SADIQ MOHAMMAD

@ sadiq.build@gmail.com +91 7416414823 md-sadiq-md MdSadiqMd

WORK EXPERIENCE

Dognosis - R&D Engineer

October 2025 - Present Bangalore, India

- Solving **cancer prediction** by leveraging dogs olfactory capabilities, delivering accurate diagnostics from video-based sniff analysis
- Built **multi-stage video processing pipeline** YOLO-based dog tracking and ROI detection, centroid interpolation at 60 FPS, and stabilized 256x256 follow-cam crop generation via FFmpeg for FERAL (V-JEPA 2) model inference
- Trained and deployed **V3 Fusion LSTM** architecture combining kinematic time-series (head position, velocity, acceleration) with scalar context features (sniff duration, session progress, tortuosity)
- Managed **Raspberry Pi camera fleet** for automated daily session capture by debugging streaming protocols, service orchestration, and multi-network connectivity for reliable field deployment
- Managed cloud and on-prem infrastructure by automated data ingestion, processing, and backup workflows across distributed storage and compute environments
- Built **multi-modal sensor synchronization** matching high-frequency biosignals with video timelines, enabling time-aligned feature extraction for model training pipelines
- Overall worked with Python, OpenCV, YOLO, Airflow, TimescaleDB, PostgreSQL, Linux, FFmpeg, Docker, AWS, Rust

Scale.jobs - Founding Team Engineer

September 2024 - October 2025 Hyderabad, India

- **Only developer** in the company **besides the CTO**
- Wrote AI content generation pipelines using LangChain and OpenAI SDK, while integrating Langfuse for prompt versioning, tracing, evaluation, and production monitoring
- Built **AI Interviewer** that conducted conversational mock interviews different for each and every job role, generated personalized feedback
- Optimized frontend performance using WebAssembly and TanStack, significantly improving rendering speed and overall user experience for large interactive applications
- Overall worked with TypeScript, tRPC, Hono, Prisma, PostgreSQL, QStash, LangChain, Langfuse, OpenAI SDK, TanStack, WebAssembly (WASM), and PostHog

HCL - Tech Intern

June - September 2024 Remote

- Worked in **Resume Application Tracking System (ATS)** team
- Applied **TF-IDF** for keyword density analysis and relevance scoring
- Built OCR-based document processing and text extraction pipelines
- Overall worked with Python, TensorFlow, Scikit-Learn, NLP, TF-IDF, OCR, and document processing

SKILLS

Languages & Frameworks: Rust, Typescript, Next.js, Tanstack
Databases: PostgreSQL, Redis, Time-Series Databases
Devops: Docker, Kubernetes, Terraform, CI/CD, AWS
Additional Skills: RESTful APIs, gRPC, tRPC, Websockets, webRTC, FFmpeg, RabbitMQ, Kafka, Nix, 3D Guassian Splatting

PROJECTS

CatalystSVM

- Latency-aware ZK batch proving system for SVM-based blockchains eliminating per-transaction proving bottleneck via adaptive batch aggregation
- Built deterministic off-chain execution engine (4-opcode ISA on BTreeMap state) generating structured traces with pre/post state hashes, feeding SP1 (RISC-V zkVM) guest program to re-execute batches and produce compressed SNARK proofs (1.2MB, 28s CPU proving, 1.2s verification)
- Implemented Anchor on-chain verifier program plus Rust validator/verifier-node services, benchmarking CLI with P50/P95/P99 latency, throughput, and batch-efficiency metrics across 5 synthetic workload scenarios
- Technologies Used: Rust, SP1 zkVM, Solana, Anchor, Tokio, Clap, Plotters, SHA-256, RISC-V

TraceZero

- A privacy proxy for transactions on Solana using ZK proofs, blind signatures, and Tor routing
- Implemented Groth16 ZK-SNARK circuit (Circom) with depth-20 Poseidon Merkle tree proving membership without revealing leaf index, achieving <200k compute units on-chain verification with binding hash to prevent proof front-running and parameter substitution attacks
- Built privacy infrastructure combining Tor network (Docker SOCKS5), X25519 ECDH stealth addresses with BN254 field compatibility, AES-256-GCM payload encryption, and Rust relayer service managing persistent Merkle state with atomic writes and SHA-256 checksums
- Developed Tanstack dApp with WASM ZK proof generation, integrated Solana wallet adapters and built transaction tracer tool to verify privacy guarantees by analyzing on-chain trace chains
- Technologies Used: Rust, Axum, Tokio, Anchor, Solana, Circom, Snarkjs, Tanstack, Typescript, Docker, WebAssembly and some math

Multi-Party Computation

- Implemented Shamir's Secret Sharing for secure multi-party computation with Solana signing
- Built Pedersen Distributed Key Generation with Feldman VSS for threshold setup
- Created Threshold Signature Scheme for t-of-n EdDSA/ECDSA signing without key reconstruction
- Added Zero-Knowledge Proofs: Schnorr proofs, range proofs, and share verification
- Technologies: Rust, Shamir SSS, Pedersen DKG, Threshold Signatures, ZKP, curve25519-dalek, Tokio

EDUCATION

Bachelor's of Technology (B.Tech) - Electronics and Communication Engineering

Velagapudi Ramakrishna Siddhartha Engineering College

2021-2025 **CGPA**: 8.00