



Spaghetti Lagitti

Anti-Tower-Defense

By: Konsta Jalkanen, Teemu Hiltunen, Lauri Heiskanen, Aleksi Natunen, Venla Kuusinen, Vili Jokela, Petra Rautiainen, and Tuomas Lättilä



WWHW for our project

WHY?: For the memes, for the glory. We wanted to make a game and game jam was ideal motivator to do one.

WHAT?: Anti-Tower-Defense, with pudding. It was a idea demo to test if reversed Tower Defence had an idea in it.

HOW?: Yes, it does kind of work. Made with Open-source software Godot.

WAU effect: It crashes when you get to the end.



Preparations

Before the start of the event we had planned to discuss potential designs, which game engine we would use and how each of us were planning to participate in the event. Our decisions landed on Godot open source game engine which seemed to be the best for beginners and for our decision to make a 2D game. As for the game itself, we wanted to go with the anti-tower-defense -style and hoped that the theme of the year could be fitted for our decision.



Preparations, Learning Game engine

After the decisions about which game engine to use and which type of game we would be making, it was suggested that everyone who were going to participate in the event should get to know the engine on at least some level so the actual event time wasn't used to learn to use the Godot UI, game object structure and how the scenes and nodes work. In the end, only two did look Godot before the game jam.



Preparations, Planning where to be.

Considering our participation in last year's Finnish Game Jam event, and the problems it caused for us and our effectiveness, we decided to participate in the event by gathering at one of our team member's place to actually build the game. This decision was made with the thought that as our team was quite large, and our tendency to be quite vocal that it would be best for other jammers to relocate our group elsewhere.



Bonus Preparations, Few hours before the start

As a bonus preparation before the actual event time, we decided that we should get everyone's environments and version control up, so we could have a smooth start of the event. Teemu created the github repository to which we would store the project and added contributors for the repository, so all we had to do was to teach ourselves how to use git with Godot. Alas, we had no idea that Godots official plugin "Godot Git Plugin (4.1+)" was quite an experience with it working best for two of our five developer members. The fast decision to dump the whole plugin and access version control via command line tools and github desktop turned out to be **the way**.



Main Idea, Tower defence but reversed

The initial design for our game was that in our game, there would be a couple of routes that the player could send our characters "Telebatties" in a certain order to try to get to the finish line through the barrage of towers "Monkeys" which would shoot bullets "Trash" towards the characters. We didn't necessarily want to create a bullet and shooting mechanics to destroy the Telebatties but rather discourage their access to the finish line.



Main Idea, Mixing the theme to the Idea

As our story for the game was basically that at the beginning, we had a crying baby (as shown in the cut scene) and Telebatties were trying to go cheer the baby up by getting to it and there were these Monkey towers which would try to stop the Telebatties movement by throwing trash and other objects towards them. So getting enough Telebatties to the baby would be the end goal of the game and the player would win.



Main Idea, Design choices

Each level would have waves to allow the player to try to get to the finish line with a certain amount of Telebatties to access the next level. Before and after each wave the player would be able to set up to six Telebatties acredits in the shop menu. Each Telebattie had their own stats, so using different ones would provide different strategies to achieve the goal.



Extravagant evening, Starting the project

As we had prepared ourselves a couple of hours before the event and felt that this setup might actually work, we gathered with most of our team to the event site to hear the long awaited theme of the Jam with fingers crossed it would be a fitting one. Once we heard the theme, we were honestly a bit worried as the initial idea we had had to be changed in some way that would create amusement literally, among players, or the audience.



Nightlights, first night

Luckily, we had creative force in our team and as we reassembled in one of our team members' apartment we had a vision how we could make the game amusing via graphics and audio, maybe with an occasional intended bug here and there. Great, now we have something concrete to move towards and our first night we basically discussed, planned and ate and most of all, drank. Friday was maybe not so productive in the development part but it was essential for the design part and it allowed us to start the project with a bang at Saturday

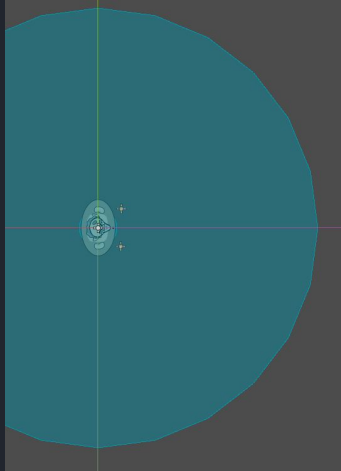


The New Dawn, Saturday morning

As our super design from friday had been completed to a point that we could start the project development on saturday morning we started with the first mechanics and testing the idea further. Using krita to make background images and using Godot's pathing allowed us to create a lot of maps in a short time and each map had common components so they would be easy to implement, in theory at least, allowing us to focus more on the features.

