

Hiromi Shikata

Full Stack Developer

CONTACT

Website <https://hiromishikata.github.io/profile/>
Email shikata@uminoseisaku.com
Phone +81-50-5354-8192

LOCATION

Region Tokyo, Japan

PROFESSIONAL SKILLS SUMMARY

Backend Development (Expert): RESTful API, Clean Architecture, DDD, Swagger, Serverless, AWS, Docker, docker-compose, MySQL, Postgres, DynamoDB, MongoDB, Redis, TypeScript, Go, Python, Rust

DevOps (Expert): CI/CD, CircleCI, Testing Automation, Documentation Automation, SchemaSpy, CloudMapper, e2e Testing

Infrastructure (Senior): AWS ECS Fargate, AWS Lambda, Terraform, Ansible

Frontend Development (Intermediate): React, Fargate, Storybook, BroserStack, Vue, Angular

WORK EXPERIENCE

2020-03-01 TO CURRENT

Back-end development & Management (2days/week) at Urban development startup company

Clean Architecture, Micro Service Architecture, Monolithic Repository, Ubuntu, TypeScript, Dart, Open API Generator, Express, Flutter, firestore, Postgres, Feature Flags, firebase, firebase, firebase functions, firebase config, Prisma 2, OpenAPI, Docker, Docker Compose, GCP CloudSQL, GCP App Engine, CircleCI, GitHub, GitHub Issues, GitHub Projects, github-flow, Zapier, Slack,

- They didn't have a CI / CD environment for a mobile app that was already in operation, so it was a project to help build.
- I have set up from automatic deployment and have set up integration tests.
- I put off unit testing because the application was already in production and quality assurance was an urgent need, and I needed to change the testable code.
- Since I was talking about tasks in slack because the working hours and places were different, information was scattered, and I could not move forward due to useless communication without prioritizing work, so I created an environment to organize information with task management tools. I have put in place a system that can maintain permeation and discipline.
- At that time, the most type-safe ORM in the TypeScript family seemed to be Prisma, so I introduced Prisma.
- I have moved to a monorepo configuration to get the benefits of monorepo. Since the cost of CI has become critical, I have changed to execute jobs only where there are differences.
- In order to make it easier for the company to expand rapidly and the implementation with only part-time members could not keep up with the business demands and troubles erupted and to incorporate new technology, I hired overseas engineers and built a team centered on overseas members.
- Since there are a wide variety of nationalities and physical locations where I operate, I have prepared rules such as real-time meeting times and timezones, and created an environment that facilitates development such as 1on1 and scrum events.
- Since work other than development was also stuck in large numbers due to labor shortage, I introduced an online assistant service to create a system to reduce the workload in the company and also get involved in development work, manual testing, onboarding, various tools and procedures, communication I have built a system to become a hub of.
- As the number of people increased and the awareness of everyone's eyes and commitment to delivery began to diminish, I changed the system and changed to a system that employs Scrum's experienced PM to form a small team.
- Business side member couldn't decide the specifications and only the deadline was approaching, there was no one who could sort out the business issues, and the person who entered in the PM roll worked better. Since the size of the whole movement became invisible, I increased the number of PM roles and eliminated it from the ambiguity of the business layer, and I are building a system to go around without me.
- Since I couldn't grasp the situation of each engineer, I changed 1on1 to work with PM and both of me, and now I talk to PM about the project and the engineer's own

	<p>task, longer term, interest. I tried to keep the value of working at this company high by grasping the direction and where I want to improve my skills.</p> <ul style="list-style-type: none"> My success in hiring people who are better than the average Japanese engineer in about a month, the cost is about one-third that of hiring Japanese people, and my client appreciate that it is designed according to the business philosophy.
2020-02-08 TO 2020-12-23	<p>Back-end development (4days/week) at Hardware maker</p> <p>Ubuntu, AWS CloudFormation, AWS Redshift, AWS CloudFormation, AWS Batch, AWS CodePipeline, AWS CodeBuild, AWS CodeCommit, ShellSpec, Bazel, Docker, BitBucket, JIRA, Slack,</p> <ul style="list-style-type: none"> It was an improvement project for a data lake that is already in operation. Deployment and testing were done manually, so I automated it. I did the automation with AWS CodeBuild, CodePipeline, CodeCommit because I didn't want to reach the AWS environment and Atlassian. Deploying and testing on a data lake was an interesting project with many considerations that differed from normal application development. Since the amount of data used in the integration test is large, I placed the data for each version in s3 and built it to specify the version of the data in the test code. Since it was a team that did a lot of mob programming and had a relatively constant conversation, I proposed and experimented with the development of a trunk base. Since there was a lot of Shell as part of the process, I adopted ShellSpec as the test framework and I created a sample to build and test with Bazel.
2019-08-26 TO 2020-02-29	<p>Back-end development (3days/week) at Web service company</p> <p>Node, TypeScript, koa, Swagger, CleanArchitecture,DDD, TDD, Jest, Docker, ECS(Fargate), RDS, MySQL, ElastiCache, Redis, GitHub, CircleCI, Linux, Ubuntu, Alpine, Slack, Trello, Scrum, 1on1, KPT,</p> <ul style="list-style-type: none"> Designed system architecture, decided FW, tools. Developed APIs, wrote e2e and unit tests. Controlled developers, tasks, meetings. Built CI/CD environment
2018-11-01 TO 2019-07-31	<p>Infrastructure, Back-end development (3days/week) at AI startup company</p> <p>Docker, Terraform, Ansible, MongoDB, Redis pub/sub, Artillery, TypeScript, Serverless, DynamoDB, GitHub, CircleCI, Linux, Ubuntu, AMI, Slack</p> <ul style="list-style-type: none"> Migrated to Atlas from MongoDB on EC2. Built pub/sub system with ElastiCache. Replaced the application built on ECS with Docker on AWS ECS(Fargate). Built all in Terraform, and prohibited manual work. Done Load testing & Stress testing with Artilley. Automatic checking of container and infrastructure codes is perfomed in CircleCI. Developed a new Slack bot by TypeScript on Lambda for our shuffle lunch.
2018-11-01 TO 2019-03-31	<p>(1days/week) at HR startup company</p> <p>nginx, Ansible, Jupyter Notebook, Terraform, Docker, AWS SageMaker, Python3, Linux, Ubuntu, AMI, GitHub, CircleCI, Slack,</p> <ul style="list-style-type: none"> Built A/B test environment for Machine learning models on AWS SageMaker by Terraform & AWS cli. Provisioned Docker container by ansible. Checking terraform, shells code at CircleCI.
2018-03-01 TO 2019-07-31	<p>Backend development (3days/week) at IoT startup company</p> <p>Python2, Python3, Flask, Flasgger, TypeScript, koa, Swagger, Serverless, Docker, DynamoDB, API Gateway, Lambda, Terraform, Ansible, CircleCI, Linux, Ubuntu, CleanArchitecture, TDD, DDD, GitHub, Slack, JIRA, Confluence</p> <ul style="list-style-type: none"> Redeveloped the server used AWS Lambda & API Gateway to use Serverless FW. The design was simple but redesigned to Clean Architecture because of its high reliance on external service. Introduced type hint of Python, CI/CD, Swagger, Document on CircleCI. Provisioned development machine by Ansible.

