

Deep Sharma

(623) 297-7216 • dshar112@asu.edu • sharmacodes.com • linkedin.com/in/deepsharma993

EDUCATION

Master of Science in Computer Science - Arizona State University, Tempe, Arizona (GPA: 4.0) May 2026

Bachelor of Technology in Computer Science Engineering - SRM Institute of Science and Technology, India (GPA: 3.5) May 2021

- Courses: Data Structures, Algorithm Analysis and Design, Database Management Systems, Computer Networks

TECHNICAL SKILLS

Programming Languages: Java, Python, JavaScript, C, C++, Kotlin, Bash

Front-End: HTML, CSS, React.JS, Bootstrap, Tailwind CSS, Redux, Jetpack Compose

Tools, Databases: Node.JS, MySQL, MongoDB, PostgreSQL, Azure DevOps, AWS Cloud formation, Jenkins, Terraform, Kubernetes, Git

Certifications: HashiCorp Certified: Terraform Associate (002), Microsoft AZ-900, AZ-104, AZ-204, AZ-400

EXPERIENCE

Software Engineer 2, Capgemini, Pune, India July 2022 - July 2024

- Developed a one-click solution using ServiceNow, Python, Terraform, and Azure DevOps to automate cloud resource provisioning, decreasing deployment time by 50%.
- Led a team of two junior developers for Terraform code remediation project, designed solutions, planned sprints, and reviewed and closed backlogs following project timeline.
- Automated 100 EC2 spin-ups/turn-offs utilizing AWS Lambda, cutting down monthly costs by approximately 40%.
- Deployed Bitbucket Datacenter on AWS leveraging a Blue-Green approach, ensuring 99.99% availability, with CloudFormation, Docker, and EKS (Kubernetes) for RDS.
- Recognized with Capgemini Propeller Award' twice (2021, 2022) for outstanding efforts in upskilling multiple teams in DevOps practices and tools.

Software Engineer, Capgemini, Pune, India June 2021 - July 2022

- Designed Azure and AWS Terraform modules to streamline cloud infrastructure setup, accelerating DevOps adoption across organization and cutting time of fresh provisioning by 30%.
- Implemented AWS Lambda to create a ServiceNow incident whenever websites went down, alerting the Ops team to take quick action and reducing website downtime by 25%.
- Established approval gates and policies in 20+ Azure DevOps repositories, enhancing code quality and diminishing errors.
- Documented code design changes, drafted 10+ SOPs, and provided 15+ knowledge transfers (KTs) to operations teams, improving adoption of DevOps practices and infrastructure as code across organization.

Software Engineering Intern, Capgemini, Mumbai, India January 2021 - June 2021

- Conducted an in-depth analysis of a countrywide traffic accident dataset covering 49 U.S. states.
- Leveraged Azure Blob Storage for efficient data management. Key contributions include ETL operations with Azure Data Factory and Databricks, and visualizations with Power BI.
- Unveiled trends include monthly accident-prone cities, state-wise accident statistics, and severity ratios based on weather.
- Visualized findings through data visualization widgets, contributing to informed decisions in road safety.

Web Developer Intern, KGE Technologies Pvt. Limited, Chennai March 2019 - June 2019

- Redesigned client websites with NodeJS, Django, and PostgreSQL, enhancing performance and adding new features.

PROJECTS

TerraZure [github.com/hadessharma/terraform-react-node-Azure-deployment]

- Empowered users to deploy cloud resources via the Terraform framework, TerraZure offers a user-friendly GUI and forms, allowing for easy deployment management without requiring deep Terraform knowledge.
- Built frontend in React JS with Firebase authentication, and implemented cloud provider authentication (AWS and Azure) using service principals and IAM.
- Adopted NodeJS for backend with an Express server, leveraging Terraform modules receiving variable input from a frontend form.

Phishing Website Detection [github.com/shoviknandy/Phishing-detection---ML]

- Engineered an application to detect phishing URLs using XGboost Algorithm.
- Trained model incorporated 26 features, and 10,000 diverse URLs, and achieved an impressive 96% accuracy rate.

Rap Sheet

- Developed a rap sheet system for law enforcement departments, allowing cops to store and manage data on lawbreakers.
- Utilized object-oriented programming with C++ to design a CRUD (Create, Read, Update, and Delete) application.
- Launched as a CLI-based application with MySQL as backend to store data, application later incorporated JavaScript for its front end.