Portfolio — 2026

Ryan Muhammad Hildan

A software engineer who also works on *computer vision.*

My full name is Ryan Muhammad Hildan, an Informatics Engineering graduate from UNIKOM. I build fullstack web apps with Laravel and Next.js, and image processing systems with OpenCV and FastAPI.

ryanmhildan@gmail.com
github.com/ryanmhildan
Open to full-time roles

For three years I studied while working: drawing machine parts, then keeping the books.

That taught me stock, money, and permissions follow rules you cannot guess at. I bring that into every app: numbers are computed on the server, roles are kept clearly apart, and the flow follows how people actually work.

On the computer vision side, I prefer pipelines that can be explained step by step, then taken out of the notebook and turned into APIs and apps other people can use.

Next, I want to grow as an AI Engineer or Data Scientist: taking data all the way to a model, testing the results honestly, then running it inside systems people use every day.

What I use

Computer vision & ML

OpenCV, scikit-learn, scikit-image, NumPy, K-Means & HSV segmentation, GLCM, SVM

Geospatial

Rasterio, GeoTIFF, OpenDroneMap, Leaflet

Backend

FastAPI, Laravel 11, Pydantic, REST API, PostgreSQL, MySQL

Frontend

Next.js, React, TypeScript, Livewire, Tailwind CSS

Tools

Docker, Git, GitHub, VS Code, Autodesk Inventor

2025 3 months	**Fullstack Web Developer Intern** PT. Sinar Mulia Plasindo Lestari, Bandung	Built the Koperasi POS app with a team: inventory, savings and loans, the member portal, and role-based access.
2022–2025 while studying	**Mechanical Design Engineer & Accounting Staff** CV. Bata Api Teknik, Bandung	Designed technical components in Autodesk Inventor and prepared production drawings. Kept the books, financial reports, and Bills of Materials. What I learned about stock and money flow here went straight into the POS system.
2022–2026 GPA 3.32	**B.Sc. Informatics Engineering** Universitas Komputer Indonesia (UNIKOM), Bandung	Thesis on detecting and mapping rice harvest readiness from drone imagery, using edge computing and GIS.

Rice Harvest Readiness from Drone Imagery

Raw drone photos in, a field map and a harvest verdict out. A three-tier system that runs fully offline on a single field laptop, with no cloud service.

5 cm	12	60%
per pixel	REST endpoints	harvest threshold

- Stitches drone photos into a georeferenced orthophoto at 5 cm per pixel with NodeODM in Docker.

Python · OpenCV · Rasterio · FastAPI · SQLite · Docker · NodeODM · Next.js · Leaflet

