

APN Partner Deal Acceleration Program –**Project Plan**

[*Maruti Suzuki* ] – [MothersonSumi INfotech & Designs Ltd.] – [Date]

|  |  |  |
| --- | --- | --- |
|  | **Partner Credentials on AWS**  **Submitted By**  MothersonSumi INfotech & Designs Limited  **MMM-DD-YYYY** |  |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Changes** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1 Project Overview 4](#_Toc64441351)

[1.1 Executive summary 4](#_Toc64441352)

[1.2 Business Requirement 4](#_Toc64441354)

[1.3 Pain Points / Challenges in the current environment 5](#_Toc64441355)

[1.4 Project Success Criteria 5](#_Toc64441413)

[1.5 Pre-Requisites 6](#_Toc64441416)

[1.6 Dependencies 6](#_Toc64441417)

[1.7 Assumptions 6](#_Toc64441419)

[1.8 In-scope 7](#_Toc64441420)

[1.9 Out of Scope 7](#_Toc64441421)

[1.10 Risks and Mitigation 9](#_Toc64441422)

[1.11 Raci Matrix 10](#_Toc64441425)

[2 Solution Architecture Diagram 11](#_Toc64441476)

[2.1 Architecture on AWS 11](#_Toc64441477)

[2.2 Overview of the Architecture 12](#_Toc64441482)

[3 Project Execution / summary of milestones & deliverables 14](#_Toc64441483)

[3.1 Expected AWS Cost Breakdown by Services 14](#_Toc64441484)

[3.2 Acceptance 15](#_Toc64441485)

[4 Resources & Cost Estimates 16](#_Toc64441486)

[4.1 Project Sponsor(s) / Stakeholder(s) / Project Team 19](#_Toc64441487)

[Appendix A – technical project plan for migration project 20](#_Toc64441488)

[Appendix B – Pilot Migrations in Mobilize phase 22](#_Toc64441489)

[1 Project Overview 4](#_Toc64441351)

[1.1 Executive summary 4](#_Toc64441352)

[1.2 Business Requirement 4](#_Toc64441354)

[1.3 Pain Points / Challenges in the current environment 5](#_Toc64441355)

[1.4 Project Success Criteria 5](#_Toc64441413)

[1.5 Pre-Requisites 6](#_Toc64441416)

[1.6 Dependencies 6](#_Toc64441417)

[1.7 Assumptions 6](#_Toc64441419)

[1.8 In-scope 7](#_Toc64441420)

[1.9 Out of Scope 7](#_Toc64441421)

[1.10 Risks and Mitigation 9](#_Toc64441422)

[1.11 Raci Matrix 10](#_Toc64441425)

[2 Solution Architecture Diagram 11](#_Toc64441476)

[2.1 Architecture on AWS 11](#_Toc64441477)

[2.2 Overview of the Architecture 12](#_Toc64441482)

[3 Project Execution / summary of milestones & deliverables 14](#_Toc64441483)

[3.1 Expected AWS Cost Breakdown by Services 14](#_Toc64441484)

[3.2 Acceptance 15](#_Toc64441485)

[4 Resources & Cost Estimates 16](#_Toc64441486)

[4.1 Project Sponsor(s) / Stakeholder(s) / Project Team 19](#_Toc64441487)

[Appendix A – technical project plan for migration project 20](#_Toc64441488)

[Appendix B – Pilot Migrations in Mobilize phase 22](#_Toc64441489)

**Disclaimer**

This deck outlines general guidance from AWS on what expectations we have to cover broader base of customer requirements. The intent is to make it easier for APN partners to work on funding requirements and reduce the cycle time. With sample text to refer, this is helpful for partners in building comprehensive SoW (Statement of Work). However, this deck shouldn’t be looked at as an ideal SoW. Sections identified below may not always apply and based on specific customer requirements, the contents of SoW will have to be updated/carved out by the partner team.

Please seek your own legal advice when writing SoW for customers

# Project Overview

## Executive summary

MothersonSumi INfotech & Designs Ltd. (MIND) is a part of Joint venture between **Samvardhana Motherson Group** (SMG) of India and **Sumitomo Wiring Systems** of Japan (SWS).

