Telefónica

On the way towards “service APIs”

The document provides some discussions to help Telco Global API Alliance conduct the work towards a successful implementation of best-in-class service APIs.

Razón social
00.00.2015

Introduction

Analyst reports claim that API integration work accounts for 50% of the time and cost of building a digital platform

- The cool things we want to create rely on integration; the winners will be those who do the integration well.

As-in situation: API nuance forces developers to..

- sourt out the same complexity repeteadly
- remap and transform data
- rewrite programs for error handling, polling, and other critical functionality

Result: brittle, point-to-point solutions that are poorly document and hard for colleagues to maintain

This presentation outlines trends and best practices for API specification and development, so that Telco Global API Alliance leverage on them to work out service APIs.

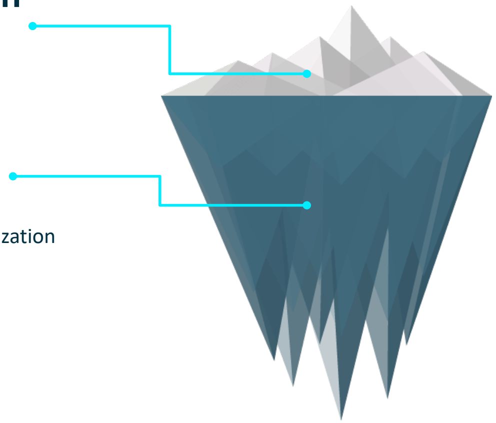
The two faces of an API

Connecting to an API is easy

- First Call to the Sandbox

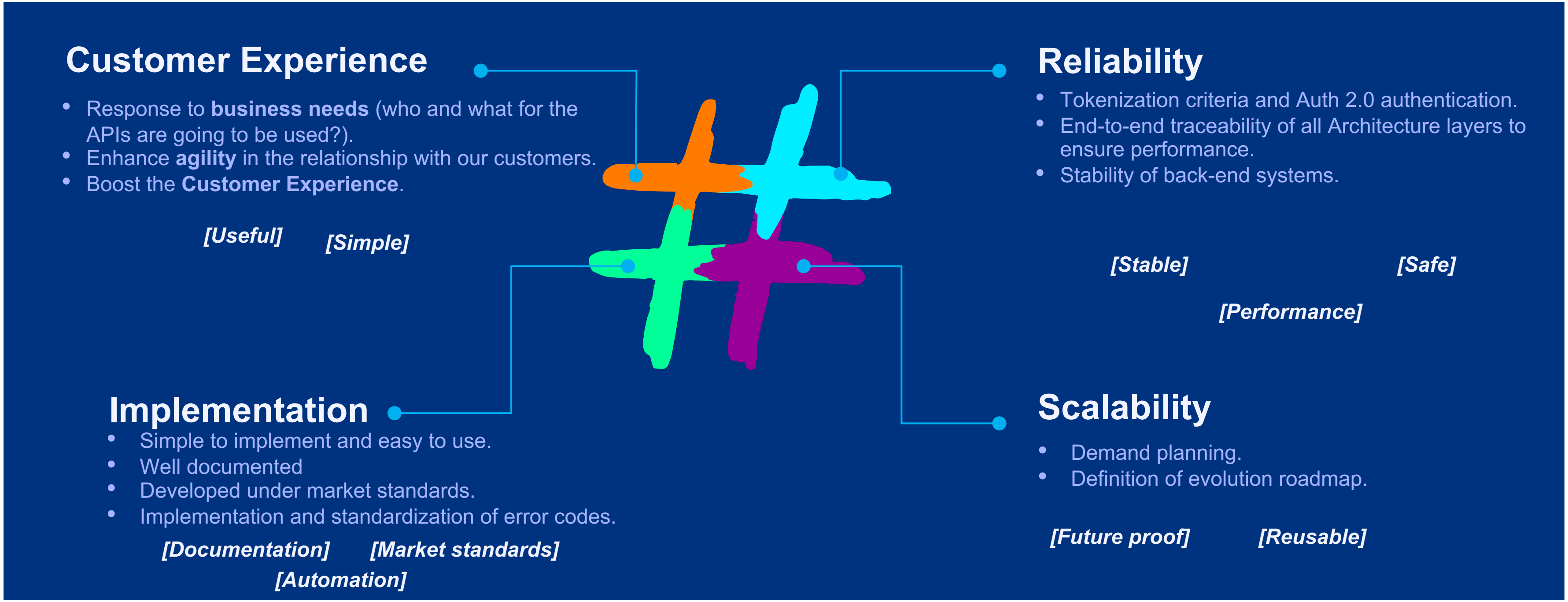
Integrating with an API is Hard

- Authentication and Authorization
- Map & Transform
- Workflows
- Events & Polling
- Bulk & File Management
- Error Handling & Recovery



Introduction

The key drivers to have a successful API are:



Customer Experience

- Response to **business needs** (who and what for the APIs are going to be used?).
- Enhance **agility** in the relationship with our customers.
- Boost the **Customer Experience**.

