Hiromi Shikata

Full-stack Engineer & Engineering Manager with 19 years of experience

Contact

Website https://hiromishikata.github.io/profile/

Email shikata@uminoseisaku.com

Phone +81-50-5354-8192

Location

City Minato-ku

Region Tokyo

About

I'm a full-stack engineer with extensive experience in startups, having contributed to over 46 projects.

Professional Skills Summary

Backend Development (19+ years): TypeScript, Go, Python, Rust, Clean Architecture, DDD, RESTful API, Swagger, OpenAPI, Serverless, GraphQL, Docker, docker-compose

Database (19+ years): MySQL, Postgres, DynamoDB, MongoDB, Redis, ElasticSearch, OpenSearch

DevOps (17+ years): CI/CD, GitHub Actions, Testing Automation, Documentation Automation, SchemaSpy, CloudMapper, E2E Testing, Unit Testing, Integration Testing, CircleCI

AI Integration (2+ years): OpenAI, Claude, Gemini, langchain, AutoGen, AutoGPT, Autonomous AI

Infrastructure (12+ years): CDK, SST, Docker, docker-compose, AWS ECS Fargate, AWS Lambda, Terraform, Ansible, CloudFormation, Kubernetes

Frontend Development (15+ years): React, TypeScript, shadcn, Tailwind CSS, Storybook, BrowserStack, Vue, Angular, Aurelia

Management (4+ years): Agile, Lean, Trunk-Based Development, Full-Remote team leading, Full-Flexible team leading, git-flow, GitHub Flow, GitHub Issues, GitHub Projects

Dislike: Ruby, PHP, iOS, Scrum, JIRA, GitLab, Slack, Old, legacy environments

Work Experience

2025-02-27 to 2025-03-31

Full-remote work AI engineer at **Division of a leading AI company**

Python3, AutoGen, OpenAI, GPT-4o, Gemini, ReactJs, Gatsby, AutoGenStudio, pytest, GitHub Actions, GitHub Projects, GitHub Issues, Slack, Azure, SQLite,

- This was a project to develop a prototype of AIAgent for a manager of a client company.
- Although it was a short-term project of one month, we were evaluated for the speed of feature development, the preparation of a task management environment for the team, the preparation of a CI/CD environment, task assignment control for young members, and the fact that we considered multiple methods to realize the project even if it was not definitively possible.
- Although the unit price was higher than other engineers, we received feedback that the performance was sufficient and good.
- Since it was a small and short-term project, we only implemented the minimum management and automation in a compact manner. After it was decided to continue the project, we made it clear that the necessary work would be left as tasks and intentionally postponed.
- We divided the test policy into parts that behave undeterminedly in LLM and the tools used by LLM, and made it possible to automatically perform unit tests and integration tests on the tools.
- We built a CI environment with GitHubAction so that various basic problems such as unit tests, API startup, and front-end builds could be automatically detected.
- The client's feedback was starting to become vague, and they seemed unsure about what they should achieve with AIAgent, so we set out specific use cases and scenarios and worked to contribute to the client's workflow.

- For the demo in the final report, we implemented both the AI's behavior when initially using AIAgent, and behavior based on an understanding of various user expectations.
- We considered the scenario for the demo in the final report and how it could be developed in the future based on possible changes for the client, and submitted a draft document.

2025-01-01 to 2025-02-28

Remote Full-stack Engineer & Engineering Manager at **Welfare Product Replacement for Japanese Market Startup**

Golang, OpenAPI, AWS Lambda, AWS RDS Aurora, MySQL, SST, React, TypeScript, playwrite, Jest, shadcn, Clean Architecture, DDD, GitHub Actions, GitHub, GitHub Issues, GitHub Projects, Docker, docker-compose, Slack, Swagger, kubernetes, Flutter, ReactNative

- The system structure had become inflexible and cumbersome, which became a challenge when we decided to fully pursue an opportunity discovered during a pivot.
- The backend, frontend, and infrastructure were all optimized for the previous product, bloating to about 8 times the size needed for required functionality, so we decided to rebuild.
- We were using Kubernetes which had become burdensome both in terms of maintenance effort and cost, so we switched to AWS Lambda.
- The existing system used Flutter, but based on current team members' skill sets and recruitment market research, we decided to use ReactNative.
- As existing members were busy with other projects, we hired new members and got them contributing immediately as effective team members through an automated onboarding process.
- In this team, we recruited excellent members through team referrals rather than my own. Out of over 30 people who helped over nearly 3 years, only 2 members left voluntarily - one to start a business and another to care for family. This suggests team members are satisfied with the current development team environment.
- When some roles became difficult to fill using existing methods and traditional approaches showed signs of stagnation, we first hired assistants to strengthen our engineering recruitment system. This initiative was successful, leading to 4 excellent candidates among over 30 applicants, successfully strengthening the team.
- While there aren't many technical challenges due to the nature of the product, schedules are tight so we continue to minimize non-implementation time to manage this.

2024-05-28 to Present

Remote Full-stack Engineer & Engineering Manager at **AI Product Implementation at Startup**

Golang, Python3, OpenAI API, Anthropic API, Gemini, Selenium, AutoGen, AWS Lambda, DynamoDB, AWS SQS, Postgres, AWS RDS Aurora, SST, React, TypeScript, shadcn, Clean Architecture, DDD, GitHub Actions, GitHub, GitHub Issues, GitHub Projects, Docker, docker-compose, Slack, Jest, Swagger, OpenAPI, RESTful, playwrite

- Progressed from R&D results to product development, moving to full-scale implementation based on business requirements.
- Separated the system architecture into a data and process management system and an AI-focused system to avoid using type-unsafe Python in parts that should be programmatically controlled, ensuring more stable programming.
- Built a serverless architecture using AWS Lambda as the execution environment due to the product requiring simultaneous processing of thousands of requests.
- Implemented custom queue execution to overcome scaling limitations when the number of Lambda functions executable from SQS reached its limit, despite Lambda's individual limits not being an issue.
- Frequently evaluated various models and switched to better performing ones as LLM options increased with monthly improvements in AI performance and cost reductions.
- Built an automated, asynchronous management system to handle increased management overhead as successful recruitment grew the team to around 16 members.
- Established a fully remote, flexible work hours system without regular meetings, implementing goal sharing, priority decision rules, and engineer performance evaluation systems that reduced daily management time to about 2.5 hours. Still working on improvements targeting under 1 hour.
- Introduced a daily evaluation system that revealed previously undetectable issues, enabling problem detection and resolution by the next day despite the fully flexible environment.
- Accurate individual engineer performance evaluation enabled quantitative assessment for rate adjustments (increases or decreases) and team composition decisions. I feel resulting in an estimated doubling of team cost efficiency.

