Green AI Hub Mittelstand — Strategic Portfolio Analysis

Code Maturity, License Compliance, Commercial Viability & Go-To-Market Strategy for 24 Open-Source AI Repositories

AI Portfolio Assessment

March 24, 2026

Independent, externally produced report. This document was prepared independently by **Robert Rettig** and is *not* an official publication of the Green-AI Hub Mittelstand, the BMUV/BMUKN, ZUG gGmbH, or any other organization mentioned herein. It was created with the assistance of AI. All content is provided *without any warranty* as to accuracy, completeness, or timeliness; any liability for errors or for decisions taken on the basis of this document is excluded to the extent permitted by law.

© 2026 Robert Rettig · created with AI assistance · legal notice: https://www.retteten.de/impressum.html

Executive Summary

The **Green AI Hub Mittelstand** is a German Federal Ministry for the Environment (BMUV) initiative that supports small and medium enterprises (SMEs) in applying artificial intelligence for resource efficiency and environmental protection. Its GitHub organization hosts **24 substantive repositories** spanning 14+ industry verticals — from manufacturing quality control and predictive maintenance to LLM-powered field service and carbon tracking.

This report provides a comprehensive assessment across six dimensions:

1. **Code Maturity Audit** — scoring all 24 repos on structure, documentation, dependencies, testing, deployment, and data availability.
2. **Technical Pattern Recognition** — identifying 8 reusable architectural clusters and an extractable shared SDK roadmap.
3. **License Compatibility Matrix** — classifying every repo for SaaS, white-label, fork-and-sell, and internal-tool exploitation.
4. **Value Chain Mapping** — positioning each tool in the industrial value chain with buyer personas and pain-cost estimates.
5. **Product-Market Fit Scoring** — ranking all 24 repos by commercial viability using a weighted 5-criterion framework.
6. **Go-To-Market Strategy** — complete GTM plans for the top 3 commercially viable products.

Key findings:

- **Testing is a near-total blind spot**: 23 of 24 repos have zero functional tests; no CI/CD pipelines exist.
- **17 of 24 repos are MIT-licensed** (fully permissive); 5 are GPL-3.0 (copyleft); 2 have conflicting license declarations.
- **Top 3 products for commercial launch**: *TicketAI* (LLM ticket triage), *CarbonLens* (ML carbon tracking), and *BathGuard* (electroplating monitoring) — all MIT-licensed, with working or near-working demos.
- **Documentation is the strongest dimension** (avg 3.4/5); testing is the weakest (avg 1.1/5).

1. Organization Overview

The **Green AI Hub Mittelstand** is funded by the German Federal Ministry for the Environment, Nuclear Safety and Consumer Protection (BMUV). It aims to help SMEs deploy AI for measurable environmental benefits.

Objectives

- **Sustainability through AI**: Deploying AI technologies to reduce energy and resource consumption in businesses.
- **Knowledge transfer**: Practical AI training for the German *Mittelstand* (SME sector).
- **Pilot projects**: Developing and testing concrete AI applications with measurable environmental impact.

Target Audience

German SMEs that want to:

1. Apply AI-based solutions for more sustainable processes,
2. Reduce their ecological footprint,
3. Benefit from practical training and consulting services.

Services

Service	Description
Consulting	Individual support for identifying AI potential
Workshops	Practical training on Green AI topics
Pilot Projects	Joint implementation of AI solutions with environmental focus
Network	Exchange with other companies and research institutions

Links: Green-AI Hub Mittelstand (official website) — An AI initiative of the German Federal Environment Ministry (BMUV/BMUKN), coordinated by Zukunft — Umwelt — Gesellschaft (ZUG) gGmbH.

2. Repository Inventory

Organization: Green-AI-Hub-Mittelstand
Total Repositories: 25 (including 1 org-config repo `.github`)
Inventory Date: March 23, 2026

#	Repository	Domain	License	Stars	Forks
1	suspekt-demonstrator	Furniture / Visual Inspection	MIT	1	0
2	ai-based-sales-forecasting	Retail / Demand Planning	MIT	0	0
3	AI-based-Vehicle-Energy-Consumption	Fleet / Logistics	MIT	0	0
4	AI-based-Tool-Health-Monitoring	Metalworking / Pred. Maintenance	GPL-3.0	0	0
5	AI-based-Copper-Casting-Optimization	Metals / Process Optimization	GPL-3.0	1	0
6	pcb-error-prediction-toolbox	Electronics / Quality Prediction	MIT	1	0
7	AI-static-roof-assistance	Construction / Structural Eng.	GPL-3.0	1	0
8	XAI-replenishment-analysis	Supply Chain / Material Planning	GPL-3.0	1	0
9	Insole Toolbox (3D Print)	Healthcare / Orthopedics	MIT	11	2
10	LLM-support-ticket-processing	Field Service / NLP	MIT	1	0
11	AI-monitoring-of-3D-printing	Additive Mfg. / QC	MIT	3	0
12	Reduce-Foodwaste-Dataset	Food Retail / Forecasting	MIT	2	1
13	SNN-Conversion-Demonstrator	Green AI / Neuromorphic	MIT	1	0

#	Repository	Domain	License	Stars	Forks
14	SUSPEKT	Furniture / Digital Product Passport	GPL-3.0	1	0
15	AI-driven-turning-machine-optimizer	CNC Machining	MIT	1	0
16	RAG-LLM-Demonstrator	NLP / Knowledge Management	MIT	5	2
17	LSTM-based-Green-AI-Scheduler	Green AI / Scheduling	MIT	1	0
18	AI-textile-quality-assurance	Textiles / Quality Control	MIT	1	0
19	Textile-Design-and-Forecasting	Textiles / Sustainability	MIT	1	0
20	TwinFicient	HVAC / Digital Twin	Disputed	1	0
21	Structural-Capacity-Computation	Construction / Structural Eng.	Disputed	2	0
22	Distributed-Carbon-Tracker	Green AI / MLOps	MIT	5	0
23	Electroplating-AI-Demonstrator	Surface Treatment / Mfg.	MIT	1	0
24	Visual-Quality-Control-Demonstrator	Manufacturing / QA	MIT	5	0
25	.github	Org Config	—	1	0

Repo 25 (`.github`) is a GitHub organization profile page and is excluded from all analyses.

3. Repository Profiles by Domain

3.1 Visual Quality Control (5 repos)

Repos: 1 (suspekt-demonstrator), 11 (AI-monitoring-3D-printing), 14 (SUSPEKT), 18 (AI-textile-quality-assurance), 24 (Visual-QC-Demonstrator)

Shared pattern: All five repos use camera/scanner images + deep learning models to detect product defects. PyTorch is the dominant training framework; YOLO (Ultralytics) appears in three repos; ONNX export is used in two.

1 – suspekt-demonstrator (MIT)
Three-camera inspection and labeling setup for System180 modular furniture components. Uses OAK-1 Max depth cameras with edge inference (Jetson). Clean `src/` layout with `pyproject.toml`. Docker not available; hardware-specific. *Maturity: Advanced Demonstrator.*

11 – AI-based-monitoring-of-3D-printing (MIT)
Detects errors in 3D printing via Faster R-CNN (ResNet50 FPN v2, PyTorch Lightning). Provides scripts for dataset download, training, and inference. Not actively maintained. *Maturity: Demo (scripts only, no web UI).*

14 – SUSPEKT (GPL-3.0)
Digital Product Passport system for System180 furniture. FastAPI backend uses 4 YOLO models with NMS ensemble to classify furniture type and condition from photos. Neo4j graph database for inventory. Docker Compose deployment. **Critical issues:** 5 hardcoded credentials (Neo4j password, HERE API key, auth credentials, session secret). *Maturity: Prototype.*

18 – AI-textile-quality-assurance (MIT)
Cost-effective scanner-based QC test bench for textiles (pilot with Kostler GmbH). Three AI approaches: ResNet50v2 transfer learning, U-Net autoencoder anomaly detection, and persistent homology + random forest. Best dependency license documentation in the organization. Requires Linux + physical scanner. *Maturity: Prototype.*

24 – Visual-Quality-Control-Demonstrator (MIT)
Dual YOLO object-detection models for manufactured product defect checking. Flask web UI for image upload. Pre-trained models AND labeled training data included. Fully pinned dependencies. *Maturity: Demonstrator.*

3.2 Time-Series Forecasting (5 repos)

Repos: 2 (sales-forecasting), 3 (Vehicle-Energy-Consumption), 8 (XAI-replenishment), 12 (Reduce-Foodwaste-Dataset), 17 (LSTM-Green-AI-Scheduler)

Shared pattern: XGBoost is the workhorse model (3 of 4 code repos). pandas + CSV loading is universal. Calendar/weather features repeat across retail forecasting repos.

