

<Customer Name>

Review

##### Prepared by: <consultant>

##### Date: dd/mm/yyyy

Table of Contents

[1 High Level Overview 4](#_Toc496256648)

[1.1 Background 4](#_Toc496256649)

[1.2 Objectives 4](#_Toc496256650)

[1.2.1 Out Of Scope 4](#_Toc496256651)

[1.3 Document Organization 4](#_Toc496256652)

[1.4 Summary of Findings 5](#_Toc496256653)

[2 Performance Review 6](#_Toc496256654)

[2.1 Performance and Stress Testing Practices 6](#_Toc496256655)

[2.1.1 Recommendations 6](#_Toc496256656)

[2.2 Networking Practices 7](#_Toc496256657)

[2.2.1 Recommendations 7](#_Toc496256658)

[2.3 Logging and Audit Practices 7](#_Toc496256659)

[2.3.1 Recommendations 7](#_Toc496256660)

[2.4 Caching Practices 7](#_Toc496256661)

[2.4.1 Recommendations 7](#_Toc496256662)

[3 Application Architecture 8](#_Toc496256663)

[3.1 Architecture Principles 8](#_Toc496256664)

[3.1.1 Recommendations 8](#_Toc496256665)

[3.2 Integration Patterns 8](#_Toc496256666)

[3.2.1 Recommendations 8](#_Toc496256667)

[3.3 Service Reliability 9](#_Toc496256668)

[3.3.1 Recommendations 9](#_Toc496256669)

[3.4 Common Services/Re-use 9](#_Toc496256670)

[3.4.1 Recommendations 9](#_Toc496256671)

[3.5 Services Registry/Service Portal 9](#_Toc496256672)

[3.5.1 Recommendations 10](#_Toc496256673)

[4 Technical Architecture 11](#_Toc496256674)

[4.1 Runtime Deployment Architecture 11](#_Toc496256675)

[4.1.1 Recommendations 11](#_Toc496256676)

[4.2 Application Deployment 11](#_Toc496256677)

[4.2.1 Recommendations 11](#_Toc496256678)

[4.3 High Availability 11](#_Toc496256679)

[4.4 Disaster Recovery 11](#_Toc496256680)

[4.4.1 Recommendations 12](#_Toc496256681)

[4.5 Scalability and Capacity 12](#_Toc496256682)

[4.5.1 Recommendations 12](#_Toc496256683)

[4.6 Monitoring 13](#_Toc496256684)

[4.6.1 Recommendations 13](#_Toc496256685)

[4.7 DevOps 13](#_Toc496256686)

[4.7.1 Recommendations 13](#_Toc496256687)

[4.8 Security 13](#_Toc496256688)

[4.8.1 Recommendations 13](#_Toc496256689)

[5 Code Review 15](#_Toc496256690)

[5.1 Groovy Component 15](#_Toc496256691)

[5.1.1 Concern 15](#_Toc496256692)

[5.1.2 Recommendation 15](#_Toc496256693)

[5.1.3 Context 15](#_Toc496256694)

[5.2 Synchronous Until Successful and HTTP Outbound 15](#_Toc496256695)

[5.2.1 Concern 15](#_Toc496256696)

[5.2.2 Recommendation 15](#_Toc496256697)

[5.2.3 Context 15](#_Toc496256698)

[5.3 Caching Comarch Token in separate API 16](#_Toc496256699)

[5.3.1 Concern 16](#_Toc496256700)

[5.3.2 Recommendation 16](#_Toc496256701)

[5.3.3 Context 16](#_Toc496256702)

[Appendix A Deployment Architecture 17](#_Toc496256703)

[CloudHub Networking 17](#_Toc496256704)

[Appendix B Performance Review 18](#_Toc496256705)

[Caching Strategy Reference Design 18](#_Toc496256706)

[In-Memory Object Store 18](#_Toc496256707)

[Persistent object store – “\_defaultUserObjectStore” 18](#_Toc496256708)

[Custom Solution (Preferred Solution for Token retrieval) 18](#_Toc496256709)

[Description… 18](#_Toc496256710)

[Reference Solution Design 18](#_Toc496256711)

[Appendix C Target Technical Architecture 19](#_Toc496256712)

[Continuous Integration 19](#_Toc496256713)

[Deploy Mule Application via Maven Plugin 19](#_Toc496256714)

**Sign Off**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Name** | **Function** | **Company** | Signature |
| dd/mm/yyyy | <full name> | <e.g. Architect> | <company> |  |

**Document References**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Document Name** | **Document Title** | **Author** |
|  |  |  |  |
|  |  |  |  |

**Document Versioning**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Author** | **Version** | **Change Description** |
| dd/mm/yy | Full name | 0.1 | e.g. Initial version. |
|  |  |  |  |
|  |  |  |  |

# High Level Overview

## Background

MuleSoft were engaged by <customer> to conduct a combined architecture and code review as part of the <name> project.

Describe the background.

## Objectives

E.g. The key objective of the review was to assess the implementation of the MuleSoft Anypoint Platform (Technical Architecture) and the core integration patterns and code (Application Architecture). It focused on the <objective>, but the recommendations also aim to provide a set of best practices in readiness for go-live with the future strategic initiatives.

### Out Of Scope

… add out of scope….

