

full stack software engineer

Filippo Scotto

 Livorno, Italy
 filipscotto@gmail.com
 <https://pippogit.github.io>

My name is **Filippo Scotto**, I am a full-stack software engineer who loves to work with modern web technologies such as **React/NextJS**, but I also have a solid academical background in **computer engineering**, **machine learning**, and **networking**.

Skills, Programming Languages & Technologies

Frontend

- Languages: Javascript, TypeScript, HTML/CSS
- Libraries and frameworks: React, NextJS, TailwindCSS, ChakraUI, jQuery
- Design: basic knowledge of Figma, Photoshop, Illustrator

Backend

- Languages: PHP, Python, NodeJS, Java
- Libraries and frameworks: Laravel, Flask, ExpressJS
- Databases: relational (MySQL and PostgreSQL), no-sql (MongoDB)
- APIs: GraphQL, REST
- Tools: Docker

Low-Level

- Languages: C, C++, basic ASM knowledge

Machine Learning & Data Engineering

- Libraries: Pandas, SciKit, PyTorch, Numpy
- Tools: MATLAB

Work Experience

- 2022 **Food Delivery Platform - Full Stack Developer**
Delivery Aggregator SRL
Implemented a **food delivery platform**:
- **Front-end**: Typescript, NextJS, Jotai Atoms, and ChakraUI
 - **Back-end**: PHP, Laravel, GraphQL, and PostgreSQL with GIS extension
- 2021 **Design and implementation of a Business Management Web App**
Freelance project for ITS Telecomunicazioni SRLs
Designed and implemented a **business management** web application:
- **Front-end**: Javascript, React, MaterialUI
 - **Back-end**: Google Firebase

Education

- | | | |
|------|---|----------------------|
| 2021 | University Of Pisa
MSc in Computer Engineering | 110/110
cum laude |
| 2018 | University Of Pisa
BSc in Computer Engineering | 101/110 |
| 2013 | ITI G. Galilei Livorno
Secondary School Diploma in Information Technology | 100/100 |

Appendix: University Projects

Here is reported a partial list of the projects I developed during my university years. If you want to see more, please visit my website: <http://pippogit.github.io>.

Foodle Food Images Classifier

Computer Vision, Information Retrieval, and Deep Learning

Implementation of a Computer Vision toy application that can be used to identify and classify food images by using CNN and features indices. The application was implemented with vanilla Javascript and HTML.

Usecase and architecture: The users can use their smartphone to scan the food, and the image will be provided to the server that will extract the features using a **Deep Neural Network**. The features will finally be used to perform a **k-NN classification** using an **ElasticSearch Index** to find the candidate class. The results will be provided back to the users in a matter of seconds since the scan had begun.

See more: <https://pippogit.github.io/u/foodle>

HOME: HOpefully sMart NEws aggregator

Data Mining and Machine Learning

Implementation of a machine-learning-powered news aggregator using Python and HTML/CSS for the front-end application.

Usecase and architecture: the Python server periodically fetches news from the most reliable sites using RSS feeds. Each article is then processed by a feature extractor that performs information retrieval techniques to extract the keywords. The features are then fed to a **Support Vector Machine** that classifies each article. The user can request a personalized news feed using a **REST API (Flask WebServer)** and can also like an article to optimize the feed. The users' data are stored on a **MongoDB database**.

See more: <https://pippogit.github.io/u/home>

WebSecure Chat

Cybersecurity

Implementation of a client-server Instant Message service that used cybersecurity mechanisms to provide a TOFU (Trust On First Use) **end-to-end encryption**.

Usecase and architecture: each user must have an **RSA** pair key provided by the organization that hosts the server. When a user wants to begin a chat session with another user, they will use the RSA keys to digitally sign a fresh **AES** key, that will be encrypted using the public key of the recipient. The recipient will be able to verify the key and the origin, if everything is good, the e2e encrypted chat session will begin.

See more: <https://pippogit.github.io/u/chat>



scan to see full portfolio