Hongfei Zhou

Email: mich.zhou1989@gmail.com

> I have 11+ years of technical experience in backend development, testing, and deployment. Familiar with Google Cloud Platform and Kubernetes. Strong problem-solving and analytical skills, combined with a passion for staying up-to-date with emerging technologies. Have worked on developing backend, frontend, CI/CD, and automation platforms.

Key Skills

- **Programming Language**: Kotlin, Java, Python
- **Backend**: Spring Boot, JPA, Spring MVC, RESTful, GraphQL, GRPC
- **Cloud**: Google Cloud Stack (GCS, Pub/Sub, Big Query, DNS, Monitoring, etc), Kubernetes, Docker.
- **Database**: PostgreSQL, MySQL, DBT, Big Query, OpenTSDB, Big Table, MongoDB.
- **Tools**: Gradle, Maven, Jenkins, GitHub Action, Skaffold, Kustomize, Helm, Git, Shell.

Experience

Senior Software Engineer @ Crisp Inc. (Motiion Inc.), *from 2019.04 to present*

> Responsible for various projects and tasks covering the backend, infrastructure, monitoring, and CICD.

- Backend Developer for adding new features and maintaining existing features of the system
  - Add new APIs like Location, Tag, Annotation
    - Set up GraphQL support with **kotlin-graphql** for new services.
    - Creating **PostgresSQL** tables and service entity models to enable the new API.
  - Add Credential Verification API to support triggering and getting the status of verification running status
    - Set up **GRPC** to enable communication between different backend services.

- Support GraphQL **DataLoader** to boost the API performance.

○ Develop DSL to support WebDriver downloading

  - Use **Selenium WebDriver** to support writing code to automatically operate Chrome to download reports.
  - Wrapped with a self-developed **DSL** to lower the effort of writing browser automation code with better logging and debugging.

○ Develop a new task-running framework for executing daily tasks with autoscaling

  - New framework uses kotlin **Reflection** to reduce the complexity of runner code.
  - Provided better **object-oriented** programming capabilities.
  - Eliminates all callback style coding and improves code cleanness for easier maintenance.
  - Set up metrics to enable **Horizontal Auto Scalar** to control the runner replicas.

○ Add support to download various vendor reports from different portals

  - Download the reports from various sources like Amazon (API), Walmart (Browser), Loblows (GCS), etc.
  - Parse the report files and save data to **Google Big Query**.
  - Using **Pub/Sub** to connect report downloading and parsing services.
  - Write **DBT** to provide a cleaned version of the **Big Query** table for downstream teams.

○ Support Multi-Factor Authentication to access portals with MFA enabled

  - Integrate with Gmail to receive verification emails.
  - Modify the WebDriver DSL to support extracting **OTP** code from text messages and emails.

○ Support Task/Resource Reservation with heartbeat check

  - Add reservation service to support locking a resource for exclusive use.
  - Integrate the service with the heartbeat service to ensure no lock starvation.

- Support Looker Embedded URL generation

  - Integrate with **Looker** to generate an embeddable URL with auth signature.
  - Embed the URL to the front end to display Looker's dashboard.

- Switch from **Quartz** scheduling to **Spring** Scheduling

- Developer for infrastructure setup and maintenance

  - Set up metrics collection, monitoring, and alerting

    - Add support in backend services to expose metrics.
    - Set up **Prometheus** to collect metrics and forward metrics to **Stackdriver**.
    - Set up **Grafana** to new metrics in Prometheus.
    - Create Alert Policies on Stractdriver and alert with PagerDuty and Slack.

  - Maintain backend service deployment with Kubernetes manifests

    - Write composable manifest for Kubernetes **Deployments**, **Services**, **Ingresses**.
    - Use **Kustomize** to combine the manifests and use **Skaffold** to automate the build and deploy process.

  - Set up PGAdmin instance to enable production database query

    - Set up Kubernetes Deployment to run PGAdmin.
    - Make the PGAdmin configuration version controlled on GitHub.
    - Expose the instance behind **Google IAP** with proper **LoadBalancer** and **SSL Termination**.

  - Set up a staging environment for penetration testing

    - Set up an isolated environment similar to the production.
    - Enable public network access with **Ingress** and **Cloud DNS**.

- Developer for various OPS tasks

  - Enable the Jenkins pipeline to build modules with the code change.

    - Support code change detection.

- support dependency check to allow the module to build if a depending module has code change.

  ○ Migrating from Jenkins pipeline to GitHub Action Workflow for CICD

  - Write GitHub Action to perform CICD steps for individual modules.
  - Enable various external system support with GitHub Action such as Firebase and Slack.

  ○ Add key rotation to the developer environment to ensure short-lived keys.

  ○ Upgrade the Kotlin version.

  ○ Add a status page for the company's product.

Senior Software Engineer @ StubHub, eBay Inc, *from 2012.6 to 2019.04*

- Lead developer for **Discount Cofund**. Implemented the feature and update iLog integration.

  Add extra price items to the system and update the integration with iLog to take extra parameters for new price items.

- Lead developer for **Optimizely AB Testing**. Integrate the AB Testing with Optimizely.

- Lead developer for **Price Markdown**. Design the structure and implemented the feature for marking down prices.

  Leverage Java 8 & multi-threading & Couchbase to perform high-performance price markdown. Giving the business team the ability to adjust the price throughout the entire site.

- Lead developer for **Test Data Service**. Designed the architecture of the system and implemented the backend of the system.

  Using Java with Spring MVC to build a web service for serving test data. The system is built with MongoDB for data storage and Jenkins for data generation.

- Lead the development of **Test Execution System**. Designed the structure of both frontend and backend. Implemented frontend code and major backend code.

  Built a system for executing automation testing. Using Java/Spring MVC/ AngularJS to build the system. Data storage is done by MongoDB. Test execution with Jenkins. The system is also integrated with JIRA to periodically update TC information.

- Solely designed the structure of **API automation framework** and implemented both **API framework** and **responsive web page framework**.

  > Using Java and URLConnection to build an automation framework for testing SOAP and RESTFul API. Integrated the system with the existing UI testing framework, including life cycle management, logging, and report generation.

Education

- **Bachelor of Engineering, Computer Science**, Shanghai Jiaotong University, 2008 - 2012.