# Luka Pokrajac

**Belgrade, Serbia**
**Gmail:** lukapokrajac04@gmail.com
**GitHub**: github.com/LukaPokrajac
**LinkedIn:**linkedin.com/in/luka-pokrajac-4304ab1a5/

## Experience

**Iritel | Production** Technician | *June 2025 – May 2026*

- Assembled and soldered power-electronics components for industrial equipment while maintaining strict quality-control standards.

**Mei-Ta | Maintenance** Technician | *August 2023 – April 2024*

- Programmed, configured, and operated 6-axis CNC grinding machines for high-volume production of precision cutting tools.
- Optimized machining parameters and process settings to improve tool life, dimensional accuracy, and production throughput.
- Worked within an industrial manufacturing environment using automated production equipment and quality-control procedures.

## Education

**University of Belgrade – School of Electrical Engineering (ETF)** *2024 – 2025*

***Technical School Obrenovac – Electrical Technician for Process Control*** *2019 – 2023*

## Projects

**IoT Telemetry Pipeline:** github.com/LukaPokrajac/iot-telemetry-pipeline

- Built a **real-hardware** telemetry pipeline for a mushroom fruiting chamber: an **ESP32**-C6 (**ESPHome**) publishes 7 environmental **sensors** (air/soil temperature, humidity, soil moisture, pressure, lux) over **MQTT**.
- Normalized heterogeneous sensor topics into a single canonical event schema and decoupled ingestion from storage with a **Kafka** stage, so the producer never blocks on the database.
- **Containerized** Kafka (**KRaft** mode) and **Grafana** with an auto-provisioned **PostgreSQL** data source for one-command setup.
- Wrote custom PlatformIO/C++ firmware for an ESP32-CAM (GC2145), converting RGB565 frames to JPEG in software where hardware JPEG wasn't available.

**Wear Index ETL Pipeline:** github.com/LukaPokrajac/wear-index-pipeline

- Designed and implemented a fully automated hourly **ETL pipeline** that ingests live weather data from the Open-Meteo **API** and computes a custom "feels-like" temperature index using rolling-window averages and wind-chill logic.
- Engineered **idempotent incremental** loads with 3-hour overlap and **PostgreSQL** UPSERT (ON CONFLICT) to ensure zero data duplication while maintaining **low latency**.
- **Orchestrated** the workflow with hourly **Airflow DAGs**, using **BashOperator** to isolate the Airflow runtime from ETL dependencies and to auto-refresh **materialized view**s for instant querying.
- Containerized two separate stacks with **Docker** Compose sharing a network; enabled one-command local setup via **Makefile**.
- Deployed production instance on **AWS EC2** with persistent PostgreSQL storage on **RDS**

**Real-Time Telemetry Streaming Pipeline:** github.com/LukaPokrajac/spacecraft-telemetry-pipeline

- Built a mission-control-style **streaming data pipeline** ingesting simulated spacecraft telemetry at 1 event/sec through a full `Python → Kafka → Postgres → Streamlit` stack.
- Implemented a Python **producer/consumer** pattern with **Kafka** for real-time stream transport, including data validation, cleaning, and rule-based **anomaly detection** (temperature spikes, fuel drops, signal gaps).
- Designed a dual-storage model (raw + cleaned) in **PostgreSQL** and built a live **Streamlit** monitoring dashboard for time-series visibility.
- Added a **Spark** window-aggregation job for batch processing practice; containerized Kafka and Postgres with **Docker** Compose and automated setup via **Makefile**.

**E-commerce Analysis:** https://github.com/LukaPokrajac/olist-ecommerce-analysis

- Merged and cleaned multiple relational tables with **pandas** from the Olist Brazilian e-commerce dataset, resolving missing values and inconsistent timestamp formats across orders, deliveries, and products.
- Performed delivery-performance analysis by comparing estimated vs. actual delivery dates, identifying key bottlenecks in logistics.
- Created insightful visualizations with **matplotlib** of price distributions and delivery-delay patterns that revealed actionable business insights for pricing and fulfillment strategy.