting to talk about class necessities.

Olivia Sprint 2

Did?

The team was tasked with getting familiar with pygame on a basic level. We all created a window and a rectangle. Also, Created documentation and converted it to physical for Ada.

Will do?

I will create the Main.py for the deliverable. Creating a window and clock for our game, as well as starting to talk about class necessities

Perkins Sprint 2

Progress

- Once we knew what our project was we installed pygame and familiarized ourselves with it.
- I made a window containing a rectangle that you could move along the x and y axis.

Moving forward

I plan to turn the rectangle into a sprite with an array of images to add a movement animation.

Schnell Sprint 2

Sprint 1 Review

Sprint 1 consistented of downloading PyCharm to my computer, as well as ensuring that a python interpreter was connected to PyCharm. The deliverable was simply a Python file that said "Hello World."

Sprint 2 Goals

For Sprint 2, my goal is to get more familiar with Pygame by creating a window and changing a few of the window's properties. Additionally, I would like to draw a colored shape in the window.

SPRINT 3

Mead Sprint 3

What I did?

I wrote a Main.py starting a pygame window and game running loop.

What I'm going to do?

I'm going to add a diagonal direction of motion for the tanks.

Olivia Sprint 3

Did?

I was tasked with the readme and creating a bullet to shoot out of the tank. While still familiarizing myself with pygames. I created the readme file and create a bullet.

Will do?

I will now create a class for the bullet so it can be called and used multiple times throughout the game.

Perkins Sprint 3

Progress

- The Player class was added to establish fundamentals. It grants the ease of creating multiple players.

- The Player class simplifies my ability to give each character multiple sprites to iteratethrough.
- Key presses determine which image to cycle to by changing the heading of the tank.
- I created the executive summary.

Moving Forward

I was tasked with creating the scoreboard.

Schnell Sprint 3

Sprint 2 Review

Sprint 2 consisted of downloading Pygame to my computer and importing it into the Tank Game project. I successfully named the window, set a background color, and drew a colorful shape in the window.

Sprint 3 Goals

For Sprint 3, my goal is to create a Tank class and an implementation of the class in main. I want to have two pictures of tanks, with one tank controlled by the up-down-right-left arrow keys and the other tank controlled by the A-S-W-D keys.

SPRINT 4

Mead Sprint 4

Did?

I attempted to add diagonals to the game. I was only able to create the sprites.

Will Do?

reference the sprites in the code and start to set up the directions.

Olivia Sprint 4

Did?

Created Class for bullet in main and moved it to its on class bullet.py

Will do?

Implement and make it work in main.

Perkins Sprint 4

Progress

- I added an arcade style font to be used for the scoreboard
- A black bar was added to the top of the screen to better visualize the score
- The score for each player was added onto the black bar, by default its zero.

- Each score total is color coded to the color of the player to be instinctually intuitive.

Moving Forward

I will be turning the scoreboard into a jumbotron with animations.