[Useful] ***[Simple]***

Reliability

- Tokenization criteria and Auth 2.0 authentication.
- End-to-end traceability of all Architecture layers to ensure performance.
- Stability of back-end systems.

[Stable] ***[Safe]***

[Performance]

Implementation

- Simple to implement and easy to use.
- Well documented
- Developed under market standards.
- Implementation and standardization of error codes.

[Documentation] ***[Market standards]***

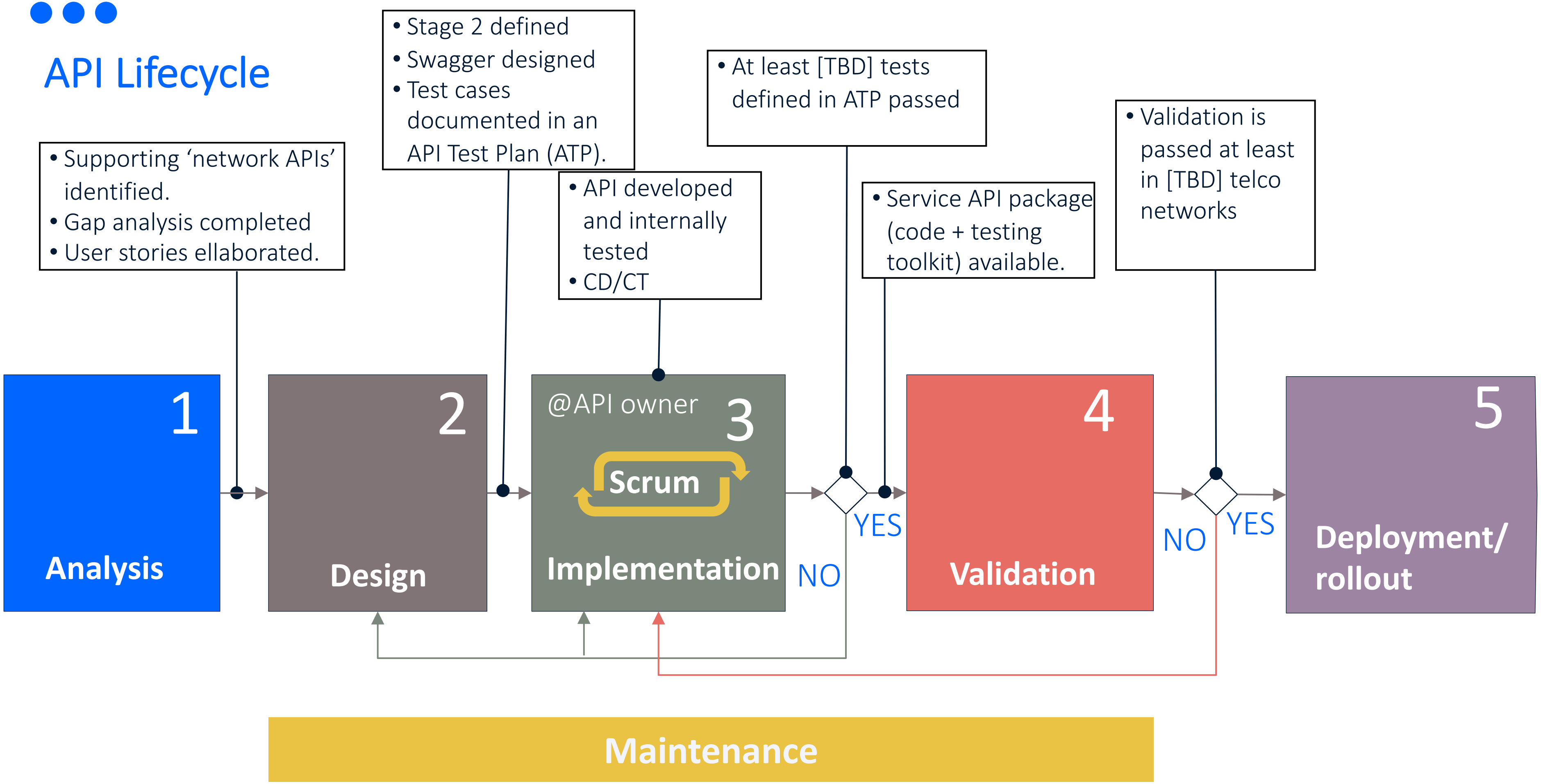
[Automation]

