MANUAL D'USUARI

API Manager
Proveïdor d'APIS

v3.0

Í N D E X

1. INTRODUCCIÓ

1.1. Objecte

L'objecte d'aquest document és descriure el ***funcionament operatiu de l'API Manager***. Es tracta d'un manual pràctic on es fa un recorregut per les diferents seccions, descrivint els passos i les operacions que es poden realitzar ***des del punt de vista del desenvolupador/proveïdor de les APIs***.

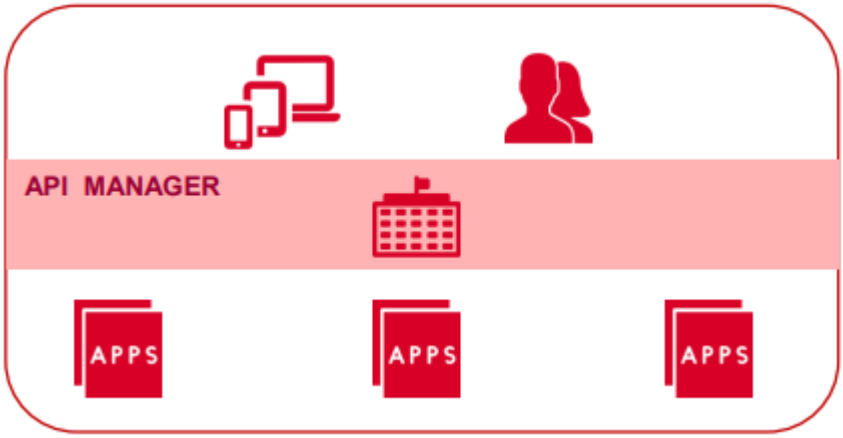
1.2. Abast

Els principals apartats que es tracten són:

- Descripció del servei de l'API Manager (APIM).
- Metodologia de treball
- Gestió de subscripcions
- Analítiques de consum.

1.3. Què és una plataforma d'API Manager?

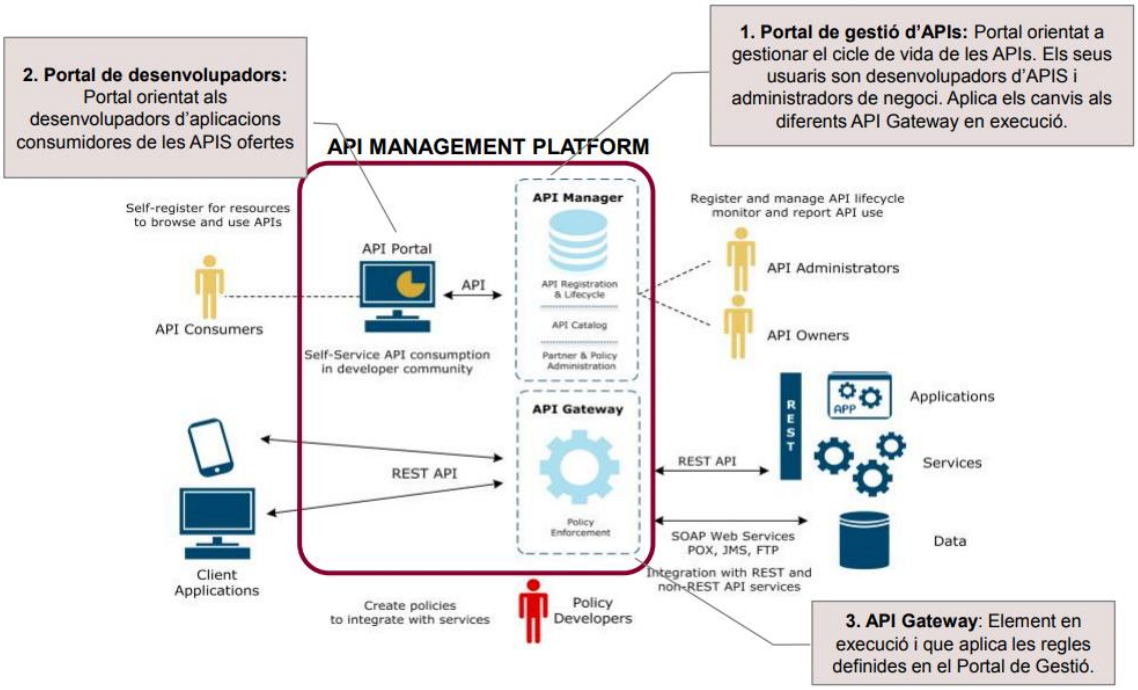
API MANAGEMENT (API: Acrònim "Application Program Interface"): És una plataforma que proporciona una capa de govern i control de l'execució dels serveis que s'exposen a tercers via API.



El podem definir com un element que s'encarrega de resoldre la part comuna de la complexitat inherent a la publicació dels serveis via API.

Una plataforma APIM consta de tres elements bàsics:

- ***Portal per a desenvolupadors*****:** Portal web d'accés als possibles consumidors de les APIs on poden consultar les APIs disponibles, informació de com utilitzar-les, i demanar accés al seu ús amb un previ registre per a obtenir un token d'accés.
- ***Portal de gestió d'APIs*****:** Portal web d'accés als gestors de l'API on poden consultar les APIs disponibles, administrar els accessos, plans i analitzar les dades d'accés.
- ***API Gateway*****:** Element tecnològic (escalable segons necessitat) on passen les peticions en el moment d'execució. Rep les definicions de les APIs i dels permisos dels diferents portals d'administració. És on arriben les peticions dels clients registrats i que re-dirigeix la petició als backends.

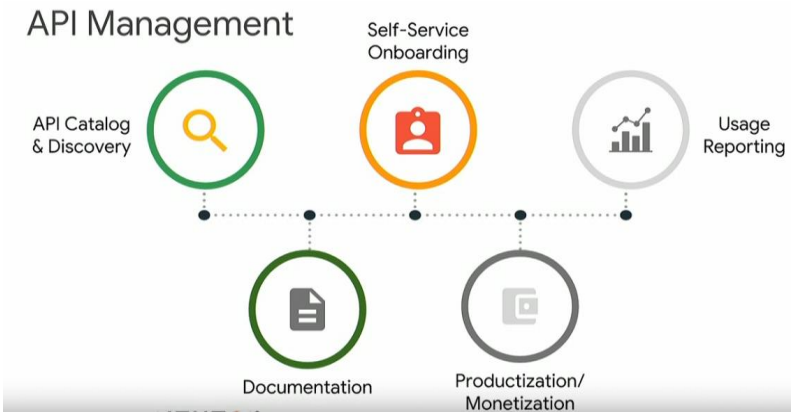


1.3.1. Funcionalitats

Una plataforma API Manager aporta unes funcionalitats bàsiques que el diferencien d'una altra categoria de productes:

Funcionalment:

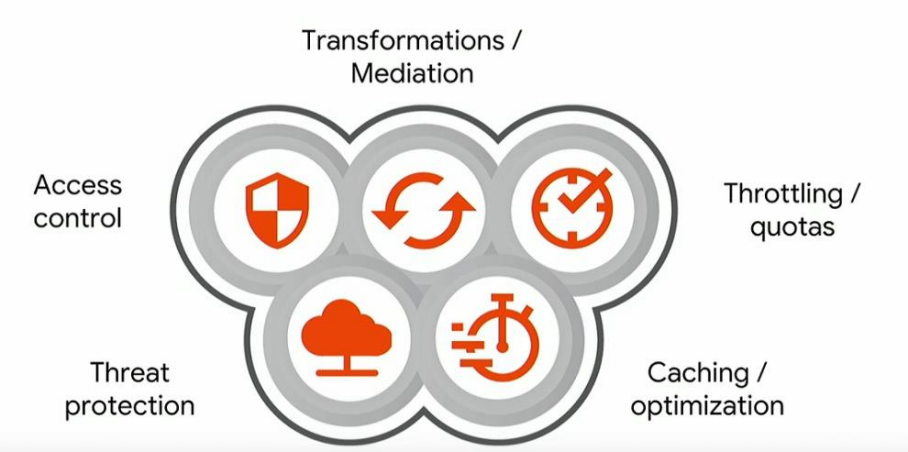
- Disposa d'un catàleg amb funcionalitats per a versionar i descobrir APIs.
- Disposa de funcionalitats d'autoservei en la subscripció a les APIs.
- Aporta un portal on publicar documentació associada a la utilització de les APIs.
- Proporciona accés a generar reportings sobre l'ús de les APIs.
- Permet associar cost a la utilització de les APIs i gestionar el seu repartiment.



Tècnicament:

- Control d'accés i definició de plans: disposar d'un registre de "clients/consumidors" de les nostres dades.
- Oferir diferents nivells de servei i control del consum (throttling): prioritzar peticions de determinats

"clients" i regular quotes de consum.
- Realitzar analítica unificada de les dades que es publiquen (qui accedeix a que, temps de resposta, etc.)
- Seguretat, establir polítiques de caché, prevenció davant d'atacs, protecció dels sistemes de backoffice.
- Aïllar els consumidors dels publicadors de serveis i dades.
- Transformar i combinar APIs.



1.3.1.1. Control d'accés

Registre de clients/consumidors:

- Registre de totes les Aplicacions consumidores d'APIs
- Informació de cadascuna de les organitzacions consumidores d'APIs.
- Informació dels Tokens associat a cada aplicació (qui té permís).
- Suspensió de Tokens i inclús eliminació de les Aplicacions.
- Accés global o per a un grup d'APIs.

1.3.1.2. Portals

Pels creadors d'APIs:

- Gestió del cicle de vida de les API: dissenyar i crear una API, associar a un Backend, versionar, deprecar, eliminar.

Pels consumidors d'APIs:

- Portal de desenvolupadors personalitzable i configurable:
  - Documentació de les APIs generada automàticament.
  - Codis d'exemple en diferents llenguatges de programació.
  - Simulacions de peticions.
  - Blogs i Fòrums de discussió.

1.3.1.3. Control de consum i nivells de servei

La solució permet:

- Establir límits de consum associats a plans d'ús.
- Limitar tant la concurrència com el número total de peticions per franja de temps.

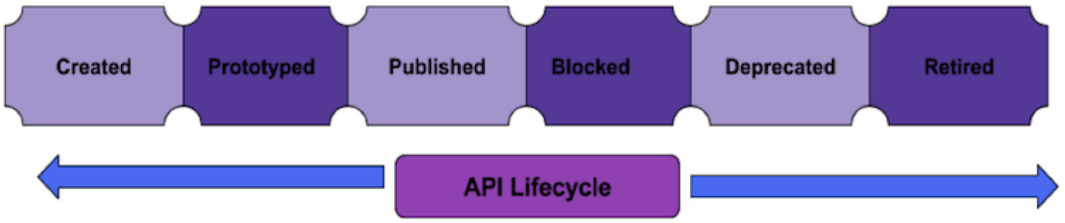
1.3.1.4. Anàlisis d'ús i Monitorització

La solució ofereix eines d'analítica unificada:

- Analítica global de totes les APIs.
- Analítica segmentada per APIs o agrupació d'APIs.
- Panells configurables (per IBM).
- Exportació en brut de dades per ser explotades amb eines d'anàlisi complex.

1.3.1.5. Gestió del cicle de vida de l'API

Les APIs publicades en un APIM tenen un cicle de vida orientat a gestionar i minimitzar l'impacte en els consumidors d'aquestes (subscriptors).



1.3.1.6. Seguretat

La gestió d'usuaris està separada per consumidors i creadors d'API. La seguretat està basada en perfils i àmbits (multi-tenant).

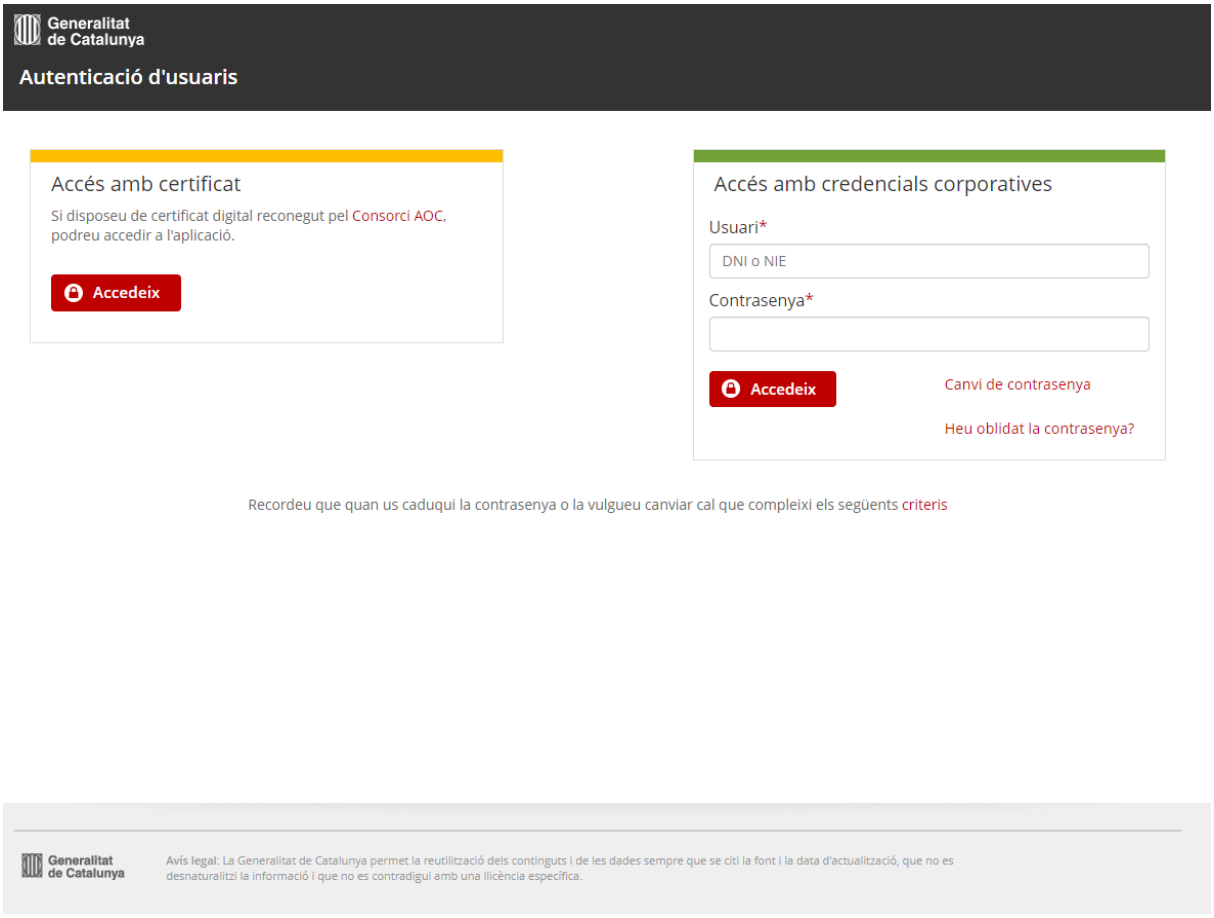
Es proporciona protecció contra les principals vulnerabilitats definides per OWASP:

- Denegació de serveis.
- Injecció de codi.
- Cross-Site Scripting.
- Seguretat a les APIs de manera robusta i senzilla.
- Elements de cache per protegir el backend i millorar el rendiment.

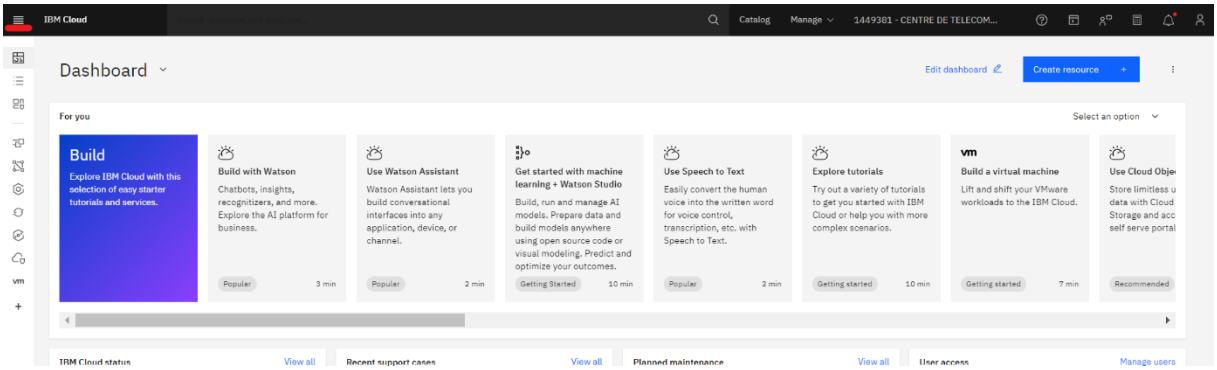
1.4. Access a l'API Manager (IBM API Connect)

Per poder entrar a l'API Manager, primer un usuari s'ha de registrar a l'IBM Cloud i l'OFT ha de donar-li l'accés de lectura per defecte (també pot donar-li accés per gestionar subscripcions en el seu espai). L' adreça a la qual ha d' accedir és: https://cloud.ibm.com/authorize/gicar

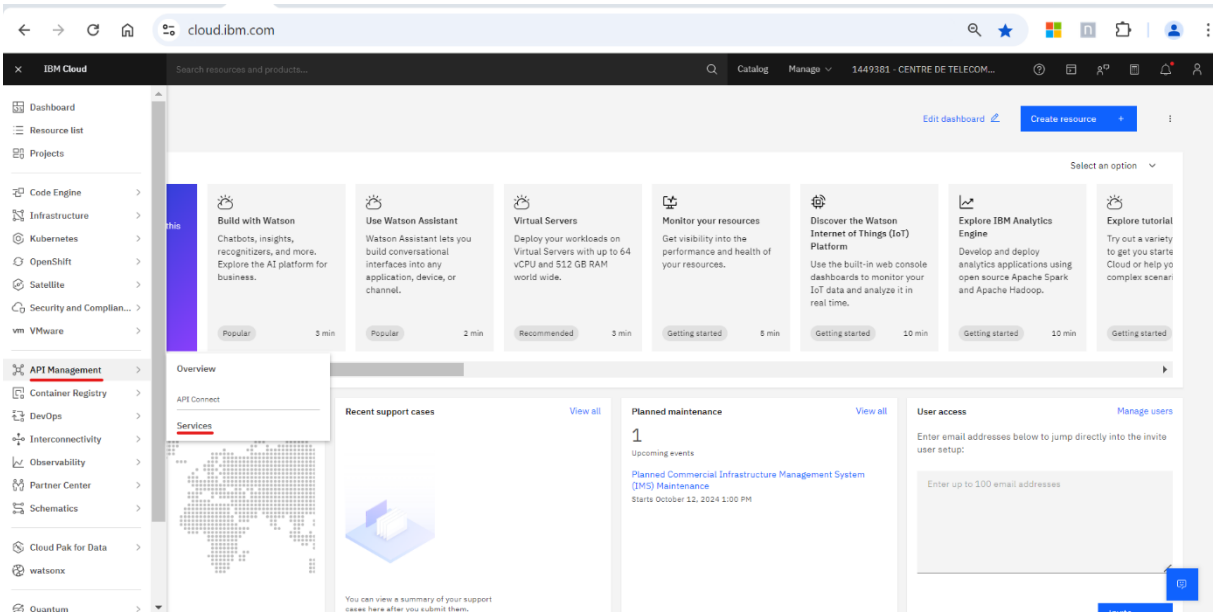
Aquesta url redirigeix a la pàgina de login de ***GICAR***, on un s'ha de registrar amb el seu usuari de GICAR.



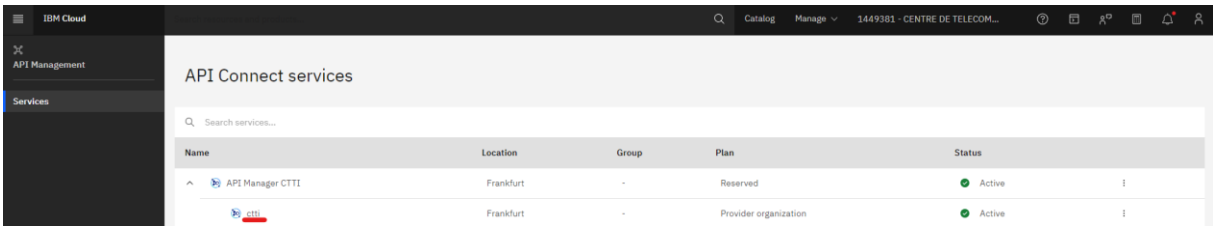
Un cop registrat, l'usuari pot accedir a la consola de gestió de l'API Manager de CTTI polsant primer a la cantonada superior esquerra de la pàgina d'inici de l'***IBM Cloud***.



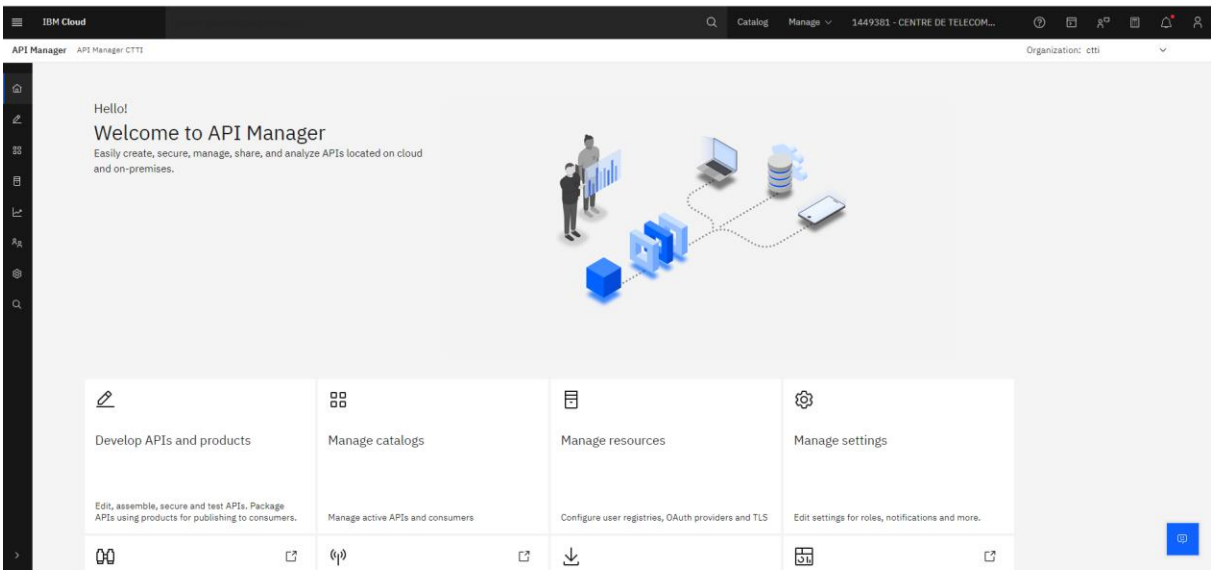
A continuació, ha de polsar en ***API Management / Services***.



Després, s'ha de polsar en ***ctti***.



Això permetrà que es carregui la pàgina principal de la consola de gestió de l'API Manager.



2. PLATAFORMA API MANAGER CORPORATIVA GENCAT

2.1. Descripció de la plataforma API Manager de GENCAT

CTTI basa la seva infraestructura per al ***APIM*** en un model de cloud “híbrid”, amb capacitat de gestionar i connectar APIs tant en entorns locals (on-premise) com en el núvol, diferenciant-se clarament dos grans blocs.

D'una banda, estaria la part ***Cloud*** subscrita a un servei ***SaaS (IBM API Connect Reserved Instance)*** ofert per IBM, des d'on es realitza la gestió i control de les APIs, brindant monitoratge, anàlisi i administració de les APIs.

