Resume of Sundar R

Cloud / Solution Architect

PERSONAL INFORMATION

Name	Sundar R
Contact number	91-8072291198
Email	sundar4consulting@gmail.com
Current Role	Solution/Cloud/ Full Stack Architect

SUMMARY

- Architect for largest retailer –Retail account and built unified order Platform.

- 17+ Years of IT Industry experience with Specialization in Digital Transformation, Information Management Consulting, Enterprise Application Integration, Service-Oriented architecture, Cloud application Development.

- Worked at onsite for 3 years in one of largest bank in USA, 2 Years in Canada in one of Top5 Banks & 1 Year in United Kingdom for largest old Retailer.

- Expertise in latest Technology trends such as Micro services, Safe-Agile, Cloud Application (AWS, Azure, and Hybrid), Dev Ops, Data Engineering, and ML & AI Solutions.

- Portfolio Architect for leading largest UK retailer on Clothing &Home, International division of Enterprise Architecture Team

- Consulting Proficiency in Cloud advisory (roadmap on adoption) Program, Data Driven Digital Transformation Framework, Enterprise Capabilities Map, Safe Agile Value chain.

- Possess deep understanding and knowledge of Integration, Application Modernization, Portfolio assessment, Value chain adoption, Microservices framework, and Full Stack Engineering

EDUCATION

- **Bachelor of Computer Science**, University of Madras, 2004

TRAINING & CERTIFICATIONS

- Sun Certified Java Developer, Web developer, Webservices
- IBM MDM
- SAFE – Agile Certified
- TOGAF – Architecture Forum
- Design Thinking – Interaction Design Foundation
- IBM – Cloud Computing Foundation (Bluemix)
- AWS - Cloud Advisory, CAF, Well Architected – Boot Camp
- QWIKLABS – Cloud Solution Architecture Pathway.
- Microservices Boot Camp - Richardson
- Full Stack Engineering – Plural Sight

Cloud Architect

ARCHITECTURE COMPETENCY

Key Responsibilities*:*

- Worked with product owners, subject matter experts, and product managers to design fit-for-purpose solutions.
- Worked with clients to understand the business case for adopting cloud native approach to developing and delivering software.
- Developed client's enterprise technology, API strategy and cloud deployment models.
- Facilitated removal of client technical barriers and manage architectural runway to ensure engagement success.
- Ensured technical collaboration between pods teams.
- Provide guidance on continuous integration and test-driven development practices.
- Presented capabilities and custom demos of offerings and solutions to clients.
- Mentored and develop local architects.
- Developed thought leadership content, use cases and business cases.
- Delivered road map consisting of technology competency areas – Cloud Native, Microservices and APIs, Reactive Systems, Observability and SRE, Analytics & Streaming Pipelines, Devops – CI/CD

Capabilities / Skills / Knowledge areas:

- Excellent communication and presentation skills, both verbal and written, in English.
- Codes software components using JavaScript, Java, Spring Framework, relational databases and message queues.
- Develops approaches to testing and writes unit tests using Junit or other test frameworks.
- Creates and maintains CI builds using Jenkins, Concourse or similar tools.
- Experience in designing and deploying cloud-native enterprise applications in public or private cloud platforms (e.g., AWS, Azure, GCP, OpenShift / K8s + Containerization).
- Experience with Serverless, Reginal Deployment, Self-Healing, Compute, Auto scaling, Storage, Gateway, API Management, Networking and Security Constructs
- Deep understanding of microservices architecture concepts and how to implement them.
- Strong understanding of required patterns (microservices design: bounded context, event driven and operational isolation; 12 factor apps and principles; API design, management, and implementation).
- Demonstrated knowledge of network and security architectures.
- Proven knowledge of resilient design patterns (redundancy, autoscaling, health checks, failover strategies, avoidance of cascading failures, operational isolation, etc)
- Data, data modelling and database management- Knowledge of various database technologies and use-cases (e.g., Relational, NoSQL, Graph, Caching Options, etc.)
- Understanding of Enterprise Architecture Governance and regulatory concerns
- Deep understanding of software programming fundamental concepts.
- In-depth understanding of Domain Driven Design.
- Experience with continuous integration/deployment tools and best practices in DevOps.
- Has led complex programs and agile development teams.

