Arshiya Gupta

Software Engineer | (447)9021772 | **Email** | **Linkedin**

Education

University of Illinois at Urbana-Champaign *Champaign, IL*
B.S. in Computer Science, Minor in Statistics and Mathematics | GPA: 3.9/4.00 May 2026

Skills & Interests

Languages: C/C++, Python, Java, JavaScript, TypeScript, Rust, HTML/CSS
Frameworks/Tools: React, Next.js, Node.js, Spring Boot, Flask, NumPy, Pandas, Prometheus, Docker, Kubernetes
Relevant Coursework: Database Systems, Cybersecurity, Systems Programming, Data Structures, Algorithms

Experience

Axon
Software Engineering Intern May 2025 - August 2025
- Improved system modularity and independent service scaling by 100% by replacing database writes with HTTP API calls.
- Reduced test setup time by 40% by building a configurable Gatling test tool using parameterized load profiles.
- Prevented 5+ production outages by enabling real-time latency and throughput monitoring using Grafana dashboards.
- Increased test coverage by 50% for ingestion workflow handling 70PB/month by deploying a Scala-based load testing platform on Azure.

PayPal Inc.
Lead Software Engineering Intern May 2024 - August 2024
- Enhanced system reliability reducing production incidents across 3+ services by implementing error handling modules.
- Secured 5+ CI/CD pipelines cutting token exposure risk by 95% by enforcing TLS and masking credentials in logs.
- Reduced auth-errors by 33% by implementing token-based systems with expiration and role-based access control policies.
- Boosted Order API performance by 35% supporting 3M+ daily transactions via code refactoring and query optimization.

HackIllinois
API Team Sept 2024 - Present
- Scaled API capacity to support 5k+ hourly requests by designing RESTful backend architecture with 30+ endpoints.
- Ensured secure API access for 5k+ users by implementing multi-factor JWT authentication with key rotation and encryption, reducing unauthorized access attempts by 90%.
- Increased event check-in reliability by 30% by developing a microservice API with WebSocket and end-to-end testing.

CS Student Ambassadors / Research Scholars (CS STARS) January 2023 - Present
- Reduced data collection costs by 80% by engineering zero- and few-shot learning, eliminating historical data requirements.
- Increased predictive accuracy by 30% by developing a Rust hyperparameter optimizer using Bayes optimization techniques.

University of Illinois at Urbana-Champaign
Undergraduate Teaching Assistant January 2024 - Present
- Designed MIPS-based SpimBot challenge, demonstrating CPU architecture concepts for a cohort of 500+ students.
- Reduced grading time by 75% across 10k+ daily submissions by scripting automated workflows in Python and Bash.
- Implemented buffer overflow protection and secure memory addressing within the low-level system architecture.

Projects

SkyTrack: Airplane Tracking System January 2025 - April 2025
- Reduced flight lookup latency by 70% by designing normalized schema with multi-attribute indexing on flight schedules and aircraft metadata.
- Maintained data integrity across 6+ tables by implementing foreign key constraints, custom triggers, and cscading updates.
- Reduced notification latency to 1.5 seconds by optimizing trigger-based alerts for flight status changes.

Notionize: Weekly automation tool December 2024
- Reduced weekly planning time by 70% by building a Discord-based task sync bot with smart multi-source parsing.
- Synced 1k+ daily events between Google Calendar and Notion with 99% uptime using Flask and Notion API.
- Optimized API efficiency reducing calls by 80% and slashing response time from 2s to 300ms with SQLite caching.

Password Manager September 2024 - November 2024
- Achieved 100% data confidentiality by implementing zero-knowledge encryption eliminating plaintext exposure.
- Increased vault resilience by 80% by designing secure sharding across multiple encrypted files, protecting against corruption and targeted file attacks.
- Reduced brute-force success by 75% by implementing progressive time-locked delays after failed attempts, preventing automated attacks while maintaining user access.