Las características clau de la instància API Connect Reservada son:

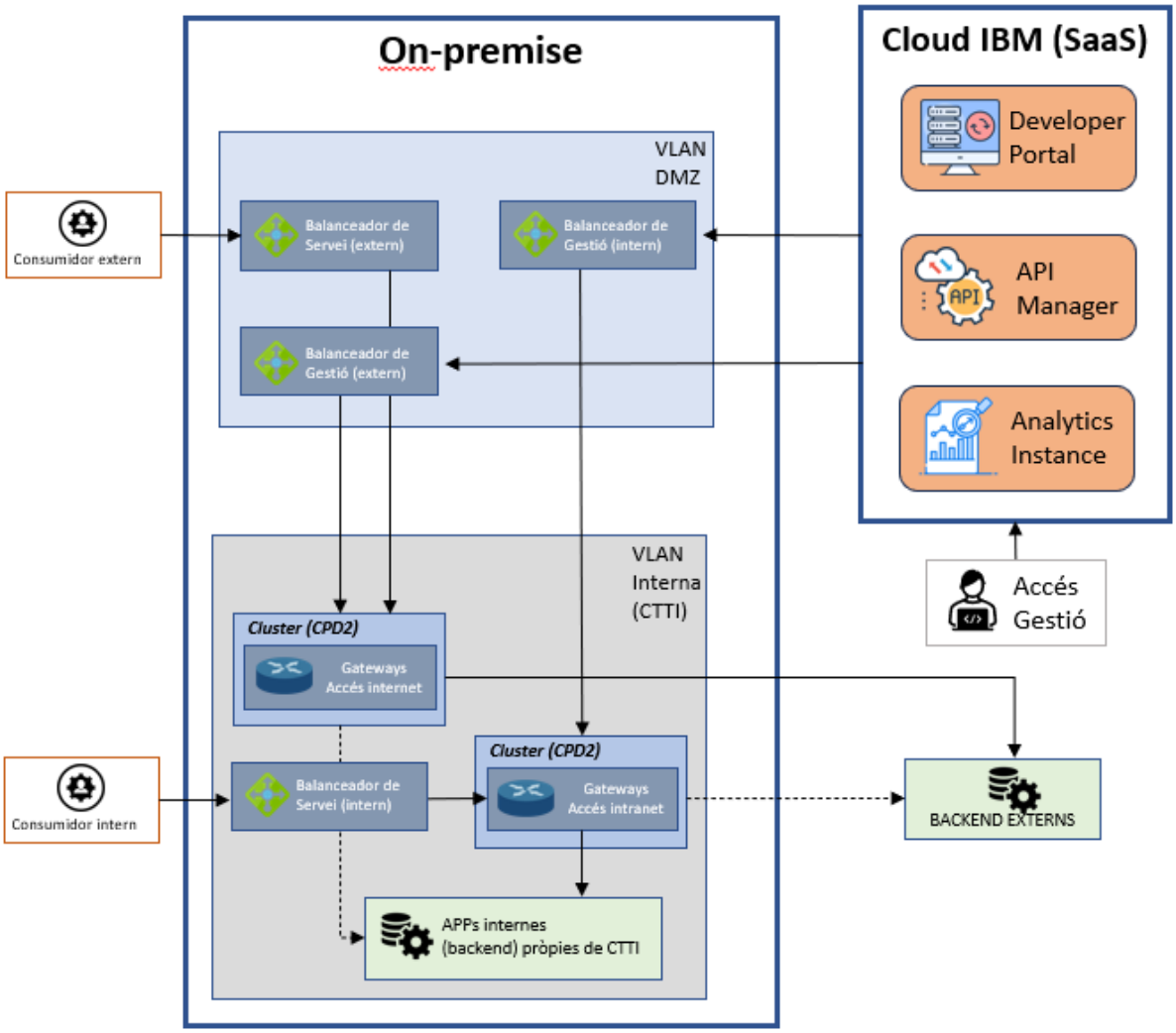
- Inici de sessió comuna amb altres serveis d'IBM Cloud mitjançant GICAR via integració OIDC o IBMid
- Aïllament d'uns altres que utilitzen el servei públic
- Gestionada, supervisada i operada per l' equip d' Operacions d' API Connect
- Desplegada en diversos servidors dins de la zona de centre de dades a efectes de resiliència

Los principals components del mòdul SaaS d'instància reservada son:

- ***Mòdul de gestió de APIs*****.** En dita modulo, es realitzen totes les tasques d'administració (gestió transversal de les APIs i el seu cicle de vida, gestió d'accessos, veure analítiques, configuracions).
- ***Portal de Desenvolupadors*****.** Plataforma per on els desenvolupadors podran accedir i consultar les APIs disponibles per al seu consum, subscrivint-se a les adequades.
- ***Gateways*****.** Són portes d'enllaç empresarial dissenyades per a exposar de manera segura les dades i les aplicacions empresarials onsevulla que resideixin: en local i en clouds. ***En concret, CTTI usa Gateways propis on-premise, en lloc d'usar els Gateways de la instància reservada*****.**
- ***Analytics*****.** Permet als usuaris obtenir informació sobre el rendiment de les APIs, l'ús dels recursos ajuda a l'organització a la detecció de problemes de rendiment, la identificació de patrons d'ús i la presa de mesures per a millorar la seguretat.

D'altra banda, estaria el bloc ***On-Premise*** que comprèn tots aquells components necessaris per a la comunicació i interconnexió (segments de xarxa, balanceadores, Gateways, etc..) i els elements del backend amb les funcionalitats dels serveis a explotar oferts per CTTI, tant a nivell intern (intranet) com a extern (internet). D'aquest bloc, destacar que les Gateways es troben desplegades en ***CPD2***.

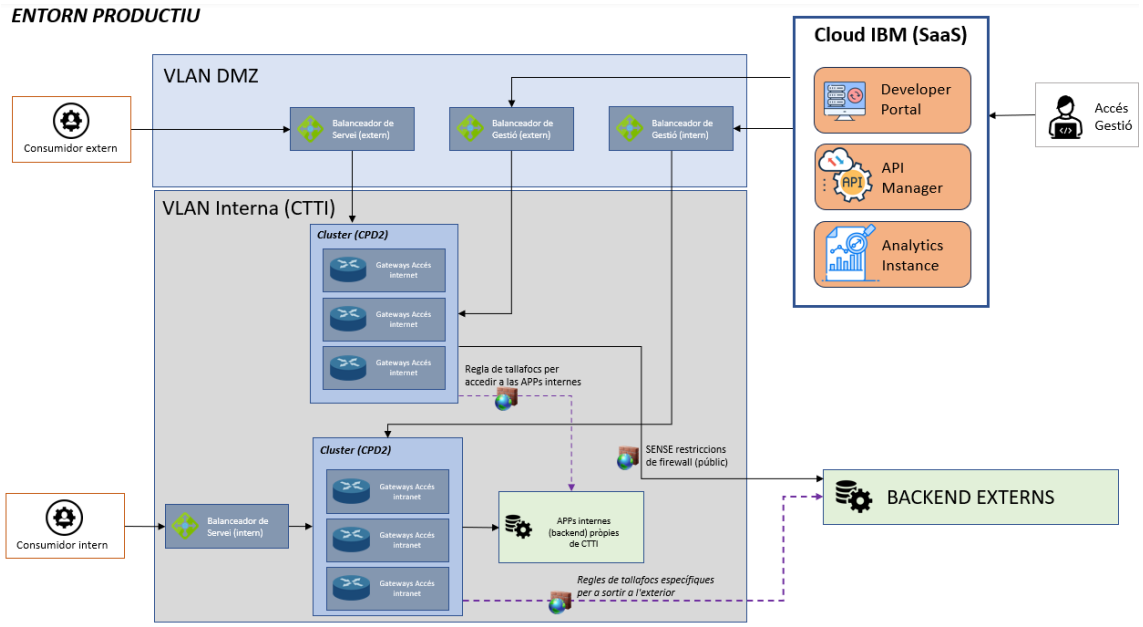
A nivell de connexió, els consumidors accedeixen als diferents segments de xarxa a través dels balanceadores de servei corresponents (intern i extern), els quals connecten amb les respectives Gateways, els quals realitzen el control i redirecció de les peticions als elements backends corresponents, ja sigui a nivell intern de CTTI, com a nivell extern.



A nivell d'arquitectura, la plataforma de API Manager en CTTI compta amb dos entorns definits, ***entorn productiu i entorn no productiu***, distribuïts en dos segments de xarxa (***VLAN Interna*** i ***VLAN DMZ***). Les VLAN són una manera de dividir una xarxa física en xarxes lògiques separades, la qual cosa permet segmentar el trànsit i millorar la gestió i seguretat d'una xarxa.

Entorn productiu (PRO). És l'entorn que es troba a la disposició dels usuaris finals per al consum de les APIs en un model d'alta disponibilitat. Amb la següent distribució:

- 6 Gateways: distribuïdes en 2 clusters de 3 Gateways cadascun, un donant servei a peticions des de la DMZ i un altre a peticions des de la VLAN Interna.
- 4 Balanceadores:
  - 2 Balanceadores per a distribuir les peticions a nivell de gestió sobre els components que conformen la plataforma.
  - 2 balanceadores per a distribuir les peticions a nivell de servei, és a dir, les aplicacions manen les peticions a aquests balanceadores per a després accedir a les endpoints de les APIs publicades en els Gateways.



Entorn no productiu (PRE). Ofereix un entorn similar al de producció tant a nivell de gestió com de servei, encara que no està configurat amb alta disponibilitat. Així, la distribució en el CPD2 que conforma aquest entorn és la següent:

- 2 Gateways: un donant servei a peticions des de la DMZ i un altre a peticions des de la VLAN Interna
- 4 Balanceadores amb la mateixa configuració que en l'entorn productiu

Els proveïdors usen aquest entorn per a desplegar els seus APIs en una primera fase, i realitzar totes les proves necessàries com són proves d'integració, proves d'acceptació d'usuaris prèvies per a assegurar el posterior desplegament en producción.

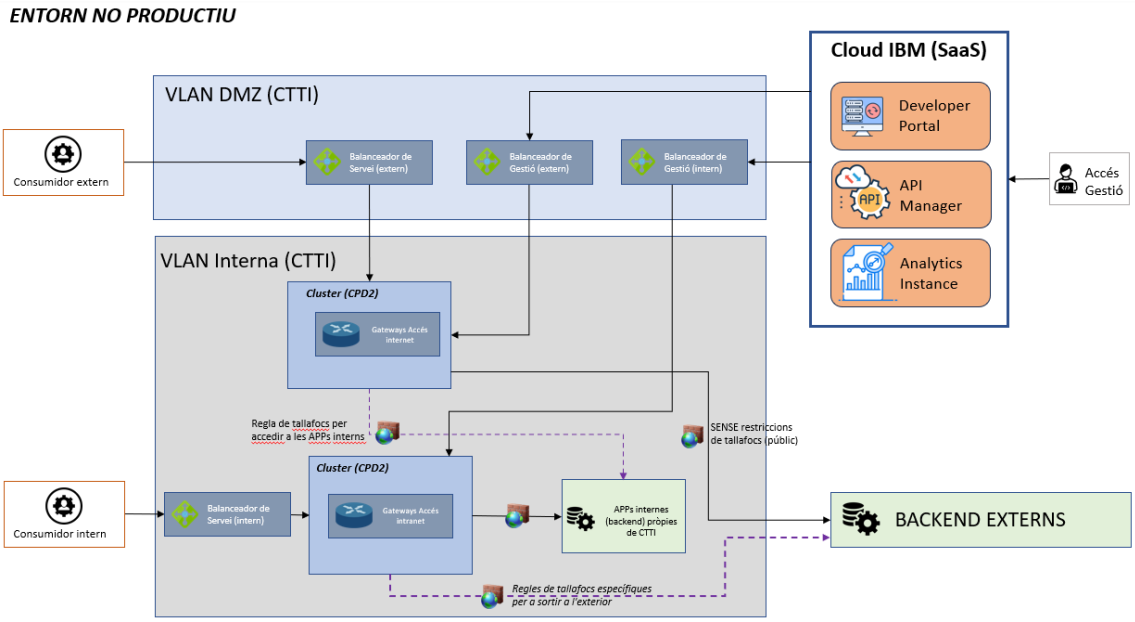
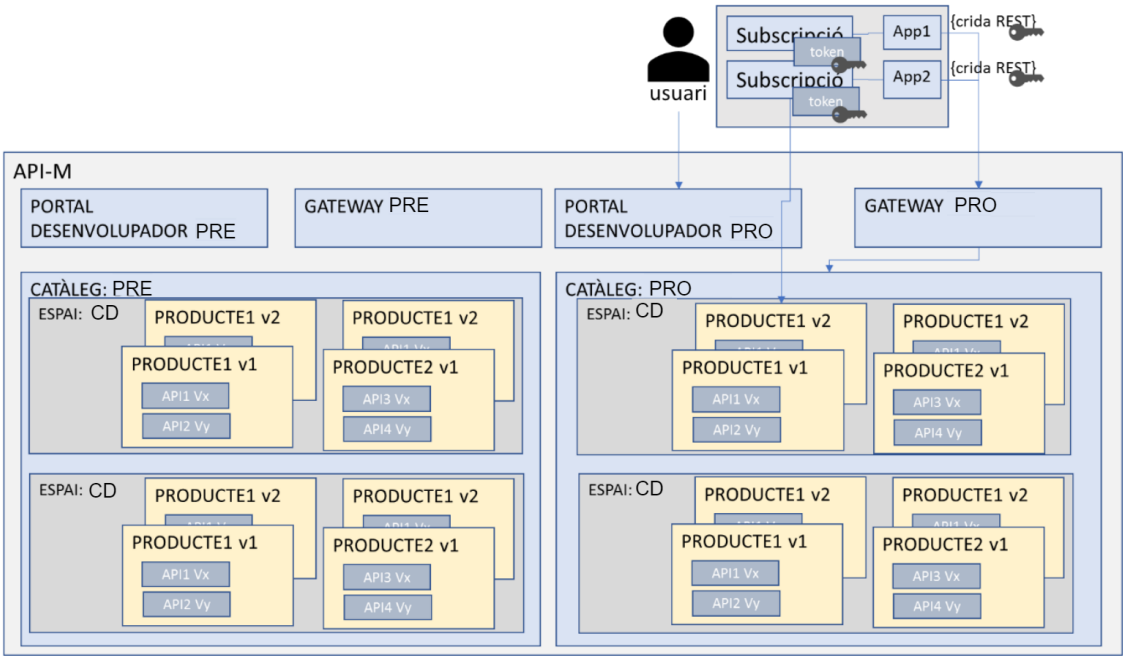
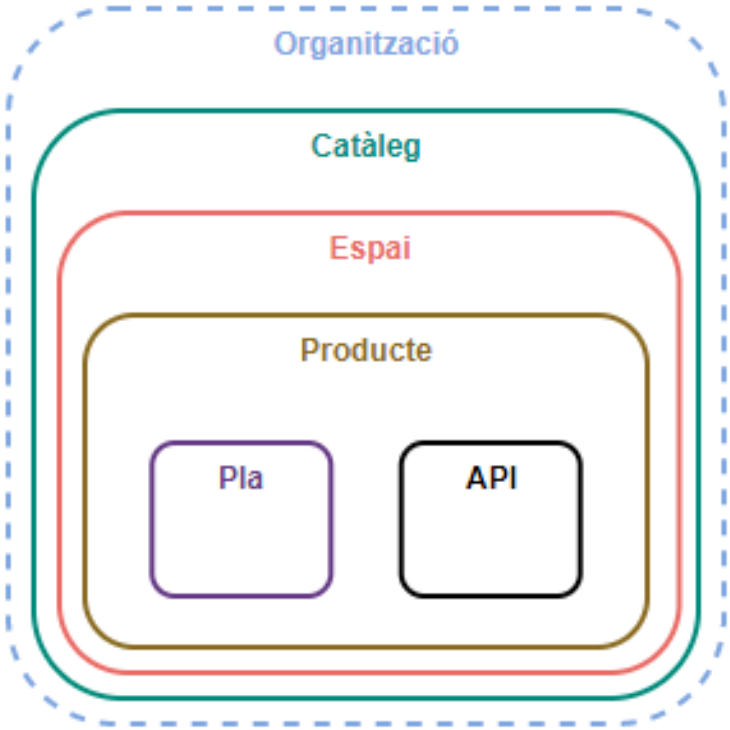


Diagrama conceptual de la implantació de l'API Manager corporatiu GENCAT (el nombre de catàlegs i espais pot variar, a les imatges s'il·lustra només amb un catàleg de PRE i PRO):



2.2. Organització

El servei d'API Manager s'organitza seguint el model que ofereix la solució API Connect, de la següent manera:



2.2.1. Catàlegs

Conté una col·lecció de productes. Permet separat els Productes i les APIs i la seva publicació en els diferents entorns disponibles; PRE i PRO.

L'API Manager corporatiu disposa actualment de quatre catàlegs:

1. Privat Preproducció (***privat-pre***).
2. Public Preproducció (***public-pre***).
3. Privat Producció (***privat***).
4. Public Producció (***public***).

Els catàlegs de PRE estan orientats a ser els catàlegs no productius on provar les APIs, mentre que els catàlegs de PRO són l'equivalent a l'entorn Productiu.

Els catàlegs de PRE disposen també d'un portal de publicació on també es pot comprovar com queda publicada la documentació. A més, es pot aplicar el versionat de les APIs abans de fer-ho en els catàlegs PRO.

Aquest son els portals corresponents als 4 catàlegs del servei de l'API Manager de la Generalitat en modalitat d'infraestructura compartida. Aquest portals proporcionen l'entorn de treball necessari a l'usuari per consumir productes i APIs, a més de crear i gestionar Apps.

- PRE**:**
  - Privat: https://portal.db40-c57f0fcb.eu-de.apiconnect.appdomain.cloud/ctti/privat-pre/
  - Public: https://portal.db40-c57f0fcb.eu-de.apiconnect.appdomain.cloud/ctti/public-pre/

- PRO**:**
  - Privat: https://portal.db40-c57f0fcb.eu-de.apiconnect.appdomain.cloud/ctti/privat/
  - Public: https://portal.db40-c57f0fcb.eu-de.apiconnect.appdomain.cloud/ctti/public/

És necessari registrar-se a cada catàleg (auto-registre).

2.2.2. Espai

Un catàleg es particiona en espais per separar la gestió dels Productes entre equips. Dins de cada catàleg es definiran Espais dedicats a cada codi de diàleg per a la gestió de les seves APIs. Aquests espais es crearan amb la nomenclatura ***CDXXXX***, sent XXXX el valor del codi de diàleg del projecte.

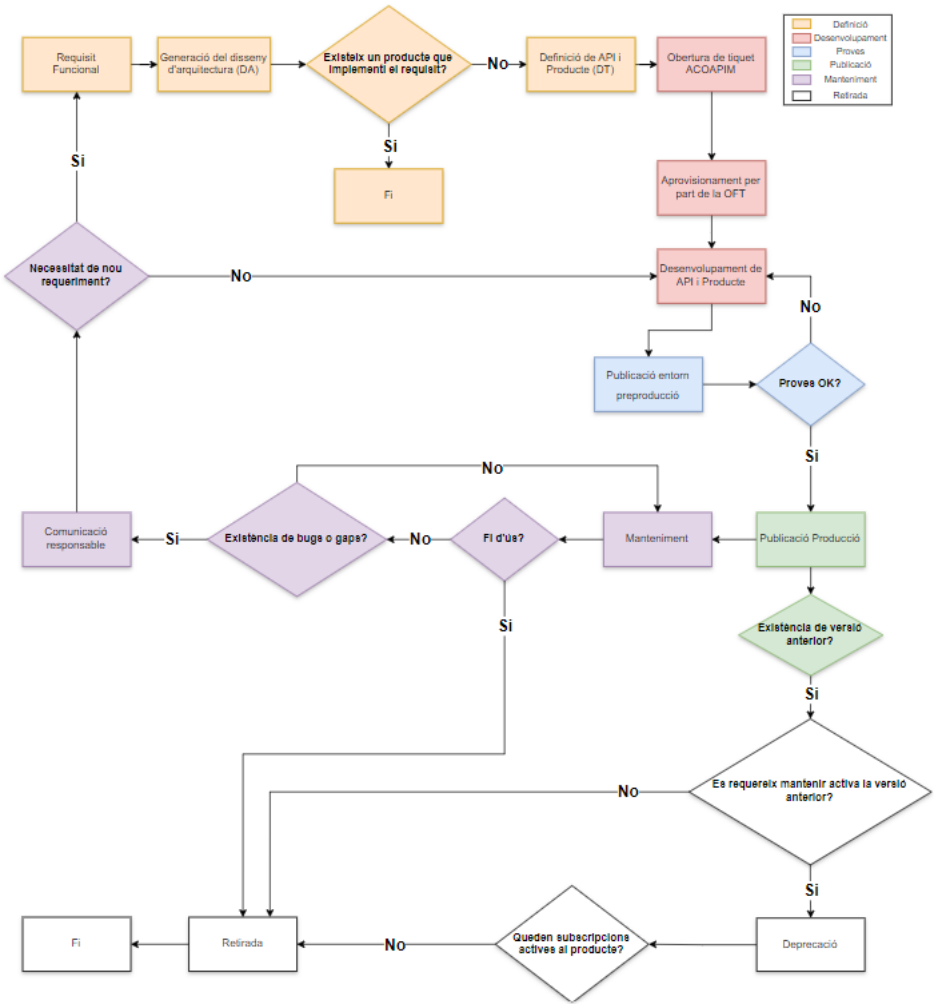
2.2.3. Productes

Les APIs s'organitzen per Productes, que passa a ser la unitat mínima a versionar i desplegar. Els productes son paquets que contenen tan les APIs com els Plans que les acompanyen. En el portal del desenvolupador, les subscripcions es demanen a nivell de producte. Una API ha d'estar al menys en un producte i en un pla. Un pla controla l'accés i l'ús a les APIs.

3. CICLE DE VIDA D' UNA API

Una API passarà per diferents estats des de la seva concepció fins que sigui possible consumir-la en producció, sent aquest el seu cicle de vida. Un bon cicle de vida de desenvolupament ha de tenir com a objectiu exposar l' API des d' una fase primerenca del desenvolupament, permetent als consumidors iniciar la seva integració amb rapidesa, alhora que reduir el temps de desplegament en l' entorn de producció.

Per això, es defineix un cicle de vida d' una API per a l' ecosistema de CTTI amb les següents fases:



3.1. Fases del cicle de vida

3.1.1. Definició

El cicle de vida de l'API comença amb aquesta primera fase, en la qual primer es realitza una etapa iterativa d'anàlisi i descobriment.

Primerament, l' equip proveïdor haurà de realitzar una anàlisi dels productes publicats actualment en l' espai del codi d' aplicació corresponent, per saber si existeix algun producte que satisfaci aquesta necessitat de manera parcial o completa.

Existiran les següents casuístiques:

- ***Producte existent que compleix la funcionalitat de manera completa:*** en aquest cas, el flux acabaria.
- ***Producte existent que compleix la funcionalitat de manera parcial:*** en aquest cas, es realitza un versionat del producte existent afegint la nova funcionalitat.
- ***Producte inexistent:*** en aquest cas, s' haurà de generar un producte nou.

Un cop s' ha identificat la necessitat de realitzar un nou desenvolupament i si ha de ser inclòs en un producte existent o un de nou, es procedirà a la definició completa del canvi o nou desenvolupament. Aquesta informació s'ha de reflectir en els documents de ***Descripció d'Arquitectura (DA)*** i ***Disseny Tècnic Detallat (DT)***.