2024-04-01 TO 2024-05-27

Remote Full-stack Engineer & Engineering Manager at **AI Product R&D Startup**

Python3, OpenAI API, Anthropic API, Llama, ChromeExtension, Selenium, PyAutoGUI, AutoGen

- Implemented internal tools leveraging improved LLM (Large Language Models) accuracy. Built autonomous systems aimed at executing tasks typically performed by new employees. Initially, when tasks exceeded 3-4 steps, LLMs would forget necessary operations or perform unintended actions. Resolved this by optimizing LLM specifications and task structures. Currently achieving over 90% success rate for tested tasks.
- Improved LLM task execution stability by breaking down tasks into smaller components and providing examples, after identifying common error patterns based on task characteristics.
- Modified the system to work with APIs from OpenAI and two other major AI companies simultaneously. Structured certain tasks to utilize more cost-effective or faster LLMs. Investigated and implemented multi-agent frameworks to handle complex tasks with autonomous correction capabilities. The multi-AI agent architecture enables autonomous behavior for complex scenarios, resulting in more versatile and reliable solutions.
- Led a 4-person team, initially struggling with task distribution. The project started with limited AI expertise across the team, requiring extensive research while under pressure to deliver quick results. Implemented agile systems for the remote team, accommodating members across different countries and time zones.
- Enhanced tool demonstrations for potential clients and investors by adding intuitive displays showing AI thought processes and actions, addressing initial uncanny behavior. Implemented configurable settings to toggle explanatory animations and displays to optimize performance.
- Explored OS-level computer control functionality, including investigating the ability to interpret visual information for task operations. Various approaches were tested, but single LLM solutions proved inadequate for visual instruction interpretation. Multimodal approaches showed better results but still struggled with pixel-precise click location accuracy. Combined with existing research, showing only 20% accuracy, led to exploring alternative approaches.
- Currently investigating methods similar to retail product and origin recognition systems observed 5-6 years ago, collaborating with domain experts. While browser control constitutes most tasks, development focus remains on web interface manipulation due to current instability in image-based instruction.
- Experimented with an OSS Chrome extension capable of handling 3-step tasks, but discontinued development due to high integration costs with existing successful implementations.
- Implemented multiple strategies to reduce token count while preserving essential information, addressing prompt token limits.
- Initially achieved 60-70% success rate with source code generation for automated operations. Improved to nearly 100% by implementing AI-friendly wrapper functions with programmatic internal processing.
- Dedicated time to testing new LLMs, frameworks, and products in a rapidly evolving landscape. Team development led to peak costs of ¥10,000 per hour, prompting investigation into open-source LLMs and newer options like Grok for cost and speed optimization.
- Currently working on restructuring tasks for smaller LLMs, as simple replacement of expensive/slow LLMs failed. Found success with further task decomposition, planning structural modifications for the next phase to enable finer task granularity.

2022-01-01 TO 2023-08-31

Technical Advisor at **Manufacturing Industry SaaS Startup**

Team Development Management, Recruitment Support, Technical Debt Resolution, Organizational Structure Improvement

- Provided support to a newly appointed leader who was struggling with direction after the previous manager's departure.
- Started from a challenging position where even brief 10-minute conversations revealed numerous organizational and technical issues cascading continuously, indicating high stress levels.
- Addressed the leader's scattered thinking by helping systematically identify and evaluate all challenges regardless of size or category, prioritizing them by importance and urgency. After this exercise, we focused on the most critical issues to create a clear path forward.
- Conducted regular online meetings to assess progress and plan next steps. My role was primarily asking questions, allowing the leader to do all the analysis and decision-making. After about three meetings, the leader gained confidence and momentum, making this support structure unnecessary.
- As immediate issues were being resolved, we noticed emerging anxiety about long-term direction. Started an initiative to identify what information would help clarify future actions. Also considered what information would help the CEO's decision-making if needed. After adjusting communication between the leader and CEO, within about 2 months, the annual planning became clearer and the leader's uncertainty noticeably decreased.
- Once long-term direction was established, the next challenge was task volume exceeding capacity. Recruitment was ongoing but applications were low, causing about a month's delay in planned team expansion. Developed plans to broaden recruitment efforts and secure internal resources. Implemented more proactive recruitment strategies beyond just free job boards, successfully securing multiple quality team members.
- Addressed the issue of the leader lacking time for strategic work due to being overwhelmed with immediate tasks and unable to address long-term challenges.
- Analysis of the leader's time allocation revealed excessive manual work on tasks that could be automated, and time spent preparing work to make it easier for team members. As team members developed, we expanded their responsibilities and communication channels, enabling the leader to dedicate 50% of time to future initiatives.
- As the team grew, self-management became a challenge. Included management team members in meetings to discuss and implement structural improvements. This resulted in stable team control under managers.
- Modified meeting structure to include technically-oriented members to help address technical debt that the leader had been handling alone since joining.
- Aimed for team self-direction without top-down leadership by starting with the same process of issue identification and evaluation we initially did with the leader. Through discussions, team members and leader aligned on values and problem recognition, resolving initial challenges within 3 weeks while broadening team perspective. Currently continuing discussions with the same members for further improvements.

2023-02-01 TO 2023-08-31

Remote Full-stack Engineer (3 days/week, 2-person development team) at **Early Childhood Programming Education Product Development at US Subsidiary of Educational Startup**

AWS lambda, CDK, Node.js, React, AWS app sync, App sync simulator, DynamoDB, Dynamo local, Dynamo Admin, Postgress, Graphql, Ubuntu, TypeScript, Express, DDD, Clean Architecture, GitHub Actions, GitHub, GitHub Issues, JIRA, confulence, GitHub Projects, Trunk-based Development, Mermaid, Docker, docker-compose, Slack, Jest

- Developed a product to promote sales during the dashboard development phase.
- Reduced team size to just 2 people to meet tight release deadlines.
- Applied DDD strictly and invested time in normalization due to extensive functional requirements. Generated almost all code from strict definitions to meet release deadlines. Auto-generated code included tests, helping prevent defects.
- Switched GraphQL definitions from manual writing to model-based auto-generation, and released the generation tool as open source.
- Minimized team size to reduce communication overhead, securing over 80% of time for coding.
- Managed 45 entity models and produced 110,000 lines of code (excluding published libraries and comments) in 4 months with just 2 people, of which 45,000 lines were auto-generated using our custom Generator.
- Contributed to business expansion as the product was later also sold by the Japanese parent company.

