



Ahmed Said Dursun

Date of birth: 08/04/2002 | **Place of birth:** Istanbul, Türkiye | **Nationality:** Turkish |

Phone number: (+39) 3515268264 (Mobile) | **Phone number:** (+90) 5071230804 (Home) |

Email address: ahmeddursunofficial@gmail.com | **Email address:** ahmeddursun11@gmail.com |

LinkedIn: [Ahmed Said Dursun](#)

ABOUT ME

Dedicated Computer Engineer with comprehensive experience in both cutting-edge Artificial Intelligence and robust Embedded Systems. Demonstrated success in delivering end-to-end projects, ranging from multi-threaded IoT industrial testing ecosystems to advanced NLP and RAG-based legal AI assistants. Combines strong analytical skills in deep learning and big data analytics with practical expertise in software architecture, hardware-software communication, and multi-platform deployment. Eager to contribute technical versatility and innovative problem-solving skills to a forward-thinking engineering team.

WORK EXPERIENCE

LCC TECHNOLOGY – ANKARA, TÜRKİYE

IOT SOFTWARE ENGINEER – 01/07/2024 – 01/09/2024

End-to-End System Development: Designed and developed comprehensive IoT and embedded software solutions, bridging physical hardware (ESP microcontrollers) with high-level desktop and backend applications (C#, Python).

Hardware-Software Communication: Established robust and reliable communication protocols (such as Serial Port) to ensure seamless, low-latency, real-time data transfer between sensors and monitoring systems.

Application Engineering: Built a multithreaded Windows-based testing application using C# and .NET Framework for real-time sensor data acquisition, processing, and hardware testing.

Data Visualization & UI: Developed interactive, user-friendly interfaces incorporating dynamic charting and custom GDI+ graphics to track pressure trends and live sensor metrics.

Automated Reporting & Security: Automated quality control processes by processing raw test data and exporting it into secure, structured, and encrypted PDF reports using iTextSharp and BouncyCastle.

R&D SOFTWARE ENGINEER – 16/02/2026 – 01/10/2026

Designed and developed a complete industrial glove integrity test system compliant with 21 CFR Part 11 regulations, consisting of multi-node embedded firmware, a desktop control application, and an HMI touch panel interface.

Embedded Firmware (ESP32-S3, C++/Arduino):

- Architected a multi-core, multi-threaded firmware using FreeRTOS with task pinning across dual cores (Core 0 & Core 1), mutex-based UART synchronization, and queue-driven inter-task communication
- Implemented ESP-NOW wireless Master-Slave architecture for real-time multi-device coordination
- Developed drivers for Modbus RTU pressure sensors, ADS1115 16-bit ADC, DS3231 RTC, and RDM6300 125kHz RFID readers
- Engineered shared UART multiplexing between RFID and Modbus peripherals with FreeRTOS semaphore protection
- Integrated ArduinoOTA for wireless firmware updates, LittleFS for persistent configuration, and SD_MMC for data logging
- Managed builds and deployments via PlatformIO with custom partition tables and PSRAM support

Desktop Application (Python/PyQt5):

- Built a full-featured PyQt5 GUI application with role-based access control, electronic signatures, and complete audit trail logging
- Implemented real-time serial communication with ESP32 devices using pyserial with multi-threaded RX/TX queues
- Generated 21 CFR Part 11 compliant PDF test reports using ReportLab and Matplotlib with cryptographic data integrity verification (cryptography, bcrypt)
- Utilized SQLite for local relational data storage with hash-based record integrity
- Packaged the application as a standalone Windows executable using PyInstaller and Inno Setup installer

HMI Interface (DWIN T5L, 8051 C51):

- Programmed DWIN industrial touch display with custom UI pages and serial command protocol

Key Technologies: ESP-IDF, FreeRTOS (multi-core task scheduling, semaphores, queues), ESP-NOW, Modbus RTU, ArduinoJson, LittleFS, SD_MMC, ADS1115, DS3231, RDM6300, PyQt5, pyserial, ReportLab, Matplotlib, bcrypt, cryptography, SQLite, PyInstaller, PlatformIO, Inno Setup

INTERN – HAVELSAN – 04/08/2025 – 05/09/2025 – ANKARA, TÜRKİYE

Vulnerability Assessment & Penetration Testing: Executed comprehensive vulnerability assessments and penetration testing across network infrastructures, identifying potential security flaws and recommending actionable remediation strategies.

