

Dhruv Patel

dhruvrpa@usc.edu | 213-249-4951 | LinkedIn: dhruvrpatel16 | GitHub: dhruvp-8

2707 Portland Street, Los Angeles, CA 90007

EDUCATION

University of Southern California

Master of Science in Computer Science; GPA: 3.92/4.0

Los Angeles, CA

Expected May 2020

Relevant Coursework: Analysis of Algorithms, Foundations of Artificial Intelligence, Web Technologies, Information Retrieval & Web Search Engines, Augmented, Virtual & Mixed Reality, Software Architectures

Dharmsinh Desai University

Bachelor of Technology in Computer Engineering; GPA: 4.0/4.0

Nadiad, India

Aug 2014 - May 2018

TECHNICAL SKILLS

- **Languages:** Python, Go, Java 8, JavaScript, TypeScript, C, C++, Swift, PHP, Bash
- **Systems:** Linux, Weenix, bpf, netconsd
- **Technologies:** Django, Flask, Laravel, Spring Boot, Node.js, Express.js, Polymer, Angular, React, Redux, GraphQL, D3.js, REST, Boto3, Redis, RabbitMQ, Elasticsearch, Grafana, Celery, Airflow, Kafka, Storm, MySQL, Postgres SQL, MongoDB, Firebase, Cocoa Pods
- **Cloud & Infra:** AWS (Lambda, Step Functions, S3, IAM, DynamoDB, CloudWatch, API Gateway, SQS, SNS, VPC, EC2, GuardDuty, Kinesis Firehose, EKS, ECS, EMR), GCP (Big Query), Terraform, Jenkins, SaltStack, Vagrant, Docker, Kubernetes, Envoy

PROFESSIONAL EXPERIENCE

BlueJeans Network

Software Engineer Intern

San Jose, CA

May 2019 - Aug 2019

- **Smart Meetings:** Developed a feature on BlueJeans Web Client to assign Action Items, Decisions and Reviews to specific members present during a meeting by driving the project from conceptualization to testing and execution. (MongoDB, Java, Spring Boot, React)
- **Network Monitoring:** A real-time **on-demand network logging system** with improved cost savings
 - * Built a system to monitor **VPC Flow logs** on AWS using **Kinesis Firehose** (for batching the log stream data), **S3** (for storing historical data), **Lambda** (written in Python using Boto3), **Elasticsearch** (for indexing the logs) and **Grafana** (for visualization)
 - * Engineered a solution to reduce the size of Elasticsearch Cluster by **50%** to provide massive economics on log indexing by requesting logs on-demand via API Gateway
- **Alerting Tools:** A set of tools for alerting suspicious behavior on AWS to the DevSecOps team
 - * Designed and developed an alerting system to notify severe GuardDuty Findings in real-time using Lambda, CloudWatch and SNS
 - * Implemented an **IAM Policy Checker** to detect highly permissive policy changes by comparing it with previous policy versions using AWS Zelkova (Beta), IAM, Lambda, CloudWatch and SNS
 - * Alerting the configuration changes in NACLs, Route Tables and Security Groups using Lambda, CloudWatch, SQS and PagerDuty
- **Serverless CI/CD Pipeline:** Automating deployments of Lambda Functions on AWS
 - * Developed a **deployment pipeline** for deploying Lambda Functions on AWS using **Terraform** for provisioning infrastructure (storing state information on DynamoDB to prevent locking) and **ECS Jenkins** with Auto-healing to update the lambda code from BitBucket using S3

Information Sciences Institute

Graduate Research Assistant

Los Angeles, CA

Oct 2018 - Apr 2019

- **Knowledge Graphs:** Automating integration of complex geological models to forecast effects of human activities on natural resources by developing REST APIs in Java with **SPARQL for querying Knowledge Graphs** and UI in **Polymer with Apollo Client** (for GraphQL queries) along with **D3.js** to describe various ontological relations.

Indian Institute of Technology, Bombay

Software Engineer Intern

Mumbai, India

Dec 2017 - Apr 2018

- **Lexical Simplification:** Designed and developed APIs in Python using **Django REST Framework** for QA System, user-base management (with OAuth 2.0 API Security) and Wordnet Visualizer. Used **React with Redux** to populate the data in frontend.

PROJECTS

- **Arancia - Distributed KV Store** 🗄️: A cloud based distributed Key-Value Store based on **2PC protocol with consistent hashing**
 - Implemented a **Proxy Server in Go** to serve authenticated requests to KV Store from multiple clients via one ingress and egress port
 - Developed the REST Client using Flask with Unicorn Server and containerized the application using Vagrant
 - Scaled out **Celery with RabbitMQ as the message broker** for **scheduling regular snapshots of KV Store (replication factor 2)**
 - Developed a performance benchmark tool and captured **P99.9 GET to 14ms** and **P99.9 SET to 60ms**
- **Twitter Stream Analysis** 🗣️: Real-time data streaming using Twitter4j, Zookeeper, Kafka and Storm
 - Using Storm Topology to **generate a list of popular words used in twitter**. Ingested data from a Storm spout and a Kafka spout and processed downstream using Storm Bolts. Developed a real-time word cloud for analysis
- **EBay Product Search:** Developed a **web client using Angular 7** and a **mobile client using Swift in iOS** for searching product catalog on Ebay. Incorporated autocomplete search using **Solr**. (Node.js, Angular 7, Express.js, Swift, Cocoa Pods, iOS, Solr, Lucene)
- **P2P File Transfer** 🔄: Designed a **real-time browser-to-browser communication system** for transferring files from one device to multiple devices without uploading files to a remote server using **WebRTC**. (Node.js, AngularJS, Peer.js, SendGrid, Heroku)