Aquests requisits inclouran, entre d' altres:

- Funcionalitat/cas d'ús
- Producte/catàleg on s'inclourà
- Plans de consum
- Basepath
- Mètodes
- Esquemes d' entrada, sortida i error
- Seguretat de cada recurs
- Endpoints per a cada entorn

En el document de ***Descripció d'Arquitectura***, quan es faci referència a l'API Manager dins del punt de vista de context, cal indicar el Gateway/xarxa en el qual es disposarà l'API:

- Intranet
- internet

Dins del ***Document Tècnic Detallat***, haurien d' estar incloses les següents dades per a cadascuna de les APIs amb les quals es treballarà:

- ***Nom*** de l' API.
- Operacions, seguint el format ***VERB /path***. ***Per exemple:*** POST /example.
- ***Seguretat*** de l' API o de cada operació, que pot ser:
    - Authorization Code
    - Authorization Code amb PKCE
    - Client Credentials
    - Mutual TLS
- ***Acords d' interfície***, tant amb consumidor com amb backend per a cadascuna de les operacions definides.
- ***Polítiques customitzades*** a utilitzar durant l'acoblament, com poden ser ***recuperació de variables***, ***validació d'entrada/sortida***, etc.

Sobre el producte sobre el qual es treballarà s' haurien d' indicar:

- ***Nom*** del producte.
- ***Plans*** de subscripció.

3.1.2. Desenvolupament

És en aquesta fase quan demanen un tiquet d'acompanyament al projecte ***d'ACOAPIM de JIRA***. Aquí és on entra l'oficina tècnica. Demanen la creació dels espais per al seu codi d'aplicació, obtenir permisos d'accés al Cloud/API Manager i les urls dels portals i Gateways.

Un cop el document de definició quedi complet, es passarà a dissenyar aquesta definició de l' API. La recomanació és que es realitzi en el format ***OpenAPI 3.0***, llevat que existeixi un impediment del seu ús per algun gap o requisit. El desenvolupador comptarà amb l'ajuda d'una plantilla per a ***YAML*** que bé podrà pujar a ***l'API Toolkit***, on realitzarà els canvis corresponents per completar el disseny, o bé modificarà en un editor de text prèviament fins a completar el disseny per després pujar-la a l'API Toolkit. Es recomana la primera opció per les facilitats que dona l'eina d'API Toolkit, alhora que permet la realització de proves

en local en poder connectar-se amb ***l'LTE (Local Test Environment)***. Realitzada la primera definició de l' API, es realitzarà la implementació de l' acoblament a l' API Toolkit.

Després d'acabar el desenvolupament de la/s API/s, es procedeix a generar el producte corresponent.

3.1.3. Proves

Un cop l' acoblament de l' API s' hagi implementat, es començarà amb les ***proves unitàries***, realitzades mitjançant l' eina de ***LTE*** per part del desenvolupador.

Una vegada realitzades i validades les proves unitàries, es publicarà en el catàleg de ***preproducció*** el producte associat amb l'API, on es realitzaran les proves necessàries (***integrades i d'usuari***).

Les publicacions en els entorns es realitzaran seguint la metodologia de ***DevOps*** definida més endavant, que fa ús de ***pipelins automatitzades***.

3.1.4. Publicació

Un cop les proves d' usuari finalitzin correctament i s' hagi validat per complet el nou desenvolupament, es publicarà el producte a l' entorn de ***producció*** usant els ***pipelins*** de ***DevOps***. Si es desplega una nova versió d' un producte existent, la versió anterior passarà a la fase de ***retirada*** del cicle de vida.

3.1.5. Manteniment

Un cop el producte hagi estat publicat en producció, serà consumit pels clients.

Durant el transcurs del temps en què aquest producte és consumit, pot ocórrer que surtin ***incidències***, com que el rendiment no sigui l' adequat, que existeixin bugs o gaps en la seva implementació o que apareguin noves necessitats per part dels consumidors. Això significa que els productes han de ser mantinguts mentre estiguin desplegats, havent-hi un suport davant d' incidències, per tal de mantenir-los actualitzats a les necessitats actuals i assegurar un funcionament correcte en tot moment. Aquest suport i manteniment serà portat mitjançant l' obertura d' incidències en ***Remedy***, com correspon a la normativa de CTTI.

Els consumidors, en trobar un funcionament erroni o anòmal, obriran un tiquet Remedy i aquest tiquet serà resolt bé per ***l'equip proveïdor i/o l'oficina tècnica de l'API Manager***, en funció de quin sigui el problema trobat i la solució requerida.

Durant el transcurs de la resolució del tiquet Remedy, pot ocórrer que es trobi algun bug, gap o requeriment nou de l' API. En aquest cas, els responsables de l'equip proveïdor n'hauran de tenir constància, bé perquè siguin ells qui l'hagin descobert o bé perquè l'oficina tècnica se'l traslladi (si és ella la que té assignada el tiquet i la que descobreix el bug, gap o requeriment), i revisaran quin tipus de canvi s'ha d'implementar a l'API, podent ser només un ***patch*** que solucioni un bug existent o una nova ***versió major o minor*** que implementi un nou requeriment trobat.

3.1.6. Retirada

Quan es desplegui una nova versió d' un producte, s' haurà de determinar si cal mantenir activa la versió anterior.

Si cal mantenir activa la versió anterior, es procedirà a ***deprecar*** el producte, en el qual quedarà visible per als subscriptors actuals, però no es podran realitzar noves subscripcions, llevat que es requereixi permetre per algun motiu noves subscripcions durant un temps limitat (veure documents que continguin les directrius ***d'Estratègia de versionat i Bones Pràctiques*** per a més informació).

En el cas que no sigui necessari mantenir activa la versió anterior o el producte vagi a deixar de ser utilitzat, aquest producte es mourà a un ***estat retirat*** i les APIs que el conformen passaran a un ***estat offline***, sent retirades i eliminades posteriorment de l' API Manager, ***finalitzant així el seu cicle de vida***.

3.2. Actors, rols i tasques

Els actors que participen en el cicle de vida de les APIs, juntament amb els seus corresponents rols i tasques associades, són els següents.

Rol	Tasques
Responsable APIs de l'Àmbit	• Responsable funcional de les APIs publicades. • Responsable de gestionar el cicle de vida de les APIs. • Responsable d'aprovar les peticions d'accés a les APIs. • Accedeix al portal de gestió per a executar les tasques d'extracció d'estadístiques.
Proveïdor d'APIs (lot Ax)	• Desenvolupador de les APIs. • Publicarà el codi i parametrització de l'API a publicar a través de SIC. • La tasca de publicació s'automatitzarà en el SIC.
Consumidor d'APIs (lot Ax)	• Consumidor de les APIs publicades. • Sol·licita accés a les APIs a través del portal del desenvolupador.
Oficina Tècnica APIM	• Responsable tècnic (administrador) de la plataforma SaaS d'API Connect i de su correcte funcionament. • Gestió de la capacitat global de la plataforma. • Extracció dels indicadors d'ús. • Aprovisionament inicial dels accessos de cada àmbit. • Coordina la resolució de les incidències del servei APIM, redirigint l' equip responsable en cada cas.
CPD2	• Aprovisionament i l'administració de la infraestructura dels gateways.

Quant a com es trasllada això a la plataforma i el cicle de vida d'una API, IBM API Connect compta amb la funcionalitat de rols d'usuari que permeten limitar la visibilitat i permisos dels usuaris que utilitzen la plataforma. Per tant, definint els rols necessaris per administrar el cicle de vida d' una API, es pot associar un o diversos dels rols d' usuari de la plataforma a diferents membres de l' equip responsable d' arquitectura, i segregar la responsabilitat de totes les tasques involucrades.

Actualment, se segueix el següent model de rols per a l' organització proveïdora CTTI a la plataforma d' API Manager:

- ***Owner.*** Té associats tots els rols de la plataforma. Aquest rol el té una única persona, que es correspon amb el ***responsable per part de CTTI de la plataforma*** d'API Manager.
- ***OT.*** Les persones que són ***membre de l'oficina tècnica d'API Manager*** compten amb aquest rol i s'encarreguen de ***mantenir i administrar la plataforma***. Aquest rol es dona mitjançant l'ús del grup d'accés ***CTTI – API Manager OT*** de l'***IAM d'IBM Cloud***, el qual té associats tots els rols de la plataforma, menys el rol d'***Owner***.
- ***Developer.*** Aquest rol es dona mitjançant l'ús del grup d'accés ***CTTI – API Manager Developer*** de l'***IAM d'IBM Cloud***, el qual té associat el rol de ***Viewer*** de la plataforma. CTTI només permet als desenvolupadors (***Proveïdor d'APIs***) l' accés a la plataforma en ***mode de lectura***, per tal d' evitar que puguin modificar desenvolupaments i configuracions d' altres proveïdors.
- ***Community Manager.*** Aquest rol es dona a un ***responsable d'un projecte (Responsable APIs de l'Àmbit)*** en el cas que aquest projecte requereixi de la necessitat ***d'administrar les seves subscripcions***. Per a això, es creen dos grups d'accés a l'IAM amb el nom ***CTTI – CDXXXX – PRE i CTTI – CDXXXX – PRO***, on les ***XXXX*** seran substituïdes pel codi d'aplicació corresponent, de manera que coincideixi amb el nom de l'espai al qual es donarà accés. Aquest grup d' accés dóna el rol de ***Community Manager*** de la plataforma només dins l' ***espai*** corresponent.

Els consumidors (***Consumidor d'APIs***) es registraran als portals del desenvolupador, unint-se a una organització consumidora. Dins d' aquesta organització, podran tenir el rol d' ***Owner, Administrador, Desenvolupador o Visor***, rols que en aquest cas tenen ***permisos diferents*** als anteriorment definits, atès que només aplicaran dins del ***Developer Portal***.

- Els usuaris de l'organització consumidora amb rol de ***Visor*** podran veure, però no modificar res del contingut dins de l'organització consumidora.
- Els usuaris amb rol de ***Desenvolupador*** podran crear aplicacions amb les quals subscriure' s als productes.
- Els usuaris amb rol d' ***Administrador o Owner*** podran administrar els membres i els seus permisos, alhora que les aplicacions i subscripcions.

Un cop comentat el model de rols a la plataforma, es passa a definir quins rols i equips intervenen en cada fase del cicle de vida d'una API.

1. ***Definició***
   En aquesta fase, intervé l' ***equip proveïdor de les APIs***, juntament amb el ***Responsable APIs de l'Àmbit***, i poden intervenir altres equips en les definicions dels ***DA*** i ***DT*** que s' han de generar previ al desenvolupament i implementació.
2. ***Desenvolupament***
   En aquesta etapa, estaran involucrats l' ***equip proveïdor***, l' ***oficina tècnica d' API Manager*** i, possiblement, altres equips de CTTI, si cal realitzar algun tipus de configuració que no depengui de l' oficina tècnica, com, per exemple, obrir una regla de firewall o pujar fitxers necessaris per al desenvolupament als Datapowers.

   Dels rols per a una organització proveïdora, intervindran el rol d'***OT***, ja que entra l'equip de l'oficina tècnica per donar els accessos i permisos a l'equip proveïdor; el rol ***Developer***, ja que se'ls donarà aquest rol als desenvolupadors; i, possiblement, el rol ***Community Manager***, si es demana.
3. ***Proves***
   En aquesta etapa, intervindran l' ***equip proveïdor***, els ***equips consumidors***, l' ***equip de SIC***, atès que s' hauran de generar les pipelins que permeten als desenvolupadors el desplegament automàtic a la plataforma d' API Connect, i l' equip de l' ***oficina tècnica de l' API Manager*** en el cas que hagi d' aprovar alguna subscripció o per validar els plans i seguretat dels productes a desplegar.

   Dels rols per a una ***organització proveïdora,*** podran intervenir els mateixos rols que per a l' etapa passada. Al seu torn, podran intervenir ***tots*** els rols associats a una ***organització consumidora***.
4. ***Desplegament en Producció***
   En aquesta etapa, intervindrà l'***equip proveïdor***, que s'encarrega de desplegar les APIs via pipelins; l' ***oficina tècnica de l' API Manager***, en cas d' haver de realitzar o aprovar subscripcions i validar els plans i seguretat.

   Pel que fa als rols per a una ***organització proveïdora***, podran intervenir tots els que intervenen en les fases anteriors.
5. ***Manteniment***
   En l' etapa de manteniment, podran intervenir els ***equips consumidors***, l' ***equip proveïdor***, l' ***oficina tècnica d' API Manager*** i ***altres equips***, ja que, en cas que l' equip consumidor obri una incidència, aquesta pot requerir la intervenció d' aquests equips.

   Dels rols per a una ***organització proveïdora***, podran intervenir els mateixos rols que per a les etapes anteriors. Al seu torn, podran intervenir ***tots*** els rols associats a una ***organització consumidora***.
6. ***Retirada***
   En l'última fase del cicle de vida, intervindrà l'***equip proveïdor***, executant les conseqüents pipelins

per acabar retirant i eliminant les APIs. Al seu torn, podria intervenir de forma puntual l'***oficina tècnica d'API Manager***.

Pel que fa als rols per a una ***organització proveïdora***, podran intervenir els mateixos rols que per a les etapes anteriors.

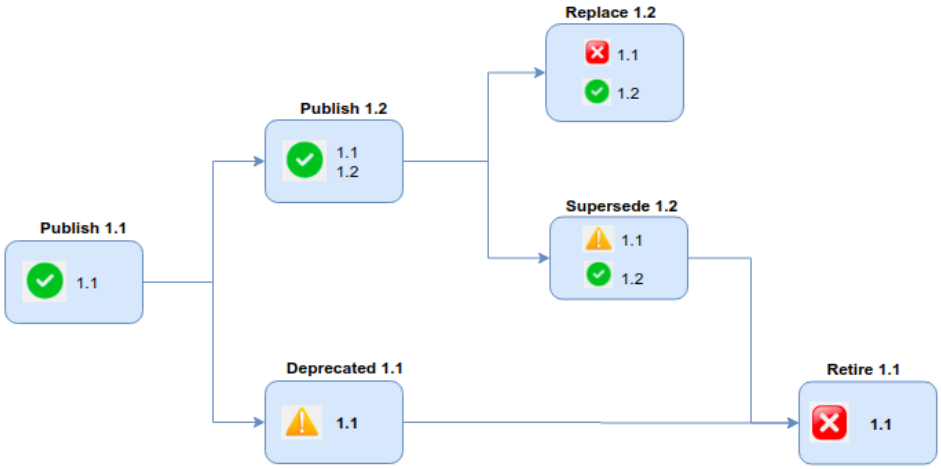
A continuació, es mostra una taula que resumeix l' esmentat.

Fase/Equips	Equip proveïdor APIs	Consumidors APIs	Oficina Tècnica APIM	Altres equips (SIC, CPD2, Seguretat…)
Definició	X			X
Desenvolupament	X		X	X
Proves	X	X	X	X
Desplegament en Producció	X		X	
Manteniment	X	X	X	X
Retirada	X		X	

3.3. Gestió del cicle de vida

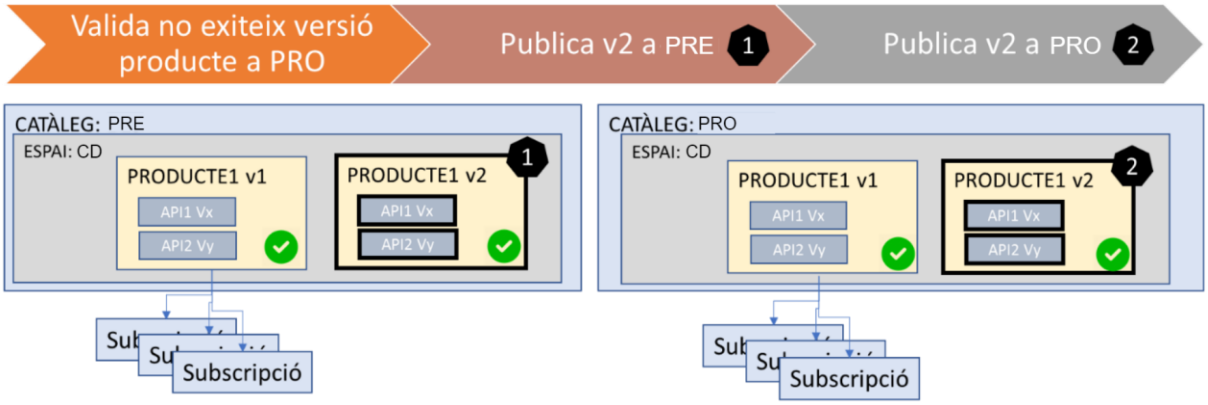
En API Connect, les APIs s'agrupen en Productes. El versionat va a nivell de producte i, per tant, les subscripcions es fan a productes.

El cicle de vida de l'API es gestionarà a través de pipelines al SIC, on s'oferiran tasques a nivell de producte:



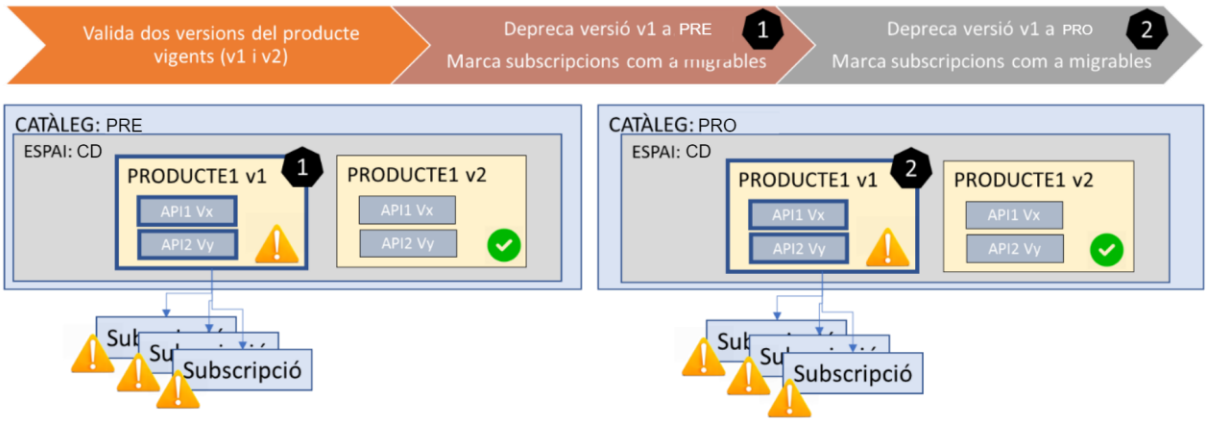
3.3.1. Publish

- Publicar una nova versió d'un Producte.
    - Si existeix a PRO la mateixa versió del producte -> falla.
    - Si existeix a PRE la versió del producte -> la sobreescriu.



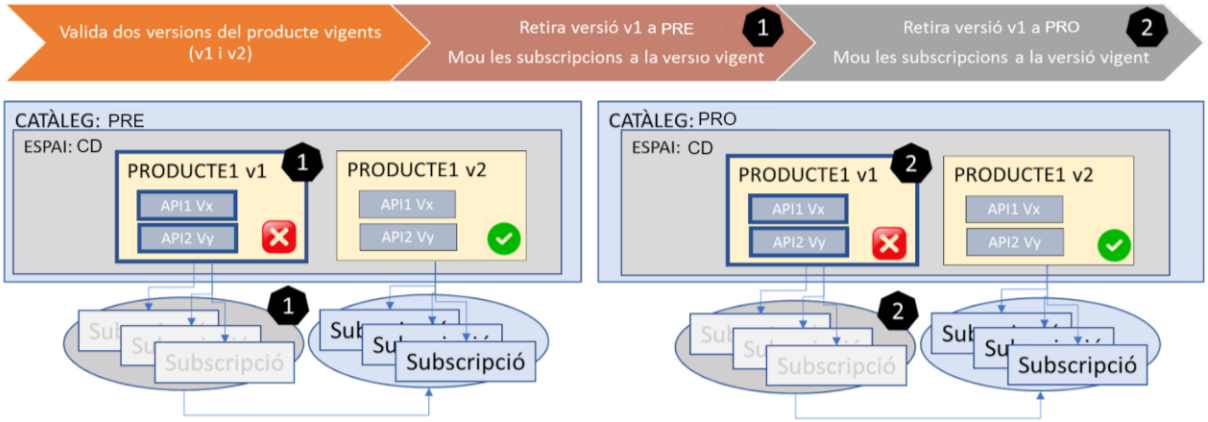
3.3.2. Supersede

- Deprecar una de les versions vigents del producte.
    - Posterior a un publish (2 versions vigents).
    - Depreca una versió i marca les subscripcions vigents com a migrated.
    - No es poden afegir noves subscripcions a la versió deprecada.



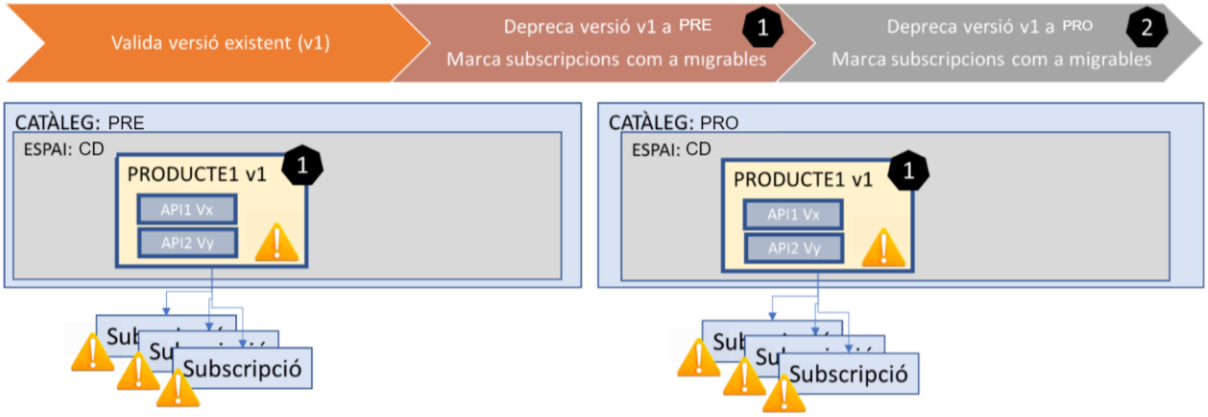
3.3.3. Replace

- Retirar una de les versions vigents del producte.
    - Retira una versió deixant una altra existent com a vigent.
    - Mou les subscripcions de la versió de producte retirat a la vigent.



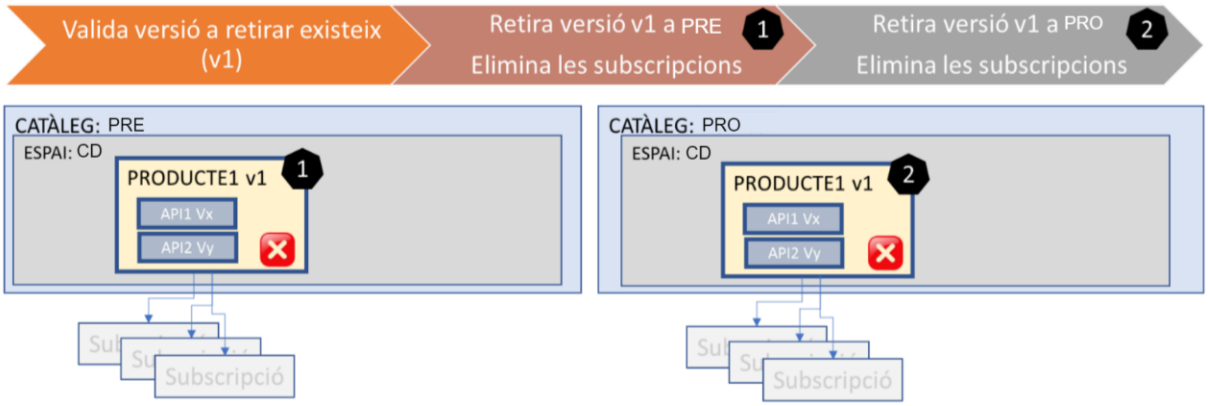
3.3.4. Deprecate

- Depreca una versió de producte sense deixar-ne cap com a vigent.
    - No es poden afegir noves subscripcions al producte.



3.3.5. Retire

- Retira una versió de producte sense deixar-ne cap com a vigent.
    - Les subscripcions al producte es perden.



