Chase Allen

Sr. Golang Developer

https://github.com/chase-allen-tech
☎ +1 678 250 8501
✉ chase.allen.tech@gmail.com
Lilburn, GA, US, 30047

PROFILE

I'm a software engineer with around seven years of experience building and delivering products that make real business impact. Over the years, I've taken projects from the idea stage all the way through to production, working closely with both technical and non-technical teams. I enjoy solving complex problems, improving development processes, and helping teams deliver quality results on time. My technical focus includes database design and management, and I work daily with languages and frameworks like Golang, Python, Node.js, and React.

[CORE SKILLS]
- Golang |Node.js | Python | PHP | Java
- REST APIs | Lambda | GraphQL | Serverless
- Django | Flask | Laravel
- MySQL | PostgreSQL | MongoDB | Sqlite | Firebase
- React | Angular | Typescript | HTML5 | CSS | Bootstrap | Tailwind | Material UI
- AWS | Netlify | Vercel | Heroku
- Linux | Terraform | CI/CD | Pipeline| Jenkins |Kafka | Docker | Kubernetes

I have gained further hands-on experience in industry-standard development, which I am looking to apply to a growing business. I believe in hardship and hiring me would not fail your decision.

Thank you

Chase

EDUCATION

Bachelor of Engineering **Tsinghua University** 2012-2016 Beijing	GPA **92**/ 100
Master of Computer Science **University of Kent** 2016-2018 Canterbury	GPA **3.77**/ 4.0

CERTIFICATIONS

Go (Intermediate) **Verified**	Go (Basic) **Verified**	Python (Basic) **Verified**	React (Basic) **Verified**	JavaScript (Basic) **Verified**
Go (Inter)	Go (Basic)	Python	React	Javascript

SKILLS

Golang	+7 yr	●●●●●
Gin	+4 yr	●●●●●
Fiber	+4 yr	●●●●●
GRPC	+4 yr	●●●●●
Kafka	+4 yr	●●●●●
Kubernetes	+4 yr	●●●●●
REST API	+7 yr	●●●●●
Lambda	+5 yr	●●●●●
GraphQL	+5 yr	●●●●●
AWS	+5 yr	●●●●●
Java	+3 yr	●●●●●
Node.js	+7 yr	●●●●●
Python	+7 yr	●●●●●
React/Angular	+5 yr	●●●●●
Typescript	+5 yr	●●●●●
Django	+5 yr	●●●●●
SQL	+7 yr	●●●●●
MongoDB	+5 yr	●●●●●
Docker	+4 yr	●●●●●
CICD	+4 yr	●●●●●
Jenkins	+3 yr	●●●●●
Linux	+5 yr	●●●●●

STRENGTHS

Communicative

Open-minded, patient and empathetic. Can contribute to any conversation out of curiosity for the subject.

Keen Learner

Fueled by inquisitiveness and passion to move forward. Learns quickly and effectively due to self-motivation.

EMPLOYMENT

Software Engineer (Golang)

Nike

📅 Jan 2025 - Present 📍 CA, US

- Developed and deployed data pipelines using **Databricks** for large-scale data processing.
- Implemented CI/CD workflows in **Jenkins** to automate Databricks **notebook** deployment and **job** scheduling.
- Designed and deployed Databricks **Serving Endpoints** for real-time **ML** model inference using **Golang**.
- Designed and implemented **Python**-based machine learning models for pattern recognition and predictive analytics.
- Created automation scripts in **Python** to preprocess and clean large datasets for AI training.
- Developed deep learning workflows using **Python** libraries to support natural language processing and computer vision projects.
- Built and maintained Databricks **SQL** queries, dashboards, and **alerting** mechanisms for business insights.
- Created AWS **Lambda** functions to proxy API requests through DynamoDB to Databricks endpoints, enabling secured access.
- Conducted rigorous **testing** of serving endpoints and Lambda integrations to ensure reliability and performance.
- Utilized **AWS** (EC2, S3, Lambda, IAM, DynamoDB) as the primary cloud infrastructure for application hosting and orchestration.
- Integrated **Grafana** dashboards for real-time system and model monitoring across services.
- Deployed and configured **OpenTelemetry** (otel-collector) to collect and forward metrics from cloud workloads.
- Led the development and operationalization of **AIML models**, including model training, evaluation, and serving.
- Ensured high-availability and version-controlled deployment of models using Databricks model registry and MLOps practices.
- Collaborated cross-functionally with data science, **DevOps**, and product teams to align on deployment standards and observability.
- Built resilient data **validation** layers and **monitoring** tools for production-grade ML pipelines.
- Automated performance testing for **APIs** and endpoints to support low-latency, high-throughput ML inference.
- Maintained secure and scalable infrastructure for model training and inference, integrating monitoring and alerting into each stage.

