



BRUNO KAWKA

SUMMARY

Experienced Software Engineer with deep expertise in designing, developing, and delivering scalable, high-performance solutions. Full-stack engineer specializing in Go (Golang) backend systems and React-based front-end architecture. Highly skilled in implementing concurrent applications using goroutines, channels, and mutexes to ensure safe, efficient data processing. Experienced in optimizing distributed systems, designing RESTful APIs, and building reliable microservices architectures. Adept at creating modern, responsive user interfaces with React and developing reusable, maintainable UI components. Proven ability to architect cloud-ready solutions, integrate DevOps practices, and support smooth deployment pipelines. Strong track record of driving complex projects from concept to production while collaborating effectively with cross-functional teams. Passionate about writing clean, maintainable code and improving system performance through profiling, refactoring, and best practices. Committed to leveraging innovative technologies to solve challenging problems. Continuously advancing technical skills and contributing to high-quality, production-ready software.

CORE COMPETENCIES

Frontend Engineering (React): Experienced in developing responsive, interactive user interfaces using React, Hooks, and modern component design principle

Backend Development (Golang, Java): Skilled in building scalable backend services using Go, including RESTful APIs, microservices, and clean architectural patterns.

Database Technologies: Hands-on experience with PostgreSQL, MySQL, MongoDB, and Redis, including schema design and query optimization.

Clean Code & Software Architecture: Strong focus on maintainability, modular design, scalability, and best engineering practices.

TECHNICAL SKILLS

Golang	+8 yr	AWS	+6 yr	SQL	+7 yr
Python	+8 yr	Terraform	+6 yr	NoSQL	+7 yr
Node.js	+5 yr	Docker	+6 yr	CICD	+7 yr
Java	+5yr	Kubernetes	+6 yr	React	+8 yr

EDUCATION AND TRAINING

Jagiellonian University

Sep 2012 – Jul 2016

Bachelor of Computer Science

PROFESSIONAL EXPERIENCE

Software Engineer / Gemini - (Warsaw , Poland)

Jul 2024 - current

- Built a **Slack bot** using Python to streamline processes and automate workflows within Slack channels.
 - Integrated **AI** capabilities into the Slack bot, enhancing its decision-making abilities and enabling natural language processing.
 - Designed and implemented three unique **simulation models** to effectively replicate and analyze complex real-world scenarios.
 - Optimized performance and delivered intelligent data insights by leveraging a **Generative AI** engine.
 - Employed AWS **Lambda** for serverless computing, boosting scalability while minimizing infrastructure overhead.
 - Created interactive dynamic charts and tables with **React** and TypeScript to improve data visualization and user engagement.
 - Developed **draggable** card components to enhance user interfaces, offering a more personalized and interactive experience.
 - Utilized **WebSockets** for real-time communication, enabling instant updates between the server and client.
 - Integrated **GraphQL** APIs to simplify data queries and streamline backend interactions.
 - Deployed applications to cloud platforms (**AWS** and GCP) using services like EC2, S3, and Lambda to ensure reliability and scalability.
 - Configured **GitHub hooks** to automate deployment, streamlining CI/CD pipelines.
 - Collaborated with backend teams to integrate APIs and support **real-time** data delivery.
 - Implemented **Redux** for efficient state management in complex React applications.
 - Deployed applications to cloud platforms (**AWS** and GCP) using services like EC2, S3, and Lambda to ensure reliability and scalability.
 - Configured **GitHub hooks** to automate deployment, streamlining CI/CD pipelines.
 - Collaborated with backend teams to integrate APIs and support **real-time** data delivery.
 - Implemented **Redux** for efficient state management in complex React applications.
 - Enhanced performance with techniques like code splitting and lazy loading, **optimizing** user experience.
 - Implemented server-side rendering (**SSR**) using Next.js, improving SEO and initial page load performance.
 - Managed secure and efficient database solutions with AWS **RDS**.
 - Wrote comprehensive unit and integration tests using **Jest** and **Cypress** to ensure application stability.
 - Built high-performance backend systems in **Go**, leveraging Fargate for frontend delivery.
 - Employed Behavior-Driven Development (**BDD**) using Cucumber, facilitating clear communication between developers and stakeholders.
 - Automated acceptance tests with **Cucumber**, reducing manual testing by 50% and increasing project reliability.
- Set up automated deployment pipelines with AWS **CodePipeline** and GitHub Actions, enabling continuous delivery.
- Worked with design teams to create responsive and accessible **UI** components.
 - Used AWS **S3** for static file storage and CloudFront for fast content delivery, enhancing page load speeds.

