ZIYAD MOHAMED ALBOGHDADY

Undergraduate Researcher · Machine-Learning Security · Backend Development (C# / .NET)
Sakarya, Türkiye | +90 554 027 5485 | ziyadalboghdady@gmail.com
ziyad-alboghdady.netlify.app | github.com/ziyad-alboghdady | linkedin.com/in/ziyad-alboghdady-a50ab8247

SUMMARY

Third-year Software Engineering undergraduate researching why machine-learning intrusion detectors fail outside the lab. At Sakarya University's CyberSense Laboratory, first author of a paper in preparation showing that a packet-2 DDoS detector with ROC-AUC 0.99999 had learned victim-server fingerprints rather than attack behaviour. Also builds backend systems in C# / ASP.NET Core and leads an eight-person software team. Seeking an Erasmus+ research traineeship in machine-learning security.

EDUCATION

B.Sc. Software Engineering — Sakarya University, Türkiye *Expected 2028*
Full merit scholarship · GPA 3.63 / 4.00 · 4.00 / 4.00 in the most recent semester · GPA rising every semester

RESEARCH & EXPERIENCE

Undergraduate Researcher (volunteer) — CyberSense Laboratory, Sakarya University *Aug 2026 – Present*

- Work on an ML DDoS detector running as a native Snort 3 inspector on a 1 Gb/s lab bench, classifying flows at the second packet (XGBoost, served via ONNX Runtime), over a 1.19M-flow dataset the team labelled from raw pcap.
- Showed that the model's ROC-AUC of 0.99999 rested on victim fingerprints: one SYN-ACK window value appears in 99.74% of attack flows, and with victim-side features removed the false-positive rate on 20 held-out hosts rose from 0.037% to 47–52%.
- Found shortcut substitution: removing one fingerprint moves the model to the next victim-only feature, with per-host false-positive rate tracking single-value share at r = 1.000.

Research Paper (First Author) — What a First-Packet DDoS Detector Really Learns *In preparation*

- Combines packet-2 detection, host-disjoint evaluation and an in-sensor deployment on a lab bench, a combination we did not find in prior work, to measure train/serve missing-value skew and shortcut substitution; proposes an evaluation checklist for early-flow ML intrusion detection.

TÜBİTAK 2209-A Research Project — Prompt-Injection Detection in Turkish Web Text *Proposal · 2026*

- Separate team project comparing rule-based, TF-IDF and transformer defences, and measuring how Turkish morphology, spelling variation and Turkish–English code-switching affect detection, with attack-family-disjoint splits, hard negatives and McNemar tests.

Project Manager & Backend Developer — Campusly (team project, Sakarya University) *Sep 2026 – Present*

- Lead an eight-person team (five backend, three frontend) and develop the backend of a campus platform (map, maintenance reporting with duplicate detection, events, transport) in ASP.NET Core 8 Web API, PostgreSQL, EF Core and JWT authentication.
- Own the frontend–backend API contract (OpenAPI), scope and schedule; set a mid-project checkpoint that drops a module if the core is not working by a fixed date.

Web Developer (pro bono) — Elmas Grup Gayrimenkul · bilingual real-estate website in Next.js, with one teammate *2026*

Translator (Arabic–Turkish–English) — Turkish Ministry of National Defence, IDEF · Iraqi Minister of Defence delegation *2025*

PROJECTS

Task Management API — .NET 9 | ASP.NET Core | PostgreSQL | Docker | Supabase Auth *2026*

- Containerised REST API (controllers → services → repositories) with JWT authentication in custom middleware; every query is scoped to the signed-in user, and another user's task returns 404 rather than 403 so the API never reveals it exists.
- Background job pipeline: long requests return 202 with a status endpoint, failed jobs retry with exponential backoff, and a scheduled heartbeat flags stuck jobs; checked by a 16-step automated smoke-test script.

Car Rental Management System — ASP.NET Core MVC | EF Core | PostgreSQL

- Seven-module platform on a normalised 12-table PostgreSQL schema, with constraints, stored functions and triggers enforcing business rules in the database; ASP.NET Identity role-based access, plus ERD and requirements documentation.

Cached Personnel System with Local LLM — Spring Boot | Redis | MySQL | Hadoop HDFS | Ollama *Oct 2026*

- Reads and updates personnel data through a Redis cache that periodically flushes to MySQL for durable storage, stores employee images in Hadoop HDFS, and adds a chat page backed by a local Llama 3.2 model.

TECHNICAL SKILLS

ML & Security: XGBoost, ONNX Runtime, feature engineering, data-leakage diagnosis, host-disjoint evaluation, Snort 3, pcap analysis
Languages: C#, Java, SQL, JavaScript, C++
Backend: ASP.NET Core (Web API, MVC), EF Core, ASP.NET Identity, Spring Boot, REST APIs, JWT auth, background jobs, OpenAPI
Databases: PostgreSQL, SQL Server, MySQL, MongoDB, Redis, Hadoop HDFS · schema design, normalisation, triggers & stored functions
Practices & Tools: OOP, SOLID, layered architecture, Repository pattern, Git, Docker Compose, Postman, Linux, Agile

AWARDS, CERTIFICATIONS & LANGUAGES

Awards: Rise Global Finalist, top 500 of 131,000+ (2022) · IoT & AI Competition, 2nd in Egypt, 5th in the Middle East · 24H Innovation Challenge, 2nd place (2026) · Düzce University Hackathon, Best Design (2026)
Certifications: FlyRank AI Internship, remote programme: AI Fluency certificate (2026); Backend AI Engineering capstone under review · Rise Entrepreneurship Bootcamp, University of Chicago
Languages: Arabic (native) · English C1 (IELTS 7.0) · Turkish C1 · Russian A2 · French A1