4. METODOLOGIA DE DESENVOLUPAMENT

El CTTI aposta per una metodología ***DEVOPS*** que proporciona la màxima agilitat i autonomia als equips de desenvolupament per fer els desplegaments de forma automatitzada, millorant considerablement l'eficiència. Per això, tot el cicle de vida de les APIs està integrat amb la plataforma del SIC.

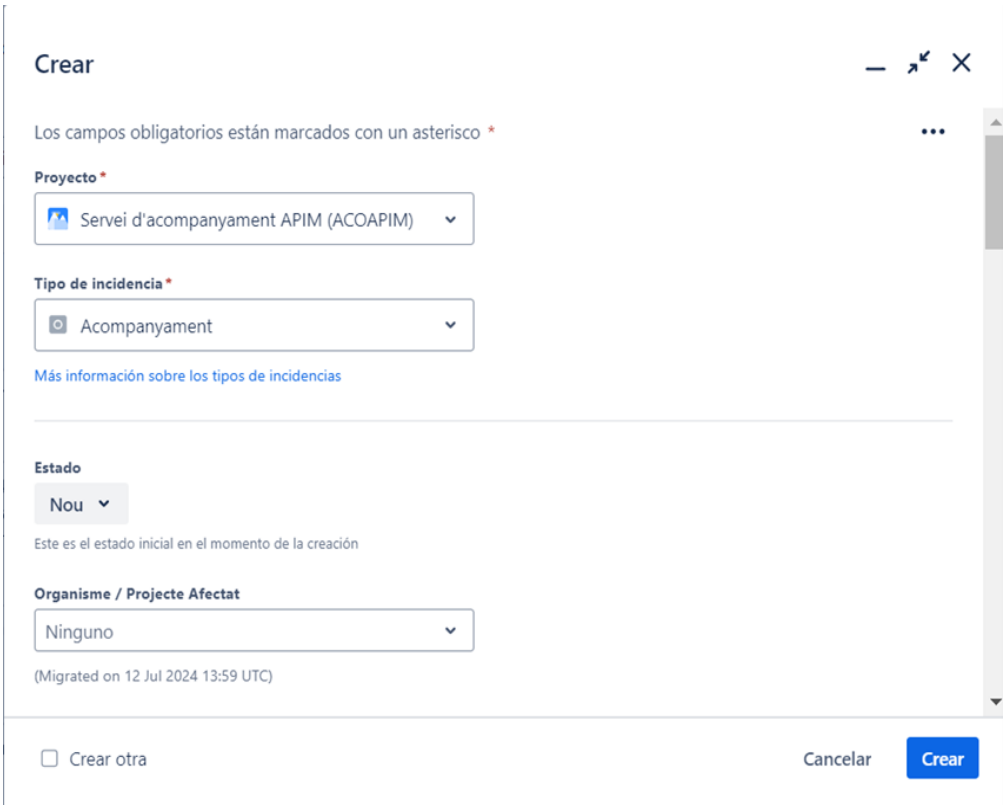
4.1. Sol·licitud d'acompanyament

Per les aplicacions en fase de projecte que s'integraran amb l'API Manager Corporatiu, la comunicació amb l'oficina tècnica d'API Manager s'ha de fer via ***CSTD*** al servei Servei Acompanyament APIM. El proveïdor d'aplicacions ha de crear una petició en aquest servei perquè l'oficina tècnica pugui configurar els requisits previs que pugui requerir el projecte per poder desplegar els seus APIs. Mentre l'aplicació estigui en fase de projecte tot el suport (Ex. configuració dels espais, publicació de l'API, consum API, …) es canalitzarà en aquesta petició.

Si no disposeu d'accés al JIRA, l'alta a JIRA es demana manant el portal ***PAUTIC*** de ***Remedy*** a la secció de ***Gestió accés d'usuaris***, sol·licitant accés a peticionar a ***ACOAPIM*** per poder obrir tiquets d'acompanyament de tot allò relacionat amb API Manager. També caldrà demanar accés per al projecte ***ACOCLDSIC*** per poder sol·licitar al SIC la configuració de les pipelins: el ***GDI*** podrà sol·licitar l'alta, la baixa i la modificació d'usuaris en l'aplicació. En el cas que se sol·liciti una petició relacionada amb un projecte JIRA serà necessari adjuntar un correu o document amb la validació del responsable del projecte. Qualsevol dubte que es tingui, es recomana contactar amb l'equip responsable via portal PAUTIC o via correu cstd.ctti@gencat.cat.

Un cop es té accés, caldrà fer el següent per sol·licitar un tiquet JIRA ACOAPIM:

- Accedir a la següent url: https://cstd-ctti.atlassian.net/jira/software/c/projects/ACOAPIM/issues
- Demanarà logejar-se primer, de manera que has d'introduir les credencials del teu usuari GICAR
- Es crea una petició nova sol·licitant l'acompanyament corresponent. S'haurien d'emplenar com a mínim els següents camps recomanats:
    - ***Organisme/Projecte Afectat***: Indicar el nom del projecte/organisme.
    - ***Si no localitzes el projecte en el desplegable, introdueix-lo aquí***: Indicar el nom del projecte a desplegar, si no es troba en el desplegable.
    - ***Resum***: Concepte de la petició, per exemple: "Sol·licitud d'accés a IBM API Connect".
    - ***Descripció***: Descripció detallada de la sol·licitud.
    - ***Codi de servei***: codi del servei per al qual es demana l' acompanyament.



Altres dades que es poden comunicar dins de la descripció o el resum per ajudar a agilitar l' aprovisionament i les configuracions inicials són:

- Informació relativa a les APIs:
    - Estar en disposició de les ***APIs a publicar***.
    - ***Endpoint PRE i PRO del backoffice*** associat a l'API a publicar, i saber en ***quin CPD*** estan les backends (per identificar necessitats de visibilitat).
    - ***On s'han de publicar*****:** Extranet/Intranet.
    - Particularitats de connectivitat, seguretat,etc.
        - ***Usuaris*** a qui donar ***accés a l'espai*** (i si algun requereix permisos de gestió de subscripcions).
        - ***Integració KeyCloak*** GICAR o particular.
- Es coneixen els consumidors de les APIs?

4.2. Desenvolupament

Els projectes han de realitzar el desenvolupament dels corresponents fitxers ***YAML*** de les APIs i Productes en format YAML. Per a això, es recomana l'ús de l'eina ***API Designer o Toolkit d'IBM***, el qual permet construir aquests fitxers amb el format i polítiques d'IBM d'una forma molt més còmoda i àgil, alhora que, combinant-la amb l'ús del ***Local Test Environment (LTE)*** d'IBM, podrien realitzar-se proves des de l'entorn local del desenvolupador.

Per instal·lar aquests productes, es proveeix una guia en la documentació (també es pot trobar més endavant en el punt 5.3), anomenada ***Guia Instal·lació i Configuració LTE de l'IBM API Connect***, en la qual es mostra com instal·lar i configurar ambdues eines.

A su vez, s'han establert pautes, plantilles i procediments per a homogenizar, facilitar, securizar i agilitzar el desenvolupament de APIs per part dels projectes dins de CTTI.

1. Homogeneïtzació de les ***regles de nomenclatura*** per a la definició de APIs dins de CTTI. Aquestes regles es poden trobar al document ***Estandards de Nomenclatura***, que es troba a la mateixa secció de documentació a ***Canigó*** que aquest document.
2. Disposició d'una ***plantilla basi*** per a estandarditzar i agilitzar la definició de APIs i productes dins de CTTI ***(especificació OpenAPI 3.0 i 2.0)***

3. Definició i incorporació de ***pautes de seguretat*** que han d'aplicar en cada tipologia i apificación. Per revisar amb més detall aquest tema, s'aconsella revisar el ***Manual de Seguretat*** que es troba a la mateixa secció de documentació a ***Canigó*** que aquest document.
4. Integració de noves capacitats implementades ***(polítiques i extensions personalitzades).***
    1. Les ***polítiques customizadas***, o ***USER DEFINED POLICIES*** dins de la terminologia d'IBM, són fragments de configuració dissenyats per a controlar un aspecte concret del procés en el servei de gateway durant el maneig d'una invocació de API en temps d'execució, satisfent així els requisits exclusius de cada projecte i permetent la personalització dels controls d'accés, seguretat i configuració depenent de la funcionalitat de cada política.
    S'han desenvolupat noves polítiques per a facilitar la creació i validació d'alguns components. Algunes d'aquestes polítiques fan usos d'unes certes propietats que estaran incloses en la plantilla del API, juntament amb una descripció del que fa cadascuna. Tota la informació sobre elles es pot trobar en el document de ***DT Politiques i Extensiones*** que es troba a la mateixa secció de documentació a ***Canigó*** que aquest document.

ID	Política	Descripció	Aplicació en CTTI
1	**Logs de Invoke** ***(ctti-invoke-log)***	Amb aquesta política podrem guardar en el log la request i response de la política de Invoke per a posteriorment ser enviats al Analytics. S'enviarà informació com la URL, capçaleres i bodi de la petició per al cas invoke-in (abans de la política invoke), i les capçaleres i bodi de la resposta per al cas invoke-out (després de la política invoke).	Amb l'ús d'aquesta política, es poden logear més etapes del flux, permetent a un desenvolupador revisar la petició abans i després d'enviar-la a les seves backends, amb el que permetrà identificar si la lògica aplicada en la seva API i el backend estan funcionant correctament, agilitzant la identificació i resolució d'errors, i ampliant la traçabilitat.
2	**Capçalera d'Arquitectura** ***(ctti-header-arch)***	A través d'aquesta política, es disposarà d'un estàndard de capçaleres custom a enviar en les peticions. Això resultarà útil en diversos contextos, com a l'hora d'informar l'identificador únic de la petició (UUID) a les capes d'integració subsegüents, o incloure el token utilitzat en l'autenticació per a la seva validació en nivells inferiors del sistema.	Es defineix una capçalera d'arquitectura per a totes les peticions que passen per API Connect. Aquestes capçaleres s'usaran per a controlar, negociar i millorar la interacció entre clients i servidors, així com per a garantir la seguretat i l'eficiència de les comunicacions.
3	**Validar IP Origen** ***(ctti-validate-IP) (no disponible***	Es tracta d'una política que permet controlar qui invoca o no a la API en funció de la IP origen que ve informada en la capçalera de la petició	Aplicant aquesta política, es reforça la capa de seguretat en les invocacions realitzades sobre les APIs, atès que permetrà controlar i restringir les IPs des

	en aquest moment)	"x-client-ip". Es validarà la IP d'origen contra la llista de IPs definides en la propietat corresponent del API. Si aquesta IP d'origen no es troba en la llista, es mostrarà l'excepció que correspongui.	de les quals s'invoca a una API concreta.
4	**Logs Custom** ***(ctti-custom-log)***	Aquesta política s'encarregarà de guardar el contingut de variables com una combinació de clau-valor dins del missatge que es mana per defecte al Analytics.	Aquesta política permetrà al desenvolupador extreure les informacions que consideri que és útil per a poder monitorar la funcionalitat del API.
5	**Gestió d'errors** ***(ctti-error-management)***	Política que proporciona un format d'error definit, amb la qual els errors que es generin després de la invocació de la API arribin en un format homogeni per a tots els casos. Format estàndard o customizado.	Amb aquesta política, es pot realitzar una estandardització del format dels errors generats per la API, bé en el format estàndard definit per a tot CTTI o bé en un customizado per al projecte, facilitant al desenvolupador implementar correctament una part crítica dels desenvolupaments com és la gestió d'errors.
6	**Recuperació de variables** ***(ctti-get-variables)***	Aquesta política s'usarà per a recuperar el valor de les variables emmagatzemades en diferents fitxers en el DataPower. En aquests fitxers es poden incloure tot tipus de variables (contrasenyes, usuari, endpoints…). En desenvolupar un API, podran recuperar-se els valors de variables llegint directament dels fitxers, indicant el fitxer i les variables que es necessiti accedir en les propietats corresponents de la API.	Aquesta política facilita el procés de retornar valors rellevants i sensibles que estiguin emmagatzemats en els DataPower, permetent d'aquesta manera que la informació sensible no viatge en tot el cicle d'execució ni aparegui en la definició YAML de la API, sinó que sigui recuperada de forma més segura.
7	**Validació d'entrada** ***(ctti-validate-request)***	Política usada per a validar l'esquema d'entrada contra l'esquema definit en el YAML de la API. Aquesta política hauria d'usar-se al principi de l'acoblat del API.	Aplicant aquesta validació, es reforça la capa de seguretat dels desenvolupaments, exigint que les crides entrants es realitzin amb el format esperat (Paràmetres, Querys…etc) definit en el YAML, per a evitar que puguin arribar als backends peticions mal formades o amb contingut maliciós, assegurant d'aquesta manera la qualitat de les APIs.
8	**Validació de sortida** ***(ctti-validate-response)***	Política usada per a validar l'esquema de sortida contra l'esquema definit en el YAML de la API. Aquesta política hauria d'usar-se al final de l'acoblat del API.	Aquesta validació, igual que l'anterior, reforça la capa de seguretat i ajuda a assegurar la qualitat de les APIs i dels backends.

2. Les ***extensions*** són mòduls de programari (lògica i codi) dissenyats per a intervenir en diferents fases del cicle de vida de les sol·licituds i respostes API, permetent-nos influir en el comportament de la nostra infraestructura de manera dinàmica. Les extensions dins del API Connect, igual que les polítiques, es desplegaran en els Gateways (Datapowers) però, a diferència d'aquestes últimes, les extensions sempre s'executaran de manera global per a totes les APIs desplegades en la plataforma.

S'han implementat unes noves extensions en la plataforma de API Manager per a facilitar la creació i validació d'alguns components que s'executaran de manera global per a totes les APIs desplegades en la plataforma. El seu comportament dependrà dels valors configurats en les propietats que es troben en l'especificació de la plantilla de API proporcionada als desenvolupadors. Tota la informació sobre elles es pot trobar en el document de ***DT Politiques i Extensiones*** que es troba a la mateixa secció de documentació a ***Canigó*** que aquest document.

ID	Política	Descripció	Aplicació en CTTI
1	**Pre-request** **Validació CORS**	Aquesta extensió valguda els CORS que li arriben per part del Front amb els CORS que la API tindrà configurada. En el cas que no coincideixi la informació que ve del Front amb la API, es mostrarà un missatge d'error. Es disposa d'una sèrie de propietats de CORS en la API, amb valors que hauran de definir-se abans de crear-la	Aquesta extensió reforça la seguretat de les APIs. S'aferma el control d'accés a les APIs, identificant si la invocació prové d'un lloc de confiança i prèviament registrat/identificat, evitant d'aquesta manera possibles forats en els accessos a les APIs
2	**Pre-request** **Generar Trace Id (UUID)**	Aquesta funció(extensió) generarà un UUID quan no vingui informat en la capçalera per part del Front. La informació que contindrà el UUID (identificat únic universal) ens servirà per a poder fer un seguiment complet del flux d'una crida	Gràcies a aquesta extensió, es disposa de la traçabilitat de les invocacions al llarg de tot el flux d'execució des del API fins al backend, permetent identificar-la i separar-la d'entre la resta de les traces de log, fent un seguiment més eficaç i ràpid que permet analitzar i identificar qualsevol error o informació rellevant
3	**Post-response** **Gestió de capçaleres de seguretat**	Seguint l'estàndard OWASP Secure Headers Project, també anomenat OSHP, eliminarà les capçaleres que hagin de ser eliminades i generarà les capçaleres de seguretat adequades sobre la base de l'estàndard	Aquesta extensió reforça la seguretat de les APIs, aplicant l'estàndard de seguretat OWASP de manera que ens assegurem que no s'envia de tornada als sol·licitants cap informació sensible o no desitjada en la resposta

5. S'han incorporat pautes per a la utilització de manera efectiva de l'eina ***toolkit d'IBM*** amb totes les noves capacitats implementades per a l'ús dels ***desenvolupadors*****.** Aquestes pautes es poden trobar en el document de ***Manual d'importació de polítiques*** que es troba a la mateixa secció de documentació a ***Canigó*** que aquest document.
6. Guia detallada de ***com desenvolupar un API*** des de zero en local, a través de l'eina toolkit. Es pot trobar aquesta guia al document anomenat ***Guia de desenvolupament***, ubicada a la mateixa secció de documentació a ***Canigó*** que aquest document.
7. ***Estandardització dels plans de consum***. De cara a estandarditzar els plans de subscripció dels productes que poden usar els desenvolupadors dins de CTTI, s'han definit i acordat els següents tiers, amb els següents límits màxims per a cada tier:

Pla	Límit màxim Rate	Límit màxim Burst
Default Plan	100/h	10/min
Bronze	1000/h	100/min

Silver	2000/h	200/min
Gold	4000/h	400/min

Per tant, els projectes dins de CTTI hauran d'acollir-se a un d'aquests plans. Els límits establerts són els límits màxims per a aquest tier, podent cada tier tenir un valor entre 1 i el valor màxim que apareix a la taula per a cada límit.

Els plans, per tant, hauran de seguir la següent nomenclatura:

- **Nom:** el nom del pla haurà de començar per un dels quatre tiers i estar en minúscules, separant cada paraula per guions. En cas que es vulgui afegir alguna cosa més que el nom del tier, s'haurà d'introduir a continuació, després d'un guió. Exemples: gold-auth, default-plan-users, silver-external-clients, etc.
- **Títol**: títol haurà de ser igual que el nom, cada paraula començar en majúscula i ha de contenir espais entre cada paraula en cas d' existir.
- **Descripció:** descripció del pla amb prou detall per ser fàcilment entendible, si és necessària. També es pot deixar amb un valor igual al títol.

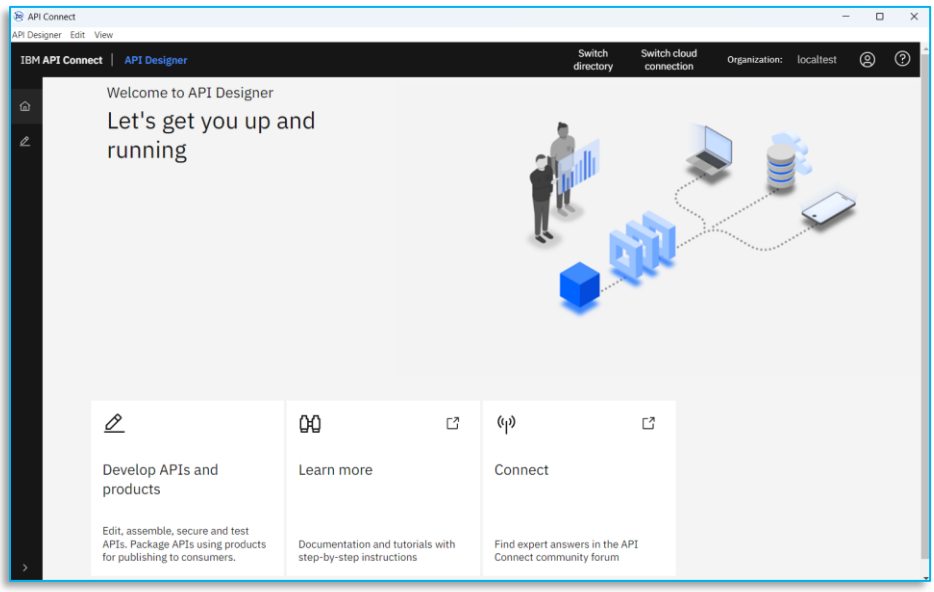
```
plans:
  gold:
    title: Gold
    description: Gold
```

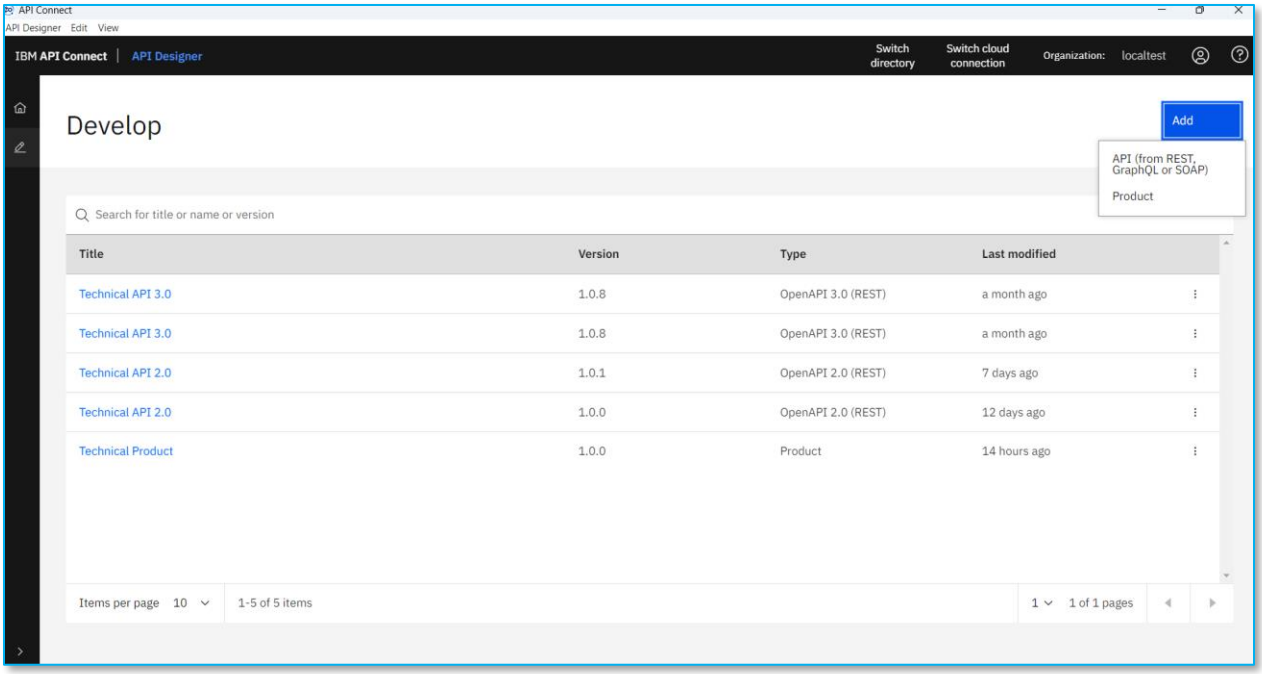
En cas de necessitar-ho, podran sol·licitar una excepció, via petició Jira a la OFT.

4.2.1. Exemple desenvolupament API i producte des de Toolkit

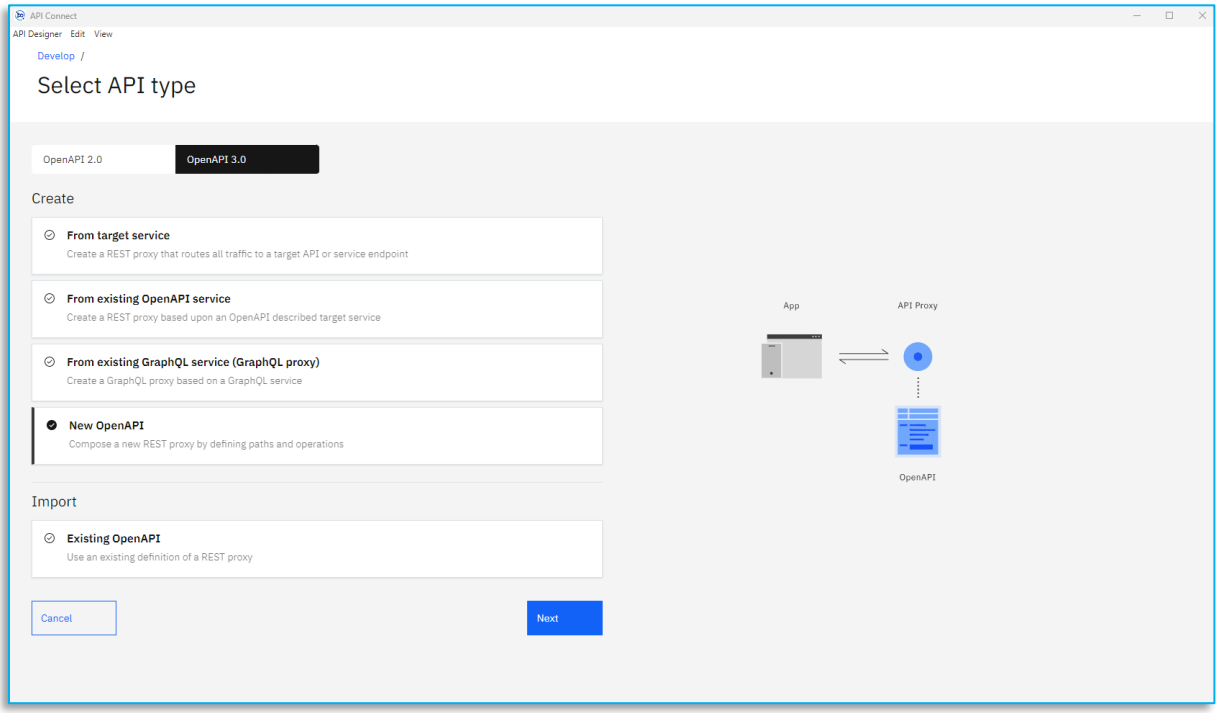
Es procedeix a mostrar com desenvolupar un API des de zero usant l'eina de API Designer/Toolkit d'IBM.