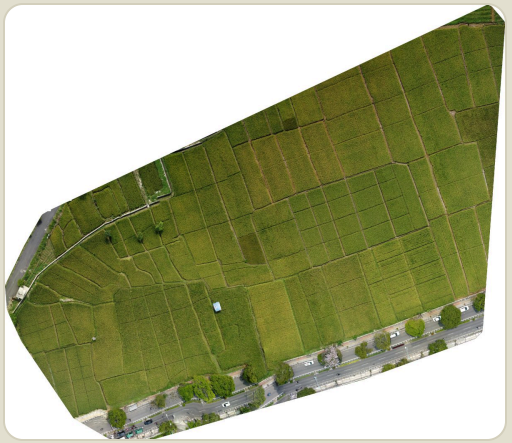
github.com/ryanmhildan/rice-harvest-web

github.com/ryanmhildan/rice_classification_api

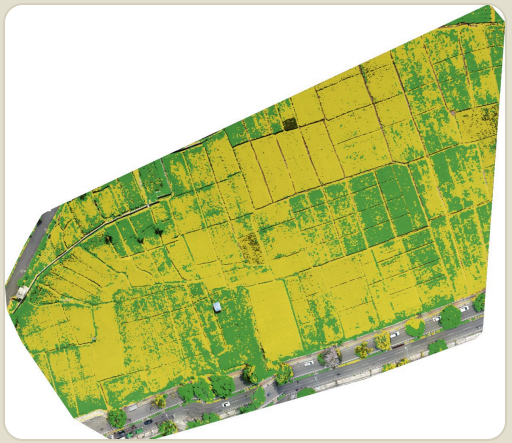
rice-harvest-web.vercel.app



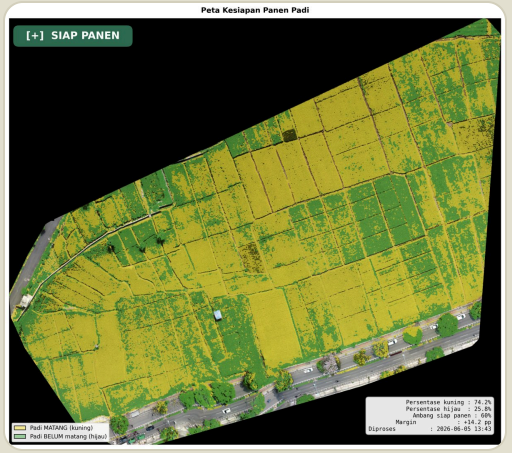
From drone photos to a verdict



Orthophoto stitched by NodeODM



Per-pixel classification: yellow ripe, green unripe



Ready: 74.2% yellow, 14.2 points above the threshold



Not ready: 19.8% yellow, most plots still green

RABmate AI

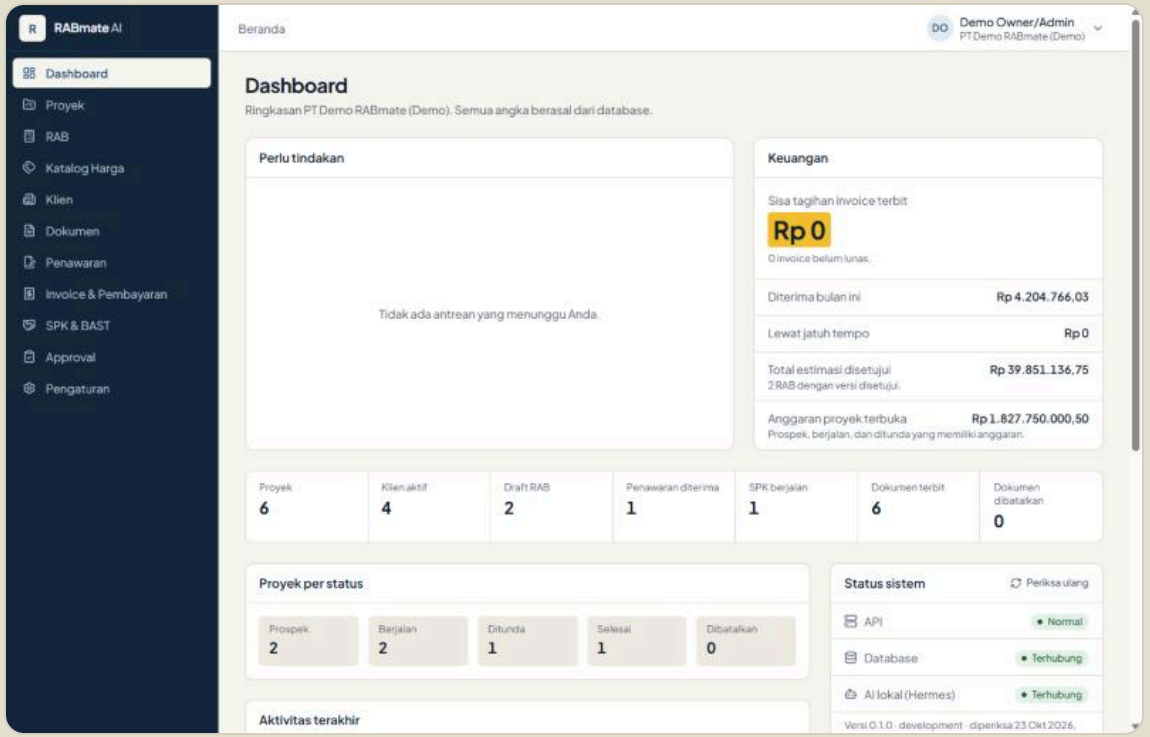
An app for drafting cost estimates (RAB) and the documents that follow, from quotation to handover report. Its assistant runs locally on the user's machine and only reads data, it never changes a number.