MIND is a provider of end-to-end software and engineering design solutions to companies around the globe. MIND started as an IT arm of the group in the year 2000 to support the IT needs of Samvardhana Motherson Group and Sumitomo Wiring Systems worldwide. MIND has further ventured into European and American Market to customers who are non-SWS and SMG to expand our services.

MIND's headquarters and development centers are in Noida (near New Delhi), India. MIND is a CMMi Level 5, an ISO 9001:2008 and ISO 27001 certified company. Since its inception in 2000, MIND has emerged as a strong world class IT Company with projects across the globe. MIND has multi-lingual software development capabilities including Japanese and German.

MIND is a Microsoft Gold Certified Partner, AWS, Azure & Google Cloud Service Provider, Oracle GOLD OPN partner and partner with other big IT brands.

MIND has Data Center (Level 3) services, Security Consulting Services Enterprise IT Helpdesk (Multi-lingual), Remote Application Management, Performance Management & Capacity Planning, Network Management Services and Application Hosting. MIND has defined Business Continuity (BC) and Disaster Recovery (DR) plans to mitigate risk of business disruption for its customers.

Maruti Suzuki India Limited (MSIL), a subsidiary of Suzuki Motor Corporation, Japan, is India’s largest passenger car maker. India’s first Company to produce and sell more than a million cars in India in a year, Maruti Suzuki is credited with having ushered in the automobile revolution in the country. The Company, formerly known as Maruti Udyog Limited, was incorporated as a joint venture between the Government of India and Suzuki Motor Corporation, Japan in February 1981.

## Business Requirement

Maruti wants to build a Health Monitoring System for various components/parts of car manufactured to prevent defective parts to pass through to customers. This will help to reduce cost of vehicle call backs and subsequent repair.

This includes ingestion of 20 TB data from different cloud providers to AWS cloud and performing preprocessing on 20 TB once and then 2 TB monthly data. After ingestion to do analytics on data and build machine learning models and storing data in S3.

## Pain Points in the current environment

Challenges faced by the *CUSTOMER* in the current environment include

* Performance bottlenecks during peak hours of the day/week/month/at the time of batch data accessing.
* Mapping of data fields.
* Accessing data remotely from their on-prem application.
* Partitioning of data keeping in mind, the local time zone conversions.

## 















## Project Success Criteria

* Integrity check of the partitioned data by verifying the record count.
* Quicksight dashboard published with insights.
* Deployed API tested from client end using alpha-beta testing process.

## Pre-Requisites

* Architecture diagram, documentation, inventory and performance details of the existing environment will be made available
* AWS Administrator/necessary access to AWS Partner to start and work on the project
* Customer to provide support on understanding the business logic
* Assign an executive to work collaboratively with joint accountability of the program

## Dependencies

* Keys to access data on their on-premise database.
* Network bandwidth requirement for end user connectivity to AWS.
* Integration between API gateway and on-premise application.
* Dependencies from on-premises data center.

## Assumptions

* Billing of AWS services will be handled by MSIL.
* Joint effort is required between MIND & MSIL business mapping
* Quality Checks are limited to the basic checks
* Anything which is not covered as a part of this scope can be taken as a part of data engineer/ ML engineer / data scientist efforts.

## In-scope

### Scope-1 – Data Analysis & Integration - Data Lake Creation

Data accumulation from various sources and do analysis of its usefulness in modelling.

* 1. Integrate & collection of data from all data sources
  2. Setting up pipeline for data ingestion from on premise to AWS Data Lake.

### Scope-2 – Component Health Status Prediction ML Modelling

Create a Component Health Status Prediction Model using a suitable modelling technique, using following data, with selection of various available data

1. IMEI Number
2. Vehicle ID
3. Average Speed
4. Distance
5. Latitude
6. Longitude

### Scope-3 (Monthly) – Monitoring & Governance

1. Monitoring of data ingestion pipeline, data transformation jobs, metadata creation jobs and Data Lake resources
2. Governance for managing IAM users, managing access to AWS services/resources
3. Data Access Policies for data lake
4. Usage monitoring of the resource for cost optimization.
5. Billing alerts/notification.

