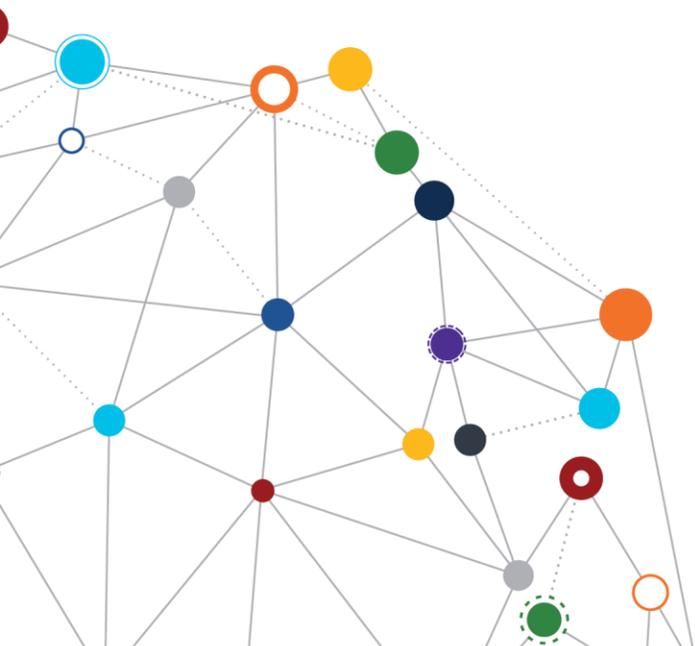


OFFICE OF
INFORMATION
AND TECHNOLOGY

Application Programming Interface (API) Enterprise Design Pattern

API Documentation Standard

February 2019 | Enterprise Program Management Office



VA



U.S. Department of Veterans Affairs
Office of Information and Technology



Table of Contents

1	Context	3
2	Problem	3
3	Approach	3
3.1	Adhere to API Documentation Principles	4
3.1.1	Adhere to VA Nomenclature.....	5
3.1.2	Determine and Gather Key Attributes.....	5
3.2	Document APIs in OpenAPI Format	7
3.2.1	Set Info Object	7
3.2.2	Set Server Object	8
3.2.3	Set Paths Object.....	8
3.2.4	Set Components Object	9
3.2.5	Set Security Scheme Object	10
3.2.6	Set Tag Object	10
3.2.7	Set External Documentation Object	10
3.3	Store API Documentation in the Appropriate Repository	11
4	Application	11
5	Impacts	13
	Appendix A: Resources and Relevant Capabilities/Limitations	14
	Figure 1: Example of Metadata.....	5
	Table 1: Change Matrix	2
	Table 2: Development Pathway Guidance.....	3
	Table 3: Minimum Attributes.....	5
	Table 4: Open API Document Sections	7
	Table 5: Documentation Repository Guidance.....	11
	Table 6: DEA User Stories.....	12

Table 1: Change Matrix

Version	Date	Description of Updates
1.0	6/29/2018	API EDP Segment 2 document approved
2.0	2/22/2019	Updated implementation, tools, testing, inventory guidance

1 Context

Application programming interfaces (APIs) enable business agility across the Department of Veterans Affairs (VA). *Maintaining a description* of an API supports understanding of the API's business context for both the development community and non-technical users. API *documentation* contains important notes about the API, providing *definitive guidance to developers* to help them code and reuse existing capabilities.

2 Problem

Variation in VA API documentation and management demonstrate that a standard for documenting APIs throughout VA is necessary. The *lack of common information* impairs the ability of developers to reuse APIs within VA.

3 Approach

API documentation must be developed, implemented, and deployed *using common guidelines* that are applied to all APIs exposed by VA, and used by both API consumers and providers. VA's standard for API documentation is the *OpenAPI Specification* (formerly Swagger).¹ Table 2 provides guidance for project teams with new API developments and with legacy APIs.

Table 2: Development Pathway Guidance

New API Developments	Legacy API Projects ²
<ul style="list-style-type: none"> All new API developments should follow Representational State Transfer (REST) constraints, if feasible for the use case.³ Any currently maintained RESTful APIs or new APIs should be documented using the attributes in Table 3. 	<ul style="list-style-type: none"> Project teams with legacy (non-RESTful) APIs that do not follow the Open API specification for documentation must create a plan of action for migrating to Open API, if feasible for the use case. This plan must follow the approved documentation criteria in this Enterprise Design Pattern (EDP) document.

¹ For more information on the *OpenAPI Initiative*, refer to <https://www.openapis.org/about>.