- Demonstrated project management leadership.

ACHIEVEMENTS

Cloud Architect

Details
Awarded Best Integration architect for New Product rollout – Easy online payroll
Won a Hackathon conducted by largest retailer in UK – Virtual reality business case Proof of concept
Won a Hackathon for largest retailer -US for Implementation Machine learning solution – Basket Analysis (Aprior Algorithm)
Presented a Paper on 'Service-Oriented Architecture and Business Integration' and won 2nd prize for the same in Technology Day
Event Organizer of year for conducting employee engagement Program for 6000 associates on latest trends & technology– Tech Fest.
Great Deal winner Award for winning RFP worth 2mn $
Awarded best Architecture proposal solution in TACTICS – Architecture Conference
Best Demo Stall award during yearly BFS Technology Townhall – RoboAdvisor for Investments
Applied Patent in largest services company for IPR "Approach to design an efficient Online payment processing system using Half Sync /Half async Pattern"
Awarded for Best question in Tata Dialogue Forum and met IHCL CEO for lunch and shared the thought process from industry point of view. Special invitee to meet Group Chairman

----

SKILLS SUMMARY

Languages/Frameworks/Libraries, Databases, Dev tools, Business Domain Knowledge, Management Skills	Exp Level (1-5) 1 = New, 5 = Expert	Year Exp	Last used (Year)
Java	4	16	Current
J2EE	4	15	2021
Spring Boot	4	10	Current
Microservices	4	10	Current
IBM Master Data Management	3	5	2020
REST Services	4	15	current
API Documentation (Swagger, OPENAPI)	3	5	2020
API – Management (Lifecycle)	3	5	2020
API – Security (Oauth2, Okta, Architecture)	3	5	2021
Webservices – WSDL/SOAP, registry	3	5	2015
Full Stack – Angular	3	5	Current
Full Stack – React	3	3	2021
Cloud Advisory	4	5	Current
Enterprise Application Integration	4	8	2021
Cloud Architecture (Compute, Storage, Data)	4	5	Current
Software -AG / WebMethods	3	7	2019
Spring Ecosystem (Security, JPA, REST, etc)	3	10	Current
Docker & Kubernetes	3	5	Current
CI /CD - Devops	3	5	Current
IFW / IFX - Framework	2	3	2018
Cloud Application Development	3	5	Current
Azure /AWS – Well Architected Framework	3	5	Current
Application Modernization	4	8	Current
Scripting (Unix Bash, Shell, Ansible)	3	3	2020
Solution Design & Architecture	4	12	Current
Banking, Retail	3	15	Current
Leadership, Learning, Initiative, Problem Solving, Teamwork, Innovation	4	16	Current

Cloud Architect

PROFESSIONAL EXPERIENCE

(Highlighted few projects below for brevity– Additionally supported multiple projects as dual responsibility)

Largest retailer USA – Unified Order Platform
Principal Architect
November 2022 – December 2023

Role: Principal Architect largest Modernization Program.

Technologies: GCP, Spring Boot, Microservices, Kubernetes, Java, Data Engineering, Migration

Summary: Build a unified order platform consist of 40+ microservices in GCP platform with containerized deployment.

GCP Components: Spanner, Pub / Sub, Firestore, Cloud Functions, Monitoring, Java Client library

Architecture: Event Driven Architecture, REST, Cloud Well Architected Principles, Resiliency, Performance, Thread Executors, Active Monitoring,

1. Architecture Meetings: Regular meetings with Macys architects and Google SME played a pivotal role in shaping the architecture and vision of the project. The constructive discussions and collective decision-making greatly contributed to its success.
2. Performance and Order Management Platform: The efficient functioning of our performance and order management platform ensured smooth operations and customer satisfaction. Attention to detail and continuous improvement efforts were commendable.
3. Handling Peak Traffic: We tackled high traffic periods with resilience, optimizing the system to handle a surge of users and maintaining optimal performance throughout.
4. Java Performance: Proficiency in Java programming played a significant role in enhancing the overall performance and stability of the project. The code optimization techniques employed were impressive.

