ABHISHEK THULASI

Founding Systems Engineer

Vadodara, Gujarat | LinkedIn | GitHub | Portfolio

Executive Summary

Full-Stack Systems Engineer specializing in distributed backends, high-concurrency architecture and cross-platform infrastructure. I engineer systems from the ground up—treating infrastructure costs, resource constraints, and user retention as the defining engineering metrics. Proven track record of bridging high-level client frameworks (Flutter/Svelte) with low-level cores (Rust/Golang) to scale resource-efficient architectures universally, deploying strictly safe, native-speed compute across distributed backend services and cross-platform clients alike.

Core Competencies

Systems & Backend: Golang, Rust, C/C++ (via Bindgen/FFI), Concurrent WebSockets, WebAuthn/FIDO2, Memory-Mapped I/O.

Infrastructure & Data: AWS (ECS/Fargate/Aurora/ElastiCache), PostgreSQL, Valkey, SurrealDB, Docker.

Client & Local-First Architecture: Svelte 5, Flutter, Dart FFI, LWC, TypeScript, Tailwind CSS, SQLite, IndexedDB.

Professional Experience

Quantum Arcadia | *Chief Technology Officer* — *Vadodara, Gujarat* | **Jan 2026 – Present**

- **Asynchronous Spatial Streaming Blueprint:** Identified fatal Out-Of-Memory (OOM) constraints on standalone VR hardware (8GB VRAM limit). Architected the technical blueprint for an Addressable Spatial Grid, dictating platform-specific dynamic chunk loading to ensure cross-platform viability between Gen-9 Consoles and Meta Quest 3.
- **Binary Delta Patcher MVP:** Engineered a custom patching CLI in Rust to eliminate massive CDN bandwidth costs. Utilized memory-mapped I/O, `bsdiff`, and `zstd` to generate byte-level patch archives, achieving a ~98% reduction in update payload size.
- **Frictionless Identity Infrastructure:** Built a FIDO2/WebAuthn identity provider (Golang/Svelte). Implemented an email-first OTP flow with strict rate-limiting, utilizing Valkey for microsecond ephemeral validation prior to permanent PostgreSQL persistence.
- **Decoupled Backend Topology:** Architected a high-concurrency Golang monolith. Isolated stateless profile payloads (Protobuf via gRPC) from a stateful WebSocket layer handling social matchmaking and dynamic UDP server provisioning via Edgegap.

Altonexus | *Salesforce Developer* — *Remote* | **Dec 2024 – Dec 2025**

- **Enterprise API Architecture:** Engineered a metadata-driven Polymorphic REST API via a Factory Pattern to decouple custom logic. Enabled single-endpoint hierarchical SObjects upserts, reducing integration setup time by ~90% across major enterprise deployments.
- **Fleet Management ERP:** Architected a maritime ERP digitizing the chartering lifecycle. Built an LWC Daily Vessel Log with a custom grid editor for real-time tracking, and a dynamic Document Generation Engine for automated contract version-control.

InfoBeans | *Software Engineer* — *Vadodara, Gujarat* | **Jul 2022 – Dec 2024**

- **Frontend App Architecture:** Engineered the mobile frontend for India's largest adhesive manufacturer, structuring a complex "Super App" housing 7 distinct integrated applications. Built a highly scalable library of reusable Flutter components for a 1M+ user base.
- **Workflow Automation:** Shipped a custom OCR automation engine in $<$30 days to replace a deprecated native Salesforce scanning tool. Leveraged Flutter, Gemini API, and complex Salesforce transaction mapping to reduce manual data entry time by ~90%.
- **Financial Tooling & Document Generation:** Developed a real-time LWC ROI calculator processing complex pricing heuristics. Implemented client-side PDF generation, enabling dynamic deal structuring that drove multi-million rupee revenue increases.

Selected Engineering Projects

Raw WebSocket Architecture | *Golang, WebSockets, SurrealDB, Svelte* — *Systems Research* | **2024**

- Engineered a full-duplex chat system from scratch to deeply explore raw TCP persistence and memory management. Leveraged Go channels and goroutines within a custom Hub-and-Client architecture to safely multiplex raw WebSocket streams across concurrent users, explicitly bypassing managed real-time abstractions (e.g., Socket.io).

LocalSync (Local-First Sync Engine) | *Rust, Flutter, Dart FFI, SQLite* — *Architecture Exploration* | **2025**

- Built a prototype for a local-first active directory sync designed to bypass cloud storage and recurring operational costs. Validated the core peer-to-peer device communication pipeline over local Wi-Fi, utilizing a Rust core to drive a cross-platform Flutter UI via highly efficient FFI integration.

Architecture Decision Records (Selected)

Compute Workload Bifurcation (ECS Express vs. Fargate)

Decision: Abandoned AWS App Runner due to vendor deprecation. Architected a bifurcated deployment on the ECS control plane: routing stateless REST APIs through low-overhead ECS Express Mode, while isolating long-lived, stateful WebSockets onto ECS Fargate.

Rationale: Maintained the rapid iteration speed of a single codebase while optimizing infrastructure for network protocols. Express Mode handles rapid burst-scaling for the Identity API, while Fargate provides the granular network and timeout control required to prevent WebSocket connection drops.

Telemetry Serialization & API Bifurcation (Protobuf vs. JSON)

Decision: Implemented gRPC/Protobuf for all high-frequency game data and WebSockets, restricting JSON/REST strictly to infrequent initial authentication requests.

Rationale: Mitigated severe Garbage Collection (GC) stutter on mobile hardware caused by parsing large JSON strings in C#. Protobuf's tightly packed binary format bypassed massive heap allocations and mathematically reduced payload size, directly lowering AWS data transfer out (DTO) operational costs.

Selection of In-Memory Data Store (Valkey vs. Redis)

Decision: Intercepted default Redis integrations to provision AWS ElastiCache (Valkey).

Rationale: Identified Valkey as a 1:1 drop-in replacement for Redis infrastructure, securing a 33% reduction in infrastructure costs. This shift protected startup runway without sacrificing sub-millisecond API performance for OTP validation and rate-limiting.

Cross-Platform Compilation Fences (Assembly Definitions)

Decision: Enforced strict, compiler-level "fences" using Unity Assembly Definitions (.asmdef) across the entire project repository.

Rationale: Eliminated the risk of "platform bleeding" between Gen-9 PC and Meta Quest 3 builds. Mandated an Interface-First architecture that strictly isolates third-party plugins and hardware SDKs, preventing obfuscated DLLs from polluting the global assembly and breaking CI/CD pipelines.

Game Launcher Tech Stack (Rust + Flutter FFI)

Decision: Built a cross-platform Flutter Frontend connected to a Rust Core via `flutter_rust_bridge` (FFI).

Rationale: Consolidated UI development into a single framework to eliminate the operational overhead of multi-platform teams. Leveraged Rust's zero-cost abstractions and manual memory management to guarantee the patching engine (BLAKE3 hashing, memory-mapped I/O) runs at maximum bare-metal disk speed.

Education

Navrachana University | *B.Tech, Information Technology (CGPA: 7.92)* *Vadodara, IN* | **2018 – 2022**