

# Jonatan Kłosko

jonatanklosko@gmail.com · github.com/jonatanklosko

## About

An open source developer from Poland, working at Dashbit. Formerly a member of the World Cube Association Software Team and a speedcuber. Enjoys simplicity, both in software, and in life.

## Public education

*2018 – 2022*

**AGH University of Science and Technology, Kraków, Poland**

BS in Computer Science

Faculty of Computer Science, Electronics and Telecommunications

Thesis: “A platform for testing multi-population evolutionary algorithms using the BEAM virtual machine”

## Work experience

*2021 – Present*

**Dashbit, Software Engineer**

Dashbit is a company with the goal of advancing the Elixir ecosystem through continuous adoption and sustainable open source development. In this spirit, my work has been focused mostly on open source projects related to the Elixir ecosystem, most notably Livebook. An essential part of the job is collaboration with the team, open source maintainers and other members of the community through discussions and code reviews. Since 2025, I have been a part of Dashbit’s consulting team (Elixir Development Subscription), helping clients ship Elixir applications confidently through code reviews and discussions.

*2016 – 2025*

**World Cube Association, Software Team member**

The WCA is a nonprofit organization governing Rubik's Cube competitions, running thanks to numerous volunteers. As one of such, throughout the years I have been working on several software projects affecting the community worldwide. Being on the team has been a great opportunity to collaborate and exchange experience with other developers. Apart from purely technical tasks, the job also involved communicating with other (often non-technical) teams to provide them with information and discuss their ideas and needs.

# Projects

*2021 – Present*

## **Livebook**

A web application for writing interactive and collaborative code notebooks for Elixir. Livebook focuses on reproducible notebooks with readable source, it supports real-time collaborative editing, code completion, inline documentation and provides interactive components, such as charts, tables and inputs. On top of that, notebooks can be deployed as apps, which enables users to automate workflows and build internal tools. Built with Elixir using Phoenix LiveView.

*2025 – Present*

## **Tidewave**

A coding agent for full-stack web app development, focused on vertical integration across multiple frameworks. Tidewave runs on the user's machine and it embeds the user's web application, integrating with their existing development setup. The agent has access to the embedded page to programmatically validate and debug the changes it makes. It can also interact with the server and the database via framework-specific tools. Tidewave understands the web page and provides the agent with relevant source locations to set it on the right track. Tidewave can be used on top of general-purpose coding agents via the ACP protocol, composing with the user's existing tooling and focusing on excelling at specific areas. Built with React and Elixir.

*2022 – Present*

## **Bumblebee**

A library providing state-of-the-art neural network models, allowing anyone to download and perform machine learning tasks with few lines of code. The library includes implementation of individual models and streamlines the process of loading pre-trained parameters from Hugging Face Hub. Furthermore, it provides end-to-end pipelines for specific tasks, such as image classification, text generation, speech transcription and many more. Those pipelines can be integrated directly into an Elixir application and run across multiple machines and GPUs. Built with Elixir using the Axon neural network library.

*2025*

## **Pythonx**

A library providing a Python interpreter embedded in Elixir, in the same memory space. The library allows the user to evaluate Python code and conveniently convert between Python and Elixir data structures. It transparently installs Python and dependencies specified by the user to ensure reproducibility across environments. The library is implemented using Erlang NIFs (Native Implemented Functions). Built with C++ and Elixir.

*2025*

### **Fine**

A library streamlining C++ interoperability in Elixir. The library is designed as an API complementary to the existing C API provided by Erlang (`erl_nif`), enhancing the developer experience when implementing NIFs in C++. Notably, the library automatically converts between Elixir and C++ data types, based on function signatures. It also offers smart pointers to safely manage C++ objects living across multiple NIF invocations. Built with C++.

*2021*

### **Tree-sitter Elixir**

An Elixir grammar definition for the Tree-sitter incremental parsing system. This project is used in production by GitHub for source code highlighting and code navigation.

*2021*

### **Meow**

A framework for composing and running multi-population evolutionary algorithms with support for GPU and distributed computing. Evolutionary algorithms are a broad category of biology-inspired numerical methods for solving optimisation problems. The primary motivation behind Meow was applying the tensor computational model to evolutionary algorithms to enable cross-platform hardware acceleration. This project was a part of my bachelor thesis and a subsequent paper publication. Built with Elixir using the Nx tensor library.

*2019 – 2025*

### **WCA Live**

A web application used at WCA competitions for entering scores, displaying live rankings and performing administrative tasks. It helps to run hundreds of competitions in dozens of countries a year. The application consists of a GraphQL API built in Elixir with a PostgreSQL database and a web client built in React. It automatically scales to multiple instances under heavier traffic and employs Elixir distribution capabilities to transparently use all instances as a single system. All of the infrastructure is expressed using Terraform.

*2016 – 2020*

### **WCA website**

The first WCA project I got involved in, quickly becoming one of the most involved contributors. The website is the primary source of information about competitions, competitor profiles, records and rankings. It provides many administrative tools necessary for WCA teams to perform their tasks. My activity included building new features, fixing bugs and server outages, code reviews and discussions. Developed mostly using Ruby on Rails and MySQL.

## Talks

2025

### **ElixirConf Eu**

Title: Livebook behind the cover

2023

### **ElixirConf Eu**

Title: Powerful machine learning at your fingertips

## Languages

Native Polish, Advanced English