Successfully implemented thread pool capacity configuration for async threads, HPA scaling based on unacknowledged messages, and handling shutdown events with utmost precision. Efforts in ensuring PUB/SUB throughput monitoring, rate limiting, POD resilience measures, K8 Node Capacity planning, PUB/SUB ordering of messages, and fine-tuning HTTP connection settings.

ANZ – Integration
Solution / Cloud Architect
June 2022– Oct 2022
Technologies Used: GCP, Nginx, Apigee, Integration, Rules

Worked on Migration of Apigee to Nginx.
Provided road map and consultation for migrating intergeneration functionality to Nginx.

PayPal – Merchants Digital Replatform
Solution / Cloud Architect
April 2020 – June 2022
Technologies Used: GCP, Spring Boot, Microservices, Apache Spark, Java, Data Engineering, Migration

Working on Unified onboarding/reporting program for Merchant division of recently acquired PayPal products (Braintree, Hyperwallet, Venmo). Define solution blueprint for Data migration strategy on GCP Cloud and enable unified reporting using Apache spark/Scala ecosystem. Define Architecture Principles, process and methodologies, implementation strategy for report generation using microservices nomenclature and standards.

Cloud Architect

M&S Transformation: Capability & Strategy
Portfolio Architect – Clothing & Home
January 2018 – March 2020
Technologies Used: Microsoft Azure, Spring boot, Archimate, C4-Model, Java, CI/CD - Kubernetes

Conceptualized & implemented reference architecture for application modernization using framework & system pattern. Architecture blueprint comprising Domain Driven Design centric, microservices, API, 12 Factor APP, reusable templates, frameworks. Design and implement scalable, clustering enabled distributed cache layer using Apache-Ignite

Liaison between business & Technology to adopt new initiatives, address paint points, assist in process improvement initiatives.

BMO-Wealth Management - Smart Folio
Solution Architect
April 2015 – December 2017
Technologies Used: Spring MVC, Spring boot, Java, CI/CD – Kubernetes, IBM WebSphere, REST Services

Architected the RoboAdvisor-self servicing platform for wealth management line of business comprising of functionalities such as onboarding, credit risk, Digital Signature leveraging bank's system of record by adopting technical framework SPA, Microservices & Agile Methodologies

BMO-MFA Web /HUB-Credentials
API Architect
October 2013 – March 2015
Technologies Used: Java, J2EE, Spring-Struts, Webseal, App Scan, Policy Server, ISAM

Develop Integration Solution architecture blueprint for Investor line with Centralized Identity Access Management Components like FIM IAM, ISAM, Webseal, Policy Server. Implement API for Wealth management authentication & authorization business function.

BMO - Customer 360
MDM Advisory Consultant
June 2012 – September 2013
Technologies Used: Visio, PowerPoint, MS Office, IBM Master Data Management, IBM PIM

Develop MDM Implementation roadmap including channel cut-over strategy, Layout sequencing and milestone. Provide traceability matrix for business attribute to services, user interface and batch process. Recommended HUB/MDMSOA Service model & strategic solution for BMO One-ID approach

BOA-Ecommerce Borneo Platform
Senior Designer
August 2009 – May 2012
Technologies Used: Struts, Spring MVC, Hibernate, Maven, IBM WAS, Mysql, CAST

Design and implement Online banking applications from legacy platform to Borneo framework. Business -Authentication system, Account Overview details, Payment, Transfer, Help & Support etc.

Rearchitect /Replatform the web application aligned to customer activity view leveraging IFW Standards.

BOA-Customer System Architecture
Onsite Lead / Senior Developer
November 2007 – July 2009
Technologies Used: IBM MDM, IBM PIM, Java, J2EE, IBM WAS, Postman, Shell Scripts, DS, QS

Design and implement 360-degree view of customer to support single view for customer applications.