² The group of legacy APIs includes those APIs in existence on or before the publication date of this document.

³ In some cases, APIs may need lower latency or higher performance in transferring large amounts of data that cannot be accommodated by REST. REST constraints are reflected in the approved technical standards in the VA Technical Reference Model (TRM) at <http://trm.oit.va.gov/>. External vendors may utilize a less comprehensive site at <https://www.oit.va.gov/services/trm/>.

3.1 Adhere to API Documentation Principles

All APIs **must** be documented using the approved interface definition language (IDL), the Open API specification. API documentation must be validated to ensure that it adheres to the specification.

- API documentation **must** reflect pertinent methods (e.g., GET, POST, DELETE, PUT) and operations for the service that the API exposes.⁴ The API documentation will reflect customer-facing information about the service in a way that is compliant with any applicable service-level agreement (SLA).
- The details of OpenAPI are in Section 3.2 of this document. This approved standard is reflected in the VA Technical Reference Model (TRM) and in associated compliance criteria (Refer to Section 4 of this document.).⁵

Project teams implementing APIs should consider the following design principles when documenting and planning APIs.

- Projects responsible for developing an API should follow “contract-first” design practices by creating the API documentation during initial development. It is only after iterating on the API definition that the service is implemented. Creating the documentation in advance increases the probability of building a service that meets the needs of clients.
- Consider using API documentation markup tools (e.g., Swagger editor, Swagger Hub, etc.), mocking services, code validation tools, and code generation tools.
- Consult the API standards established by the General Services Administration’s (GSA) 18F Group.⁶
- Consider the purpose of the API (e.g., presentation, orchestration, system) and its uses in the initial design and documentation of the API.
- Consider the privacy implications of the API and develop a privacy statement describing the purpose, uses, and types of personal information that will be processed by the API. Thus, the design and use of data **must** be consistent with the original purpose of the information collection; any secondary uses **must** be described.
- Consider design practices (e.g., 18F API Standards) when developing API documentation. For example, consider accommodations for including relevant metadata. The documentation should include sufficient metadata to enable clients to calculate total data, and determine when and how to fetch the next set of results. See Figure 1 for an example from 18F.

⁴ One exception is when it is determined for security reasons that access to all the Path Object Information must be subject to access controls.

⁵ Refer to the Open API TRM entry on the internal VA network at <http://trm.oit.va.gov/>. External vendors may utilize a less comprehensive site at <https://www.oit.va.gov/services/trm/>.

⁶ Refer to API standards established by the General Services Administration’s (GSA) 18F Group at <https://github.com/18F/api-standards>.

```
{
  "results": [ ...actual results ... ],
  "pagination": {
    "count": 2340,
    "page": 4,
    "per page": 20
  }
}
```

Figure 1: Example of Metadata

3.1.1 Adhere to VA Nomenclature

Projects should align to the following existing VA nomenclature when developing naming conventions for each API and its documentation.⁷

- Namespaces for APIs should use data elements that are described in the VA Enterprise Architecture (EA) Data and Information Domain, as available; namespaces should represent business entities that align to the Business Reference Model (BRM).⁸
- When using structured data, project teams should consider using the JavaScript Object Notation (JSON) schema across databases, data, and Open API documentation to define access to an API. JSON Schema provides the ability to define references across and between individual schema.

3.1.2 Determine and Gather Key Attributes

Table 3 displays a set of attributes that are applicable for OpenAPI specifications that are based on the Representational State Transfer (REST) architecture style.

Table 3: Minimum Attributes

Attribute	Description	Open API Field
API Name	A name for the Application Programming Interface (API)	Info Object
Description	A clear explanation of the function of the method/resource; this should be a description of the API's purpose, functionality, value, and user community.	Info Object
Version	Current API version	Info Object

⁷ For naming convention considerations, refer to the ASG API Playbook Naming Standards Section at https://github.com/department-of-veterans-affairs/ES-ASG/wiki/05.01-ASG_API-Playbook_Naming-Standards_Section.

⁸ Refer to the VEAR Data and Information Domain at <https://vaausdarapp41.aac.dva.va.gov/ee/request/home>.

