

## SUMMARY

**12+ Years of Full Stack/Embedded and Distributed System Engineer/Data Science Engineer with emphasis on Network Security , Observability Platform and Enterprise Application Development and Management.**

- **One Year at Walmart Global Tech** Building monitoring observability Agent, Ingest pipeline and tools to help auto Root Cause Analysis System to automatically detect and mitigate the incidents at Walmart scale with SRE alerting and monitoring rules.
- **One plus Year at Workday** Leading and implementing Financial Banking and Settlement Module/Application for Digital Disbursements and maintenance of Settlement for Payroll, Student, Supplier, Customer, Expense and Quick Payees.
- **Five plus years at Microsoft** Leading and engineering suite of projects across Microsoft CDS/Dynamics 365 CRM Apps, Power Platform(Power Apps) and including Teams using Azure from ground zero to completion including geo deployments.
- **Three years and 9 months at Cisco System** of professional experience in the design & development of AAA(Authentication Authorization and Accounting ) infrastructure & Protocols in Linux for **Data center Switches and Routers**
- Software Engineer Intern at **Juniper Networks** and **Alcatel-Lucent Enterprise**.
- Having 8+ years of solid professional experience in design, development, testing, deployment, monitoring of Web based and desktop applications, API endpoints using **Java, Spring Boot, WPF, .NET and ASP NET with Azure Cloud**.
- Having solid experience in **Single Page Application(SPA)**, responsive and rich UI/Front End using Javascript, Typescript, HTML, CSS, **React**, Angular, JQuery, AJAX, XML, CSHTML.
- Having solid experience in Unit Testing using MS Unit, xUnit and Integration tests.
- Having solid experience in Relational Databases, Non Relational Database with proficiency in data modeling, database programming using **SDKs and ORM tools**.
- Proficient in Full Stack Web and Desktop CSS3(Foundation, Tailwind), HTML5(React, Angular), SQL Server, UML, NodeJS and ASP.NET Web App, REST/SOAP API, WPF and Web Services(Service-oriented and Microservice pattern) in Multi-tenant environment.
- Proficient in .Net, C#, Python, Typescript/Javascript(React, Redux,npm,fluentUI), Powershell.
- Solid experience in Object Oriented Design, **MVC**, SOLID design principles and design patterns .
- Experience leading, dev, design, test and deployment of Customer Engagement apps including Field Service, Customer Service, Sales apps, Customer Service chat App across channels WhatsApp, WeChat, Line, facebook, twitter & Chat.
- Solid work experience across software stack and cloud & On-Premise services- User ready Apps(Web, Mobile and desktop), App infra/platform, OS Infra, Networking, System/app programming .
- Proven Technical Leadership, Team member and Project Management skills(Agile Scrum teams).

## EDUCATION

- **Master of Science, Computer Science** **Aug 2014 - MAY 2016**  
**GPA : 3.7**  
University of Utah, Salt Lake City  
**PROJECTS:**
  - Implemented the paper **"OpenFlow Random Host Mutation: Transparent Moving Target Defense using Software Defined Networking"** using Python and POX Controller.
  - Implemented the paper **" SMORE: Software-Defined Networking Mobile Offloading Architecture"** using RYU SDN controller and Python. Here we have realized offloading selected user traffic to the cloud without changing underlying LTE Mobile Architecture.

### **MAIN PROJECT- fastnet : Ultra high speed Network protocol for latency sensitive applications**

Low latency between servers in a Data center network is very critical for latency sensitive applications. Regular TCP/IP network stack involves copying a data frame at multiple places and many context switches. Hence We propose a new protocol, which runs on top of Intel's Data Plane Development Kit(DPDK) bypassing the regular kernel TCP/IP stack to achieve the low latency requirements.

- **Bachelor of Engineering, Information Science and Engineering** **Aug 2006 – Jul 2010**  
**Marks : 78.36 %**  
University Visvesvaraya College of Engineering (UVCE), Bangalore University

## TECHNICAL SKILLS

- **Programming Languages** : C ,Bash, Python, C#, Go, Typescript/Javascript and, C++, Java.
- **Database** : SQL Server, Entity Framework, MySQL
- **No SQL Database** : Cassandra, MongoDB, Neo4j, Janus Graph Db, Elasticsearch
- **Client side** : CSS3(Bootstrap, Foundation, Tailwind), HTML5(React, Angular), JQuery, webpack
- **Back End** : Java, C#, NodeJS and ASP.NET, REST/SOAP, UML, OData, Web hooks, Web Sockets
- **Cloud Tech** : Azure cloud, Google Cloud Platform, AWS, Apache Kafka, Rabbit Mq