Scalability

- Demand planning.
- Definition of evolution roadmap.

[Future proof] ***[Reusable]***

API Lifecycle



- Supporting 'network APIs' identified.
- Gap analysis completed
- User stories ellaborated.

- Stage 2 defined
- Swagger designed
- Test cases documented in an API Test Plan (ATP).

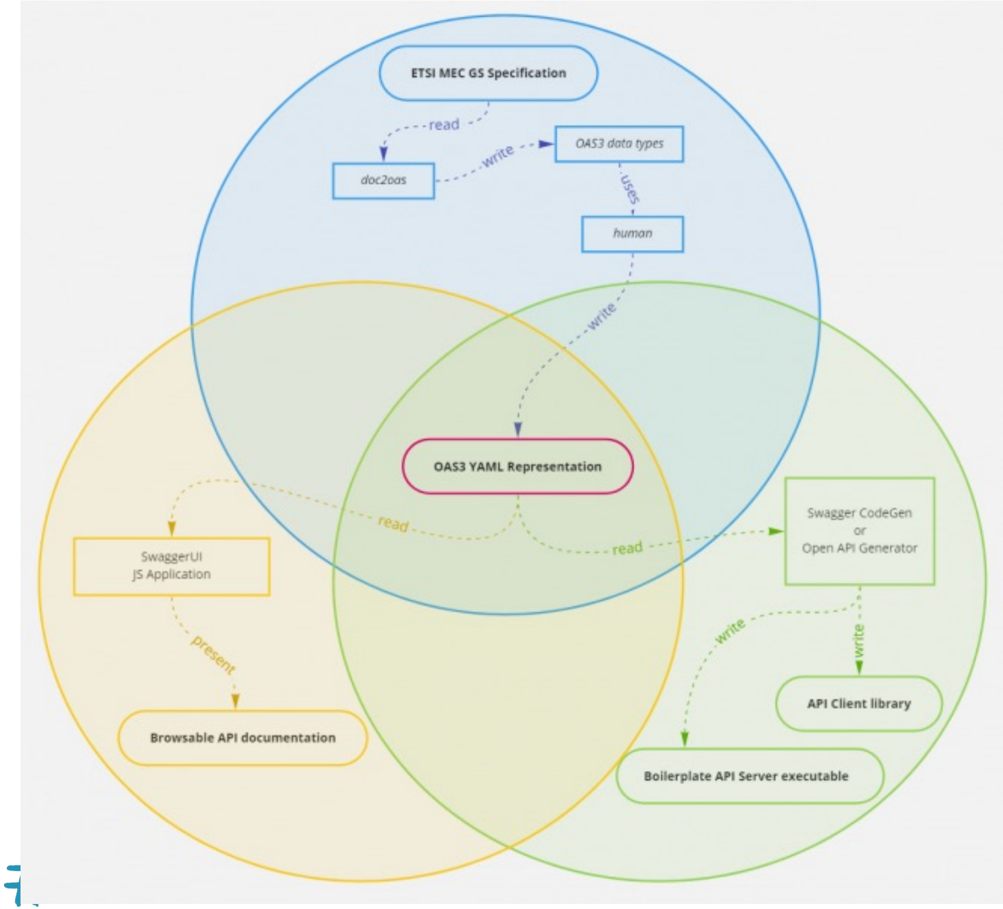
- API developed and internally tested
- CD/CT

- At least [TBD] tests defined in ATP passed

- Service API package (code + testing toolkit) available.