Malware Analysis & Threat Intelligence: Performed static and dynamic malware analysis within isolated, controlled sandbox environments to dissect malicious payloads, extract Indicators of Compromise (IoCs), and document threat behaviors for internal reporting.

Security Tool Proficiency: Leveraged industry-standard offensive and defensive security toolsets, utilizing Kali Linux for exploit execution, Splunk (SIEM) for log analysis and anomaly detection, and Metasploitable environments for testing vulnerability frameworks.

Incident Response Operations: Actively collaborated with cross-functional team members during simulated incident response tabletop exercises, modeling real-world cyberattack scenarios to refine threat mitigation protocols and disaster recovery procedures.

COMPUTER VISION INTERN – MIMBYTES – 08/09/2025 – 06/10/2025 – ANKARA, TÜRKİYE

Computer Vision & Deep Learning: Engineered an Automatic Number Plate Recognition (ANPR) system utilizing Python, OpenCV, TensorFlow, and PyTorch for robust license plate detection and CNN-based OCR.

Model Optimization & Analysis: Conducted rigorous performance, speed, and accuracy evaluations (TP, FP, FN metrics) across varying lighting conditions (daytime/nighttime) to fine-tune the detection and recognition pipelines.

Edge Computing Deployment: Deployed and tested optimized machine learning models on edge devices, including Raspberry Pi and NVIDIA Jetson Nano, enabling real-time, on-site video processing.

System Integration & Security: Integrated extracted license plate data into SQL databases to establish automated access control, entry-exit tracking, and enhanced security monitoring ecosystems.

EDUCATION AND TRAINING

10/09/2022 – 09/06/2026 Konya, Türkiye

BACHELOR'S DEGREE IN COMPUTER ENGINEERING Selçuk University

- **Programming Languages:** Python, C, C++, C#, Java.
- **Software Engineering:** Object-Oriented Programming, Data Structures, Algorithms, Software Design, and Development.
- **Artificial Intelligence and Machine Learning:** Deep Learning, Neural Networks, Computer Vision, and Natural Language Processing.
- **Database Systems:** SQL, Relational Databases, Database Design, and Management.
- **Computer Systems:** Operating Systems, Computer Architecture, Embedded Systems, and Computer Networks.
- **Web Development:** Front-End and Back-End Development, REST APIs, and Web Technologies.
- **Mathematics and Theoretical Foundations:** Linear Algebra, Calculus, Differential Equations, Discrete Mathematics, Probability, and Statistics.
- **Additional Skills:** Git, Software Testing, Debugging, Problem-Solving, and Technical Documentation.

Website <https://www.selcuk.edu.tr/> | **Level in EQF** EQF level 6

20/09/2024 – 30/07/2025 Pavia, Italy

BACHELOR'S DEGREE ERASMUS IN COMPUTER ENGINEERING Pavia University

Main subjects covered: Deep Learning, Data Science and Big Data Analytics, Cloud Computing, Natural Language Processing, Information Retrieval and Recommender Systems.

Occupational skills acquired: Machine learning and deep learning model development, data analysis, big data processing, cloud-based computing concepts, NLP and text mining techniques, recommender systems, Python-based programming, teamwork, academic presentation skills, and working in an international English-speaking academic environment.

Website <https://en.unipv.it/en> | **Level in EQF** EQF level 6

- Artificial Intelligence and Machine Learning
- Deep Learning and Neural Networks
- Data Science and Big Data Analytics
- Algorithms and Advanced Computer Science
- Software Engineering and Programming
- Data Mining and Information Retrieval
- Natural Language Processing
- Cloud Computing and Distributed Systems
- Statistical and Mathematical Methods for AI

Level in EQF EQF level 7

● **LANGUAGE SKILLS**

Mother tongue(s): **TURKISH**

Other language(s):

	UNDERSTANDING		SPEAKING		WRITING
	Listening	Reading	Spoken production	Spoken interaction	
ENGLISH	C1	C1	C1	C1	B2

Levels: A1 and A2: Basic user; B1 and B2: Independent user; C1 and C2: Proficient user