534	188	6
backend tests	frontend tests	access roles

- Totals, tax, discounts, and payment terms are computed on the server with Decimal. Issued documents are frozen as snapshots with a SHA-256 hash.
- Every catalog price keeps its supplier, reference, and verification status.
- A local assistant (Hermes Agent + Ollama) drafts line items, extracts items from PDF/XLSX files, and answers project questions with cited sources.

Next.js · FastAPI · PostgreSQL · Ollama · Docker

github.com/ryanmhildan/rabmate-ai

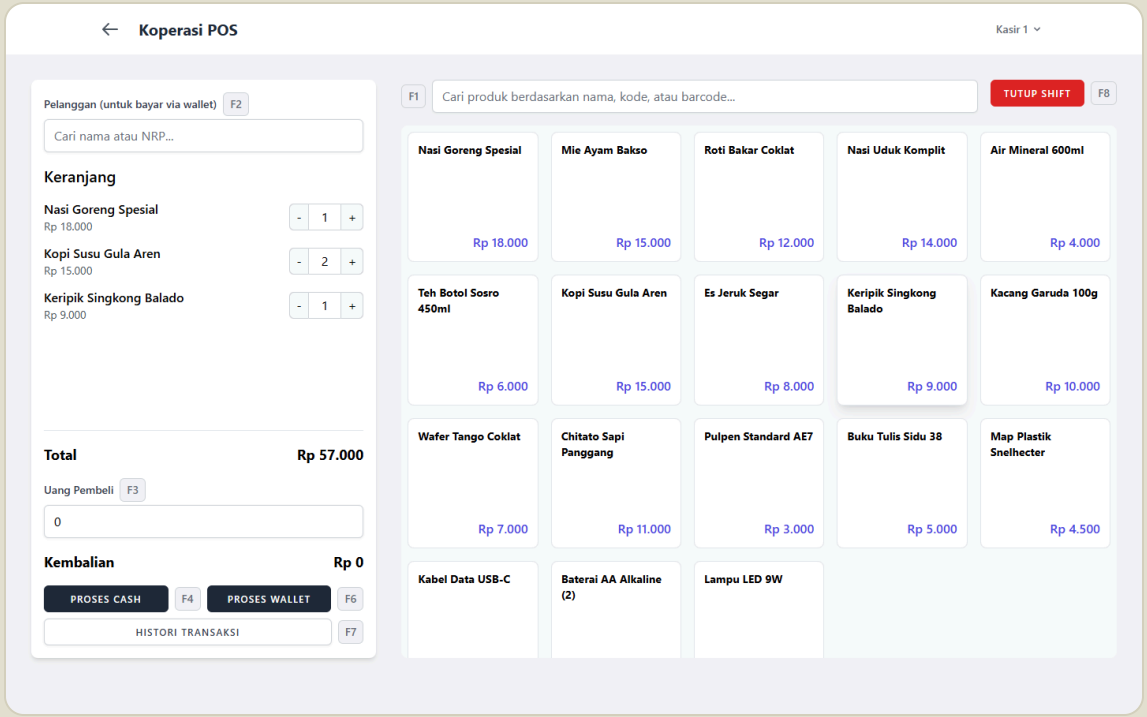


Ryan Muhammad Hildan

Koperasi POS

One system for the till, the stockroom, and a cooperative's finances. Built with a team during a three-month internship, used by managers, cashiers, and members with different permissions.

- A point of sale with per-location shifts, cash or member-wallet payment, and F1–F9 keyboard shortcuts.
- A full inventory module: products, units, locations, stock, stock movements, and goods received notes (GRN).
- Savings, loans, installments, and cash withdrawals, plus a member portal for personal financial history.
- Role-based access with Spatie Permission for managers, cashiers, and members.