2 – ai-based-sales-forecasting (MIT)
End-to-end sales forecasting pipeline with Flask backend + Vue frontend. Developed with GROUP SCHUMACHER. Docker support for both components. Requires proprietary sales data. *Maturity: Prototype.*

3 – AI-based-Vehicle-Energy-Consumption (MIT)
Compares fuel vs. electricity consumption across fleet vehicles using tsfresh feature engineering, H2O GBM models, SHAP explainability, and an R Shiny dashboard. Multi-language (R + Python + OSRM). All data NDA-gated. *Maturity: Prototype (data excluded).*

8 – XAI-replenishment-analysis (GPL-3.0)
Dash multi-page dashboard for explainable AI-driven material replenishment planning. Uses XGBoost (via Darts) with conformal prediction. Dockerized with gunicorn. Sample CSV data included. Bilingual DE/EN. **Issue:** Monolithic 3,100+ line `functions.py`. *Maturity: Prototype.*

12 – Reduce-Foodwaste-Dataset (MIT)
Anonymized real-world pastry-sales dataset (Aug 2021–May 2024) from a German confectionery chain. Includes weather, holidays, and scaled sales figures. Kaggle competition companion. **Best data self-containment in the organization.** *Maturity: Dataset/Competition.*

17 – LSTM-based-Green-AI-Scheduler (MIT)
Schedules GPU-intensive AI training during renewable energy peaks using LSTM-based energy mix forecasting. Includes pip-installable scheduler library, pre-trained models, and sample data. **Critical issue:** `requirements.txt` missing all ML dependencies. *Maturity: Prototype (broken install).*

3.3 Anomaly Detection and Process Monitoring (3 repos)

Repos: 4 (Tool-Health-Monitoring), 5 (Copper-Casting-Optimization), 23 (Electroplating-AI-Demonstrator)

Shared pattern: scikit-learn anomaly detectors (Isolation Forest, SVM) and autoencoder-based detection. Sensor data flows through: feature extraction, anomaly scoring, threshold check, alert.

4 – AI-based-Tool-Health-Monitoring (GPL-3.0)
Milling-tool wear monitoring at Bosch Homburg using edge-sensor feature extraction and autoencoders. Dash visualization. Requires proprietary Bosch sensor data. No Docker, no tests. *Maturity: Prototype.*

5 – AI-based-Copper-Casting-Optimization (GPL-3.0)
Docker Compose dashboard for batch comparison and defect detection in copper heat-exchanger tube casting (with MPG GmbH). NiceGUI web app + CNN training scripts. Well-pinned dependencies. Requires proprietary production data. *Maturity: Prototype.*

23 – Electroplating-AI-Demonstrator (MIT)
Fully Dockerized IoT monitoring pipeline: simulated pH/temperature/turbidity sensors, MQTT message bus, InfluxDB storage, scikit-learn anomaly agent (Isolation Forest + SVM), Grafana real-time dashboard with Slack/Teams alerting. Developed with 4Packaging. **One-command deployment** (`docker compose up -d`). Self-contained simulated data. *Maturity: Demonstrator. Best overall deployment readiness.*

3.4 LLM and Knowledge Management (2 repos)

Repos: 10 (LLM-support-ticket-processing), 16 (RAG-LLM-Demonstrator)

Shared pattern: Identical RAG pipeline (document, chunk, embed, store, retrieve, rerank, generate). Both use sentence-transformers for embedding and support local LLM inference.

10 – LLM-assisted-support-ticket-processing (MIT)
RAG-based system that triages field-service tickets into remote-fix vs. on-site-required, including spare-part suggestions. Two-component design: *Ingestor* (pgvector knowledge base from PDFs via docling) and *Coordinator* (FastAPI + RAG). Supports both local Ollama and AWS Bedrock LLMs. Docker Compose. Developed with Fieldcode GmbH. **Best dependency management** (fully pinned). **Issue:** SQL injection via string formatting. *Maturity: Prototype (production-ready API).*

16 – Retrieval-Augmented-Generation-LLM-Demonstrator (MIT)
From-scratch RAG implementation by DFKI showing SMEs that LLMs run locally on a laptop. Flask + SocketIO web interface with side-by-side vanilla-vs-RAG comparison. ChromaDB vector store, web crawler, DeepL translation. 5 stars, 2 forks (highest engagement in org). **Critical issues:** Platform-locked to Apple Silicon (mlx_lm); hardcoded DeepL API key. *Maturity: Demonstrator.*

3.5 IoT and Digital Twins (3 repos)

Repos: 7 (AI-static-roof-assistance), 20 (TwinFicient), 23 (Electroplating-AI-Demonstrator – also in 3.3)

Shared pattern: InfluxDB for time-series storage, Grafana for dashboards, Telegraf for data collection, Eclipse BaSyx AAS for digital twin representation, Docker Compose orchestration.

7 – AI-static-roof-assistance (GPL-3.0)
Multi-language polyglot system (Java/Spring Boot + Dart/Flutter + C#/.NET) for structural analysis of aluminum roofs using Eclipse BaSyx AAS digital twins. Docker stack with Keycloak auth. **Only repo with real JUnit tests.** Ships dummy model (real model confidential to Kalzip). **Issue:** LICENSE file is GPL-3.0 but README states MIT. *Maturity: Prototype.*

20 – TwinFicient (Disputed: MIT file vs. GPL-3.0 README)
Digital twin for KUBLER's energy-efficient hall heating systems. Best Docker stack in the organization: 5 modular Compose files covering BaSyx AAS, InfluxDB, Grafana, Prometheus, Loki. Extensive third-party license documentation. **Blockers:** License conflict; AGPL-3.0 Grafana and SSPL MongoDB in stack; requires proprietary sensor data. *Maturity: Prototype.*

3.6 Green AI Meta-Tools (3 repos)

Repos: 13 (SNN-Conversion-Demonstrator), 17 (LSTM-Green-AI-Scheduler – also in 3.2), 22 (Distributed-Carbon-Tracker)

These three repos address the *meta-concern* of reducing AI's own environmental impact. Together they form a complete Green AI lifecycle: **Plan** (17: schedule training in green windows), **Train efficiently** (13: SNN conversion), **Measure** (22: carbon tracking).

13 – SNN-Conversion-Demonstrator (MIT)
Flask web dashboard comparing Spiking Neural Networks vs. standard ANNs for energy-efficient inference. Runs on Xylo IMU neuromorphic hardware with real-time energy comparison. Cleanest typed Python in the organization. **Blocker:** Requires physical Xylo TestBoard. *Maturity: Demonstrator (hardware-locked).*

22 – Distributed-Carbon-Tracker (MIT)
Pip-installable Python package (`carbontracking`) that measures power usage and CO2 emissions of ML workloads across distributed setups. Client–server architecture with dashboard. Published to PyPI (v0.0.11). Uses ENTSO-E API for real-time grid carbon intensity. Self-contained. **Issues:** Python 3.10 only; committed build artifacts; zero tests. *Maturity: Prototype (PyPI published).*

3.7 AutoML and Specialized Tools (5 repos)

Repos: 6 (pcb-error-prediction), 9 (Insole Toolbox), 15 (turning-machine-optimizer), 19 (Textile-Design), 21 (Structural-Capacity)

6 – pcb-error-prediction-toolbox (MIT)
Professional multi-service architecture: Django + Celery API, React + TypeScript frontend, nginx reverse proxy. Predicts PCB defect rates from Gerber-file features using AutoGluon (AutoML). Developed with Schaltungsdruck Storz + DFKI. **Best deployment in the organization:** 6-service Docker Compose. Requires proprietary data for retraining. *Maturity: Prototype.*

9 – Training-and-Prediction-Toolbox-for-3D-Printable-Orthopedic-Insoles (MIT)
Multi-component toolbox: Django web app, auto-sklearn training, Autodesk Fusion 360 plugin, scan preprocessing. Developed with Herges GmbH. Most-starred repo (11 stars, 2 forks). **Critical security:** Hardcoded Django SECRET_KEY, DEBUG=True, ALLOWED_HOSTS=[“*”]. *Maturity: Prototype.*

15 – AI-driven-turning-machine-optimizer (MIT)
Python scripts for CNC warm-up optimization and part-dimension prediction using TPOT AutoML. Developed with Heismann GmbH. Database and scripts excluded for confidentiality. **Effectively non-functional** without proprietary data. *Maturity: Demo/Research.*

19 – AI-based-Sustainable-Textile-Design-and-Forecasting (MIT)
Next.js web-based recommender for durable textile compositions (with INTEX

GmbH). Minimal AI content. Leftover GitLab template in README. *Maturity: Prototype (limited value).*

21 – Structural-Capacity-Computation (Disputed: MIT file vs. GPL-3.0 README)
Flask web interface for approximate load-bearing capacity calculations (reinforced concrete + steel) per Eurocode standards. Rule-based engineering, no ML. Developed with Concular. Self-contained reference data. *Maturity: Prototype.*

4. Code Maturity Audit

4.1 Methodology

Each repository was inspected on its GitHub main branch for file/folder structure, README content, dependency manifests, test presence, Docker/deployment configurations, data availability, and security posture. Scores range from 1 (worst) to 5 (best) per dimension.