- Validation is passed at least in [TBD] telco networks

API development



- Handbook for API development: https://opensource.zalando.com/restful-api-guidelines/index.html#100

- Examples from SDOs: ETSI ISG MEC: https://mecwiki.etsi.org/index.php?title=OpenAPI_development_guidelines
- The left-side figure provides a high-level overview of the Open API Specification (OAS) creation and usage – example applied to MEC ISG

As shown on the figure, OAS YAML file is used in 3 different contexts:

- To accurately reflect the API
- To provide graphical UI documentation as an input to SwaggerUI JS Application
- To provide out-of-the-box API client library/package or server stub as an input to code generation tool

Way forward

● ● ● ● ●

Razón social
00.00.2015

Actions for the ‘commonalities’ workstream (1/2)

1. **Document the principles/criteria that the in-scope service APIs shall fulfill -> OPAG collaboration**
   - The input from customers (e.g. hyperscalers) is a must. Customer shall declare their expectations on the usability of APIs.
   - Used as a basis to decide whether a SDO API is eligilible for direct usability or not.
2. **Draft a template to describe user stories.**
   - Business-driven use cases is what really should drive the API design + implemnetation work in the TGA Alliance.
3. **Decide on the service API spec format**
   - There is a tendency in the industry in going for OSA3
   - However, industry-wide validators and benchmarks (e.g. TMForum) keep using Swagger 2.0.
4. **Decide on the authentication and authorization solution**
   - Authentication mechanisms: OAuth2, OpenID, SAML, TLS, JWT Authentication,…
   - Authentication workflow: for some service APIs, the operator may be required to call an identity provider (IDP) service.

Actions for the ‘commonalities’ workstream (2/2)

5. **Decide on the testing principles and tools for validation in telco network(s)**
   - Document this is a must, so as to ensure consistency.
6. **Decide on the criteria to go from step 3 (implementation) to step 4 (validation)**
   - This is important to manage expectations, especially on API developer side.
   - We do not need to have the API fully implemented – fleeing from traditional waterfall approach.
   - Example: At least [TBD] tests defined in the ATP passed.
7. **Decide on the criteria for API documentation publication**
   - Documentation should include: 1) complete API specifiation, with all methods and attributes; 2) API code and version; 3) succeed tests cases and related documentation
   - In order to manage expectations from customers and external fora/SDO, it is recommedable to publish ‘sucess test cases’ vs ’ATP’ (i.e. all test cases planned for the service API to be 100% functional). The goal is to get this ratio increased for every new API release/version.

What Telefónica can provide to the TGA initiative?

Razón social
00.00.2015

For prioritized API families, TEF can contribute in the….

Analysis stage

- Identification of network capabilities (functional requirements) -> *collaboration with OPAG*
- Gap analysis, in relation to existing 3GPP network APIs -> *collaboration with OPAG.*
- Ellaboration of user stories

Design stage

- Stage 2 solution for service APIs, including operations & notifications + input/output attributes (based on info model) -> *collaboration with OPAG*
- API Test Plans (ATP) Design

Provision of governance and lifecycle tools

- Centralized portal to manage the governance and lifecycle of in-scope APIs.
- This portal includes the following tools: API visor, code generator, document repository, certification/validation toolkit, user mgmt. & profiles. Further details in the next slide.
- Portal instance dedicated for the TGA initiative.

Validation stage

- Contribute to the testing of developed service APIs in our lab facility, leveraging certification/validation toolkit.

TEF API Design Phase

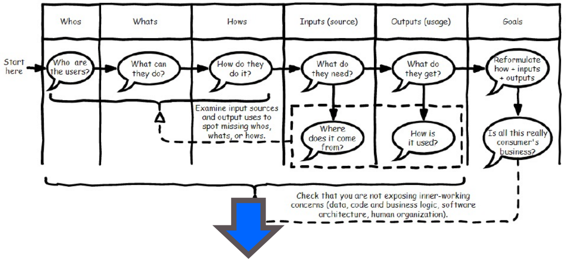
API Design Patterns

Customer Experience

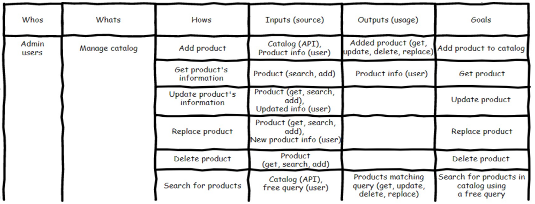
Respond to business needs

API Design Patterns

To make yourself a set of basic questions that are summarized in the API Goals Canvas Methodology.



