

MUHAMAD RAVI WICAKSONO, Software & Data Engineer

Jakarta, Indonesia, +6289648951108, ravier0001@gmail.com, porto.raviwicaksono.tech

LINKS

[LinkedIn](#), [Portfolio](#), [Github](#)

PROFILE

I'm a Software & Data Engineer with 6 years of experience, focused on large-scale web scraping, data engineering and backend development. At Hubexo I run 3,000+ production scrapers across the UK, DACH, ANZ, US and Asia on Apache Airflow and AWS, turning raw construction and planning data into complete, customer-ready datasets. I also built the GraphQL API that serves millions of those records to customers, and an AI agent that writes new scrapers around 30x faster. Outside Hubexo I build Invictus, an LP trading tool on EVM networks that does over US\$5M in trading volume a month. I work mostly in Python, Airflow and SQL, use LLMs from OpenAI, xAI Grok, DeepSeek, Gemini and Ollama, and have strong full-stack experience with React.js, Next.js, TypeScript, Node.js, Nest.js, Ruby on Rails and Laravel.

WORK EXPERIENCES

Oct 2025 — Present

Web Scraping, Data & Backend Engineer, Hubexo

Jakarta

Web Scraping & Airflow Orchestration

- Scaled data collection to 3,000+ production scrapers covering planning, construction and tender data across the UK, DACH, ANZ, US and Asia (UK councils, Swiss cantons, Sweden, Australia, New Zealand and Singapore), all orchestrated with Apache Airflow on AWS MWAA.
- Kept daily data delivery reliable by shipping 300+ production fixes across every region (proxy, SSL, CAPTCHA, rate limiting and site changes).
- Cut proxy-related failures on 100+ council scrapers with a rotating proxy downloader that rotates IPs and automatically fails over between Zyte and Bright Data.
- Unlocked Cloudflare / Turnstile-protected sources that the team could not previously scrape, using cloudscraper, curl_cffi and CAPTCHA-solving services.
- Reduced duplicated engineering effort with a shared helper library for ASPX-based sources, now used by multiple developers across the team.
- Replaced manual back-scraping with automated, priority-based batch campaigns triggered through the MWAA CLI, plus tooling that recovers stuck queued or running DAG runs.
- Made one codebase cover daily, scheduled and historical runs by generating Airflow DAGs dynamically (per schedule, lookback window and search type) with task mapping and task groups.
- Gave the team early warning of broken sources with harvest monitoring that publishes CloudWatch metrics per region and source and raises SNS alarms for missing output and parse errors.
- Handled hard-to-scrape portals (ASP.NET ViewState forms, sessions, pagination, JavaScript rendering) with Requests, BeautifulSoup, Selenium / undetected-chromedriver and headless browsers.
- Recovered data locked in PDFs and scanned documents with PyMuPDF, pdfplumber and Tesseract OCR.

AI Scraper Agent (Mandor)

- Sped up scraper development by around 30x with Mandor, an AI agent that turns a portal URL into a production-ready scraper using Claude (Haiku, Sonnet and Opus).
- Removed most manual QA with a multi-agent pipeline (scout, analyser, mapper, config and code generators) whose QA agent runs, validates and self-fixes every scraper it generates.
- Made generated scrapers deployable as-is: they plug into the Airflow Source / Crawl / Parse pattern and ship with field-level schema validation and coverage reports.
- Extended Mandor to PDF-based portals, extracting key-value and table records with pdfplumber.
- Let Mandor get past protected sites through an MCP browser server (SeleniumBase and Camoufox stealth backends, proxy and CAPTCHA handling), containerised with Docker.
- Opened Mandor to the whole team by moving it to AWS EC2 on the Claude API, with no local setup needed.