● **SKILLS**

work as a team | SQL | C / C# / C++ | Object Oriented Programming | Microcontroller programming such as Arduino ESP8266 & ESP32 | Quick-learning | Analytical Thinking. | computer programming | Python (computer programming) | Communication Protocols | Hardware Integration | TensorFlow, Keras, Scikit-learn, PyTorch, .NET, ASP.NET MVC 5, Shiny, Selenium WebDriver, Laravel | computer vision | natural language processing | Retrieval-Augmented Generation | PySpark Python | use concurrent programming | Parallel programming | C | C++(11/14) | STL | UML | OOPS | QT | SOLID Principles | C++11 Multithreading | Git | Agile & V

● **HOBBIES AND INTERESTS**

AI Systems & Data Engineering

I am passionate about exploring the intersection of Big Data and Artificial Intelligence. My personal projects focus on building intelligent data pipelines and designing RAG (Retrieval-Augmented Generation) architectures. I constantly experiment with modern NLP techniques, indexing mechanisms, and machine learning models to solve complex real-world information retrieval problems.

● **PROJECTS**

30/10/2025 – 30/11/2025

Corporate Tax Law RAG Assistant | FastAPI, Llama 3.1 8B GGUF, llama.cpp, ChromaDB, Sentence Transformers

Parsed Corporate Tax Law No. 5520 and its General Communiqué into hierarchical JSON records while preserving article numbers, section paths, headings and document types as retrieval metadata. Linked communiqué sections to the corresponding law nodes through article-number constraints, Turkish-aware text normalisation, title-level Jaccard similarity and parent-to-child fallback for nested sections.

- Deduplicated records by stable document ID, retained the longest version of repeated content and generated embeddings in batches of 64 with paraphrase-multilingual-mpnet-base-v2. Persisted vectors, source text and Chroma-compatible metadata in an HNSW collection configured for cosine distance, allowing the index to be rebuilt deterministically from processed legal data.
- Implemented hybrid retrieval by combining the top four dense-vector results with up to two non-duplicate lexical matches. The lexical path uses Turkish tokenisation, stop-word filtering and query-term overlap; the final six-document context is reconstructed from the canonical JSON store and labelled with law article or communiqué section identifiers for traceable answers.
- Ran Llama 3.1 8B Instruct locally from a Q4_0 GGUF model through llama.cpp, using a 16K context window and deterministic generation settings. Designed source-constrained prompts that reject unsupported questions and

require article-level references, and exposed the local inference layer through typed FastAPI /chat and /health endpoints.

10/03/2025 – 20/03/2025

Air Quality Forecasting with Deep Learning

Developed and trained multiple deep learning architectures (Feedforward, LSTM, Transformer) to forecast air pollutant levels using time series data.

Engineered end-to-end data preprocessing pipelines with Scikit-learn and implemented automatic differentiation and optimization algorithms using Python and PyTorch.

Evaluated and compared model performance across architectures using RMSE metrics to analyze the efficacy of attention mechanisms and network depth.

Visualized prediction versus ground truth results, documenting data-driven insights applicable to real-world environmental monitoring systems.

Technologies Used: Python, PyTorch, Scikit-learn, LSTM, Transformers, Time Series Analysis, Data Visualization.

01/04/2025 – 30/05/2025

Reddit Interaction & Content Analysis Pipeline | PRAW, Kafka, PySpark Structured Streaming, GraphFrames, Parquet, Docker Compose

- Implemented both historical backfill and live ingestion for r/Turkey with PRAW. The collector supports a configurable 365-day window and 10,000-post limit, runs submission and comment streams in parallel threads, applies retry and rate-limit handling, and publishes keyed JSON events to separate reddit_posts, reddit_comments and reddit_interactions Kafka topics with acknowledgements enabled.

- Derived graph-ready reply edges while ingesting comments by mapping each comment author to the parent-comment or submission author and recording source, destination, weight, timestamp and post ID. Configured Spark 3.4 with the GraphFrames package so these edges can feed influence and community-analysis stages such as PageRank and label propagation.

- Built independent PySpark Structured Streaming consumers with explicit schemas for posts, comments and interaction edges. Kafka payloads are deserialised with from_json and written in append mode to partition-friendly Parquet datasets, with a separate checkpoint directory for every query to support restartable, fault-tolerant processing.