2022-11-01 TO 2023-01-31

Remote Full-stack Engineer (3 days/week, 3-person development team) at **Middle School Teacher Dashboard Development at US Subsidiary of Educational Startup**

AWS lambda, CDK, Node.js, React, AWS app sync, App sync simulator, DynamoDB, Dynamo local, Dynamo Admin, Postgress, Graphql, Ubuntu, TypeScript, Express, DDD, Clean Architecture, GitHub Actions, GitHub, GitHub Issues, JIRA, confulence, GitHub Projects, Trunk-based Development, Mermaid, Docker, docker-compose, Slack, Jest

- Joined an agile development team at a US subsidiary of a Japanese startup, formed after dissolving an 8-person offshore development team.
- Applied DDD strictly and invested time in normalization due to extensive functional requirements. Generated almost all code from strict definitions to meet release deadlines. Auto-generated code included tests, helping prevent defects.
- Switched GraphQL definitions from manual writing to model-based auto-generation, and released the generation tool as open source.
- Minimized team size to reduce communication overhead, securing over 80% of time for coding.
- Relocated to Mexico to align with US time zones, establishing rapid incident response capabilities.
- Established communication protocols enabling curriculum team assistants to autonomously handle progress tracking and CS department issue reports.
- Continuously improved meeting processes through visualization and advance text sharing to address communication challenges in first-time collaboration with native English speakers.

2022-06-01 to Present

Remote Full-stack Engineer (3 days/week, Initial development team of 6 engineers) at **Consumer Foodtech Startup**

Scrapy, Spider, Playwright, kubenetes, AWS lambda, AWS Batch, Ubuntu, TypeScript, Dart, Express, OpenAPI, Flutter, Mermeid, Puppetier, gRPC, Golang, Python 3, RedShift, firestore, DDD, Clean Architecture, Microservice Architecture, Monolithic Repository, firebase, GitHub Actions CircleCI, GitHub, GitHub Issues, GitHub Pages, GitHub Projects, Trunk-based Development, Mermaid, OpenSeatch, OpenSeatch Dashboard, Docker, docker-compose, Slack, Jest

- Joined a startup with an existing system and team structure.
- Brought in to help replace an underperforming outsourced team.
- Led new feature development while coordinating system replacement with newly joined team members.
- Built OpenSearch environment from data ingestion to query optimization and tuning for search functionality.
- Developed RPA solutions for external service integration, automating manual operations.
- Implemented a divide-and-conquer architecture anticipating numerous integration points, creating a scalable codebase that maintains speed.
- Performed code reviews for Flutter frontend development as the only experienced Flutter developer.
- Improved meeting structure and communication flows to address challenges in information sharing and problem identification within a fully remote team with many junior members during organizational changes.
- Established unit testing practices from scratch, creating a foundation that encouraged other team members to write tests.
- Addressed credential management issues by updating credentials tied to departed engineers and implementing founder-level credentials to prevent future disruptions during team changes.
- Implemented CI/CD with CircleCI and GitHub Actions, automating most manual processes. Team members became proficient and began exploring additional CI tools independently.
- Created a CircleCI orb for managing monorepo CI files using divide-and-conquer approach. Released as open source after seeing similar needs in other projects.
- Established trunk-based development with multiple daily deployments through appropriate team structure and codebase organization.
- Adopted text-based documentation formats (OpenAPI, Mermaid) with rich ecosystems to prevent documentation-code divergence. Hosted on CircleCI with proper access control for relevant team members.
- Built an auto-updating documentation system using GitHub Pages redirects to handle changing CircleCI artifact URLs, ensuring documentation links remain valid.
- Provided guidance and support to junior members struggling with implementation.
- Made proposals regarding cost, operations, and structure during fundraising challenges.
- Built an efficient multi-timezone remote team by hiring skilled global talent with strong communication abilities.
- Restructured infrastructure to reduce increasing proxy costs.
- Gradually expanded team member responsibilities through incremental cross-role interactions, building toward self-sufficiency.
- Established normalized information sharing routines for asynchronous remote work, ensuring clear focus areas, task priorities, and status updates without requiring constant real-time communication.
- Created a roadmap for breaking down a monolithic service that was limiting agility, balancing user value delivery with business initiatives and refactoring. Currently in progress, maintaining flexibility for weekly changes common in startup environments.

2022-03-01 to 2022-07-31

Project Manager (5-person development team) at **E-commerce Site Full-scratch Development**

FlutterFlow, Firebase, GitHub, Lark, CircleCI, GitHub Actions

2021-02-01 to 2021-08-31

Remote Android App Engineer (1-3 days/week) at **Retail Store Advertising Product**

Android, Kotlin, GitHub, GitHub Actions, Testing Automation, GitHub Projects, Confluence, Slack, Mockito

- Provided technical support from Android application design through implementation.
- Established unit testing using Mockito and built CI/CD environment using GitHub Actions, as no tests existed previously.

2022-02-01 to Present

Technical Advisor at **Consumer App Startup**

Flutter, Dart, AWS, MySQL, Docker, docker-compose, GitHub, Slack, Terraform Cloud, JIRA, Confluence

2021-12-01 to 2022-03-31

Technical Verification and Project Management at **Research Results SaaS Conversion**

MATLAB, Python, GitHub, CircleCI

2020-12-01 to 2021-03-31

Remote Full-stack Engineer & Project Manager (0.5 days/week, 2-3 Engineers, 1 Assistant/QA) at **Cloud-based Hardware Image Analysis MVP Development**

Ubuntu, TypeScript, Dart, Express, FlutterFlow, Flutter, Mermaid, no-code, firestore, DDD, Clean Architecture, Microservice Architecture, Monolithic Repository, firebase, fireauth, firebase functions, GitHub Actions, CircleCI, GitHub, GitHub Issues, GitHub Projects, github-flow, Slack, Jest, Trunk-Based Development

- Project focused on making an analysis system accessible through a browser.
- Built a structure to minimize effort and cost while maximizing iteration speed and frequency.
- Chose no-code solution capable of outputting Flutter code, enabling business side members, assistants, and designers to directly update screens.
- Implemented automated CI checks to ensure safety for screen changes made without engineering oversight.
- Hired an engineer strong in architecture design and backend development, as being the sole technical resource would limit capacity and pose continuity risks.
- Implemented assistant service for recruitment to ensure high-quality hires from a large candidate pool.
- Achieved 40% reduction in running costs by hiring overseas engineers with equivalent skills.
- Established rules and translation environment to support fully remote global team.
- Conducted domain discussions and design with business side members, creating abstractions aligned with company expansion direction, minimizing communication confusion.
- Built on Firebase to leverage FlutterFlow compatibility, aiming for fastest possible completion.
- Successfully implemented small start approach as business side members correctly understood MVP definition.
- Designed express as abstraction layer to avoid Firebase function dependency, while enabling OpenAPI generation.

