

# Nija Mashruwala

Senior DevOps/Prod Eng/SRE

nija.mashruwala@gmail.com

650 492-5668

## Work Objective:

I am more of a generalist than a specialist. The collaborative work environment that fits me best has leadership with a strong and coherent vision, management who make decisions based on hard data and realistic projections, coworkers who challenge and inspire me professionally and personally, and, most importantly, a company where we create terrific products that enrich the world.

## Things I enjoy working with and making:

Designing scalable and maintainable systems; mentoring; Unix/Linux of any flavor; smooth deploys and services migrations; keeping it simple, efficient, and thoughtful; working smarter and not just harder; organized and searchable documentation and training guides; tools that enable people to get their work done faster and better in less time; robust, elegant, and secure setups

## Interesting Technologies List:

Programming (Python, Go), Cloud Platforms and Content Delivery (AWS, Akamai), Infrastructure as Code, (Terraform Cloud), Observability (Datadog, Honeycomb), Virtualization/Containerization (Docker/ Kubernetes), Continuous Deployment (Spinnaker, CircleCI, Jenkins, TeamCity), Configuration Management (Puppet, Chef, Ansible), DNS, Load balancing and Traffic Shaping (F5, HAProxy, nginx, AWS), Key-Value Data Stores (Zookeeper, Cassandra), Relational Databases (MySQL, Postgres), Operating Systems (Ubuntu, CentOS, Amazon Linux)



## LaunchDarkly

2021 2022

DevOps/Platform Engineer.

- Key result: Increased developer velocity by migrating Terraform workspaces and Ansible AWS resources to Terraform Cloud
- Key result: Verified regulatory compliance by running regular Disaster Recovery tests
- Key result: Increased developer observability by writing a production-ready Terraform module for Honeycomb Refinery in AWS, and creating a reusable Spinnaker pipeline to deploy to and upgrade Refinery in multiple environments
- Key result: Enabled long-term Datadog log retention to comply with Federal data retention policies
- Key result: Migrated Spinnaker auth from Google Groups to Okta
- Key result: Increased observability of CockroachDB by adding Datadog monitors and alerting



## Salesforce

2016 2020

Senior SRE/Production Engineer.

- Key result: Standardized account addition, deletion, and group management through a custom Go client-server which sped up the largest on-call task
- Key result: Continuous deployment pipelines for EMR and AMI standardization across all clients allowing for developer self-service
- Key result: Wrote Chef cookbooks to standardize Kafka, Kafka sidecar apps, Zookeeper, and Zookeeper sidecar apps
- Key result: Docker Registry set up and maintenance for all Dockerized deployments
- Key result: Wrote modularized Infrastructure-as-Code Terraform modules to allow for rapid infrastructure deployment

- 👉 Udacity 2015 2015
- DevOps (Generalist) and Network Engineer.
- Key result: Due to policy and process changes, ad-hoc response teams composed of developers, managers, and on-call ops people are able to quickly and calmly handle incidents and planned outages in an auditable and sustainable way.
  - Key result: Udacity gained the ability to set up secure endpoints between the AWS, Google App Engine (GAE), and corporate office infrastructure. This was underlying infrastructure for faster database migrations and replications, and disaster recovery.
  - Key result: Documented and set up the continuous deployment process for multiple micro services.
  - Key Result: Built out and upgraded corporate network and VPN with tightened security and improved usability
- 👉 Airbnb 2014 2015
- DevOps for Production Infrastructure.
- Key result: AWS security became auditable and consistent company-wide for all services and developer accounts.
  - Key result: High response SRE and Ops duties - mentoring, training, cross-team collaboration, custom dashboards.
- 👉 Roku 2014 2014
- Senior Networks Operations Engineer / DevOps for Roku's production and supply networks.
- Key result: Multiple scalable and templated Windows deployments used in production and test environments
  - Key result: Internal tools scripting and development used to run internal reports
- 👉 Apigee 2012 2014
- Telco DevOps for Apigee's on-premise installation for AT&T
- Key result: Delivered an iterable version of Apigee's on-premise API platform solution, complete with monitoring, documentation, configuration management, and data migration support built from the ground up to support ATT's text-to-speech service
  - Key result: Trained and hired Accenture consultants to support the newly built infrastructure, including the configuration management and custom tools built on a modified version of OpenStack
- 👉 Jelli 2011 2012
- Production Operations and Corporate IT
- Key result: Increased uptime and stability due to increased VM and network redundancy and better monitoring
  - Key result: Increased physical and data security and lowered monthly cost due to data center



Sling Media

2007 2011

## Production IT / DevOps

Common web technologies - LAMP (Linux/CentOS, Apache/Tomcat, MySQL, PHP/Python), Cloud Services (Amazon Web Services, RightScale, Akamai), and local and global Network Traffic Management (F5 BigIP load balancers); Server virtualization (VMWare ESX and ESXi, RightScale/AWS); racking, stacking and configuring servers in data centers (HP Chassis & Blades, HPOA, Rackable); Environment build outs and migrations (dev, staging, qa, production); Application deployment (public-facing) and 24/7 on-call support; Troubleshooting across applications, systems, and networks; bash shell scripting; Technical writing & documentation; customization and management of a secured content management system (Drupal, PHP) and ticketing system (Kayako); monitoring (Zabbix, PHP); project management and Agile development support (SDLC); interviewing and hiring

- Key result: Push-button release management and documentation of micro services, including staggered rollout, traffic shaping, and A/B testing
- Key result: Due to Dev and Ops inclusion during the white-boarding and PoC phases, micro services were shepherded through the SDLC resulting in fully monitored services and eventless production deploys. This drastically increased uptime and reliability of all services
- Key result: Dynamic, self-service monitoring of deployments and services for developers
- Key result: Dynamic and up-to-date inventory of physical and virtual assets

# Nija Mashruwala

Senior DevOps/Prod Eng/SRE

*nija.mashruwala@gmail.com*  
650 492-5668

## Continuing Education and Education:

Hackbright - 3-month coding bootcamp focusing on Python and full-stack engineering

Completed Stanford mini-courses: Relational Algebra (100%)

Foothill/De Anza Colleges studying Computer Science, Enterprise Computer Networks, Network Security and Math

Rensselaer Polytechnic Institute studying Computer Engineering; involvement with the ACM (running cable, remote access, systems and network administration)

## Additional Information:

On the career and education front, I continue to take classes and workshops, as well as study subjects in my free time. On the personal front, my love of and enthusiasm for technology started in a middle-school desktop publishing class. I've found that I excel at explaining complex concepts in an easy-to-understand way. Because I have diverse interests, from open-source software to sustainable backyard gardening, I easily build strong interpersonal rapport and strengthen teams.