# Karim Atef Shaaban

Alexandria – Egypt
+20 1159485872 • karimshabaan170@gmail.com
GitHub LinkedIn Portfolio

## Education

**Arab Academy for Science, Technology and Maritime Transport.** **AAST.**
*Bachelor of Science in Computer and Systems Engineering* *2020–2025 (One Preparatory Year)*
*GPA 3.7* (Excellent with Grade of Honor)

## Technical Skills

- **TypeScript**, **Java**, **Python**, **C**, **C-Sharp**.
- **Data Structures, Algorithms, OOP, Cloud Computing** and **System Design**.
- **Full-Stack Development (Next.js, Express.js), Version Control, API Testing, CI/CD and Containerization (Docker, K8s)**.

## Achievements

- **Among top 3 percent of students** Egyptian GSC 2019.
- **Honorable mention in AAST competitive programming contest.**

## Experience

**Freelance Full-Stack Software Engineer** *2025 – Current*

- Developed, maintained and hosted full-stack web applications using modern frameworks.
- Collaborated with international clients to build features, resolve production issues, and meet deadlines.
- Improved application performance, SEO, and Core Web Vitals.
- **Epco Group Company Website:** Rebuilt EPCO's website using a modern tech stack, reducing the homepage size from 8 MB to 1 MB (8× faster on mobile) and improving technical SEO to achieve a first-place Google ranking. Added a full marketplace with individual product pages, automatic bilingual support, and automatic light/dark theme detection based on user preference.

## Selected Projects

- **Smart and Secure Harvesting System (Graduation Project):** Designed and devleoped an automated harvesting system using IoT, AI, and cloud technologies in a modular monolith architecture. Integrated a robotic arm controllable via a mobile app. Used the MQTT protocol to handle real-time sensor data. Deployed a serverless Node.js backend on GCP for the web application. Implemented a YOLO-based AI model to assess crop ripeness and diagnose diseases. Stored data in MongoDB hosted on AWS and managed image uploads with Amazon S3. Developed a Next.js web application with a dashboard for real-time monitoring. Configured CI/CD pipelines with GitHub Actions for all servers.
- **Tickets Reservation System (TypeScript, Express.js, Docker, Kubernetes, NGINX, Redis):** Developed a scalable microservices-based system for buying and selling tickets, allowing users to reserve tickets for a limited time before automatic order expiration. Implemented services for authentication, tickets, orders, payments, and expiration. Utilized Skaffold to streamline local Kubernetes development. Applied optimistic concurrency control to ensure data consistency across distributed services. Automated testing with Jest and CI pipelines via GitHub Actions. Published a shared NPM package for reusable logic across services.

## Certified Online Courses

- Algorithmic Toolbox.
- Data Structures.
- Meta Front-End Developer.
- Meta Back-End Developer.

## Languages

- **Arabic:** Mother Tongue.
- **English:** Full Professional Proficiency.
- **French:** Elementary Proficiency.

## More Projects

- **E-Commerce Website (Next.js, Express.js, MongoDB, Tailwind CSS, AWS):** Developed an e-commerce platform with Next.js using server-side and client-side components. Features include JWT-based authentication, role-based access control, admin product management, shopping cart, advanced search, sorting, pagination, and filtering. Implemented both light and dark themes. Ensured security with protection measures against XSS, NoSQL injection, and DDoS. Authored detailed API documentation using postman.

- **Discussion Platform (React, Node.js, Express.js, Microservices, Event Bus):** Built a discussion platform using a microservices architecture with asynchronous communication via a custom event bus. Implemented event persistence to ensure delivery to services that were temporarily offline. Developed the frontend with React and the backend with Node.js and Express.js.

- **SIC One-Pass Assembler** and **SIC Disassembler (Java):** Developed a toolset for the Simplified Instructional Computer (SIC) using Java, which includes both a one-pass assembler and a disassembler. The assembler converts assembly code into byte machine language while generating a symbol table. The disassembler uses reverse engineering techniques to convert machine language back into assembly code.

- **ATM Machine Simulator (Java, SQL):** Designed and developed a comprehensive ATM simulation project on the computer, featuring a user-friendly GUI interface enriched with CSS animations. The project deals with essential banking operations, aiming to replicate real-world ATM interactions. It allows users to register new bank cards, manage deposits and withdrawals, check account balances, review transaction histories, and execute inter-account transfers. The project operates by integrating a MySQL database to store and manage customer data.

- **Advertising Agency Website (React, Bootstrap, Express.js, MongoDB):** Developed a responsive website for an advertising agency using React for the frontend, styled with HTML, CSS, and Bootstrap, emphasizing principles of UI/UX design. Integrated data fetching from a Node.js backend with a MongoDB database, ensuring dynamic content updates and smooth user interactions.

- **Restaurant API (Python):** Developed a Django-based API for restaurants, incorporating authentication, authorization, search capabilities, and rate-limiting functionalities. Designed with a focus on security, scalability, and efficient data management to provide reliable service for users and administrators.

- **Flight Booking System (C-Sharp):** Created a user-friendly flight ticket booking app using C-Sharp with a GUI. Users can register, log in securely, book flights, and easily pay for tickets via a checkout form. Customer data is managed using files.

- **Connect-Four Game (Java):** Two-player game where players take turns dropping their colored disks. The objective of the game is to connect four of your disks vertically, horizontally, or diagonally. The winner is the player having the greater number of connected-four disks. Features include human vs human mode, human vs computer mode, and undo/redo. The game is played in text mode using the command-line interface.