2020-03-01 TO 2022-02-28

Remote Full-stack Engineer (2 days/week, Team growth from 3 to 35 engineers) at **Urban Development Startup Support**

Ubuntu, TypeScript, Dart, Express, Flutter, firestore, Postgres, Google CloudSQL, Clean Architecture, Microservice Architecture, Monolithic Repository, Feature Flags, firebase, fireauth, firebase functions, firebase config, Prisma 2, OpenAPI, Docker, Docker Compose, CloudSQL, Google App Engine, CircleCI, GitHub, GitHub Issues, GitHub Projects, github-flow, Zapier, Slack, Jest, Jasmin, Trunk-Based Development

- Helped establish CI/CD environment for an existing mobile application that lacked one.
- Set up automated deployment and integration testing infrastructure.
- Prioritized integration testing over unit testing due to urgent quality assurance needs and requirement for testable code changes.
- Implemented task management tools to organize scattered information and streamline communication, as team members worked across different times and locations using Slack.
- Introduced Prisma as it was the most type-safe ORM available for TypeScript at the time.
- Migrated to monorepo structure for its benefits. Optimized CI costs by implementing differential job execution.
- Built international team focused on overseas engineers to address implementation challenges when part-time members couldn't keep up with business requirements.
- Implemented online assistant service for recruitment to efficiently screen large candidate pools and reduce internal workload, while also handling non-development tasks.
- Transitioned to microservices architecture to support company's expansion plans and facilitate adoption of new technologies.
- Created templates for new services to accelerate microservices development. While this enabled rapid launches with quality assurance through automated testing and deployment, it somewhat limited adoption of new technologies.
- Established rules for real-time meetings and timezone management, along with 1on1s and scrum events, to accommodate team members across multiple nationalities and locations.
- Integrated assistant team into development operations as communication hub, handling manual testing, onboarding, tools, and procedures.
- Hired experienced Scrum PMs to lead smaller teams when alignment and delivery commitment began declining with team growth.
- Increased PM roles to address issues with unclear specifications, business challenges, and project visibility, creating a self-sustaining structure.
- Modified 1on1 structure to include both PM and myself, with PMs handling immediate project/engineer concerns while I focused on long-term growth, interests, and skill development.
- Received recognition for successfully hiring engineers superior to average Japanese engineers within one month, reducing operational costs to 1/3 compared to local hiring, and aligning system design with business philosophy.

2020-02-01 TO 2020-12-31

Remote Full-stack Engineer (4 days/week, 8-person development team) at **Major Electronics Manufacturer Data Lake Platform Construction**

AWS CloudFormation, AWS Redshift, AWS Batch, AWS CodePipeline, AWS CodeBuild, AWS CodeCommit, ShellSpec, Bazel, Docker, BitBucket, JIRA, Slack, Bats, Ubuntu, SQL, Trunk-Based Development

- Worked on improving an existing data lake platform.
- Automated previously manual deployment and testing processes.
- Implemented automation using AWS CodeBuild, CodePipeline, and CodeCommit, avoiding Atlassian tools in the AWS environment.
- Found the project interesting due to unique deployment and testing considerations specific to data lake environments.
- Built integration testing infrastructure with versioned data in S3, allowing test code to specify data versions due to large data volumes.
- Handled large-scale data integration and analysis from multiple core systems including product sensor data, accounting systems, and inventory management systems.
- Proposed and experimented with trunk-based development in a team that frequently practiced mob programming and maintained constant communication.
- Created sample build and test execution using ShellSpec framework and Bazel for extensive shell script processing.
- Prioritized compatibility with team's existing technology stack when making technical decisions, ensuring smooth transitions from current technologies.

2019-09-01 TO 2020-01-31

Remote Full-stack Engineer (3 days/week, 4-6 person development team) at **New Service Development and Construction**

Linux, Ubuntu, Alpine, TypeScript, Node.js, koa, TypeORM, MySQL5.6, Redis, GitHub, CircleCI, Slack, Docker, docker-compose, OpenAPI, AWS ECS Fargate, AWS ElastiCashe, AWS RDS, GitHub projects

- Led technology selection process (languages, frameworks, middleware, libraries, tools, etc.)
- Handled backend design, implementation, testing, code review, and DevOps
- Implemented RESTful interface design, Clean Architecture for application structure, and followed GitLab flow, DDD, TDD for development processes
- Advanced team building and improvement initiatives (1on1s, KPT retrospectives, study sessions)
- Managed offshore team coordination (daily check-in meetings, design sharing, task control)
- Also attempted business layer coordination, though this aspect was less successful

2018-11-01 TO 2019-07-31

Remote Infrastructure Engineer (3 days/week, 3 Application Engineers, ~20 Employees) at **AI Startup Infrastructure Environment Construction**

Linux, Ubuntu, AMI, HCL, MongoDB, Redis, GitHub, CircleCI, Slack, Docker, Docker Compose, Ansible, Terraform

- Improved infrastructure environment for a rapidly growing system initially built by a single application engineer.
- Migrated MongoDB to Atlas for database redundancy.
- Built new cache server using ElasticCache.
- Designed application servers for scalability.
- Specifically, replaced EC2-based applications with Docker and migrated to AWS ECS (Fargate).
- Built all new environments using Terraform, prohibiting manual operations to ensure environments could be safely destroyed and recreated at any time.
- While Serverless development wasn't possible in this environment, aimed to match Serverless framework user experience.
- Given dedicated infrastructure repository, implemented automated container checks and infrastructure code testing in CircleCI to reduce manual effort while ensuring reproducibility and safety during changes.
- Developed new Slack bot when unable to find suitable existing solution for corporate needs, which was well-received.

2018-09-01 TO 2018-12-31

Remote Infrastructure Engineer (1 day/week, 3-person development team) at **AI Startup Embedded Environment Construction**

Linux, Ubuntu, Python3, GitHub, CircleCI, Slack, Docker, Ansible, AWS SageMaker, Jupyter Notebook, Terraform

- Requested to create A/B testing environment to improve accuracy of machine learning engine and its integration components.
- Implemented AWS SageMaker based on client request and its suitability for future availability requirements.
- Built overall infrastructure using Terraform, with AWS client handling SageMaker components not yet supported by Terraform.
- Created development environment supporting both Ansible and Docker by building Docker images with Ansible.
- While focused on infrastructure rather than application development, implemented CI/CD to automatically trigger model training upon Git push of AI engine code.

