

Nils Mathieu



I'm a motivated and self-driven computer science student yearning to start a software engineering career. I like writing stories, producing music and developing cool things in Rust or in C. I like to create complex systems and write pretty lines of code. I make sure to understand everything I do and love explaining it to others.

2019 • Baccalauréat

(+33) 7 81 88 95 91



2019 - 2021 • Classe Préparatoire (MP)

nils.mathieu.contact@gmail.com



2021 - 2022 • School 42

<https://nils-mathieu.fr/cv/>



Dynamic

I can quickly adapt my work and method to new and unexpected constraints within the development of a complex system. This includes properly documenting the code to make it easier for others to modify it with no fear.

I develop systems knowing that they will be modified in the future (new features, bug fixes). I know when, why, and how to correctly add complexity to a code base.

Communicative

I can communicate with other members of the team, including non-developers, about what I do and what I need. I can explain fairly difficult concepts in plain English and more generally, I like to teach what I know to others.

I'm a native French speaker with a proficiency in English.

Soft Skills

Independent

I know the objective, target, and scope of the project I am working on and am able to take pragmatic decisions in consequence. I have the self-confidence to take action, but also know when to ask when needed.

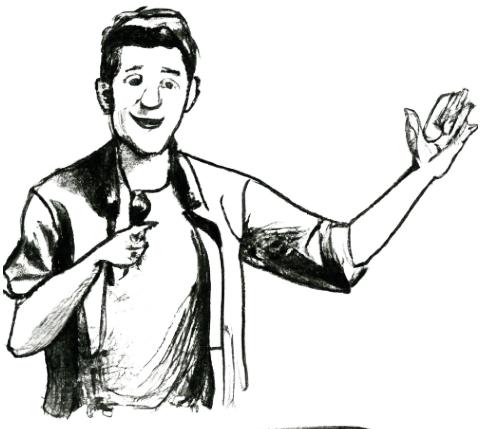
Quick Learner

I can quickly learn new technologies, languages or tools. I go as deep as I can to make my understanding of said technology as efficient and correct as possible.

I love learning new things and perfecting what I already know.

Rigorous

I know what I do, and do what I know. Writing software is both a matter of getting things done, and doing those things correctly. I always make sure that the solution I come up with is the correct one, and not necessarily the easiest one.



Rust

The Rust programming language has been my favorite thing since I started using it in 2018. I consider myself an expert in the language, and am able to write high-performance applications and tools with it. Most notably: an ECS-based game engine and an audio sequencing tool.

C#

C# will always have a special place in my heart, as it was the first language I used to actually *do* things. I know the language fairly well and have used it mainly to create games and standalone applications.

HTML/CSS/JS/TS

I know how to create static web pages with HTML, CSS and JavaScript. This very resume has been crafted manually with those tools. I have some familiarity with React. I have created a simple multiplayer game powered by websocket using those tools.

Optimization

I've always liked optimizing systems, taking every bit of performance I possibly can out of it.

I have a good understanding of how a CPU works, and how my code gets compiled into machine instructions. I think about cache misses, about SIMD, and know how to correctly compare two functions using benchmarks and profiling tools.

Embedded Systems

I have experience writing and deploying code to bare metal targets such as older consoles or microcontrollers. I also have written my own (pretty buggy) bootloader in Rust for a (very simple and bare bones) operating system project.

Mathematics

I have knowledge in various fields of Mathematics including analysis, probabilities, combinatorics, and linear algebra. My studies in *classes préparatoires* taught me how to correctly articulate mathematical proofs.

Specifically, I use linear algebra extensively every time I work on a game from scratch, or more generally, anytime I write a real-time application.

C/C++

I am proficient in the C and C++ programming languages. I have used both languages equally during my time at School 42 to create video games, shells, 3D renderers, web servers and various libraries.

Vulkan

I know the outline of the Vulkan API, and know how a rendering pipeline works, from the vertex shader to surface presentation. I have written a simple 3D renderer using this API for my own game engine (in Rust). This includes writing custom shader programs in GLSL or WSL and compiling them into SPIR-V modules.

Known Tools



Linux



Discord



Docker



Notion



Git



GitHub

Signal Processing

I have written my own simple audio sequencer, including custom effects such as low-pass and high-pass filters, reverb and delay.

More generally, I have some experience with signal processing and have knowledge about the mathematics and physics behind it.

Real-time Networking

I have experience developing a real-time multiplayer server (and clients to go with it), from scratch, working directly over the TCP protocol. That includes handshaking, authentication and compression of data frames.

