

|  |  |
| --- | --- |
| **Name: Kumar Mahadevan**  **Designation: Enterprise Architect Cell No: 9881096713**  **E-mail id:** [**kumar.devanmaha@gmail.com**](mailto:kumar.devanmaha@gmail.com) |  |
| **Educational Qualifications:**   |  |  |  | | --- | --- | --- | | **Title of the Degree with Branch** | **College/University** | **Year of Passing** | | **BACHELOR DEGREE (Comp Sc)** | Pune University | 1992 | | **XII** | Mumbai Board | 1986 | | **X** | Pune Board | 1984 |   **Experience History:**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Name of the Company** | **Designation** | **Address of the employer** | **From** | **To** | **Duration (Years)** | | Persistent Systems Ltd | Enterprise Architect | HinjeWadi Pune | December 2020 | Date |  | | Cognizant Technologies Pvt Ltd | Senior Architect | Plot # 26, Rajiv Gandhi Infotech Park, Hinjewadi, Pune | March 2009 | August 2020 | 11 | | Wipro Technologies Ltd | Systems Manager | Plot # 2, Rajiv Gandhi Infotech Park, Hinjewadi, Pune | May 2000 | Feb 2009 | 9 | | DSS InfoTech Pvt Ltd | Systems Analyst | DSS InfoTech Pvt Ltd, Nalin chambers, Pune | August 1997 | April 2000 | 3 |   **Overall IT experience:** 22+ years  **Overall Architecture experience:** 18+ years  **Training experience**: 4 years  **Major Domain work area:** Banking and Financial Services  **Major Architecture frameworks**: TOGAF 9, AWS cloud architecture  **Average team size of projects:** 25 – 30 Team members  **Relevant Project Summary: Key project Details**   |  |  | | --- | --- | | **Title: OPTIMUS App** | | | **Duration** | 1/12/2020 – Date | | **Location** | Offshore, Pune | | **Environment / Tools** | Microservices Architecture, Kubernetes, Spring Boot, React Native | | **Project Description** | The OPTIMUS Mobile App is targeted towards enabling bank customers to open an Online Savings Account & make the journey seamless and easy. OPTIMUS offers various features to its clients like Customer Onboarding, VKYC, Product cross-selling, Bundling. The Bank has decided to go for a Microservices architecture approach and rearchitect their existing system for a better Time to Market & Scaling the system for increasing their customer base. | | **Client Description** | Large Investment Bank | | **Role** | Enterprise Architect | | **Responsibilities** | 1. Analyze current state architecture of OPTIMUS 2. Based on the business drivers, functional/NF requirements, constraints, develop a target state architecture blueprint for a microservices based approach for OPTIMUS 3. Adopt a Domain Driven Design approach for identifying microservices. 4. Adopt an API led approach consisting of Experience API’s. Process API’s, System API’s for designing the microservices 5. Mentor development team for an end to end development, deployment using CI/CD principles for deploying the microservices to a Kubernetes cluster environment |  |  |  | | --- | --- | | **Title: Benchmark Platform – Cloud Native** | | | **Duration** | 1/3/2019 – 30/07/2020 | | **Location** | Offshore, Pune | | **Environment / Tools** | AWS public cloud, Lambda, API Gateway, RDS (Aurora), SQS, S3, Cognito, CodeStar, CodeCommit, CodeDeploy, Code Pipeline, Micro services Architecture | | **Project Description** | The client’s current environment had a challenge of processing different critical benchmark data using batch processes. Accuracy, High availability & configurability were the major aspects of the benchmark data. They were currently using a legacy-processing framework using batch. They were witnessing other firms working with greater agility & wanted to leverage the cloud’s infrastructure capacity model.  Since the firm had budget constraints, cost optimization was at the forefront while moving to cloud.  The target Architecture uses a Serverless model, running microservices breaking the file processing & calculation logic of the benchmark data providing the best choices from a cost, agility standpoint | | **Client Description** | European financial company | | **Role** | Cloud Native solutions architect | | **Responsibilities** | 1. Analyze current state architecture 2. Based on the business drivers, functional/NF requirements, constraints, develop a target state architecture blueprint for AWS cloud by evaluating different AWS service options eg Container vs Serverless. 3. Adopt a Domain Driven Design approach for splitting the monolith batch processing state application to a set of microservices. 4. Mentor development team for an end to end development completely on AWS cloud to use different cloud native services |  |  |  | | --- | --- | | **Title: Solution Architect enablement program** | | | **Duration** | 01/03/2018 – 21/2/2019 | | **Location** | Offshore, Pune | | **Environment / Tools** | Archimate 3.0 modelling, Banking Industry Architecture Network(BIAN) domain services, Live project case studies, Architecture Artifacts for producing architecture deliverables, TOGAF 9.1, Cloud environment AWS | | **Project Description** | This training program was envisioned to train groups Architects to enable them to become cloud native architects as a part of the digital skilling strategy. | | **Client Description** | Cognizant Academy | | **Role** | Solutions Architect | | **Responsibilities** | * Conceptualizing the entire training program based on cloud native concepts * Coming up with an Enterprise