Misha Bergal

Sr. Software Engineer

Profile

I am a very experienced software engineer who has worked as a team lead, a principal engineer in a team, and a single contributor. I am a generalist, so technologies are secondary to building cool, useful stuff. I believe that people should be kind and emphatic, that they should work in teams, and that TEAM == PRODUCT.

Languages: Python 3, TypeScript, C#, Ocaml

Styles: OOP, DDD, FP, Type Driven, Test Driven

Databases: Postgres, SQL Server, Mongo, Firebase

Architectures: REST, Events, CQRS

Stacks: Node, .NET Core

Front-end: Angular, React

Clouds: AWS, GCP, Kubernetes

Employment History

Sr. Software Engineer at SmileDirectClub, Nashville, TN

May 2021 — November 2022

Was helping friends build something that has never been done before. Python 3, CUDA, AWS (Lambda, ECS, Batch, EventBridge), GitHub Actions, Terraform (CDK), DataDog

Single backend software engineer on an AI team (15 people) tasked with building 2D to 3D software converting 2D teeth pictures into a 3D teeth mesh, that is then used to create custom alignment treatment plans.

Designed and implemented ML inference-running back-ends (Python 3, PyTorch, CUDA, FastAPI), CI/CD pipelines (GitHub Actions), and infra on AWS (Lambda, ECS, Batch), model monitoring software, and integration between ML services and the company's business processes (EventBridge, HTTP APIs).

The problems in AI space are different from straight-forward line-of-business applications and require a special set of methods and infra, since:

- they need large amounts of development-time data (training and testing) and runtime (weights)
- significant compute resources that are CPU and GPU bound

I developed and operated:

- Data storage and retrieval infra (based mostly on DVC and S3)
- Model training and testing pipelines

Details

1317 Dolen Place
Iowa City, 52246
USA
319-9368228
misha.bergal@gmail.com

Links

Stack Overflow
GitHub
No Twitter
No Facebook

Skills

TypeScript, Python 3, C#

ReasonML/Ocaml, F#, Elm, a bit of Haskell and even C++

Node.js, .Net Core, Python 3

React, Flux, Redux, Elm architecture, unidirectional data flow, MVVM

Tactical DDD, OOD/OOP, Functional Programming (know what Ma -> (a -> Mb) -> Mb is), ORM, fast (Unit) and slow (integration,acceptance) testing

HTTP API (RESTful and not)

AWS (EC2, Lambda, EventBridge, RDSl, SQS, EKS and other services), GCP (GKE and other services), Kubernetes, Helm, Pulumi (TypeScript), CDKTF (TypeScript), Prometheus, Grafana

GraphQL, Hasura, Firebase Real Time Database, BigQuery, MongoDb, MySQL, Oracle 11g, SQL Server

- Model hosting runtime allowed for efficient and cost-effective model execution in the cloud. This runtime supported different execution methods (Lambda, ECS, Batch) depending on model requirements such as (CUDA or CPU, response time, compute cost, etc). This included the Python code for runtime itself, CDKTF infra definitions, and DataDog metrics for operations.
- CI/CD pipelines for model runtime, integration services, and iOS mobile apps.

Also, a big part of my job was to provide all kinds of support to a dozen or so ML engineers. These included git, CI/CD, Python, pytest, and general software development education.

Principal Backend Engineer at ReelHealth, Los Angeles

September 2020 — May 2021

Great tightly knit team and an important mission in COVID times doing front-end and back-end in Ocaml! Node.js, ReasonML/Rescript, React, GraphQL, Hasura, AWS, PostgreSQL, k8s

As a part of a very small team, developed (mobile and web front-ends, back-ends, infra) and operated (monitoring and operational fixes) a set of applications for COVID testing on the sets of show productions for Netflix, Amazon, and other studios. The team was so I did all parts of it from front-end development in React/Rescript to setting up and operating PostgreSQL AWS RDS and Rancher k8s cluster.

I would like to note that programming in Ocaml was a very nice experience that I would love to have more of.

Principal Software Engineer at IDx Technologies, Coralville, IA

October 2018 — August 2020

First-ever FDA-Approved autonomous diagnosis AI device, startup team. C# .NET Core, React + TypeScript, Azure DevOps, AWS (EC2, RDS, EKS, and others), k8s, Pulumi + TypeScript, MySQL, Nginx, Prometheus, Grafana.

Led one of the teams that developed:

Class "C" (Death or Serious Injury is possible as a result of device malfunction) medical device software - the first FDA-approved autonomous diagnostic device. This included FDA-compliant SDLC with extensive requirements documentation and tracing, unit, system, and integration testing. The front-end was written in React+TypeScript and C# running on .NET Core using HTTP APIs to communicate with external services.

Infrastructure and process support software (there is a lot to class C devices besides the actual software).

Cloud applications to support the integration of the above software with other health information systems running on AWS EKS (managed k8s).

HIPAA-compliant AWS Cloud infrastructure for the above software with infrastructure as code via Pulumi.

We brought up many juniors to mid- and senior levels, and I consider that to be one of the significant non-business achievements.

Senior Software Engineer at Pear Deck, Iowa City, IA

April 2016 — August 2018

Startup with the mission of creating safe educational spaces for kids. JavaScript and Flow, Node, Angular, Java, Python, Elm, Reason ML+React, GKE, Firebase, Mongo, k8s, Helm, Segment, BigQuery

Developed a web application to help teachers engage students, especially in challenging environments with large class sizes, so that everybody could be heard and safe. Consistently pushed out new features while keeping the system up and running and bringing in new developers.

Did full cycle of product development - from UX design to implementation to ongoing support, cool stuff like Flashcard Factory https://www.youtube.com/watch?v=afmp1zqPyQI (this video is by our user, not us!). We were able to do that by working in small vertical teams, constantly aligning, and having a shared vision of what the product, code, and process should be. We never left anybody alone with a problem, situation, or task.

As one of the two most-experienced devs, I put steady efforts into gradually improving product architecture, code quality, system maintainability, and, most importantly, creating a shared vision of where we were going technically as a development team.

Implemented a collection of micro-services running on GKE + AppEngine.

Provided product/UX design ideas, suggestions, and feedback.

Did all DevOps stuff like deployment procedures, k8s configurations (including automatic load balancing), configured GCP. Monitored application up-time and performance, detected and fixed application and infrastructure problems.

Education

6 years (B.Sc.), Leningrad Polytechnic Institute (Go Polar Bears!), Leningrad, USSR

References

References available upon request