Mahdi Mamashli

bitsgenix@gmail.com | github.com/m-mdy-m | m-mdy-m.github.io | Iran

Summary

Backend developer with experienced in event-driven architecture and framework design. Creator of Gland framework with broker-based messaging achieving protocol-agnostic communication. background in DSA, and technical documentation.

Technical Skills

Languages: TypeScript, JavaScript, Python, Bash, LaTeX
Backend: Node.js, Bun, Express, NestJS, event-driven patterns, WebSocket, pub/sub
Databases: PostgreSQL, Redis, MongoDB
Architecture: Event brokers, dependency injection, modular design, SOLID principles, GoF design patterns
Testing: Unit, Integration, E2E, Smoke, Stress testing
DevOps: Docker, CI/CD pipelines
Documentation: Technical writing, API documentation, architectural decision records
Tools: Vim, Linux/Unix, Git, shell scripting,

Experience

Framework Architect & Backend Developer **Jan 2023 – Present**
Open Source & Independent Projects *Remote*

- Designed and implemented Gland Framework—event-driven backend architecture with broker system supporting HTTP/WebSocket, modular component isolation, and reflection-based dependency injection
- Built QIKS caching system achieving 1M+ ops/sec with O(1) complexity, implementing LRU/LFU/MRU eviction policies, TTL expiration, and dependency graph cascade invalidation
- Developed Gland Events—radix tree-based messaging broker with wildcard subscriptions, namespace isolation, and multi-broker mesh networking with loop prevention
- Authored 15+ technical articles (10K+ views) on event-driven patterns, OOP principles, algorithm analysis, and backend architecture
- Maintained 20+ open-source repositories with comprehensive documentation, test coverage, and architectural decision records

Key Projects

Gland Framework | *TypeScript, Event-Driven, DI Container* |

- Event broker with radix tree routing, wildcard patterns, namespace isolation for 10K+ events/sec
- Reflection-based DI container with automatic controller/channel registration and lifecycle hooks
- Protocol adapters (Express implemented, WebSocket planned) with strategy-based execution patterns
- Comprehensive test suite with unit and integration tests achieving 80%+ coverage

Gland Events | *TypeScript, Message Broker, Pub/Sub* |

- Radix tree event broker with wildcard subscriptions and namespace isolation
- IOEvent pattern for type-safe request–response with strategy execution
- Multi-broker mesh networking with propagation tracking and automatic loop prevention

QIKS Caching System | *TypeScript, Performance Engineering* |

- In-memory cache with 1M+ ops/sec, O(1) complexity using Map/WeakMap adapters
- Multiple eviction policies (LRU, LFU, MRU) with TTL/idle timeout and dependency graph invalidation
- Event monitoring system, namespace support, pattern matching for batch operations
- Functional programming interfaces with comprehensive unit and integration tests

AGAS HTTP Client | *TypeScript, Bun* |

- Bun-powered CLI and library with event-driven lifecycle, interceptors, and streaming support
- Request history, saved requests, automatic retries with Docker deployment

Vero Platform | *TypeScript, PostgreSQL* |

- Knowledge platform with peer review workflow, reputation system, and structured debate framework
- Complete SRS documentation, database design, ADR records, and architectural planning

Additional Tools

- **Cop**: Telegram bot for group moderation with Grammy framework, PostgreSQL, warning system, and admin panel
- **NExa**: Python CLI tool for dev.to article fetching with Docker deployment, pretty-printing, and category filtering
- **VEX**: Personal Vim configuration with LSP integration, custom keybindings, and plugin management via vim-plug

Recognition

Framework Creator – Gland event-driven backend architecture
Technical Author – 15+ articles (10K+ views), 3 comprehensive technical books
Open Source – 20+ maintained repositories with documentation and test coverage