Primer, s'accedeix a l'eina del Toolkit en local. Una vegada en la finestra principal, es prem en “Develop APIs and products”, per a començar amb el desenvolupament de la API.

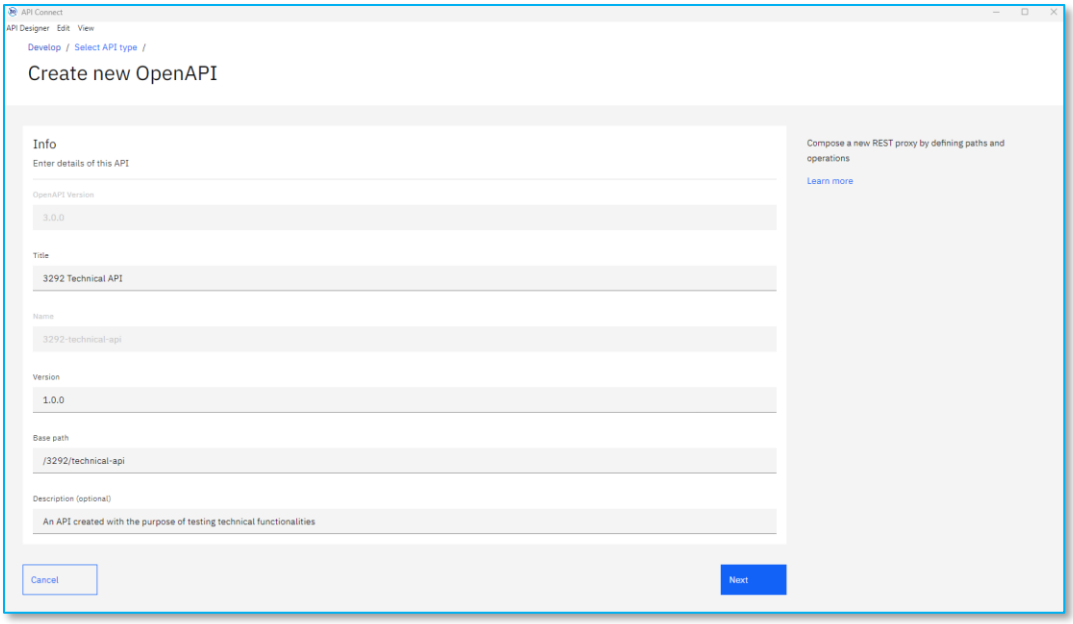




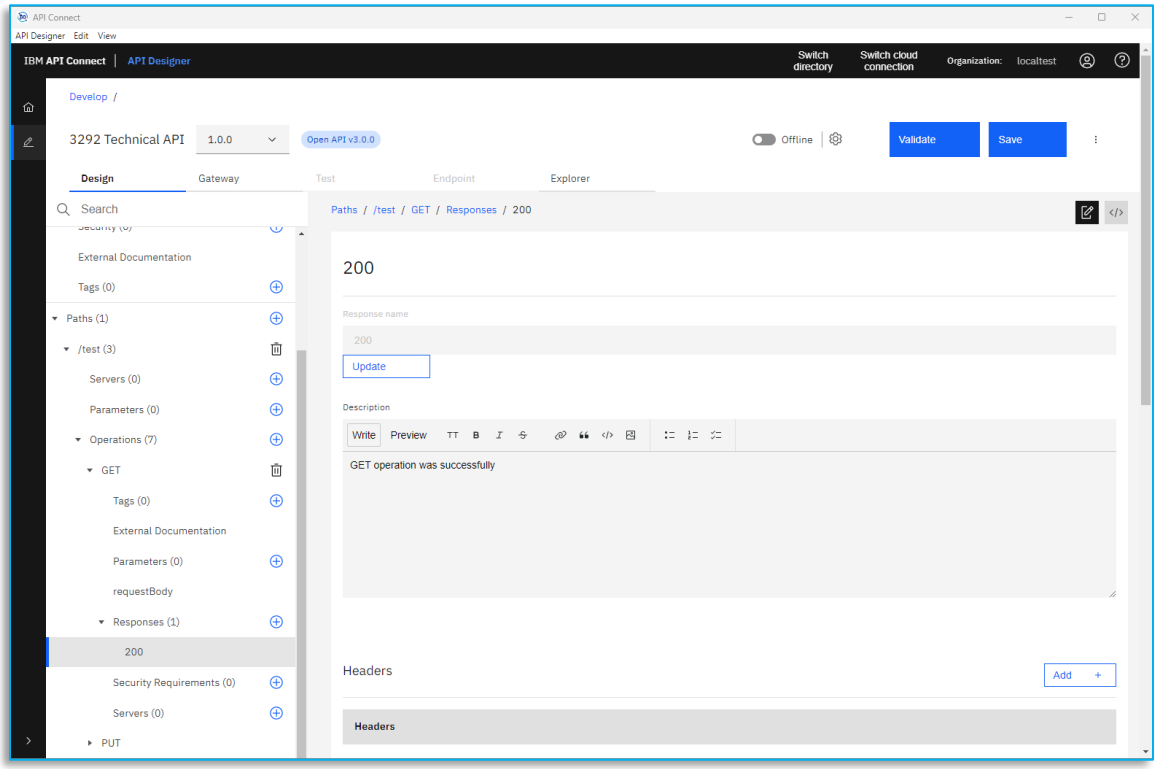
A continuació, s'ha de seleccionar el tipus de API que volem construir. En aquest exemple, se selecciona ***New OpenAPI*** perquè sigui de tipus REST, i li donem a ***Next***.



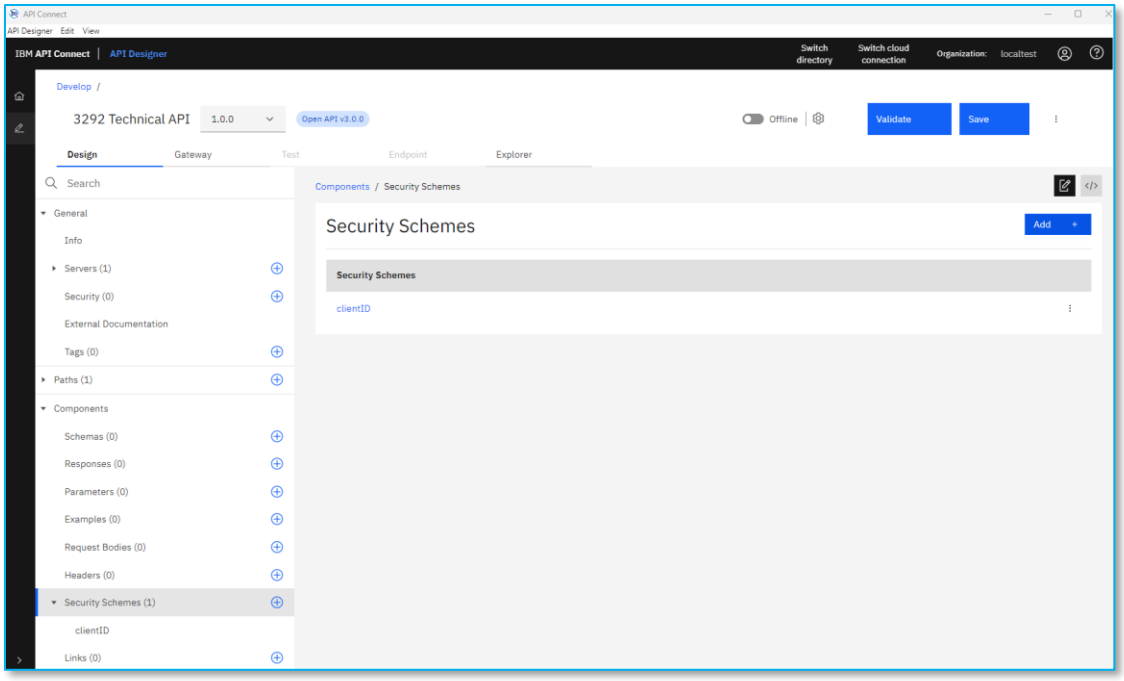
Després de continuar, s'haurà d'emplenar el nom de la API en el camp ***Title***, i automàticament s'emplenarà el Base path, el qual es pot modificar. La versió s'ha de deixar com apareix, ja que seria la 1.0.0.



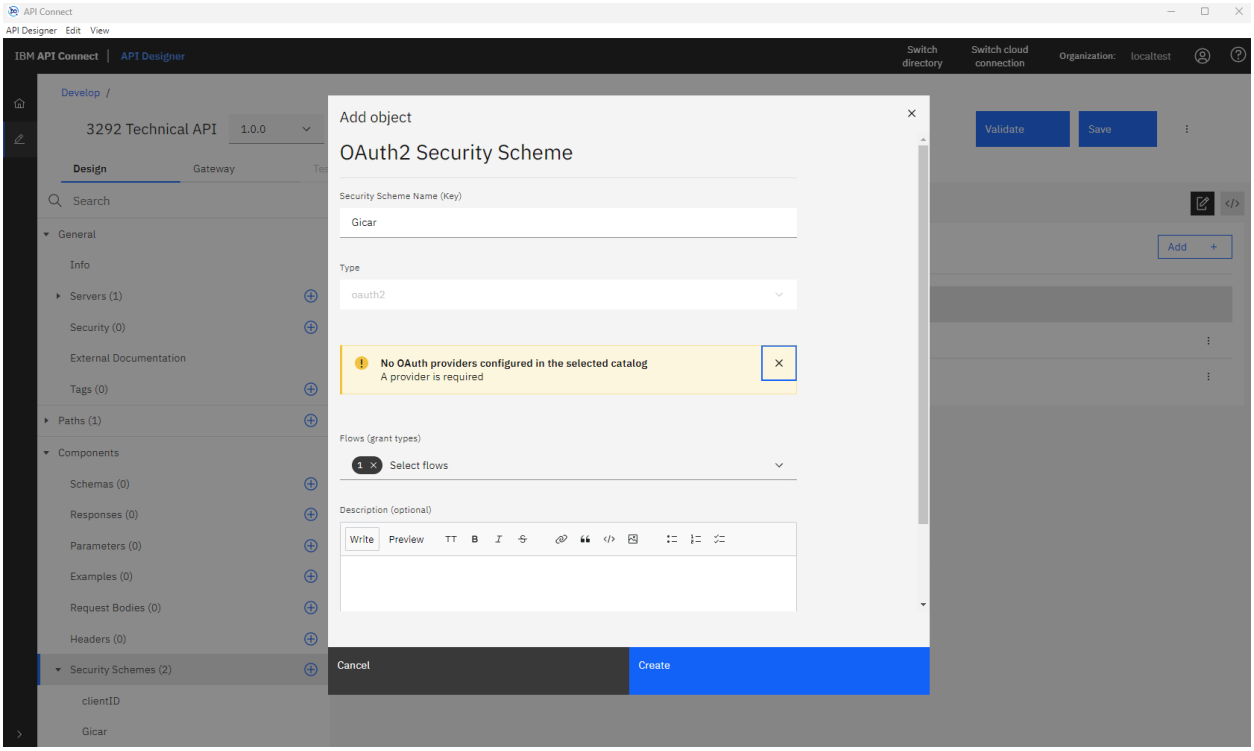
Les operacions es generen de manera automàtica per al path que ve definit per defecte. Dins de cada operació, com ara GET, es poden definir les respostes en cas de OK o de KO.



Una vegada definit el path amb les seves operacions, es procedeix a definir els esquemes de seguretat en la secció ***Security Schemes***. Per a això, es prem en el botó ***Add***.

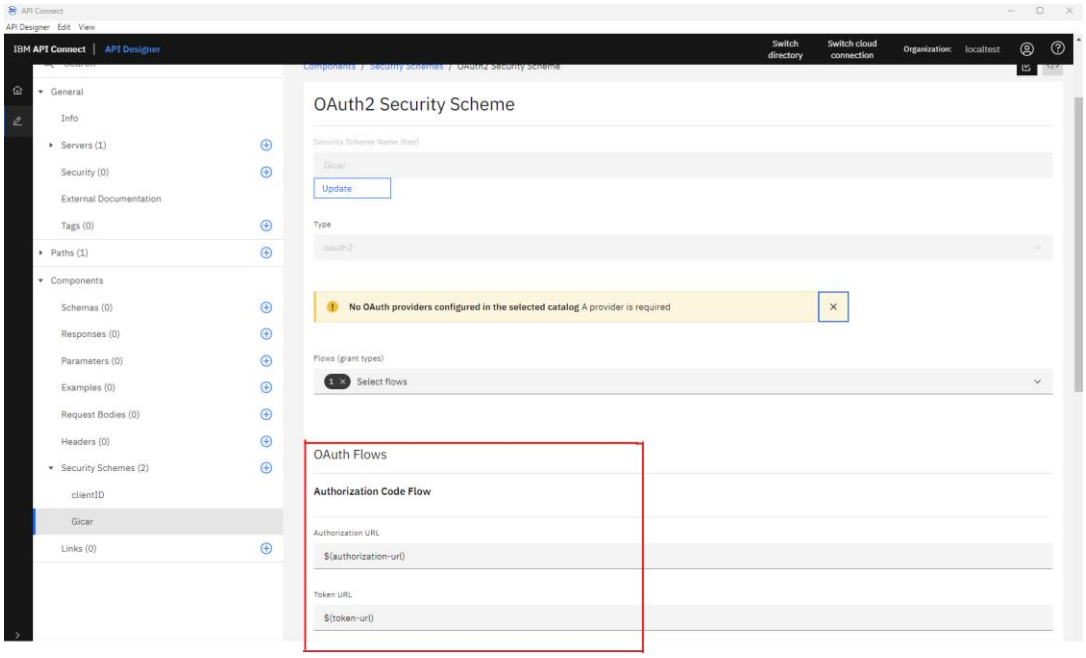


Es configura el protocol de seguretat OAuth2 per a securizar la API. Per a això, es tria el tipus de seguretat ***oauth2***, s'emplena dins del camp Name (Key) el valor que es desitgi, en aquest cas ***Gicar*** i en el camp Flow, es tria el flux de ***AccessCode***. Després d'emplenar aquests camps, es prem en Create.



***Nota**: Es pot veure que per pantalla es mostra l'alerta de "No OAuth providers configured in the selected catalog". Continuarem i aquest punt el deixarem així ja que es tractarà més endavant.*

Una vegada creat, s'emplenen la resta dels camps corresponents. El valor de cada camp dependrà del proveïdor de OAuth que es vagi a usar.



A continuació, es presenta un exemple de com s'emplenen els camps:

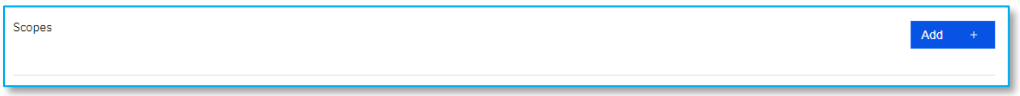
- *En el camp Authorization Url, s'estableix el valor: $(authorization-url)* ***(*)***
- *En el camp Token Url, s'estableix el valor: $(token-url)* ***(*)***
- *El camp Description es pot deixar en buit.*

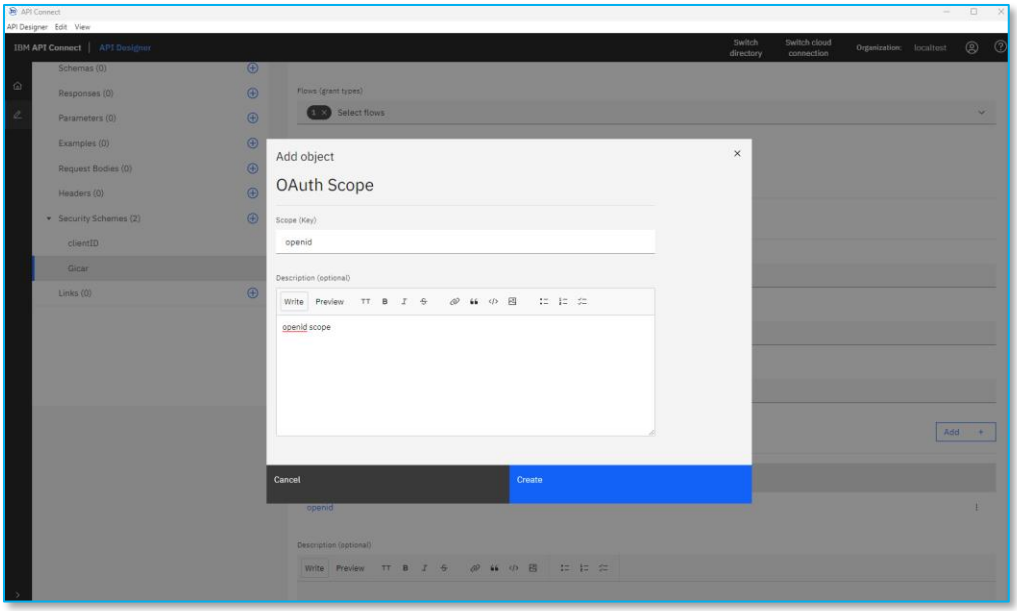
***** *Els valors es parametritzen com a* ***variables****, ja que posteriorment s'acte-setean pels valors definits en les* ***propietats*** *o* ***propietats de catàleg*** *corresponents. S'explica això en una diapositiva posterior.*

Després d'emplenar aquests camps, s'han d'informar els ***Scopes*****.**

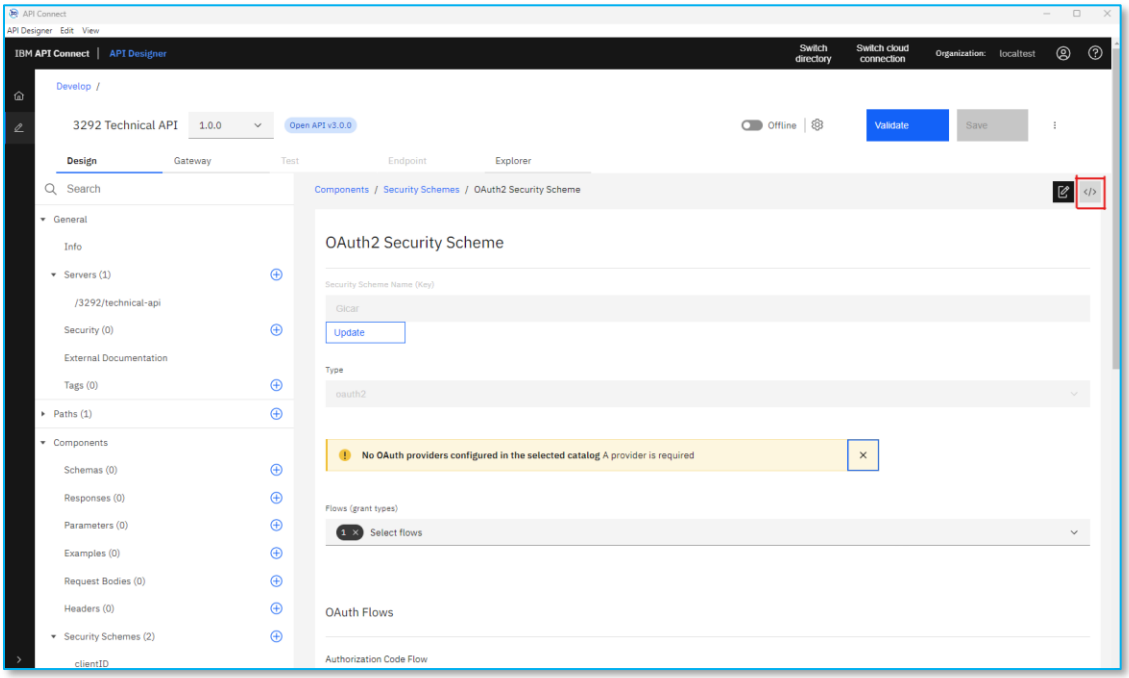
Dins del OAuth2 s'han de definir els Scopes necessaris, si n'hi hagués. En aquest cas, s'han configurat les següents Scopes:

- ***openid: openid scope***

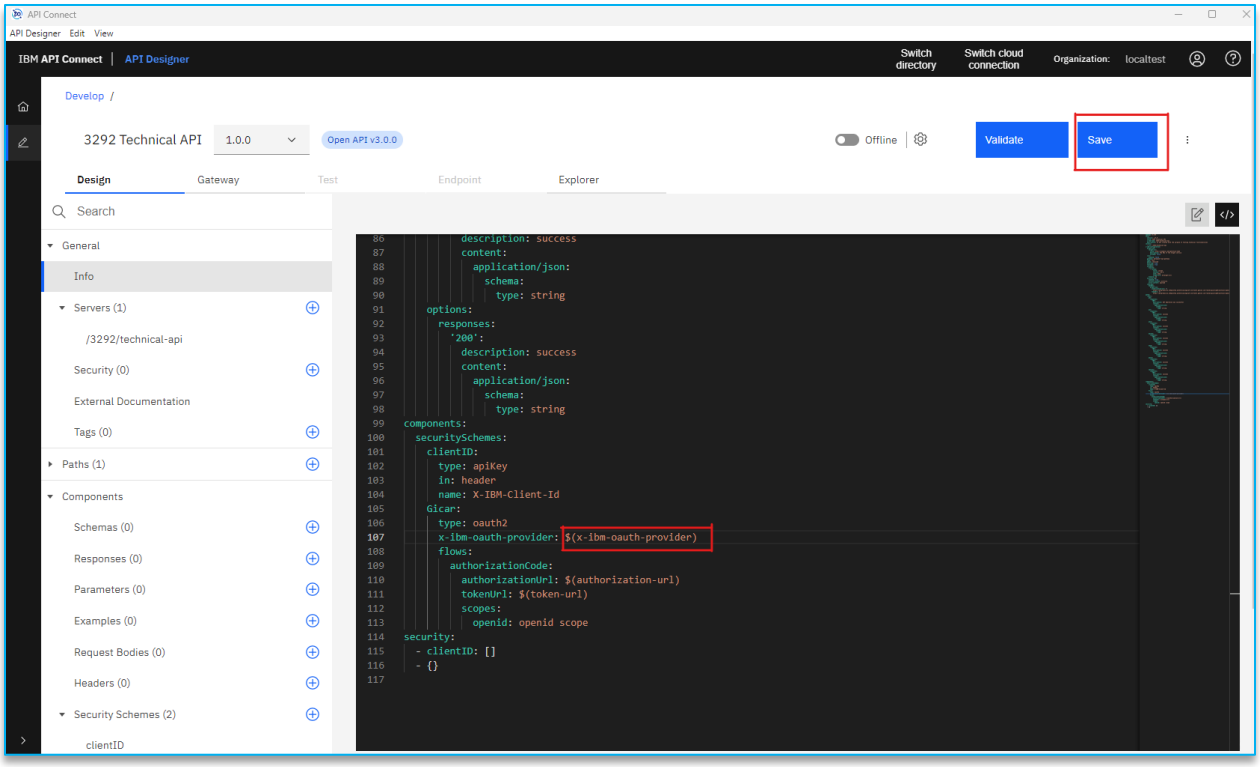




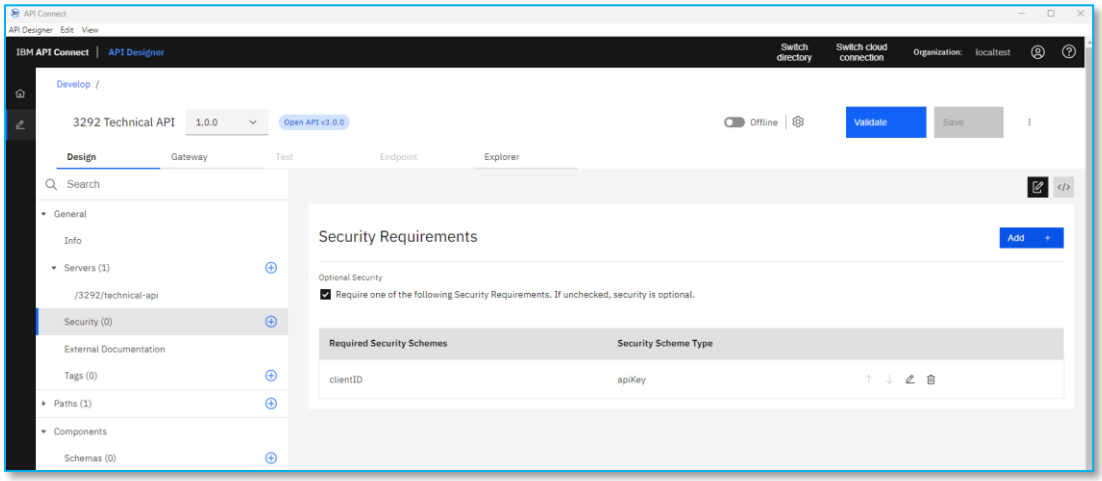
Per a llevar l'alerta que es mostra per pantalla de "*No OAuth providers configured in the selected catalog*", s'ha d'accedir al YAML del API mitjançant el botó ***Source*.**



