

# KEHINDE GIWA

Full-Stack engineer building with React.js, Next.js, TypeScript, Node.js and Spring Boot,

Portfolio website here: [kennyg.vercel.app](https://kennyg.vercel.app).

Phone: +2349068998656 | Email: [giwatwins@gmail.com](mailto:giwatwins@gmail.com) | [LinkedIn](#) | [GitHub: github.com/Kenny-204](#) | [LeetCode](#)

## EDUCATION

---

### University of Lagos

Bachelor of Science, Computer Science

GPA: 4.31/5.0

Relevant Courses: Data Structures and Algorithms, Automata Theory, Operating Systems, System Design, Networking.

Lagos, Nigeria  
Expected: AUGUST 2027

## SKILLS

---

**Programming Languages:** Python, JavaScript, TypeScript, C#, Java.

**Tools & Frameworks:** React.js, Next.js, Node.js, Express.js, Spring Boot, Socket.io, PostgreSQL, MongoDB, Redis, Docker, Firebase, Supabase, Cloudinary, Tailwind CSS .

## EXPERIENCE

---

### Frontend Engineer Intern

WILD FUSION DIGITAL, Nigeria.

OCTOBER 2025 – DECEMBER 2025

- Collaborated with a team of 5 developers to build CatalystAI, an AI-powered career advisory platform, contributing across authentication, onboarding, and AI-driven reporting features
- Reduced sign-up friction by consolidating a 5-step multi-page onboarding flow into a single form, enabling users to reach the core product faster.
- Built a feature that analyzes user assessment responses to generate personalized career path recommendations, surfacing actionable guidance tailored to each user's profile.
- Integrated Paystack into the portal to manage tier upgrades, unlocking premium reports and AI advisory features based on user access levels

**Key Technologies:** Next.js, Firebase, Gemini API, VAPI

### Frontend Engineer (Contract)

TEE'S IMPORT, Nigeria.

NOVEMBER 2024 – DECEMBER 2024

- Designed and implemented a shipping cost calculator to automate fee computation across **200+** product categories, eliminating manual overhead.
- Developed **automated pricing logic** factoring weight, category, route, and currency conversion, enabling real-time, accurate estimates for customers

**Key Technologies:** React.js

## PROJECTS

---

[Search Engine](#) | [github.com/Kenny-204/Search-Engine](https://github.com/Kenny-204/Search-Engine)

- Built a document search engine with a Google-style results page in Java Spring Boot, supporting document upload, full-text indexing, and ranked search results with matched term snippets
- Built an extensible document parsing layer supporting PDF, plaintext, DOCX, XLSX, and PPTX formats
- Implemented a text processing pipeline that tokenizes content, normalizes terms (lowercasing, stopword removal, stemming), and computes TF-IDF scores for indexing
- Stored the inverted index and document metadata in Redis, enabling fast ranked retrieval.
- Implemented a prefix tree (trie) for autocomplete, enabling real-time search suggestions as users type
- Uploaded and served original documents via Cloudinary, decoupling file storage from the search index

[Ludoly](#) | [github.com/Kenny-204/ludoly](https://github.com/Kenny-204/ludoly)

- Built a real-time multiplayer Ludo game supporting 2–4 players with WebSockets (Socket.io) and a server-side game state reducer, ensuring consistent state across all clients
- Implemented a rules engine handling complex piece movement logic including home stretch entry, capture mechanics, and bonus turn triggers on a double-six roll
- Managed per-room game state with Redis, enabling multiple concurrent game sessions with isolated state
- Designed a data-driven board rendering system using a shared coordinate function for both movement calculation and piece positioning
- Built a React frontend with a Socket.io context for real-time event handling, a GameStateProvider for shared state, and localStorage persistence for reconnection support

[Streampot](#) | [github.com/Kenny-204/Streampot-frontend](https://github.com/Kenny-204/Streampot-frontend)

- Engineered a movie streaming app integrating the TMDB REST API, providing access to 1M+ movies with cast, trailers, and genre metadata
- Built the frontend with React, TypeScript, and Tailwind CSS, embedding third-party streams via iframe for in-app playback
- Architected the backend with Node.js, Express, TypeScript and MongoDB, implementing authentication, user watchlists, and search functionality