Dimension	Score 1	Score 3	Score 5
Code Structure	Flat/monolithic	Reasonable modules	Clean modular architecture, proper packaging
Documentation	No README or stub only	README with setup + usage	Full API docs, bilingual, architecture diagrams
Dependencies	No manifest, no versioning	requirements.txt, some pinning	Locked deps, small footprint, all maintained
Test Coverage	Zero tests, no CI/CD	Some test files, no CI	Full test suite, CI/CD pipeline
Deploy Readiness	Raw scripts, manual setup	Dockerfile present	Docker Compose, env management, one-command deploy
Data Dependencies	Requires proprietary data	Examples/schema documented	Self-contained or anonymized data included

4.2 Master Scoring Table

#	Repository	Struct.	Docs	Deps	Tests	Deploy	Data	**Avg**	**Risk**
1	suspekt-demonstrator	4	3	4	2	3	3	**3.2**	Medium
2	ai-based-sales-forecasting	4	4	3	2	4	2	**3.2**	Medium
3	Vehicle-Energy-Consumption	2	4	1	1	1	2	**1.8**	High
4	Tool-Health-Monitoring	3	4	2	1	1	1	**2.0**	High
5	Copper-Casting-Optimization	4	4	4	1	4	2	**3.2**	Medium
6	pcb-error-prediction-toolbox	4	4	3	1	5	2	**3.2**	Medium
7	AI-static-roof-assistance	4	4	3	3	4	2	**3.3**	Medium
8	XAI-replenishment-analysis	3	4	3	1	4	3	**3.0**	Medium
9	Insole Toolbox (3D Print)	3	4	2	1	3	2	**2.5**	Medium
10	LLM-support-ticket-processing	4	4	4	1	4	3	**3.3**	Low
11	AI-monitoring-3D-printing	2	3	2	1	1	4	**2.2**	Medium
12	Reduce-Foodwaste-Dataset	2	3	1	1	1	5	**2.2**	Low
13	SNN-Conversion-Demonstrator	4	4	3	1	2	3	**2.8**	Medium
14	SUSPEKT	3	3	3	1	4	2	**2.7**	High
15	turning-machine-optimizer	2	3	2	1	1	1	**1.7**	Medium

#	Repository	Struct.	Docs	Deps	Tests	Deploy	Data	**Avg**	**Risk**
16	RAG-LLM-Demonstrator	3	4	3	1	2	4	**2.8**	Medium
17	LSTM-Green-AI-Scheduler	3	4	2	1	2	3	**2.5**	Medium
18	AI-textile-quality-assurance	3	4	3	1	2	2	**2.5**	Medium
19	Textile-Design-Forecasting	2	2	3	1	1	3	**2.0**	Low
20	TwinFicient	4	4	3	1	5	2	**3.2**	High
21	Structural-Capacity-Comp.	3	3	2	1	1	4	**2.3**	Medium
22	Distributed-Carbon-Tracker	3	3	3	1	2	4	**2.7**	Low
23	Electroplating-AI-Demo	4	3	3	1	5	4	**3.3**	Low
24	Visual-QC-Demonstrator	2	2	4	1	1	4	**2.3**	Low

4.3 Dimension Averages

Dimension	Average	Assessment
Code Structure	**3.0**	Acceptable — most repos have reasonable modularization
Documentation	**3.4**	Good — strongest dimension; consistent README quality
Dependencies	**2.7**	Below average — inconsistent pinning, some missing manifests
Test Coverage	**1.1**	**Critical gap** — 23 of 24 repos have zero functional tests
Deploy Readiness	**2.6**	Bimodal — 7 repos have Docker; 10 repos have nothing
Data Dependencies	**2.8**	Mixed — 6 repos are self-contained; 10 require proprietary data
Overall	**2.6**	

Dimension	Average	Assessment

4.4 Risk Distribution

Risk Level	Count	Repositories
Low	6	LLM-support-ticket (10), Foodwaste-Dataset (12), Textile-Design (19), Carbon-Tracker (22), Electroplating (23), Visual-QC (24)
Medium	14	Repos 1, 2, 5, 6, 7, 8, 9, 11, 13, 15, 16, 17, 18, 21
High	4	Vehicle-Energy (3), Tool-Health (4), SUSPEKT (14), TwinFicient (20)

4.5 Organization-Wide Findings

CRITICAL: Universal Testing Gap

- **23 of 24 repos** score 1/5 on test coverage (only AI-static-roof-assistance has JUnit tests).
- **Zero CI/CD pipelines** across the entire organization.
- **Impact:** The single biggest blocker for commercial adoption; any code change could introduce regressions undetected.
- **Recommendation:** Establish an org-wide GitHub Actions template with minimum pytest/JUnit smoke tests; require passing CI before merge.

HIGH PRIORITY: License and Security Red Flags

Issue	Affected Repos	Severity
GPL-3.0 copyleft (blocks proprietary distribution)	4, 5, 7, 8, 14	High
LICENSE file contradicts README	7 (GPL file, MIT README), 20 (MIT file, GPL README)	High
Hardcoded credentials committed	14 (5 creds), 9 (SECRET_KEY), 16 (DeepL key)	Critical
DEBUG=True in production config	9, 8	Medium
ALLOWED_HOSTS=[“*”]	9	Medium
SQL injection via string formatting	10	Medium

Systemic Issues Across All 24 Repos

Issue	Prevalence	Impact
No test suites	23/24 (96%)	Critical
No CI/CD pipelines	24/24 (100%)	Critical
Hardcoded credentials	6+ repos	High
GPL-3.0 or license conflicts	7 repos	High
Proprietary data excluded	14+ repos	Medium
No Docker deployment	12+ repos	Medium
Single contributor	10+ repos	Medium
Platform lock-in (HW/OS)	5+ repos	Medium

5. Technical Pattern Recognition

Across 24 repositories spanning 8+ domains, **eight dominant technical pattern clusters** emerge — each representing an opportunity for code consolidation and shared-library extraction. Today, every repo reinvents its own image-preprocessing pipeline, Flask boilerplate, Docker Compose skeleton, and data-loading utilities.

By extracting these into a shared **"Green AI Hub SDK"** with pluggable domain adapters, the organization could:

- Reduce per-project boilerplate by 40–60%
- Enforce consistent security, testing, and deployment baselines
- Enable faster onboarding for new pilot partners
- Create a unified "Green AI" brand layer

5.1 Pattern Clusters

Cluster 1 — Visual Quality Control (5 repos)

Repo	Input	Model	Framework	Inference API
(1) suspekt-demonstrator	OAK-1 camera	Custom CNN	PyTorch, ONNX	Flask
(11) AI-monitoring-3D-printing	Camera images	Faster R-CNN	PyTorch	Script
(14) SUSPEKT	Uploaded photos	4x YOLO + NMS	Ultralytics	FastAPI
(18) AI-textile-quality	Scanner images	ResNet50v2 + U-Net + TDA	TF + PyTorch, ONNX	NiceGUI
(24) Visual-QC-Demo	Uploaded images	Dual YOLO	Ultralytics	Flask

Extractable: `gaih-visual-qc` — OpenCV preprocessing, YOLO/ONNX inference wrappers, Flask/FastAPI serving templates, bounding-box visualization.

Cluster 2 — Time-Series Forecasting (5 repos)

Repo	Domain	Models	XAI
(2) sales-forecasting	Retail	Prophet, XGBoost	No
(3) Vehicle-Energy	Fleet	XGBoost (H2O GBM)	No
(8) XAI-replenishment	Supply chain	XGBoost (Darts)	SHAP
(12) Foodwaste-Dataset	Food retail	(dataset only)	N/A
(17) LSTM-Scheduler	Energy grid	LSTM	No

Extractable: **`gaih-timeseries`** — CSV loading with schema validation, weather/calendar feature engineering, XGBoost/Prophet/LSTM trainers, SHAP explainability, conformal prediction.

Cluster 3 — Anomaly Detection (3 repos)

Repo	Sensor Type	Algorithm	Alert
(4) Tool-Health	Vibration/force	Autoencoder	Dash
(18) textile-QA	Scanner images	U-Net autoencoder	NiceGUI
(23) Electroplating	pH, temp, turbidity	Isolation Forest + SVM	Grafana

Extractable: **`gaih-anomaly`** — Isolation Forest, autoencoder, One-Class SVM detectors with configurable thresholds and Grafana/Slack alerting.

Cluster 4 — RAG / LLM Service (2 repos)

Repo	Vector Store	LLM Backend	Ingestion
(10) LLM-ticket-processing	pgvector	Ollama / Bedrock	PDF (docling)
(16) RAG-LLM-Demo	ChromaDB	mlx_lm (Apple)	Web crawler

Extractable: **`gaih-rag-service`** — Document ingestion (PDF/HTML/CSV), pluggable vector stores (ChromaDB/pgvector), pluggable LLM backends (Ollama/Bedrock), cross-encoder reranking. **Highest commercial-value extraction** — RAG-as-a-service is in strong market demand.

Cluster 5 — IoT Monitoring / Digital Twin (3 repos)

Repo	AAS Framework	Time-Series DB	Visualization
(7) roof-assistance	Eclipse BaSyx v2	—	Flutter
(20) TwinFicient	BaSyx v2+v3	InfluxDB	Grafana
(23) Electroplating	—	InfluxDB	Grafana

Extractable: **`gaih-iot-stack`** — Docker Compose template with MQTT, InfluxDB, Grafana, Telegraf, optional BaSyx AAS adapter.

Cluster 6 — Green AI Meta-Tools (3 repos)

Repo	Function	Integration
(22) Carbon-Tracker	CO2 tracking	pip package
(13) SNN-Demo	ANN vs. SNN efficiency	Flask web app

Repo	Function	Integration
(17) LSTM-Scheduler	Green-energy scheduling	pip package

Extractable: **`gaih-green-ai`** — Carbon tracking + energy scheduling + SNN comparison. **This is the organization's unique differentiator.** No other open-source project offers an integrated Green AI toolkit.

Cluster 7 — AutoML Wrapper (3 repos)

Repo	Framework	Data Type
(6) pcb-error-prediction	AutoGluon	PCB Gerber files
(9) Insole Toolbox	auto-sklearn	Pressure plate
(15) turning-machine	TPOT	CNC sensor data

Extractable: **`gaih-automl`** — Unified API over AutoGluon/TPOT/auto-sklearn with ONNX export. Lowest extraction priority (AutoML frameworks already abstract).

