CJ PRODUCTIONS

POSTMORTEM:
WAVE VOYAGER

Contents

1 Game Design

The newest version of the game contains the most important parts that make the game to a Music, Indie, Racing game.
While developing we decided to remove some of the planned features like the enemies, choosable vehicles, a shop where the player can buy upgrades for his spaceships as well as collectibles, but instead of that we wanted to take more effort in getting a correct working beat detection to deliver the feeling to the player that he is actually riding the chosen song.
Some of those planned features like the shop and collectibles were removed because we underestimated their workload and others of them like enemies were removed because we noticed that they would not fit in such a "rhythm-game" or that they would be hard to implement in an appropriate way.

2 Tasks of the Team Members

At first we split the project in 2 big tasks: the song analysis and the unity part.

So at the beginning Jan-Eric Schober was responsible for researching information tot he song analysis and Chris Loch was responsible fort he whole unity part oft he game.
Towards the end we were both working on those 2 parts together.

Overall we can say that we slit the tasks properly and that the workload for both of us was balanced.

3 Used Tools

Here are some of the most important tools we used for our project:

Unity3D engine

Since Unity3D is a very powerful game engine and has a very good documentation which is easy to understand, we easily decided to use it.

Visual Studio

Visual Studio was used to implement the code for the C# scripts which are used in unity.

Team Speak 3 & Team Viewer

Remote meetings were held via Team Speak 3 and Team Viewer.

Adobe Photoshop
We used Photoshop to create the mockups and some of the ingame art.

Gogs
We used Gogs as our VCS (version control system) which is powered by Git.

Whatsapp
We used it to plan meetings and to communicate as fast as possible with each other. Since it is a mobile application and you can use it on the go, it was a good choice because we both were available at almost any time over Whatsapp.

4 What Worked Well

Teamwork
The teamwork between us was superb. It was based on a great communication and when a problem occured we helped each other out to solve it as soon as possible.

Getting started with Unity3D
Getting started with unity worked very well because of the excellent documentation and lots of tutorials about specific problems you will encounter while developing a game.

5 What Worked Not so Well

Finding appropriate APIs
It was very hard to find an appropriate API for the song analysis or respectively the beat detection. There were plenty of good APIs like nAudio or FMOD but they did not fit very well for our purpose.

6 Summary

In conclusion we can say that we have learned a lot throughout the educational process and that our team worked fine. It was definitely a good and fun experience and it can be recommended without any remorse.