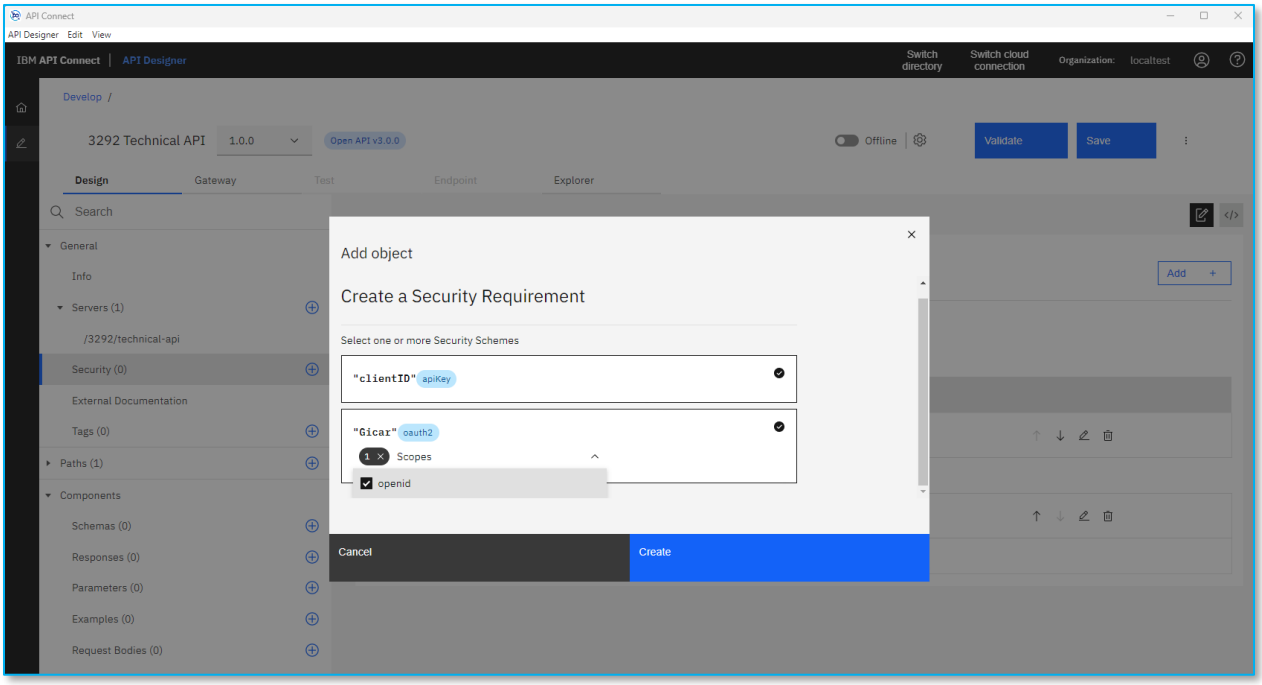
Per sota de la línia ***"type: oauth2"*** posem el proveïdor de OAuth que es vagi a usar. Per al nostre exemple, s'ha indicat la següent línia de text en el YAML: ***x-ibm-oauth-provider: $(x-ibm-oauth-provider)***. Posteriorment li donem al botó de ***Save*** per a guardar els canvis.



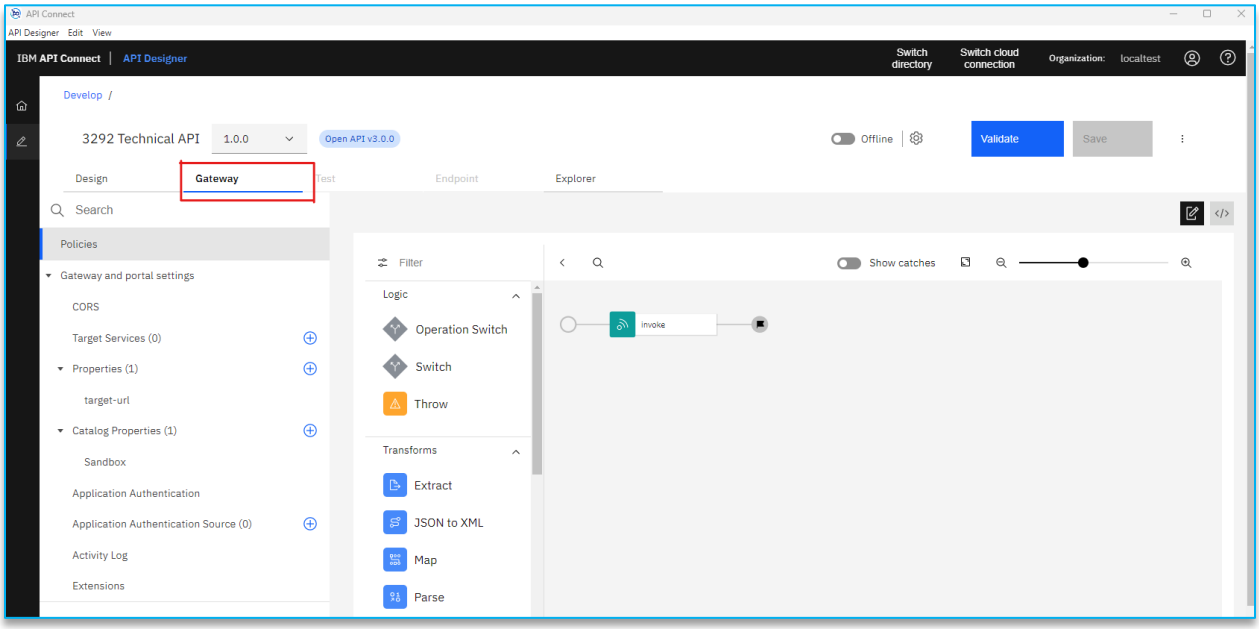
Una vegada definit l'esquema de seguretat, es procedeix a configurar la API amb aquest esquema. Per a això, s'accedeix a la pestanya ***Security*.**



Es prem sobre el botó ***Add*** per a afegir el nou protocol de seguretat. Se selecciona la combinació de ***ClientID + OAuth2*** (amb el Scope que definim abans seleccionats) i es prem en el botó ***Create***.



Una vegada que s'ha acabat de dissenyar les configuracions bàsiques del API, es procedeix a dissenyar l'acoblat. Per a això, cal dirigir-se a la pestanya ***Gateway*.**



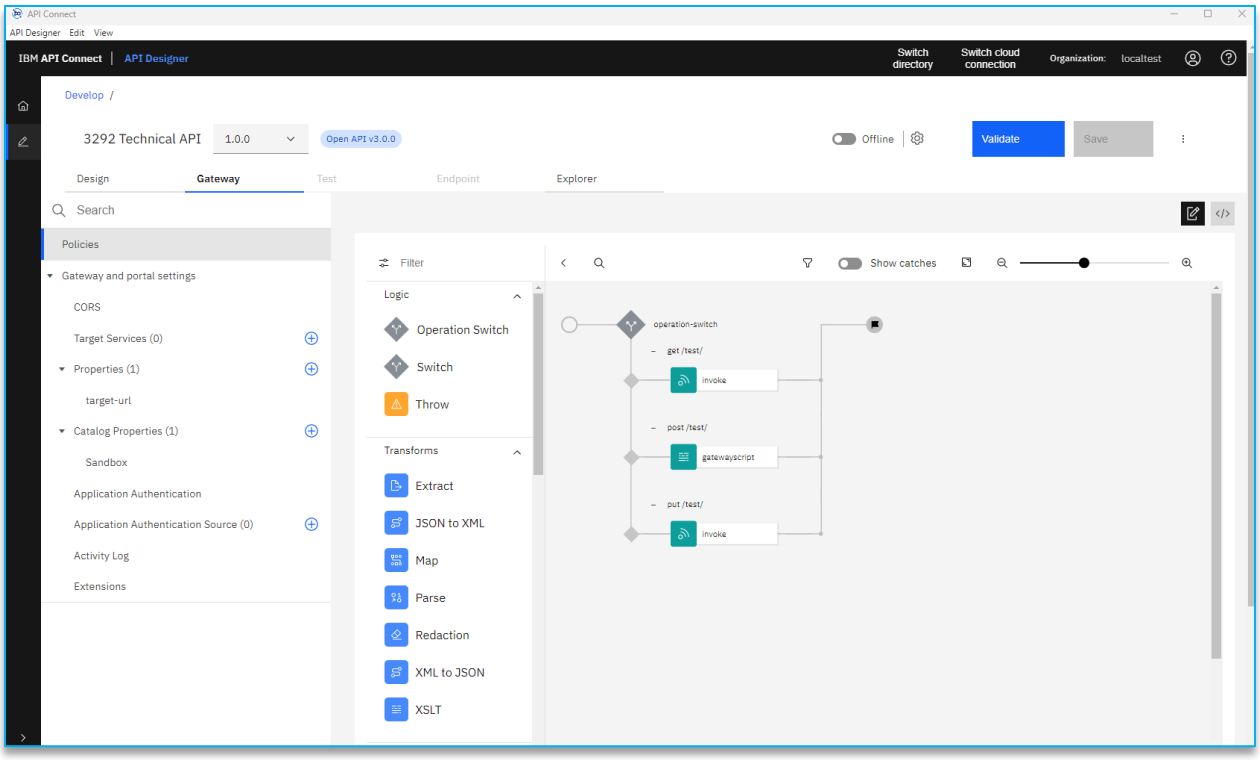
Dins de l'acoblat es poden incloure les polítiques que es considerin necessàries, d'entre les que pot ser:

- ***Invoke***: Per a cridar a un altre servei des de l'acoblat.
- ***Map***: Per a aplicar transformacions al flux d'acoblat i especificar relacions entre variables.
- ***GatewayScript***: Per a executar un programa de Datapower GatewayScript, codificat per l'usuari.
- ***SetVariable***: Per a agregar una variable d'encapçalat o llevar/establir una variable de temps d'execució.

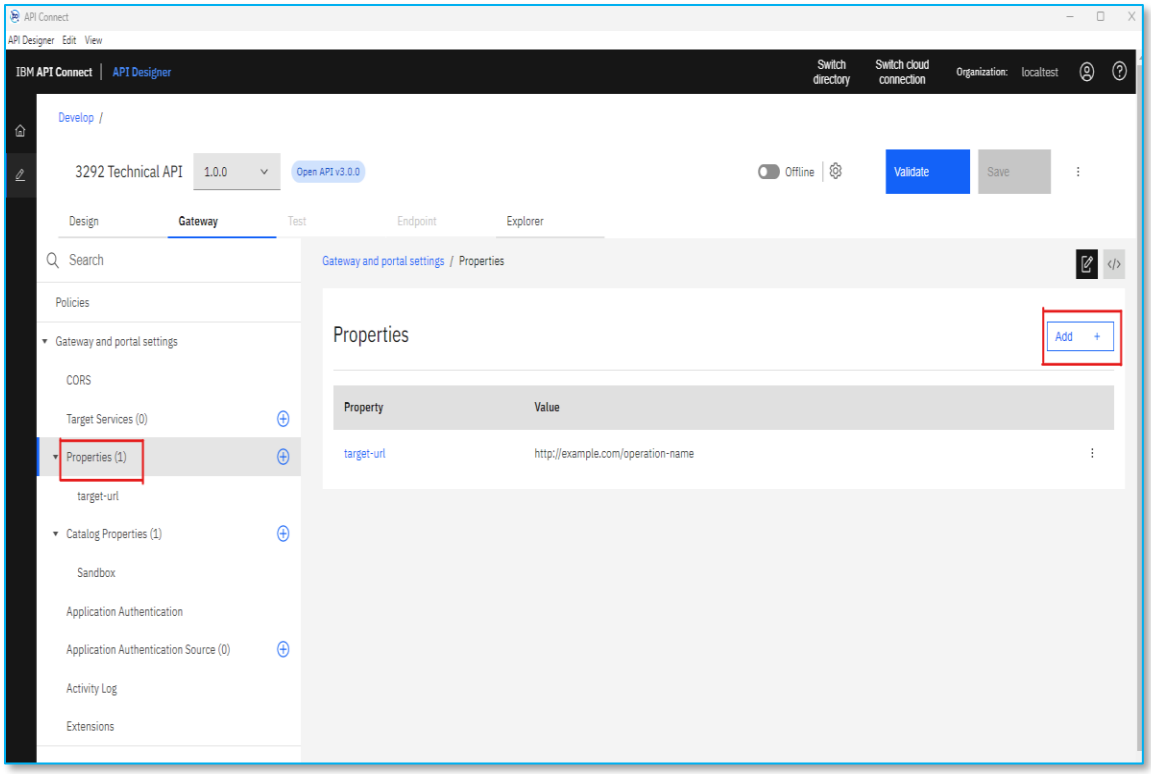
Per a consultar les informacions sobre l'ús i funcionament de les polítiques, revisar la documentació oficial d'IBM.

En aquest exemple, s'han definit les següents polítiques per a l'acoblat:

- ***Invoke (GET)*:** Redirigeix a la pàgina especificada en la variable target-url en realitzar una anomenada al API amb l'operació GET.
- ***GatewayScript (POST)*:** S'executa un script que genera com a resposta el missatge de 'ok' en realitzar una anomenada al API amb l'operació POST. El seu contingut és el següent:
- ***Invoke (PUT)*:** Redirigeix a la pàgina especificada en la variable target-url-1 en realitzar una anomenada al API amb l'operació PUT.

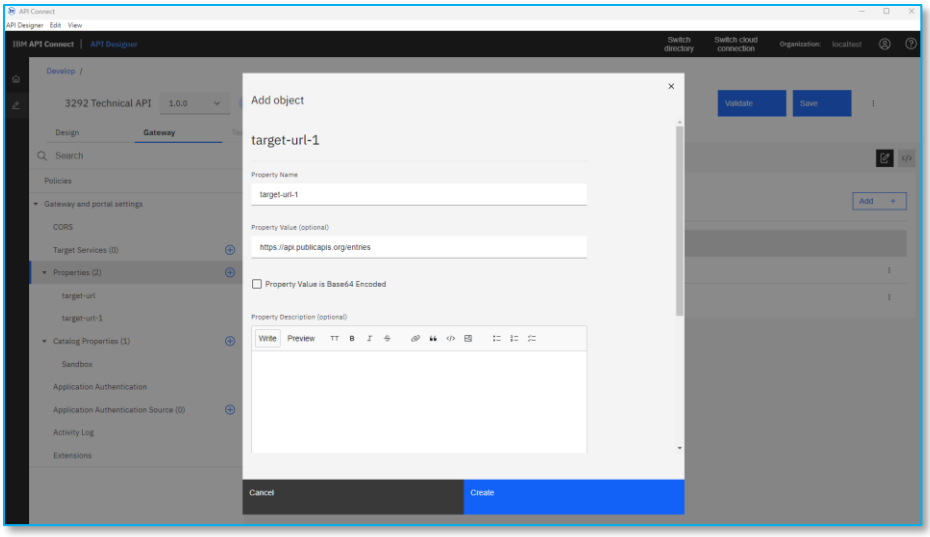


Perquè els invokes de l'acoblat anterior puguin funcionar correctament, es defineixen els valors de les variables ***target-url*** i ***target-url-1***. Per a això, cal dirigir-se a Properties i prémer en el botó ***Add*.**

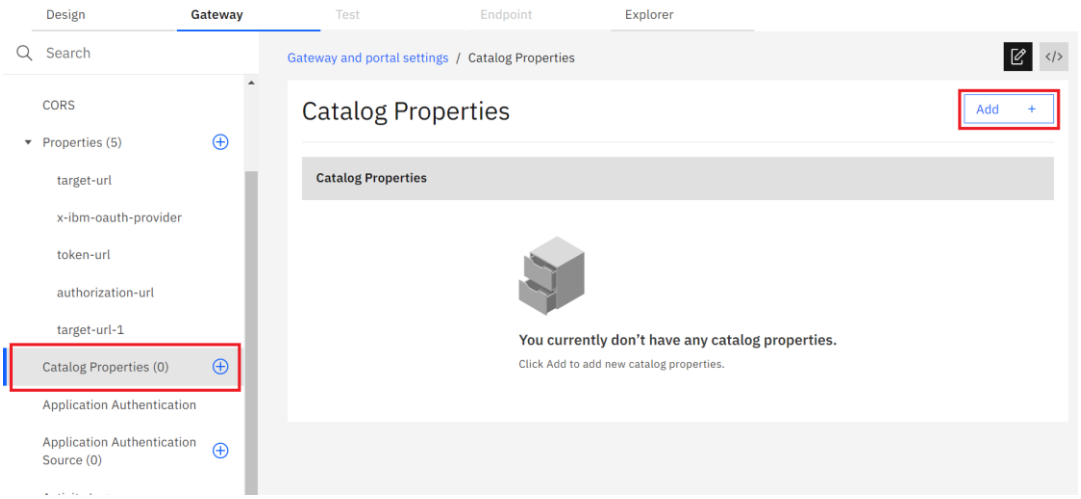


Es crea la variable ***target-url-1*** i s'introdueix la ***url*** a la qual es vol redirigir el API com a valor de la variable (Property Value), així com altres variables com a ***x-ibm-oauth-provider***, ***token-url*** i ***authorization-url***. En aquest cas es defineixen els següents valors per a les propietats.

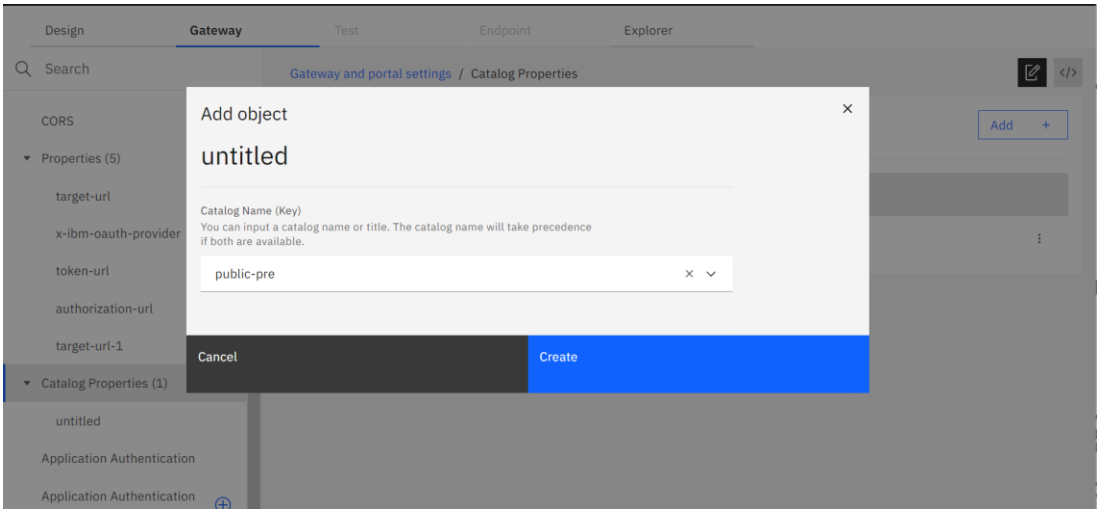
- ***target-url-1****: https://api.publicapis.org/entries*
- ***token-url****: https://preproduccio.endpointma.autenticaciogicar4.extranet.gencat.cat/realms/gicarcpd4/protocol/openid-connect/token*
- ***authorization-url****: https://preproduccio.endpointma.autenticaciogicar4.extranet.gencat.cat/realms/gicarcpd4/protocol/openid-connect/auth*
- ***x-ibm-oauth-provider*****:** *apic-keycloak-cpd4*



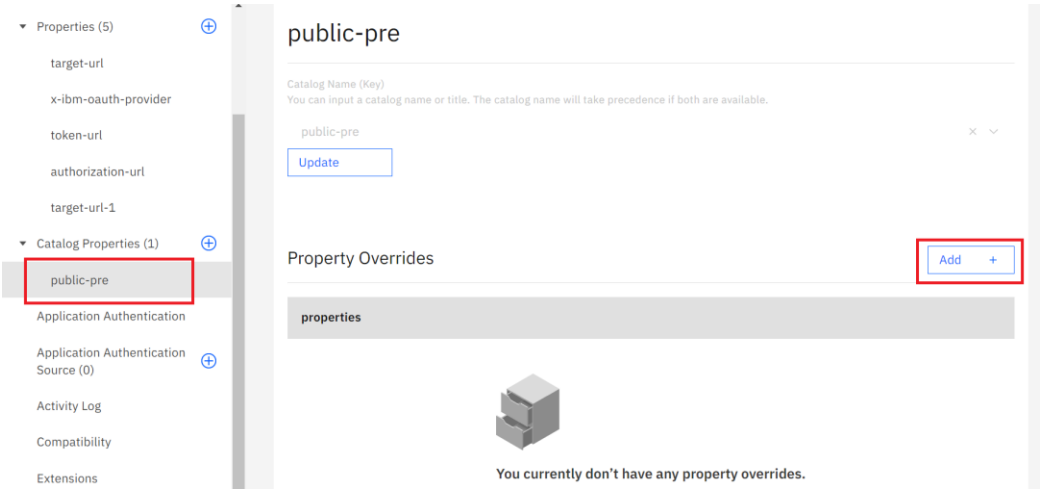
El API permet configurar diferents valors per a les mateixes propietats en funció del catàleg (entorn) on es despleguin. D'aquesta manera, es pot desplegar un sol YAML tant en el catàleg de PRE com el de PRO, usant en cada entorn els valors corresponents a aquest entorn. Per a configurar això, primer s'ha d'anar a la secció de ***Catalog Properties*** i prémer en ***Add*****.**

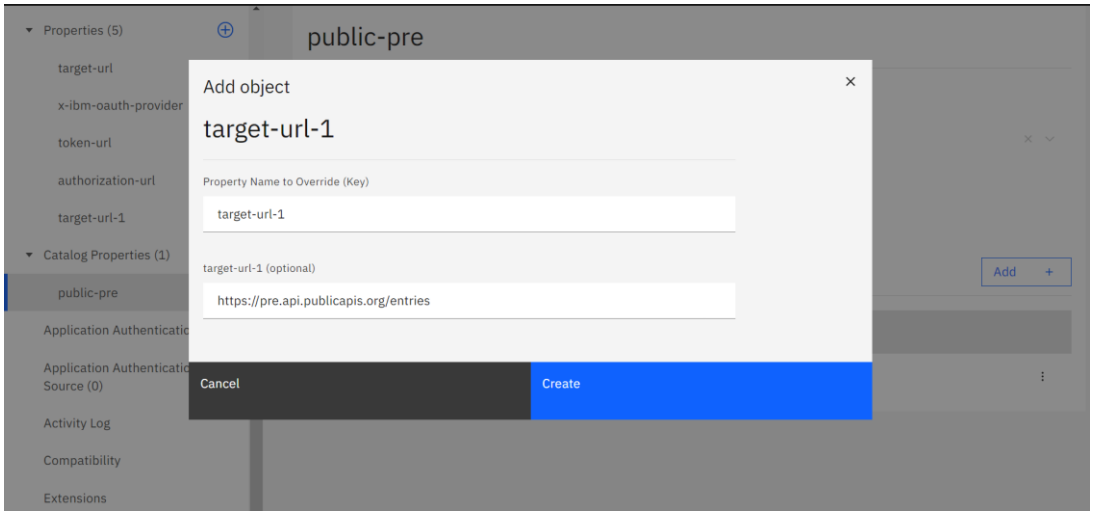


A continuació, s'insereix el nom del catàleg corresponent que es vulgui configurar i es prem en ***Create***. En aquest cas, es tria configurar public-pre.