Cluster 8 — Web Application Scaffold (14+ repos)

Flask is the dominant web framework (7 repos, 29%). Every Flask app independently reinvents CORS configuration (often insecure), static file serving, template rendering, and error handling.

Extractable: **`gaih-web-template`** — Cookiecutter project with Flask/FastAPI factory pattern, Dockerfile, docker-compose.yml, CI/CD template, pytest scaffold, and `.gitignore`.

5.2 Shared Module Extraction Roadmap

Priority	Module	Effort	Impact	Repos Served
P0	`gaih-web-template` (scaffold)	2 weeks	Fixes security + deployment for 14+ repos	14+
P0	`gaih-green-ai` SDK	4 weeks	Organization's unique differentiator	3 + all future
P1	`gaih-iot-stack` Docker template	2 weeks	One-command Industry 4.0 monitoring	3+
P1	`gaih-rag-service` microservice	6 weeks	Turnkey RAG backend	2+
P2	`gaih-visual-qc` library	6 weeks	Unifies 5 vision repos	5
P2	`gaih-anomaly` library	3 weeks	Reusable anomaly detection	3+
P3	`gaih-timeseries` library	8 weeks	Diverse domain requirements	5
P3	`gaih-automl` wrapper	4 weeks	AutoML already abstracts well	3

5.3 Technology Frequency

ML/DL Frameworks

Technology	Count	Repos
scikit-learn	8	2, 3, 4, 8, 14, 18, 23, 24
PyTorch	6	1, 11, 13, 14, 18, 24
XGBoost	3	2, 3, 8
YOLO (Ultralytics)	3	14, 24, 1
TensorFlow/Keras	2	5, 18
ONNX Runtime	2	1, 18

Infrastructure

Technology	Count	Repos
Docker / Docker Compose	9	2, 5, 6, 7, 8, 10, 14, 20, 23
Flask	7	2, 13, 16, 17, 21, 22, 24
PostgreSQL	3	7, 10, 15
InfluxDB + Grafana	2	20, 23
Eclipse BaSyx AAS	2	7, 20

6. License Compatibility Analysis

6.1 License Primer

MIT License — Most permissive. Grants unrestricted rights to use, modify, and sell. Derivatives can be closed-source. Only obligation: include original copyright notice. No SaaS clause.

GPL-3.0 — Strong copyleft. Derivative works distributed to third parties must also be GPL-3.0 with full source disclosure. SaaS loophole: server-side use without binary distribution does *not* technically trigger copyleft, but this is legally contested.

AGPL-3.0 — Closes the SaaS loophole: network interaction triggers copyleft. Relevant because Grafana, Loki, and Promtail (used in repos 20 and 23) are AGPL-licensed.

SSPL v1.0 — MongoDB's license: offering SSPL software as a service requires open-sourcing your *entire* stack. Not recognized as open-source by OSI. Used in TwinFicient's Docker stack.

6.2 License Inventory

#	Repository	LICENSE File	README States	Effective	Conflict?
1	suspekt-demonstrator	MIT	MIT	**MIT**	No
2	ai-based-sales-forecasting	MIT	MIT	**MIT**	No
3	Vehicle-Energy-Consumption	MIT	MIT	**MIT**	No
4	Tool-Health-Monitoring	GPL-3.0	GPL-3.0	**GPL-3.0**	No
5	Copper-Casting-Optimization	GPL-3.0	GPL-3.0	**GPL-3.0**	No
6	pcb-error-prediction-toolbox	MIT	MIT	**MIT**	No
7	AI-static-roof-assistance	GPL-3.0	MIT	**GPL-3.0**	**Yes**
8	XAI-replenishment-analysis	GPL-3.0	GPL-3.0	**GPL-3.0**	No
9	Insole Toolbox	MIT	MIT	**MIT**	No

#	Repository	LICENSE File	README States	Effective	Conflict?
10	LLM-support-ticket-processing	MIT	MIT	**MIT**	No
11	AI-monitoring-3D-printing	MIT	MIT	**MIT**	No
12	Reduce-Foodwaste-Dataset	MIT	MIT	**MIT**	No
13	SNN-Conversion-Demonstrator	MIT	MIT	**MIT**	No
14	SUSPEKT	GPL-3.0	GPL-3.0	**GPL-3.0**	No
15	turning-machine-optimizer	MIT	MIT	**MIT**	No
16	RAG-LLM-Demonstrator	MIT	MIT	**MIT**	No
17	LSTM-Green-AI-Scheduler	MIT	MIT	**MIT**	No
18	AI-textile-quality-assurance	MIT	MIT	**MIT**	No
19	Textile-Design-Forecasting	MIT	MIT	**MIT**	No
20	TwinFicient	MIT	GPL-3.0	**Disputed**	**Yes**
21	Structural-Capacity-Comp.	MIT	GPL-3.0	**Disputed**	**Yes**
22	Distributed-Carbon-Tracker	MIT	MIT	**MIT**	No
23	Electroplating-AI-Demo	MIT	MIT	**MIT**	No
24	Visual-QC-Demonstrator	MIT	MIT	**MIT**	No

Notes on conflicts: Repo 7: LICENSE file is GPL-3.0 (authoritative), README incorrectly states MIT. Repos 20 and 21: LICENSE file contains MIT text, README states GPL-3.0. In the absence of clarification, the more restrictive license (GPL-3.0) must be assumed.

6.3 Commercial Exploitation Matrix

Legend: Y = Permitted, N = Blocked/triggers copyleft, C = Conditional (legal gray area)

#	Repository	License	SaaS	White-Label	Fork+Sell	Internal	Key Blocker
1	suspekt-demonstrator	MIT	Y	Y	Y	Y	Hardware lock-in
2	sales-forecasting	MIT	Y	Y	Y	Y	Proprietary data
3	Vehicle-Energy	MIT	Y	Y	Y	Y	NDA-gated data
4	Tool-Health	GPL-3.0	C	N	N	Y	Copyleft + proprietary data
5	Copper-Casting	GPL-3.0	C	N	N	Y	Copyleft + proprietary data
6	pcb-prediction	MIT	Y	Y	Y	Y	Needs sample data
7	roof-assistance	GPL-3.0	C	N	N	Y	Copyleft; license conflict
8	XAI-replenishment	GPL-3.0	C	N	N	Y	Copyleft
9	Insole Toolbox	MIT	Y	Y	Y	Y	Security remediation needed
10	LLM-tickets	MIT	Y	Y	Y	Y	SQL injection fix needed
11	monitoring-3D	MIT	Y	Y	Y	Y	Minimal structure
12	Foodwaste-Dataset	MIT	Y	Y	Y	Y	Dataset only
13	SNN-Demo	MIT	Y	Y	Y	Y	Xylo hardware required
14	SUSPEKT	GPL-3.0	C	N	N	Y	Copyleft + 5 leaked creds

#	Repository	License	SaaS	White-Label	Fork+Sell	Internal	Key Blocker
15	turning-machine	MIT	Y	Y	Y	Y	Non-functional w/o data
16	RAG-LLM-Demo	MIT	Y	Y	Y	Y	Apple Silicon only
17	LSTM-Scheduler	MIT	Y	Y	Y	Y	Broken requirements.txt
18	textile-QA	MIT	Y	Y	Y	Y	Linux + scanner required
19	Textile-Design	MIT	Y	Y	Y	Y	Low AI content
20	TwinFicient	Disputed	N	N	N	C	License conflict + AGPL/SSPL
21	Structural-Comp.	Disputed	C	C	C	Y	License conflict
22	Carbon-Tracker	MIT	Y	Y	Y	Y	Python 3.10 only
23	Electroplating	MIT	Y	Y	Y	Y	Hardcoded InfluxDB token
24	Visual-QC-Demo	MIT	Y	Y	Y	Y	No deployment config

Summary: 17 repos are MIT (fully permissive for all models), 5 are GPL-3.0 (internal use only; SaaS conditional on legal review), 2 are disputed (blocked until resolved).

6.4 License Risk Tiers

Tier 1 — Fully Safe for All Commercial Models (17 repos)

MIT-licensed with no downstream copyleft risks. Fork, sell, white-label, or deploy as SaaS with zero copyleft obligation.

#	Repository	Avg Score	Commercial Readiness
10	LLM-support-ticket-processing	3.3	HIGH — clean RAG, production API
6	pcb-error-prediction-toolbox	3.2	HIGH — best Docker deployment
23	Electroplating-AI-Demonstrator	3.3	HIGH — one-command deploy, self-contained
2	ai-based-sales-forecasting	3.2	Medium — needs proprietary data
1	suspekt-demonstrator	3.2	Medium — hardware lock-in
22	Distributed-Carbon-Tracker	2.7	Medium — PyPI published, niche

Tier 2 — Internal Use Only / SaaS with Legal Review (5 repos)

GPL-3.0 licensed. Safe for internal tools. SaaS *may* be safe via the ASP loophole but requires legal sign-off. Cannot be white-labeled or sold as proprietary products.

#	Repository	Avg Score	Constraint
7	AI-static-roof-assistance	3.3	GPL (LICENSE) vs. MIT (README)
8	XAI-replenishment-analysis	3.0	GPL confirmed
5	Copper-Casting-Optimization	3.2	GPL confirmed
4	Tool-Health-Monitoring	2.0	GPL + proprietary data
14	SUSPEKT	2.7	GPL + 5 hardcoded credentials

Tier 3 — Blocked Until License Resolved (2 repos)

Contradictory license declarations. No commercial use until maintainers clarify.

