Starving Worms,
Crawling to Victory

Cléa Benz, Nicolas Imhof & Adrien de Gottrau

Alpha Release Report

We are at the alpha release. The main things we worked on for this update are the assets, the user interface and the compatibility with mobile devices. While we made very good progress on the user interface, there is still much work to do with the assets and the mobile controller has still room for improvement. In the following we give an updated overview about our layered requirements.

Legend :

green = done and finished
orange = still some tuning to do
red = to be done for the final game

Functional minimum :
- one simple map (with grid structure)
- one fruit (cherry)
- worm spawning and walking (shortest path) to the opponent's fruit
- terrain elements: forests (that react to the seasons), mountains
- primitive graphics
- player interface (information display: seasons, energy)
- ability to change the seasons

Low target :
- simple animations
- singleplayer (primitive AI)
- ability to switch the terrain of the sections
- more terrain elements: water, mountain paths
- game menu

Desirable target :
- two worm clans
- advanced graphics
- advanced AI
- small tutorial
- good game balance
- game presentation video

High target :
- multiplayer
- additional maps
- very good game balance
- intuitive user interface
- map generator
- ingame intro (video)

Extras:
- worms fight each other
- worms evolving
- event at each season start
- interactive tutorial
- campaign
- dynamic weather system

- chat system
- many terrain elements
- towers
- extra terrain elements, extra worm types
- fog of war

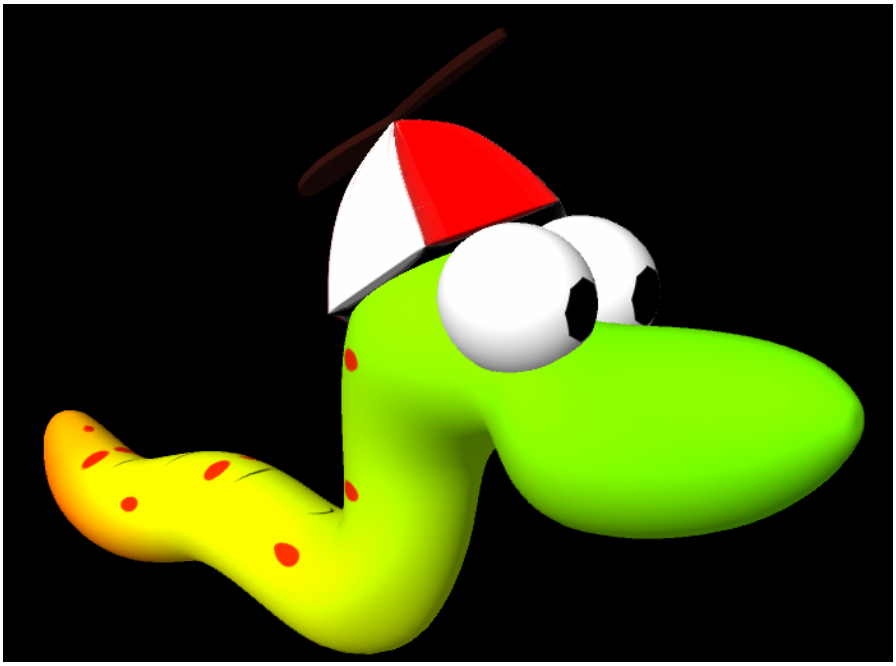
Assets and User Interface

One of the main focus since the interim report was the whole asset and user interface package. It was a very time consuming task, but in the end we are really happy about the modeling we did. We really tried to make every asset from scratch and avoided downloading from the internet as much as possible. We created the animations either in Maya or directly in Unity depending on which software was better suited for it.
We used Maya, Inkscape and Paint.net for almost everything, depending on what we needed. The result of this work is a really good looking user interface and some really nice assets in the game. We also managed to keep an overall design scheme for the game, so that the game graphics look consistent and coherent.

The user interface creation was a really time consuming task, because the best way to find the right place for every part is by trying it out and see what the changes look like. After some trial and error, we managed to get a properly looking interface, that scales depending on the device screen size. This will allow the game to be played on small or big screen without seeing any difference in the interface.

In the end we are quite happy with our graphics and with our user interface. We probably won't change them a lot until the final version though small changes are yet sure to be made.

