

# Brian Morrison II

Senior Software Engineer



## Contact Info

Yorkville, IL  
708-341-2229  
brian@brianmorrison.me  
brianmorrison.me/portfolio  
youtube.com/brianmorrisonme  
github.com/bmorrisondev  
linkedin.com/in/brianmmdev

## Skills

### Languages & frameworks

Golang • C# • TypeScript • Node.js  
SQL • PostgreSQL • MySQL  
MongoDB • DynamoDB  
REST • MCP

### Infrastructure & tools

Docker • Kubernetes • GitHub  
Actions • Terraform • AWS • Azure  
NGINX • Linux/Windows Server  
Jenkins • Octopus • Shell Scripting

### Frontend

React • Vue • jQuery • AngularJS  
Vanilla JavaScript

## Testimonials

### Brandon Nicoll

*Staff Software Engineer — Drata*  
"I had the pleasure of working closely with Brian at Temeda. Brian is an exceptional software engineer who consistently impressed me with his skills, dedication, and unique abilities."

### Holly Guevara

*VP Developer Marketing — PlanetScale*  
"I directly managed Brian for nearly 2 years. His expertise in content creation, documentation, programming, and understanding of marketing/sales positions him as an incredible fit for any team."

### Shlomi Noach

*Software Engineer — PlanetScale*  
"I had the pleasure of collaborating with Brian at PlanetScale, where he served as a technical content creator. Brian consistently produced quality blog posts, showcasing his strong grasp of various database systems concepts."

## Profile

Senior Software Engineer with 15+ years of experience designing scalable backend systems, leading architecture decisions, and delivering robust APIs for web, mobile, and IoT platforms. Proven track record of solving performance bottlenecks, modernizing legacy systems, and deploying secure cloud-native applications in fast-paced product teams.

## Employment History & Accomplishments

### Lead Back End Developer at Temeda LLC — Naperville, IL

July 2019 - June 2022

Temeda is a telematics IoT Platform. I worked on various projects related to the .NET API, the Golang GraphQL API, and was a key member in the architecture decision-making process. I also assisted with the AngularJS & React front end applications as needed.

- Built a GraphQL API written in Go and deployed to a docker container in Azure using ACI, App Services, and API Management. The API is used for dynamically querying data from the front end, translating GraphQL queries into optimized SQL queries for best performance.
- Optimized front end load times by creating an algorithm that calculated the number of map objects in an area and placing a point at the average long/lat to display the total object count. Work included both front end logic extensions and custom API endpoints.
- Created multiple data transform services that ingest data from IoT devices/platforms and convert data to a common object model used by our internal systems.
- Utilized a test-driven development strategy to modernize a reporting engine built with C# and scheduled with Hangfire.

### Developer Educator at PlanetScale — Remote

June 2022 - March 2024

PlanetScale is a hosted database provider built to support massive databases through sharding. My primary responsibility was to manage our documentation by adding content to guide developers on how to best use PlanetScale, as well as work with various engineering teams to keep the documentation updated as new features shipped. I also contributed to our blog and YouTube channel with educational content focusing on the platform and MySQL.

- Created the first set of official PlanetScale GitHub Actions for developers to automate operations on their PlanetScale database such as creating database branches, generating branch passwords for CI pipelines, and managing deploy requests used to merge branches.
- Authored a set of DevOps guidelines for the PlanetScale docs (now located in the blog) that provided general information on the eight phases of DevOps, as well as PlanetScale-specific use cases for each phase.

### Senior Developer Educator at Clerk — Remote

March 2024 - Current

Clerk is a user management platform built for developers. My primary responsibility is to demonstrate the value Clerk can provide developers through blog articles, videos, and consistently engaging with users on social media to provide assistance. I also facilitated partnerships with other dev tool companies for collaborative works.

- Built an MCP server used with Cursor to generate reports using AirOps with drafted markdown content. These reports would provide suggestions to increase the performance of articles with answer engines like ChatGPT and Perplexity.
- Built a GitHub Actions workflow to automate generating an MJML newsletter package for uploading to our newsletter platform. This workflow integrated with Dub.sh to generate attribution links automatically.
- Created an educational content package consisting of several blog articles and videos to support our updated Supabase integration. This content covered the technical details on how Clerk and Supabase integrate using Row Level Security (RLS) by guiding the user through building a multi-tenant web application with Next.js.