- **Desktop App** : Java, WPF App
- **Technologies** : Java, C#, .Net, powershell, npm
- **Tools** : Atlas Jira and Crucible, IIS, Nuget, azure devops, Git, gdb debugger, Efence, Strace, Clear Case, TCP Dump, Static Analyzer, acme, SVN
- **Microsoft Products** : Microsoft Dataverse, Dynamics, Teams, azure, M365/office suite.
- **Microsoft Dev Tools** : Jarvis, Kusto, Icm
- **Protocols** : HTTP/HTTPS, SSH, TLS/SSL, AAA, LDAP, RADIUS, SNMP

## PROFESSIONAL EXPERIENCE

### STAFF SOFTWARE ENGINEER

Walmart/Vorizo IT  
Sunnyvale, CA [Remote]  
Consultant

SEPT 2023 - PRESENT

- **Application Dependency Map[Go, Big Table, Kubernetes, Javascript, HTML, CSS]** Built Application dependency map by pulling the information from various sources including ingestion pipeline/storage where the network data is ingested in order to visualize the impacted applications deployed across Walmart Cloud Native Platform, One Ops and physical machines.
- **Observability Agent [.Net Core 6, C#, Azure (ADLS), GCP(Google storage) ]** Enhanced the agent to meet various requirements of Walmart to capture even more fine grained Machine summary and Network data and stream it to ADLS for further processing. Agent captures machine summary data by running various system utilities and system calls in Net Core and captures Network data by running packet capture(pcap) library every minute. It compresses captured fragments into a document and streams it to the hourly append only files on ADLS.
- **Xingest [Java, .NET Core, Elasticsearch, Kibana]** Ingest pipeline to read the fragment documents from ADLS, processes and aggregates the fragments over the configured minutes based on the machine quorum and ingest into Elasticsearch. It ingests a network document per minute and a machine summary document over the configured minutes. Initial task was to Enhance the existing ingest to scale to more than 60 clusters spanning ~3500 pods and be able to successfully ingest the network and machine data into ElasticSearch. Ingest was broken at various points . Identified bottlenecks and fine tuned various parameters like concurrency, memory limit, CPU to successfully process the incoming traffic in real time.
- **Xingest-Go [Go, .NET Core, GCP(Big Query)]** Re-implemented the **Xingest** with the minimum required feature set efficiently. Here the task was to identify the nuances of system level and the most efficient ways of parsing, processing and streaming the same onto Big Query. We were able to efficiently and successfully re-implement the ingest pipeline without missing feature parity and is being used in production.
- **Ingress Gateway Rate Limiter [Java, Python, Next JS, JS, HTML, CSS, Kubernetes]** Gathered requirement, analyzed and developed ingress gateway Rate limiter to rate limit application traffic to protect ecommerce application from unanticipated volume of traffic by working closely with GitOps team. It included the levers to authenticate and authorize the change request in order to rate limit application traffic.

### SENIOR SOFTWARE ENGINEER

Alaska Airlines  
Seattle, WA [Remote]  
Consultant

MAR 2023 - AUG 2023

- **RocketMiles Partnership Integration[ASP NET, C#, CSHTML, MVC, Javascript, HTML, CSS, JQuery, Ajax]** Integrated the RocketMiles Hotels partnership by designing and integrating a **search widget** on Alaska Air monolith.
- **Car Trawler Integration[ASP NET C#, CSHTML, MVC, Javascript, HTML, CSS, JQuery, Ajax]** Integrated CarTrawler Search Widget for seamless Car booking from Alaska Air Com.
- **Monolith To Microservices and Microsites[ASP NET, C#, React, Javascript, HTML,CSS]** Completed and Demoed POCs to move components from Monolith to Microsites and Microservices.
- **Automatic Quota Redeposit[ASP NET, C#, Azure, REST]** worked with the Team to build API Endpoint and required Azure functions to subscribe to ASQ Canceled Reservation Subscriber along with Azure Infrastructure deployment, PR, ARM and CI/CD pipeline to automatically redeposit Flight Pass quota at Caravelo.