2018-03-01 TO 2019-07-31

Remote Server-side Engineer (3 days/week, 6-person team with 1 server-side engineer) at **IoT Company Backend Development**

Linux, Ubuntu, Python2, Python3, Flask, Flasgger, DynamoDB, GitHub, CircleCI, Slack, Docker, Serverless, Ansible, Swagger

- Restructured Python-based AWS Lambda API server for improved manageability.
- Introduced Serverless Framework to improve maintainability of existing backend built with Lambda and API Gateway without framework, which was using APIs for automation.
- Implemented CI/CD with automated testing and deployment, as the simple design had become difficult to test due to heavy external service dependencies and complete lack of tests.
- Achieved 100% coverage in domain layer unit tests after adopting Clean Architecture for new design.
- Enhanced documentation by integrating framework with Swagger and hosting in CI. Created HTML/JS renderings of sequence diagrams and other materials difficult to express in markdown, ensuring easy access and updates for all team members.
- Migrated from Python 2 to Python 3, introducing type hints and mypy analysis to prevent bugs.
- Automated local development environment setup for Ubuntu and Mac using Ansible.

2018-03-01 TO 2018-10-31

Remote Backend Engineer (4 days/week, 25-person development team) at **Advertising Delivery Service Development**

Linux, Ubuntu, Golang, Echo, Goswagger, MySQL, Aerospike, Redis, GitHub, CircleCI, Slack, Docker, SchemaSpy, Ansible, DockrCompose

- Built advertising delivery service and management interface using Golang.
- Joined project with existing codebase foundation.
- Redesigned management interface API service using Clean Architecture in later phase, making it more testable and easier to modify.
- Set up and introduced Swagger and Docker development environments which were previously lacking.
- Organized skill improvement events during lunch breaks and after work hours, as team had many young members needing knowledge enhancement in design, development tools, and workflow processes.
- Created and shared Ansible playbooks for Ubuntu and Mac development environment setup, as no documentation or scripts existed previously.

2018-02-01 TO 2018-02-28

Remote Backend Engineer (4 days/week, 2-person development team) at **New Service User Web Interface Development**

Linux, Ubuntu, Alpine, Kotlin, spark, exporsed, MySQL, GitHub, Backlog, CircleCI, Slack, Swagger, API Blueprint, Gradle, Docker, AWS, SchemaSpy

- Built API server using Kotlin.
- Assisted with ongoing development by implementing CI and setting up environments, as project lacked CI/CD infrastructure and had uncertain API/DB designs.
- Created database documentation through reverse engineering and established CI-based system for continuous documentation updates, addressing communication issues between teams.
- Conducted JavaScript-related investigation and troubleshooting due to shortage of frontend developers.
- Internal development team was dissolved as project was outsourced due to risk assessment.

2017-03-01 TO 2017-09-30

Full-stack Engineer (5 days/week, 3 days remote, 6-person development team with 2 server-side engineers) at **Advertising Creative Production Venture Company System Renewal**

Linux, Ubuntu, CentOS, TypeScript, Elixir, nodejs, express, phoenix, Postgres, GitHub, ZenHub, CircleCI, Slack, Swagger, ansible, mix, TravisCI, Codebeat, Codacy, Docker, Docker Compose, AWS EC2

- Handled entire process from requirements definition through language selection, design, development, and testing.
- Renewed existing system.
- Rebuilt part of existing system as separate system with revised functionality.
- Later tasked with rebuilding main system as well, handling language selection for one system while assisting with development of the other.

2016-11-01 TO 2017-02-28

Remote Full-stack Engineer (3 days/week, 4-person development team with 3 developers) at **Advertising Creative Production Venture Company System Development and Operation Support**

Linux, Ubuntu, CentOS, JavaScript, CSS, Node.js, LoopBack, AngularJS, MongoDB, GitHub, CircleCI, Slack, Swagger, GitHubProject

- Handled development and testing phases.
- Developed additional features and provided operational support for advertising production support system.

2016-08-01 TO 2017-02-28

Remote Server-side Engineer (2-5 days/week, 4-person development team with 1 developer) at **IoT Startup Server-side Application Development**

Linux (Ubuntu), Scala, sbt, play2, MySQL, Ansible, GitHub, CircleCI, Slack, Asana, Swagger

- Started building new system separate from existing demo system when mass production became feasible, following the joining of a member with hardware mass production experience in April.
- Built highly decoupled, clearly structured system with functional separation to support multiple approaches toward final goals and future expansion.
- Achieved microservices-like architecture, though not initially planned as such.
- Handled requirements definition, language selection, design, development, and testing phases.

2016-06-01 TO 2017-09-30

Remote Android App Engineer (2 days/week, 1 day remote, ~20 team members with 3 Android developers) at **IoT Device Integration Android App Operation**

Linux (Ubuntu), Scala, sbt, Android, CircleCI, GiHub, DeployGate, GitHubIssue, Slack, Beacon, Bluetooth, BLE

- Updated design, added features, and fixed bugs for app integrating with voice communication devices using Bluetooth Classic and BLE.
- Found project interesting with many challenging aspects involving Bluetooth, Scala, and audio processing.
- First experience with Scala Android development, providing many learning opportunities.
- Took extra care with documentation, comments, and Slack communication to avoid misunderstandings, as team included many non-Japanese members requiring natural English communication.
- Handled investigation, development, and testing phases.

2016-04-02 TO 2016-05-31

Remote Android App Engineer (3 days/week, 3-person team with 1 Android developer) at **IoT Beacon Device Integration Android App Platform Change**

Linux (Ubuntu), Java, Gradle, Fabric, Android, CircleCI, GiHub, PivotalTracker, Slack, Beacon

- Modified existing Android app to accommodate beacon device platform changes.
- Updated BLE Service, Characteristic, and other components to match new hardware specifications.

2016-4-01 TO 2016-07-31

Engineer (4-5 days/week, 3-person team) at **IoT Startup Demo Server-side Application and Hardware Development**

Linux (Ubuntu), javascript, Arduino, 3DCAD, nodejs, AngularJS, TwitterBootstrap, SQLite, MySQL, npm, bower, Grunt, Sequelize, Ansible, GitHub, Slack, Asana, Swagger

- Developed web server application and hardware prototypes for IoT startup product.
- Joined as full-time employee after helping with the service part-time since December 2015.
- Led hardware prototype development using 3D printing, machining, and metal fabrication.
- Handled software side including demo system design, database & API server, and management interface development.
- Conducted technical research for service and specification consolidation, prioritized requirements, and developed verification hardware for mass production.
- Established project management and documentation processes while building development framework.
- Handled investigation, design, development, and testing phases.