Note –

1. Anything which is not covered as a part of this scope can be taken as a part of data engineer / ML Engineer / Data Scientist efforts.

### Scope-4 (Need Based) – Data Engineering Activity – New Data Source Setup

1. Building pipeline for ingesting data from new data source.
2. Creating jobs for data transformation and metadata creation.
3. Transforming the data to flat files, encrypting the data and sharing the data with consumers.
4. Monitoring & notification setup for data ingestion pipeline, data transformation jobs and Data Lake resources for the new data source.

\*\*Typical Data engineering role activities are

* Create data repositories
* Identify and implement a data-ingestion solution
* Identify and implement a data-transformation solution
* AWS Services/tools - EMR, Glue, Athena, S3, DMS, Kinesis etc.

### Scope-5 (Need Based) – Data Scientist Activity –

1. Any tasks as per data scientists’ role / activities

\*\*Typical Data Scientist role activities are

* Sanitize and prepare data for modelling
* Perform feature engineering
* Analyze and visualize data for machine learning
* Machine Learning Modelling
* Implementation and Operations

## Out of Scope

Any third-party components deployment or third party software solution configuration

1. Any report generation
2. Any production deployment related activity including DevOps pipeline and infrastructure
3. Procurement of any software, tools or pertinent licenses unless specifically mentioned in this SOW
4. Any upgrades required to other systems to enable them to work with the new setup
5. Training for clients’ team
6. Non-functional requirements like application load testing, benchmarking is responsibility of the customer
7. No SLAs are defined in terms of application up-time, page load time, throughput of Maruti Suzuk is, and availability
8. Issues arising out of DoS attacks (Denial of Service), malware, virus and security related issues. However, these will be attended by partner on a best effort basis and charged on actual efforts
9. Any unplanned changes to the AWS design, new technology stack support, deployment or infrastructure will be out of scope. If and when so desired, these must go under a Change Management process
10. Any licenses / tools cost not specified in this proposal will be customer ’s responsibility
11. Any security / legal / compliance audits

## Risks and Mitigation

|  |  |
| --- | --- |
| **Risk** | **Mitigation** |
| **EMR cluster sizing is not appropriate** | Currently, no performance stats available. Like-to-like sizing from on-premises environment  While executing the task, partner & customer will collect statistics in the existing, and AWS environment  After a month of running in production, sizing of EMR cluster will be re-assessed to optimize for cost and performance. |
| **Single AZ setup** | Educate and sensitize the customer highlighting the potential of business impact; customer owns the risk |
| **Change of architecture** | While moving to production there can be change of the architecture which may lead to change in cost |
| **Stringent timelines, any delay will have a cascading effect** | Requested artifacts should be provided within 2 business days. Also, a dedicated PM from customer would be required for governance activities |
| **Performance bottlenecks impacting overall SLA** | Performance testing to be done by *customer* during the implementation phase with production-like setup in a separate environment.  Observations to be shared with development team for required course corrections. |
| **Lack of support from business, existing partner** | Manage project timelines through regular governance agreed mutually by partner and customer at the time of project initiation. Escalate in timely fashion in case of any issues/risks |
| **Lack of testing assets and tools to validate the implementations** | Customer to provide the input & output for comparison testing from their existing application |
| **Technical issues while executing the migration to AWS** | AWS Business support plan will be purchased |