Whos
Whats
Hows
Inputs (source)
Outputs (usage)
Goals
Start here
Who are the users?
What can they do?
How do they do it?
What do they need?
What do they get?
Reformulate how + inputs + outputs
Examine input sources and output uses to spot missing whos, whats, or hows.
Where does it come from?
How is it used?
Is all this really consumer's business?
Check that you are not exposing inner-working concerns (data, code and business logic, software architecture, human organization).



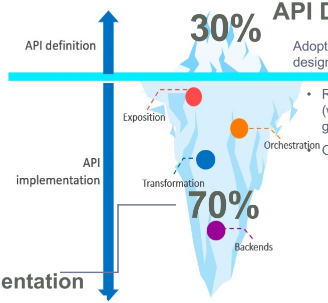
Whos	Whats	Hows	Inputs (source)	Outputs (usage)	Goals
Admin users	Manage catalog	Add product	Catalog (API), Product info (user)	Added product (get, update, delete, replace)	Add product to catalog
		Get product's information	Product (search, add)	Product info (user)	Get product
		Update product's information	Product (get, search, add), Updated info (user)		Update product
		Replace product	Product (get, search, add), New product info (user)		Replace product
		Delete product	Product (get, search, add)		Delete product
		Search for products	Catalog (API), free query (user)	Products matching query (get, update, delete, replace)	Search for products in catalog using a free query

Implementation

Strictly the API **is only the interface** that a certain software exposes

But you can't have a good API without a good implementation both parts are needed

API = Interface + Implementation



API Definition

Adopt the concept of "API First" design in the organization.

- Response to business needs (who and what for are the APIs going to be used?).

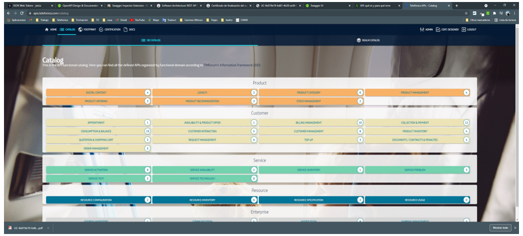
Organize by functional domain

API Implementation

Mapping the Network Southbound interfaces with the API proposes (Nortbound)

Scalability

- Have clear policy of versioning and APIs evolution
- Have good documentation and test cases to enhance reuse
- Have a Portal where APIs can be inventoried by functional criteria and have their technical information available



Catalog

This is the API functional catalog. Here you can find all the defined APIs organized by functional domain according to TMForum's Information Framework (SID)

TEF API lifecycle & governance toolkit

TEF has a tool made available to developer teams for the certification of T-Open APIs (Telefónica adaptation of TM Forum Open APIs).

Tool capabilities:

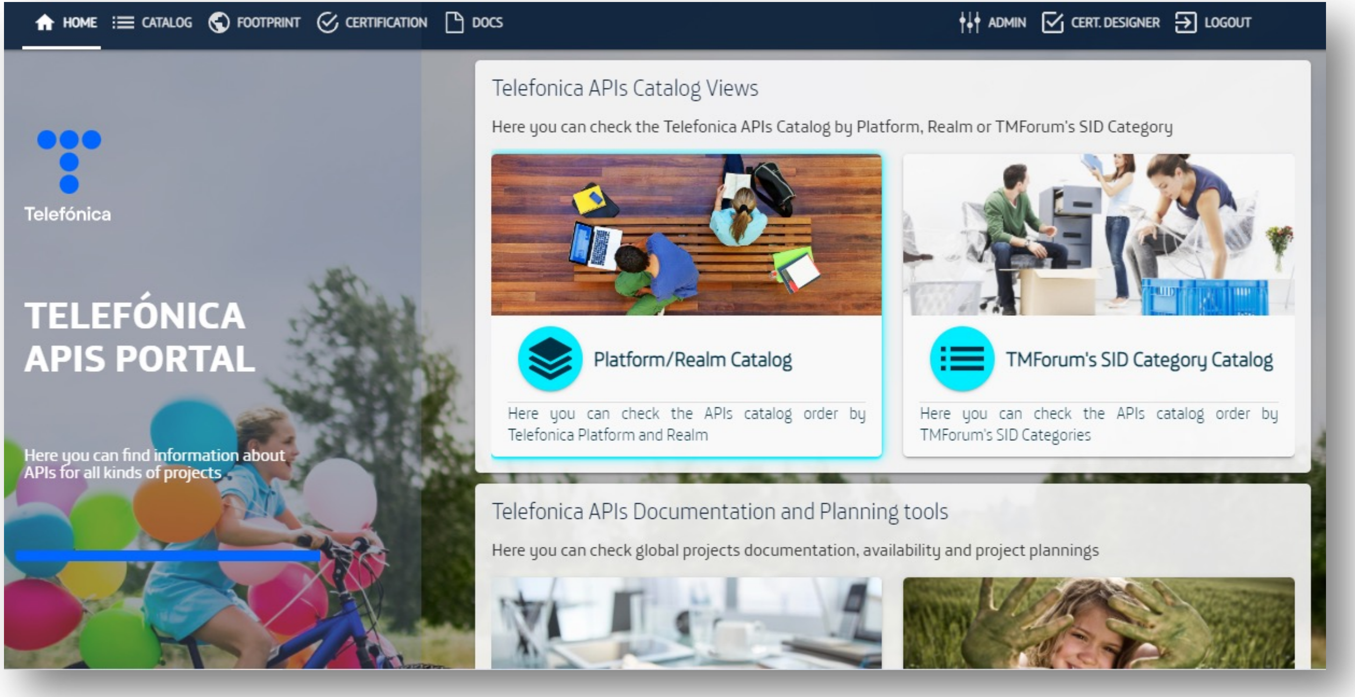
- Configuration of (deployment) environment: to register the endpoint where developer hosts the microservices
- Configuration of environment variables
- Test campaign automation: automation of tests cases detailed in the API Test Plan
- Support for Q&A process
- Reporting