- Engineered high-performance, scalable front-end applications using modern JavaScript frameworks such as React, TypeScript, and Next.js, delivering responsive and fluid user experiences.
 - Designed and maintained reusable UI component libraries, ensuring visual consistency, accessibility (WCAG), and streamlined development across multiple product teams.
 - Automated CI/CD pipelines using **Jenkins**, significantly accelerating software release cycles and ensuring deployment consistency.
 - Collaborated with UX designers to translate wireframes and prototypes into intuitive, pixel-perfect interfaces with strong attention to detail and cross-browser compatibility.
 - Orchestrated **container** deployments to achieve high availability and optimized load distribution for critical services.
 - Optimized rendering performance through advanced state management, asynchronous data handling, and code-splitting, reducing load times and boosting application responsiveness.
 - Integrated REST and GraphQL APIs into front-end applications, ensuring efficient data flows and crafting seamless interaction patterns between UI and backend services.
 - Implemented automated testing using Jest, Playwright, and React Testing Library to improve UI reliability and prevent regressions across rapid release cycles.
 - Modernized development workflows by setting up CI/CD pipelines for frontend apps, ensuring consistent builds, automated deployments, and minimal downtime.
 - Configured real-time alerting systems to notify teams about anomalies or potential issues, facilitating faster incident response.
 - Created dynamic dashboards and data-driven visualizations using libraries like D3.js, Chart.js, or Grafana panels to enhance user insight and engagement.
- Automated the deployment of GraphQL services with TypeScript on AWS Lambda, while developing custom APIs to meet specific business needs.
- Collaborated with operations teams to fine-tune alert configurations, ensuring actionable and relevant notifications.
 - Improved site accessibility and usability through semantic HTML, ARIA roles, responsive layouts, and performance profiling using Chrome DevTools and Lighthouse audits.
 - Built internal tooling such as UI playgrounds, design-system documentation sites, and DevTools extensions to improve developer experience and productivity.
 - Collaborated closely with backend, product, and DevOps teams to deliver cohesive end-to-end solutions and maintain high-quality user-facing functionality.
 - Championed frontend best practices, code reviews, and architectural decisions, elevating the overall quality and maintainability of large-scale UI applications.

- Leveraged expertise in AWS CDK to define and deploy reusable cloud resources, improving team productivity and resource management across DevOps projects.
- Mentored junior developers, provided guidance on design patterns and clean code principles, and contributed to a culture of continuous improvement across engineering teams.
- Applied a diverse toolkit, including Helm, Jenkins, and Kubernetes, to champion DevOps best practices and cloud-native deployment strategies.

Backend Developer / XB Software(Warsaw , Poland)

Jul 2019 - Apr 2020

- Participated in all phases of the SDLC, including requirement gathering, data modeling, analysis, architecture design, and project development.
- Created use cases, sequence diagrams, class diagrams, and state chart diagrams using UML notations and Rational Rose.
- Designed efficient solutions to address business requirements for e-learning platforms.
- Proposed and implemented data schema changes for a Python/Django application, addressing tech debt with minimal effort while improving product quality.
- Developed RESTful APIs in Golang to handle diverse system management tasks, including real-time chat, MySQL and MongoDB management, and integrations with Zoom, Vimeo, and Docusign APIs in a secure Linux environment.
- Built and deployed serverless applications in Golang on AWS Lambda.
- Set up a serverless environment with Node.js using Express.
- Used Golang's CPU and memory profiling tools and containerized components using Docker and Kubernetes.
- Spearheaded the engineering effort to develop a React.js-based product, deployed on AWS Fargate.
- Implemented dynamic data visualizations with Highcharts.js, a rich text editor with Froala, real-time chat using Pusher, and notification systems via email and SMS. Improved SEO using various AWS services, including API creation for AWS Lambda functions.
- Resolved performance and functional bugs in the core product written in Golang.
- Designed and implemented a stage timeline feature in the Web UI client using JavaScript and React. Enhanced CLI functionality with stage deletion implemented in Golang. Leveraged Golang to interact with Docker CLI tools.
- Managed logistics operations by optimizing delivery routes and schedules for shipping carriers and truck fleets, reducing operational costs and improving efficiency.
- Integrated Go microservices into a frontend project originally based on Python Rest APIs, adding features such as impersonation.
- Developed backend connections using gRPC and tested APIs with Ginkgo and GoMega frameworks.
- Created a Golang microservice to efficiently read and process large datasets from a PostgreSQL database.
- Installed, monitored, and maintained PostgreSQL and MySQL databases, implementing monitoring systems, alerting mechanisms, backup and recovery processes, and performance tuning for both systems and SQL queries.
- Developed microservices for data synchronization and automation between two database services.
- Defined microservice prototypes, organized project structure, and implemented middleware layers for scalability.