2018-03-01 TO 2018-10-31	Backend development (4days/week) at Ads tech startup company Go, Echo, Goswagger, Swagger, MySQL, Aerospike, Redis, Linux, Ubuntu, SchemaSpy, Docker, Ansible, docker-compose, GitHub, CircleCI, Slack, <ul style="list-style-type: none"> Developed back-end server, batch. Redesigned API of admin console with a Clean Architecture. Introduced Swagger, Docker, Ansible. Made ERD from real schema, built automatically in CircleCI job Organized a study session on design, how to use tools, and development flow. 		
2018-02-01 TO 2018-02-28	Backend Developer (1days/week) at Fintech company Kotlin, spark, Linux, Ubuntu, SchemaSpy, Swagger, API Blueprint, Gradle, Docker, AWS ECS Fargate, CircleCI, MySQL, GitHub, Backlog, Slack,		
2017-03-01 TO 2017-09-30	Backend Developer (5days/week, includes remote work: 3days) at Ads creative company TypeScript, Elixir, Node, Express, phoenix, postgres, Linux, Ubuntu, CentOS, GitHub, ZenHub, CircleCI, Swagger, Ansible, mix, Codebeat, Codacy, Docker, AWS, Slack		
2016-11-01 TO 2017-02-28	Full Stack Developer (3days/week) at Ads creative company JavaScript, Node, loopback, Angular, MongoDB, CSS, Linux, Ubuntu, CentOS, GitHub, CircleCI, Swagger, GitHub projects, Slack,		
2016-08-01 TO 2017-02-28	Back End Developer (5days/week) at IoT startup company Scala, sbt, play2, MySQL, Linux, Ubuntu, Ansible, GitHub, CircleCI, Asana, Swagger, Slack,		
2016-06-01 TO 2017-09-30	Android Developer (2days/week, includes remote work a day) at IoT startup company Scala, sbt, Android, Linux, Ubuntu, Bluetooth, BLE, Beacon, CircleCI, GitHub, DaployGate, GitHub issues, Slack,		
2016-04-01 TO 2016-05-31	Android Developer (3days/week, full remote) at IoT startup Java, Gradle, Fabric, Android, Linux, Ubuntu, CircleCI, GitHub, Pivotal Tracker, Bluetooth, Beacon, BLE, Slack,		
2016-04-01 TO 2016-07-31	Back End & Electronic Prototype Developer (4days/week) at IoT startup Javascript, Arduino, 3D CAD, OnShape, Node, Angular, Twitter Bootstrap, MySQL, Linux, Ubuntu, GitHub, Slack,		
2016-03-01 TO 2016-03-31	Back End Developer (3days/week, full remote) at IoT startup JavaScript, HTML, Node, npm, bower, Grunt, MySQL, Linux, Ubuntu, Sequelize, CircleCI, GitHub, Swagger, Slack,		
2016-01-16 TO 2016-02-28	Back End Developer (3days/week, full remote) at IoT startup JavaScript, HTML, Node, express, Swagger, npm, bower, Grunt, pm2, SendGrid, KiiCloud, GCP, sinon, jasmine, mocha, CircleCI, GitHub, GitHub issues, Twitter Bootstrap, Linux, Ubuntu, Slack,		
2015-12-01 TO 2016-03-31	Back End Developer & Electronic Prototype Developer (1days/week) at IoT startup JavaScript, Node, Angular, Twitter Bootstrap, SQLite, npm, bower, grunt, sequelize, Swagger, Ansible, GitHub, Slack,		
EDUCATION	Collage <ul style="list-style-type: none"> <i>Qualified Bornsetter</i> 		
2004-04-01 TO 2007-03-31			
LANGUAGES	Japanese	Native speaker	
	English	Limited Working	
PROFILES	GitHub	Meeting (30min)	LinkedIn
	Twitter		

PERSONAL

IoT, Zapier, IFTTT, Universal Remote Control
Film, Buffing Compounds, Nail