Ali Panesh

Almaty, Kazakhstan

☎+7(705)804-87-42 | ✉ paneshali3@gmail.com | panesher | in panesher

EDUCATION

Higher School of Economics

Moscow/Remote

Bachelor of Computer Science, ML department

2019-2023

EXPERIENCE

Yandex

Remote

Software Engineer C++ | Bank unit

Oct 2022 - Present

Designed and led smart the cache microservice for screen to maintain the consistency across main screens. Implemented multiple set-cookie and bson<->json conversion avoiding stack overflow in **userver** framework.

Implemented backend to use a restricted bank mobile application without a bank account.

Gologin

Remote

Deep Learning Engineer | Goless

Jul 2022 - Sep 2022

Made and trained the model for **object detection** with mAP@0.5 over **90%** to parse websites using **PyTorch** and **YOLOv5**.

Developed microservice in **Python** from scratch, which handles this model. The service detects bounding boxes for the website picture and writes it as a response to **MongoDB**. We chose **AWS S3** for storing images and **RabbitMQ** as a queue for this microservice.

Junior Software Engineer Chromium| Orbita Browser

Nov 2021 - Jun 2022

Found vulnerabilities in the fingerprint in getClientRect, canvas and WebGL images, etc. and fixed them with **C++**.

Created a rebasing script for **Chromium** using **Node.JS**, **Python**, and **git patches**, which decreased the time of rebasing between chromium branches by **90%**.

Yandex

Moscow

Intern Software Engineer C++ | Delivery unit

Jul 2021 - Oct 2021

Implemented the feature that reduced driver rejection by **10%**.

SKILLS

- **Programming languages**: C/C++, Python, SQL, Golang, JavaScript, Bash
- **Technologies**: Git, Linux, Docker, PostgreSQL, MongoDB, ClickHouse, Kafka, Redis, MapReduce
- **Languages**: English: upper-intermediate, Russian: native

PROJECTS

- **Diploma work: Dialogue Subject's Recommendation based on Knowledge Graphs**. Graph translation-baded model achieved hit@10 **32%** on Tipical-Chat, **37%** on Daily Dialog.
- NewSQL database. It's written using **C++**, **ragel** and **bison** for parsing queries, **LLVM IRBuilder** for JIT compilation for execution query. Every query running in RAM. Database use file for each table.
- **Chat** is a console **client-server C++** application based on **Boost.Asio** using **socket**, **thread**, **JSON**. The README contains all commands and instructions.