Architecture Foundation Architecture reference model & a Systems architecture reference model with reference to a case study. These reference models provide the basis for building the solutions architecture blue print for a Cloud Native solution both from a technology architecture [ eg AWS Cloud native services] & Application architecture [ e.g. micro services based architecture with 12 factor app ] * Mentoring / coaching participants during their entire solutioning phase |  |  |  | | --- | --- | | **Title: Project One – Enterprise Architecture** | | | **Duration** | 25/7/2017– 21/2/2018 | | **Location** | Offshore, Pune | | **Environment / Tools** | TOGAF 9 EA framework, Archimate 3.0, AWS public cloud | | **Project Description** | The client’s Current Consumer marketing systems have evolved over time by bolting-on systems, manual workflows and complex processes and not catering to their customers’ expectations and changed preferences.  The client is envisioning replacing the current ‘bolted-on’ systems and complex workflows with integrated, flexible, fast, and efficient processes and applications – with data in real time. These new applications will provide capability to apply science around customer’s interactions in order to proactively deliver insights, increase the speed of work and provide real time information on customer’s brand engagement and behavior. These would be based on modular Plug and Play architecture and will provide strong process control and management (FI/SOX, Regulatory oversight, Audit). Cloud based architectures will be leveraged to achieve Business Agility and Operational efficiencies thru Auto scaling, Fault tolerance, Security features.  To this effect, the client has initiated a project named Project One to replace marketing, campaign and reporting functionality currently provided by another vendor as a service and to add few new functionalities as well. | | **Client Description** | A major retailing firm | | **Role** | Enterprise Architect | | **Responsibilities** | 1. Using TOGAF ADM process for coming up with the Target state solution, model ADM Vision phase by understanding stakeholder concerns, mapping them to the Org goals, Architecture Principles 2. Model the Architecture definition phases & define the Business Architecture, Application Architecture, Integration Architecture, Security Architecture based on the Architecture Vision & guiding principles 3. Based on the ADM architecture definition & Architecture reference models come with a solution strategy for implementation eg In order to cater to the Guiding principle of Scalable & Flexible systems AWS was chosen to leverage Auto Scaling, Fault tolerance & Security features of cloud. The Architecture Principles proved as a one-stop shop for decision making. 4. Communication with different other vendor partners eg Expert for AWS cloud for ensuring buy in from all partners for proper definition & execution of an end-end solution 5. Use detailed NFR matrix (RACI) performing due diligence on existing performance numbers, enable different vendor partners to come up with their own performance strategy based on the performance breakup of individual components. This ensured performance ownership & realizing target performance required for any use case 6. Use NFR performance / capacity tasks to be included in the project plan 7. Document major architecture decisions by providing architecture alternatives & coming up with a best approach for the solution 8. Implementation Governance [ NFR traceability using tools, dashboards for status reports ] |  |  |  | | --- | --- | | **Title: Enterprise Architecture (TOGAF) Enablement** | | | **Duration** | 1/1/2017 – Date | | **Location** | Cognizant Academy Offshore, Pune | | **Environment / Tools** | TOGAF 9.1, Archimate 3.0 | | **Project Description** | This enablement program aims at internalizing the capability drive for TOGAF 9 certification within Cognizant Technologies. This drive is the first step towards OPEN GROUP’s accreditation within Cognizant. | | **Client Description** | Internal | | **Role** | Enterprise Architect [ Subject Matter Expert ] | | **Responsibilities** | * Review of OPEN GROUP’s course content for TOGAF 9.1 certification & planning for content customization to suit training needs. * Design of content customization for e-learning with inputs from OPEN GROUP’s sources * Design & development of case study for TOGAF enablement using Archimate 3.0 modelling tools. * Mentoring team on content preparation & participating in reviews * Designing practice tests & mock exams for both Part1 & Part2 combined certification course for TOGAF 9 * Delivery of TOGAF enablement program over a period of 6 weeks using a blend of e-learning & classroom channels * Connecting with participants over CONNECT sessions & mentoring, guiding them for understanding practical aspects of TOGAF, exam preparation. |  |  |  | | --- | --- | | **Title: Trading Middle office Technology modernization** | | | **Duration** | 1/5/2016 – 31/12/2016 | | **Location** | Offshore, Pune | | **Environment / Tools** | Jboss application server, Oracle 11g, Spring 4.x, Angular JS, Eclipse, Maven 3.x, JMS, Java 8, Quartz, Jenkins, Junit, CodePro Analytix | | **Project Description** | This project aims at modernizing the existing trading applications written in Visual Basic into a