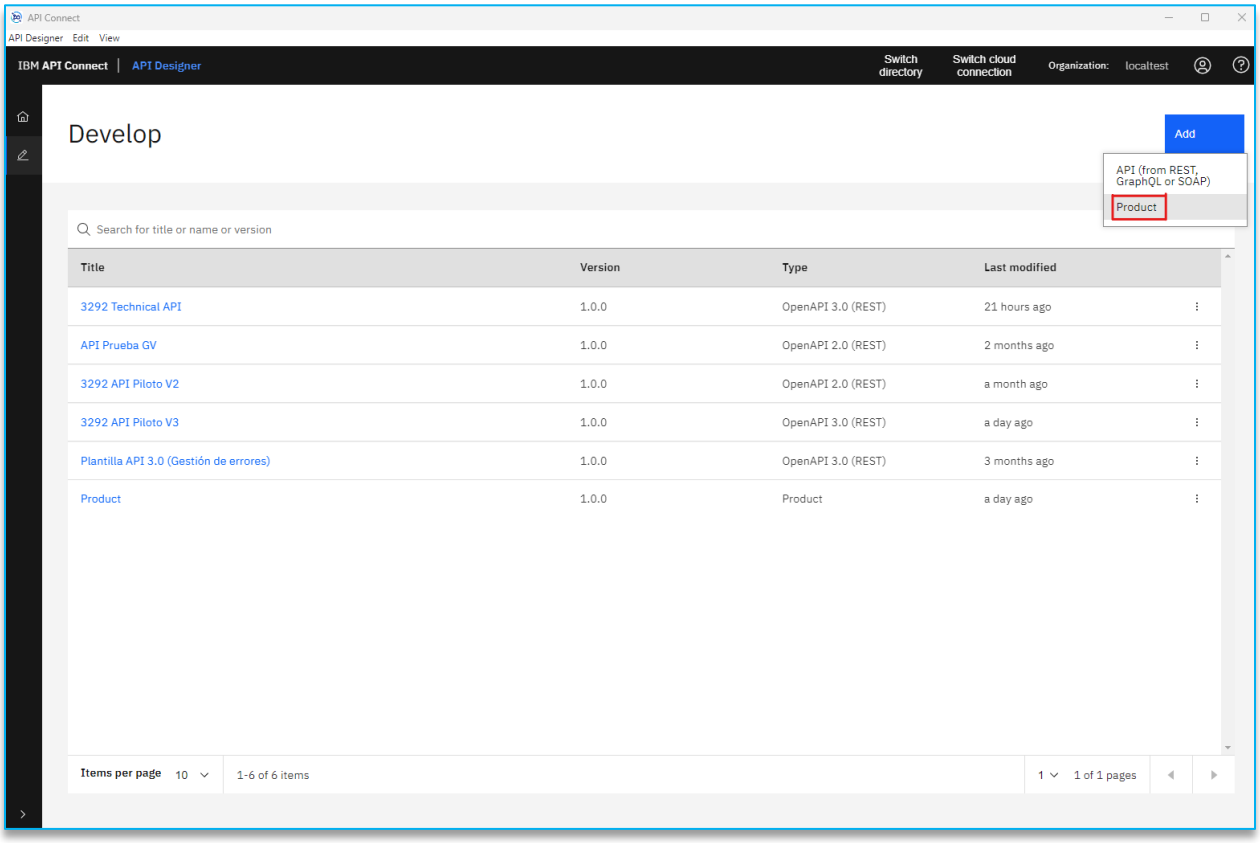


Una vegada s'hagi creat aquesta secció, es prem en ***Property Overrides -> Add*** i s'afegeix, en el camp ***Property Name to Override***, el nom d'una propietat ***existent (requisit imprescindible)*** el valor del qual ha de ser diferent quan es desplegui en el catàleg de public-pre. En el camp següent, s'estableix el seu valor per a aquest catàleg. En aquest cas, es posa d'exemple la propietat ***target-url-1***, que adquirirà el valor de ***https://api.publicapis.org/entries*** quan la API es desplegui en qualsevol catàleg menys quan es desplegui en ***public-pre***, que llavors obtindrà el valor de ***https://pre.api.publicapis.org/entries***.

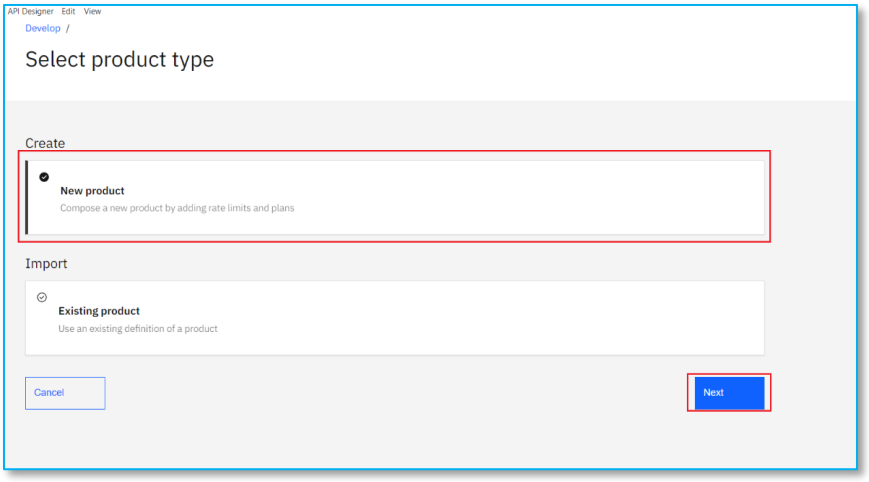




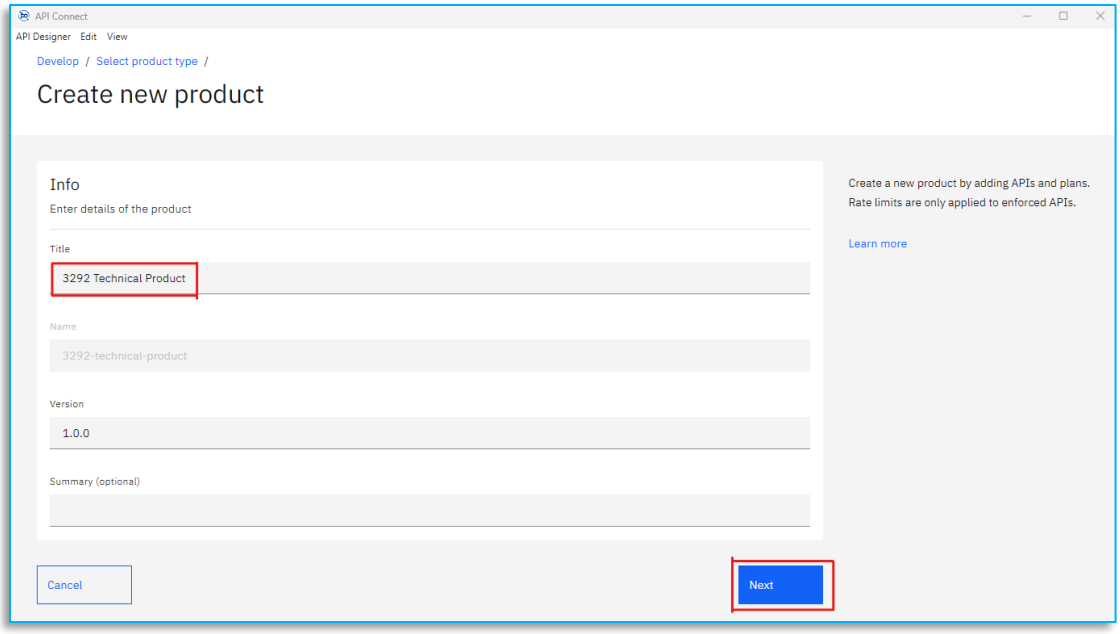
Després de finalitzar el disseny del API, es procedeix amb la creació del producte. Per a això, es torna a la secció principal prement sobre al botó  i posteriorment es prem en els botons ***Add – Product***.



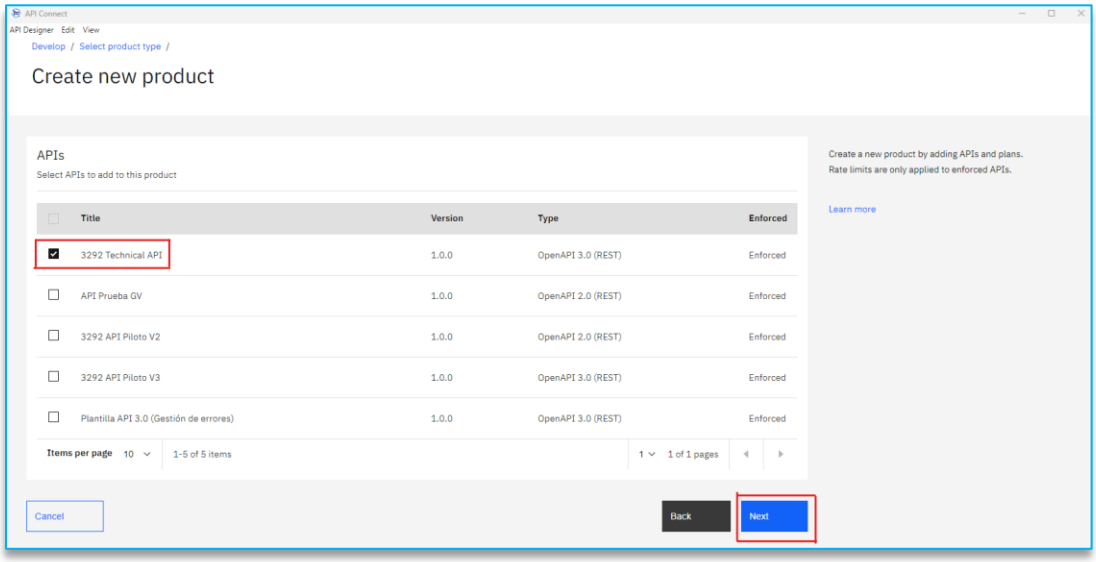
Seleccionem ***New product*** i li donem a continuar.



S'introdueix el títol del producte i se li dona a ***Next***.

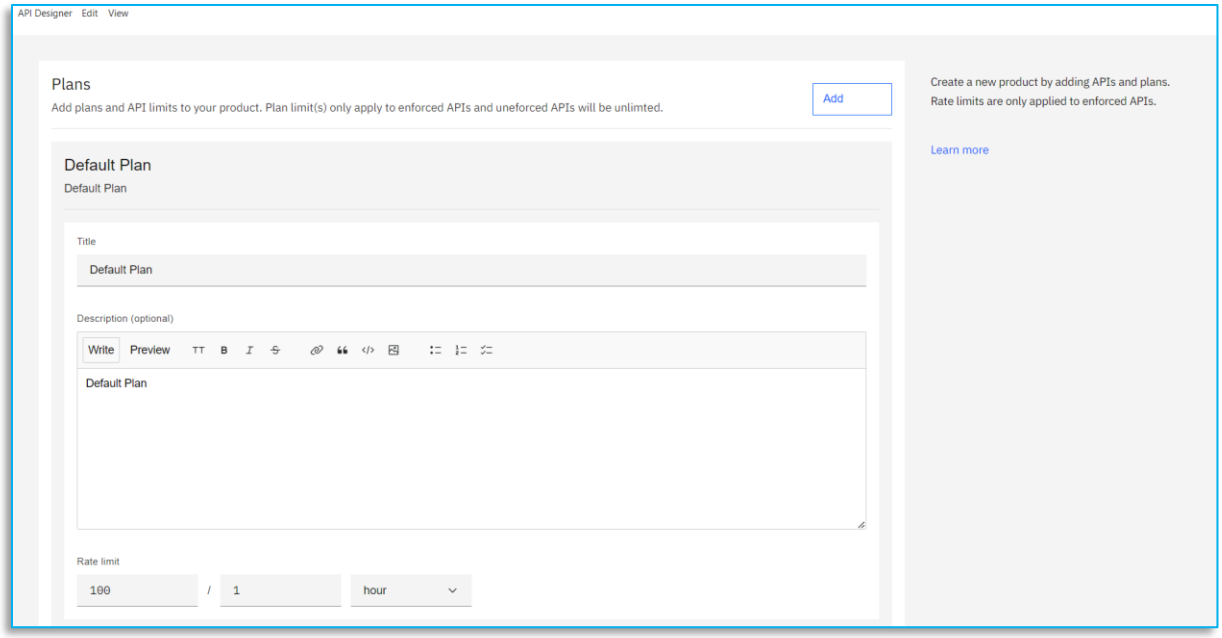


Es trien els APIs que es vagin a associar a aquest producte i se li dona a ***Next***. En aquest cas, es tria a "3292 Technical API".

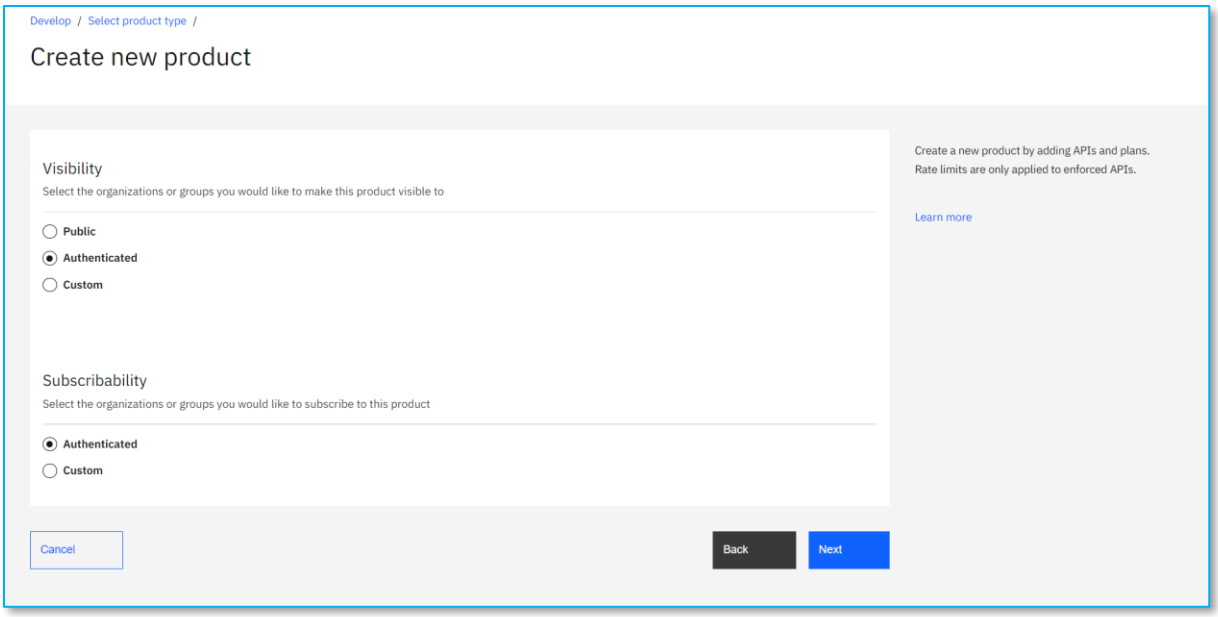


A continuació, es defineixen els plans del producte, juntament amb els límits de ***Rate i Burst***. Un exemple del pla podria ser el següent:

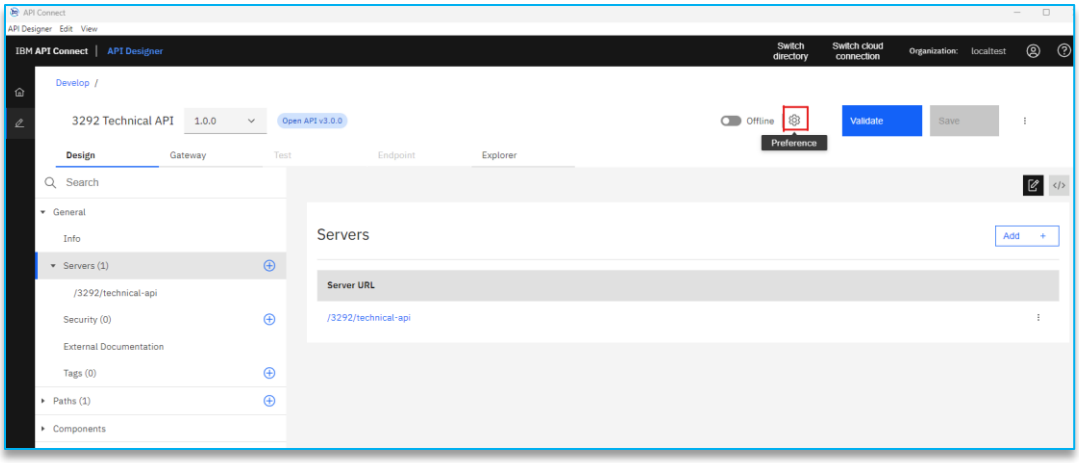
***Title**: Default Plan* ***Rate Limit**: 100 anomenades/hora* ***Burst Limit**: 10 anomenades/minut*



Es tria la ***visibilitat*** del producte i el seu ***Subscribability*** per a procedir. Es continua a través del botó ***Next*** i, en la finestra ***Summary*** següent, es poden repassar els passos que s'han realitzat abans de prémer en el botó ***Done*** per a finalitzar aquesta primera configuració.

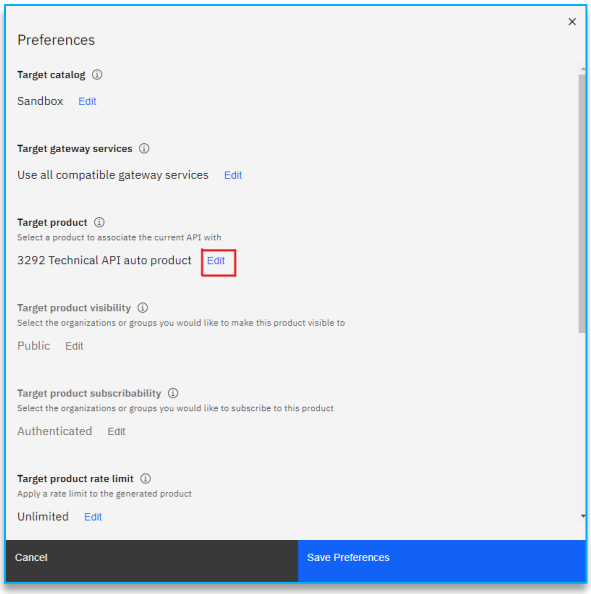


Previ a l'activació del API, primer es configura el producte al qual s'associarà el API. Per a això, es prem sobre la roda que es troba al costat del botó ***Validate***.

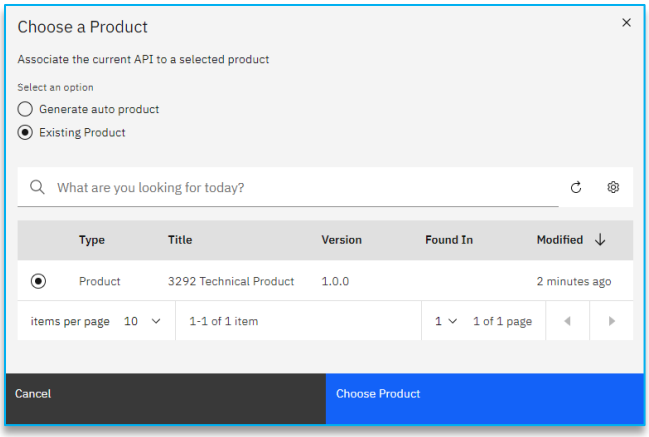


Prement sobre el botó ***Edit*** en la secció de ***Target Product*** , es tria, dins de la llista de ***Existing Product***, el producte que s'ha creat anteriorment.

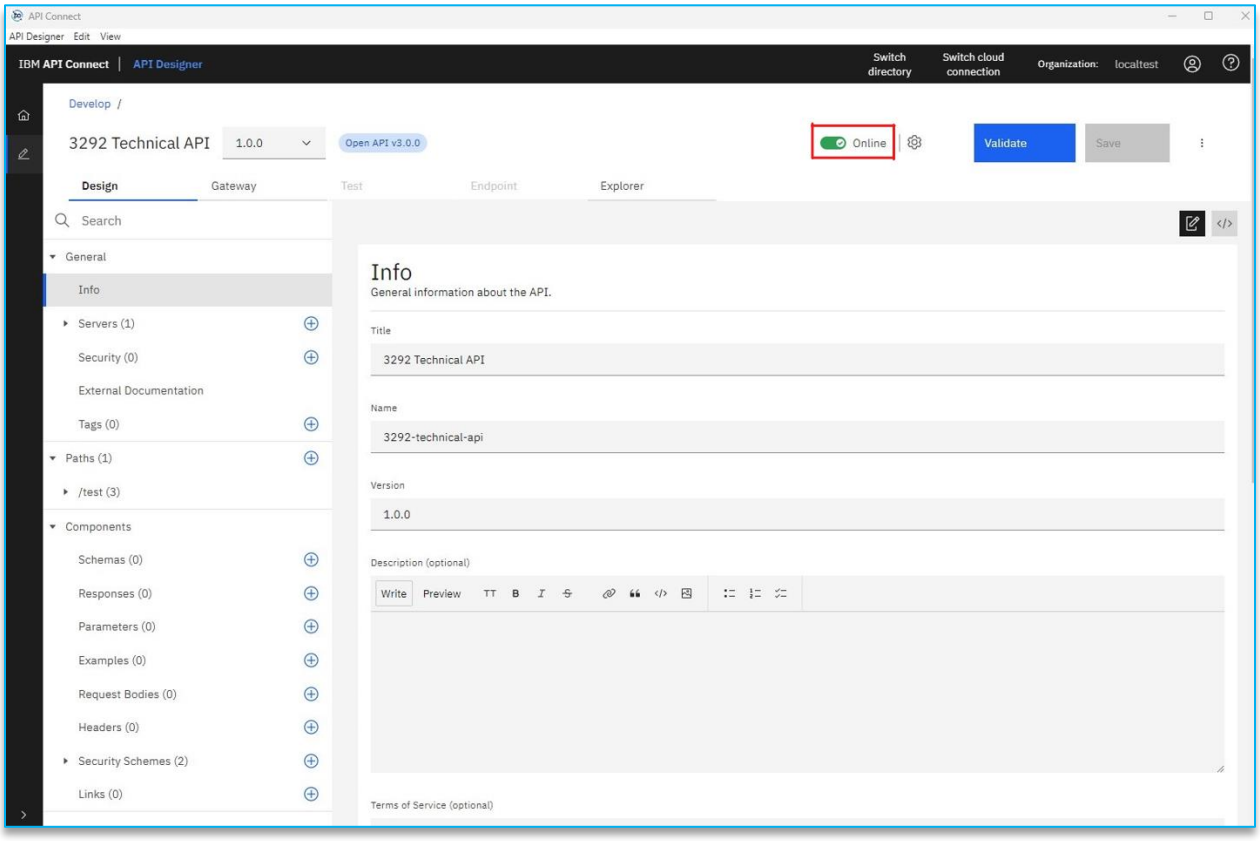
Finalment, es prem sobre el botó ***Choose Product*** i en ***Save*** per a guardar les configuracions.



