Harrison James

Backend Engineer · Python, Django, PostgreSQL

Lagos, Nigeria · open to remote · harriajames97@gmail.com · craigryy.github.io · linkedin.com/in/harrison-james-bb0839148 · github.com/Craigryy

SUMMARY

Backend engineer and co-founder of Kinomos HQ, where I lead development of two Django products: Kinomos, a business directory with subscription billing, and SeeBlu, a short-video social app. I focus on the parts that must not break (payments, data integrity, access control) and back them with automated tests, code review and CI: 850+ tests on Kinomos, 400+ on SeeBlu. 3 years of building and shipping Python web apps, and a habit of writing up what I learn.

SKILLS

Languages	Python, JavaScript, TypeScript, SQL
Backend	Django, Django REST Framework, Wagtail, Flask, Celery, Django Channels (WebSockets), REST API design
Data	PostgreSQL (incl. connection pooling), Redis, SQLAlchemy, MongoDB, Pandas
Payments	Paystack (cards, recurring billing, bank transfer, webhooks), Kora, OAuth 2.0 (Google, GitHub, Facebook), Web Push
Infrastructure	Docker, Docker Compose, Nginx, GitHub Actions CI/CD, DigitalOcean (Droplets, Managed PostgreSQL, Spaces), Cloudflare (Pages, Turnstile), Render, Terraform, Ansible
Frontend	React, Next.js, Astro, HTML, CSS
Practices	Git (feature branches, pull requests, code review), automated testing, structured logging, technical writing

EXPERIENCE

Lead Developer · AcctTechs, remote — Nov 2026 – Present

AcctTechs, accttechs.com · online store for digital accounts · Django, Wagtail, PostgreSQL, Tailwind CSS, Kora, Docker, DigitalOcean

- Built the frontend from the design mockups (Tailwind CSS, phone-first) and the Django backend: categories, search, cart, Kora checkout in naira verified server-side and by signed webhook, encrypted stock, and delivery on screen and by email once payment is confirmed.
- Configured the email (Zoho Mail inboxes, ZeptoMail for site emails, SPF, DKIM and DMARC), the domain's DNS, and Cloudflare security: Full (strict) SSL, WAF rules, rate limits and Turnstile, with the server reachable only through Cloudflare.
- Built the infrastructure as code: Terraform for DigitalOcean and Cloudflare, Ansible for server setup, Nginx, nightly backups to R2, and deploys that roll back on a failed health check; 430+ automated tests.

Co-founder & Lead Developer · Kinomos HQ Ltd, Lagos — Jul 2025 – Present

Kinomos, kinomos.com · business discovery and directory platform · Django, Wagtail, PostgreSQL, Celery, Redis, DigitalOcean

- Lead the codebase in a two-person engineering team: 950+ commits and 220+ pull requests merged through a develop branch, GitHub Actions CI and code review, with 850+ automated tests.
- Built subscription billing on Paystack for business listings: recurring card charges, automatic retries for declined cards every 30 minutes, bank-transfer payments, a renewal grace period, and Celery jobs for trial-expiry reminders.
- Closed an access-control gap that let users reach vendor analytics through a direct business-detail route, and moved media off the public object-storage domain onto the site's own URLs.
- Diagnosed and fixed production failures: a template syntax error that crashed every forum topic page, comment markup leaking into rendered pages, and PostgreSQL connection-pool errors on DigitalOcean (pooled connections by default, a direct-connection switch for work the pooler can't handle).
- Built full-text business search, community forums (django-machina) and real-time chat (Django Channels).

SeeBlu, seeblu.ng · short-video social app I founded, in testing · Django 5.2, PostgreSQL, Paystack, Docker, Cloudflare Pages

- Split SeeBlu into a static site on Cloudflare Pages and an API on our existing DigitalOcean server and database cluster, so it runs with no new hosting bill. Every page is rendered from the Django templates at build time; sessions travel in a header, not a cookie; HTTPS certificates renew automatically; a strict Content-Security-Policy and Cloudflare Turnstile keep out injected scripts and bots.
- Designed one-time payments with server-side prices, purchase snapshots, row-level locks and database constraints so credits can never be spent twice; each payment is confirmed once, by verify call or signed webhook, checking amount, currency and payer.
- Built the notification pipeline: Web Push sent by a worker woken through PostgreSQL LISTEN/NOTIFY, at high urgency so locked phones ring, with a retry cap so one failing device can't hold up the rest.
- Traced failing Paystack calls to Cloudflare rejecting Python's default User-Agent (error 1010) and fixed it; fitted the app into a 512 MB host by merging background workers into one process.
- Covered billing, messaging, moderation and notifications with 400+ automated tests.

Developer · Scriptures and Power, scripturesandpower.com

Sep 2023 – Present

Christian publishing site built by a two-person team · Wagtail, Django, PostgreSQL, JavaScript, Docker, DigitalOcean

- Built the Events section end to end: Wagtail page models for events (dates, times, location, online-meeting and live-stream details), a month calendar computed on the server with keyboard-accessible days, countdowns to upcoming events and automatic handling of past events.
- Built the Resources section: a Wagtail page whose StreamField holds the resources, with filtering and pagination that only appears when it's needed, plus its page design and a redesigned welcome page.
- Made the site installable as a progressive web app (service worker), added template caching, custom 404 and 500 pages and structured logging.
- Moved deployment off Heroku and set up the team's developer tooling: Makefile, linting, Django Debug Toolbar and Poetry.

Freelance Web Developer · Self-employed, remote

Oct 2023 – Present

- Design and build websites and web apps for businesses with Django, React and Next.js, with attention to speed, security and hosting.

Data Analyst Intern · Heritage Energy Operational Services Ltd, Warri

Jul 2021 – Jan 2022

- Supported the senior data analyst on the analysis of oil and water samples from OML-30 field operations.
- Organised periodic field data in Excel and showed colleagues how to record it consistently; reported progress to senior engineers in an agile team.

PROJECTS

Recipes API · Django REST Framework, Docker, Terraform, AWS, GitHub Actions · github.com/Craigryy/recipes

- Modular Terraform (S3 remote state, DynamoDB locking) provisions VPC, ECS Fargate, ALB, EFS and IAM in one command; the pipeline builds the image, pushes to ECR, applies on merge and destroys on branch cleanup.

Pycam · Django, OAuth 2.0, Docker, Render · github.com/Craigryy/pycam

- Photo editor with Google, GitHub and Facebook sign-in, in-browser editing and sharing back to Facebook.

This portfolio · Astro, Sveltia CMS, GitHub Actions · craigryy.github.io

- Static site with a git-based CMS that publishes on every push; self-hosted fonts after tracing build failures to parallel downloads on a slow network.

EDUCATION

BSc Biochemistry · Delta State University, Abraka
2018 – 2021 · CGPA 4.0 / 5.0

Self-taught in software: The Complete Web Development Bootcamp and The Complete Python Pro Bootcamp (Udemy, 2023)

WRITING

Technical articles on medium.com/@CraigryJames, including:

- Flask vs Django: choosing the right framework (2024)
- Comparing MongoDB, SQLite and Postgres (2023)
- Building a full-stack app with Flask and React (2023)