Schnell Sprint 4

Sprint 3 Review

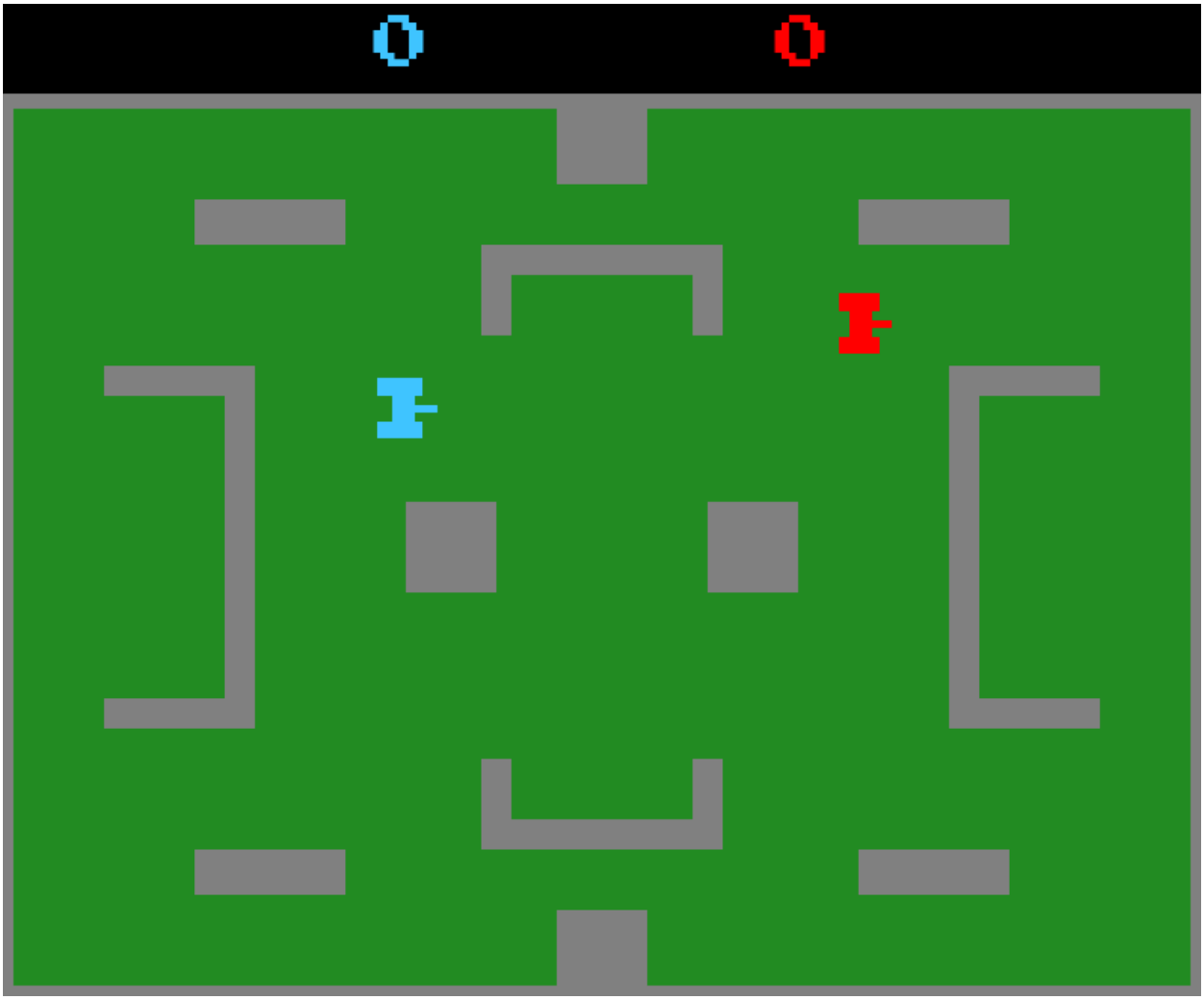
Sprint 3 consisted of creating a simple Tank class. I then imported the class into main. In the same folder as main, I added two pictures of cartoon tanks I found online. The tanks could spawn at the given XY coordinates, as well as with a given size restriction. I implemented logic in the loop in main so that one tank would be controlled by the up-down-right-left arrow keys and the other tank controlled by the A-S-W-D keys. Jonathan also worked on Tank controls. Since he went further then me (he made multiple sprites for the tanks, and changed the direction the tank faced based on the keys pressed), his class will supersede mine. Although my code for this won't likely be in the final implementation, it was enjoyable to get more familiar with Pygame.

Sprint 4 Goals

For Sprint 4, my goal is to create a Wall class. As the name implies, this class will implement the walls in the game. My main goal for this is to implement logic that constrains where the tanks can drive. Thus, I do not want tanks to be able to drive through walls or outside of the screen. I

hope to implement logic to achieve this goal, as well as potentially logic to prevent the tanks from driving over each other.

End of sprint 4



Jonathan's scoreboard

SPRINT 5

Mead Sprint 5

Did?

I continued my progress toward my diagonals trying to decipher the code. However, I also had to complete physical documentation.

Will Do?

finish up and polish up the physical documentation.

Olivia Sprint 5

Did?

Got bullet to move in main but got stuck with how to make it move when the tank is a different direction.

Will do?

Make the tank shoot from any direction and add the point to score board.

Perkins Sprint 5

Progress

I added an animated fire ball sprite to make loops around the scoreboard.Text displaying "Tank Game" was also added to the scoreboard to loop in the opposite direction.

Moving forward

Schnell Sprint 5

Sprint 4 Review

Sprint 4 consisted of creating a Wall class and implementing it in main. First, I created a .tmx file using "Tiled" that basically just consisted of a bunch of tiles organized into the wall structure shown in main. Each tile contains an integer called 'solid' that is set to 1, indicating that the walls are solid. (An integer was used instead of a boolean as some issues with booleans may exist in Tiled.) The Pytmx library was then imported into the tank program and utilized to incorporate the tilemap. At first, the "y" coordinates were inconsistent with what was shown on the screen. For example, some areas that the wall appeared could be driven through and some areas that no wall was shown would block the tank. After adjusting the "y" coordinates to account for the scoreboard, the walls began to work as intended. In main, logic needed to be implemented so that the tanks could not drive through walls and so they functioned as expected. I did not implement logic to prevent the tanks from driving over each other.