ETL, Data Engineering & AI

- Delivered clean, enriched, ready-to-ship construction and planning datasets for customer-facing products through config-driven ETL pipelines, one Airflow DAG per task.
- Improved address and location accuracy by using LLMs (OpenAI API) to parse unstructured notices, standardise applicant and site addresses and resolve exact geolocation.
- Found and re-delivered records that never reached customers, using validation scripts that deduplicate, check unique references, reconcile counts and produce daily validation reports.
- Automated delivery to client S3 buckets via cross-account roles, with Bitbucket Pipelines CI deploying to multiple regional MWAA environments.

Backend Development (HarvesterAPI)

- Gave customers and ETL teams a single source for planning applications, decisions, appeals and document metadata through HarvesterAPI, a Nest.js + TypeScript GraphQL API (Apollo Server).
- Improved query performance and reliability over Oracle tables with millions of records by optimising raw SQL (views, MERGE, connection pooling).

- Kept each customer's data isolated with profile-based multi-tenant access per customer and region, plus capped pagination and input validation.
- Turned raw scraped JSON into a consistent, normalised GraphQL schema of 18 types and 100+ fields.
- Eliminated duplicate ingestion with an idempotent SQS FIFO consumer that loads scraper S3 events into Oracle.
- Shipped safely to development, staging and production on AWS Elastic Beanstalk via Bitbucket Pipelines, with Sentry and Winston for error tracking.

Jul 2026 — Present

Fullstack Engineer, Invictus

Remote

- Built Invictus, a concentrated-liquidity (LP) tool on EVM networks that now handles over US\$5M in trading volume a month through a website and a Telegram bot.
- Turned a multi-step manual process into one action by automating approvals, funding swaps and LP minting, with live position pricing and PnL.
- Made PnL and volume stats always match the books, with no reconciliation job, by designing an append-only PostgreSQL ledger with in-transaction rollups.
- Protected users' custodial wallets with AES-256-GCM / HKDF key encryption, per-wallet nonce locking, Telegram approval for web trades and withdrawals, and a strict Content Security Policy.
- Cut RPC load so that many concurrent pool searches cost a single scan, using a Redis cache, a single-flight lock and rate limiting.
- Let the team change fees, per-account pricing and referral payouts without a redeploy through an admin panel, and drove growth with referral tracking and shareable PNG referral cards.
- Built the stack end to end: Node.js, TypeScript, Express, PostgreSQL, Redis and viem on the backend, and Next.js 16, React 19, shadcn/ui and Tailwind CSS on the dashboard.

Jun 2025 — Aug 2025

Fullstack Engineer, Wonodadi Wetan

Pacitan

- Delivered the Wonodadi Wetan village website end to end as the sole engineer, owning UI/UX, the public site, an admin CMS and the REST API, from first commit to production.
- Public Website (Next.js 15, React 19, TypeScript)**
- Gave villagers and visitors one place for village information with 20+ pages: news (berita), agenda, village marketplace (lapak), photo and video galleries, facilities, history, vision and mission, an interactive organisation chart, legal documents and public information.
 - Improved search indexing and social sharing with server-side rendering: TanStack Query prefetching with dehydrate / hydrate, per-page dynamic metadata, and OpenGraph images auto-cropped to 1200x630 through Cloudinary.
 - Made population data easy to understand with an interactive statistics page of bar, pie and doughnut charts (Chart.js) across 6 demographic datasets, including gender, age, education, occupation and religion.
 - Achieved high PageSpeed scores by serving images in modern formats (WebP / AVIF) through Cloudinary auto-format and auto-quality, with responsive sizing, lazy loading and dynamic imports for non-critical components.
 - Improved usability with debounced search, pagination, dark mode, share buttons and a lightbox gallery, and migrated the site from the Pages Router (JavaScript) to the App Router (TypeScript).
- Admin CMS Dashboard (React, Vite, Redux Toolkit)**
- Let village staff publish on their own, with no developer involved, through a custom CMS for news, events, marketplace products, facilities, galleries and population data, with a rich-text WYSIWYG editor, multi-image posts and backdated publishing.
 - Kept the server light by uploading images straight from the browser to Cloudinary bucket storage and saving only the returned URLs.
 - Kept every screen in sync without manual refreshes using RTK Query with tag-based cache invalidation across 14 API services.
 - Secured admin access with cookie-persisted JWT sessions, protected routes, automatic logout on 401 responses and admin-only menus (role-based access).
- Backend REST API (Express.js, Prisma, PostgreSQL)**
- Designed and built a REST API of 95 endpoints across 15 resources, each with CRUD, search and pagination, on a PostgreSQL schema of 18 Prisma models.
 - Made every article, event and product shareable with SEO-friendly unique slugs, with multi-image support per record.
 - Protected write operations with JWT authentication (httpOnly cookie or Bearer token), bcrypt password hashing, owner-or-admin authorisation and a CORS allowlist.
 - Migrated the village population records from CSV files into PostgreSQL with an idempotent Prisma seed script.

