Nathan Yee

281-865-6605 | nathanyee.cv | nathanyee99@gmail.com | linkedin.com/in/nathaniyee | github.com/nathaniyee

Education

The University of Texas at Austin — Austin, TX
Bachelor of Science in Computer Science, Minor in Business, Minor in Statistics & Data Science — *May 2027*
GPA: 3.78
Relevant Coursework: Algorithms, Operating Systems, Computer Architecture, Data Structures, Software Engineering, Machine Learning, Data Management, Linear Algebra, Differential Equations, Statistics, Discrete Math, Calculus I-III

Technical Skills

Languages: Python, Java, C++, JavaScript, TypeScript, C, SQL, Bash, Go, Rust, Swift, Kotlin, R, HTML/CSS
Technologies: Git, GitHub, Linux/Unix, React, Node.js, Next.js, Flask, Nginx, REST APIs, Docker, Kubernetes, React Native, AWS, PostgreSQL, MySQL, MongoDB, Firebase, Grafana, Postman, Jira, Figma, PyTorch, TensorFlow, Jupyter

Experience

Software Engineer Intern — May 2026 – Present
Apple — *Cupertino, CA*
- Build full-stack internal tools with React and Spring Boot for Apple Newsroom, supporting platform development

Software Engineer Intern — May 2025 – August 2025
ForeFlight — *Austin, TX*
- Developed full-stack React components for ForeFlight's web platform, supporting thousands of active pilots daily
- Implemented web version of ForeFlight's mobile Checklist tool, enabling 3x faster creation and 60% usage growth
- Created responsive UI in React/TypeScript with editable components, dynamic rendering, and full mobile parity
- Refactored legacy code for data binding, input validation, and nested hierarchy; reduced server error rate by 45%

Software Engineer — September 2024 – Present
Longhorn Racing — *Austin, TX*
- Develop an internal lap timing and telemetry logging tool to synchronize data entry and streamline vehicle testing
- Automate data collection and test documentation using React and Firebase, reducing manual logging time by 80%
- Leverage MoTeC i2 to analyze vehicle telemetry, identifying performance trends and informing setup adjustments
- Coordinate setup of vehicle dynamics/electronics tests via custom instrumentation, cutting execution time in half

Software Engineer Intern — August 2024 – December 2024
Phenom — *Austin, TX*
- Overhauled Phenom web architecture with React to implement new business logic, boosting user sign-ups by 37%
- Improved mobile navigation scores by 56% via Lighthouse by migrating static components to responsive layouts
- Reduced server error rates by 95% by refactoring existing Sharetribe backend logic for athlete asset management

Projects

Drive Day Log | *React, TypeScript, Firebase, Firestore* — January 2026 – May 2026
- Built a vehicle testing documentation tool using React/Firebase to capture session metadata and performance data
- Designed role-based user interfaces to streamline lap timing, penalty tracking and session management on mobile
- Implemented cloud storage and session architecture with Firestore, supporting concurrency and multi-device access

Sentinel | *React, TypeScript, Next.js, FastAPI, Supabase, Gemini API, Twilio* — April 2026
- Built web app to assist 911 operators by aggregating and prioritizing distress calls; placed 2nd out of 100+ teams
- Developed FastAPI pipelines for call ingestion and incident state management (Gemini), improving triage by 90%
- Implemented trend detection using cosine similarity clustering of call embeddings to escalate lower-priority calls

NBA Matchup Predictor | *Python, Playwright, scikit-learn* — August 2024
- Engineered custom Playwright scrapers to parse and extract advanced stats from 5000+ NBA box scores at scale
- Constructed 50+ predictive features (eFG%, TS%, weighted ELO) from raw data via rolling-window aggregations
- Built and trained a Ridge Regression model in scikit-learn, achieving ∼70% predictive accuracy on game outcomes

Pintos | *C, x86 Assembly* — August 2024 – December 2024
- Built multithreading with locks, semaphores, and an MLFQ scheduler, improving prioritization & CPU utilization
- Implemented virtual memory with page tables and demand paging, improving memory efficiency by 30% at load
- Extended file system with syscall handling and full directory support, reducing benchmarked I/O latency by 50%