EMMANUEL OPPONG

318-245-4129 | emmanuelopponga07@gmail.com | linkedin.com/in/emmanuel-oppong-acheampong | github.com/emmaeng700
emmanuelopponga.com

Education

Grambling State University | **Graduating, May 2028**
Bachelor of Science in Computer Science | *Grambling, Louisiana*

Relevant Coursework: Object-Oriented Programming, Software Engineering, Data Structures & Algorithms, Operating Systems
Awards: Dean's List 24 and 25, **Presidential Scholarship**

Experience

Research Assistant | ML under Dr. Dileon Saint-Jean | **Jan 2025 – Present**
Grambling State University | *Grambling, LA*

- Investigated and prototyped the use of **AES-GCM to encrypt one-time passwords (OTPs)**, ensuring confidentiality and integrity. Utilized Redis for secure storage and rapid retrieval of encrypted OTPs.
- Developed a **caching mechanism using Redis Sets and TTL** to block OTP reuse and automatically expire OTPs after a configurable interval, ensuring that no OTP was reissued **within 10 minutes**.
- Explored advanced ML models to **detect fraudulent activity in OTP requests**, such as suspicious IP addresses or excessive failed attempts. Incorporated dynamic policies to flag anomalies and strengthen security measures.
- Proposed and tested methods to **dynamically adjust OTP time-to-live (TTL) based on user behavior patterns**. Integrated additional context signals (like typing speed and device location) for continuous risk assessment, prompting extra verification steps upon detecting anomalies.

Software Engineer Intern | Fullstack Engineer Trainee | **Feb 2023 – Feb 2024**
Ericsson | *Remote*

- Engineered a **Kafka-streaming microservice on Kubernetes** processing 400K+ events/day across **6** topics, replacing **45-minute manual SQL workflows** with **REST API** calls **delivering sub-200ms latency** for 30+ business users.
- Architected an event-driven system using **Apache Kafka, F# and S3** that aggregates and **serves 10K pricing records via RESTful endpoints**, enabling real-time access to historical trade data previously locked in disconnected systems.
- Built a **React dashboard** visualizing distributed pipeline event timelines in real time, **reducing debug/resolution time** from **60 minutes to 10 minutes** for a 30-engineer team.

Software Engineer Intern | Fullstack Intern | **May 2022 – Aug 2022**
Noon | *Abu Dhabi, UAE*

- Developed a **scalable IoT data pipeline** (**Apache Kafka Connect, MQTT, PostgreSQL**) streamlining sensor data processing and cutting downtime detection by **40%**.
- Created **50+ automated unit & integration tests** (SpringBoot, JUnit, Postman), improving backend reliability by **80%**.
- Implemented a **real-time** push notification system(**Google Firebase**) delivering status updates with under **10ms** latency.

Projects

MateMatch (ColorStack Hackathon) | *Next.js, React, Google Cloud, Firebase, Puppeteer, Cheerio*

- Built a secure web app with Firebase Auth for streamlined sign-ins and Firestore **for real-time sync**, enabling **interns to discover, list, and share housing via location-aware** features and dynamic filters.
- Used **puppeteer-core with browserless.io** over a secure **WebSocket to crawl Airbnb listings**, extracting prices, amenities, coordinates, and ratings from data-testid elements. Enforced timeouts and custom user agents for stealth, persisting data in Firestore for real-time filtering under Firebase Cloud Functions.
- Built a **responsive interface with Shadcn/UI, Radix UI, and Tailwind CSS** for saved listings, shared housing feeds, and real-time chat. Integrated Google Maps and **the Distance Matrix API** for quick commute-time calculations.

Go Container | *Go, Linux*

- Built a **container runtime in Go** with process isolation by leveraging Linux namespaces to give each container its own process space, filesystem, and hostname without affecting the host.
- Implemented **resource controls using cgroups** to cap the number of processes and automatically clean up after container exit, ensuring reliable and efficient system utilization.
- Set up a secure, self-contained filesystem environment using **chroot and mounting pseudo-filesystems**, providing complete separation from the host and preventing unwanted interference.

NanoSQL | *C*

- Built a **high-performance SQLite-clone database engine in C**, implementing **B-tree indexing**, persistent storage, and custom memory management.
- Engineered a **command-line interface with efficient data serialization**, parameterized SQL operations (INSERT/SELECT), and integrity validation.
- Architected **a complete B-tree implementation featuring dynamic node splitting**, rebalancing, and optimized page allocation.

Technical Skills

Languages: Python, Java, JavaScript, TypeScript, Go, C
Experienced with: Spring Boot, GraphQL, PostgreSQL, MongoDB, Apache Spark, Apache Kafka, Docker, Kubernetes, JUnit, AWS S3, Git, Express.js, Node.js, Flask, APIs, Firebase, React.js, Next.js, HTML, CSS, Bootstrap,Postman