Cloud Architect

Provided JEE Solution leveraging IBM MDM product to enable single reference or customer system of records. Develop extensible framework for merger and acquisitions

Deep understanding Data management solution aligned to industry standards consists of Data quality process such as Address & Name Standardization, Duplicate Suspect processing, & other data transformation / quality enhancements.

- -

BOA-Object Orchestration Services
API Developer
April 2006 – October 2007
Technologies Used: Contivo, WTX, IBM XI50 , Webmethods, IS, Java, WSDL, SOAP, IBM MQ, Postman

Provide Comprehensive integration solution adopting 3T Framework namely – Transact, Transform, Transport. Implement transformation rules and XSLT development using Contivo & WTX.

Design & Implement IBM Data Power XI50 for large scale API Management solution. Develop business transformation rules using XSLT and enforce security standards for appropriate access.

- -

BOA - Enterprise Application Integration
Developer
February 2005 – March 2006
Technologies Used: Software AG Webmethods, IS, Trading Networks, Modeler, Java, XML, WSDL, SOAP

Develop components for Bank integration system leveraging Web Methods components to orchestrates and integrates disparate customer system data and helps to synchronize data across the enterprise. Develop API to support integration across bank for different business functionality

- -

Initial Learning Program
Trainee
December 2004 – January 2005
Technologies Used: Software Engineering, Data Structure

Imparted industry aligned training program both on technology and behavioural competency. Learnt Corporate

culture and best practises to solve problem by software engineering principles. Understand the importance of

Data Structure and its practical implementation using lab exercises.

- -

Cloud Architect

Project Implementations

Project 1: Global Cloud Migration

Led a complex migration project for a multinational company, successfully transitioning their entire infrastructure to GCP. Utilized Google Cloud Storage, Google Compute Engine, and Google Cloud VPN to ensure a seamless migration with minimal downtime. Implemented Cloud Armor for enhanced security and used Cloud Monitoring for real-time performance tracking. The solution reduced operational costs by 20% and improved system reliability.

Project 2: Data Analytics Platform

Developed and deployed a robust data analytics platform using GCP and Databricks for a leading financial services firm. The platform leveraged BigQuery for data warehousing, Dataproc for running Apache Spark and Hadoop jobs, and Cloud Dataflow for real-time stream processing. The solution enabled real-time insights generation, which improved decision-making speed by 30% and operational efficiency by 25%.

Project 3: E-commerce Platform Modernization

Architected and implemented a modern, scalable e-commerce platform on GCP for a major retail client. Utilized Google Kubernetes Engine (GKE) for container orchestration, Cloud SQL for relational database management, and Firebase for real-time data synchronization. The new platform significantly enhanced performance, boosted security with IAM roles, and improved the user experience, resulting in a 15% increase in customer satisfaction.

Project 4: IoT Solution Integration

Designed a comprehensive IoT solution on GCP for an industrial manufacturing company. Integrated various sensors and devices using Cloud IoT Core, and processed the data with Cloud Functions and Cloud Pub/Sub. The solution provided real-time monitoring and analytics through Looker, allowing for predictive maintenance and reducing downtime by 40%.

Project 5: Disaster Recovery System

Devised and implemented a robust disaster recovery system on GCP for a healthcare provider. Used Google Cloud Storage for data backup, Google Cloud Run for deploying containerized applications, and Cloud DNS for managing domain names. The system employed Cloud Spanner for global database management and ensured data integrity and availability in case of any system failures, reducing recovery time by 50%.

Summary of GCP Microservices Project Implementation Experience - A Game-Changing Approach

The synopsis of architecture experience and my recommendation for GCP implementation as cloud solution architect.

In today's technology-driven world, GCP has established itself as a leading cloud computing platform, offering a wide range of services to enhance digital innovation. One of the key aspects of GCP is its ability to support the implementation of microservices architectures, revolutionizing the way applications are developed and deployed.

I have outlined steps involved in successfully implementing microservices using GCP, the design considerations, and the architectural practices to follow for one of the largest retailer on US East region.

