DAVID IAN **VASANDANI**

ADDRESS 2215 Scott Ave, Los Angeles, CA 90026 • **MOBILE** (626) 484-2485 • **EMAIL** david@vasandani.me
LINKEDIN /davidvasandani • **GITHUB** /davidvasandani • **TWITTER** @davidvasandani

PROFILE

Senior Operations Engineer with experience in managing servers and containers in multiple clouds. I take pride in my ability to break down problems and derive creative solutions, based on thorough research and following best practices. My positivity and energetic communication abilities allow me to thrive individually and in teams. I'm passionate about automation, containerizing, monitoring, and learning new technologies by reading documentation, blogs, and on-hand experimentation.

SKILLS

- Packer, Terraform & Terragrunt, Consul, Vault, Docker stack, and Ansible.
- AWS Beanstalk, ECS, CloudWatch, ASG, SpotFleet, DMS, IAM, AppMesh and CodePipeline.
- Rancher, Kubernetes, Helm, ChartMuseum, AWS EFS PVC, ALB and Nginx Controllers.
- Excellent **documentation** and **workflow management**.
- **Driven** individual, mindful of multi-tasking, deadlines, and providing quality deliverables.
- **Energetic verbal** communication skills.
- Proven accomplishment **lecturing** in classrooms and **mentoring** colleagues.

EXPERIENCE

Lead Infrastructure & Build Engineer @ sweetgreen • Los Angeles, CA • 2019 – Present
- Broke down a locally-run Terraform monorepo into smaller repositories deployed by CircleCI, each with its own backend state; this reduced the blast radius and increased developer velocity.
- Decoupled an Ember frontend from a Rails backend into a static, single-page application to increase deployment frequency.
- Built out blue-green infrastructure to enable canary deployments and fast rollbacks using weighted target groups on AWS ALBs.
- Initiated and documented a tech-wide, on-call rotation in OpsGenie, leading the team to incrementally improve service run-books.
- Continuously improving developer experience; examples include introducing localstack, enabling hot reloading for local development, and granting bastion access via IAM.
- Introduced and leveraged Gruntworks library of reusable, production-tested infrastructure modules to make sweetgreen's infrastructure codebase more sustainable.
- Evangelized and led the Datadog implementation, centralizing observability and reducing costs.
- Standardized infrastructure and deployment processes across 15+ existing services, reducing the time to launch of new services and overhead costs associated with onboarding new team members.

Senior Operations Engineer @ LearnZillion • Remote • 2016 – 2019
- Improved developer experience by leveraging Docker for Mac.
- Migrated self-hosted EC2 instances of Elasticsearch, Redis, and Postgres to AWS AES, ElastiCache, and RDS, increasing reliability and decreasing overhead.
- Reduced infrastructure cost by moving testing environments to ECS on SpotFleet.
- Leveraged short-term Reserved Instance contracts to further reduce infrastructure costs.

- Migrated from CircleCI 1.0 to 2.0.

Senior Systems Engineer @ Humanitarian OpenStreetMap Team • Remote • 2016 – 2018
- Migrated applications from a broken cloud server running an old version of Docker where everything was manually created without Dockerfiles to a Docker Swarm cluster provisioned by Ansible and Terraform, and continuous deployment with CircleCI.
- Lead weekly community meetings to communicate changes with stakeholders.

Automation Engineer @ Warby Parker • New York, NY • 2014 – 2016
- Designed an AWS VPC with security as a top priority and migrated Rackspace Public Cloud instances.
- Moved to immutable infrastructure employing auto-scale groups, blue-green, and canary deployments with Packer, Ansible, Terraform, and Consul.
- Utilized CloudWatch metrics to auto-scale application servers based on CPU and memory and worker servers based on queue depth.
- Monitored servers using collectd and services using Consul with Sensu for alerting.
- Increased deployments from 1x/week to over 400x/month over two years.
- Cut over 40% of the infrastructure bill by rightsizing the servers and autoscaling.