web based application based on the Java technology stack. The system acts as a front end to different legacy systems (OMS) that actually contain the workflow for executing the Order management process. | | **Client Description** | Large Investment Bank | | **Role** | Solutions Architect | | **Responsibilities** | * Review of existing architecture & design and suggesting appropriate changes where required. Proposed micro services based architecture against the modernization plan * Proposed a target state architecture for the modernization plan * Recommend use of frameworks like Code Pro analytix for achieving 100% code coverage * Suggest resource optimization methodologies for running batch jobs that process huge volumes of data intermittently. * Assist / mentor team members for any technical issues during development * Perform design reviews, code reviews & suggest improvements where required. * Produce high level technical architecture document based on the current design changes * Act as a technology anchor for overseeing different architectures within the business unit, collaborating with Junior Architects and mentoring them where necessary. |  |  |  | | --- | --- | | **Title: Gateway Next Gen reengineering** | | | **Duration** | 1/1/2014 – 30/04/2016 | | **Location** | Offshore, Pune | | **Environment / Tools** | Apache Tomcat 7.x (on private cloud), Oracle 11g, Spring 4.x, JPA 2.1, Angular JS, Eclipse(Luna), Maven 3.x, Spring integration 4.x, JMS, Activiti BPM 17.x, Hazel cast | | **Project Description** | This project aims at reengineering an existing Pricing Engine that used the concept of batches for pricing trades to a more automated, event based & flexible pricing architecture | | **Client Description** | Large Investment Bank | | **Role** | Solutions Architect | | **Responsibilities** | * Review of existing architecture of a system (MODIS) & design and suggesting appropriate changes where required. * Evaluation / selection of BPM tools for use within Gateway Next Gen * Participate in defining the architecture of the pricing system (Gateway) taking into account factors like state management, failover, fault tolerance, recovery, and performance. * Assist / mentor team members for any technical issues during development * Perform design reviews, code reviews & suggest improvements where required. * Analyzing performance related issues & implementing performance tuning methodologies for the UI & Server side tiers. * Re-Architect a project that aims to replace an ETL based product for application integration with a real time message processing system. * Suggestions for transforming Gateway Next Gen to a micro services based architectural approach. * Act as a technology anchor for overseeing different architectures within the business unit, collaborating with Junior Architects and mentoring them where necessary. | |  |

|  |  |
| --- | --- |
| **Title: Workbench Portal Architecture evaluation & reengineering** | |
| **Duration** | 01/06/2012 -- 20/12/2013 |
| **Location** | Offshore, Pune |
| **Environment / Tools** | Flex 4.x, Websphere 7.0, Hudson, Harvest, Java EE. |
| **Project Description** | Workbench Portal is a target information delivery solution providing a single point entry to workbench reporting tools. Workbench reporting lets users to access menu of reports, group reports into folders, schedule reports to run automatically & send reports to local printers, e-mail accounts, file folders & secure corporate internet sites in a number of standard formats like PDF.  The primary challenge of workbench flex UI is the performance of different application components with home page as the main component. |
| **Client Description** | Large Investment Bank. |
| **Role** | Architect |
| **Responsibilities** | * Requirements analysis & design for change requests. * Analyzing existing workbench flex applications for performance problems * Suggesting Performance improvement solutions. * Re Architecting / designing the applications both on the UI & Java front for implementing performance improvement * Implementing performance improvement solutions * Providing various architectural / design change recommendations for any change requests by evaluating the current architecture * Team mentoring |

|  |  |
| --- | --- |
| **Title: Client Life Cycle – Reengineering (Investment Bank)** | |
| **Duration** | 04/04/2011 – 09/30/2011 |
| **Location** | Offshore, Pune |
| **Environment / Tools** | Flex 3.0, Blaze DS, Pure MVC framework, Spring 3.x, JBOSS Drools (Rules engine), MAF (Integration framework layered on Mule ESB), Maven 2.x |
| **Project Description** | CLC is a strategic initiative aimed at reengineering the SCOT (Strategic Client On boarding Tool) for a large Investment Banking client.  The project aims at replacing the existing J2EE based SCOT application with Web 2.0 technologies like flex in the UI layer & providing features like configurable UI driven by a rules engine. |
| **Client Description** | Large Investment Bank |
| **Role** | Flex architect |
| **Responsibilities** | * Analysis of different flex frameworks & recommendation for the project. * Defining Flex architecture for the UI layer using Pure MVC * Recommending design strategies for challenging technical problems like rules driven UI, Matrix workbench integration in the UI layer. * Recommending design methodologies for the Authentication & Authorization use case. * Identify & build reusable component libraries in flex that will be used across projects * Team mentoring. |

