

Aswin Vodapally

Software Architect — Automotive SDV · Android Automotive · Embedded Linux

Hyderabad, India · Open to relocation and remote
aswin14887@gmail.com · +91 8008702607 · linkedin.com/in/ashwinvodapally · aswinvodapally.com

Software architect with 14+ years in embedded and automotive software, the last 9 focused on in-vehicle platforms and Software-Defined Vehicle systems. Technical authority for an 8-engineer team delivering across three General Motors vehicle lines, spanning Android Automotive OS, embedded Linux/Yocto, and Xen-based virtualization. Architected the middleware layer now shared between the Linux and AAOS IVI stacks, and led Android bring-up as a virtualized guest on consolidated vehicle hardware. Currently driving applied-AI adoption in automotive engineering — on-device LLM inference in AAOS, a domain-tuned automotive model, and AI tooling for the platform development workflow.

CORE COMPETENCIES

Architecture	In-vehicle system and middleware architecture · Multi-OS virtualization · Cross-platform framework design · Interface and HAL design · Trade-off analysis and design authority
Domain	Software-Defined Vehicle (SDV) · Android Automotive OS (AAOS) · IVI/HMI platforms · Embedded Linux · Automotive communication (SOME/IP, DDS, COVESA VSS)
Applied AI	On-device/edge LLM inference · Domain-specific model fine-tuning · AI-assisted engineering workflows · Local LLM deployment
Leadership	Technical leadership of distributed teams · Architecture and code review gates · Mentoring · OEM stakeholder engagement (on-site GM, Canada) · RFQ technical evaluation and response authoring

PROFESSIONAL EXPERIENCE

PeopleTech Group Inc. (acquired by Quest Global)

Technical Lead — Automotive Software Platform

Hyderabad, India & on-site GM Canada
Oct 2017 – Present

SDV Platform & Middleware Architecture

2024 – Present

- Architected a middleware stack built on SDV community standards (COVESA VSS, Eclipse SDV, SOME/IP, DDS) shared across both the Linux and Android-based IVI platforms, removing duplicated platform work between two parallel stacks.
- Led Android Automotive bring-up as a Xen guest VM, owning kernel fragment, /init, fstab and device configuration — enabling multiple operating systems to share consolidated vehicle hardware.
- Owned the Embedded Linux/Yocto platform for IVI, including custom distribution and layer design for vehicle-specific targets.
- Led Apple CarPlay integration on the Linux stack against MFi specification, covering transport, session and certification requirements.
- Design authority for an 8-engineer team; established architecture and code review gates and mentored engineers toward senior ownership.
- Built next-generation platform prototypes that shaped OEM technical direction: AAOS 17 with combined cluster + IVI and Multi-User Multi-Display (MUMD); zonal audio for front-passenger and rear-seat zones; and on-device LLM inference running as an AAOS system service.
- Technical evaluation of inbound OEM RFQs and authoring of RFQ responses — solution approach, architecture rationale and supporting case studies — as the technical voice in pre-sales.
- Rapid prototypes against evolving OEM requirements, working directly with BSP engineers on hardware/software integration and system-level fault isolation.

Android Automotive HMI & Framework — General Motors

2017 – 2024

- Architected and delivered production HMI applications — Connectivity, Settings, Audio and unbundled CarUiLib — across three GM vehicle lines.
- Designed Android framework services enabling cross-brand and per-brand theming from a single codebase, allowing one HMI implementation to serve multiple GM marques.
- Extended AAOS framework and VHAL/HAL layers for vehicle-specific signals and integrations.
- Worked on-site with GM in Canada as technical point of contact, translating OEM requirements into platform design and resolving cross-team integration blockers.

- Ariel (General Electric Healthcare)** — Android-based portable ultrasound device and its companion application marketplace, developed under regulated medical-device constraints.
- Designed the JNI boundary between the Android application layer and native C++ imaging libraries, integrating system-level modules into the product stack.
 - Built the inter-application communication framework using AIDL/IPC across the device’s multiple applications, plus a custom Account Manager for secure session and credential handling.
 - Developed the GE Marketplace integration — app discovery, download, install, update and uninstall workflows — and the device connectivity and settings subsystems.
 - Delivered the Over-The-Air application update proof-of-concept adopted as a product direction.
 - Applied ProGuard-based hardening across the codebase for IP protection, followed by end-to-end product validation.

Earlier Experience

2011 – 2015

- Android & Mobile Application Development**
- Launchship Technology Solutions Pvt Ltd**, Hyderabad — Software Engineer Nov 2013 – Mar 2015
- K M Innazol IT Solutions Pvt Ltd**, Hyderabad — Software Engineer Nov 2012 – Oct 2013
- Mobbisys Pvt Ltd**, Hyderabad — Software Engineer Oct 2011 – Oct 2012
- Android application and platform development: custom UI frameworks across multiple screen densities, RESTful service integration (JSON/WCF), and native library integration.

APPLIED AI IN AUTOMOTIVE ENGINEERING

- **On-device LLM inference in the vehicle.** Prototyped `llama.cpp` running as a system service inside Android Automotive OS, establishing the feasibility of local, offline-capable inference on IVI hardware without cloud dependency.
- **Domain-tuned automotive model.** Trained a lightweight LLM on an automotive-specific dataset and deployed a local LLM environment, giving the team a private inference capability suitable for OEM-confidential material that cannot leave the network.
- **AI tooling for the platform development workflow.** Built internal AI skills that automate recurring engineering work: Figma design-pack ingestion into implementation, project scaffolding, forking Google upstream repositories into the internal tree, and the commit-to-Gerrit change-request workflow.

SELECTED ARCHITECTURE DECISIONS

- **Kotlin and MVVM for HMI, adopted early (2018).** Migrated the Java-based HMI applications to Kotlin on an MVVM architecture ahead of broad automotive adoption, accepting team retraining and a mixed-language build in exchange for null-safety, testable view models and a clean separation between HMI logic and vehicle data sources — the structure the applications still build on.
- **Standards-based middleware over a proprietary abstraction.** Built the shared Linux/AAOS middleware on COVESA VSS and SOME/IP rather than an in-house signal layer — slower initially, but it kept the vehicle signal model portable across both IVI stacks and aligned with the OEM’s SDV roadmap.
- **Runtime resource overlays over per-brand application forks.** Implemented cross-brand theming through RROs against an unbundled CarUiLib rather than forking the HMI applications per marque — accepting the indirection and the overlay build-and-validation burden in exchange for one application codebase across multiple GM brands, with a brand refresh shipping as an overlay change that never touches application logic.

TECHNICAL SKILLS

Languages	Kotlin, Java, C++, Rust
Platforms	Android Automotive OS (AAOS), Android Framework, Embedded Linux, Yocto, Xen Hypervisor
Automotive Architecture	SOME/IP, DDS, COVESA VSS, Eclipse SDV, SOAFEE, VHAL/HAL, BSP integration, CarPlay (MFi)
Applied AI Tooling	llama.cpp, local LLM deployment, lightweight model fine-tuning, AI-assisted development tooling
	Git, Gerrit, Jenkins, Docker, CI for embedded builds

EDUCATION

- Master of Computer Applications** — JNTU, Hyderabad, India 2007 – 2010

PROFESSIONAL DEVELOPMENT

- Internal AAOS platform training — delivered to 8 engineers
- Google Automotive Partner Bootcamp — annual AAOS platform releases

WRITING

- Technical writing on automotive software architecture and SDV at aswinvodapally.com