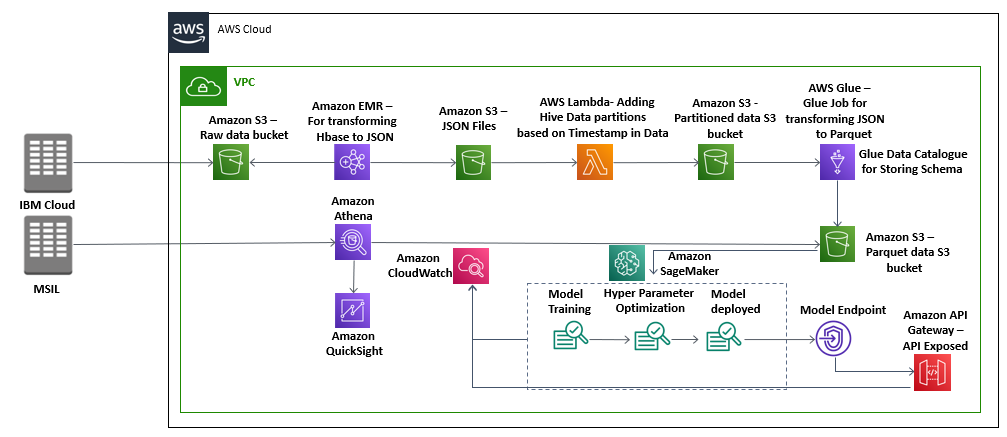
## Raci Matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tasks/Activities** | **Responsible** | **Accountable** | **Consulted** | **Informed** |
| Project initiation & Kick-off | MIND | MARUTI SUZUKI | AWS | MARUTI SUZUKI |
| Infra setup and configuration - Foundation | MIND | MARUTI SUZUKI | AWS | MARUTI SUZUKI |
| Discovery | MIND | MIND | AWS/ MARUTI SUZUKI | AWS/ MARUTI SUZUKI |
| Design | MIND | MIND | AWS/ MARUTI SUZUKI | AWS/ MARUTI SUZUKI |
| Implementation | MIND | MIND | AWS/ MARUTI SUZUKI | AWS/ MARUTI SUZUKI |
| Code Build | MIND | MIND | AWS/ MARUTI SUZUKI | AWS/ MARUTI SUZUKI |
| Code Review | MARUTI SUZUKI | MARUTI SUZUKI | AWS | AWS |
| Validation | MIND | MARUTI SUZUKI | MIND | AWS |
| Sign-off | MARUTI SUZUKI | MARUTI SUZUKI | AWS/ MIND | AWS/ MIND |



# Solution Architecture Diagram

## Architecture on AWS



## Overview of the Architecture

**Step-1**

In this step it shows how the business problem is solved with utmost efficiency by using various techniques to reach the goal. Here Textract is being used which is an AWS service though which we can perform text extraction.

1. The invoices will be uploaded through the front-end through an API call and that will store the documents in a specified S3 bucket
2. The uploading can be a single document or batches of documents.
3. After the documents have been uploaded to S3 bucket a Lambda function will be triggered
4. This Lambda function is for Pre-Processing the document where the document will be refined and then stored to another S3 bucket.
5. After the pre-processed document Is stored in the S3 bucket another Lambda function will be triggered.
6. This triggered lambda function is for Post-Processing of the document.
7. Here the document is processed by calling Textract API functions on the document where the it will be sent to textract for processing and will be getting a json response back from textract.
8. There can be 2 outcomes-
   1. The successful invoices
   2. Flagged invoices
   3. The successful invoices will be stored in S3 as a JSON file.
   4. The flagged invoices will go to other S3 bucket folder.
9. After the json output is uploaded In S3 bucket it then will trigger another lambda function which will take this json output and save it in DynamoDB.
10. After saving the data in dynamodb following information will be displayed on the screen-
    1. Header information: Supplier Name, Invoice No., Invoice Date, GSTN no.
    2. Footer Information: Taxes & Invoice Amount
    3. Line level: Item level information like part no., qty & amount
11. After this we will have to send data from dynamodb to an ERP and for that we have few options-
    1. API option
    2. DMS
    3. Python script from SQL
    4. CSV import and export
12. Here we chose API option so whenever anyone hits the API a lambda function will create a DB request which will pull the data from DymanoDB and provide it to the requested person.

**Step-2- Augmented AI functionality**

After post-processing of invoice we can get 2 outcomes-

* + The successful invoices/ data properly extracted
  + Flagged invoices/ not properly extracted

Flagged invoices will be rectified using A2I which is a service build by AWS.