Steps:

1. Select the tests to be run
2. Fill in the variable values
3. Launch certification and check the results
4. Certification report

Catalog & Documentation centralization Portal

When we are working in the interfaces design to create API Contracts It's so important the catalog centralization in order to make accessible, checkable and categorized all the information with a friendly user interface focused in all kind of profiles (Developers, consumers, exposers…)



TELEFÓNICA APIS PORTAL

Here you can find information about APIs for all kinds of projects

Telefonica APIs Catalog Views

Here you can check the Telefonica APIs Catalog by Platform, Realm or TMForum's SID Category

Platform/Realm Catalog

Here you can check the APIs catalog order by Telefonica Platform and Realm

TMForum's SID Category Catalog

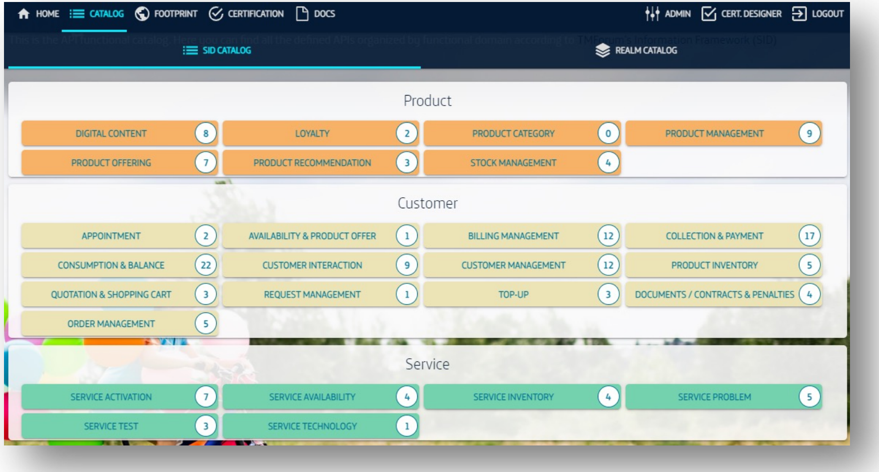
Here you can check the APIs catalog order by TMForum's SID Categories

Telefonica APIs Documentation and Planning tools

Here you can check global projects documentation, availability and project plannings

Catalog & Documentation centralization Portal

This kind of centralization makes available tools like API Catalog, API Contract Viewer, API Documentation Linked, API Code generator, Planification Tools, Testing Tools…



This is the API functional catalog. Here you can find all the defined APIs organized by functional domain according to TMForum's Information Framework (SID)

