

Arpan Das

+917534927815 | dasarpan471@gmail.com

<https://www.linkedin.com/in/arpandas-adasarpan/> | <https://github.com/adasarpan404>

SUMMARY

Backend Software Engineer with 4+ years of experience building high-throughput distributed systems using **Node.js** and **AWS**. Experienced in **event-driven architectures**, **caching systems**, **asynchronous processing**, and large-scale messaging platforms handling millions of events daily. Strong focus on **system reliability**, **performance optimization**, and scalable backend infrastructure.

TECHNICAL SKILLS

Languages: JavaScript, TypeScript

Backend Technologies: Node.js, Express.js, Fastify, REST APIs, WebSockets, Microservices Architecture, Authentication & Authorization, API Design

Frontend Technologies: React.js, HTML5, CSS3, Tailwind CSS, Material UI, Redux Toolkit, Context API, React Query, Responsive UI Development

Databases & Caching: MongoDB, Redis, Elasticsearch, Redis Lua Scripts, Cache-Aside Pattern, Distributed Caching

Cloud & DevOps: AWS (Lambda, ECS, Fargate, EC2, SQS, S3, CloudWatch, ELB), Docker, Kubernetes (Familiar), Git, GitHub Actions, GitLab CI/CD, Linux

Messaging & Distributed Systems: Bull Queue, Redis Pub/Sub, MongoDB Change Streams, Event-Driven Architecture, Asynchronous Processing, Queue-Based Systems, Distributed Job Processing, Provider Routing

Monitoring & Reliability: AWS CloudWatch, Logging, Fault-Tolerant Systems, Retry Mechanisms, Blue-Green Deployment, Performance Optimization

AI & Integrations: Google Gemini API, Retrieval-Augmented Generation (RAG), Vector Search, Embedding Models, Prompt Engineering, Recraft Vectorize, Recraft V3 SVG, Zoho Extension SDK, Stripe SDK

Other Tools: Socket.IO, Postman

EXPERIENCE

Senior Backend Engineer

Feb 2022 – Present

Guni SMS – Messaging Platform

- Engineered and scaled backend infrastructure for a high-throughput messaging platform, building reliable distributed services for **asynchronous processing**, provider routing, and real-time communication workflows.
- Architected a multi-stage AWS **Lambda** bulk messaging pipeline (L1→L2→L3) using **SQS**, **S3**, **Redis**, and **MongoDB** to process **1M SMS/MMS messages per hour** (278 msg/sec sustained), with provider batching reducing outbound HTTP requests by 99% and **Redis** DLR caching enabling high-speed inbound reply correlation.
- Engineered scalable multi-provider SMS reply processing pipelines using **AWS Lambda**, **SQS**, **Redis**, and **MongoDB**, processing **100k+ MO messages/day** with provider-agnostic webhook orchestration and asynchronous event handling.
- Owned **production deployment reliability** by leading blue-green deployment strategies using **AWS ELB**, enabling seamless feature rollouts with minimal operational disruption and faster rollback recovery during failed deployments.
- Designed **AWS Lambda + SQS** pipelines to handle high-throughput workloads, increasing processing capacity **10X** while maintaining system stability.
- Designed and optimized a **distributed** multi-provider messaging routing layer with automated fallback strategies, reducing SMS costs by **33%** and MMS costs by **50%** while maintaining high delivery success rates.
- Designed and implemented **Redis-backed** authentication and user-profile caching using Lua scripts, **Pub/Sub** invalidation, and cache-aside patterns, reducing auth-related **MongoDB** reads by 90%.
- Built distributed cache invalidation and high-throughput Redis pipelines using **ioredis**, **Redis Pub/Sub**, and **MongoDB change streams**, improving multi-instance consistency and reducing API latency on cache hits by **60–85%**.
- Engineered distributed background job processing using **Bull** and **Redis** queues, offloading asynchronous workloads from synchronous APIs and reducing average request latency by **60%** across core messaging services.
- Designed and maintained scalable **REST** and **WebSocket** services using **Node.js (Express/Fastify)**, supporting real-time messaging workflows with emphasis on reliability, modular architecture, and long-term maintainability.

- Implemented real-time data updates using **Socket.IO** and **MongoDB Change Streams**, reducing redundant API calls by 40%.
- Designed and developed a Zoho CRM extension using **React.js**, **JavaScript**, **Zoho Extension SDK**, and **REST APIs**, enabling users to send, receive, and manage SMS/MMS communications directly within Zoho CRM.
- Led backend security hardening initiatives by remediating critical application and infrastructure vulnerabilities, supporting **ISO 27001** certification readiness and reducing operational security risks across production systems.
- Drove **code quality** and team productivity by mentoring junior developers, reviewing critical backend changes, and promoting scalable engineering and debugging practices.

Senior Software Engineer (Short-Term Contract)

Dec 2025 – Feb 2026

FluidFit.ai

Remote, USA

- Architected and optimized **multimodal** AI media pipelines integrating **Google Gemini** APIs for image and video generation, improving generation reliability, response consistency, and operational scalability.
- Engineered high-performance image processing services using **Sharp** and **Jimp**, implementing advanced enhancement controls including brightness, contrast, sharpness, shadows, highlights, opacity, and midtone correction for production-grade media workflows.
- Integrated and orchestrated multiple generative AI models including **Imagen 4**, **Veo 3.1**, **GPT Image 1.5**, **Qwen Image**, and **ImagineArt** to support scalable multi-model content generation and provider fallback strategies.
- Owned the complete development lifecycle of vector generation and image vectorization features, building **React.js** frontends and integrating **Recraft Vectorize**, **Recraft V3 SVG**, and **Google Gemini APIs** to automate SVG asset creation, editing, and transformation for production design workflows.

PROJECTS

Limitly

<https://www.npmjs.com/package/limitly>

Open Source Project – npm

June 2026 – Present

- Built and published **limitly**, a production-ready distributed rate limiting library supporting **6 frameworks** (Express, Fastify, Hono, Koa, Bun, and NestJS) and **4 storage backends** (Redis, Valkey, DragonflyDB, and Memcached).
- Implemented high-performance rate limiting algorithms including **Sliding Window** and **Token Bucket**, with atomic operations using Redis Lua scripts and CAS-based Memcached updates, supporting custom strategies, IP/API-key/user-based keys, fail-open policies, and standards-compliant rate-limit headers.
- Engineered for large-scale production workloads, achieving **20,000+ requests/sec** on a single Redis node with **P95 latency below 5ms**, featuring Redis Cluster compatibility, automatic key expiration, ioredis integration, and support for APIs processing millions of requests per hour.
- Designed extensible developer APIs with built-in metrics and framework-agnostic middleware, including NestJS guards/decorators and Bun middleware composition, complemented by comprehensive documentation, examples, and sensible defaults to accelerate adoption.

OpenFeature.dev

Open Source Project

- Contributed to OpenFeature.dev by writing PHP getting-started documentation using Docusaurus to improve developer onboarding and adoption of standardized feature flagging practices.

EDUCATION

Dr. A.P.J. Abdul Kalam Technical University

Bachelor of Technology, Computer Science And Engineering

Lucknow, India

Aug. 2018 – June 2022