Full Stack Software Developer IV

Intuit

📅 Oct 2023 - Dec 2024 📍 CA, US

- Built and deployed a **Slack bot in Python** to automate team workflows and streamline communication within Slack channels.
- Added **AI-driven features** to the bot, including natural language understanding and decision support to make interactions more intuitive.
- Designed and ran **three simulation types** to test and model real-world use cases, helping teams visualize different scenarios before rollout.
- Used a **GenAI engine** to fine-tune performance and generate intelligent insights from complex datasets.
- Leveraged **AWS Lambda** for serverless execution, which improved scalability and reduced infrastructure overhead.
- Created **interactive data visualizations** in React and TypeScript, integrating dynamic charts and tables for richer reporting dashboards.
- Added **draggable UI cards** and other interactive elements to enhance engagement and make interfaces more customizable.
- Implemented **WebSockets** for real-time client-server communication, ensuring data updates appeared instantly across connected users.
- Integrated **GraphQL APIs** to simplify data fetching and improve backend efficiency.
- Deployed multiple services on **AWS and GCP**, using EC2, S3, and Lambda for scalable hosting and quick recovery.
- Set up **GitHub hooks** to automate deployments and strengthen CI/CD workflows.
- Partnered closely with backend developers to ensure smooth API integrations and reliable real-time data syncing.
- Implemented **Redux** for state management in React, simplifying complex data flows and improving maintainability.
- Optimized app performance through **lazy loading, code splitting**, and other front-end best practices.
- Conducted **peer code reviews** and introduced coding standards to maintain high-quality, consistent output.
- Built an **event-driven architecture in Go (Golang)** to handle real-time data streams efficiently.
- Developed and secured **REST and gRPC APIs** for seamless inter-service communication.
- Managed **infrastructure as code** using AWS CloudFormation and Terraform, ensuring reproducible and scalable environments.
- Added **terraform test** to validate configurations before deployment, improving compliance and minimizing production risks.
- Used **Next.js with SSR** to improve SEO and optimize initial page loads.
- Managed **AWS RDS** databases with a focus on performance tuning and data security.
- Wrote **unit and integration tests** with Jest and Cypress to catch regressions early and maintain reliability.
- Maintained high-performance **backend services in Go**, and deployed front-end components using AWS Fargate.
- Architected and scaled **microservices** in Golang to support cloud-native applications with high uptime.
- Designed **data warehouse and data lake** systems to support analytics and large-scale data processing.
- Built **ETL pipelines** to feed real-time analytics and machine learning workloads across departments.
- Worked with data engineers using **PySpark and Spark SQL** to enhance business intelligence reporting.
- Implemented robust **Python APIs** to ensure smooth integration between front-end and back-end systems.

- Took advantage of **Go’s concurrency model** to boost responsiveness under high-load conditions.
- Applied **BDD principles with Cucumber** to improve collaboration between technical and business teams.
- Automated **acceptance and regression testing** with Cucumber, cutting manual QA time by half.
- Built **CI/CD pipelines** using AWS CodePipeline and GitHub Actions for faster, reliable deployments.
- Worked with design teams to deliver **responsive, accessible UIs** that looked great across devices.
- Used **AWS S3 and CloudFront** for static asset delivery, cutting load times and improving reliability.
- Actively participated in **Agile ceremonies** (planning, stand-ups, retrospectives) to maintain project momentum and alignment.