#	Repository	Blocker
20	TwinFicient	MIT file vs. GPL README + AGPL Grafana + SSPL MongoDB
21	Structural-Capacity-Computation	MIT file vs. GPL README (easily resolved)

7. Commercial Viability Assessment

7.1 Value Chain Mapping

#	Repository	Value Chain Stage	Primary Buyer	Pain Cost	WTP Segment
1	suspekt-demonstrator	Output – inspection	QA Engineer	Rework, returns	SME
2	sales-forecasting	Reporting – planning	Ops Manager	Overproduction waste	Mid-market
3	Vehicle-Energy	Reporting – fleet analytics	Fleet Manager	Fuel cost, CO2	Mid-market
4	Tool-Health	Process – pred. maintenance	Maintenance Eng.	Downtime, scrap	Mid-market
5	Copper-Casting	Process – defect detection	Process Engineer	Scrap, energy waste	SME
6	pcb-prediction	Output – yield prediction	QA Engineer	Overproduction	Mid-market
7	roof-assistance	Input – structural design	Structural Eng.	Material waste	Enterprise
8	XAI-replenishment	Reporting – inventory opt.	SC Manager	Overstock costs	Mid-market
9	Insole Toolbox	Process – digital fabrication	Orthotist	Manual measurement	SME
10	LLM-tickets	Process – ticket triage	Service Director	Truck-roll cost	Mid–Ent.
11	monitoring-3D	Process – in-process QC	Production Eng.	Failed prints	SME
12	Foodwaste-Dataset	Reporting – demand forecast	Data Scientist	Food waste	SME
13	SNN-Demo	Reporting – efficiency bench	CTO / ML Eng.	Inference energy	Mid–Ent.
14	SUSPEKT	Output – product passport	Sustainability Off.	EU DPP compliance	SME
15	turning-machine	Process – warm-up opt.	Process Engineer	Energy waste	SME
16	RAG-LLM-Demo	Process – knowledge retrieval	CTO	Info silos	SME–Mid

#	Repository	Value Chain Stage	Primary Buyer	Pain Cost	WTP Segment
17	LSTM-Scheduler	Process – workload sched.	ML Engineer	Training energy	Mid–Ent.
18	textile-QA	Output – fabric inspection	QA Engineer	Defective fabric	SME
19	Textile-Design	Input – material selection	Designer	Short fabric life	SME
20	TwinFicient	Process – energy opt.	Facility Manager	Heating cost	Enterprise
21	Structural-Comp.	Input – structural assess.	Structural Eng.	Over-demolition	Mid-market
22	Carbon-Tracker	Reporting – sustainability	ML Eng. / Sust. Off.	ESG compliance	Mid–Ent.
23	Electroplating	Process – bath monitoring	Process Engineer	Bath contamination	SME
24	Visual-QC-Demo	Output – visual inspection	QA Engineer	Missed defects	SME

Value chain distribution: Process (42%) > Reporting (25%) > Output (21%) > Input (12%). The portfolio is heavily weighted toward factory-floor and quality-gate interventions.

Primary buyer persona: QA Engineer / Production Manager (8 repos) — suggesting the strongest GTM motion is *“AI-powered quality control for manufacturing SMEs.”*

7.2 Product-Market Fit Scoring

Criteria: Problem Urgency (25%), Market Size (20%), Technical Differentiation (20%), Replication Effort (20%), Ecosystem Integration (15%).

#	Repository	Urgency	Market	Tech Diff.	Repl. Effort	Ecosystem	**Score**	**Rank**
10	LLM-ticket-processing	5	5	4	4	5	**4.55**	**1**
22	Carbon-Tracker	5	5	5	3	4	**4.50**	**2**
23	Electroplating-AI	4	3	3	5	4	**3.75**	**3**
6	pcb-prediction	4	3	4	3	4	**3.60**	4
14	SUSPEKT	5	4	4	2	3	3.75	5 (GPL)
17	LSTM-Scheduler	4	4	5	2	4	3.75	6 (broken)
2	sales-forecasting	4	4	2	3	4	3.40	7
8	XAI-replenishment	4	3	3	3	4	3.40	8
20	TwinFicient	4	2	4	2	4	3.20	9
16	RAG-LLM-Demo	3	5	2	2	4	3.15	10

Top 4 repos by PMF are all MIT-licensed. The GPL-3.0 repos cluster in the moderate tier — even if their PMF were higher, the license limits exploitation to SaaS-only (with legal risk) or internal use.

PMF Tier Distribution

Tier	Score Range	Count	Share
Strong PMF	3.5+	4 repos	17%
Moderate PMF	2.5–3.49	10 repos	42%
Weak PMF	below 2.5	10 repos	42%

7.3 Marketing Readiness Audit

Assessment dimensions: Use-Case Narrative, Visual Documentation, Demo Environment, Pilot/Case Study Reference, SEO/Discoverability. Score: 1 (nothing) to 5 (launch-ready).

#	Repository	Narrative	Visuals	Demo	Pilot Ref.	SEO	**Mktg Score**	Key Gap
16	RAG-LLM-Demo	5	3	3	3	3	**3.4**	Apple-only lock-in
9	Insole Toolbox	4	2	3	4	3	**3.2**	Hardware-gated
23	Electroplating	3	4	5	3	1	**3.2**	Needs video content
6	pcb-prediction	4	3	4	3	1	**3.0**	No screenshots
8	XAI-replenishment	4	3	4	3	1	**3.0**	No UI screenshots
10	LLM-tickets	4	3	4	3	1	**3.0**	Needs demo video
12	Foodwaste Dataset	4	2	3	4	2	**3.0**	Needs model showcase
22	Carbon-Tracker	4	2	3	3	3	**3.0**	Needs polished dashboard

Systemic gaps affecting 20+ repos: No quantified business results (100%), no demo video/GIF (100%), low GitHub stars (83%), no GitHub Topics (83+%), no social preview image (100%), technical-only README framing (83+%).

Combined PMF x Marketing x License Verdict

Repo	PMF	Marketing	License	**Verdict**
(10) LLM-ticket-processing	**4.55**	3.0	MIT	**GO** — highest PMF, clean license
(22) Carbon-Tracker	**4.50**	3.0	MIT	**GO** — regulatory tailwind (CSRD)

Repo	PMF	Marketing	License	**Verdict**
(23) Electroplating-AI	**3.75**	3.2	MIT	**GO** — best demo readiness
(6) pcb-prediction	**3.60**	3.0	MIT	GO with work — needs screenshots + metrics
(16) RAG-LLM-Demo	**3.15**	3.4	MIT	GO with work — port to cross-platform
(14) SUSPEKT	**3.75**	2.6	GPL	**BLOCKED** — relicense or abandon
(20) TwinFicient	**3.20**	2.8	Disputed	**BLOCKED** — resolve license first

8. Go-To-Market Strategy

The three highest-PMF, MIT-licensed, launchable repositories form a coherent **"Green AI for SMEs"** commercial portfolio.

8.1 Product 1: TicketAI (LLM-support-ticket-processing)

> *"Resolve more tickets remotely. Dispatch fewer trucks. Cut field service costs."*

Positioning

For service desk managers and field service directors at mid-market equipment manufacturers and IT service providers **who** struggle with high truck-roll rates and slow ticket resolution, **TicketAI** is an AI-powered ticket triage system **that** reads your product manuals, learns from your ticket history, and recommends whether an issue can be fixed remotely or requires an on-site visit — including which spare parts to bring.

Unlike generic AI chatbots (Zendesk AI, Freshdesk Freddy), TicketAI **ingests your actual PDF manuals and wiring diagrams** via RAG, giving technically accurate answers grounded in your specific equipment. **And** it runs entirely on your own infrastructure with local LLMs (Ollama), so **no customer data ever leaves your network**.

Channel Strategy

Channel	Role	Priority
Direct inside sales	Target DACH field-service companies (20–200 technicians)	Primary
SI partners	ServiceNow / Jira SM consultancies	Secondary
AWS Marketplace	"Bedrock-powered" listing for AWS-committed enterprises	Tertiary
Open-source community	MIT repo as top-of-funnel; convert stars to demos	Ongoing

Pricing

Model	Structure	Price Point
Open Core (recommended)	Free core RAG engine + premium connectors, SSO, analytics	Free / 499/mo / 1,999/mo

Model	Structure	Price Point
Per-ticket metered	0.10–0.50 EUR per ticket analyzed	500–5,000/mo
Implementation + subscription	5K–15K setup + 999/mo SaaS	27K–57K/year
Self-hosted license	Annual on-premise license (data sovereignty)	15K–30K/year

Competitive Differentiation

Competitor	TicketAI Advantage
Zendesk AI / Freddy AI	Ingests *your* PDF manuals via RAG — equipment-specific, not generic
AWS Contact Center Intel	Runs locally via Ollama — zero data leaves your premises
ServiceNow Virtual Agent	Open-core, vendor-neutral; 10–50x cheaper for mid-market
Custom GPT / ChatGPT Enterprise	RAG pipeline grounds answers in verified documentation
DIY RAG (LangChain + Pinecone)	Turnkey Docker deployment — no ML team needed

Ideal Pilot Client

Attribute	Profile
Industry	Industrial equipment OEM, HVAC service, elevator maintenance
Size	50–500 employees, 20–200 field technicians
Pain signal	Truck-roll rate >40%; resolution time >48h; >200K EUR/year field service cost
Pilot scope	30-day free pilot, one product line (~500–2,000 tickets)
Success metric	20%+ reduction in unnecessary dispatches = 30K–100K EUR annual savings

GTM Timeline

Weeks	Milestone
1–2	Fix SQL injection; add API auth layer; record demo video
3–4	Build Jira Service Management connector
5–6	Launch landing page; publish blog post; submit to HN/Reddit
7–8	Run 3 pilot engagements (DACH field service companies)
9–10	Collect pilot metrics; write first case study
11–12	Launch Open Core pricing; list on AWS Marketplace

8.2 Product 2: CarbonLens (Distributed-Carbon-Tracker)

> *"Know the carbon cost of every model you train. Report it. Reduce it."*

Positioning

For ML engineering teams and sustainability officers **who** need to measure, report, and reduce the carbon footprint of their AI workloads — especially under EU CSRD and AI Act obligations — **CarbonLens** is a pip-installable carbon tracking toolkit **that** instruments any Python ML pipeline and generates auditable emissions reports.