A2I starts as a human loop which works as a correction mechanism for flagged invoices.

1. The output that will be given by A2I will be saved into a S3 bucket which then will trigger another lambda function.
2. This lambda function will parse A2I output to readable json format as it contains nested json with other unwanted data
3. The data will be stored in the same S3 bucket where the success invoice outputs are stored for further process

**Step-3-enabling different invoice templates**

Creating a solution for different kinds of invoice templates.

Will analyse meaningful headers and contents to be extracted from the invoices.

Will change/manipulate the headers being used in old code.

# Project Execution / summary of milestones & deliverables

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Scope-1 Schedule** | | | | | | | | | |
| **Activity** | **Wk1** | **Wk2** | **Wk3** | **Wk4** | **Wk5** | **Wk6** | **Wk7** | | |
| Setup of AWS Account |  |  |  |  |  |  |  | | |
| Create IAM Roles and Users |  |  |  |  |  |  |  | | |
| Setup S3 Bucket for creating Data Lake |  |  |  |  |  |  |  | | |
| Configure pipeline to ingest data from IBM to Data Lake |  |  |  |  |  |  |  | | |
| Create EMR Cluster to transform data to CSV |  |  |  |  |  |  |  | | |
| Create and configure Lambda to store the data in hive partitions in S3. |  |  |  |  |  |  |  | | |
| Create Glue job to convert data to Parquet format. |  |  |  |  |  |  |  | | |
| **Activity** | **Wk8** | **Wk9** | **Wk10** | **Wk11** | **Wk12** | **Wk13** | **Wk14** | **Wk15** | **Wk16** |
| Primary Data Exploration. |  |  |  |  |  |  |  |  |  |
| Clearing Issues, observations, Queries, Pending items |  |  |  |  |  |  |  |  |  |
| Business Process mapping / Data Mapping to critical features for component health status prediction |  |  |  |  |  |  |  |  |  |
| Data Exploration - with Selected key Features |  |  |  |  |  |  |  |  |  |
| ML Modelling –Multiple Iterations – Train, Test, Accuracy |  |  |  |  |  |  |  |  |  |
| Comparison with Manual Forecasting |  |  |  |  |  |  |  |  |  |
| UAT |  |  |  |  |  |  |  |  |  |

**\*Wk – Week**

**For Scope-2: (applicable after end of scope-2)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Monitoring & Governance** | **M1** | **M2** | **M3** | **M4** | **M5** | **M6** |
| **Monitoring & Governance Activities** |  |  |  |  |  |  |

**\*M – Month**

## Expected AWS Cost Breakdown by Services

The monthly estiMaruti Suzuki for this project can be reviewed by following the below link:

[https://calculator.aws/#/estimate?id=aeb39c68ff97b68f52caaff444003d94f6e9128e](https://calculator.aws/" \l "/estimate?id=aeb39c68ff97b68f52caaff444003d94f6e9128e" \o "https://calculator.aws/#/estimate?id=aeb39c68ff97b68f52caaff444003d94f6e9128e" \t "_blank)