- Added a Turkish content-analysis branch that normalises post and comment text, applies case-insensitive nationalist-keyword matching, extracts the detected terms and routes matching records to dedicated analytical Parquet outputs. Containerised Kafka, ZooKeeper and MongoDB services with Docker Compose and added stream/progress monitoring for operational visibility.

01/10/2024 – 20/11/2024

Hybrid Neural Search Engine

Information Retrieval Architecture: Engineered a sophisticated search engine utilizing PyTerrier and Sentence Transformers to enhance document retrieval accuracy and relevance.

Model Evaluation & Comparison: Conducted comprehensive comparative analyses between traditional lexical models (BM25, TF-IDF) and modern dense retrieval techniques using Bi-Encoder embeddings.

Ranking Optimization: Implemented and fine-tuned a hybrid ranking pipeline that fused semantic and keyword-based scores, successfully improving the system's overall retrieval performance as measured by the NDCG@10 metric.

16/02/2026 – 30/09/2026

Sterility Test Isolator HMI | .NET 8, C#, WPF, SQLite, QuestPDF

- Developed a desktop HMI for a sterility test isolator, covering the complete operating story from system overview and manual control to automatic sequences, leak testing, decontamination, aeration, production, alarms, and diagnostics. The screens were organised around operator decisions, equipment states, and clear recovery paths rather than isolated controls.

- Designed the SQLite data layer for users, roles, system settings, recipes, batches, glove tests, individual results, and electronic signatures. Audit records are linked with previous-record and current-record SHA-256 hashes, allowing the application to verify the integrity of the event chain and identify the point at which a record no longer matches.

- Created QuestPDF reports for audit trails, batches, and glove tests, including metadata, test results, responsible users, and integrity status. The reporting layer converts operational data into reviewable records for validation, investigation, and handover activities.

16/02/2026 – 30/09/2026

Cleanroom Monitoring & DWIN HMI Communication System | ESP32-S3, FreeRTOS, Modbus RTU, Dual RS485, DWIN HMI, ESP-NOW

- Developed the embedded communication layer for an Iraq field installation in which three cleanroom monitoring devices, a PLC/SCADA endpoint, and a DWIN HMI exchange temperature, humidity, pressure, air-speed, door, lighting, and alarm data through a shared Modbus register map.
- Configured two ESP32-S3 hardware UARTs as RS485 half-duplex channels and separated device polling from SCADA/HMI writes with FreeRTOS tasks pinned to processor cores. Mutex-protected shared data and command queues prevent concurrent bus access and preserve deterministic device commands while live measurements continue to update.
- Implemented batched register reads, timeout/error handling, status derivation, and DWIN compatibility fallbacks from Function Code 16 to sequential Function Code 6 writes. Also prepared build, firmware upload, serial logging, and ESP-NOW pairing/bridge utilities to support commissioning and field troubleshooting.

16/02/2026 – 30/09/2026

Brake Test Data Analysis & Reporting Application | Python, PySide6, pandas, NumPy, PyQtGraph, Matplotlib, openpyxl

- Created a desktop analysis workflow that imports Excel/CSV brake-test files, cleans and scales channels, trims test periods, and calculates derived engineering signals such as elapsed time, travelled distance, angular/linear velocity, and acceleration.
- Built interactive dashboards for dynamic plotting, cross-test comparison, fade analysis, correlation review, and period-level trends. Users can move from raw measurements to an engineering interpretation in one interface and export evidence as Excel, CSV, PDF, or PNG outputs.
- Prepared Windows and macOS packaging flows, separating analysis logic from presentation and export layers so that new test channels and reporting requirements can be added without rewriting the full application.

16/02/2026 – 30/09/2026

Internal Task & Document Management Platform | Node.js, Express.js, SQLite, JavaScript, HTML, CSS

- Developed an internal web platform for task planning, time tracking, bulk operations, department-based permissions, messaging, notifications, announcements, dashboards, and automation rules. Helmet, request rate limiting, authentication, and role checks protect the application layer and restrict business actions.
- Implemented document upload, folder organisation, version history, edit locks, and restoration, supported by session controls, audit logs, and backups to keep work items and their documents traceable.