Unlike CodeCarbon (single-node, no dashboard) or cloud carbon dashboards (AWS/Google), CarbonLens **tracks distributed training across multiple nodes**, correlates with real-time grid carbon intensity via ENTSO-E, and provides a centralized dashboard. **And** it is fully open-source (MIT) — your compliance data stays yours.

Channel Strategy

Channel	Role	Priority
Developer-led growth (PLG)	`pip install carbonlens` to GitHub stars to dashboard upsell	Primary
Content marketing + SEO	Blog: "CSRD for ML Teams", "AI Act energy reporting"	Primary
MLOps platform partnerships	Plugins for MLflow, W&B, Kubeflow, ClearML	Secondary
ESG consulting channel	Big 4 / boutique ESG consultancies advising on CSRD	Tertiary

Pricing

Model	Structure	Price Point
Freemium + Team Dashboard (recommended)	Free pip package + paid hosted dashboard, CSRD exports	Free / 99/mo / 499/mo
Usage-based	0.01–0.05 EUR per tracked training run	50–500/mo
Enterprise annual	Dedicated instance, SSO, audit trail, on-premise	10K–25K/year

Competitive Differentiation

Competitor	CarbonLens Advantage
CodeCarbon (OSS)	Distributed multi-node tracking with centralized dashboard
ML CO2 Impact (web calc)	Measures actual power draw via RAPL/powermetrics, not estimates
AWS Carbon Footprint Tool	Per-experiment, per-model granularity on any infrastructure
Weights & Biases	Purpose-built for carbon: ENTSO-E grid data, CSRD export format

Ideal Pilot Client

Attribute	Profile
Industry	AI consultancy, fintech/insurtech, automotive R&D, pharma ML
Size	5–50 ML engineers training on GPU clusters
Pain signal	CSRD requirements received; sustainability officer asks about AI carbon footprint
Pilot scope	Instrument 5–10 training pipelines for 30 days; first carbon report
Success metric	First per-model carbon report; identify top 3 most carbon-intensive models

GTM Timeline

Weeks	Milestone
1–2	Broaden Python support (3.9–3.12); add test suite
3–4	Build hosted team dashboard (FastAPI + React)
5–6	Write MLflow integration plugin; add CSRD export template
7–8	Publish “CSRD for ML Teams” blog series (3 posts)
9–10	Launch Product Hunt + HN; submit to PyData talk CFP
11–12	Run 5 free pilots with EU ML teams
13	Launch freemium pricing; publish first carbon benchmark report

8.3 Product 3: BathGuard (Electroplating-AI-Demonstrator)

> *"Never lose a batch to a bad bath. AI-powered chemical monitoring for surface treatment."*

Positioning

For process engineers and quality managers at electroplating, anodizing, and wet-chemical surface treatment shops **who** rely on manual pH test strips and periodic lab samples, **BathGuard** is a turnkey IoT + AI monitoring system **that** continuously tracks pH, temperature, and turbidity, detects anomalies in real-time using machine learning, and alerts your team **before** a bath degrades enough to produce defective parts.

Unlike expensive industrial SCADA systems (Siemens MindSphere, Rockwell FactoryTalk) costing 50K–200K EUR, BathGuard **deploys in one Docker command**, costs a fraction, and is purpose-built for surface treatment chemistry. **And** it is open-source (MIT) — no recurring vendor fees.

Channel Strategy

Channel	Role	Priority
Industry trade shows	ZVO, SUR/FIN, NASF — where plating shop owners gather	Primary
Direct pilot program	"Free 30-day BathGuard pilot" targeting DACH shops with >5 baths	Primary
Sensor hardware partner	Bundle with Atlas Scientific probes + Raspberry Pi as "Starter Kit"	Secondary
Chemical suppliers	Partner with Atotech, MacDermid Alpha as value-add	Tertiary

Pricing

Model	Structure	Price Point
Starter Kit (recommended)	Sensor probes + RPi gateway + pre-configured software	1,500–3,000 EUR/bath
Subscription SaaS	Cloud dashboard, alerting, multi-site aggregation	99–299/mo per site
On-premise enterprise	Custom deployment + sensor integrations + model retraining	15K–30K + 5K/year

Competitive Differentiation

Competitor	BathGuard Advantage
Siemens MindSphere	10–50x cheaper; purpose-built for wet chemistry
Rockwell FactoryTalk	Works with any sensor over MQTT, not just Rockwell PLCs
SCADA systems	ML models trained on plating-specific failure patterns
Manual monitoring	Continuous real-time monitoring vs. 4–8 hour sampling gaps
Custom IoT build	Turnkey Docker deployment; pre-trained models included

Ideal Pilot Client

Attribute	Profile
Industry	Electroplating shops, anodizing facilities, PCB wet-process lines
Size	10–100 employees, 3–20 active bath lines
Pain signal	2+ batch failures/quarter; >5K EUR/year manual lab analysis
Pilot scope	1–2 baths, 30 days alongside existing monitoring
Success metric	Detect 1+ anomaly that manual monitoring would have missed

GTM Timeline

Weeks	Milestone
1–2	Externalize credentials; pin Docker images; add tests
3–4	Build MQTT adapter for Atlas Scientific sensor kit
5–6	Assemble 5 pilot sensor kits (RPi + probes + SD card)
7–8	Record demo video; create product one-pager (DE + EN)
9–10	Exhibit at ZVO regional event or Hannover Messe side event
11–14	Run 3–5 free 30-day pilots at DACH plating shops
15–16	Collect pilot data; write first case study
17	Launch Starter Kit pricing; open online ordering

8.4 Portfolio Synergy

The three products form a coherent **"Green AI for SMEs"** portfolio:

Dimension	TicketAI	CarbonLens	BathGuard
PMF Score	4.55 (Rank 1)	4.50 (Rank 2)	3.75 (Rank 3)
Primary Buyer	Service Director	VP Eng. + Sust. Officer	Plant Owner
Revenue Model	Open Core SaaS	Freemium + Dashboard	Starter Kit + SaaS
Price Range	Free–1,999/mo	Free–499/mo	1,500 kit–299/mo
Time to Revenue	10–12 weeks	12–13 weeks	15–17 weeks
Year-1 Revenue (5 customers)	60K–120K EUR	30K–60K EUR	38K–76K EUR
Primary Channel	Direct sales + SI	Developer PLG + content	Trade shows + kits
Regulatory Tailwind	Moderate	**Strong** (CSRD + AI Act)	Moderate

Cross-Sell Motions

From	To	Scenario
BathGuard	CarbonLens	"Now that you monitor baths with AI, do you know the carbon cost? CarbonLens tracks it and generates your CSRD report."
TicketAI	CarbonLens	"Your RAG system processes 5,000 tickets/month. CarbonLens shows the kgCO2 per ticket for your sustainability report."
BathGuard	TicketAI	"When BathGuard detects an anomaly, TicketAI triages the service ticket and recommends remote adjustment vs. technician visit."

Unified Brand Positioning

> **Green AI Hub** — *Open-source AI tools that make your operations smarter and your footprint smaller. Built in Germany. Backed by the Federal Ministry for the Environment.*

This positions the portfolio at the intersection of **operational efficiency** (save money) and **environmental responsibility** (reduce footprint) — exactly where EU regulatory pressure and SME cost optimization meet.

Appendix: Quick Reference

License Quick Reference

License Type	SaaS	White-Label	Fork+Sell	Internal
MIT repos (17)	Yes	Yes	Yes	Yes
GPL-3.0 repos (5)	Conditional	No	No	Yes
Disputed repos (2)	No	No	No	Conditional

Top 5 Repos for Commercial Reuse

Rank	Repository	Score	Strengths	Remediation
1	Electroplating-AI-Demo	3.3	MIT, Docker, self-contained	Add tests; improve docs
2	LLM-support-ticket	3.3	MIT, clean RAG, production API	Fix SQL injection; add tests
3	pcb-error-prediction	3.2	MIT, 6-service Docker Compose	Add tests + sample data
4	AI-static-roof-assist.	3.3	Only repo with tests; best arch.	Resolve GPL/MIT conflict
5	Copper-Casting-Opt.	3.2	MIT, pinned deps, Docker	Add tests + sample data

9. Independent Consultant Opportunity Map

This chapter reframes the 24-repo portfolio through the lens of an **independent AI developer-consultant** with expertise in marketing, business value chains, and cross-domain AI products targeting European SMEs: *What can I build, sell, or advise on — starting today?*

9.1 Opportunity Matrix

Axes: X = Ease of Commercialization (left = hard, right = easy) | Y = Market Impact Potential (bottom = niche, top = broad).