Sprint 5 Goals

For Sprint 5, my goal is to create a simple start screen. Basically, the screen will just show up prior to play and give the user the option to

either start the game or exit the program. If time permits, I hope to implement logic to prevent the tanks from colliding with one another. However, like last sprint, this will be a secondary priority.

End of sprint 5

TMX map described in Jordan's Sprint recap

Sprint 6

Mead Sprint 6

Did?

I updated and upgraded the physical documentation. I also started gathering images for the How To Play part of the documentation.

Will do?

Finish polishing the Sprint documentation and create a final Deliverable section in physical documentation.

Olivia Sprint 6

Did?

Helped Jonathan with moveing bullet class to get it to work and collision detection for if hits opposite player.

Will do?

Sprints are over. Just present to class on Monday.

Perkins Sprint 6

Progress

I added a respawn function for later implementation.

Moving forward

I'll be helping out with getting the bullet to work since its been trouble.

Schnell Sprint 6

Sprint 5 Review

For Sprint 5, I followed a tutorial and ended up creating both a start screen and a game over screen. I used ChatGPT4 to generate art work, then editted it to include the name of the game. For both the start screen and the game over screen, "Enter" either starts or

restarts the game, and "X" quits the game. In main, game states were implemented so this could function properly. I did not get to the secondary priority of preventing tank collision.

Sprint 6 Goals

For Sprint 6, my goal is to handle tank collision. This has been a secondary priority for me in the last two sprints, so I would like to get this completed. Also, I will add simple guidelines / instructions for the user.

Sprint 7

Mead Sprint 7

Did?

I completed the final physical documentation for the project. I also made some time to take care of the difficult and delicate task of dealing with Bhavya.

Will Do?

Present the completed deliverable as this is the last sprint.

Perkins Sprint 7

Progress

- The bullet was ridiculously complicated but its all over now.
- Both players can shoot a bullet color coded to their tank.
- A cooldown timer to shoot was added
- Hit detection was added, hit box needs tinkering

Moving forward

- Connect hit detection to our death animation
- Connect hit detection with incrementing our scoreboard
- Connect hit detection to our respawn function

Schnell Sprint 7

Sprint 6 Review

In Sprint 6, tank collision was handled. It was implemented relatively easily by an addition to the player class and a few game updates in main. For each directional key pressed, the other tank will be checked for in a similar fashion to how the wall is checked for.

If the other tank is found, it moves one "step" in the opposite direction of the key pressed. Also, an instructions menu was implemented. Very similar logic to the start menu and game over menus was utilized to make this happen.

Sprint 7 Goals

For Sprint 7, my task is to create the PowerPoint aspect of our final project. I am a major PowerPoint nerd, so I am definitely looking forward to this! :)

HOW TO PLAY

SETUP

First, we need to install the game.

Installation

1. Ensure you have Python installed on your system.
2. In the Command Prompt, install pytmx using pip as it's a dependancy:

   ```
   pip install pytmx
   ```

3. Install the Pygame library using pip:

   ```
   pip install pygame
   ```

4. Clone or download the repository containing the game files.

Running the Game

1. Navigate to the directory where the game files are located. (Tank-Game/Code/Tank Game)
2. Run the following command in the Command Prompt:

   ```
   python Main.py
   ```

3. Enjoy playing the Tank Game!

HOW TO PLAY

Objective

The objective of the game is to outmaneuver your opponent and score points by eliminating them using your tank's firepower.

Controls for Player 1 (Blue Tank):

- W: Move Up Screen
- S: Move Down Screen
- A: Move left
- D: Move Right
- V: Fire Cannon

Controls for Player 2 (Red Tank):

- UP Arrow: Move Up Screen
- DOWN Arrow: Move Down Screen
- LEFT Arrow: Move left
- RIGHT Arrow: Move Right
- M: Fire Cannon

Hitting the 5 key also brings you back to the main menu while in gameplay.

ENJOY!! FROM THE SCRUMMY TANKERS

- Carson Mead (Scrum Master)
- Jon Olivia
- Jonathan Perkins
- Jordan Schnell