2016-03-04 TO 2016-03-31

Remote Server-side Engineer (3 days/week, 5-person development team with 1 server-side engineer) at **Sensor Gadget Log Collection and Visualization Service**

Linux (Ubuntu), javascript, html, nodejs, npm, bower, Grunt, MySQL, Sequelize, CircleCI, GiHub, Swagger, Slack

- Developed a service to collect, store, and visualize sensor data using client's gadgets.
- As this was intended for sales demonstrations of a new client service, prioritized building demo-ready features quickly, skipping non-essential components like authentication.
- Responsible for database design, API server implementation, and customizing the gadget SDK to send data to our server.
- Given time constraints and team composition including designers and frontend engineers, focused on loose coupling between client and server sides through API-only integration.
- Implemented efficient data aggregation methods and timing to meet summary display requirements while minimizing operational overhead.
- Provided Swagger API specifications from the early stages to accelerate client-side development.
- Responsible phases included research, design, development, and testing.

2016-01-16 TO 2016-02-28

Remote Server-side Engineer (3 days/week, ~3 team members, 1 server-side engineer) at **Authentication Service Development for Data Visualization Smart Gadget**

Linux (Ubuntu), javascript, html, nodejs, npm, bower, Grunt, mp2, SendGrid, KiiCloud, GoogleCloudPlatform, CircleCI, sinon, Jasmine, Mocha, CircleCI, GiHub, GitHub Issues, Slack, TwitterBootstrap

- Built user registration functionality and interface, login API for apps, and user information update/validation components.
- Large-scale client project involving multiple companies collaborating on a single system.
- Multiple companies worked on specific parts of the overall system, with communication routed through the prime contractor. Given limited development time and difficulty understanding details outside our scope, proceeded cautiously to avoid rework.
- Proactively proposed API specifications and provided early working environments to partners to identify any misalignments early.
- Created executable API documentation using Swagger to make APIs easily testable and clearly understood, preventing development bottlenecks. Swagger also improved development speed and simplified verification, helping reduce overall work hours.
- Storage, environment, and architecture were pre-specified. The designated BaaS was new to me, requiring extra time for research, behavior verification, and appropriate design considerations, making development take longer than usual.
- Despite tight scheduling, met deadlines and received positive client feedback.
- Responsible phases included research, design, development, and testing.

2015-12-01 to 2016-03-31

Full-stack Engineer (1-3 days/week, ~3 team members) at **IoT Startup Server-side Application and Hardware Development**

Linux (Ubuntu), javascript, javascript, nodejs, AngularJS, TwitterBootstrap, SQLite, npm, bower, Grunt, Sequelize, Ansible, GitHub, Slack, Swagger

- Developed web server application and hardware prototypes for IoT startup product.
- In early stages, assisted with consultation, design, and development of APIs and sites/apps for service demonstration prototypes.
- Responsible phases included research, design, development, and testing.

2015-12-02 to 2016-01-29

Remote Android App Engineer (3 days/week, ~4 team members, 1 Android developer) at **Social Event Service Android App Development**

Linux (Ubuntu), Java, Gradle, Android annotations, Android Spring REST client, CircleCI, Git, DeployGate, ChatWork

- Developed Android app for a new service focused on planning events and recruiting participants.
- Planning and design were completed, with database, API server, and iOS app being developed in parallel. Integrated APIs into the Android app as they became available.
- App structure remained simple as it primarily focused on data display.
- As this was a remote project and the client lacked Android expertise, provided daily progress updates through Chatwork including written reports and work forecasts, beyond just source code pushes.
- Implemented CircleCI and automated DeployGate updates as continuous integration wasn't initially planned, ensuring client could always monitor app status.
- Responsible phases included design, development, and testing.

2015-11-06 to 2016-01-15

Remote Full-stack Engineer (3 days/week, ~3 team members, 1 developer) at **Service Integration Site Prototype Development**

Linux (Ubuntu), javascript, html, nodejs, AngularJS, mysql, npm, bower, Grunt, CircleCI, GiHub, GitHub Issues, Slack, Sequelize

- Developed prototype for integrating existing services based on user-specified conditions.
- Built as a prototype to test functionality before full implementation of client's planned in-house service.
- Involved from early stages in conceptualizing features and necessary screens.
- Prioritized rapid development methods for quick functionality testing, though these methods weren't intended for production use.
- Responsible phases included research, design, development, and testing.

2015-10-26 to 2015-11-05

Remote Android App Engineer (3 days/week, ~3 team members, 1 Android developer) at **Data Collection IoT Gadget Integration Android App Development**

Linux (Ubuntu), Java, Gradle, Android annotations, Android Spring REST client, CircleCI, GitHub, Deploy Gate, GitHub Issues, Slack

- Developed Android app for remotely collecting and displaying data from IoT gadgets.
- Took over project from part-time engineer who couldn't meet planned demo deadline due to time constraints.
- iOS app development was already in progress with clear specifications to match, preventing time loss and misunderstandings during requirement confirmation.
- With only about 10 days available and lacking environment for rapid development with libraries and frameworks, created new project and migrated existing functionality.
- Despite tight timeline, met deadline and received positive client feedback.
- Assisted with investigation of iOS app gadget integration issues using information gathered during Android development.
- Responsible phases included research, design, development, and testing.

2015-09-14 TO 2015-10-25

Remote Android App Engineer (3 days/week, ~4 team members, 1 Android developer) at **IoT Product Integration Android App BLE Technical Verification Development**

Linux (Ubuntu), Java, Gradle, Android annotations, Android Spring REST client, Beacon, CircleCI, Git, PivotalTracker, Slack

- Conducted technical verification of Android app for Beacon device firmware determination.
- Created prototype app testing both Android 4 and 5 as OS requirement was Android 4.0+.
- Being my first Beacon app development, started with thorough research. Learned extensively about signal content, reading/writing operations, and notification systems.
- As app needed to locate gadget position, required constant signal reception. Verified background operation and process restart scenarios when OS terminated the process.
- Despite time constraints, met deadline and received positive client feedback.
- Assisted with investigation of iOS app gadget integration issues using information gathered during Android development.
- Responsible phases included research, design, development, and testing.

2015-05-01 TO 2015-08-31

Full-stack Engineer (3 days/week, 1-person team) at **Hardware Startup Product Development Support**

Linux (Ubuntu12), arduino, C++,

- Developed product prototype using microcontroller, touch panel, and Bluetooth.
- Grateful for opportunity to work on hardware development despite limited experience in the field.
- Project involved many unfamiliar technologies, requiring research from fundamentals.
- Though not directly used, learned about CAD, 3D printing, and machining tools, expanding technical options for product development.
- Responsible phases included microcontroller program design/development and circuit design/development.