Each repo is placed according to its combined PMF score, license freedom, deployment readiness, and data self-containment (X-axis) versus addressable market size, regulatory tailwind, and buyer urgency (Y-axis).

```
Market Impact
(broad)
   5 |                                        [10] TicketAI
     |                           [22] CarbonLens
     |              [6] pcb-pred            [2] sales-fcst
   4 |  [14] SUSPEKT*       [8] XAI-repl
     |         [20] TwinF*     [17] LSTM-Sched
     |  [7] roof*                    [16] RAG-Demo
   3 |  [4] Tool-Health*          [23] Electroplating
     |  [5] Cu-Cast*         [9] Insole
     |         [21] Struct*      [13] SNN-Demo
   2 |  [3] Vehicle-E             [24] Visual-QC
     |                    [11] 3D-print-mon
     |  [15] turning-m   [18] textile-QA
   1 |                    [12] Foodwaste-DS   [19] Textile-D
     +----+----+----+----+----+----+----+----+----+----+
     1         2         3         4         5
                   Ease of Commercialization
                            (easy -->)

 * = GPL-3.0 or disputed license (blocks white-label / fork-and-sell)
```

Quadrant Summary

Quadrant	Label	Repos	Strategy
Top-Right (easy + high impact)	**Stars**	10, 22, 2, 23	Productize immediately — SaaS or Starter Kit

Quadrant	Label	Repos	Strategy
Top-Left (hard + high impact)	**Bets**	14, 20, 6, 7, 8, 4, 5, 17	Invest in remediation (license, tests, data) then productize
Bottom-Right (easy + low impact)	**Cash Cows**	24, 9, 13, 16, 12, 19	Package as consulting accelerators or training material
Bottom-Left (hard + low impact)	**Avoid**	3, 15, 11, 18, 21	Reference only; do not invest unless client-funded

9.2 Service Packages

Package A — Quick Audit and Integration (Consulting)

> *"We assess your operations, match you with proven Green AI tools, and integrate one into your workflow — in 4 weeks."*

Attribute	Detail
What	2-day on-site assessment + tool matching + 2-week integration sprint
Deliverable	Gap analysis report, tool recommendation, working proof-of-concept on client infrastructure
Repos Used	Any of the 24 — selected per client domain. Most common: 23 (Electroplating), 10 (TicketAI), 2 (sales-forecasting), 8 (XAI-replenishment)
Target Client	Manufacturing SMEs (50–500 employees) exploring AI for the first time; sustainability officers needing quick ESG wins
Pricing	5,000–15,000 EUR per engagement (fixed price)
Margin Driver	Reusable assessment templates + pre-configured Docker stacks = 60%+ gross margin after 3rd engagement
Upsell Path	Package B (productize the PoC) or retainer for maintenance and model retraining

90-day capacity: 3–4 engagements = 15K–60K EUR revenue.

Package B — Turnkey SaaS Product (Productization)

> *"We take a proven open-source Green AI tool, harden it, host it, and sell it as a managed service — you get a branded product without building from scratch."*

Attribute	Detail
What	Fork an MIT-licensed repo, fix security/testing gaps, add multi-tenant auth + billing, deploy on managed cloud
Product Candidates	**TicketAI** (repo 10): LLM ticket triage SaaS
Target Client	DACH mid-market companies (200–2,000 employees) with recurring operational pain
Pricing	Open-core: Free tier + 499–1,999 EUR/mo per team/site
Investment Required	4–8 weeks solo development for MVP; 2K–5K EUR cloud infrastructure/month at scale
Margin Driver	80%+ gross margin on SaaS subscriptions after infrastructure costs

Attribute	Detail
Key Risk	Single-developer dependency; mitigate with IaC (Terraform) and CI/CD from day 1

90-day target: 1 product MVP live + 2–3 pilot customers on free tier.

Package C — White-Label Platform (Licensing)

> *"We license a ready-made AI platform that your team can rebrand, customize, and deploy to your own customers."*

Attribute	Detail
What	Pre-packaged, white-label-ready version of a Green AI Hub tool with customization hooks, documentation, and support SLA
Product Candidates	**Visual QC Platform** (repos 1+14+24 combined): unified defect detection for any manufacturing vertical
Target Client	System integrators, industry consultancies, and OEMs who resell to their own SME customers
Pricing	15K–50K EUR setup license + 3K–8K EUR/year support and updates
Investment Required	8–12 weeks to build white-label layer (theming, config, multi-tenant)
Margin Driver	License revenue is 90%+ margin; support retainer covers ongoing costs
Prerequisite	Must resolve GPL conflicts for repos 14, 20; only MIT repos in initial offering

90-day target: Platform architecture defined + 1 LOI from a system integrator partner.

9.3 Ninety-Day Action Plan

Phase 1 — Foundation (Days 1–30)

Week	Action	Deliverable	Revenue
1	Fork top-3 MIT repos (10, 22, 23); set up private repos with CI/CD (GitHub Actions + pytest)	3 stabilized forks with passing tests	—
2	Fix critical security issues: SQL injection in repo 10, hardcoded credentials in repo 23, Python version lock in repo 22	Security-clean branches	—
3	Build consulting assessment template (reusable across Package A clients): intake questionnaire, tool-matching matrix, PoC checklist	Assessment Kit v1.0 (PDF + Notion)	—
4	Launch minimal web presence: one-page site + 3 product landing pages (TicketAI, CarbonLens, BathGuard) + LinkedIn announcement	Live website + first inbound leads	—

Key milestone: By Day 30, you have 3 hardened forks, a repeatable consulting process, and a public presence.

Phase 2 — First Revenue (Days 31–60)

Week	Action	Deliverable	Revenue
5	Outreach: contact 20 DACH field-service companies (TicketAI targets); 10 ML teams (CarbonLens targets); 5 plating shops (BathGuard targets)	35 outreach emails/calls	—

Week	Action	Deliverable	Revenue
6	Run first Package A engagement (consulting audit) — ideally a warm contact from a previous network	Gap analysis report + PoC deployed	5K–15K EUR
7	Deploy TicketAI MVP on managed cloud (Railway/Render/Hetzner) with basic auth + Stripe billing stub	Live SaaS instance with free-tier onboarding	—
8	Publish first content piece: "How German SMEs Can Cut Field Service Costs 30% with Open-Source AI" (LinkedIn + blog)	1 long-form article + social amplification	Inbound leads

Key milestone: By Day 60, you have first consulting revenue (5K–15K EUR) and a live SaaS MVP accepting signups.

Phase 3 — Scale Signals (Days 61–90)

Week	Action	Deliverable	Revenue
9	Run 2 more Package A engagements (pipeline from Phase 2 outreach)	2 delivered audits + PoCs	10K–30K EUR
10	Onboard 3–5 free-tier pilot users for TicketAI SaaS; instrument usage analytics (PostHog/Plausible)	Active pilot users with usage data	—
11	Write "CSRD Compliance for ML Teams" guide featuring CarbonLens; submit talk proposal to PyData/MLOps Community	Guide published + CFP submitted	Inbound leads
12	Review Phase 1–3 metrics; decide: double down on Package A (consulting) or Package B (SaaS) based on pipeline	90-day retrospective + Q2 plan	—

Key milestone: By Day 90, you have 15K–45K EUR consulting revenue, a SaaS MVP with pilot users, content-driven inbound pipeline, and a data-informed decision on where to invest Q2.

90-Day Financial Summary

Revenue Stream	Conservative	Optimistic
Package A consulting (3 engagements)	15,000 EUR	45,000 EUR
Package B SaaS (free tier, no revenue yet)	0 EUR	0 EUR
Package C licensing (pipeline only)	0 EUR	0 EUR
Total 90-day revenue	**15,000 EUR**	**45,000 EUR**
Costs (cloud, tools, travel)	2,000–4,000 EUR	3,000–6,000 EUR
Net	**11,000–13,000 EUR**	**39,000–42,000 EUR**

10. Risk and Dependency Register

10.1 Top 10 Risks

Risk 1 — Near-Zero Test Coverage

Attribute	Assessment
Category	Technical Debt
Description	23 of 24 repos have zero functional tests. No CI/CD pipelines exist anywhere. Any modification — bug fix, feature addition, dependency update — risks silent regressions.
Likelihood	**HIGH** — regressions are near-certain the moment code is changed.
Impact	**HIGH** — a shipped regression in a client deployment destroys trust and creates liability.
Mitigation	Before touching any repo, add a minimum smoke-test suite (5–10 tests covering critical paths). Set up GitHub Actions CI. Budget 2–3 days per repo for test scaffolding. For the 3 priority repos (10, 22, 23), this is built into the 90-day plan (Phase 1, Week 1).

Risk 2 — Government Funding Sunset

Attribute	Assessment
Category	Funding / Continuity
Description	The Green AI Hub Mittelstand is a BMUV-funded initiative. Government projects have fixed funding cycles (typically 3–5 years). When funding ends, maintainers (often research staff) move on. Repos go unmaintained.
Likelihood	**HIGH** — this is the normal lifecycle of publicly funded R&D projects.
Impact	**HIGH** — unmaintained upstream means security patches, dependency updates, and bug fixes fall entirely on downstream users (i.e., you).
Mitigation	Fork all repos you depend on into your own organization from Day 1. Treat upstream as “inspiration, not dependency.” Build your own CI/CD, your own release cadence, your own security patching process. The MIT license explicitly permits this.