### SENIOR SOFTWARE ENGINEER

WORKDAY  
Pleasanton, CA

JAN 2022 - MAR 2023

- **Workday Financial Application[Java, Javascript, HTML, CSS, React, Xpresso, MySQL, UML]** Involves leading capabilities around Financial Applications - design, implementation, testing, deployment of Workday Financial Apps
- **Banking and Settlement of Digital Disbursements Payment[Java, Javascript, HTML, CSS, UML]** Settlement of Digital Account payment requests to Payees using Contact Info framework(Mobile Phone Number and Email) and Xpresso.
- **Prenote Filter[Java, Javascript, HTML, CSS, UML, Entity Framework]** Inbound Direct Debit non-zero dollar amount Prenote filter for Settlement and Worker Bank Accounts.

#### Description/Responsibilities

- Designed and developed backend using Object-Oriented principles and simplified through design patterns.
- Modeled and designed Financial App data using Relational Database(MySQL), UML (Miro) and the corresponding ORM Tools.

- Designed, developed and tested Front End using HTML, CSS, Javascript, Xpresso.
- Solid and Extensive adoption of Agile Methodology - Release kickoff, Sprint planning, Grooming meetings, Sprint reviews, and Standups.

SOFTWARE ENGINEER II

MICROSOFT  
Redmond, WA

MAY 2016 - JAN 2022

- **Dynamics CRM Apps [C#, .NET, Javascript, HTML, CSS, REST API, Azure Cloud]** Integrated and extended FieldOne Sky onto Microsoft CRM
  - **Microsoft SQL Server** is used as a Data store and data modeled using relational properties.
  - Dynamics 365 **Entity Framework tool** is being used as an ORM tool to access Database.
  - Client side business logic is implemented with **Javascript, HTML and CSS** including attribute visibility, default values and field validations.
  - Server side business logic is implemented in Dynamics 365 plugins using **C#, ASP NET** which runs during the HTTP request pipeline.
- **Application-Platform Separation [.NET, ASP NET, C#, Javascript, HTML, CSS, REST API]** Involved in an effort to separate CRM application functionality into applications such as Sales, Service, Marketing solutions.
  - **Microsoft SQL Server** is used as a Data store and data modeled using relational properties.
  - Dynamics 365 **Entity Framework tool** is being used as an ORM tool to access Database.
  - Client side business logic is implemented with **Javascript, HTML and CSS** including attribute visibility, default values and field validations.
  - Client side data CRUD is performed through **REST APIs** compliant with **OData** best practices.
  - Server side business logic is implemented in Dynamics 365 plugins using **C#, ASP NET** which runs during the HTTP request pipeline.
- **Field One Migration Tool[.NET, C#, WPF, Xaml]** designed, developed and implemented Field One migration tool, which onboards customers from Field one sky environment into the latest Field Service version.
  - **Built a WPF Application** that provides UI to connect source Field One Sky instance and target Field Service instance, analyzes the source data, processes and transforms source data corresponding to version of target and performs CRUD/upsert requests into target instance.
  - **Microsoft SQL Server** is used as a Data store and data modeled using relational properties.
  - Dynamics 365 **Entity Framework tool** is being used as an ORM tool to access Database.
  - Setup **repo** and **CI pipeline** using **Azure Devops**.
  - Client side business logic is implemented using **WPF controls - Grids, Lists, Text, Buttons**.
  - State is stored in the local machine so that tool can be restarted or resumed from where it left off.
  - View Model is bound to View and interacts with the Model/Business layer through commands defined with View and View Model.
  - App business logic is implemented in using **C#** and **Dynamics 365 .NET SDK**.
- **Field Service self-service Portal [C#, .NET, Javascript, HTML, CSS, REST API, Azure Cloud]** Designed and extended AdxStudio Field Service extension into Field Service self-service portal to enable self-service capabilities to serve B2C customers.
  - **Microsoft SQL Server** is used as a Data store and data modeled using relational properties.
  - Dynamics 365 **Entity Framework tool** is being used as an ORM tool to access Database.
  - Setup **repo** and **CI pipeline** using **Azure Devops**.
  - AdxStudio portal metadata including configuration is stored as portal entities in Dynamics 365.
  - AdxStudio portal configuration data is separately zipped and packaged into Dynamics 365 solution and the corresponding App module.
  - Client side business logic is implemented with **Javascript, HTML and CSS** including attribute visibility, default values and field validations.
  - Client side data CRUD is performed through **REST APIs** compliant with **OData** best practices.
  - Server side business logic is implemented in Dynamics 365 plugins using **C#, ASP NET** which runs during the HTTP request pipeline.
- **Field Service App [.NET, C#, Javascript, Typescript, HTML, CSS, REST API]** In order for the better maintainability and extension of the Field Service Front End, converted and refactored all the ~157 Field service entity libraries into typescript Project with thorough testing.
  - **Microsoft SQL Server** is used as a Data store and data modeled using relational properties.
  - Dynamics 365 **Entity Framework tool** is being used as an ORM tool to access Database.
  - Client side business logic is implemented with **Typescript, HTML and CSS** including attribute visibility, default values and field validations.
  - Client side data CRUD is performed through **REST APIs/Web APIs** compliant with **OData** best practices.
  - Server side business logic is implemented in Dynamics 365 plugins using **C#, ASP NET** which runs during the HTTP request pipeline.
- **Field and Customer Service Apps [C#, .Net, Typescript/Javascript, REST]** starting from Engineering fundamentals to design, dev, deployment of FCS Apps(~M MAU)/features covering SDLC verticals, upgrades, customizability etc