Technologies: React, Golang, GraphQL, AWS, Github, Node

Golang/Python Automation Engineer

World Wide Technology

📅 Jun 2023 - Oct 2023 📍 GA, US

- Built and maintained high-performance applications in Golang and Python, emphasizing scalability, fault tolerance, and real-world reliability.
- Created and maintained **Helm charts** to simplify containerized deployments across multiple environments, reducing setup complexity for the ops team.
- Automated **CI/CD workflows** with Jenkins, cutting release times and ensuring consistent deployment quality across services.
- Managed container orchestration and load balancing to guarantee **high availability** and efficient resource use.
- Integrated **JFrog Mission Control** for artifact management, improving software distribution and version control across teams.
- Used **Spinnaker** to visualize and automate cloud-native deployments, making rollouts safer and easier to track through all environments.
- Strengthened infrastructure security by incorporating **Teleport**, giving engineers secure, auditable access to sensitive systems.
- Developed an internal **Slack bot** that automated daily updates and improved cross-team communication.
- Set up **alerting systems** to flag anomalies early, helping the team respond to production issues in real time.
- Built **Grafana dashboards** to visualize key metrics — application performance, uptime, and user activity — giving leadership real visibility into system health.
- Enhanced monitoring by integrating **Prometheus**, capturing granular metrics that led to better performance tuning and reliability.
- Managed **cloud infrastructure on GCP**, including GKE, Cloud Storage, and Cloud Functions, to deliver scalable, highly available apps.
- Automated **GCP CI/CD pipelines** using Jenkins, Terraform, and Cloud Build, minimizing manual work and speeding up feature delivery.
- Reduced cloud costs by ~25% through **auto-scaling, instance right-sizing**, and better usage monitoring.
- Automated the deployment of **GraphQL services** (TypeScript → AWS Lambda) and built custom APIs to connect internal systems.
- Worked closely with the ops team to fine-tune alert configurations — ensuring only relevant, actionable alerts reached on-call engineers.
- Combined **Grafana and Prometheus** into a unified monitoring stack that significantly cut unplanned downtime.
- Designed and deployed **scalable messaging systems** using Google Cloud Pub/Sub for real-time data processing, improving throughput by ~30%.
- Tuned **Google Cloud Datastore** queries and data models to cut latency by 20% in high-traffic systems.
- Streamlined provisioning with the **AWS Cloud Development Kit (CDK)**, enabling developers to define reusable infrastructure in code and deploy faster.
- Helped standardize DevOps practices by introducing reusable **CDK modules**, which boosted productivity and simplified resource management.
- Provided continuous feedback loops between development and operations, driving smoother releases and stronger collaboration.
- Leveraged tools like **Helm, Jenkins, and Kubernetes** to champion DevOps best practices and deliver reliable cloud-native deployments.

Technologies: Golang, Python, Spinnaker, Terraform, Mission Control, Jenkins, Kubernetes, Prometheus, Grafana, Concourse, Teleport, helm chart, GraphQL

Golang Developer | Principal Software Engineer

Eduforma

📅 Jul 2021 - Jun 2023 📍 Lyon, France (Remote from US)

- Involved in **SDLC** phases including requirement gathering, data modeling, analysis, architecture design & development of the project.
- Responsible for developing use cases, sequence diagrams, class diagrams, state chart diagrams using **UML** notations &Rational Rose.
- Constructed solutions to solve business requirements in an efficient way for e-learning.
- Proposed and led data schema changes of **Python/Django** application that improved the product with minimum effort tech debts.