Product

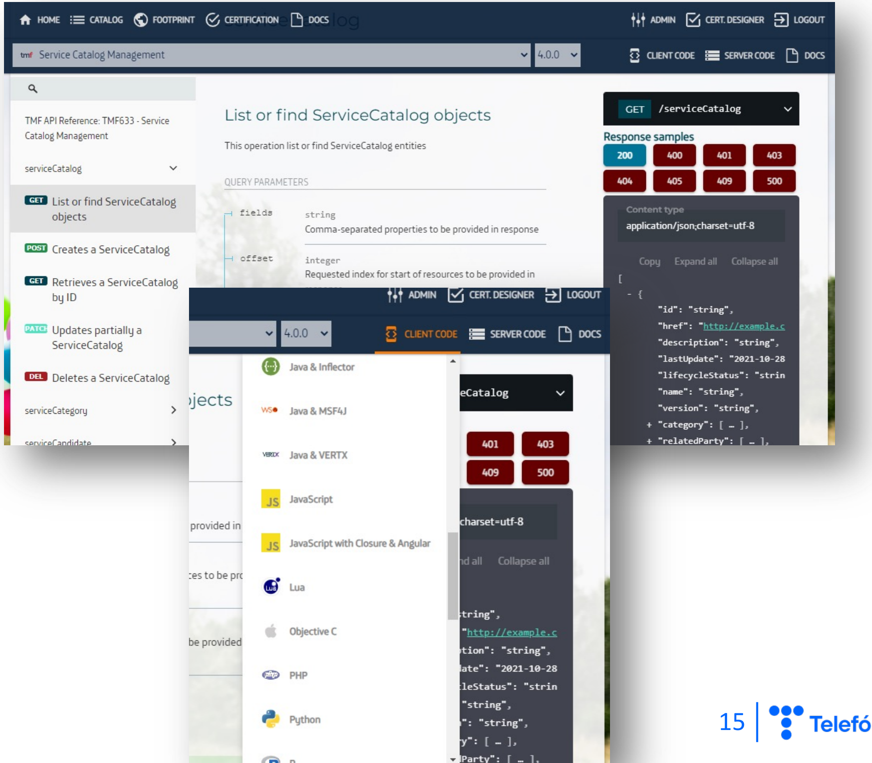
- DIGITAL CONTENT 8
- LOYALTY 2
- PRODUCT CATEGORY 0
- PRODUCT MANAGEMENT 9
- PRODUCT OFFERING 7
- PRODUCT RECOMMENDATION 3
- STOCK MANAGEMENT 4

Customer

- APPOINTMENT 2
- AVAILABILITY & PRODUCT OFFER 1
- BILLING MANAGEMENT 12
- COLLECTION & PAYMENT 17
- CONSUMPTION & BALANCE 22
- CUSTOMER INTERACTION 9
- CUSTOMER MANAGEMENT 12
- PRODUCT INVENTORY 5
- QUOTATION & SHOPPING CART 3
- REQUEST MANAGEMENT 1
- TOP-UP 3
- DOCUMENTS / CONTRACTS & PENALTIES 4
- ORDER MANAGEMENT 5

Service

- SERVICE ACTIVATION 7
- SERVICE AVAILABILITY 4
- SERVICE INVENTORY 4
- SERVICE PROBLEM 5
- SERVICE TEST 3
- SERVICE TECHNOLOGY 1



TMF API Reference: TMF633 - Service Catalog Management

serviceCatalog

- GET List or find ServiceCatalog objects
- POST Creates a ServiceCatalog
- GET Retrieves a ServiceCatalog by ID
- PATCH Updates partially a ServiceCatalog
- DEL Deletes a ServiceCatalog

serviceCategory

serviceCandidate

List or find ServiceCatalog objects

This operation list or find ServiceCatalog entities

QUERY PARAMETERS

fields	string Comma-separated properties to be provided in response
offset	integer Requested index for start of resources to be provided in