|  |  |
| --- | --- |
| **Title: Prime Trade Rich Internet Application Proof of Concept** | |
| **Duration** | 09/07/2009 – 02/26/2010 |
| **Location** | Offshore, Pune |
| **Project Description** | Prime services division of CS is in the process of redefining their UI for the prime trade application to take advantage of the rich internet & web 2.0 features.  The POC itself was a means to evaluate different RIA technologies namely Flex, Silverlight, JavaFX, Swing against UI richness, performance for depicting real time stock updates on the UI. |
| **Client Description** | Prime Brokerage Firm |
| **Role** | Flex Architect |
| **Responsibilities** | 1. Requirements analysis & use case elicitation. 2. End to end architecture of the system using Flex, Livecycle Data Services Messaging. 3. Performance tuning the POC for accommodating exceptionally large number of messages to be received by the front end. 4. Preparation of performance test plan, performance testing the application using tools like DSStress. |

|  |  |
| --- | --- |
| **Title: Research Link Enhancements** | |
| **Duration** | 07/04/2005 – 02/23/2007 |
| **Location** | Onsite - Offshore |
| **Project Description** | The Research Link application is an integrated equity research portal. The main function of this portal is to distribute the equity research documents to its recipients. Subscribers can view the research paper through the Research Link site or can get the document through FTP or e-mail .The research documents are prepared by analysts and are published using the publishing application. Research Link does not control the publishing of the research documents. The Publishing application places the documents in a directory and Research Link picks up the document for further processing and distribution.  Technically speaking, the distribution engine (or notification engine) is the heart of the system which is primarily responsible for sending the latest research documents to its subscribers. The distribution engine operates in both asynchronous & synchronous mode. The synchronous mode of operation helps external applications to interact with the engine in order to send notifications to interested parties. In order to enhance the performance, the subscribers of the distribution engine are invoked in parallel (through multithreading)  The application also has support for multiple languages. |
| **Client Description** | Large Investment Bank |
| **Role** | J2EE designer /Architect |
| **Responsibilities** | * Preparation of design models like use case diagrams, activity diagrams, sequence diagrams, class diagrams for new functionalities * Evaluating the current processes that are used within the application and recommending improvements wherever necessary * Defining standards and best practices and recommending them to the clients to be used within the project * Leading the team and providing technical guidance to them * Participating in peer reviews for all the modules that are currently being developed * In future, I will be involved in all the major architectural / design changes that this application will undergo as a result of integration with various other subsystems |

|  |  |
| --- | --- |
| **Title: Mutual Funds Transfer Of Accounts (MFTOA)** | |
| **Duration** | 09/08/2003 – 06/24/2005 |
| **Location** | Offshore, Pune |
| **Project Description** | TOA – transfer of assets – is used to transfer customer assets in/out of the firm. Customers can sweep their assets into one account of a broker from another broker whenever they wish so. This application handles these types of asset transfers. E.g. a customer transfers all his mutual funds holdings from Broker A to the firm. The users can input transfer request related information. Once the user is satisfied that all the necessary documents and forms have been obtained from the customer, he can trigger a print job that prints out all the documents necessary for requesting an asset transfer to a broker/ mutual fund company. It also creates remainder reports for the users to follow-up on any of the pending transfers. ACAT is a system that automates the process of assets transfer. MFTOA can also update information to be sent to ACAT for further processing.  This project basically aimed at legacy integration. The system existed in the firm on HPS, which is outdated & incurs high costs for support. The plan was to replace the existing HPS front end with an open source platform like J2EE & integrate them with the existing Mainframe system by replacing the HPS backend code with COBOL & adding a layer of web service framework on top of the COBOL programs. The web service intermediary (RRBUS) is an in –house middleware project developed by the firm |
| **Client Description** | Large Brokerage Firm |
| **Role** | J2EE Designer / Architect |
