

jakubsopel7@gmail.com

+48 531 133 571

Opole, Poland

## EDUCATION

- **Opole University of Technology**  
*Engineer of computer science*  
Enrolled in 2020

- **ZSE Opole**  
*High school with the field of computer science technologies, degree with the title of the IT technician*  
Completed in 2020

## [Link to github](#)



### ravel1 - Overview

Hey I'm Jacob. ravel1 has 31 repositories available. Follow their...  
[github.com](#)

# Jakub Sopel

## Python developer

## PROFILE

Hi, my name is Jacob, I've been working for almost 3 years as a backend developer. I always pursue my code to be as efficient as possible. Im eager to learn cool new technologies and expand my knowledge as much as possible.

## SKILLS

- REST API
- Python
- Flask
- Docker
- Elasticsearch
- Working in team
- Django
- Django rest framework
- GIT
- Linux

## EXPERIENCE

MATEUSZ JASIŃSKI SOFTWARE – 07.2021 – 01.2024

- System Architecture and Design:
  - Spearheaded the design and development of backend systems, ensuring scalability, reliability, and performance.
  - Collaborated with the architecture team to design data models and database schemas for optimal efficiency.
- API Development:
  - Developed and maintained RESTful APIs, enabling seamless communication between frontend and backend systems.
  - Implemented authentication and authorization mechanisms to ensure data security and user privacy.
- Database Management:
  - Designed and optimized database queries, reducing response times and enhancing overall system performance.
  - Implemented database migrations and ensured data integrity through regular maintenance.
- Integration and Third-Party Services:
  - Integrated third-party services and APIs to enhance system functionality and provide additional features.
  - Collaborated with the integration team to troubleshoot and resolve issues related to external service dependencies.
- Code Quality and Testing:
  - Implemented automated testing procedures, resulting in a significant reduction in bugs and enhanced software reliability.
  - Conducted code reviews to ensure adherence to coding standards, best practices, and maintainability.

# COMMERCIAL PROJECTS

## **User Dashboard**

*2023*

This project is being created to replace an old legacy user dashboard that houses all the projects, profiles and statistics for them in a one central place.

Tech stack: django, drf, elasticsearch

## **Content Feature**

*2023*

This feature consisted of creating a brand new set of apis used to display and interact with all content types mentioned below.

It required refactoring and rewriting a significant amount of legacy endpoints, the priority was speed.

Tech stack: wagtail, django, drf, docker, elasticsearch

## **Global Search System**

*2022*

Project aggregating all content in a multi million record database.

Medical Content (articles, polls, quizzes), medical professionals, clinics, hospitals, non-for-profits.

Feature required creating a new elasticsearch index as well as custom serializers for each type to unify the data for the frontend application.

Tech stack: django, drf, docker, elasticsearch

## **Home page**

*2022*

Project required creating a brand new and modern landing page accommodating wagtail cms for dynamic content placement.

Tech stack: django, drf, wagtail.

## **Global wallet system**

*2022*

Project consisted of a global system for purchasing a premium currency to be used for promoting ones own medical profile as well as an admin panel for managing said packages.

Achieved test coverage was 90%.

Tech stack: django, drf, stripe payments

## **Hub Feature**

*2021*

Similar project to the community, this time associating medical professionals in a given field, based on their claimed profile in a large ecosystem. Project also included an admin panel.

Tech stack: django, drf, elasticsearch, PostgreSQL, Docker, black, isort

## **Community Feature**

*2021*

This project consisted of creating a fully fledged social board associating users in a close area based on the location. The board supported all necessary social features, posting, commenting reacting. Project also included an admin panel.

Tech stack: Django, drf, elastic search, PostgreSQL, Docker, AWS