2015-01 TO 2015-10

Full-stack Engineer (2-4 days/week, ~13 team members, 7 developers, 1 server-side) at **Hardware Startup Company Development Support**

Linux (Ubuntu12, Ubuntu14), arduino, C++, javascript, nodejs, AngularJS, TwitterBootstrap3, MongoDB, Jenkins, Git, Ansible, Yeoman, Apache2, Julius, OpenCV, Slack, Redmine, Swagger, elasticsearch, fluentd, kibana

- Developed new IoT product.
- Valuable learning experience working with hardware and electronic circuits despite no prior experience.
- As the team member with longest attendance hours and development project experience, involved in basic design, engineer task management, and schedule coordination.
- Faced challenges in schedule management, task allocation, information sharing, and development process establishment due to team composition: using unfamiliar technologies (various sensors, voice recognition, image recognition), mostly part-time engineers including many students unfamiliar with team development tools and practices.
- Focused on creating easily separable designs and maintaining flexible development structure to handle frequent changes with part-time members, while emphasizing team skill improvement.
- Responsible phases included server-side design/development, hardware circuit design/development, and enclosure software design/development.

2014-09-01 TO 2015-04-30

Full-stack Engineer (4-person team, 1 developer) at **Business System Development**

Windows, Linux, Scala, Java, Javascript, Gradle, Skinny (Scala), Flyway, Jetty, Tomcat, Android annotations, Android Spring REST client, TwitterBootstrap3, AngularJS, jQuery, Robolectric, Postgres9.4, Jenkins, Git, Redmine

- Developed complete prototype business system from server-side through client and Android app for browser and Android use.
- Grateful for opportunity to work with all new technologies: Scala for server-side, AndroidAnnotations for Android app, and AngularJS for JavaScript framework.
- Participated in user meetings and specification discussions, proposing additional features based on clear understanding of requirements and intentions.
- With small team and fluid specifications, focused on rapid development to quickly demonstrate usage scenarios.
- Database contained 27 tables.
- Responsible phases included design, development, and testing.

2014-04-01 TO 2014-08-31

Native App Engineer (5-person development team) at **SDK Operations Team**

Windows, Linux, Java, Gradle, C++, Cocos2dx, SQLite, Jenkins, Nexus, Git, SVN

- Became responsible for SDKs used by internal Android apps for authentication and billing.
- With multiple SDKs to manage, used Jenkins to improve operational safety through automated releases and static analysis for quality assurance.
- Implemented Gradle build system for SDK build and release processes in response to team request.
- Created custom Jenkins plugins when existing plugins couldn't meet functionality requirements or had poor performance when forcibly integrated.
- Developed Cocos2dx plugin for SDK (Android-side and common components).
- Responsible phases included design, development, and unit testing.

2013-09-27 TO 2014-03-31

Native App Engineer (5-person development team) at **App Operations Team**

Windows, Java, Shell(a little bit), Groovy(a little bit), SQLite, Git, Jenkins

- Worked on a long-running app branded with client's main business name that couldn't be discontinued.
- Due to app's history and code readability issues, several months were allocated for refactoring.
- Developed an image selection screen as a reusable library.
- Having primarily worked on server-side, faced challenges with client-side development including resource optimization, event handling complexity, case variations, device-dependent behaviors, and UI-specific controls.
- Given time for quality maintenance, implemented Jenkins on team server for static analysis and build automation.
- Responsible phases included design, development, and unit testing.

2012-12-06 TO 2013-09-26

Engineer (14-person development team) at **Gaming Division Core System**

Linux, Java, FTL, HTML, CSS, JavaScript, shell, Spring, MySQL, SVN, Git, Apache, Tomcat, Memcached, Jenkins, Android

- Developed and operated system used by multiple games.
- System required extensive API usage due to authentication being handled by core platform and each game providing APIs for our use.
- Encountered unique challenges with high API usage, such as performance issues and traffic from large games impacting API servers of newly released smaller games. Addressed by adjusting processing frequency to prevent concentrated server access and consulting with metrics team about caching allowances.
- Given critical integration role and potential for friction, maintained close communication with game teams and proactively investigated issues. Learned various approaches through studying different server architectures and frameworks used by each game.
- Gained valuable experience with high-traffic site architecture patterns.
- Infrastructure: ~15 application servers, 3 database servers (1 master, 2 slaves) at the time
- System scope: ~20 screens, ~50 tables
- Also handled WebView app conversion (Android) for browser games. With 6 target games and more expected, implemented all non-game-specific processing (like metrics) as framework components, keeping game-specific logic in minimal thin classes.
- Responsible phases included design, development, and testing.

2012-08-16 TO 2012-12-05

Engineer (20-person development team) at **Mobile Browser Game Operations**

Linux, Java, FTL, HTML, CSS, JavaScript, shell, Spring, Seaser, MySQL, Apache, Tomcat

- Operated browser games for feature phones and smartphones.
- Managed approximately 40 servers to handle high traffic volume.
- Gained valuable experience with architecture patterns specific to high-traffic websites.
- Responsible for feature development (from per-feature DB design through unit testing) and customer support investigation.

2012-05-01 TO 2012-08-15

Engineer at **Custom Service Development**

Linux, Java, JSP, HTML, CSS, JavaScript, shell, Struts, MySQL, Apache, Tomcat

- Developed site construction system
- System scope: 83 screens
- Database: 39 tables
- Responsible for all project phases

2011-10-07 TO 2012-04

Engineer (13-person development team) at **Major Electronics Manufacturer Division Production Management System Construction**

Windows, Java, JSP, PL/SQL, Struts FW, Oracle, Apache

- Developed production management system for major electronics manufacturer division.
- Used Java/JSP for UI processing and PL/SQL for database updates.
- System scope: 73 screens, 122 tables.
- Responsible for Java-side detailed design, development, and testing.
- Handled process from detailed design through implementation and testing.
- Performed pre-release program modifications, investigations, and performance tuning.
- Initially one of 4 Java developers assigned through November. Due to recognized technical ability, became sole Java technical lead retained through March release and post-release period. Also received requests for long-term maintenance role.
- Despite waterfall development style, limited time and resources led to irregular development scenarios like implementing specifications provided verbally without design documents.
- With no design documents available and only source code for reference, analyzed and understood program design from source when making changes or modifications.
- As team's web application expert, provided technical proposals and explanations of web-specific constraints to support design process.
- With tight release schedule, maintained extra careful health management, achieving perfect attendance while focusing fully on work.