- Led the engineering effort of developing a new product.
- **Golang REST API** development which is handling different APIs for system management including real-time chat, MySQL & MongoDB management, Zoom, Vimeo, Docusign API integration, security in **Linux** env.
- Developed serverless applications in Golang and deployed them to AWS **lambda**.
- Setting serverless environment with **Node.js** using express.
- Collaborated cross-functionally with Product Owners and **Media** Operations to gather and analyze requirements for scalable content processing workflows.
- Designed state-aware REST APIs and backend services using Golang to support real-time **media** metadata management for live events.
- Implemented workflow **automation** for media ingestion pipelines using AWS Step Functions and integrated observability with Prometheus and Grafana.
- Refactored legacy backend modules written in Python into efficient, maintainable Golang services, significantly reducing execution latency.
- Hands on experience in **GOCPU** and **Memory profiling** tools, dockerize components with **docker & Kubernetes**.
- Led the engineering effort of developing a new product based on **React.js** which is deployed in **Fargate**.
- Implemented **charts** using Highcharts.js, rich text editor with Froala, real time chat with Pusher, email & SMS notifications, SEO with various AWS services including creating APIs for AWS Lambda functions.
- Worked on fixing functional & performance related bugs in core product written in **Golang**.
- Created stage timeline in Web UI client using JavaScript and React, and implemented stage deletion in CLI using Golang. Used Go to run the Docker Command Line Interface tools.
- Managed **logistics** operations involving shipping carriers and truck fleets, optimizing delivery routes and schedules to enhance efficiency and reduce operational costs.
- Coordinated small package delivery solutions by collaborating with carriers, ensuring timely shipments and adherence to service level agreements (**SLAs**) to improve customer satisfaction.
- Integrated **Go Microservices** from **Python RestAPIs** to the frontend project while setting impersonate feature.
- Experience working with **GRPC** to connect backend API's. Used Testing tools Like Ginkgo and **Go mega** framework. Working on framework APIs like Struts, spring, iBatis and Hibernate.
- Implemented **Angular** Router to enable navigation from one view to the next as agent performs application tasks.
- Using GO, developed a **microservice** for reading large volume of data from **PostgreSQL** database.
- Install, monitor, and maintain **PostgreSQL&MySQL** software, implement monitoring and alerting, implement backup and recovery processes, provide system and SQL performance tuning and assist in business process integration with various data sources.
- Developed **Microservices** as APIs for Automation and keeping the data in synchronization between two database services.
- Defined of prototypes for **Microservices**, organized the project structure, and set up middlewares.
- Working in a high-volume environment, providing expertise and hands on experience with **Kafka** connect using schema registry.
- Provides online & on call coverage for production support and administration of MongoDB, and SQL on **Hadoop**.
- Integrated **Jenkins** to GIT where build triggers automatically when we push for CI/CD.
- Developed the HTTP based web services Client which calls up the existing web services to get the application related data which comes in **XML** format.
- Implemented performance tuning and memory management techniques in **PySpark** jobs, reducing runtime by 40% and minimizing resource consumption.
- Integrated **PySpark** workflows with AWS S3, Redshift, and Hadoop ecosystems, ensuring scalable, fault-tolerant data pipelines for real-time analytics.
- Deployed Docker applications on Google Cloud Platform (**GCP**) on app engine.
- Developed backup and recovery engine using VMware vSphere APIs, Golang, and **RabbitMQ** Message Bus.
- Developed SQL Statements as well as **PL/SQL** Stored Procedures for searching contractors depending on selection criteria in **Oracle**.
- Designed and implemented Container orchestration systems with **Docker Swarm** and **Kubernetes**.
- Used **GraphQL** for complete and understandable description of the data in our API, gives clients the power to ask for needs.
- Used **Hibernate** Query Language & Custom Queries to query data from Oracle in DAO.
- Wrote **Hibernate** Criteria queries for data retrievals and have performed the ORM mappings to map java entities to database tables.
- Created an automated CI/CD pipeline for overall project, on code push the pipeline would run unit tests, code quality tools, and build would trigger producing a deployable artifact that was automatically deployed to the AWS.
- Implementing the routing technique in Golang using Gorilla Mux as request router and dispatcher for matching incoming requests to their respective handler.
- Designed and implemented scalable applications using Golang on **Google Cloud Platform**, leveraging services such as App Engine and Cloud Functions to optimize performance and reduce latency.
- Developed RESTful APIs in Golang and deployed them on **GCP**, utilizing Cloud Run and Cloud Pub/Sub for efficient microservices communication and implementing CI/CD pipelines with Google Cloud Build for streamlined deployments.
- Executed comprehensive testing strategies using **terraform test**, identifying and resolving potential issues in infrastructure code early in the development cycle, which resulted in a 40% reduction in post-deployment incidents.
- Contributed to the maintenance and evolution of a large-scale, cloud-native application stack using AWS **CDK**, ensuring best practices in security, cost-optimization, and performance across all cloud resources.
- Coordinated Unit Testing by **JUnit**, Easy Mock, Integration and User Acceptance Testing and test cases for bean components.
- Used **GIT** for maintaining the version and JIRA for defect tracking.

- Used Log4j for Logging and Used Quality Center for Bug Reporting Tool.

