

department of veteran’s affairs

api developer playbook

A picture containing clipart

Description generated with very high confidence

**API Developer Playbook Blueprint Documentation**

**Maps to Framework Set #1**

**Version 0.1**

**May 30, 2018**

**Prepared by:**

**Jordan Braunstein**

**Max Girin**

****

****

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Name** | **Version** | **Description** |
| 05/30/2018 | Jordan Braunstein  Max Girin | 0.1 | DRAFT version of Outline distributed for ASG Review |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

[1. Introduction 6](#_Toc515383085)

[1.1 Purpose and Use 6](#_Toc515383086)

[1.2 VA API Vision 6](#_Toc515383087)

[1.3 Vendor Agnostic 6](#_Toc515383088)

[1.4 MuleSoft Applicability 6](#_Toc515383089)

[2. Standards 6](#_Toc515383090)

[2.1 Development Lifecycle 6](#_Toc515383091)

[2.2 Industry Standards 6](#_Toc515383092)

[2.2.1 FHIR 6](#_Toc515383093)

[2.2.1.1 Argonaut 6](#_Toc515383094)

[3. API Layers 6](#_Toc515383095)

[3.1 Loose Coupling 6](#_Toc515383096)

[3.2 Policy Impacts 6](#_Toc515383097)

[3.3 API Management Impacts 6](#_Toc515383098)

[3.4 Layer Responsibilities 6](#_Toc515383099)

[3.4.1 Experience 6](#_Toc515383100)

[3.4.2 Process 6](#_Toc515383101)

[3.4.3 System 6](#_Toc515383102)

[3.4.3.1 Native System API 6](#_Toc515383103)

[3.4.4 Decision Tree 6](#_Toc515383104)

[4. API Contracts 7](#_Toc515383105)

[4.1 Design First Methodology 7](#_Toc515383106)

[4.2 API Contract Standards 7](#_Toc515383107)

[4.2.1. API Specifications 7](#_Toc515383108)

[4.2.2 Swagger and RAML 7](#_Toc515383109)

[4.2.2.1 Conversions 7](#_Toc515383110)

[4.3 MuleSoft Designer Studio 7](#_Toc515383111)

[5. Naming Conventions 7](#_Toc515383112)

[5.1 API Naming Standards 7](#_Toc515383113)

[5.1.1 Alignment with FHIR and Argonaut 7](#_Toc515383114)

[5.1.2 Alignment with Version Control 7](#_Toc515383115)

[5.2 Environment Properties 7](#_Toc515383116)

[5.3 Configuration Parameters 7](#_Toc515383117)

[5.4 API REST and SOAP Standards 7](#_Toc515383118)

[5.4.1 RESTful URI Naming Standards 7](#_Toc515383119)

[5.4.2 SOAP Message XML Standards 7](#_Toc515383120)

[5.4.3 JSON Schema Naming Standards 7](#_Toc515383121)

[5.4 Objects and Methods 7](#_Toc515383122)

[5.4.1 Canonical Naming 7](#_Toc515383123)

[5.5 Middleware Artifacts (flows, sub-flows, transformations, variables) 7](#_Toc515383124)

[6. Configuration Management 7](#_Toc515383125)

[6.1 API Configuration Management 7](#_Toc515383126)

[6.1.1 Version Numbers 7](#_Toc515383127)

[6.1.2 GitHub Integration 7](#_Toc515383128)

[6.1.3 Branching and Merging 7](#_Toc515383129)

[6.2 Environment Configuration 7](#_Toc515383130)

[7. Unit Testing 8](#_Toc515383131)

[7.1 Unit Testing Approach 8](#_Toc515383132)

[7.2 Agnostic Unit Testing 8](#_Toc515383133)

[7.3 MuleSoft MUnit 8](#_Toc515383134)

[7.3.1 Mocking Features 8](#_Toc515383135)

[7.4 Integration with Continuous Integration Continuous Deployment 8](#_Toc515383136)

[8. Developer Setup 8](#_Toc515383137)

[8.1 MuleSoft Developer Setup 8](#_Toc515383138)

[8.1.1 Maven 8](#_Toc515383139)

[8.1.2 GitHub Plugin 8](#_Toc515383140)

[8.1.3 Munit Setup 8](#_Toc515383141)

[9. Exception Handling 8](#_Toc515383142)

[9.1 Exception Handling Approach 8](#_Toc515383143)

[9.1.1 Transactions vs. non-Transactions 8](#_Toc515383144)

[9.2 Global Exceptions 8](#_Toc515383145)

[9.3 API-Level Exceptions 8](#_Toc515383146)

[9.4 Middleware-Level Exceptions 8](#_Toc515383147)

[9.5 Message Process Failures and Retries 8](#_Toc515383148)

[9.5.1 Message Retries 8](#_Toc515383149)

[9.5.2 Handling Message Failures 8](#_Toc515383150)

[9.6 Exception Logging 8](#_Toc515383151)

[9.7 Notification and Alerting 8](#_Toc515383152)

[9.8 Integration with Queuing 8](#_Toc515383153)

[10. Baseline Logging 9](#_Toc515383154)

[10.1 Logging Framework Used within Each API 9](#_Toc515383155)

[10.2 Logged API Messages 9](#_Toc515383156)

[10.3 API Audit-Level Logging 9](#_Toc515383157)

[10.4 Integration with Third-Party Log Aggregators 9](#_Toc515383158)

[11. Baseline API Security 9](#_Toc515383159)

