

# Emmanuel Isenah

## WEB DEVELOPER

emmanuelisenah@gmail.com • +(234)-810-347-6101

Ever since seeing 'Hello world!' appear in the console of my first C++ app in 2012. I knew I was hooked on programming. I have since gained experience ever since in creating end-to-end web applications using Django, PostgreSQL, and Python. I'm an extremely organized and efficient self-starter, with proven ability in web development, complex data structures and algorithms, server architecture, automation, and troubleshooting. Able to work remotely immediately if needed.

## EDUCATION

### Rivers State University

Aug 2018 - Jun 2023 • Port Harcourt, Rivers

Electrical Engineering

## SKILLS

### Languages

Python

Javascript

HTML/CSS

### Frameworks

Django

Bootstrap

Django REST framework

### Database

PostgreSQL

SQLite

### Others

Git

REST

Linux

## EXPERIENCE

### Junior Full-Stack Developer at HouseDeals Africa

Oct 2021 - Apr 2022 • Abuja, Nigeria

I worked as an intern on the company's software front-end design along with the SDLC of the project in collaboration with other peer programmers. My daily workflow consisted of performing error analysis on already existing codebase, assisting with IT issues and building /accessing databases. I, together with my team of peers developed a full Django web application in my time of internship.

## PROJECTS

### Rumble - (Not currently live)

Rumble is an web application aimed to make ordering of food easier and accessible over the internet using web payments and digital currencies. As it stands, the backend consists of a Django server which is built in a manner to meet REST best practices. The server handles authentication, authorization, error validations and efficiently queries postgres DB through ES6 standards. The user interface was designed using plain CSS along with bootstrap. <https://github.com/Armadillidiid/food-delivery-ecommerce>

### CS50 Finance

This is a mini stock market web application I developed where users can buy and sell stock. Not only can the users buy and sell, but they also could manage their portfolios of stocks. It was created by querying IEX API to get real-time stock prices. Hence a small project here , so Flask and SQLite was the preferred choice of framework and database. <https://github.com/code50/86387828/tree/main/finance>