

Resume Tim Sabo

- Age: 24 years old
- Address: Tarwestraat 42
- City: 8400 Oostende, Belgium
- Tel: +32 (0)484 798 611
- Email: Tim.tili.sabo@gmail.com
- Portfolio: <http://timsabo.morior.net>

Education

- Bachelor in Multimedia & Communication Technology (3 years)
- Bachelor in Digital Arts & Entertainment (3 years)

Computer skills (please see page 2)

- Programming in C# (5 years experience) and C++ (3 years experience)
- OpenGL & DirectX
- Shader programming in GLSL, HLSL and fixed pipeline iOS shaders
- Modeling (3DS Max)
- Texturing
- Game design
- Scripting
- Photoshop, Flash, Illustrator, After Effects
- Java, Javascript, VB.Net, C#, PHP, ASP.net, JSF, Ruby
- ISS, Apache, SQLServer, MySql

Awards

- Second place for the DAE awards in the 2D game category 2008-2009
- Best Art Design Intel Level Up 2010 (Pomana)
- Semi-Finalist Imagine Cup 2010 (Pomana)

Language Skills

- Dutch, English and French

Work Experience

- Freelance application development summer 2010 (2 Months)
- Internship at [Sandbox](#) (UCLAN, Preston, UK) (3 Months)
- Visual Effect Designer for The Shadow Sun
- Intern at Ossian Studios (4.5 Months) <http://www.ossianstudios.com/>

Personal Interests

- Replacement responsible in the Ostend board of youth affairs
- Reading
- Hurdy-Gurdy

Programming – Tim Sabo

AI/Gameplay

- Implementation of 2D grid based path finding system (A*)
- Implementation of mesh based path finding system (A*)
 - Ported to unity (C++ to C#)
 - Heavily optimized for Ossian's prototype game
- Real time navigation mesh generation
 - Based upon the Delaunay triangulation, again heavily optimized to work real time in Unity
- 2D polygon based line of sight calculation, uses the above mesh generation algorithm to create a line of sight mesh (based on the works of <https://code.google.com/p/straightedge/>)
 - The code was a bit unreadable so I wrote my own implementation.
- Implementation of a steering behavior model (boids) <http://www.red3d.com/cwr/boids/>
- Collision avoidance AI for CSpace

Engines/Games

- Remake of Baldur's Gate
 - Using Win32 and GDI
 - A star path finding using a grid
- PirateBuster
 - Defense game
 - Made in DirectX 9
 - Optimized to draw way too much stuff
 - Custom physics implementation
- RustEngine, a DirectX 10 engine
 - Written from scratch
 - Uses custom HLSL shaders
 - With custom biped animation system
 - Real-time shadow mapping
 - Uses Geometry shaders
 - Real-time mesh generation (roads) using splines
 - Custom particle system
- CSpace, an OpenGL game/engine
 - Written from scratch
 - Uses custom GLSL shaders
 - Custom font rendering with caching
 - Custom particle system for a space RTS
 - Procedural generation of planets/asteroids/solar systems (including materials)
 - Optimized engine to draw multiples of the same mesh. (Ships, asteroids...)