Examples images:



Extra smooth Cherrom-worm

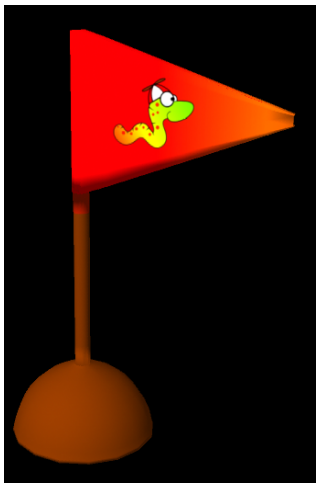
TODO image

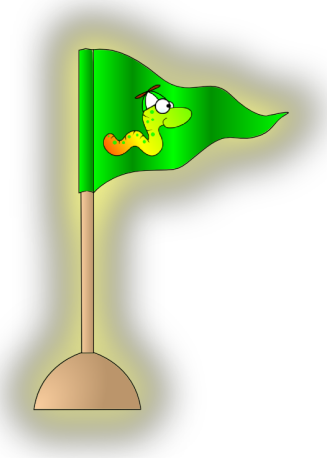
Ingame Applom worm

Ingame season changer & ingame square changer

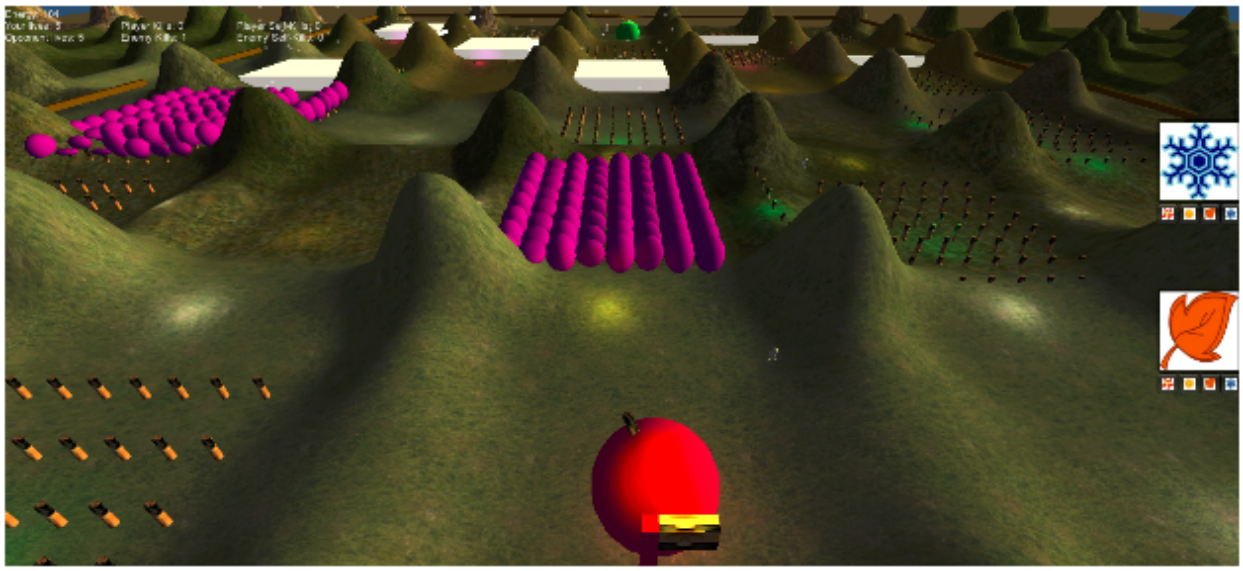
Season changer & Energy display

Square changer

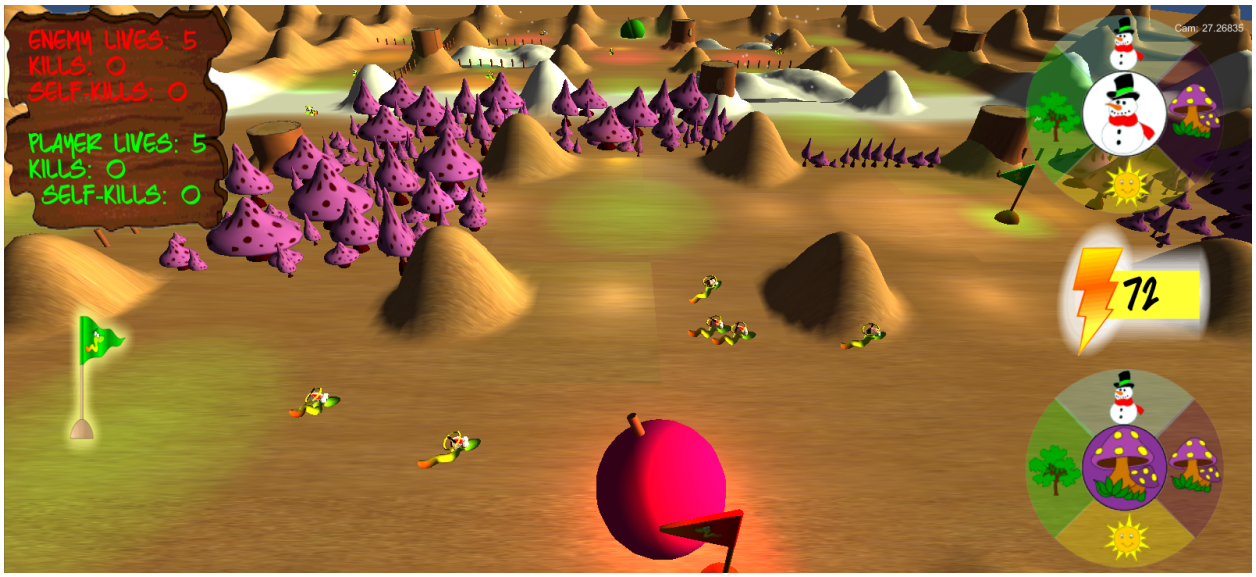




Flag 3D & 2D



Game appearance at the interim report



Alpha release

Android Mobile Controller

For controlling the touch inputs on the android device we used the provided functions of Unity. In general the touch inputs distinguish between the number of inputs and are capable to say when an input starts, is continued or is ended. With those functions we implemented

a swipe and a zoom function. Both functions are restricted by the viewing field we want to provide to the user. The difficulty was to find the right speed for the swipe function, which should be calculated according to the screen width of the device.
Additional the user should be able to set the flag into the map. This calls for a function where the user drags the 2D flag to the desired place. At this location the 3D flag occurs.

Menu - Option
Maybe some users like to switch of the volume, so we added an option selection where the user can regulate the sound volume. Additional we intend to add there some options to change the terrain size.

Performance

Another problem we ran into and which we discovered when we ran the game on a mobile device was the performance. In the beginning the game was strongly lagging. We realized that one big issue was the tree model that we created by ourself. After we have removed it completely from unity, the frame rate dramatically increased again on the desktop pc from around 20 to 40. The model had a normal amount of triangles, so we don't really know yet what was wrong with that.
