Advance Programming in CPP, Assignment 1

Lior Buchmeier
Tal Weiss

April 2025

1 Introduction

The Tanks Game project is a turn-based simulation of a battle between two players controlling tanks on a 2D grid. Each tank can move, rotate, and shoot shells to eliminate the opponent while navigating around obstacles like walls and mines. The simulation runs automatically using simple "AI" algorithms to determine movement and attack actions.

2 Architecture

The program is divided into several modules:

- Entities: (Entity,LivingEntity,Tank,Wall,Shell,Mine):
  Represent all objects on the game board.
- Game Manager (GameManager):
  Manages the game flow, applies entity updates (and tests the legality of these actions), board state, collisions, and win condition checks.
- Algorithms (game_algo):
  Contains basic AI functions like enemy chasing and shooting decisions.
- Utilities (utils):
  Shared code for directions, actions, logging, and helper functions.

3 UML Class Diagram

Key Points:

- LivingEntity adds direction and movement to entities.
- Tank and Shell can move and perform actions.
- GameManager owns and manages all entities and players.

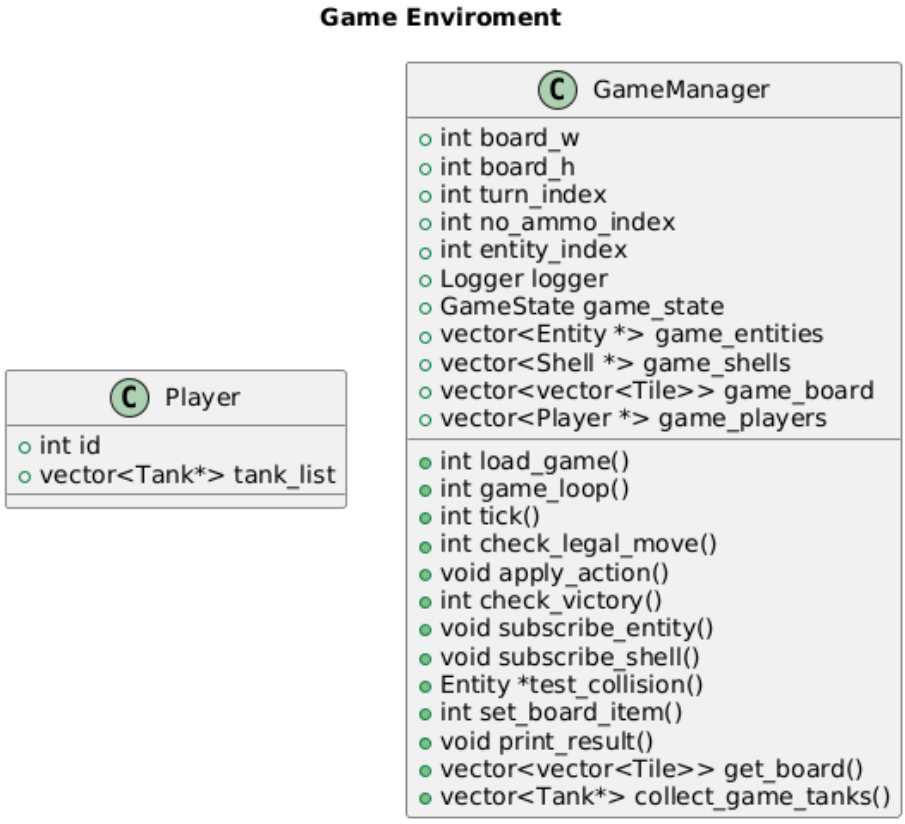

Figure 1: UML for the Game Environment, aka: the GameManager

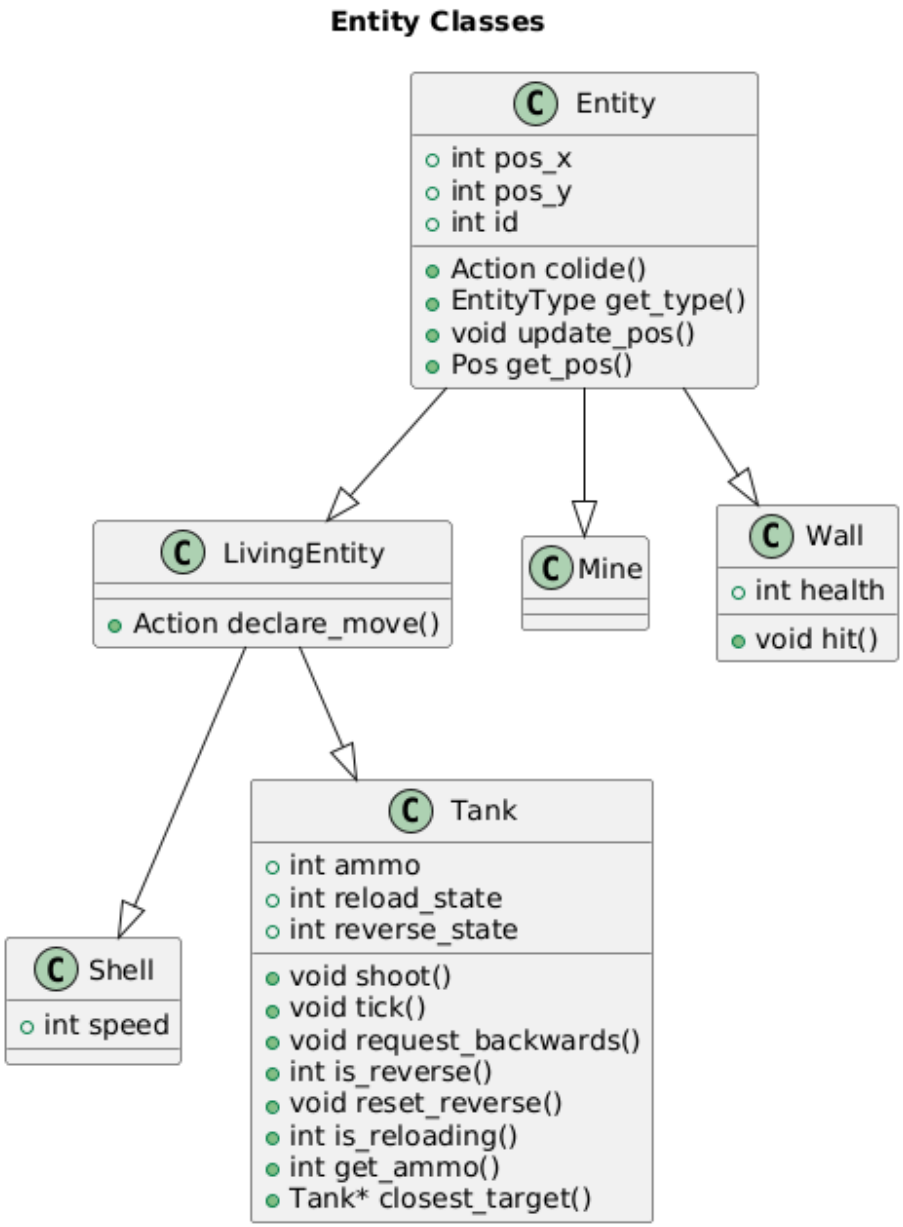

Figure 2: UML for the entities in game

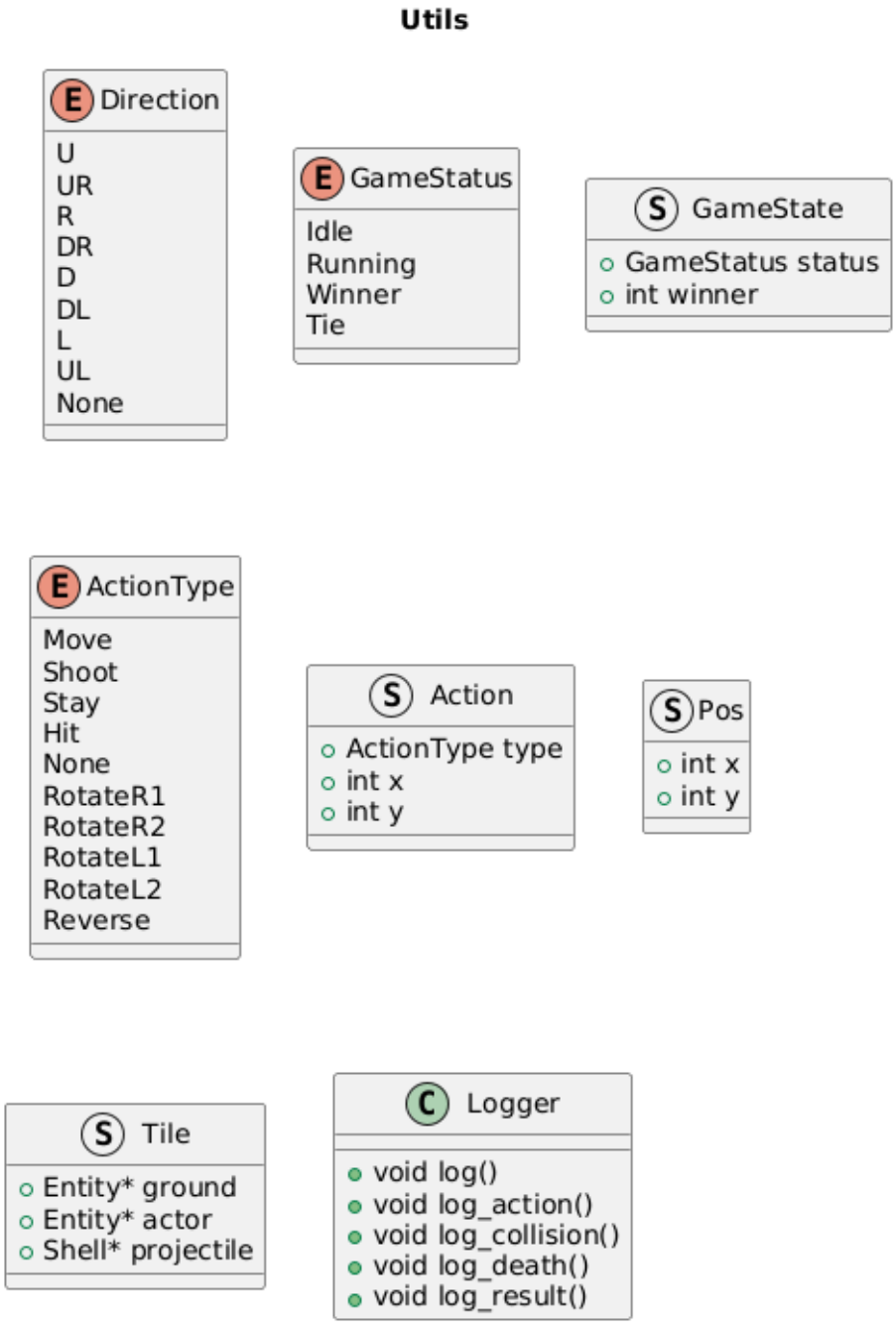

Figure 3: UML for the utils classes.

4 Game Flow

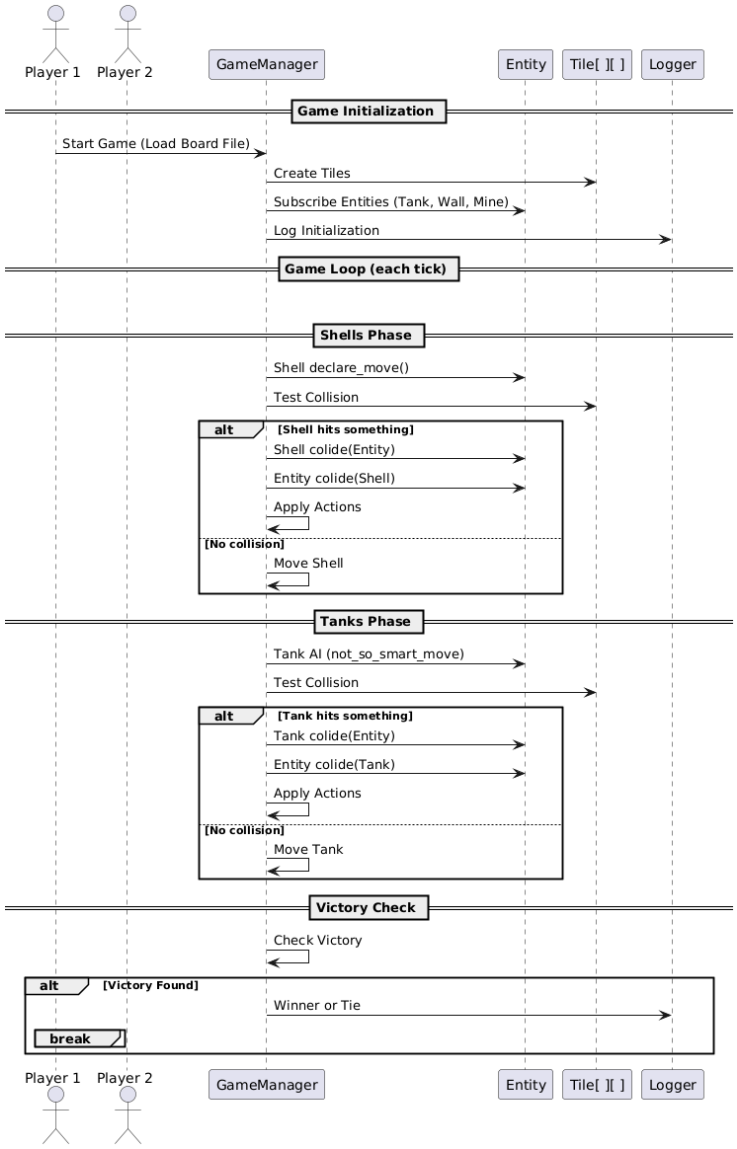

Figure 4: The general game flow

5 Important Design Features

- **Tile-Based Board:**
  Allows a ground object and a living entity to exist in the same cell.

- **Wrap-Around World:**
  Tanks and shells moving off one edge appear on the opposite side.
- **Simple AI:**
  Focused on local decisions: chase the nearest target, shoot if aligned.
- **Diagonal Collisions:**
  Checking three tiles in diagonal movements to simulate broader collisions.
- **Ram Condition:**
  The tank algorithm will try to ram the enemy when ammo is depleted, this is since draw is better than losing.

6 Known Limitations

- AI does not try to predict future enemy movements.
- Tanks do not avoid shells very intelligently beyond immediate proximity.

7 Future Work

- Add a testing unit for the next projects.
- Organize better the pipeline (EventBus,...).
- Add UI for the game,(an attempt was made using rust FFI for use in fullstack app).
- Try to add a strong RL based AI.