Attribute	Description	Open API Field
URL	API endpoint	Info Object
URL Parameters	A list of parameters used on this resource/method, including types, special formatting, rules, and requirements	Path Item Object
Endpoints and Operations	Describes methods (e.g., GET, PUT, POST, DELETE, OPTIONS, HEAD, PATCH, TRACE) and operations to invoke methods	Path Item Object
Security	Authentication (e.g., HTTP basic authentication, API key, Open Authorization (OAuth), OpenID Connect)	Security Requirement Object
Access Type	Internal/External, Trusted/Anonymous	External Documentation Object
Response Codes	HTTP Status and Error Codes	Responses Object
Samples/Examples	<ul style="list-style-type: none"> • A sample call, with correlating media type and body • A sample response, including media type and body • Code examples for multiple languages, including all necessary code • Software development kit (SDK) examples (if SDKs are provided) showing how to access the resource/method; and how to use the SDK for the language 	External Documentation Object
Points of Contact (PoC)	<ul style="list-style-type: none"> • Business PoC: The primary sponsor representative for the API that provides requirements and guidance on decisions that impact business functions; and that are supported by the API • Technical PoC: Responsible for day-to-day management of the API; ensures that all technical system components (software, infrastructure, platforms, network, security) are operational and integrated to support successful API functionality 	Info Object for Technical PoC; External Documentation Object for Business PoC

3.2 Document APIs in OpenAPI Format

Project teams should follow the Open API specification format to document APIs. The teams should follow the most recent OpenAPI version.⁹ An overview of the Open API document schema is provided in Table 4.

Table 4: Open API Document Sections

OpenAPI Field	Data Type	Example	Note
openapi	String	openapi: 3.0.2	Required by OpenAPI Specification
info	Info Object	See Section 3.2.1	Required by OpenAPI Specification
servers	Servers Object	See Section 3.2.2	
paths	Paths Object	See Section 3.2.3	Required by OpenAPI Specification
components	Components Object	See Section 3.2.4	
security	Security Requirement Object	See Section 3.2.5	
tags	Tags Object	See Section 3.2.6	
externalDocs	externalDocs Object	See Section 3.2.7	

The following subsections provide details on each of the main sections of the Open API format. A more detailed reference of the Open API specification is found in the Reference Section of this document.

3.2.1 Set Info Object

The Info Object must contain metadata to define the description, point of contact (PoC), and other information. The following example illustrates fields in this object for an Open API Specification definition of an API.

Example:

```
{
  "info": {
    "title": "Example Veteran App",
```

⁹ The most recent OpenAPI version is available at <https://github.com/OAI/OpenAPI-Specification>.

```

    "description": "Veteran experience application to show aggregated view of
Veteran information.",
    "termsOfService": "https://developer.va.gov/explore/terms-of-service",
    "contact": {
      "name": "John Smith",
      "url": "https://developer.va.gov/explore/exempleveteran",
      "email": "veteransupport@va.gov"
    },
    "license": {
      "name": "CC BY-SA 4.0",
      "url": "https://creativecommons.org/licenses/by-sa/4.0/"
    },
    "version": "1.0"
  }
}

```

3.2.2 Set Server Object

The Server Object defines connection data for the basepath of the server that hosts the API and service; the following provides a simple example. Additional server variables, such as username and port, can also be specified.¹⁰

```

{
  "servers": {
    "description": "Health record system API example",
    "url": "http://examplesystem.va.gov/exampleAPI/"
  }
}

```

3.2.3 Set Paths Object

Follow Open API specification to set Path Item Object for all relevant endpoints and operations. This object contains the main details of RESTful methods (e.g., GET, PUT, POST, HEAD) for the API. An abbreviated example is provided in this section. When documenting all methods and operations, project teams are expected to have a much larger Path Item Object.

```

{
  "paths": {
    "/results": {
      "get": {
        "description": "Returns all medical results the system can access",
        "responses": {
          "200": {
            "description": "A list of results",
            "content": {
              "application/json": {
                "schema": {

```

¹⁰ For additional details on fields that can be specified, refer to the Open API Specification at <https://github.com/OAI/OpenAPI-Specification>.

Separately, the Operation Object within the Paths Object can include links to external documentation for a given operation. The Tags Object can also include links to external documentation.

3.3 Store API Documentation in the Appropriate Repository

Project teams must place API documentation in one of two VA repositories, as shown in Table 5. Efforts are underway to consolidate these repositories into one site in the near future. This document will be updated to reflect this change when it is complete.

Table 5: Documentation Repository Guidance

API Developer Portal	Enterprise Service Collaboration Portal (ESCP)
<ul style="list-style-type: none"> Application Programming Interfaces (APIs) can be documented using the VA API Developer Portal, available at https://developer.va.gov. This site is accessible to external users and third parties (e.g., partners, hospital systems, research labs). 	<ul style="list-style-type: none"> APIs can be documented using the Enterprise Service Collaborative Portal (ESCP), available at https://escp.va.gov.¹⁵ Users of ESCP site can enter service requests to interface with the VA project team and obtain information on APIs and services.

