



Hello, my name is [Vlad Faust](#).

I am a jack of anything IT, master of some; see my [CV](#).

You can contact me via [X](#), [GitHub](#) or [Telegram](#), or e-mail me directly at [hey at vladfaust.com](mailto:hey@vladfaust.com).

Curriculum vitae

Throughout my 10+ years of experience in Information Technology, **I've tried everything** from compiling COFF after LLIR lowering to designing a resilient ultra-scalable fleet of FAAS workers, from Substrate and IPFS to event-driven stock price prediction.

Comprehending a technology is like surfing on the waves of technology trends, on the surface of the greatest whirlpool of singularity. Comprehension means fundamental, decomposed knowledge enough to build a solution at **any scale**.

For me, it is not about thoroughly learning API of yet another web framework, it is instead getting hands on the set of tooling a technology offers, so that I may choose the **best framework suitable for the problem**. If I were to compile my 10+ years of experience into one line, that would be the following:

Right tooling is paramount.

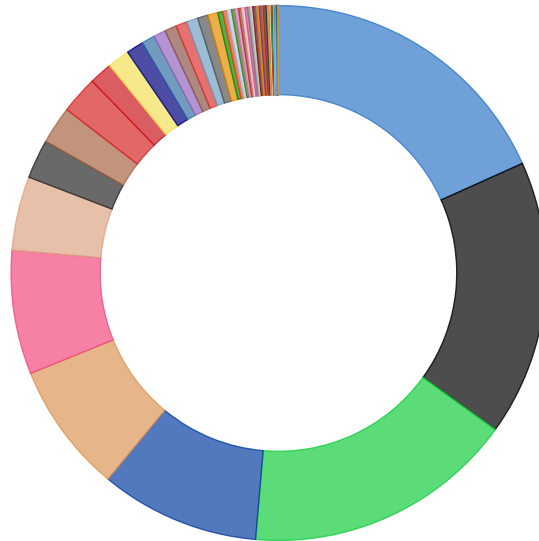
Here is an incomplete list of technologies I've had experience with, in the order of encounter, starting back in 2013: Pascal, [vJass](#), cJass, C, C++, Java, git, [LibGDX](#), PHP, [Unity](#), C#, Ruby, Telegram, [Heroku](#), [Dokku](#), SQLite, PostgreSQL, Redis, Docker, JavaScript, Gulp, Grunt, Ruby On Rails, [Hanami](#), [Roda](#), [Sequel](#), [Crystal](#), LLVM, Kubernetes, [Firecracker](#), [Tiny Core Linux](#), [buildpacks.io](#), compiler development, machine learning, [Tensorflow](#), VueJS, TypeScript, [Deno](#), [Zig](#), [PEG](#), Rust, EVM, Solidity, [Substrate](#), [IPFS](#), [ZKP](#), ChatGPT, llama.cpp, Godot, [tRPC](#), [Zod](#), [Tanstack](#)...

- **2013:** It all began with Warcraft® III™ maps, such as [UFS Arena](#). I did all the **coding** (cJass, a C-like domain-specific language) and **design**, and my maps gained some popularity in the Russian-speaking community.
- **2014:** First steps in commercial product development: [Anogram](#), a social anonymous service project. I did the **design and backend coding** in PHP. Unfortunately, the trend had quickly deteriorated.
- **2015:**

