

# Ahmed Salah El-Din Shemais

## Software Engineer

✉ ahmed.shemais98@gmail.com ☎ 01151693352 📍 Cairo, Egypt 🗣 Ahmed Shemais 🌐 Ahmed Shemais

### Profile

Versatile Software Engineer with 3+ years of experience building robust, scalable applications for both cloud and desktop. Expertise in creating cloud-native microservices, responsive frontends, and integrated desktop software. Passionate about leveraging the power of AI to solve complex challenges and deliver high-performance, impactful solutions.

### Education

**Bachelor's Degree - Systems & Biomedical Engineering,** *Cairo University*

2016 – 2021

### Work Experience

**Software Engineer,** *Voyance*

2023 – Present

Contributed to the end-to-end development of a large-scale, cloud-native medical imaging platform (VPAX) and its associated C# acquisition software (VoyanceX). Worked across the full stack, from backend microservices and databases to the responsive ReactJS micro-frontend. Authored system design documentation and maintained system security.

#### Key Projects & Achievements:

- **VPAX (Cloud-Native Medical Imaging Platform):**
  - **Architecture & Performance:** Engineered the system to handle **150x peak traffic** loads and **cut database costs by over 30%** by implementing **advanced query optimizations** and **caching strategies**, which also significantly improved query performance.
  - **AI Integration:** Integrated generative AI models into core healthcare workflows, boosting practitioner efficiency.
  - **Frontend:** Actively contributed to the full frontend lifecycle, from rapid prototyping to building production-ready features. Focused on enhancing UX/UI and helped **decrease application load times by 25%** through performance investigation, optimization, and implementing a Backend for Frontend (BFF) pattern.
  - **Security:** Implemented a granular Role-Based Access Control (RBAC) system, ensuring strict, secure data isolation in a complex multi-tenant platform.
  - **Integrations:** Developed seamless integrations with external cloud providers, including Dropbox and OneDrive, to streamline user data workflows.
- **VoyanceX (X-Ray Acquisition Software):**
  - **Core Development:** Developed and maintained the C# acquisition software, ensuring its seamless integration with the VPAX cloud system.
  - **Hardware Integration:** Enhanced hardware compatibility across diverse client setups by integrating the software with new devices.

### Freelance Development

2021 – 2023

- Developed and deployed custom websites for clients using technologies such as NodeJS, WordPress, and Moodle.
- Managed continuous client communication to ensure the delivered solutions were perfectly aligned with their needs.

### Publications

**A. Shemais et al., "Cloud-Based AI-Enhanced Dual-Mode System For Automatic Coronary Artery Calcification Detection and Quantification",**

2024

*2024 41st National Radio Science Conference (NRSC), New Damietta, Egypt, 2024, pp. 270-277, doi: 10.1109/NRSC61581.2024.10510468. ☑*

### Projects

**Access and Registration Hospital Information System,** *Java, Android, C, Proteus, PHP*

- Eases the burden of queuing by using our application to easily reserve appointments and directs the user to complete any required checkups using our smart kiosks that measure heart rate, SpO2, blood pressure etc.
- Allows easy and quick inpatient record management utilizing NFC tags and an accompanying android app.

**E-Commerce App,** *NodeJS, NextJS, BullJS, TypeScript, MongoDB, Redis, NATS, Stripe, Jest, Docker, Kubernetes, GitHub CI.* ☑

A ticket shop app that allows users to browse, buy and sell tickets for various events.

- Demonstrates how to build a scalable, reliable and secure e-commerce website with a **microservices** architecture.
- Built using a **NodeJS** backend connected through **REST** API to **NextJS** frontend, **protected** through **JWT** authentication.
- Utilizes **Mongo & Redis** databases for storage, **NATS** event bus for communication, and a payment system using **Stripe**.
- Unit testing is done with **Jest**, workers with **BullJS**, **CI/CD** with GitHub and Deployment using **Kubernetes**.

**Vending Machine API,** *Java, Spring Boot, Spring Boot Security, Hibernate, PostgreSQL, Swagger, OAS*

A vending machine API where you can deposit money, buy and sell products, manage stock, and withdraw your balance.

- Uses **JWT**-based authentication to implement role-based authorization.
- Custom exception handlers to unify possible errors.
- Implements **OpenAPI** Specifications using **Swagger**.

### **DMaxCare, A Dental Services-Oriented HIS** [↗](#)

- Designing and Implementing the DB, Backend, and Frontend.
- Used Technologies: NodeJS, MySQL, Sequelize ORM, jQuery, Bootstrap

### **Temperature Controller Module, ATmega32, C, SPI, Proteus, MISRA-C** [↗](#)

- Simulates a heating system with the ability to reach a targeted temperature determined by the user.
- Operates in 4 states (Standby, Operational, Normal, Error) triggered when certain temperatures are reached.
- Uses the SPI protocol to communicate with TC72, ADC and PWM to control the heater and an LCD to display the temperature.

### **Store API, NodeJS, NestJS, TypeORM, MySQL**

A REST API where users can put items on sale and buy them while generating invoices.

- Implement role-based authorization using JWT-based authentication.
- Uses custom interceptors to generate unified responses.

