Summary

Go engineer specializing in distributed systems and cloud-native solutions. Proven track record of optimizing system performance and contributing to open source.

Experience

•	Pvotal Technologies Backend Engineer	Eagle Point, United States remote 2024-03 – Present	
 Built a distributed betting engine and concierge services from the ground up with a team of 4 engineers. Developed services using event sourcing and DDD principles for platform infrastructure, resulting in 40% improved system resili Created Go-based metaprogramming solutions leveraging AST package and generate command that reduced development time l code reuse 		cture, resulting in 40% improved system resilience te command that reduced development time by 30% through	
	 Built testing solutions by implementing f1 instead of k6, reducing code base by 70% Integrated with Stripe payment system to implement efficient payment refund proce Technologies: Go, gRPC, Event Sourcing, PostgreSQL, DDD, Redis, Stripe API, Kub 	esses, enhancing customer satisfaction and financial operations	
•	HENNGE Software Engineer Intern	Tokyo, Japan onsite 2023-11 - 2023-12	
	 Collaborated with Japanese engineering team to deliver features on schedule Implemented single-table DynamoDB design reducing operational costs by 25% wh Developed infrastructure-as-code using Terraform to automate deployment across n Established CI/CD pipeline that reduced deployment time from hours to minutes 		
,	Technologies: Go, AWS, DynamoDB, Terraform, CI/CD		
•	Unimade Solutions Full stack engineer	Beirut, Lebanon onsite 2022-08 – 2023-11	
	 Built and optimized live dashboards and reports from scratch with 5 engineers, reducing processing time by 30%. Collaborated with team of 5 engineers working on trading reporting systems Revolutionized database performance by optimizing critical queries, reducing execution time by 96% (25s to 1s) Engineered WebSocket authentication system enforcing single-session limits, increasing revenue by 30% as users purchased additional accounts Developed integration layer between trading platforms (MT4/5, Ctrader) and financial reporting systems Technologies: C#, ASP.NET, WebSocket, SQL, Trading Systems Integration, HTML, CSS, JavaScript 		
Ed	lucation		
		Daimut Labonan	
•	Lebanese University B.Sc Computer Science	Beirut, Lebanon 2019-06 – 2022-06	
Ce	ertifications		
п	CIA-Big Data v3.0	Huawei	
Sk	ills		
Sp Fra Da	nguages: Go, C, C#, JavaScript, Python oken Languages: Arabic (native), English (fluent), French (intermediate), Japanese (be ameworks: gRPC, ASP.NET, WebSocket, Event-driven Architecture tabases: PostgreSQL, MySQL, Redis, EventStore, DynamoDB, BigQuery hers: Docker, Kubernetes, Terraform, CI/CD, Git, GCP, AWS, Microservices, Distribu		
Op	pen Source Contributions		
	Treenq	Source code	
0	Reduced container image size by 50% through Dockerfile optimization and layer cach Implemented repository branch connectivity with comprehensive e2e test coverage Pioneered containerization strategy for frontend applications		
Т	'echnologies: Go, Kubernetes, Postgres		
Pr	ojects		
]	Haykal	■ Closed source	
E	Interprise-grade backend framework leveraging advanced metaprogramming for rapid s	ervice development	
0	Implemented high-performance caching with PostgreSQL unlogged tables, simplifying Integrated with Restate for built-in durability and reliability across distributed servic Accelerated development through automated boilerplate generation, allowing engineer tructure code	ces, eliminating complex custom state management	
Т	'echnologies: Go, gRPC, RPC Connect, Restate, Postgres		
1	Dhangkanna	Q Source code	
Н	ligh-performance distributed game server demonstrating advanced consensus algorithm	n implementation	
0	Implemented Raft consensus algorithm for distributed state management across gam Implemented custom gRPC load balancer for distributing reads and writes	-	
Т	echnologies: Go, gRPC, WebSocket, Raft		
	Fulus	• Source code	

Production-ready financial calculations library ensuring type safety for currency operations

• Created custom financial type system that prevents common currency calculation errors through compile-time checking

• Developed comprehensive test suite achieving 100% code coverage for critical financial operations