## Document Organization

The document has been broken into 4 key areas that were the focal points of the review:

|  |  |
| --- | --- |
| Section | Description |
| Performance Review | Focused on improving the performance and minimise overhead / risks of performance issues in the future. |
| Application Architecture | Focused on the core integration patterns employed to support the <project name>. |
| Technical Architecture | Focused on the current setup and deployment of the MuleSoft Anypoint Platform. |
| Code Review | Focused on the code review including recommendations on improving efficiency and maintainability of the system. |

The table below has been used to highlight the assessment status of each reviewed item in a visual manner.

|  |  |
| --- | --- |
| Status | Description |
|  | The component/subject reviewed does **not need any further attention** |
|  | The component/subject reviewed could benefit from attention but does not pose an immediate risk to the project go-live. |
|  | The component/subject reviewed should be considered for more immediate attention and poses some risk to the project go-live. |
|  | The component/subject reviews needs **immediate attention.** |

## Summary of Findings

|  |  |  |  |
| --- | --- | --- | --- |
| Status | Review Section | Comments |  |
|  | Performance Review | e.g. Performance and reliability of the MuleSoft platform and related MuleSoft applications was a core consideration during the review.  Observations indicated that … so and so. | |
|  | Application Architecture | E.g. The overall application architecture was <add your high level recommendation> | |
|  | Technical Architecture | E.g. The technical architecture <add your high level recommendation> | |
|  | Code Review | E.g. the code review was closely aligned to the Performance Review and Application Architecture. Several best practices have been recommended to improve the performance of MuleSoft applications. <… add your observations, high level> | |

# Performance Review

E.g. The Performance Review was focused on providing the best practices to optimize the MuleSoft processes even further and support the scalability of the solution in the long term.

## Performance and Stress Testing Practices

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | <give details> | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend… | |
| Should  ✓ | Recommend… | |
| Should  ✓ | Recommend… | |

## Networking Practices

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Description… | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend… | |

## Logging and Audit Practices

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Observation …. | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend…. | |
| Should  ✓ | Recommend… | |

## Caching Practices

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Observation….. | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend…. | |

# Application Architecture

E.g. The application architecture review was focused on high-level architecture principles and practices to support the integration platform.

## Architecture Principles

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | e.g. the integration principles met industry practices although some additional principles could be added for completeness. | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend…. | |
| Should  ✓ | Recommend…. | |

## Integration Patterns

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | e.g. the integration patterns satisfy the requirements of the current solution, but they require further improvements to maximise reusability, scalability, and maintainability. | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend… | |
| Should  ✓ | Recommend… | |

## Service Reliability

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | e.g. reliability patterns have been well considered and throughout the solution. However, some observations were made on the implementation of reliability within the flows. | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Could  ✓ | Recommend… | |
| Could  ✓ | Recommend… | |

## Common Services/Re-use

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | e.g. several re-usable services have been developed. However, some common logic is repeatedly coded into flows that now needs to be maintained in multiple places. This may later impact maintainability. | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend… | |
| Should  ✓ | Recommend… | |
| Could  ✓ | Recommend… | |

## Services Registry/Service Portal

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Details… | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Must  ✓ | Recommend… | |
| Should  ✓ | Recommend… | |
| Could  ✓ | Recommend… | |

# Technical Architecture

E.g. The technical architecture review was focused on the setup and configuration of the Anypoint MuleSoft platform components including the non-functional requirements such as Availability, Scalability, Monitoring and Security.

## Runtime Deployment Architecture

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Observations… | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Could  ✓ | Recommend… | |

## Application Deployment

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Observations…. | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend… | |

## High Availability

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Observations… | |

## Disaster Recovery

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Recommend… | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend… | |

## Scalability and Capacity

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Observation…. | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend… | |
| Should  ✓ | Recommend… | |
| Should  ✓ | Recommend… | |

## Monitoring

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Observation…. | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Could  ✓ | Recommend… | |
| Could  ✓ | Recommend… | |

## DevOps

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Observation… | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Should  ✓ | Recommend… | |
| Could  ✓ | Recommend… | |

## Security

|  |  |  |
| --- | --- | --- |
| Status | Details |  |
|  | Observation… | |

### Recommendations

|  |  |  |
| --- | --- | --- |
| Priority | Details |  |
| Must  ✓ | Recommend… | |
| Should  ✓ | Recommend… | |
| Could  ✓ | Recommend… | |

# Code Review

## Groovy Component

### Concern

Description….

### Recommendation

Recommend…

### Context

Description…

## Synchronous Until Successful and HTTP Outbound

### Concern

Description…

### Recommendation

Recommend…

### Context

Description…

## Caching Comarch Token in separate API

### Concern

Description…

### Recommendation

Recommend…

### Context

Description…

1. Deployment Architecture

## CloudHub Networking

Description / Diagrams….

1. Performance Review

## Caching Strategy Reference Design

Description…

### In-Memory Object Store

Advantage: Description…

Disadvantage: Description…

### Persistent object store – “\_defaultUserObjectStore”

Advantage: Description…

Disadvantage: Description…

### Custom Solution (Preferred Solution for Token retrieval)

### Description…

### Reference Solution Design

Description…

1. Target Technical Architecture

## Continuous Integration

E.g. the following diagram illustrates the reference architecture for the CI process to implement at <customer>. The architecture includes all the MuleSoft best practices in this space and can be extended or amended based on the specific customer’s requirements.

<add diagram here>

E.g. A **Continuous Integration** workflow is triggered when so and so….

**E.g. Release Manager** can approve deployment for a specific target environment… so and so….

### Deploy Mule Application via Maven Plugin

E.g. Application deploy can be performed form the CI server using a separate pom.xml file that is configured with the [Mule Plugin for Maven](https://docs.mulesoft.com/mule-user-guide/v/3.8/mule-maven-plugin#cloudhub).

The following snippet shows how to deploy a Mule application using the pom.xml file provided with this review

< add code snippet here…>