| **Responsibility** | * Peer review of requirements gathering (use case & activity diagrams) * OOAD using together J * Preparing software architecture document / design document & architecting the presentation layer with well known design patterns * Use case realization * Architecture presentation through the UML standard artifacts like use case model, analysis model, design model, deployment model, implementation model * Presentation of architecture to the client * Developing critical use cases in the system like the PDF report generation using FOP, Framework development * Participating in peer reviews, documenting best practices for each phase in the RUP & its implementation * Preparation of Java coding standards * Mentoring the development team during the cut phase * Formulating deployment strategies for CUT & SIT phase * Coordinating with onsite team during SIT / UAT phase |

|  |  |
| --- | --- |
| **Title: Employee Self Service Portal** | |
| **Duration** | 04/22/2002 – 08/29/2003 |
| **Location** | Onsite – Offshore |
| **Project Description** | The Employee Portal is envisaged to be a platform to host various self-service applications for employees in an organization. It provides employees with a common window for various day-to-day tasks. The Portal is implemented by integrating Schwab’s existing ESS applications, IDESK (An innovation group of Wipro Technologies) ESS applications, and other new ESS applications. Modules like Action Forms aids managers to perform tasks like giving a Promotion / Salary revision to an employee through a well-defined workflow enabled process. The New Hire process provides a complete workflow enabled process to enable a new employee get quick access to his basic necessities like access to email id, NT workstation id, employee id etc & settle down quickly in the organization.  This project was very challenging due to a lot of new technologies being involved. The client had demanded an architecture change right in the middle of the project. We had changed the Portal based MVC architecture to a plain Pet store based architecture within a week’s time. This also had an impact on the deployment strategy & the way in which identity management was performed & the user profile retrieval. The online integration to Peoplesoft was another big challenge in this project |
| **Client Description** | Large Brokerage firm |
| **Role** | J2EE designer / Architect |
| **Responsibility** | * Defining multi-tiered software architecture in the elaboration phase. Preparation of technical architecture / design document * OOAD using Together J, preparation of PUGH Matrix & FMEA during design, preparation of UML artifacts like design model, deployment model etc * Framework Development / development of critical use cases * Preparation of Java coding standards * Preparation of unit test cases using JUnit * Conducting peer reviews * Unit & system testing * Mentoring the team during development   Apart from these responsibilities, I also participated in the major decision making process of re-architecting the presentation layer & refactoring the design for Action Forms |

|  |  |
| --- | --- |
| **Title: Marsh Wrap Up** | |
| **Duration** | 07/09/2001 – 03/29/2002 |
| **Location** | Offshore, Bangalore |
| **Project Description** | Marsh is one of the leading insurance brokerage services firm in US. Wrap-up is a specific type of insurance product catering to the needs of the large construction project owners.  Marsh Wrap is a program created by Marsh to meet the insurance requirements of the construction industry. The target clientele for Marsh for this program are principals / owners of construction programs. Marsh Wrap would create a framework where the client of Marsh (principal / owner) would provide the details of his construction program and obtain a cost-effective insurance program arranged through Marsh. It envisages that at a later stage, claims and risk management of these programs will also be handled through Marsh Wrap. Functionality: - The client decides on the construction project. The Client either appoints a prime contractor and through him sub-contractors. The Client may also appoint many contractors directly without naming any of them as a prime contractor. The client has to appoint many contractors, as one single contractor may not be equipped to carry out the various tasks required in a huge construction contract. The client therefore splits the entire work into various small groups of activities. Such small groups of activities are called bid packages. It is assumed that each bid package will be handled by a single contractor. Each bid package is further broken down into CSI line items. Each specific construction activity will be assigned a CSI code. The split of activities into bid packages and CSI line items are done on the basis of the standards established by the Constructions Standards Institute. Contractors respond to the client with their bids for each bid package. The bids would include all their costs and profits. It would also include the cost of insurance relating to the particular bid package. The client chooses the contractors it wishes to deal with and accordingly awards the contracts.  The project was implemented using the RUP process, which consisted of around 17 iterations. |
| **Client Description** | Large Insurance firm |
| **Role** | J2EE designer / Module lead / Developer |
| **Responsibility** | * Study of LLD sent from the client. * Tackling discrepancies in design & suggesting improvements for the same. * Enhancements to the design model * Coding of assigned modules * Unit Test cases preparation * Coding Unit test cases classes using JUnit * Unit & integration testing |