

# Sérgio Félix

[sergiofelixdev@gmail.com](mailto:sergiofelixdev@gmail.com)

[github.com/WallQ](https://github.com/WallQ)

[linkedin.com/in/sergiofelixdev](https://linkedin.com/in/sergiofelixdev)

[portfolio website](#)

## EXPERIENCE

### **Basicamente Digital**

Mar 2022 – Aug 2022

Front-end Developer & UI/UX Designer, Internship

(6 months)

- Redesigned the company marketing website to enhance UI & UX, seeking sales increase.
- Developed the website using the React ecosystem, ensuring modern web technologies.
- Achieved a 90+ SEO score using Google Lighthouse optimization techniques.

### **Risema**

Feb 2019 – Apr 2019

Network Technician, Internship

(3 months)

- I assisted in maintaining the network architecture of affiliated companies.
- I did a case study by developing a ransomware in C++ for my professional aptitude exam.

### **Risema**

Feb 2018 – Apr 2018

Computer Technician, Internship

(3 months)

- Responsible for building and maintaining computers for clients and affiliated companies.

## EDUCATION

### **Higher School of Technology and Management - IPP**

Sep 2022 – Aug 2025

Bachelor's Degree in Computer Science and Engineering

Agile Methodologies, Algorithms, Artificial Intelligence, Code Optimization, Concurrency & Parallelism, Data Structures, Databases, Distributed Systems, Data Warehousing, ERP's, Information Systems, Memory Management, Multithreading, Object-Oriented Programming, Operating Systems, Python, Sockets, Software Architecture, Software Design Patterns, Software Engineering.

### **Higher School of Technology and Management - IPP**

Sep 2020 – Aug 2022

Higher Professional Technician Diploma in Development for the Web and Mobile Devices

Angular.js, Data Structures, Databases, Distributed Systems, Functional Programming, Ionic, Java, JavaScript, MongoDB, Node.js, Object-Oriented Programming, PHP, React.js, SQL, Software Engineering, Swift, TypeScript, UI/UX.

## Secondary School of Felgueiras

Sep 2017 – Aug 2020

Professional Certification in Management and Programming of Computer Systems

.NET, Bootstrap, C, CSS, Databases, Data Structures, Functional Programming, HTML, JavaScript, jQuery, Object-Oriented Programming, Pascal, PHP, SQL.

## PROJECTS

### Cinemaster

Jul 2024 – Present

Full-stack development, Personal

A **multi-tenant SaaS** cinema ticketing system built using a **microservices architecture** with **NestJS** for the back-end and **Next.js** for the front-end. It features a **schema-per-tenant model** on a **shared PostgreSQL database** and utilizes **RabbitMQ** for inter-service communication. The system is structured as a **monorepo** and **containerized with Docker**. **Authorization** is handled with **role-based access control (RBAC)** to manage user permissions, while utilizing a **hybrid authentication approach** that combines **JWT and session management** using **Redis**. The platform offers movie browsing with pagination, screening management, **real-time seat availability**, **Stripe integration** for ticket sales, and PDF ticket generation for easy access. A strong focus on delivering an MVP as soon as possible.

### Artificial Intelligence

Jun 2024 – Jul 2024

Full-stack development, Academic

An application designed to help users **predict whether an email is spam** using **binary classification models**. The models were trained in **Python** using **Random Forest**, **Support Vector Machine**, and **Logistic Regression machine learning algorithms**, and evaluated with **scikit-learn**. The front-end is built with **Next.js** and styled with Tailwind CSS to ensure a fast, responsive, and user-friendly experience. The back-end **API** is built with **Flask** and serves the prediction models, while a **PostgreSQL database** stores data such as email inputs, predictions, and user information.

### Organizational Information Systems

Apr 2024 – Jul 2024

Full-stack development, Academic

A **Decision Support System (DSS) application** developed using **Next.js (T3 Stack)** with **OAuth2.0 authentication**, data parsing from **SAFT files**, and **file upload** functionality. The application processes the data through a two-step **ETL (Extract, Transform, Load)** process. Initially, the data is extracted from the uploaded files, **normalized to 3NF**, validated, and loaded into a **relational database**. Next, the data is **denormalized for performance optimization**, enabling efficient queries and aggregations. **Surrogate keys** are generated along with the **precalculated values** stored in a **star schema database**, which facilitates fast decision-making by providing users with an **interactive dashboard** of stock, sales, and other business processes.