Yevhenii Shendrikov

Python Developer

jackshendrikov.com linkedin.com/in/yevhenii-shendrikov github.com/jackshendrikov
jackshendrikov@gmail.com Kyiv, Ukraine

Profile

With a total of 4 years of experience as a Python Developer, I possess a versatile skill set that spans various domains. My expertise lies in designing and overseeing data collection, analysis, and automation solutions, with a particular focus on ETL pipelines, asynchronous APIs, and data workflows. I have a background in machine learning, encompassing LSTM models, sentiment analysis, and classifiers. Additionally, I excel in code optimization, ensuring high-quality results, and providing comprehensive documentation.

Skills

Programming Languages: Python, JS, C, Java

Backend Frameworks: FastAPI, Django, Flask

Frontend: HTML5, CSS3, Bootstrap, JQuery

DBMS: PostgreSQL, MongoDB, Redis, InfluxDB, Elasticsearch, MSSQL

CI/CD: Github Actions, Azure Pipelines, Jenkins, Pytest

DevOps: Docker, Kubernetes

ETL: Airflow, Scrapy

Monitoring & Visualization: Prometheus, Grafana, Azure Dashboard

Azure: Data Factory, Functions, AKS, ACI

Other: ML, Numpy/Pandas, AsyncIO, Graylog

Professional Experience

03/2023 – present
Kyiv, Ukraine

Software Engineer, *GlobalLogic*

Stack: Python, FastAPI, ETL, Azure, InfluxDB, Docker, Kubernetes

- Collaborated on designing an Azure cloud solution, managing scenarios, repositories, and deployments through pipelines.
- Conducted in-depth research on undocumented legacy code.
- Designed and implemented Python-based ETL pipeline logic.
- Leveraged Data Factory and Python Azure Functions for workflow optimization.
- Enhanced Python codebase to significantly improve performance, including debugging complex queries.
- Engineered asynchronous APIs using FastAPI.
- Orchestrated setup and maintenance of robust CI/CD pipelines.
- Developed exhaustive test suites utilizing pytest.
- Enhanced team understanding through well-documented code and informative architecture diagrams.
- Presented demos to English-speaking stakeholders.
- Divided user stories into precisely estimated tasks for efficient project planning.
- Applied Agile methods, seamlessly switching between Scrum and Kanban as needed for the project.

05/2021 – 03/2023
Kyiv, Ukraine

Python Delevoper, *Simporter*

Stack: FastAPI, Scrapy, Django, Pandas, Jenkins, GitHub Actions, Docker, MongoDB, Elasticsearch, Redis, Prometheus, Airflow, Graylog, Flask

- Created and maintained data collection scrapers using Scrapy.
- Implemented internal APIs using FastAPI.
- Enhanced data pipeline with modularity, parallel processing, and monitoring.
- Designed Django and FastAPI admin panels for data management.
- Utilized Pandas and Numpy for data processing and analysis.
- Set up CI/CD pipelines using Jenkins and GitHub Actions.
- Established infrastructure services like Graylog and Prometheus.
- Automated and optimized data scraping and delivery processes.
- Authored technical documentation and guides.

05/2022 – present
Kyiv, Ukraine

Full-Stack Developer, *Strichka (Independent Project)* 🔗

Stack: Django, DRF, Scrapy, Heroku, Docker, GitHub Actions, JS+CSS

- Conceptualized, designed, and developed movie discovery platform, managing the entire project lifecycle.
- Engineered a robust web scraping framework utilizing Scrapy to provide real-time movie availability data.
- Developed user registration with email confirmation, enabling user engagement through ratings, comments, and watchlists.
- Developed a RESTful API (Django REST framework) with Swagger documentation.
- Implemented caching and Dockerization for optimal performance and scalability.
- Crafted a responsive front-end with CSS, JavaScript, and popular libraries.
- Enhanced Django admin panel for efficient data management.
- Maintained code quality with GitHub Actions and linters.
- Provided comprehensive documentation for reference.
- Continuously improved the platform based on user feedback.

12/2020 – 05/2021
Kyiv, Ukraine

NLP Engineer, *Freelance*

Stack: Keras, scikit-learn, Django, NLTK, Gensim, Pandas, JS+CSS

- Developed sentiment analysis models for English texts using LSTM models and machine learning classifiers.
- Designed and built a user-friendly web application with the Django framework, incorporating a trained model to analyze and categorize text documents.

Education

09/2018 – 07/2022
Kyiv, Ukraine

Computer Engineering, *NTUU 'Kyiv Polytechnic Institute'* 🔗

GPA: 3.87 (Diploma with Honours)

Publications

09/2021

Implementation of Language Processing Tools for the University Quality System, *IEEE* 🔗

Publication describes the development of an AI tool for sentiment analysis to streamline the processing of non-multiple choice surveys, reports, and comments, aiding in educational quality assessment and decision-making.

Projects

11/2022 – present

Fjord API, *FastAPI Translation App featuring a Proxy Pool* 🔗

Stack: FastAPI, MongoDB, PostgreSQL, Redis, Piccolo, Jinja, aiohttp

Fjord API streamlines text translations and proxy acquisition. It offers text translation in multiple languages, automatic language detection, access to various translation providers, rapid proxy retrieval and storage (5-8 seconds), an admin panel for simplified data management, and seamless integration with MongoDB, PostgreSQL, and Redis for efficient data storage and retrieval.

04/2022 – 06/2022

Sensus, *NLP with LSTM for Sentiment Analysis of Ukrainian/English texts* 🔗

Stack: Python, Keras, Gensim, Pandas, NumPy, NLTK, pymorphy2, scikit-learn

Sensus/Sentimento delves into sentiment analysis using LSTM models. It encompasses full model development across three iPython notebooks, covering the sentiment analysis process from data processing to comparative analysis of LSTM models. The project also offers data visualization at various stages to enhance understanding and is tailored for sentiment analysis in both Ukrainian and English texts, making it valuable for language-specific sentiment tasks.

09/2021 – 11/2021

CCompiler, *C compiler written in Python* 🔗

Stack: Python

Python C Compiler emulates a C compiler, translating C-like code into MASM32 assembly. Key features include full compilation stages, support for arithmetic operations, function declarations, looping constructs, conditional logic, bitwise operations, and variable assignment with common data types. It enables modular programming and efficient control flow for code organization and reuse.

Certificates

Python for Everybody Specialization 🔗
Coursera (University of Michigan)

Capstone: Retrieving, Processing, and Visualizing Data with Python 🔗
Coursera (University of Michigan)

Django for Everybody Specialization 🔗
Coursera (University of Michigan)

Internet History, Technology, and Security 🔗
Coursera (University of Michigan)

Languages

Ukrainian ●●●●●
Native or Bilingual Proficiency

Norwegian ●●○○○
Elementary Proficiency

English ●●●●○
Full Professional Proficiency

Italian ●●○○○
Elementary Proficiency

Interests

Cryptanalysis **Painting** **Papercutting** **History** **Chess**