

Silvano Cerza — Software Engineer

I enjoy working on different platforms, technologies and solving tough problems. I consider myself a generalist and am never scared to tackle a new challenge using unknown languages or technology. It's something that I aim for since I believe it's the best way to learn something new, and that every problem requires the right tool to be solved the best way possible. Some of the stuff I've done include CLIs in Go, web apps and Open Source frameworks in Python, desktop and mobile apps in Qt (C++ framework), Electron applications in Typescript and Clojure/Script, enhance and evolve CI/CD infrastructure on CircleCI and GitHub Workflows.

 silvanocerza.com

 [/users/4156301](https://github.com/users/4156301)

 silvanocerza@gmail.com

 [/silvanocerza](https://github.com/silvanocerza)

 [/in/silvanocerza](https://in.linkedin.com/in/silvanocerza)

Work Experience

Deepset

 deepset.ai

Jan 2023 – Present

I'm part of the Core Team and one of the main developer of Haystack and related projects.

In my day to day job I enhance release, testing, and CI processes by making them more streamlined and faster to run.

Part of my time is also dedicated to all the stuff that managing an OS project like Haystack entails, like: triaging community issues, giving support in the official Discord.

Python

GitHub Actions

Arduino

 arduino.cc

Jul 2020 – Mar 2022

I worked mainly on the ArduinoCLI, implementing new features, fixing bugs and writing tests.

I also took part on other minor supporting projects for the ArduinoCLI like external libraries, GitHub Workflows Actions and similar things.

Most of the times I managed the release process of the ArduinoCLI, from triggering the build process to communicating with marketing for social communications.

Golang

Python

TypeScript

GitHub Actions

Pitch

 pitch.com

Apr 2022 – Sep 2022

I was part of the DevOps team and focused on CI, release management and developers experience.

Most of my time was spent enhancing and evolving our CircleCI infrastructure, making the release process leaner and easier to understand, building scripts and tools to enhance the day to day job of other developers.

I also contributed in small parts to the main product application writing Clojure and ClojureScript.

Clojure

ClojureScript

Python

CircleCI

Molecular Horizon

 molhorizon.it

Jun 2019 – Jul 2020

C++

Qt

QML

Qt Quick

GitHub Actions

Evonove

 evonove.it

Feb 2015 – May 2019

Python

Django

C++

Qt

QML

Qt Quick

Work Projects

Haystack

 [/deepset-ai/haystack](https://github.com/deepset-ai/haystack)

Haystack is an open source NLP Python framework to interact with your data using Transformer models and LLMs (GPT-4, ChatGPT and alike).

I plan and implement new features, fix existing bugs, enhance legacy parts, write tests for untested parts of the code, and interact with community on its Discord.

Python

Arduino CLI

 [/arduino/arduino-cli](https://github.com/arduino/arduino-cli)

Arduino CLI is a command line tool written in Go, it's used to manage and install libraries, the tools necessary to compile and upload to different Arduino boards, and several other things.

Among its features it's also the backbone of the new Arduino IDE, using a gRPC interface to communicate with each other.

Golang

Python

gRPC

Protobuf

Arduino IDE 2

[/arduino/arduino-ide](#)

Arduino IDE 2 is the new version of the famous Arduino IDE used by millions of makers around the world.

The new version has been rewritten from scratch using Electron and the Theia framework. It uses the Arduino CLI internally to implement most of its features.

I implemented some minor feature, fix existing bugs, and handled integration of new functionalities from the Arduino CLI.

[Node](#) [Electron](#) [Theia](#) [Typescript](#) [React](#)

Qnite

[/evonove/qnite/](#)

An open source library to create charts declaratively in QML.

I contributed by adding several features like charts zoom and bug fixing.

[C++](#) [Qt](#) [QML](#) [Qt Quick](#)

Kibitzer

[moldiscovery.com/software/moka/](#)

Kibitzer is a desktop app to train computational models used to predict molecules' pKa values.

I integrated several C and C++ libraries in a multi threaded model to parallelize their execution to speed up the statistical computation for a better user experience.

[C](#) [C++](#) [Qt](#) [QML](#) [Qt Quick](#)

MKM SDK

[/evonove/mkm-sdk/](#)

An open source SDK for Magic Card Market, uses reflection to dinamically resolve the API endpoints.

[Python](#) [OAuth](#)

Talks

Monitoring Open Source Success in Arduino

Oct 2021

[/watch](#)

In this talk, I tell the story of how my team started using Datadog in an unusual way to monitor the success of one of the most important of its Open Source projects, the Arduino CLI, by gathering download information from different sources and monitoring build failures. I also explore some of the next steps we can take to gain insight into our other projects.

Personal Projects

Yellow Rats

[/silvanocerza/yellow-rats/](#)

A simple GUI written in C++ using OpenCV and ZBar, to scan and catalog comics by number and quality.

[C++](#) [OpenCV](#) [ZBar](#) [CMake](#)

Cute Code Accessor

[/silvanocerza/CuteCodeAccess](#)

Cute Code Accessor is a plugin for Unreal Engine that enables the use of Qt Creator as its IDE.

[C++](#) [Unreal Engine](#)

Bank Reader

[/silvanocerza/bank-reader](#)

A scraper for my personal bank account, used to monitor how I spend my money.

[Python](#) [Django](#) [Scrapy](#) [Lua](#) [Vue.js](#)

Datadog Agent

[/DataDog/datadog-agent](#)

Datadog Agent is an open source monitoring tool. I contributed to Six, a C++ wrapper around the CPython API supporting multiple versions of Python that can be used by other languages like Go.

[C++](#) [C](#) [CPython](#) [Go](#)

Spectrum

[/silvanocerza/spectrum](#)

An highly customizable Hugo theme based on the CSS framework Bulma.

[Sass](#) [CSS](#) [Bulma](#) [Hugo](#)

This resume

[/silvanocerza/resume](#)

This resume has been completely written in HTML and Scss. Uses a small Dart script and dart-sass to compile and minimize Scss sources.

[HTML](#) [Dart](#) [Sass](#) [CSS](#)