

Nishchay Parashar

RESEARCH ENGINEER · COMPUTER SCIENTIST · SOFTWARE DEVELOPER · SYSTEMS ENGINEER

☎ +1 (401) 471-8296 | ✉ nishchay_parashar@brown.edu | 🌐 nishchayp.github.io | 🐙 nishchayp | 🌐 nishchayp

Skills

Programming C/C++, Python, Go, Flask, Django, Node.js, Ruby, Shell Scripting, Power Shell

Databases Elastic, MongoDB, SQL, Redis, PostgreSQL

Tools & Technologies Git, REST, API, Linux, DevOps, AWS, Docker, Ansible, RabbitMQ, Jenkins, HTML

Education

Brown University

Providence, RI, USA

MASTER'S IN COMPUTER SCIENCE - GPA: 4.00/4.00

Sep. 2022 - May 2024 (expected)

- Teaching Assistant for 'CSCI 0300: Fundamentals of Computer Systems'.
- Courses: Algorithms, Operating Systems, Computer Systems, Cryptography.

Manipal Institute of Technology

Manipal, India

BACHELOR'S IN COMPUTER SCIENCE AND ENGINEERING - GPA: 3.96/4.00

Aug. 2016 - Aug. 2020

- One in only 250 global GitHub Campus Expert, Gen. Sec. CS Society (leading 1500+ students).

Work Experience

Samsung Research - Samsung Semiconductor India Research

Bangalore, India

SENIOR ENGINEER

Aug. 2020 - Jul. 2022

- Core developer for department's flagship automation platform, delivering **critical microservices** (infra provisioning, observability, notification management) with 100% uptime. Helped **save work-hours** across memory division.
- Facilitated **improvements** in test benches of **7+ memory products** by developing features - false positive detection (identifies avg 200/100k test cases per release), failure root cause analysis and failure similarity search.
- Built distributed log pipelines and complementary analytics engine. Worked extensively on search platform Elastic stack and solving **distributed-systems** performance optimization problems.
- Presented internal technical paper as **1st author** on 'Failure Analysis, Triaging and Test Optimization using Data Science and Analytics' at Samsung India Research TechCon 2020, regularly submitted papers and conducted technical demos at internal research seminars.

SOFTWARE DEVELOPER INTERN

Jan. 2020 - Jul. 2020

- Contributed with multiple research assignments, experiments and 2000+ lines of code, working extensively on **Cloud Computing** and the forward-looking concept of Composable Infrastructure. Used open-source tools like OpenStack, Puppet-Razor, Juju and Django.

Physiz AgTech

Mumbai, India

SOFTWARE DEVELOPER INTERN

May 2019 - Jul. 2019

- Worked on backend development for company's core product, implemented critical features and **developed 30+ API** routes to enable team to stay on track for global release.
- Designed and built an automated testing solution from scratch using Mocha and Chai.js. Increased code coverage from 0% to over 60%.
- Took on **DevOps responsibilities** to set up CI/CD, Kong API Gateway and manage IoT devices over the cloud through Balena.

Projects

DAMN - SSH Keys Distribution and Management Network

SKILLS: GO, DATABASE MANAGEMENT, SERVER MANAGEMENT, SHELL SCRIPTING, OAUTH, API DEVELOPMENT, ORM

- Developed web application for securely requesting, granting and revoking access to servers on the go by managing SSH keys of all node servers through a central server with the provided web portal.
- Engineered backend using Golang while working with various web APIs, OAuth protocol, scripting and databases technologies.

Weenix Operating System

SKILLS: OS DEVELOPMENT, CONCURRENCY, SYSTEMS PROGRAMMING, KERNEL DEVELOPMENT

- Building an OS based on Unix and supporting features such as multitasking, virtual memory, terminal emulation and polymorphic file system as part of Brown University's graduate level Operating Systems course.

Golang Compiler

SKILLS: COMPILER DESIGN, DATA STRUCTURES, ALGORITHMS, C/C++, FLEX, BISON, DEV OPS

- Developed a partial compiler (lexical analyzer) with error reporting and error recovery for Go (Golang) programming language.
- Created an additional web application for end-to-end demo of the compiler.

Task Queues Implementation

SKILLS: PYTHON, FLASK, RABBITMQ, CELERY, SQL, DOCKER, DOCKER-COMPOSE

- Used Celery to implement task queues with RabbitMQ as the broker so as to allow the uploading of data sets on a server to be an asynchronous task with the option to check the status and revoke task.