Step 1: Understanding Microservices and Cloud Principles

To embark on an AWS microservices implementation journey, it is essential to have a solid understanding of microservices architecture and the underlying cloud principles. Microservices involve breaking down applications into smaller, manageable components, allowing for independent development and deployment. GCP offers a plethora of cloud services that align perfectly with microservices, enabling seamless scalability, fault tolerance, and cost optimization.

Step 2: Well-Architected Framework

This framework provides best practices and guidance on designing and operating highly effective architectures that leverage the power of GCP services. By following this framework, we demonstrate our ability to design applications that are secure, performant, resilient, and efficient.

Step 3: Design Considerations for Microservices Architecture

1. Service boundaries: Clearly define the boundaries and responsibilities of each microservice to achieve loose coupling.
2. Service communication: Determine the communication patterns between microservices, such as synchronous or asynchronous, to ensure efficient data flow.
3. Data management: Plan how data will be managed across microservices, taking into account replication, consistency, and data access patterns.
4. Fault tolerance: Design your microservices to be resilient to failures, leveraging AWS services like Elastic Load Balancing and Amazon RDS for fault tolerance and automatic scaling.
5. Security: Incorporate security measures at every level, from network configurations to authentication and authorization mechanisms.

Step 4: Implementation: Putting Your Design into Action

6. Infrastructure Provisioning: Utilize Google Cloud Deployment Manager or Terraform to provision the required infrastructure resources, such as virtual private clouds (VPCs), subnets, and firewall rules.
7. Containerization for Microservices: Use Google Kubernetes Engine (GKE) to containerize your microservices for efficient deployment and management.
8. Service Orchestration: Use Google Cloud Workflows to manage the workflow and coordination between microservices.
9. Event Streaming: Employ Google Cloud Pub/Sub for real-time event streaming and processing across your microservices.
10. Monitoring and Logging: Implement robust monitoring and logging solutions using Google Cloud Monitoring and Google Cloud Logging, or third-party tools to gain insights into the performance, health, and availability of your microservices.
11. Continuous Integration and Deployment: Embrace Google Cloud Build, Google Cloud Source Repositories, and other DevOps tooling to automate your build, test, and deployment processes, fostering a culture of continuous integration and delivery.

Step 5: Architectural Practices for Success

- Use of serverless computing with Google Cloud Functions to optimize resource utilization and eliminate the need for server provisioning.
- Leverage Google Cloud Endpoints to provide a unified entry point for your microservices, enabling easy scaling, caching, and authentication.
- Implement distributed tracing with Google Cloud Trace to gain visibility into the end-to-end execution of requests across your microservices.
- Embrace a microservices-specific monitoring strategy through custom dashboards and alerts with Google Cloud Monitoring to proactively identify and mitigate issues.
- Incorporate automated testing, including unit, integration, and end-to-end testing, to ensure the reliability and correctness of your microservices.

Hands-on experience in Google Cloud microservices implementation with mastering the design considerations, architecture practices, and implementation steps outlined above, helped clients in developing scalable, resilient, and cloud-native applications. With Google

Cloud competency and know-how, implementation steps have tackled complex projects and brought innovation to the table.

Appendix B:

Project: Ecommerce/Online banking Migration to GCP

Client: Largest Financial Services provider – US WEST.

Microservices implementation:

- Continuous integration and continuous deployment (CI/CD)
- Private networking
- Data store
- Simplifying operations
    - Deploying Lambda-based applications
    - Abstracting multi-tenancy complexities
    - API management
- Microservices on serverless technologies
- Resilient, efficient, and cost-optimized systems
    - Disaster recovery (DR)
    - High availability (HA)
- Distributed systems components
- Distributed data management
- Configuration management
    - Secrets management
- Cost optimization and sustainability
- Communication mechanisms
    - REST-based communication
    - GraphQL-based communication
    - gRPC-based communication
    - Asynchronous messaging and event passing
    - Orchestration and state management
- Observability
    - Monitoring
    - Centralizing logs
    - Distributed tracing
    - Log analysis on AWS
- Managing chattiness in microservices communication
    - Using protocols and caching
- Auditing
    - Resource inventory and change management.