Arul Saxena

Portfolio: arulsaxena.com
Github: github.com/ajs808

Email: asaxena101g@gmail.com
Mobile: (408) 707-0339

Education

- **University of California, Santa Barbara** — Santa Barbara, CA
  B.S. Computer Science — *2021 - 2025*
  Regents Scholar in the College of Engineering Honors Program; cumulative GPA 3.90

Skills

- **Languages** : Python, Javascript, Java, C, C++, C#, HTML/CSS, Swift, R, SQL, OCaml, Typescript, Go
- **Frameworks**: React, Node.js, Flutter, Scikit, NLTK, Django, Flask, Bootstrap, .NET
- **Platforms**: Unix, Web, AWS, iOS, Android, Arduino, Raspberry Pi

Experience

- **Roblox** — San Mateo, CA
  Software Engineering Intern (Infrastructure: Observability and Telemetry) — *Jun - Sep 2023*
  - Developed a tool with UI to detect and subsequently disable low-value metric collectors using .NET framework. Decreased active collector count by 20%; saw 40% decrease in %CPU usage
  - Developed an administration page to view and manage canary analysis jobs for service deployments (Go, Typescript, React)
- **data.ai** — Remote
  Software Intern — *Jul - Aug 2022*
  - Learned data warehousing practices and ETL (extract, transform, load) techniques; queried databases using SQL on Snowflake
  - Using above techniques, generated mobile application performance analytics based on metrics like reviews and advertisement success rate

Projects

- **SmartLock**
  Smart lock prototype
  - Developed a Raspberry Pi device that mounts over a deadbolt to give it smart functionality
  - Enabled users to lock/unlock their door remotely through a Flask web app (secured with Google OAuth 2.0), automatically lock the door after being unlocked for a set amount of time, and unlock the door with geofencing and NFC technology

Relevant Coursework

- **Machine Learning** — **UCSB**
  Classifiers, generative models, & neural networks; applications with NumPy & scikit-learn
- **Advanced Algorithms Engineering**
  Techniques and algorithms for contest-level programming
- **Data Structures & Algorithms I & II**
  Analysis of structures and algorithms with proofs of correctness; complexity and reductions
- **Software Engineering**
  SDLC, CI/CD, version control, agile methodology, code review & quality assurance
- **Computer Graphics**
  Full 3D graphics pipeline with OpenGL: rasterization, illumination, ray-tracing, physics simulation
- **Internet of Things**
  End-to-End IoT systems; device programming, communication protocols, security, edge & cloud computing
- **Cryptography**
  Provable security: encryption schemes, attacks, pseudorandomness, digital signatures, & public key systems
- **Foundations of Computer Science**
  Formal logic & discrete math in programming contexts, algorithmic analysis
- **Computer Organization & Logic Design**
  Assembly language programming and circuit design
- **Computer Architecture**
  Circuit simulation of CPU-level processes (caching, instruction decoding, optimization & hazard handling) with PyRTL
- **Automata & Formal Languages**
  Finite & pushdown automata, Turing machines & computability
- **Translation of Programming Languages**
  Developed a full compiler for Patina—a Rust variant—using OCaml
- **Computational Science**
  Numerical algorithms using NumPy and SciPy
- **Programming Abstractions** — **Stanford University**
  Developed an end-to-end Huffman compression tool & maze-solving algorithm visualizers
- **Programming in Swift** — **De Anza College**
  Developed iOS and macOS applications using the UIKit framework