- In 2015, I released an Android game called [Jumpin Sweeties](#), written with **LibGDX**. My family and friends liked it, but it didn't gain much traction.
- With **Ruby on Rails**, I created a Bitcoin service for SMS number confirmations based on Google Voice, and after a couple of months sold it to a foreign customer for a good price. This marked the beginning of my journey as a **solo entrepreneur**.
- **2016:**
 - This was the year of **Telegram bots** for me. First, I created [@profitrobot](#), which climbed pretty high in the bot store rating. The bot allowed users to earn lunch money by completing CPA tasks and watching advertisements. Profitrobot became [quite popular](#), and I managed to sell it to another entrepreneur.
 - In parallel I worked on [@soundememesbot](#), which allowed its users to post sound memes in chats. Long story short, the bot became [massively popular all around the world](#), but I did struggle to monetize it. Required moderation efforts made me shut it down.
- **2017:** Not all the projects are destined to become viable, such that was [CashbackBot](#), a Telegram bot for **cashback services**. I've spent a lot of time building it, but the project was never launched, nor it was sold.
- **2018:** This year marked the beginning of my two-years journey in the [Crystal programming language](#) ecosystem. I loved being a part of [open-source](#) (check out my [GitHub!](#)). I created many libraries, including [web framework](#) and [ORM](#), and even **contributed into the language** itself. Also check out my [article](#) comparing Crystal to Stripe's Sorbet.
- **2020:** By 2020, I was fed with all the imperfections of the Crystal ecosystem, and decided to move and build [my own programming language](#), Onyx. Oh, that was a ride. Long story short, too much for a single person. Yet, **an enormous amount of computer science experience** gained. Just look at the plethora of Onyx compilers I was working on: [C++](#), [Rust](#) and even [TypeScript with Zig!](#) The [System Programming in 2k20](#) article is still one of the best I've written.
- **2021:** [Senior Crystal developer](#) at [NeuraLegion](#), an Israeli web security company. Nothing fancy, just some good old JSON APIs.
- **2022:** I was employed by [Byzantine.Solutions](#) on the role of a [Technical Due Diligence Officer](#). I inspected and contributed to some of the biggest crypto projects of that time. I also did some crypto-related projects of my own (see below).
- **2023:** The epiphany of AI had occurred, my mind has cracked, and I've lost the sense of what is past and what is future. Yet I managed to work on some private AI projects at the role of a [full-stack+prompt engineer](#).

Check out my [profile at Wakatime](#); since I've began tracking my time in 2018, I've coded over 7000 hours, resulting in 1000 hours a year average. Take a look at this glorious chart of my favorite languages:

TypeScript	1,421 hrs 53 mins	(18.28%)
Crystal	1,296 hrs 9 mins	(16.67%)
Vue.js	1,276 hrs 46 mins	(16.42%)
Markdown	748 hrs 54 mins	(9.63%)
Other	613 hrs 9 mins	(7.88%)
C++	582 hrs 35 mins	(7.49%)
Rust	347 hrs 33 mins	(4.47%)
JSON	183 hrs 18 mins	(2.36%)
Solidity	176 hrs 46 mins	(2.27%)
AsciiDoc	176 hrs 26 mins	(2.27%)
YAML	107 hrs 58 mins	(1.39%)
JavaScript	104 hrs	(1.34%)
Lua	80 hrs 30 mins	(1.04%)
Python	61 hrs 24 mins	(0.79%)
Docker	57 hrs 35 mins	(0.74%)
Bash	55 hrs 14 mins	(0.71%)
CMake	53 hrs 22 mins	(0.69%)
AsciiDoc	53 hrs 4 mins	(0.68%)
C	49 hrs 57 mins	(0.64%)
SQL	46 hrs 24 mins	(0.60%)
C#	23 hrs 9 mins	(0.30%)
HTML	20 hrs 50 mins	(0.27%)
Text	19 hrs 57 mins	(0.26%)
Makefile	16 hrs 39 mins	(0.21%)
GDScript3	16 hrs 36 mins	(0.21%)
TOML	16 hrs 29 mins	(0.21%)



Currently I'm working on private **AI character simulation** projects. My best stack is prompt engineering, custom AI model deployment, Rust, C++, Typescript, VueJS, NodeJS, tRPC, Zod and Tanstack.

See my blog posts (especially the [personal introduction](#)) and projects below, check my [GitHub profile](#), find my contacts in the header, and feel free to reach out if you want to work with me.

Sincerely,
Vlad.

Recent blog posts

- Tue Apr 12 2022 [The game of life](#)
- Thu Aug 20 2020 [The Onyx Programming Language](#)
- Sun Aug 16 2020 [System Programming in 2k20](#)
- Fri Aug 07 2020 [Hello, World!](#)

Project list

Singularity tech (2022-present)

Behold, the Deus ex Machina blooms, like a lotus from the murky depths.

[DeRouter](#)

[AISTories](#)

Decentralized, uncensored open-source API marketplace.

Open source Rust P2P

The original source-available AI character simulation platform.

Open source Commercial AI Crypto NFTs
TypeScript VueJS NodeJS trpc zod
Solidity ethers

Kawaii

OnlyFans, but for Web3.

Open source Commercial TypeScript
Ethereum

Offchain Café EVM

A self-hosted indexing service for Ethereum blockchain, exposing a GraphQL API.

Open source Crypto EVM Solidity
TypeScript GraphQL

IPNFT

An on-chain, digital proof of authorship for an IPFS CID, tailored to existing NFT standards.