## Acceptance

*[To conclude a project, define acceptance process here. For example:*

*Upon completion of a Phase, PROVIDER will submit the associated tangible Deliverables, to CUSTOMER accompanied by an Acceptance Form in the form set forth in Appendix B to this SOW. Upon such submission, CUSTOMER will review, evaluate and/or test, as the case may be, the applicable Deliverable(s) within eight (8) business days (the “Acceptance Period”) to determine whether or not each Deliverable(s) satisfies the acceptance criteria for the particular Deliverable in all Maruti Suzukirial respects. If the Deliverable satisfies its acceptance criteria in all Maruti Suzukirial respects, CUSTOMER will furnish a written acceptance confirmation to PROVIDER via the Acceptance Form prior to the end of the Acceptance Period. For a Deliverable that is not accepted due to a non-conformity or defect, CUSTOMER will indicate the detailed reasons for such rejection on the Acceptance Form and return the Acceptance Form together with the associated tangible rejected Deliverables, if any, to PROVIDER (a “Rejection Notice”) within the Acceptance Period. Upon receipt of a Rejection Notice, PROVIDER will promptly correct any defects or non-conformities to the extent required so that each Deliverable satisfies the requirements of this SOW and its acceptance criteria in all Maruti Suzukirial respects. Thereafter, PROVIDER will resubmit a modified Deliverable to CUSTOMER , accompanied by the Acceptance Form and the process set forth above will be repeated. However, CUSTOMER will limit its review, evaluation and/or test of each resubmitted Deliverable to determining whether or not PROVIDER has corrected the defects or non-conformities identified in the Rejection Notice and to the effects or impact which PROVIDER’s corrections or modifications have on other Deliverables or other portions of the same Deliverable. If CUSTOMER fails to provide PROVIDER with the above described Rejection Notice prior to the end of the applicable Acceptance Period, then the corresponding Deliverable(s) are deemed accepted.]*

Typical deliverables at the end of the engagement are Standard Operating Procedures, Build document. However, what the customer desires for acceptance needs to be discussed and agreed upon before beginning of the engagement

# Resources & Cost EstiMaruti Suzukis

*[List all billable and non-billable resources involved in the project]*

*APN partner are required to ensure Project Plan and the Work Break Down list is comprehensively charted out. Each task should be broken down in to as much details as possible and efforts listed down should be justifiable*

Partner Technical Team

1. Title - Name
2. Title - Name

|  |  |
| --- | --- |
| Resource | Rate (USD) / Hour |
| Solution Architects |  |
| Engineers |  |
| Other (Please specify) |  |

|  |  |  |
| --- | --- | --- |
| Project Plan and Work Break Down List | | |
| Infrastructure Creation and Implementation | | |
| Common Infrastructure & Activities | | |
| Roles | **Sub task** | **Effort required (person days)** |
| Technical Architect | Discuss the Application and Infrastructure Architecture. Understand Dependencies and Integration points |  |
| Create Document, Reviews from Customer, Corrections and Document sign off |
| Sr. Cloud Engineer | AWS Account Creation / Setup or Gain access if existing A/c and IAM (Roles, Policies, Groups and Users) Access Setup |  |
| Setup of Cloud Trail & Billing with their S3 Buckets |
| Setup Network components like VPC, OpenVPN, Subnets, Routing Tables, NAT, Bastion/RDP GW etc. as per the architecture |
| Setup of NACL's & Security Groups and configuration of security rules as per the document. |
| Setup of S3 Buckets |
| Setup of Base AMI's (App / Layer wise) with latest OS patches & software's required by the applications. |  |
|  | **Total** |  |
| Kubernetes Architecture | | |
| Roles | **Sub task** | **Effort required (person days)** |
| Sr. Cloud Engineer | Setup and configuration of the Kubernetes cluster |  |
| Sr. Cloud Engineer | Setup and configuration of the Worker Nodes |  |
| Sr. Cloud Engineer | Configuration of AutoScaler on Kubernetes cluster |  |
| Sr. Cloud Engineer | Setup and configuration of Load Balancer Ingress Controller |  |
| Sr. Cloud Engineer | Setup of Cluster Level monitoring using Open Source tools |  |
| Sr. Cloud Engineer | Setup of ElasticCache service |  |
| Sr. Cloud Engineer | Setup Client Build Environment with Fileshare |  |
| Sr. Cloud Engineer | Setup HashiCorp Vault and configure KMS |  |
| Sr. Cloud Engineer | Setup ALB and WAF then configure it to route requests to servers |  |
| Sr. Cloud Engineer | Setup RDS with MySQL |  |