GET /serviceCatalog

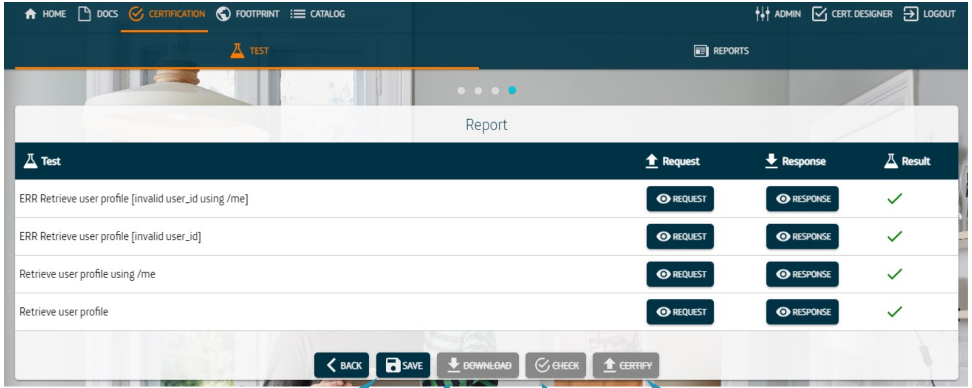
Response samples

Content type
application/json;charset=utf-8

```
[
  - {
      "id": "string",
      "href": "http://example.c
      "description": "string",
      "lastUpdate": "2021-10-28
      "lifecycleStatus": "strin
      "name": "string",
      "version": "string",
    + "category": [ … ],
    + "relatedParty": [ … ],
```

Catalog & Documentation centralization Portal

This kind of centralization makes available tools like API Catalog, API Contract Viewer, API Documentation Linked, API Code generator, Planification Tools, Testing Tools…



Report

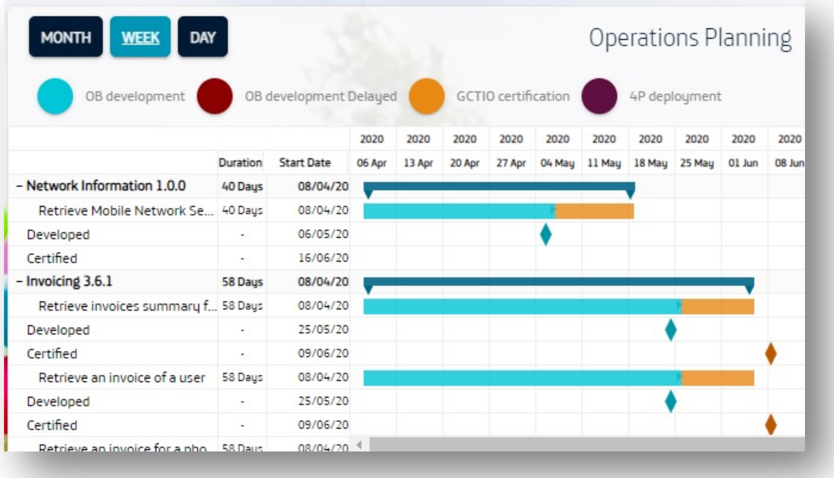
Test	Request	Response	Result
ERR Retrieve user profile [invalid user_id using /me]	REQUEST	RESPONSE	✓
ERR Retrieve user profile [invalid user_id]	REQUEST	RESPONSE	✓
Retrieve user profile using /me	REQUEST	RESPONSE	✓
Retrieve user profile	REQUEST	RESPONSE	✓

Save Report and active the other functions

Download PDF report

Check which operations are ready to deliver to CTIO

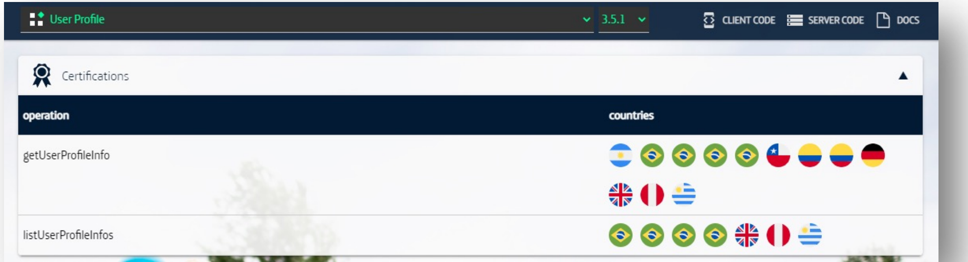
This functionality is only available for CTIO certifiers



Operations Planning

● OB development ● OB development Delayed ● GCTIO certification ● 4P deployment

	Duration	Start Date
– Network Information 1.0.0	40 Days	08/04/20
Retrieve Mobile Network Se…	40 Days	08/04/20
Developed	-	06/05/20
Certified	-	16/06/20
– Invoicing 3.6.1	58 Days	08/04/20
Retrieve invoices summary f…	58 Days	08/04/20
Developed	-	25/05/20
Certified	-	09/06/20
Retrieve an invoice of a user	58 Days	08/04/20
Developed	-	25/05/20
Certified	-	09/06/20
Retrieve an invoice for a pho…	58 Days	08/04/20



Certifications

operation	countries
getUserProfileInfo	
listUserProfileInfos	

Telefónica