Osman Sinan **Emiroğlu**

www.sinanemiroglu.github.io
o.sinan.emiroglu@gmail.com

Profile

Since 2016, I have had many chances to work on various academic and industrial projects related to digital games. I always feel lucky as my undergraduate and graduate studies put game design and development at the center of their curriculum. Up to now, I have learned to tackle different kinds of challenges in digital game development, specifically programming.

In 2018, I moved to Estonia to study digital games at Tallinn University. Besides, I have been involved in some professional projects as a programmer, including iOS, Android, Windows, WebGL, and Oculus VR platforms. I expanded my horizons by learning different areas in game development such as artificial intelligence, tweening & complex animation systems, etc. Besides, I had an opportunity to work in Helsinki within the scope of Erasmus+. Finally, I wrote my master's thesis on a simulation that I developed utilizing the Artificial Bee Colony algorithm to collect resources in RTS-like games.

I care about writing performance-friendly clean code and I always try to write the most robust and scalable code. I'm sure most companies or studios will always be sensitive to these matters at any stage of their development pipeline.

Technical Skills

- Unity & C# (Experienced)
- UE4 & Blueprint, C++ (Amateur)
- Git / GitHub / GitLab
- Oculus VR
- Visual Studio & VS Code
- Photoshop & Illustrator
- HTML & CSS
- JavaScript & TypeScript
- NodeJS & ExpressJS

Experience

Tallinn University
Unity & C# Developer
Mar 20 – Dec 20 @Tallinn

Methodyca is an adventure-based educational game to teach research methods to bachelor/master students. I was responsible to develop four mini-games. For instance, Questioniser is a 2D card game for learning about questionnaires as a data collection method. This game has a complex data structures, so I put extra effort to architect codes to make them generic and scalable.

Critical Charm Oy.
Game Programmer
Sep 19 – Nov 19 @Helsinki

A Giant Problem is a fantasy VR game where players protect their people from the king's troops as a giant. So, everything reacts to player's action. As a programmer, I contributed a couple of things like new score, pooling, animation event, and character movement system, state machine & AI.

Medit Creative Lab.
Unity & C# Developer
Nov 18 – Feb 19 @Tallinn

Invisible Narva is an iOS tablet project having an extensive background with lots of academics, students, animators, 2D, and 3D artists involved even if I was the only programmer. The prototype was completed approximately in four months. Although the final product will be designed as real world-scale AR, touch joysticks were added to move around the city. Therefore, nobody needs to go to Narva to play the game.

GES Tech. & Exhibition
Unity & C# Intern
Jun 17 – Aug 17 @Istanbul

In my first industrial experience, I developed two VR projects and a Windows touch screen project for exhibition purpose in science centers. Since projects had cinematic purposes, VR games did not require any player controls. Yet it was an immersive experience to create them.

Education

Master's Degree (2018 – 2020)

Digital Learning Games @Tallinn University, Estonia

Bachelor's Degree (2011 – 2018)

Computer Educational Technology @Bogazici University, Turkey

Hobbies & Interests

In my free time, I usually check different technologies and learn new things to provide me with more professional knowledge. Game programming made me to be interested in low level programming because I want to code more performance-friendly programs. So, I took Nand2Tetris I&II courses to learn how computers, programming languages, and compilers are working. Also, I was interested in different game engines and frameworks like UE4, Defold, Raylib, etc., so I tried them to figure out what kind of solutions they offer to develop better games for different platforms. Gradually, I realized object-oriented programming may not the best paradigm for game performance even if it's easier to code, so I started learning data-oriented programming and the ECS design pattern.