|  |  |  |
| --- | --- | --- |
| Sr. Cloud Engineer | Support to customer on application setup.  (Deployment server in case of Web/App/Api) |  |
|  | **Total** |  |
|  | **Data Migration (for all 7 customers)** |  |
| Roles | **Sub task** | **Effort required (person days)** |
| Sr. Cloud Engineer | Migrate Data from Cloud SQL to RDS |  |
| Sr. Cloud Engineer | Migrate Data from Cloud storage to S3 |  |
|  | **Total** |  |
|  | **CI/ CD** |  |
| Roles | **Sub task** | **Effort required (person days)** |
| Sr. Cloud Engineer | Setup and configure Jenkins server for the application deployment |  |
| Sr. Cloud Engineer | Setup Jobs for various services |  |
| Sr. Cloud Engineer | Take care of roll back in case of failures |  |
| Sr. Cloud Engineer | Test the entire solution end to end |  |
| Sr. Cloud Engineer | Walk through of the entire branching and workflow of DevOps setup |  |
| Sr. Cloud Engineer | Corrections on feedback |  |
|  | **Total** |  |
|  |  |  |
|  | **For v5/6 Architecture (For all 7 customers)** |  |
| Roles | **Sub task** | **Effort required (person days)** |
| Sr. Cloud Engineer | Setup of EC2 instances according to architecture |  |
| Sr. Cloud Engineer | Setup ALB and configure it to route requests to servers |  |
| Sr. Cloud Engineer | Setup RDS with MySQL |  |
| Sr. Cloud Engineer | Support to customer on application setup. (Deployment server in case of Web/App/Api) |  |
| Sr. Cloud Engineer | Cloud Watch Alerts and Setup of Standard / Custom Metrics - ASG Metrics |  |
| Cloud Watch Dashboards Setup |
| Sr. Cloud Engineer | Configure GuardDuty and AWS Config |  |
| Sr. Cloud Engineer | Handover & Documentation. |  |
|  | **Total** |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | **Data Migration (For all 7 customers)** |  |
| Roles | **Sub task** | **Effort required (person days)** |
| Sr. Cloud Engineer | Migrate Data from Onpremise/Cloud to RDS |  |
| Sr. Cloud Engineer | Migrate Data from Onpremise/Cloud storage to S3 |  |
|  | **Total** |  |
| Project Management | | |
| Roles | **Sub task** | **Effort required (person days)** |
| Project Management | Project will conduct governance with Technical Team and Customer Stakeholders |  |
|  | **Total** |  |



Cost Contribution distribution between Partner, *CUSTOMER*, AWS:

|  |  |  |
| --- | --- | --- |
| Party | Contribution (USD) | % Contribution of Total |
| *CUSTOMER* |  |  |
| Partner |  |  |
| AWS |  |  |

## 

*CUSTOMER*

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# Appendix A – technical project plan for migration project

Migration Project Plan must demonstrate a consistent methodology and process applied through multiple migration phases as exemplified below. Though specific details may vary from project to project, a solid migration framework with major phases and work areas must be clearly identified and exercised consistently across all the projects.

The migration work scope and deliverables below are for guidance and demonstration purposes. Refer to AWS [Migration Consulting Competency Checklist](https://partnercentral.awspartner.com/sfc/#version?selectedDocumentId=0690h000003pc7y) for details. Actual project details may include other work items not limited to the following areas.