[11.1 VA Standards 9](#_Toc515383160)

[11.2 OAuth 2.0 and OpenID Connect Security 9](#_Toc515383161)

[11.3 API Security Guidelines and Best Practices 9](#_Toc515383162)

[12. Re-Usability and Shared Components 9](#_Toc515383163)

[12.1 What Constitutes Reusable Artifacts 9](#_Toc515383164)

[12.2 How to Create Shared Components 9](#_Toc515383165)

[12.3 Reuse Guidelines 9](#_Toc515383166)

[12.4 Deploying and Integrating Shared Components into an API 9](#_Toc515383167)

[13. API Documentation Blueprint 9](#_Toc515383168)

[13.1 Core API Design Document 9](#_Toc515383169)

[14. References 10](#_Toc515383170)

[15. Acronym 10](#_Toc515383171)

Table of Figures

Table of Tables

[Table 1: Reference Table 8](#_Toc515341204)

[Table 1: Acronym Table 8](#_Toc515341205)

# 1. Introduction

## 1.1 Purpose and Use

## 1.2 VA API Vision

## 1.3 Vendor Agnostic

## 1.4 MuleSoft Applicability

# 2. Standards

## 2.1 Development Lifecycle

## 2.2 Industry Standards

### 2.2.1 FHIR

#### 2.2.1.1 Argonaut

# 3. API Layers

## 3.1 Loose Coupling

## 3.2 Policy Impacts

## 3.3 API Management Impacts

## 3.4 Layer Responsibilities

### 3.4.1 Experience

### 3.4.2 Process

### 3.4.3 System

#### 3.4.3.1 Native System API

### 3.4.4 Decision Tree

# 4. API Contracts

## 4.1 Design First Methodology

## 4.2 API Contract Standards

### 4.2.1. API Specifications

## 4.2.2 Swagger and RAML

#### 4.2.2.1 Conversions

## 4.3 MuleSoft Designer Studio

# 5. Naming Conventions

## 5.1 API Naming Standards

### 5.1.1 Alignment with FHIR and Argonaut

## 5.1.2 Alignment with Version Control

## 5.2 Environment Properties

## 5.3 Configuration Parameters

## 5.4 API REST and SOAP Standards

### 5.4.1 RESTful URI Naming Standards

### 5.4.2 SOAP Message XML Standards

### 5.4.3 JSON Schema Naming Standards

## 5.4 Objects and Methods

### 5.4.1 Canonical Naming

## 5.5 Middleware Artifacts (flows, sub-flows, transformations, variables)

# 6. Configuration Management

## 6.1 API Configuration Management

### 6.1.1 Version Numbers

### 6.1.2 GitHub Integration

#### 6.1.3 Branching and Merging

## 6.2 Environment Configuration

# 7. Unit Testing

## 7.1 Unit Testing Approach

## 7.2 Agnostic Unit Testing

## 7.3 MuleSoft MUnit

### 7.3.1 Mocking Features

## 7.4 Integration with Continuous Integration Continuous Deployment

# 8. Developer Setup

## 8.1 MuleSoft Developer Setup

### 8.1.1 Maven

### 8.1.2 GitHub Plugin

### 8.1.3 Munit Setup

# 9. Exception Handling

## 9.1 Exception Handling Approach

### 9.1.1 Transactions vs. non-Transactions

## 9.2 Global Exceptions

## 9.3 API-Level Exceptions

## 9.4 Middleware-Level Exceptions

## 9.5 Message Process Failures and Retries

## 9.5.1 Message Retries

## 9.5.2 Handling Message Failures

## 9.6 Exception Logging

## 9.7 Notification and Alerting

## 9.8 Integration with Queuing

# 10. Baseline Logging

## 10.1 Logging Framework Used within Each API

## 10.2 Logged API Messages

## 10.3 API Audit-Level Logging

## 10.4 Container Logging

## 10.5 Integration with Third-Party Log Aggregators

# 11. Baseline API Security

## 11.1 VA Standards

## 11.2 OAuth 2.0 and OpenID Connect Security

## 11.3 API Security Guidelines and Best Practices

# 12. Re-Usability and Shared Components

## 12.1 What Constitutes Reusable Artifacts

## 12.2 How to Create Shared Components

## 12.3 Reuse Guidelines

## 12.4 Deploying and Integrating Shared Components into an API

# 13. API Documentation Blueprint

## 13.1 Core API Design Document

# 14. References

Table : Reference Table

|  |  |  |  |
| --- | --- | --- | --- |
| Ref. ID | Reference Name | Reference | Definition |
| R1 | FHIR Standard | http://www.fhir.org/ |  |
| R2 | HTTPS Standard | <https://https.cio.gov/everything/> | All web traffic shall be HTTPS and none shall be HTTP |
|  |  |  |  |
|  |  |  |  |

# 15. Acronym

Table 2: Acronym Table

|  |  |  |
| --- | --- | --- |
| Acronym | Elaboration | Definition |
| API | Application Programming Interface | A computer function that performs a logical operation on rules or data. |
| SDK | Software Development Kit | A collection of software modules and assets that allows a consumer to apply the modules for runtime functionality. |
| ROI | Return on Investment | The benefit or payoff received from investing cost into an initiative |
| RAML | RESTful API Modeling Language | A language and approach to building API soft contracts the describe and specify API design details. |
| JSON | Javascript Object Notation |  |
| SOAP |  |  |
| XML | Xtensible Markup Language |  |
| FHIR |  |  |
|  |  |  |