Risk 3 — Hardcoded Credentials and Security Vulnerabilities

Attribute	Assessment
Category	Security / Reputational
Description	At least 6 repos contain hardcoded credentials: Neo4j passwords (repo 14), Django SECRET_KEY (repo 9), DeepL API key (repo 16), InfluxDB tokens (repo 23), auth credentials (repo 14), session secrets (repo 14). Repo 10 has SQL injection via string formatting.
Likelihood	**HIGH** — the credentials are committed in plain text on public GitHub.
Impact	**HIGH** — if any credential is still active, it is already compromised. Deploying code with these patterns to a client creates direct liability.
Mitigation	Rotate all discovered credentials immediately (notify upstream maintainers). In your forks: externalize all secrets to environment variables, add `.env.example` templates, add pre-commit hooks (e.g., `detect-secrets`) to prevent future leaks. Fix the SQL injection in repo 10 with parameterized queries.

Risk 4 — License Ambiguity (GPL Contamination)

Attribute	Assessment
Category	Legal / Commercial
Description	5 repos are GPL-3.0 (blocking white-label and fork-and-sell). 2 repos (20, 21) have contradictory license declarations (MIT file vs. GPL-3.0 README). Repo 20 additionally includes AGPL-3.0 (Grafana) and SSPL (MongoDB) dependencies that could "infect" the distribution.
Likelihood	**MEDIUM** — only relevant if you use these specific repos commercially.
Impact	**HIGH** — GPL violation lawsuits (even meritless ones) are expensive and brand-damaging. Using AGPL Grafana in a SaaS without compliance triggers copyleft.

Attribute	Assessment
Mitigation	Restrict initial commercial offerings to the 17 MIT-licensed repos exclusively. For repos 20 and 21, request written clarification from maintainers before any use. Replace AGPL Grafana with Apache-2.0 alternatives (e.g., Apache Superset, Metabase AGPL-aware edition) in your forks. Never mix GPL code into MIT-licensed products.

Risk 5 — Low Community Adoption Signal

Attribute	Assessment
Category	Market / Credibility
Description	Average stars per repo: 1.8. Average forks: 0.3. No repo has more than 11 stars. No external contributors outside the core project team. No issues or discussions from the community.
Likelihood	**HIGH** — these are niche research outputs, not community-driven projects.
Impact	**MEDIUM** — low stars reduce credibility when pitching to technical buyers ("If it were good, wouldn't it have stars?"). Also means no community will fix bugs for you.
Mitigation	Do not rely on upstream community. In your marketing, reframe the narrative: "Built by German federal researchers, hardened by us for production." Add GitHub Topics, social preview images, and demo GIFs to your forks to attract organic stars. Publish benchmark results and blog posts to build credibility independently.

Risk 6 — Proprietary Data Dependencies

Attribute	Assessment
Category	Technical / Commercial
Description	14 of 24 repos require proprietary data from the original pilot partner (Bosch, Fieldcode, Kostler, MPG, etc.). Models cannot be retrained, demos cannot be run, and the code cannot be validated without this data.
Likelihood	**HIGH** — the data will never be released (NDA-bound).
Impact	**MEDIUM** — limits which repos can be turned into products. Self-contained repos (12, 22, 23, 24) are fine; data-locked repos (3, 4, 5, 15) are effectively non-functional.

Attribute	Assessment
Mitigation	Focus commercial efforts on the 6–8 self-contained repos. For data-locked repos, offer Package A consulting (bring your own data) where the client provides their own production data and you adapt the pipeline. Create synthetic datasets for demo purposes where feasible (e.g., synthetic sensor data for repo 4).

Risk 7 — EU AI Act Compliance Gaps

Attribute	Assessment
Category	Regulatory
Description	The EU AI Act (effective August 2025, enforcement from August 2026) classifies AI systems by risk tier. Several repos may fall under "high-risk" categories: safety-critical structural analysis (repos 7, 21), medical device adjacent (repo 9, orthopedic insoles), and worker-safety monitoring (repo 4). High-risk systems require conformity assessments, risk management systems, human oversight, and technical documentation.
Likelihood	**MEDIUM** — depends on how the system is deployed and marketed.
Impact	**HIGH** — non-compliance with a high-risk classification triggers fines up to 15M EUR or 3% of global revenue. Even "limited risk" systems require transparency obligations.
Mitigation	Conduct an AI Act risk classification for each product before launch. For the top 3 products (TicketAI, CarbonLens, BathGuard): TicketAI is likely "limited risk" (transparency required). CarbonLens is likely "minimal risk" (monitoring tool). BathGuard could be "high risk" if positioned as a safety system — frame it as advisory/monitoring, not autonomous control. Add AI Act compliance documentation to each product.

Risk 8 — Single-Developer / Bus-Factor Risk

Attribute	Assessment
Category	Operational

Attribute	Assessment
Description	10+ repos have a single contributor. As an independent consultant building on these, you become the sole maintainer of your forks. If you are unavailable (illness, other projects), client deployments have zero support.
Likelihood	**MEDIUM** — inherent to solo consulting.
Impact	**HIGH** — a production outage with no one to fix it ends client relationships.
Mitigation	Automate aggressively: CI/CD, automated dependency updates (Dependabot/Renovate), health-check endpoints, uptime monitoring (UptimeRobot), and automated alerting. Document everything as if a contractor needs to take over tomorrow. By Month 6, budget for a part-time junior developer or DevOps contractor for Package B SaaS operations.

Risk 9 — GDPR and Industrial Data Privacy

Attribute	Assessment
Category	Regulatory / Legal
Description	Several repos process data that may be GDPR-relevant: repo 10 (customer support tickets may contain PII), repo 9 (foot-scan biometric data), repo 2 (sales data traceable to individuals), repo 12 (sales with location data). Industrial sensor data (repos 4, 5, 23) may be trade secrets under the EU Trade Secrets Directive.
Likelihood	**MEDIUM** — depends on what data clients feed into the system.
Impact	**HIGH** — GDPR fines up to 20M EUR or 4% of global turnover. Trade secret mishandling creates civil liability.
Mitigation	Add data processing agreements (DPA) to all client contracts. Implement PII detection and redaction in TicketAI's ingestion pipeline. For BathGuard, ensure sensor data stays on client infrastructure (edge deployment option). Never store client production data on shared infrastructure without explicit consent. Include GDPR compliance checklist in Package A assessment template.

Risk 10 — Open-Source Attribution Failures

Attribute	Assessment
Category	Legal / Reputational
Description	MIT license requires preserving copyright notices. When combining code from multiple repos into a product (e.g., Package C white-label platform merging repos 10+16), attribution chains become complex. Missing attribution violates the license and damages credibility in the open-source community.
Likelihood	**LOW** — MIT is the most permissive license; attribution is easy to comply with.
Impact	**MEDIUM** — unlikely to trigger lawsuits, but open-source community backlash (Twitter/HN) can damage brand for a consultancy that positions itself as open-source-friendly.
Mitigation	Maintain a machine-readable `NOTICE` file in every product listing all upstream repos, their licenses, and copyright holders. Use SPDX identifiers. Run `license-checker` or `scancode-toolkit` in CI to catch any unlicensed transitive dependencies. Include attribution in product About/Legal pages.

10.2 Risk Heat Map

Risk	Likelihood	Impact	**Severity**	Priority
R1: Zero test coverage	HIGH	HIGH	**CRITICAL**	Immediate
R2: Funding sunset	HIGH	HIGH	**CRITICAL**	Immediate
R3: Hardcoded credentials	HIGH	HIGH	**CRITICAL**	Immediate
R4: License ambiguity (GPL)	MEDIUM	HIGH	**HIGH**	Phase 1
R7: EU AI Act compliance	MEDIUM	HIGH	**HIGH**	Phase 1
R8: Single-developer / bus factor	MEDIUM	HIGH	**HIGH**	Phase 2
R9: GDPR / data privacy	MEDIUM	HIGH	**HIGH**	Phase 2
R5: Low community adoption	HIGH	MEDIUM	**MEDIUM**	Ongoing
R6: Proprietary data lock	HIGH	MEDIUM	**MEDIUM**	Ongoing
R10: Attribution failures	LOW	MEDIUM	**LOW**	Ongoing

10.3 Risk Mitigation Cost Summary

Action	Covers Risks	Effort	Cost
Add test suites + CI/CD to top 3 repos	R1, R8	6–9 days	0 EUR (labor only)
Fork repos to own GitHub org	R2	1 day	0 EUR
Credential rotation + secret externalization	R3	2–3 days	0 EUR
Legal review of GPL/AGPL/SSPL exposure	R4	1 day	500–2,000 EUR (lawyer)
EU AI Act risk classification (top 3 products)	R7	2–3 days	0–1,000 EUR (template)
GDPR DPA template + PII redaction pipeline	R9	3–5 days	0–500 EUR (template)

Action	Covers Risks	Effort	Cost
Attribution NOTICE files + CI license scanning	R10	1 day	0 EUR
Marketing: GitHub Topics, social previews, blog posts	R5	Ongoing	0 EUR
Total pre-launch risk remediation	**R1–R10**	**16–23 days**	**500–3,500 EUR**

The total cost of comprehensive risk mitigation (500–3,500 EUR + 16–23 days of effort) is negligible compared to the 15K–45K EUR expected 90-day consulting revenue. Every critical and high-severity risk can be addressed in Phase 1 of the action plan.

Analysis completed March 24, 2026