Laravel 11 · Livewire 3 · PostgreSQL · Tailwind CSS · Spatie Permission · Bavix Wallet

github.com/ryanmhildan/koperasi-pos

Skin Lesion Classification

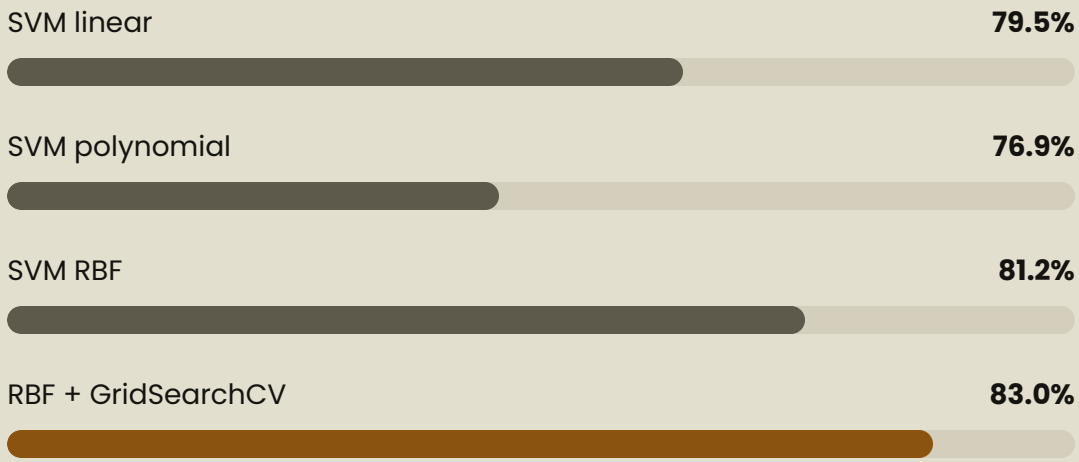
A classic computer vision pipeline that separates benign from malignant lesions across 10,605 images, without deep learning, so every feature can be explained.

10,605
dataset images

83.0%
final accuracy

- Lesion segmentation with K-Means and mask cleanup through morphological operations.
- GLCM texture features (contrast, correlation, energy, homogeneity) and shape features (area, perimeter).
- Compared three SVM kernels, then tuned with 10-fold GridSearchCV: RBF (C=100, gamma=1) went from 81.2% to 83.0% on 1,000 test images.

Python · OpenCV · scikit-image · scikit-learn · Jupyter



Ryan Muhammad Hildan
github.com/ryanmhildan/skin-cancer-classification

Monday Notch

A small notch at the top of a Windows screen. Music with synced lyrics, quick controls, a focus timer, to-dos, and notes, all one hover away.

- Three widget slots on the closed notch: battery, clock, weather, ping, FPS, or to-do.
- Six pages, from a Control Center to a focus timer and quick notes.
- A single .exe with nothing to install, the .NET runtime is bundled.



C# · .NET 8 · Windows 10 / 11

github.com/ryanmhildan/monday-notch

Discord Lyrics Spotify

The lyric line of whatever is playing in Spotify shows up as a Discord status and moves with the song. No Spotify API key and no Premium needed.

- Reads playback from Windows media controls, then matches lyrics from LRCLIB.
- A local setup page in the browser validates the token and stores it on the user's machine.

Node.js · JavaScript · LRCLIB · Windows Media API

github.com/ryanmhildan/discord-lyrics-spotify

How it works

- Checks Spotify playback every second through Windows SMTC and PowerShell.
- Fetches synced lyrics from LRCLIB when the track changes, then schedules each line against the playback position.
- Seeking resyncs the scheduler; pausing or stopping clears the status.

Have a project or a role that fits? Send an email.

Email
ryanmhildan@gmail.com

Phone / WhatsApp
+62 851-7409-6662

LinkedIn
linkedin.com/in/ryanmhildan

GitHub
github.com/ryanmhildan