4 Application

Project teams using the Veteran-focused Integration Process (VIP),¹⁶ or developing or retaining APIs, must comply with the approved standards in the VA Technical Reference Model (TRM);¹⁷ and map to the DEA user story in Table 6. Future changes in the standard will be reflected in the TRM; and in pertinent DEA user stories that are related to both API consumption and provisioning.

¹⁵ The VA Digital ESCP catalog of registered services is available at <https://escp.va.gov/ServiceCatalog/Search>.

¹⁶ Refer to the Veteran-Focused Integration Process (VIP) 3.1 Guide, April 2018, at <https://www.voa.va.gov/documentlistpublic.aspx?NodeID=27>.

¹⁷ Refer to the VA Technical Reference Model (TRM) to identify approved applications and standards on the internal VA network at <http://trm.oit.va.gov/>. External vendors may utilize a less comprehensive site at <https://www.oit.va.gov/services/trm/>.

Table 6: DEA User Stories

DEA User Story	Description	Acceptance Criteria
DEA 04.24.01	All APIs must be documented and placed in a repository accessible to VA project teams. All API documentation efforts must follow the <i>API Enterprise Design Pattern (EDP) Documentation Standard</i> . ¹⁸	<p>(1) 100% of newly developed APIs must comply with the <i>API EDP Documentation Standard</i>.</p> <p>(2) 100% of existing or legacy APIs must have a plan to be documented in accordance with the <i>API EDP Documentation Standard</i>.</p> <p>(3) 100% of API documentation must be registered on either the <i>Enterprise Service Collaborative Portal (ESCP)</i>,¹⁹ or the <i>API Developer Portal</i>.²⁰</p>

Future versions of this document will reflect updates to the DEA compliance criteria to reveal the guiding principles for API documentation. Compliance with this standard for API documentation applies to the following major project scenarios:

- All new development efforts that will be leveraging the VA Enterprise Cloud (VAEC)
- All existing APIs that expose Authoritative Data Sources (ADS) data
- All existing APIs that provide endpoints for approved Enterprise Shared Services (ESS)

Current APIs will be transitioned to the VA Developer Portal²¹ and the ESCP,²² which will be the required repositories for API documentation. In the future, these sites are expected to be merged.

¹⁸ Refer to the *API EDPs* at <https://www.oit.va.gov/library/recurring/edp/index.cfm>.

¹⁹ Refer to the ESCP at <https://escp.va.gov>.

²⁰ Refer to the VA API Developer Portal at <https://developer.va.gov>.

²¹ Refer to the VA Developer Portal at <https://developer.va.gov>. External-facing APIs, primarily used by groups external to VA, should document the APIs in the VA API Developer Portal.

²² Refer to the ESCP at <https://escp.va.gov>. APIs used by internal VA projects and teams should be documented in the ESCP.

5 Impacts

The following are potential impacts to product delivery if the API standards are not implemented:

- Documentation will continue to be limited, resulting in non-discoverable APIs.
- Newly created or currently maintained APIs will be non-compliant.
- Project teams will have difficulty reusing APIs and may create duplicate software.

Appendix A: Resources and Relevant Capabilities/Limitations

References:

- 18F GSA API Standards: <https://github.com/18F/api-standards>
- Example API Documentation, VA Benefits Intake, https://dev-api.vets.gov/services/vba_documents/docs/v0/api
- Example API Documentation, VA Facilities, https://dev-api.va.gov/services/va_facilities/docs/v0/api
- OpenAPI specification: <https://github.com/OAI/OpenAPI-Specification>
- REST Cookbook: <http://restcookbook.com/>
- VA DEA Assessment Guidance: https://vaww.portal2.va.gov/sites/asd/AERB/DEA_Assessment/DEA%20User%20Story%20Alignment/Home.aspx
- VA Directive 6551: https://www.va.gov/vapubs/viewPublication.asp?Pub_ID=829&FType=2
- VA Enterprise Services, Architecture Support Group, API Playbook, <https://department-of-veterans-affairs.github.io/ES-ASG/>

Disclaimer: This document serves both internal and external customers. Links displayed throughout this document may not be viewable to all users outside the VA domain. This document may also include links to websites outside VA control and jurisdiction. VA is not responsible for the privacy practices or the content of non-VA websites. We encourage you to review the privacy policy or terms and conditions of those sites to fully understand what information is collected and how it is used.

Statement of Endorsement: Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, and shall not be used for advertising or product endorsement purposes.