2008-10 TO 2011-10

Engineer at **Mid-sized IT Consulting Company Internal Venture**

Windows, Java, JSP, HTML, CSS, JavaScript, Struts2, MySQL, Apache

- Built various systems including point system for major restaurant chain, parts batch processing, election supporter management system, Twitter discount aggregation site, additional features for Pacific League baseball team analysis system, Pacific League portal site management system, mobile site CMS, beauty salon reservation system, and more.

2011-02-21 TO 2011-10-07

Engineer (4-person development team, plus design company staff) at **Major Restaurant Chain Portal Site**

Linux(RH), Java, JSP, Struts2, MySQL, Apache

- Built restaurant chain portal site as per client request.
- System scope: 26 screens, 85 tables.
- Handled from basic design through implementation and testing.
- Received HTML designs from design company and converted to JSP.
- Made JSP code easily understandable from HTML perspective as design company staff sometimes needed to update directly.
- Followed mobile site's code and design patterns as this was a derivative project of an existing mobile site.
- Additionally implemented PC site functionality for the separate mobile site management system.

2009-04-01 TO 2009-05-31

Engineer (3-person development team) at **Retail Store E-commerce Site and Order Management System Construction**

Linux(RedHat), Java, JSP, HTML, JavaScript, CSS, Struts2, MySQL

- Project to convert Excel-based group company mail-order management to web system.
- Customer site: 25 screens
- Admin site: 48 screens
- Database: 33 tables
- Responsible for requirements definition, basic design, DB design, detailed design, implementation, design application, testing, delivery, and data migration.
- Managed project schedule.
- Handled customer-facing e-commerce site and client-side admin site from requirements through design, implementation, testing, delivery, and data migration.
- Integrated with Yamato Transport B2 and Sagawa Express e-Hikyaku shipping slip output systems.
- Successfully delivered and launched within 2 months from requirements definition, maintaining migration schedule.
- Upon discovering low IT literacy among admin site users during usability testing, created role-specific guides combining operation manual and help documentation. Streamlined post-login homepage and menus to show only few essential functions per user, greatly improving user satisfaction.
- Implemented strict permission management due to handling of personal information.
- Minimized operational burden by enabling client-side configuration options, resulting in only a few support inquiries even after 2.5 years.

2009-02 TO 2009-04

Engineer (5-person development team) at **Restaurant Chain Group Point System Development**

Linux(RedHat), Java, JSP, HTML, JavaScript, CSS, Struts, MySQL, Apache,

- Built and operated point management/CRM system for major restaurant chain group.
- Added lottery feature to mobile site.
- Created batch process for retrieving weather information used in analytics system.
- Developed point expiration processing and pre-expiration email notification batch jobs.
- Handled design, implementation, and testing of above features.
- Successfully created mobile site, executable JAR files, and email functionality despite being first experience with these technologies, by researching and implementing independently.

2008-10 TO 2009-01

Engineer (1-person development team) at **Customer and Event Management System Development**

Linux, Java, JSP, HTML, JavaScript, CSS, Struts, MySQL, Apache

- Built system to manage customer information for client-hosted meetings and workshops, including staff attendance management and communication management.
- System scope: 46 screens, 22 tables
- Responsible for requirements definition, basic design, DB design, screen design, implementation, design application, testing, operations maintenance, and data migration support.
- Managed schedule and progress (coordinating with designers and HTML coding staff).
- First month spent creating practice programs as client coordination was incomplete and it was my first Java development experience. Successfully completed rapid development from requirements to delivery in effectively 2 months.
- Successfully managed overall project schedule for the first time, meeting promised delivery date.
- Due to frequent internal transfers and department name changes at client, made system user-friendly with bulk operations for transferring attendee staff departments and responsibilities to different staff members.
- Implemented strict per-feature permission management due to handling personal information. Designed permissions to be role-based with bulk management and user-configurable settings.

2008-03 to 2008-09

Engineer (1-person development team) at **Feature Phone Web Apps - Personal Finance Management and Group Expense Tracking (Web Development Study)**

Windows, PHP, MySQL

- Built using PHP during evenings after SAP project work and on holidays.
- Created personal finance web app to manage credit cards, electronic money, and cash in one place.
- Developed mobile site for group expense tracking and splitting to record shared expenses and ensure equal cost distribution among participants.
- Built to solve issue of tracking individual contributions when one person pays for group outings.
- Responsible for all project phases.

2008-06 to 2008-09

Engineer (Development team of ~10 people, Project scale of over 100 people) at **Major Chemical Manufacturer SAP Addon Development**

SAP, ABAP, Oracle, Windows

- SAP implementation and addon development.
- Handled detailed design of new programs, created detailed design documents, DFDs, implementation, test specifications, and testing.
- Managed specification changes.
- Performed performance tuning.
- Conducted investigations.
- Beyond regular new module development, gained experience with large-scale programming through screen program development with 4 new hires and collaborating with international team members on previously stalled screen programs.
- Upon learning design team was struggling to identify modules used by migration target modules, created and provided a program in spare time to list modules requiring simultaneous migration.
- Requested to serve as instructor for one day of ABAP training conducted jointly with subcontractor company, teaching general module development.
- Responsible phases included detailed design, development, and unit testing.

2007-08 to 2008-05

Engineer (Development team of ~10 people, Project scale of over 100 people) at **Major Electronics Retailer SAP Addon Development**

SAP, ABAP, Oracle, Windows

- SAP implementation and addon development.
- Handled detailed design of new programs, created detailed design documents, implementation, test specifications, and unit testing.
- Managed specification changes.
- Conducted investigations.
- Gained experience through OJT with support from senior team members.
- Created a personal investigation tool six months into programming that became popular and was adopted by the design team, contributing beyond assigned work. Received feedback from senior team members still with the project that the tool remains indispensable, which is very gratifying.
- Assigned as an initial team member when team was around 6 people. Team's good reputation led to expansion to over 10 members. Contributing to team's improved evaluation, which resulted in the primary consulting company requesting entire team transfer to another project during final development phase.
- Responsible phases included detailed design, development, and unit testing.

Education

2004-04-01 to 2007-03-31

Collage

- *Qualified* *Bornsetter*

Languages

Japanese	Native
English	Capable of team communication
	(Not Business-level English)

Profiles

GitHub Meeting (30min) LinkedIn
Twitter

Personal	Home Automation, IoT, Zapier, IFTTT, Universal Remote Control Mirror Polish, Film, Buffing Compounds