- Built expertise in high-volume environments, leveraging Kafka Connect and schema registry for data processing and integration.
- Provided on-call support and production administration for MongoDB and SQL on Hadoop systems.
- Integrated Jenkins with Git to enable automated builds triggered by code pushes, streamlining CI/CD pipelines.

Frontend developer / KnubiSoft (Warszawa, Poland)

Jan 2016 - Jun 2019

- Followed Agile methodology with two-week sprints and daily SCRUM ceremonies, ensuring transparent communication and predictable delivery of front-end features across the project lifecycle.
- Collaborated closely with clients and stakeholders to gather UI/UX requirements, building interactive prototypes and demo workflows that aligned with business goals and user expectations.
- Modernized a legacy Python/Django front-end by migrating to a component-driven JavaScript/TypeScript architecture, significantly improving performance, scalability, and maintainability.
- Developed RESTful API integrations within front-end applications, enabling seamless communication with backend microservices and third-party APIs.
- Built high-performance UI components and data-driven interfaces capable of rendering and processing large datasets efficiently in the browser using optimized state management.
- Designed and implemented front-end models, reusable components, and modular UI architectures to support scalable, interactive web applications.
- Implemented lightweight RPC-inspired client-side communication layers for handling large datasets and streaming responses from APIs.
- Automated front-end CI/CD flows with Jenkins and GitHub/GitLab, enabling automatic builds, tests, and deployments for every pull request.
- Developed robust front-end applications for UNIX/Linux environments, applying software design patterns, modular architecture principles, and performance best practices.
- Improved rendering performance using asynchronous JavaScript, memoisation, lazy loading, and optimized DOM updates, ensuring smooth multi-threaded-like user interactions.
- Containerized front-end applications with Docker for consistent development environments and seamless deployment to Kubernetes-based platforms.
- Integrated front-end applications with AWS-based infrastructure, using AWS CLI and Bash scripting for environment setup, asset deployment, and configuration.
- Built UI layers that integrated with cloud platforms such as AWS Lambda, API Gateway, and DynamoDB to deliver real-time, event-driven user experiences.
- Created configurable and scalable UI architectures using modern JavaScript frameworks (e.g., React, Angular, Vue), enabling rapid feature delivery and enhanced developer experience.
- Implemented JSON- and XML-based data handling in the front-end to standardize communication between distributed systems and ensure compatibility across environments.
- Built responsive, mobile-first layouts using AngularJS, HTML5, CSS3, and modern responsive design techniques to ensure excellent usability across all devices.
- Used jQuery and modern JavaScript to perform AJAX/Fetch API calls, handle dynamic content, and manage real-time updates in the UI.

- Worked extensively with both SQL and NoSQL APIs (MongoDB, Cassandra, MySQL) to design optimal UI data consumption patterns.
- Implemented real-time event logging and alerting dashboards in the front-end to provide visibility into system operations.
- Managed front-end deployments in Kubernetes, enabling scalable, highly available delivery of user-facing applications.
- Ensured cross-browser compatibility and accessibility standards (WCAG) using semantic HTML, ARIA attributes, and thorough testing.
- Integrated AWS IAM principles into front-end applications to securely access cloud resources using least-privilege access patterns.
- Created and tested front-end API integrations using Postman, Swagger UI, and browser DevTools to validate system interactions and improve debugging workflows.
- Collaborated with business teams to translate requirements into actionable User Stories and acceptance criteria in JIRA, ensuring alignment between UX, engineering, and product goals.