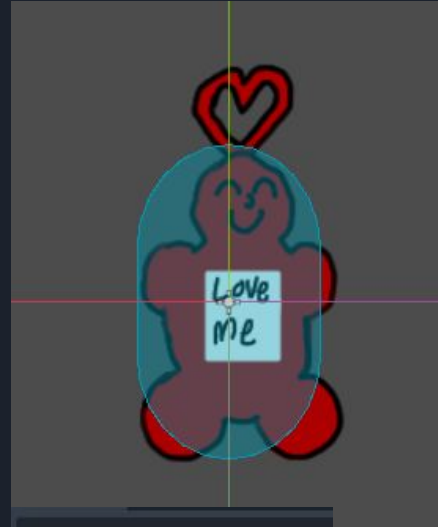
Turret And Player Character Structure

- ▼ Tower
 - CollisionShape2D
 - Head
 - Left
 - Right
 - DeathZone
 - CollisionShape2D
 - MarkerLeftHand
 - MarkerRightHand
 - AmmunitionHolder
 - AudioStreamPlayer



← Monkey turret with Area2D as a detection zone

Telebattie with → CollisionShape2D



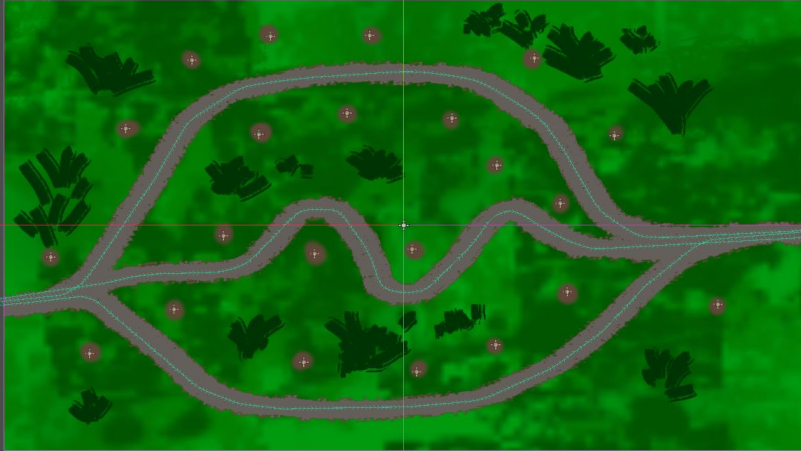
- ▼ Telebatties
 - Sprite2D
 - CollisionShape2D



Day of progress, Saturday

We created the working behavior of the map by creating a common scene "world" that would hold nodes such as Path2D, Sprite for the map image, camera to hold the view in the correct spot in the game area, and Node (folder) for towers and our telebatties characters as these were dynamically added to the game based on the players queue. This "world" would be used as a parent scene from which the levels would inherit.

First map:





Mental toll, Saturday evening

Another thing to note about Saturday was our own performance. We really hadn't slept at all, not that we didn't try but the decision to lay on one of the member's couches with others might not have been the wisest choice. So, we were a bit slow during the day and all we really got done was the basic functionality like pathing, spawning characters and towers.



Glorious Morning, Sunday morning

In our last hours of development, we were testing that the main structure of our game works and different scenes can be played in the correct sequence, and at this point we noticed that the UI did not work as expected, actually, it crashed pretty consistently. As it wasn't made by one person, it was pretty hard to debug it and I could not for the life of me figure out why it didn't work as expected. We haven't had a previous Godot project in which we created a simple UI and then decided to try and make something that would have the bare minimum requirements such as option to pick different Telebatties and insert them into a queue in order with limited amount of money and a start button to move to the next scene.



Clock was ticking, Sunday

This didn't actually take that long to make, nonetheless it used time which we didn't really have. The first version of UI can be seen in Figure 1 and the one used in the end product can be seen on Figure 2. The difference between functionalities lies within the option to insert Telebatties in specific positions in the queue as it was only made sure that the first bought Telebattie is inserted to the end of the queue so the Telebatties would be deployed in First-in-First-out principle. This simplified the process and while it actually messed up the buttons and different Telebatties, as in Purple didn't actually give Purple Telebattie but a pink one, it did work in a sense that the game could be played.

