Michał Kukowski

Address: Wrocław · · Contact: michal.kukowski@kukossw.com ·

About me

Currently, I am deeply focused on enhancing my soft skills and team management abilities. I have 3 years of experience as a Scrum Master. During that time, I managed and coached the team, organized workflows, and performed some Product Owner responsibilities such as backlog prioritization, KPI measurement, and reporting to stakeholders.

I have 7 years of experience as a C Developer (5 as a Tech Lead): 6 years in Telecommunication and over 1 year in the Linux Kernel. I primarily focus on optimization, code profiling, and tailoring wellknown algorithms for client architectures. (Feel free to contact me if you'd like to understand why a direct implementation of the insertion sort, as described in Cormen's book, is not advisable.)

In the last 3 years, I have focused on designing new code in C and incorporating some OOP principles. As a Technical Leader, I have developed two training programs for young developers.

Professional experience

TietoEVRY

Scrum Master

October 2020 – present

- Tools and technologies: Jira, JavaScript Jira scripts, JQL
- Scrum activities: Daily, Refinement, Review, Planning, Scrum Poker sessions, KPI's monitoring, developing some scripts using JavaScript to collect and process data from Jira
- Other activities: Improving process, supporting Product Owners and stakeholders, agile coaching

TietoEVRY

C Developer

October 2017 – present

- Tools and technologies: C99, git, Gerrit, Jira, bash, gdb
- Developing LTE/5G UPCDL Baseband SW in C99 (customer's DSP architecture)
- Multithreading algorithms (i.e postponing the synchronization)
- Technical leader since 2018 (mentoring teammates, preparing training program for young developers)

Nokia

C Developer

July 2016 – October 2017

- Tools and technologies: C99, gnuC, gdb, git, Gerrit, Jira, bash, Makefile, Yocto
- Linux kernel (not only drivers) and Uboot developing
- Code profiling (cycles and memory optimizations) for ARM Cortex M

Technical Skills

C99 **** GnuC ★★★★

SCRUM ★★★★ Testing C code $\star\star\star\star\star$

Jira ★★★

Nexus ★ SAFE ★ C++11/14 ★★

JavaScript ★★ GTest ★★

I agree for the Personal Data Administrator to share my personal data, in the scope of data contained in the CV and disclosed in social media confirming professional qualifications, necessary to conduct recruitment processes. Thanks to this agreement we will be able to pass on your data to other companies from the Altimi Group, ie: Altimi Sp. z o.o. SK, Altimi Sp. z o.o., Softiti Sp. z o.o. and Clients for whom we conduct recruitment processes.

EDUCATION

PhD - Computer Science, Wrocław University of Science and Technology	2019 – present
MSc - Algorithmics, FFPT, Wrocław University of Science and Technology	2018 - 2019
BEng - Computer Science, FFPT, Wrocław University of Science and Technolog	y 2015 – 2018

LANGUAGES

Polish	native
English	B2

PRIVATE PROJECTS

- K(Kukos) lib set (github) set of new useful mini libraries like (macros), (logger), (verbose assert), (unit test framework). Code has been written in C99, C17, Cgnu17. Code has been written in C with a lot of Unit tests and functional tests in C using a simple set of macros from CLS. The code has been checked also by Valgrind.
- Compiler (github) is a project from studies. We had simply custom architecture, simply language with gramma based on C/Pascal and we had to write a compiler that produces ASM mnemonics for custom arch from normal code. Code has been written in C, using CLS, Flex, and Bison. E2E tests have been reported in C as calls of some Linux commands.
- Training materials (github code with a short video(videos are in my private repository) materials about C99, C macros, Big O notation, optimization and algorithms.
- Dev training program (github) how to learn programming? How to improve technical skills? Just keep practicing! I created a list of some tasks which I analyzed and developed (not all) with junior developers. Tasks have 3 categories, benchmarks: when insort is better than quicksort?, algorithms: how to solve Vertex Cover problem?, development: write simply printf, shell, memcpy using avx.
- Data base system simulator(github) Still in progress. The code is written in C++14. The main goal of this project is to create a DBMS simulator to check different indexes and performance on different memories like SSD, PCM, and flash dies.

Papers

- W. Macyna, M. Kukowski, Adaptive merging on phase change memory, Fundamenta Informaticae. 2022, vol. 188
- W. Macyna M. Kukowski Bulk loading of the secondary index in LSM-based stores for flash memory, ADBIS 2022
- W. Macyna, M. Kukowski, Partially indexing on flash memory, DEXA 2019
- W. Macyna, M. Kukowski, Flash-aware storage of the column oriented databases, Fundamenta Informaticae. 2020, vol. 173