### **Healthcare Computerized Maintenance Management System, NodeJS, Express, SQLite, Sequelize, Socket.IO.** [↗](#)

A web-based application that helps healthcare organizations maintain their medical assets, equipment, and facilities.

- Manages work orders for preventive and corrective maintenance of medical equipment and monitors the inventory.
- Developed using **NodeJS/Express** and **Bootstrap**, with support for **SQL** Databases for storage.
- Live notifications for urgent events using **Socket.IO**, performance report generation and role-based access control.

### **ERP System For Student Organizations, NodeJS, Google APIs (Sheets, Recaptcha), MySQL, Sequelize ORM, jQuery, Bootstrap.** [↗](#)

A system that helps manage and monitor several aspects of a student organization.

- Facilitates recruitment, interviewing candidates, members management, organizing workshops and monitoring KPIs.
- Developed using **NodeJS, Express** and **Bootstrap**, integrated with Google Sheets through **Google APIs**.
- Deployed, and used by 2000+ students.

### **Self-Driving Car, ATmega32, NodeMCU, C, OpenCV, MeteorJS**

- Built a prototype car with autonomous and manual driving modes, it streams a video of its path to connected clients.
- In autonomous mode, the client app analyses the live stream using computer vision to detect lanes and determine its path.
- In manual mode, controlled using our client app made using MeteorJS where the user can also view the stream.

### **Smart Home Security System, ATmega32, C** [↗](#)

- Contains multiple features such as password-protected home entry, alarms, password change & temperature control.

### **Indoor Localization, ESP8266, Firebase, MeteorJS**

- Utilizes WiFi networks to triangulate location indoors, and displays it on a map in the accompanying mobile application.

### **8051 Elevator, Intel 8051**

- Simulates the workings of an elevator in a 5-story building using Intel 8051 microcontroller.

### **Robotic Hand**

- Built a hand model controlled by wearing a glove with embedded sensors, it copies the wearer's hand movements.

### **Infusion Pump, C++, AVR, Proteus**

Real-life simulation of an infusion pump with volumetric flowrate adjustment option.

### **Full Body Movement Simulator, C++, OpenGL** [↗](#)

- Provides a simulation of an interactive human body in OpenGL.

### **Songs Identifier, Python, PyQt5** [↗](#)

Identifies a given song by comparing it to a preconfigured songs database.

- Extracts multiple features from source songs and stores them in a database
- Identifies songs by comparing their features to existing ones in the database, and shows their similarity percentage to others.

### **Music Equalizer and Visualizer, Python, PyQt5, NumPy, SciPy** [↗](#)

A full-featured music player with audio controls and playlists.

- Produces an animated visualization of the song being played, and enables editing it using the built-in equalizer.

### **Data Analysis Tool, React, React Router, Redux, Flask, NLTK, OpenCV, Pandas, Gensim**

A data analysis tool that can process tabular, image and text data outputting and visualizing several statistics.

- Tabular: Calculates basic descriptive statistics for each feature, along with the covariance of all features using Pandas.
- Image: Generates color histograms and color-based segmentation masks using OpenCV.
- Text: Uses the word2vec algorithm to calculate word vectors then visualizes them using T-SNE.

### **Food Ordering App, React, Redux, Express, TypeScript, Mongoose, MongoDB**

A Full-Stack MERN app allowing users to make food orders and store them for processing.

### **Microservices-Based Blog, NodeJS, React, Docker, Kubernetes.** [↗](#)

A simple blog made using NodeJS REST microservices and React, with a custom event bus.

### **Medical Image Analysis Plugin, Python, PyQt5, PyTorch, Linux, AWS, Flask.** [↗](#)

A plugin for 3DSlicer aiding in health assessment by analyzing heart CT images and calculating related health risk factors.

- Developed using **Python** and **PyQt5**, implementing **multiprocessing** techniques.
- Integrated with our **Deep Learning** model for accurate segmentation aided by **image processing** to produce its results.
- Includes an accompanying **Flask** server for cloud processing, deployed to an **AWS EC2** instance.
- The results and methods of this project were published in the 2024 41st National Radio Science Conference (NRSC). [↗](#)

**AI-Powered Email Client,** *NestJS, Postgres, AI SDK, LangChain*

A full-featured email client built to streamline inbox management with generative AI capabilities.

- Developed key AI features including **thread summarization**, **automatic reply generation**, and **daily inbox digests**.
- Implemented a "Chat with your Inbox" feature using Retrieval-Augmented Generation for natural language querying.
- Built a **provider-agnostic architecture** to integrate seamlessly with multiple email services (e.g., Gmail, Outlook).

**Technical Skills**

---

**Programming Languages**

*JavaScript | TypeScript | Python | Java | C/C++/C# | PHP*

**Technologies**

*NodeJs | Git | GCP | Socket.IO | Docker | Kubernetes | Linux*

**Database**

*MySQL | SQLite | MongoDB | Redis | Sequelize | TypeORM*

**General Concepts**

*Object-Oriented Programming | Data Structures | Algorithms | Design Patterns*

**Frameworks**

*ExpressJS | Spring Boot | Flask | React | Next | Jest | PyQt | Meteor | NestJS*