But still the frame rate on the mobile device was not good enough. The profiler of the unity3d's pro version helped us then to find the bottlenecks. It turned out that some simple scripts that we wrote took about 30% of the computation time. Because the scripts were very simple, it was immediately clear that the use of unity's *FindObjectsOfType* function is horribly expensive and should definitely not be called every frame rate.
Another problem is the season change which always came together with a strong lag. The profiler revealed different sources of the problem. On thing is that it is quite expensive to activate/deactivate several gameobjects. The other thing is that the carving of the navigation mesh is a very expensive operation. We are reasoning about dropping the use of the *NavMeshObstacles* which trigger the carving. But we then have to keep track about the walkable paths in the game grid by ourselves and tell the worms where to move next.
Now the things that take the most computation time according to the profiler are the *Gfx.WaitForPresent*, for which we don't know if there can be done anything about it , and the Camera.Render and GUI.Repaint.

Conclusion and next steps

In the last weeks we managed to improve the game's appearance and achieved the fancy looking comic style that we wanted. There are still many things to improve at the controller and the assets but this will not be our main focus in the coming weeks. Instead we will now focus on the gameplay. We expect that we have to add additional rules to ensure that the game is fun to play. Also we intend to provide some options for the player, so that the game can be played in different ways to improve replayability. Of course the player's opponent, the AI is one of the main things to do next.