Jun 2023 — Jan 2024

Frontend Engineer, MoFE

Jakarta

- Launched the frontend of MoFE, a used-motorcycle marketplace, from scratch with Next.js, TypeScript, Redux, RTK Query and Tailwind CSS, turning Figma designs into a fully responsive UI and email templates.

- Led frontend delivery as the main contributor, writing around 95% of the codebase across 20 pages and 100+ components.
- Gave users real-time insights by integrating an AI-powered price prediction feature with backend services, used on the free price-check page, in the sell form and in header search.
- Made motorcycles easy to find with search and advanced filters (brand, model, year, mileage, price and location), sorting, responsive pagination, and popular, recent and similar listings.
- Simplified selling with a multi-step listing flow (bike details, photos, preview and edit), including multi-image upload with a primary photo and file type and size validation.
- Made sign-up quick with email, Google and Facebook login, password reset and a magic-link account claim flow.
- Gave sellers control of their listings with an account area: My Ads with batch status update and delete, favourites, profile and avatar editing, and public seller profiles.
- Connected buyers and sellers directly through WhatsApp contact links on every product page.
- Gave the business insight into user behaviour with custom Google Analytics events for search filters, listing views, likes and sign-ins.
- Kept data fresh and sessions safe with RTK Query server-side hydration, cache invalidation and automatic logout on expired sessions.
- Achieved strong PageSpeed scores and better search indexing by setting metadata on every page, with server-generated titles and descriptions for product pages.
- Improved publishing with listing previews, a content dashboard and role-based user management.
- Launched Tawar Cepat, a quick-bid motorcycle auction: a landing page with guide and FAQ, a detailed vehicle and seller data form to create auctions, and a secure, token-based auction page where buyers place bids against a live countdown.

Jun 2022 — Jun 2023

Frontend Engineer, Questeon

Singapore

- Launched the frontend of Questeon Creator, a dashboard where creators build location-based quests played in the Questeon mobile app, from scratch with React.js, TypeScript, Redux, RTK Query, Tailwind CSS and Bootstrap.
- Simplified quest creation with a multi-step builder (quest info, intro, places and tasks, conclusion and overview) that saves progress to the server, so creators can pick up drafts where they left off.
- Made quests more engaging by building interactive modules: image puzzles synced with the mobile app, live mobile preview, time tracker, polling, image cropping, multiple-choice image tasks, Spot-it and QR scanning.
- Gave creators 10 task types, each with its own form and preview (short answer, information, action, polling, QR scan with downloadable QR codes, check-in, photo, multiple choice, Spot-it and augmented reality), with media covers, attachments, hints and points.
- Built a Spot-it editor that slices an image into a configurable grid on canvas, so creators mark the differences with a few clicks.
- Let creators design quests on a map by placing places and sub-places on Google Maps with a location radius, and reordering places and tasks with drag and drop.
- Gave creators a clear publishing workflow (draft, on review, rejected, ready and published) with instant status changes through optimistic cache updates, plus shareable quest deep links.
- Gave creators insight into quest performance with analytics tabs for players, recent joins and reviews with a star-rating breakdown.
- Organised data fetching into 48 RTK Query endpoints across 6 API modules, with tag-based cache invalidation, a global loading state and automatic logout on expired sessions.
- Kept the app fast and resilient by lazy-loading task forms and modals, wrapping them in error boundaries, and validating every form with react-hook-form and yup.
- Delivered a seamless cross-device experience and email templates from Figma designs.
- Secured the platform with full authentication flows (sign-up, email verification, password reset and change, protected routes) and role-based user management.

