Vyatkin Alex

Email: a.vyatkinn@icloud.com
Phone: +7-912-233-33-59
Telegram: @blckvia
GitHub: Repository

About Me

Full Stack Software Engineer with experience in Backend, Frontend, Machine Learning, Computer Vision.

Skills Summary

- **Languages**: Golang, Python, TypeScript, JavaScript, SQL, Bash
- **Frameworks**: Django, Django Rest Framework, Nodejs, Expressjs, React, Redux, React Router Dom
- **Development**: Code review, Design patterns, Data Structures and Algorithms
- **Tools**: Docker, MongoDB, PostgreSQL, Git, MySQL, Postman, Linux
- **English**: IELTS: CEFR Level **B2** from 29/04/21.
- **Soft Skills**: Planning, cross-team collaboration, adaptation to change, training juniors colleagues, attention to detail, critical thinking.

Work Experience

- **Software Engineer** *Moscow, Russia*
  IT Sports *Mar 2023 - Present*
  - Rebuilt a 15-year-old PHP project into JavaScript, including creating an API. Optimized heavyweight SQL queries and implemented caching for certain queries. Achieved a tenfold increase in speed. Restructured some logic to improve efficiency.
  - Integrated two parsers into the API to simplify the extraction and processing of information to populate the website.
  - Developed a Telegram bot that meets the company's needs.
  - Mentoring the trainee, setting tasks, checking tasks, helping in solving problems, code review.
  - Project planning, selection of priority tasks, adjustment of current tasks.
- **Analyst** *Khanty-Mansiysk, Russia*
  Gazpromneft-Hantos *Dec 2017 - Apr 2022*
  - Conducted analysis and compiled a summary table for employees (about 1000 employees).
  - Monitored and verified the safe work of contractors at the field.

Projects

- **Online Shop**: **Frontend: simple ejs**. **Backend: Express.js, MondoDB:** Project Link
  - Functionality: Implemented CRUD. Viewing products is available to all users, to delete and change only the owner of the goods. Adding a product is available only to an authorized user. Part of the interface is not available for unauthorized user.
  - Payment: Cart, payment and receipt downloading implemented.
  - Privacy: Authorization followed by email notification, authentication, validation, also implemented the ability to change the password, fill out the form, receive an email and change the password via the link.
  - Security: The password is encrypted, in case of information leakage, only email from personal data will be available user.
- **News Feed, REST API**: **Express.js, MongoDB, JWT, WebSocket**: Project Link
  - Functionality: Implemented CRUD. Viewing posts is available to all users, deleting, editing, adding available only to the authorized user and owner of these posts. Implemented the ability to change the status user.
  - Privacy: Implemented validation of users and data entered on the server side.
- **REST API**: **Express.js, MongoDB, GraphQL, JWT, WebSocket**: Project Link.
  Same API as above, but with GraphQL and tests (Mocha, Chai, Sinon).
- **Stories, REST API: Django Rest Framework.**: Project Link.
  - Functionality: A simple Rest API using Django Rest Framework. The user must be logged in email and password. The user can post stories, delete, list and does not have access to stories owned by other users. The user cannot add a post if his quota is exhausted. The administrator can delete, create, list users and their stories. Administrator can set quota for each user. Initially, the user's quota is not defined.
- **GitHub Cards: Frontend: React + TypeScript.**: Project Link.
  - Functionality: Implemented a search for github users, displaying cards with information about their account. Working with the Github API.
- **Food Recipes**: **SPA: Frontend: React + React Router Dom**: Project Link.
  - Functionality: SPA app. There are many food categories available. Each category contains a certain number of dishes, it is possible to select a dish and see the description of the dish, country from which this dish is, recipe, ingredients, as well as a video from youtube. Also a search for categories was implemented. This app use external api.
- **Face Recognition**: GitHub Link.
  Implemented a number of Face Recognition algorithms comparing them to get the best result.
- **Telegram Bot**: Project Link. Telegram bot with two neural networks.

Courses

- **UDEMY**: NodeJS - The Complete Guide (MVC, REST APIs, GraphQL, Deno)
- **Deep Learning School by MIPT**: Deep learning CV specialization
- **Coursera**: Machine Learning
- **Stepik**: React, Redux, TypeScript, Javascript courses.

Education

- **The Ural Federal University, M.Sc.** *Yekaterinburg, Russia*
  Anti Crisis Management *2007 - 2012*
- **Tyumen Industrial University, M.Sc.** *Tyumen, Russia*
  Development and operation of oil and gas fields. *2015 - 2018*