- Onboarded The biggest customers by reducing the install time to ~8 hours from ~16 hours which involved investigation and optimizations around the install process, mock upgrades, and collaboration around the platform teams.
- It also involved smoother transition onto the latest version with the required hotfixes and customizations.
- **Microsoft SQL Server** is used as a Data store and data modeled using relational properties.
- Dynamics 365 **Entity Framework tool** is being used as an ORM tool to access Database.
- Client side business logic is implemented with **Javascript, HTML and CSS** including attribute visibility, default values and field validations.
- Client side data CRUD is performed through **REST APIs** compliant with **OData** best practices.
- Server side business logic is implemented in Dynamics 365 plugins using **C#, ASP NET** which runs during the HTTP request pipeline.
- **Field Service On-Prem Installer [ASP NET, C#, Azure]** Implemented authentication and authorization of Field service on-prem installer which allows installation on verifying the user license.
  - Field service installer is registered as an app in the **Azure portal**.
  - User is authenticated and consented to access resources to validate the license SKU.
  - Authorized user is granted access to install Field Service Apps.
  - Authentication flow is implemented using **C# library - Active Directory Authentication Library(ADAL)**.
- **Functional Locations [C#, .NET, Javascript, REST ]** Shipped Functional locations and Asset Area with quality hitting Million records usage in a few days.
  - Collaborated with the Finance and Operations team to move the required entities into common solution layers.
  - **Microsoft SQL Server** is used as a Data store and data modeled using relational properties.
  - Dynamics 365 **Entity Framework tool** is being used as an ORM tool to access Database.
  - Client side business logic is implemented with **Javascript, HTML and CSS** including attribute visibility, default values and field validations.
  - Client side data CRUD is performed through **REST APIs** compliant with **OData** best practices.
  - Server side business logic is implemented in Dynamics 365 plugins using **C#, ASP NET** which runs during the HTTP request pipeline.
- **Inbox [C#, Javascript/React, Redux]** Lead and Engineered Inbox experience for Customer service representatives.
  - Investigation and prototyping of retrieving and listing the conversations flowing through various channels - WhatsApp, SMS, Wechat, Line, Microsoft Teams, Messenger, Live chat and persistent chat
  - Investigation and understanding the nuances of existing Dynamics CRM constructs(PCF - Platform Custom Framework), tools (Chat widget/Control), App modules
  - Investigation, Prototyping and Integration of Inbox Control within the PCF with Session Manager
  - Collaboration with the platform teams, App teams for feature requests and regression free Release/Shipment.
  - **Microsoft SQL Server** is used as a Data store and data modeled using relational properties.
  - Dynamics 365 **Entity Framework tool** is being used as an ORM tool to access Database.
  - Setup **repo** and **CI pipeline** using **Azure Devops**.
  - **Azure blob storage** for serving static content.
  - Application cache server is implemented with **Redis key value** store for the chat messages and synced into **CRM** store at intervals.
  - Client side engine is implemented with **Javascript, React, Redux** as a **state** management library, **HTML and CSS**.
  - Chat messages are rendered as **React card components** in the chat control and chat control communicates with the Inbox App through **cross-origin post messages**.
  - **Inbox session** is loaded when Agent/user lands on Agent dashboard.
  - **Toast notification** is implemented using Dynamics 365 configuration metadata and accepting conversation via toast notification is also what opens an Inbox session with opened conversation.
  - Client side data CRUD is performed through **REST APIs** compliant with **OData** best practices.
  - Server side business logic is implemented in Dynamics 365 plugins using **C#, ASP NET** which runs during the HTTP request pipeline.
- **Unified Routing [Javascript, React, HTML, CSS, Web API]** Designed and developed responsive admin experience using react UI for Unified Routing components to configure the part of Agent assignment and Route to queue scenarios.
  - Client side engine is implemented with **Javascript, React, Redux** as a **state** management library, **HTML and CSS**.
  - Client side data CRUD is performed through **REST APIs** compliant with **OData** best practices.
  - Server side business logic is implemented in Dynamics 365 plugins using **C#, ASP NET** which runs during the HTTP request pipeline.
- **Resource Management** Lead and Engineered Crew management scenarios including URS
  - **Microsoft SQL Server** is used as a Data store and data modeled using relational properties.
  - Dynamics 365 **Entity Framework tool** is being used as an ORM tool to access Database.
  - Client side business logic is implemented with **Javascript, HTML and CSS** including attribute visibility, default values and field validations.
  - Client side data CRUD is performed through **REST APIs** compliant with **OData** best practices.
  - Server side business logic is implemented in Dynamics 365 plugins using **C#, ASP NET** which runs during the HTTP request pipeline.
- **Devops and Azure deployment** occasionally engineered devops tools/infra with extensive azure devops