Jan 2023 — Jun 2023

Fullstack Engineer, University Project

Yogyakarta

- Enabled a research team to monitor drilling data live by building a real-time monitoring website with their professor.
- Handled high-frequency data streams with an Express.js backend using Kafka and WebSocket.
- Delivered clear live data visualisation with a responsive Next.js frontend.
- Ensured reliable storage with Prisma ORM on PostgreSQL.

Apr 2022 — Jun 2022

Freelance Frontend Engineer, Owner Digital Agency

Jakarta

- Launched a fully responsive Web3 website from scratch with Next.js, Redux, RTK Query, Tailwind CSS and Bootstrap.
- Enabled on-chain token transactions by integrating Solidity smart contracts on Polygon.
- Shipped token staking and NFT reward minting, plus user management.

Jan 2020 — Jan 2022	Web Developer, Freelance	Semarang
	• Strengthened clients' online presence with custom WordPress landing pages for companies and individuals. • Delivered fast, responsive sites that perform well on desktop and mobile. • Improved clients' search visibility and ranking by applying SEO best practices.	

EDUCATIONS

Aug 2021 — Aug 2025	Bachelor of Computer Systems Networking and Telecommunications, Universitas Gadjah Mada	Yogyakarta
---------------------	--	------------

SKILLS

Web Scraping: Requests, BeautifulSoup, Selenium,undetected-chromedriver, headless browsers, cloudscraper, curl_cffi, rotating proxies (Zyte, Bright Data), CAPTCHA solving, ASP.NET / ViewState forms

Data Engineering: Python, SQL, Apache Airflow (AWS MWAA), ETL / Data Pipelines, Data Cleaning & Validation, PDF extraction & OCR (PyMuPDF, pdfplumber, Tesseract), Geocoding

AI / LLM: Claude (Claude Code, subagents, MCP, Claude API), AWS Bedrock, AI agents, OpenAI, xAI (Grok), DeepSeek, Google Gemini, Ollama (local LLMs), LLM-based structured extraction

Backend: Node.js, Nest.js, Express.js, GraphQL (Apollo), REST APIs, JWT Auth, Ruby on Rails, PHP, Laravel, Prisma, Kafka, WebSocket

Databases: PostgreSQL, MySQL, MariaDB, Oracle, MongoDB, Elasticsearch

CRM & Integrations: CRM Integration, Salesforce, Zendesk, WordPress, WooCommerce, Third-Party Plugin APIs

Frontend: React.js, Next.js, TypeScript, Redux Toolkit, RTK Query, TanStack Query, Vite, Tailwind CSS, Chart.js, React Native, SEO

Cloud & DevOps: AWS (EC2, S3, MWAA, Lambda, RDS, SQS, SNS, CloudWatch, IAM / STS, Bedrock, Elastic Beanstalk), Google Cloud, Vercel, Netlify, Cloudinary, Docker, Bitbucket Pipelines, Sentry

CERTIFICATION

Sept 2025	Data Scientist: Machine Learning Career Path, Codecademy
May 2023	Full-Stack Engineer Career Path, Codecademy
Apr 2022	Tools for Data Science, Coursera
Apr 2022	Front-End Engineer Career Path, Codecademy
May 2021	Web Development Career Path, Codecademy

LANGUAGES

Indonesian (Native)	English (Professional)
------------------------	---------------------------