16/02/2026 – 30/09/2026

INTEGRATION, QUALIFICATION & OPERATIONS

- Uyumsoft API integration — Connected internal applications with Uyumsoft services, translated application data into the required request/response structures, tested end-to-end communication flows, and investigated failed transactions through validation, error logging, and retry-oriented troubleshooting.
- IQ/OQ documentation — Prepared Installation Qualification and Operational Qualification documents that link system requirements to installation checks, prerequisites, step-by-step functional tests, expected results, acceptance criteria, evidence fields, deviations, and approval records.
- Technical procurement — Converted project needs into comparable technical requirements; researched products and suppliers, evaluated quotations and compatibility, completed purchasing activities, and followed orders through delivery for project execution.

16/02/2026 – 30/09/2026

ORABS Control Platform & Glove Leak Test Integration | Python, PyQt5, three.js, GLB/GLTF, ESP32-S3, ESP-NOW, Modbus, SQLite, Weidmüller I/O

Developed ORABS as a Windows control platform that brings remote I/O modules, field sensors, facility states, alarms, interlocks, batches, and operator workflows into one application. Integrated the independently developed Glove Leak Test System into ORABS as a reusable subsystem, preserving its device protocol and test logic while surfacing live pressure, recipes, RFID-to-port assignments, progress, results, and reports inside the common workflow.

- Structured the application around dedicated communication and worker layers for ESP32 devices and Weidmüller Modbus I/O. Device states, connection health, command/response handling, LAF and barrier signals, door states, and test progress are translated into UI-safe events, keeping long-running hardware operations separate from the PyQt interface and making faults easier to diagnose.
- Implemented the plant visualisation as an offline three.js module embedded inside PyQt5 QWebEngineView. A local GLTFLoader loads the facility model from a GLB/GLTF file, while a QWebChannel bridge carries model-ready events, door/port selections, state changes, label positions, and live LAF values between Python and JavaScript without requiring a browser or internet connection.
- Designed four independent WebGL views—front, rear, left, and right—with separate PerspectiveCamera and WebGLRenderer instances. GLB node names are parsed to map meshes to logical doors and glove ports; cloned materials visualise door status and pass/fail test states, and THREE.Raycaster enables direct model selection, camera navigation, and drag-based label placement.

- Optimised the 3D layer for continuous industrial use by throttling requestAnimationFrame rendering to 15 FPS, disposing geometries, materials, and textures when models are replaced, and presenting rapidly changing LAF values in DOM overlays rather than rebuilding the GLB scene. Pending command queues preserve state updates until the JavaScript viewer and model are fully ready.
- Added role-based access, password policies, audit trails, electronic signatures, SHA-256 integrity checks, PDF reports, database backup/migration, simulation utilities, and installer packaging. These controls support a 21 CFR Part 11-oriented design and provide traceability for users, batches, recipes, tests, and configuration changes.

16/02/2026 – 30/09/2026

Glove Leak Test System | ESP32-S3, C/C++, Python, PyQt5, ESP-NOW, DWIN HMI, RFID, Modbus RTU, SQLite

- Developed the complete standalone glove integrity testing solution across embedded firmware, an ESP-NOW master/slave device network, a DWIN operator display, and a Windows PyQt5 application. Integrated RFID-based glove identification, pressure acquisition, RTC timekeeping, battery monitoring, gasket control, pumps and valves into a single test device.
- Implemented the embedded test state machine from RFID and recipe selection through gasket mounting, inflation, stabilisation, optional reinflation, rolling-window pressure-drop evaluation, and PASS/FAIL completion. Critical pressure loss triggers an immediate failure; one-second pressure samples are retained for the test record, and a dismount interlock prevents the next cycle from starting before the previous glove is removed.
- Built the PC control layer for automatic ESP32-S3 port discovery, thread-safe serial communication, RFID/recipe synchronisation, live pressure and state monitoring, device registration, batch templates, port-level PASS/FAIL tracking, SQLite storage, PDF test and batch reports, role-based access, audit trails, and electronic signatures.
- Added SD card and NVS persistence, RTC and battery diagnostics, OTA support, structured firmware services, category-based logging, serial commissioning commands, recovery handling, database migration checks, end-to-end communication tests, and Windows installer packaging to support deployment and field troubleshooting.