Open source Crypto IPFS EVM Solidity

Fancy Contracts

A collection of EVM smart contracts for the Fancy Software apps ecosystem.

Open source Crypto EVM Solidity

Onyx, the language (2020-2022)

Onyx is a novel programming language I was working on.

[NXSF.org](#)

The Onyx Software Foundation website ([source](#)).

Open source

[Onyx standard](#)

An attempt to standardize the Onyx language ([source](#)).

Open source

[Phoenix](#)

An attempt to implement an Onyx language compiler in C++.

Open source C++

[Onyx in Rust](#)

An attempt to implement an Onyx language compiler in Rust.

Open source Rust

[Onyx in Typescript](#)

An attempt to implement an Onyx language compiler in TypeScript + Zig.

Open source Zig TypeScript

Crystal-related projects (2018-2020)

[Crystal](#) is a Ruby-inspired programming language, but compiled.

[Onyx framework](#) (2020)

A web framework with [HTTP](#), [SQL ORM](#) and [EDA](#) modules, fancy website and rich documentation.

Open source Product design Crystal SQL
HTTP Event-driven architecture VueJS SASS

[Crystal Jobs](#) (2018)

A now defunct Crystal jobs board ([frontend preview](#)).

Open source Product design Crystal VueJS
SASS

[CrystalWorld](#) (2019)

A [RealWorld](#) implementation in Crystal and Onyx the web framework.

Open source Crystal SQLite

Crystal shards (2018-2020)

A *shard* is a Crystal code package.

[TimeFormat](#)

Time spans formatting made simple.

Open source Crystal

[I18n](#)

Internationalization shard.

Open source Crystal

[HTTP::Multiserver](#)

Mount multiple web applications.

Open source Crystal HTTP

[Tarantool](#)

A [Tarantool](#) database driver.

Open source Crystal TCP

[Cake-Bake](#)

Bake Cakefile into native Crystal code.

Open source Crystal

[Callbacks](#)

Expressive callbacks module.

Open source Crystal

[Validations](#)

Validations module.

Open source Crystal

[MiniRedis](#)

A light-weight low-level Redis client.

Open source Crystal TCP

[HTTP::Params::Serializable](#)

The HTTP params parsing module.

Open source Crystal

[Migrate](#)

A database migration solution.

Open source Crystal SQL

[Stripe](#)

Stripe API wrapper.

[Tele](#)

Open source Crystal API wrapper

A convenient Telegram Bot framework.

Open source Crystal Telegram bot

[Tele::Broadcast](#)

Broadcasting for Tele.

Open source Crystal Telegram bot Redis

Ruby gems (2015-2017)

[Jbuilder::JsonApi](#)

Jbuilder meets jsonapi.org specifications.

Open source Ruby

[AssetPipeline](#)

Add Sprockets based Asset Pipeline to your Ruby applications.

Open source Ruby on Rails

[Bitcoinpay](#)

Bitcoinpay.com API client for Ruby.

Open source Ruby API wrapper Crypto

[Blockchain::Api](#)

A blockchain.info API wrapper.

Open source Ruby API wrapper Crypto

Other projects

[Jumpin Sweeties \(2015\)](#)

A casual mobile game.

GameDev Java LibGDX



[Parrot Farm \(2016\)](#)

A showcase project to demonstrate my skills at those times, hosted on Heroku.

Open source Ruby on Rails AngularJS

[Coinpricebet \(2018\)](#)

A Stellar-based betting on crypto coin prices, also [this](#) version.

Product design Crypto VueJS

[AssetFD \(2018\)](#)

An unfinished decentralized exchange interface based on Stellar.

Crypto VueJS

[Soundmemes Bot \(2016-2018\)](#)

A now defunct Telegram bot to post sound memes in chats.

Open source Product design Telegram bot

Ruby Crystal PostgreSQL Redis

[Cashback Bot \(2017\)](#)

A now defunct Telegram bot to earn cashback from purchases.

Product design Telegram bot Ruby GraphQL

API wrapper PostgreSQL Redis

[Expense Manager 2](#) (2015)

A collaborative attempt to improve an existing app.

Product design Java Android

[Cycler](#) (2014)

An attempt to create a sleep cycle Android application.

Product design Java Android

[Anogram](#) (2014)

A naive attempt to create an anonymous social network.

Product design iOS PHP

[UFS Arena](#) (2012)

A Warcraft III™ custom map ([source](#)).

Product design GameDev cJASS