- Used Azure devops for the **CI/CD pipeline** with Git as source control.
- Set up authentication and authorization through Azure App registration.
- Set up Resource Groups and Resources with the subscription.
- Setup blob storage to serve static content.
- Setup deployments using **Azure App service**.

#### **INTERN SOFTWARE ENGINEER**

**JUNIPER NETWORKS**  
Sunnyvale, CA

**MAY 2015 - AUG 2015**

- **VxLAN VTEP mac-routes and bridge-domain Statistics [ C, Python, Linux ]** : The implementation includes infrastructure to store the mac-routes statistics received from NSX controller cluster and a configuration command to enable the statistics, “show” command ( both brief and detail ) to display the statistics and “clear” command to clear the statistics.
- **EVPN with VXLAN encapsulation UT automation [ Perl ]** : Developed automation suite that includes test cases to exercise AD routes, MAC/IP routes, Ethernet segment Routes and Inclusive Multicast Routes. It also included Unicast and BUM traffic tests between Single-homed and Multi-homed ESI.

#### **GRADUATE TEACHING ASSISTANT**

**UNIVERSITY OF UTAH**  
Salt Lake City, Utah

**AUG 2014 - MAY 2015**

- **Operating Systems [Jan 2015 - May 2015]** : Assisted students in project work, programming assignments, & graded them.
- **Embedded Software [Aug 2014 - Dec 2014]** : Assisted students in project work, programming assignments & graded them.

#### **SOFTWARE ENGINEER II**

**CISCO SYSTEMS**  
Bangalore, IN

**OCT 2010 - AUG 2014**

- **Secure Shell Access [ C, Linux ]** : Developed Shell Access component which provides keyed access to underlying Bash shell with root access from the Switch Shell (Vegas Shell/VSH) by implementing Public key challenge-response mechanism.
- **Federal Information Processing Standards (FIPS) features [ C, NX-OS Switch ]** : To meet the requirements of FIPS 140-2 level 2 standards, developed features such as
  - “Integrity Test of FIPS mode configuration” using SHA message digest algorithm
  - Power-On Self Test (POST) infrastructure using OpenSSL crypto library. This supported self-tests of AES, DES, RSA, DSA, SHA, HMAC-SHA and other algorithms.
  - Developed and integrated a new entropy source for Random Number Generator (RNG) with a team of four.
- **Mapping LDAP User Group to local role on the NX-OS switch [ C ]** : Developed the feature to map group name of user on LDAP Active Directory server to local AAA role on the NEXUS Data Center Switch.
- **TLS KDF Test Harness [ C ]** : Developed TLS Key Derivation function test harness for generating Master Key and Key block.
- **AAA & Callhome Support [ C ]** : Maintenance and enhancement of AAA infrastructure and protocols for NEXUS Data Center switches. Appreciated with star and Amaze awards for fixing memory leaks, memory corruptions and buffer overflow customer bugs in AAA and Callhome.

#### **COLLEGE INTERN**

**ALCATEL-LUCENT**  
Bangalore, IN

**Feb 2010 – June 2010**

- **Dialer SNMP Agent [ C ]** : Developed SNMP agent library for Dialer feature of Omni-Access 5510 devices by implementing both read and write operations of the Dialer objects.

#### **ACADEMIC PROJECTS**

- **Bookstore management application [ Java, HTML, MySQL ]** : Developed an app to store book information & retrieve the same based on book ISBN, author and others. It also generates the bill and updates the book inventory.