Technologies: Golang, Gorilla Mux, gRPC, GoCPU, REST API, React, Django, Spring Boot/Java, Open EDX, AWS, terraform, Node.js, i18n/L10N

Senior Full Stack Developer | Golang Developer

Microsoft

Sep 2018 - Feb 2021 TX, US

- Followed **Agile** methodology to promote development process with 2 weeks of iteration and daily **SCRUM** meetings throughout the life cycle of the project.
- Involved in implementation of procedures for gathering the requirements from clients for solution demonstrations.
- **Migrated** the existing python/django application to golang based GRPC
- Developed new **RESTful API** services that work as a middleware between our application and third-party APIs that we will used using **Golang**.
- Using **GO**, developed a microservice for reading large volume of data(millions) from **PostgreSQL** database.
- Involved in building database Model, APIs and Views utilizing **Python**, in order to build an interactive web-based solution.
- Created **RPC**-based service for reading large volume of data from **PostgreSQL** using **GO** (**Golang**).
- Integrated **Jenkins** to git where the build triggers automatically when we push a pull request for **CI/CD**.
- Experienced in using **Go** technologies on **UNIX** and **Linux** platforms.
- Good understanding of Design Patterns, Multithreading and GO containers. Well versed in using Channels and goroutines.
- Containerized applications using **Docker**: Created docker files, built docker images for **microservices** to run on **Kubernetes** platform.
- Developed Open stacks **API** to Integrate with **Amazon EC2** cloud-based architecture of AWS, including creating machine Images.
- Designed configurable **REST APIs** on demand using Go. Worked on numerous go frameworks like gin, revel, beego.
- Used **GO's JSON** package for marshalling and unmarshalling.
- Utilized **GO Channels** and Routines to enhance performance of invoicing app.
- Using **AngularJS** created RWD and did data manipulations to display data in company standard format in UI
- Modified **jQuery** for **AJAX** calls to create interactive web pages on JSON response.
- Proficiency in multiple databases like MongoDB, Cassandra, MySQL.
- Used **Golang** to log different host system event and alert information to **SQL Server**.
- Worked on **Kubernetes** to manage containerized applications.
- Used **XML** to simplify data and allow for Platform Changes, as well as making data more available across the applications distributed platforms.
- **DynamoDB** resources using AWS Identity and Access Management (IAM), ensuring that APIs interact with the database following the principle of least privilege, enhancing the overall security framework.
- utilizing AWS Lambda and API Gateway, integrating seamlessly with **DynamoDB** to facilitate real-time data processing and retrieval, significantly improving application responsiveness and user experience.
- Worked on **Hibernate** for mapping the **Java** objects to relational database and SQL queries to fetch the data, insert and update the data from the database.
- Produced and Consumed Restful Web Services using **Postman** Rest Client, **Swagger UI**, to interact with different applications from third-party tools.
- Worked with Business users and stake holder for gathering business requirements and Convert the Business Requirements into User Stories using **JIRA**.

Technologies: Golang, Java, GCP, Mute plugin, gRPC, Prometheus, Concurrency, React, Material-UI, Tradingview, GraphQL, Kubernetes

PROJECTS

Univo e-learning

https://digitallearningsolution.net/auth, https://api.digitallearningsolution.net/api, https://api.2km.cloudplateforme.com/api, https://collegeavalon.cloudplateforme.com/v2-api, ... (+ 20)

This is an e-learning web app that provides course creation, grading, time tracking, notifications, real-time chat, virtual class with zoom, physical session for learners and teachers, localization with i18N/L10N. I was hired to take all the tech decisions for e-learning platform. I also developed the Go REST APIs controlling MySQL and MongoDB, kafka streaming and strategized and deployed a single domain solution to host multiple isolated React projects and API. It is designed platform-first with the entire server and infrastructure stack custom designed for performance and scalability for millions of users.

Abrahbrownagency

https://abrahbrownagency.com

A Go-based website for insurance, a third-generation operated independent company that has been serving Mother-Daughter agency and the related services for years. We helped to build an excel-like real-time platform for inventory management. Technically the system was designed using microservices best practices and strictly followed the agile app methodology. We have implemented workflows and guidelines for code consistency, readability, maintainability, and testability, and built multiple web pages with API integration.