

Brandon Willett

Education

UNIVERSITY OF ROCHESTER

- ◇ Graduated in 2018
- ◇ B.S. in Computer Science (3.82)
- ◇ B.A. in Mathematics (3.80)

Skills

BACKEND

- ◇ Go, Python, Java, and Scala
- ◇ Message Queues (Kafka, RabbitMQ)
- ◇ DAG Pipelines (Airflow, Luigi)
- ◇ Transactional Databases (Postgres, Cass)
- ◇ Analytical / Columnar Databases
- ◇ Database Sharding
- ◇ Caches (Memcached, Redis)

PLATFORM

- ◇ Version Control (Git)
- ◇ CI/CD (Github Actions, Gitlab)
- ◇ Telemetry (Prometheus, Datadog)
- ◇ Containerization (Docker, Containerd)
- ◇ Build Tools (Rake, Bazel)
- ◇ Infra as Code (CF, Terraform)
- ◇ Service Discovery (DNS, Consul, Envoy)
- ◇ Reverse Proxies (NGINX, HAProxy)
- ◇ Load Balancers (L4, L7)

CLOUD

- ◇ AWS (ALB, NLB, EC2, RDS, S3, Lambda)
- ◇ GCP (GLB, GCE, GCS, Cloud SQL)
- ◇ Azure (ALB, VMSS, ABS)
- ◇ Kubernetes (ask me about any resource)

A young New-York-based developer with experience in platform dev. Always believing that any problem can be overcome with kind communication, a good sense of humor, and just persisting everything to S3.

📞 203 258 2721

✉ brandon@willett.io

🏠 391 8th Street, New York NY 11215

Experience

DATADOG

New York, NY

Site Reliability Engineer

May 20 – Jun 23

- ◇ foo
- ◇ foo
- ◇ foo
- ◇ foo
- ◇ foo
- ◇ foo
- ◇ foo
- ◇ foo
- ◇ foo

ACTIONIQ

New York, NY

Distributed Systems Engineer

Jun 18 – Apr 20

- ◇ Designed, advocated for, and implemented our strategy for autoscaling query execution on EC2, reducing our AWS bill by more than 50%
- ◇ Drove scalability improvements for the OSS job scheduling platform Luigi, and contributed some of that back upstream
- ◇ Maintained the Prometheus cluster (later, Datadog) and led internal talks and workshops on best practices around metrics & observability
- ◇ Enabled dynamic configuration and artifact discovery
- ◇ Brought some of the first Terraform modules to AIQ, used them to build the first version of our streaming ingest platform using IaC

EDU.CHAT

New York, NY

Software Engineering Intern

Jun 17 – Aug 17

- ◇ Worked closely with a team of three other engineers, using JIRA and Git, to create a bot which classified and answered student questions
- ◇ Developed, refined, tested and integrated the new NLP question-matching model with the overall Edu.Chat platform
- ◇ Learned Python implementations for message passing, parallelization and task queue systems to distribute the workload (like Celery and Redis)