Pictures of the shops:
Figure 1

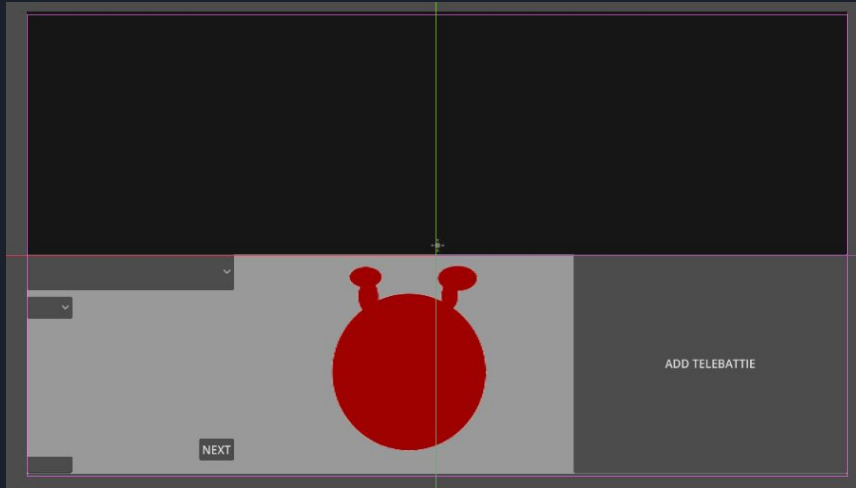
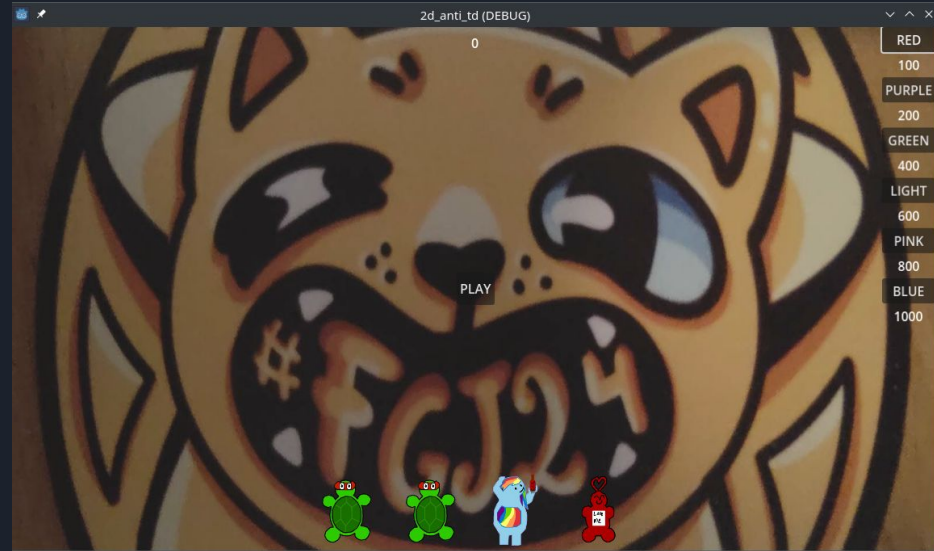


Figure 2





Final battle, a few hours before deadline

After getting the UI in working order with bare minimum, the last two hours, we were pretty much too tired to do anything useful as most of us hadn't slept basically at all since we couldn't fall asleep so the rest of the fine tuning was kind of missing. During these last hours most of our team left to sleep, and only three was left to present.



Final Thoughts

The weekend we just spent making a game took a heavy toll on our own time regarding school assignments and it created a pretty tight schedule before the event and after the event. Since we had a lot of school work to be done, we did not really reiterate that much what was accomplished during the weekend and the only things we went over was with our development group during lunch and we discussed in which things we succeeded and in which we need to improve, most of which are already discussed in this document. With the next iteration of the short development sprint, we should plan resting to get vitality up.



Future is now old man.

This inspired us to make more games. We decided that we will do another game during a weekend like it was a game jam.

Konsta may try to make 3d thriller edition from the Anti-Tower-Defence with interesting story and puzzle mechanics.



Timeline & plans, potential

If Konsta does not get summer job:

- during may figure out basic 3d model handling in Godot (28.5.2024)
- recruit Teemu for the project (29.5.2024)
- Make flesh out 3d models with blender (2.6.2024)
- Actually look what has been done and figure out if it could be a sellable product (3.7.2024)

If Teemu does not get a summer job:

- He might or might not do another game
- If he does, the might do it with friends such as Konsta
- Making a 3D game could be fun way to learn more about 3D space
- Also learning about 3D modeling and animation

Group Composition and Roles

Lauri Heiskanen	Programmer, Graphics
Teemu Hiltunen	Programmer, Design structure
Konsta Jalakanen	Programmer, Cut Scenes, Sounds, Planning, Work space, and Graphics
Vili Jokela	Visuals
Venla Kuusinen	Graphics
Tuomas Lättilä	UI, Graphics
Aleksi Natunen	Graphics
Petra Rautiainen	Sounds, Graphics