|  |  |
| --- | --- |
| Phase | Work Area and Deliverables |
| Assess | Migration Readiness Assessment (MRA) -  MRA determines *CUSTOMER* ’s readiness based on [AWS Cloud Adoption Framework](https://aws.amazon.com/professional-services/CAF/) comprised of Business, People, Governance, Platform, Security, and Operations perspectives.  Deliverables should include an assessment report with suggested actions and Statement of Work for executing the Mobilize phase next.  Total Cost of Ownership (TCO)–  The purpose of TCO analysis at assess phase is to perform rapid discovery and create TCO report.  Deliverables should include a detailed business case with focus on TCO modelling, business value assessment and detailed migration cost. |
| Mobilize | The purpose of Mobilize phase is to validate foundational migration capability and business case and plan migration project next. Mobilize may consist of the following work streams:   * Mobilize phase execution and migration planning * Portfolio discovery and analysis * Operations model assessment and design * Landing Zone design * Initial implementation Security specification * Migration pilot implementation as per [Appendix B](#_Appendix_B_–) * Migration team/organization establishment, Team RACI, training plan, and training activities * Detailed Business Case justification supported by Total Cost of Ownership (TCO) calculations   Deliverables should reflect the results of each of the work streams. |
| Migrate & Modernize | Migrate -  The migration project may consist of the following work areas on a per application basis:   * Design: migration pattern, application architecture, operations, cutover plan and process, reusable templates, migration tooling, and validation test plan * Migration: servers, databases, data, infrastructure services, followed by basic validation test * Integration: connectivity, application interfaces, operations (backup/restore, ...) * Validation: functional, performance, reliability, security, compliance * Cutover: meeting RTO and RPO with rollback plan   Modernize –  Modernization may be performed as part of migration work or post migration. The modernization project may consist of following areas on a per application basis:   * Assessment:Figure out motivation driver to modernize an application. Assess the amount of effort, time and cost to modernize. Assess the cost savings with modernization options (replatform, refactor, repurchase etc.) * Design: Migration pattern, target application architecture and AWS Services, operations, cutover plan and process, migration tooling and validation test plan. * Development: Develop or modify application to use AWS managed platform. * Validation: functional, performance, reliability, security, compliance * Cutover: meeting RTO and RPO with rollback plan   Optimize -  It may involve one or more of the following work areas.   * Cost optimization (e.g., right-sizing services, resource reservation, leveraging spot instance, monitoring and analyzing service usage and cost) * Application optimization (e.g., performance, functional, design) * Process optimization (e.g., development process automation) * Operational optimization (e.g., operations support systems, infrastructure as code))   Deliverables should reflect the results of each of the work areas. |
| Completion | Reach the project closure with the *CUSTOMER* .  Deliverables should include *CUSTOMER* ’s acceptance letter and training Maruti Suzukirials. |

# Appendix B – Pilot Migrations in Mobilize phase

The Application Migrations work stream defines an agile approach to migrate applications to AWS during the Mobilize Phase. This work stream helps *CUSTOMER* s get hands-on experience in migrating different types of applications to AWS using standard migration tools and process, working together with AWS and/or Partner migration experts. It also helps bring some of the outputs from other work streams, such as Security, Risk & Compliance, Operational Integration, and Landing Zone together through live-migrations.Although the number of applications migrated in this work stream is normally no more than 10, it is largely indicative of how the majority of applications can be migrated as detailed discovery and analysis is completed for the rest of the portfolio.

**General Guidelines for selecting applications for Pilot**

Applications that are web-based (accessed via web browsers), 2 or 3tiered (web-app-database); running a supported operating systemon virtual or physical hardware; have no dependency (or are loosely coupled) on other applications in data center/on-prem; have little (less than 1 Mbps) connectivity needs back to data center or *CUSTOMER* has Direct Connect; no shared data storage (SAN/NAS) with other applications; runs on AWS RDS supported databases ; Database size less than 20GB; not to exceed 20 server instances; preferably, stateless-architecture (can be deployed in a clustered mode using load balancer); preferably, at least 50% test automation for expedited testing/certification; preferably, well understood and documented architecture; acceptable (less than 2 hours) downtime.

In addition to above guidelines, there are other factors to be considered based on *CUSTOMER* ’s process, application criticality, commitment, SME availability.

General Examples of applications preferred for Pilot Migrations:

Online properties/Marketing sites

Intranet applications built on n-tiered architecture

Content Management Systems

Web Applications

Marketing, Sales and Service applications

General Examples of applications not recommended for Pilot Migrations:

ERPs and CRMs– SAP, PeopleSoft, Oracle ERP, Microsoft Dynamics, Seibel

Financial Reporting Systems

Data Warehouse

Information Lifecycle Management, ETL, B2B data exchanges,

EAI and middleware

Citrix-based workloads

**Outcome**

*CUSTOMER* resources trained in migration tools, AWS services, monitoring, and best-practices

*CUSTOMER* ramped-up on scalable migration factory framework