Zaid Barakat

zaidbarak@tamu.edu - (409) 310-1288 – College Station, TX
LinkedIn: https://www.linkedin.com/in/zaidfbarakat GitHub: https://github.com/zaid-brk

EDUCATION

Texas A&M University | College Station, TX **Graduation: May 2028**
Bachelor of Science, Computer Science GPA: 3.75/4.00
Dean's Honor Roll – 3/4 Semesters
Relevant Courses: Data Structures & Algorithms, Linear Algebra, Computer Systems, Software Engineering

WORK EXPERIENCE

CANAL Engineering Services | Software Engineering Intern **May 2026 - August 2026**
Attendance & Payroll Management System | Next.js, Supabase, TypeScript Dubai, UAE

- Built a production full-stack payroll platform managing **400+ employees** and automating **AED 1.8M/month** in salary computation, replacing error-prone manual spreadsheets.
- Engineered a QR-based clock-in system (rotating 65s tokens, PWA in-app scanner, manual-code fallback), eliminating **25 admin hours/month** of manual attendance logging across multiple sites.
- Replaced a 3-day manual payroll process with a Decimal.js computation engine (22 unit tests), reducing processing time by **over 90%**; supports per-diem proration, unexcused-absence deductions, holds, and admin adjustments.
- Implemented 5-tier RBAC (super admin, manager, coordinator, team leader, employee) with Supabase RLS, full audit logging, XLSX/CSV reporting, and fuzzy-match bulk employee import.

PROJECTS

GitHub Repo Q&A Agent | *Python, LangChain, Qdrant, FastAPI, Next.js, Docker*

GitHub Link: https://github.com/zaid-brk/repo-qa-agent

- Built an end-to-end RAG pipeline over GitHub repositories, file ingestion, code chunking via LangChain's RecursiveCharacterTextSplitter, vector embedding with OpenAI's text-embedding-3-small, and indexing in Qdrant, enabling natural language Q&A over codebases of **10,000+ lines**.
- Designed a tool-calling layer exposing list_files(), read_file(), and search_code() for multi-hop retrieval across files, and built a **50-question ground-truth evaluation harness** graded by an LLM-as-judge; the agent scored **100% (50/50)** across three open-source repositories (requests, click, flask).
- Added adversarial should-not-answer questions to catch hallucination. The agent correctly refuses unanswerable queries (e.g. a nonexistent production database password) instead of fabricating information. Deployed with **Render (Dockerized FastAPI) and Qdrant Cloud**, exposing a live public REST API.

Prompt Injection Red-Teaming Framework | Python, LangChain, FastAPI

GitHub Link: https://github.com/zaid-brk/injection-redteam

- Built a tool-augmented LLM agent pipeline and automated test suite of **42 prompt injection attacks**, direct override, indirect injection via web content, goal hijacking, role-switching, encoding/obfuscation, and payload splitting, benchmarked across 6 attack categories.
- Implemented a multi-layer defense system (input sanitization, system-prompt hardening, output monitoring) reducing overall attack success rate from **19% to 5%**, and goal-hijacking specifically from **100% to 14%**, across the 42-attack suite.
- Adapted attack patterns from published work (InjectAgent, OWASP LLM01) after the Real-World AI Security conference (Stanford Security Lab, 2026) showed that **input sanitization is structurally blind to indirect injection**, the payload arrives inside fetched web content, after the input filter runs, so prompt-hardening is what drove the reduction.

TECHNICAL SKILLS

Languages: Python, C++, JavaScript, TypeScript, Java, SQL, HTML, CSS
Frameworks and Libraries: LangChain, FastAPI, React, Next.js, Node.js
Tools and Platforms: Git, Docker, PostgreSQL, Supabase, Qdrant, AWS, REST APIs
AI and ML: Retrieval-Augmented Generation, Vector Databases, LLM Evaluation, Prompt Engineering