BEAUTIFUL PRESENTATION PAGE

Tank Trouble Remastered

Everyone has ever played an arcade game online. One of them is particularly remembered in our childhood: Tank Trouble. It’s a funny multiplayer game where you drive a little tank in a maze to try to shoot your friends’ tank. You and your friend are on the same keyboard. That’s why…

We introduce you **Tank Trouble Remastered** !

We have completely re-coded the game and added new features. The original game opposes two players on a map selected from a predefined (finite) set, and you only have one type of tank.

Our first objective was to code the original game. It includes, among other: dealing with collisions (of the tanks and bullets), trajectories of the bullets (with bounces off the walls), acceleration (to make the game more enjoyable to play) and user-friendly graphical interface. However, we went further with additional features, such as randomly-generated maze, several super tanks and bullets with special abilities, sound effects, game menu (to choose your tank and the map), “different game modes”…

We took full use of object oriented programming in our project since we have similar objects such as entities (tanks and bullets). Therefore, our project archive is divided into several classes (non-exhaustive list):

|  |  |
| --- | --- |
| Class name | Primary features (not exhaustive) |
| GamePanel | Start thread |
| KeyHandler | Key bindings |
| MovingEntity | Abstract daughter class of the abstract class Entity. Contains the basics of entities (movement) |
| Tank | Extends MovingEntity with collisions, displacements, draw, shoot |
| Bullet | Extends MovingEntity with collisions, displacements, draw, shoot |
| Tank\_Super and Bullet\_Super | Extends from Tank and Bullet respectively. Enables to create entities with special capacities |

Table : Division of the project in classes

We used Git to code in a synchronous way and manage versions. We spent some time at the beginning to get used to it, but it’s a great way to code together with partners. You can find hereafter the table of involvement of everyone in the project.

|  |  |
| --- | --- |
| Name | Involvement |
| Bonnaire Léo | 25 % |
| Merle Adrian | 25 % |
| Rosard Alexandre | 25 % |
| Sibileau Antonin | 25 % |

Table : Participation of members to the project

Note: The UML graphs presented here are not whole, you can find the entire diagram with the link in our bibliography, at the end of the document.

This document details some of the features implemented in the project. We chose to group classes together in packages to talk properly about their properties. We will discuss the Game core, Graphical User Interface and Entities clusters in the following document.

# Graphics

## Starting menu

# Game core

## GamePanel

## Map generation

# Entities

## Moving Entity

## Tanks

## Bullets

# Conclusion

# Bibliography

* RyiSnow [online] [accessed on 2022, march 25th]. How to make a 2D game in Java. Available at <https://youtube.com/playlist?list=PL_QPQmz5C6WUF-pOQDsbsKbaBZqXj4qSq>
* Splash sound effect <https://www.youtube.com/watch?v=nZNR5i9qN4w>
* Pew sound effect <https://www.youtube.com/watch?v=i6DRo6v78yg>