Amb el producte ja configurat, es pot procedir amb l'activació del API, seguint els passos indicats en la següent diapositiva.



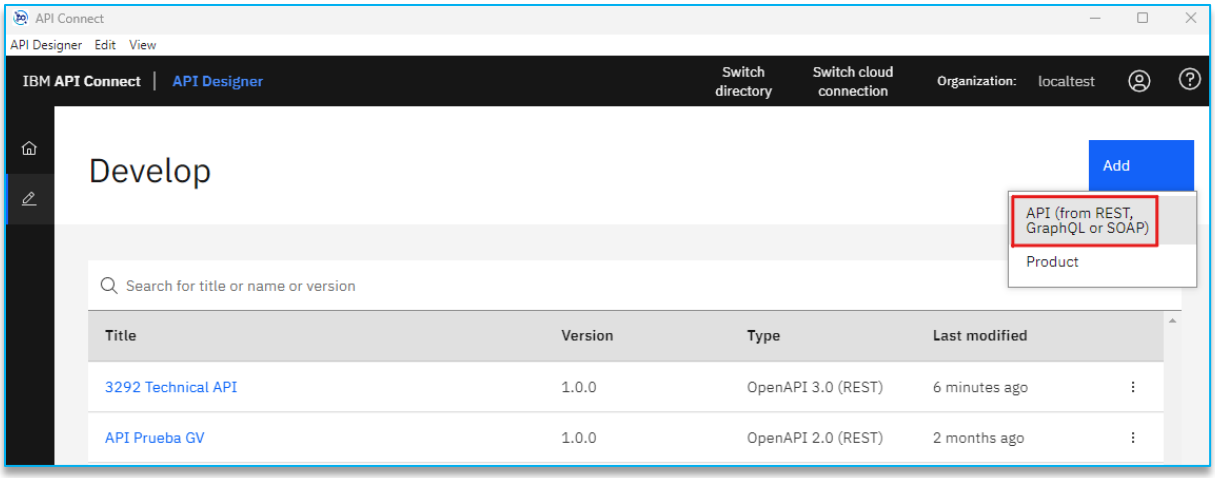
Finalment, ***s'activa la API per a procedir a realitzar les proves en l'entorn local***. Per a això, s'accedeix a la API i es llisca el ***botó Offline a En línia.***



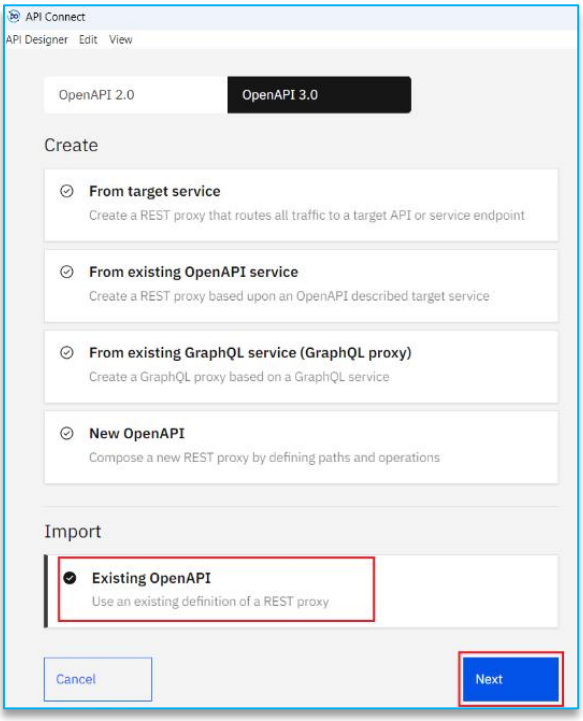
Amb això, el API ja estaria llest per a procedir amb les proves en l'entorn local. Per a això, es poden usar eines de testing com Postman.

Per a treballar en un nou API, recomanem la utilització de la plantilla de API base, la qual es pot importar en l'eina toolkit en local per a treballar en la implementació del API. Per a això, s'han de seguir els següents passos.

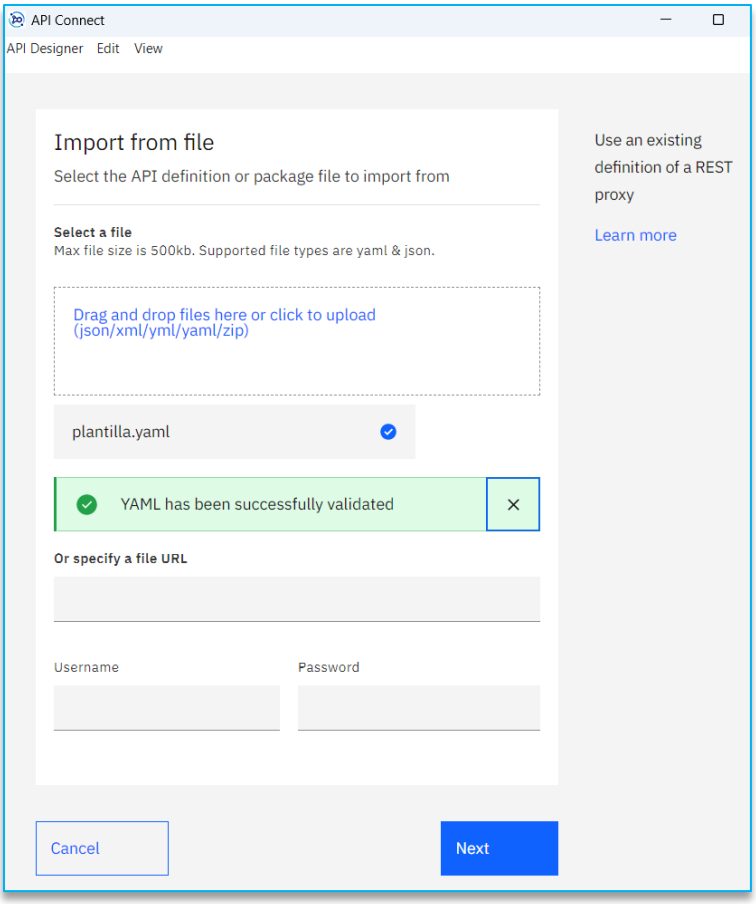
Des del Toolkit, en la vista de Disseny, es prem en el botó ***Add*** i se selecciona API:



En la següent vista, se selecciona ***OpenAPI 3.0*** (també es pot usar OpenAPI 2.0 si s'importa la plantilla del 2.0) i es marca l'opció de Existing OpenAPI, prement ***Next*** a continuació.



En la següent pantalla, es carrega el YAML de la plantilla i premem ***Next.***



4.3. Proves a través de LTE

Els projectes pueden configurar el ***Local Testing Environment*** proporcionat per IBM (requereix de ***Docker Desktop***), on es realitzen les proves unitàries dels APIs i Productes desenvolupats en l'entorn local, de cara a depurar i solucionar qualsevol error previ al desplegament en un entorn productiu del CTTI.

LTE permet provar les API a la màquina local, sense necessitat de connectar-vos a un servidor de gestió d'API Connect.

API Connect ofereix els mètodes següents per provar una API a la màquina local:

- Crida de l'API des de l'aplicació API Designer UI, que s'executa en mode en línia tal com es descriu a Testing an API.
- Cridar de l'API a l'entorn de prova local mitjançant un cURL.

Enumerem els punts principals que es traten a la documentació:

1. **Instal·lació de l'entorn local**
   - Instal·lació del LTE des de la màquina local.
   - Càrrega de les imatges a l'entorn local a un registre privat de Docker.
2. **Inici de l'entorn local**
   - Passes per iniciar les imatges de Docker.
3. **Preparació d'una API per fer proves a l'entorn local**
   - Preparació d'una API per provar-la a l'entorn local.

4. Prova d'una API a l'entorn local.

- Provar una API a l'entorn local, fent una crida a l'API REST.

S' adjunta a continuació una guia que mostra com realitzar els passos anteriors.

Guia%20Instal·lació%
20i%20Configuració%

Enllaços:

- Documentació d'IBM

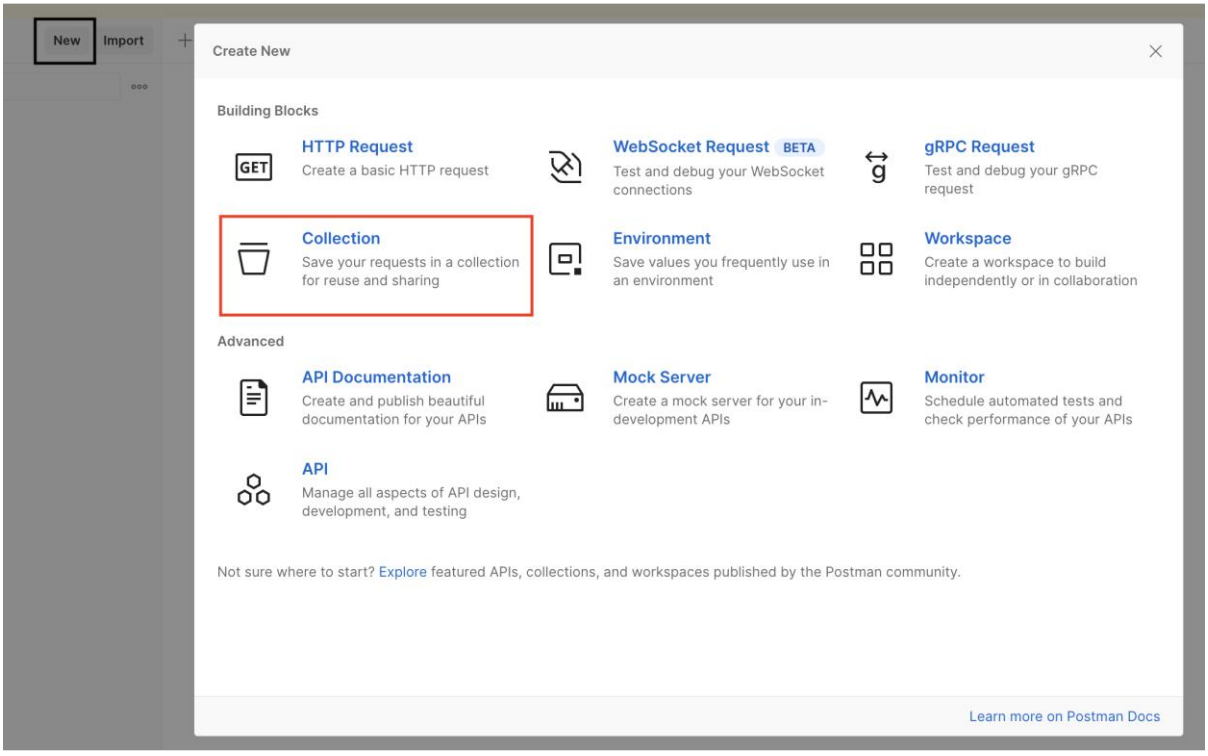
4.3.1. Testing Postman

En aquest punt, s'explicarà com crear els elements necessaris per generar ***test unitaris*** per a una API utilitzant Postman.

Col·lecció Postman:

S'haurà de crear una ***col·lecció*** de Postman, per ***Producte***. Per tant, obrim el Postman i li donem a ***New*** i després a ***Collection***.

Donarem nom a la col·lecció amb el ***nom del producte***.



Peticions de la col·lecció:

La col·lecció haurà de contenir almenys una ***petició per operació*** de cadascuna de les APIs contingudes en el producte.

En afegir la petició, seleccionem el tipus d'operació de l'API (GET, POST,...), posem la url i les dades que necessitem per poder provar l'API.

En finalitzar les nostres proves, recomanem exportar la col·lecció de preproducció per pujar-la al repositori creant les carpetes ***"test/postman"*** en l'arrel del repositori perquè estigui disponible si s'han de repetir les proves a futur.

4.4. Entrega de codi

El lliurament de codi es pot realitzar de dues maneres, en funció de si es decideix anar per ***SIC 3.0*** i les pipelins de ***Jenkins***, o es fa ús del nou ***SIC+***, el qual fa ús de ***GitHub Enterprise Cloud***.

4.4.1. SIC 3.0

Los fitxers ***YAML*** desenvolupats en local es repositaran a la plataforma corporativa de custodia de codi font del SIC, (https://canigo.ctti.gencat.cat/plataformes/sic/serveis/sic30-serveis/scm/), tal com es fa amb tota la resta d'aplicacions.

Es crearà un projecte associat al codi de diàleg per cada producte que es desplegui.

Prèviament a l'entrega de codi, se recomienda fer les següents modificacions als fitxers per adaptar-los al model de desplegament automatitzat:

- Cal eliminar les versions dels noms dels fitxers dels Productes i de les APIs, inclòs les referències internes que es fan entre els fitxers.

L'objectiu és que les versions queden definides dins dels fitxers yaml, i no al seu nom.

4.4.2. SIC+

Los fitxers ***YAML*** desenvolupats en local es repositaran en la branca ***feature*** corresponent del repositori de ***GitHub*** adequat, el qual haurà d' haver estat creat prèviament. Més informació de com generar aquests repositoris i realitzar totes les configuracions prèvies en: https://canigo.ctti.gencat.cat/plataformes/ghec/gh-adopcio-model-ghec/
Es crearà un projecte associat al codi de diàleg per cada producte que es desplegui i es fa ús del model ***GitFlow*** de branques.
Prèviament a l'entrega de codi, se recomienda fer les següents modificacions als fitxers per adaptar-los al model de desplegament automatitzat:

- Cal eliminar les versions dels noms dels fitxers dels Productes i de les APIs, inclòs les referències internes que es fan entre els fitxers.

L'objectiu és que les versions queden definides dins dels fitxers yaml, i no al seu nom.

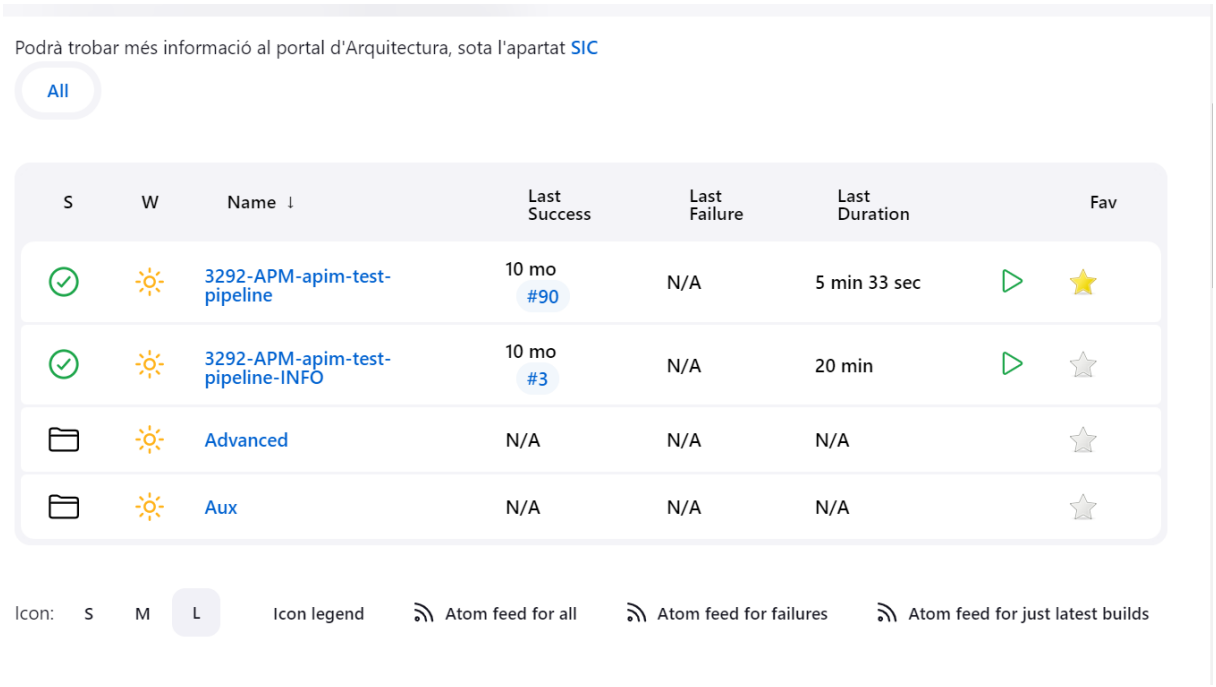
4.5. Desplegament

El desplegament es pot realitzar de dues maneres, en funció de si es decideix anar per ***SIC 3.0*** i les pipelins de ***Jenkins***, o es fa ús del nou ***SIC+***, el qual fa ús de ***GitHub Enterprise Cloud***.

4.5.1. SIC 3.0

Durant la fase de projecte, es crearà per cada producte un conjunt de pipelines al SIC que permeten gestionar el cicle de vida de les APIs.

Addicionalment a les pipelines referents al cicle de vida, es proporcionarà una pipeline que mostra informació diversa respecte del Productes: versions d'APIs incloses, nombre de subscripcions, etc.



Dins la carpeta ***Advanced***, es poden trobar més pipelines.

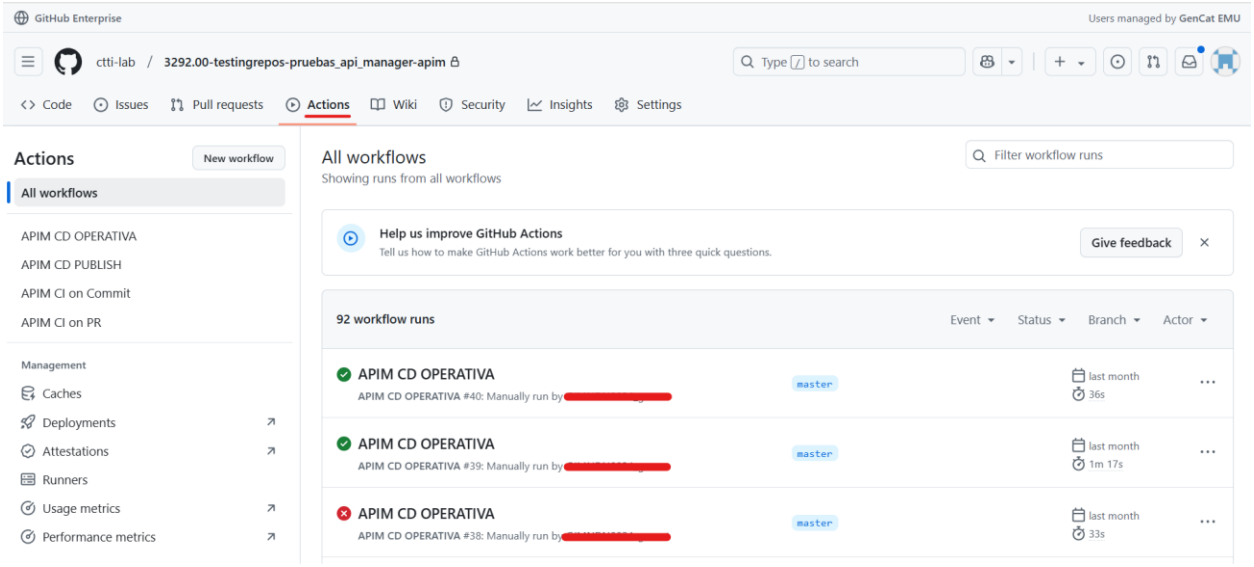
Es recomana accedir a la següent URL de Canigó, https://canigo.ctti.gencat.cat/plataformes/apim/desplegament/, on es troba tant el resum de totes les configuracions a realitzar per poder desplegar APIs i productes com la ***Guia de desplegament***, que conté tota la informació detallada. Aquesta guia també es pot trobar a la mateixa secció de documentació a ***Canigó*** que aquest document.

4.5.2. SIC+

En el cas de fer ús del ***SIC+***, els desplegaments es realitzaran mitjançant ***workflows de GitHub***.

Per a cadascun dels repositoris creats, es creen un conjunt de workflows que seran els que executin les tasques de CI/CD d'infraestructura i d'aplicació.

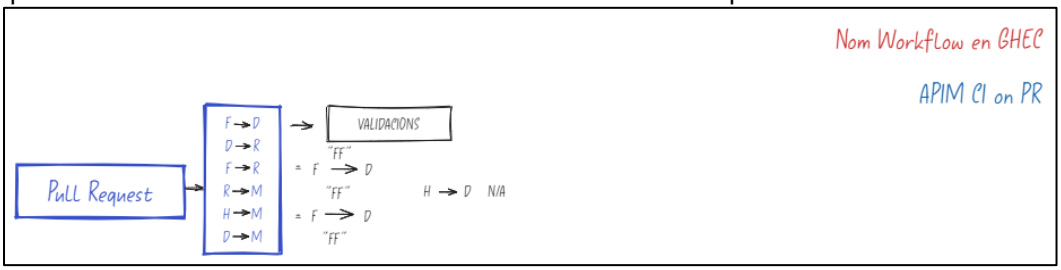
L'accés a aquests workflows es realitzarà a través de l'opció ***"Actions"*** de cada repositori a GHEC.



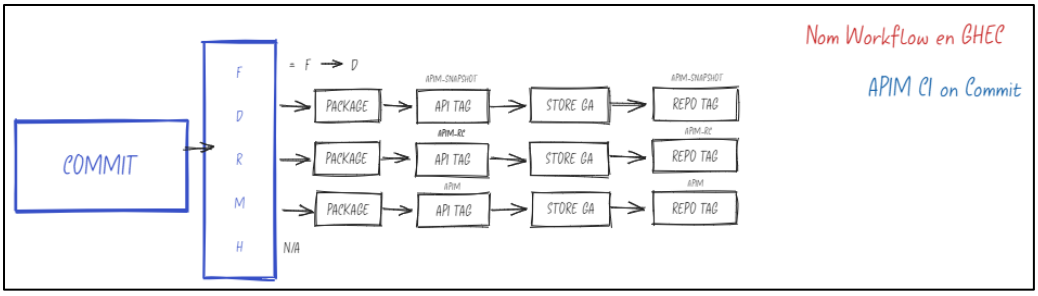
L'execució dels workflows dependran de la seva tipologia i del model definit, essent:

- ***Workflows de CI:*** Executats automàticament en la sol·licitud d'un Pull Request o en l'execució d'un Merge de dita Pull Request. S'ha diferenciat en el workflow entre ***canvis en temps de Pull Request (PR)***, que equivaldria al procés pel qual un usuari crea la PR, i encara no és validada per un moderador o usuari del repositori i ***canvis en temps de Commit***, que equivaldria al procés després d'haver-se acceptat la PR, i integrar ambdues branques involucrades.
    - ***APIM CI on PR.*** Aquest workflow, depenent d'aquestes branques que es vulguin "mergear", executarà diferents steps amb diferents jobs. Si es crea una PR d'una branca ***feature*** a la branca ***develop***, en temps d'execució es llançarà el workflow de CI que executarà els steps de validació de codi mitjançant el comandament validate de l'eina apic i aquest resultat es comenta en la PR.

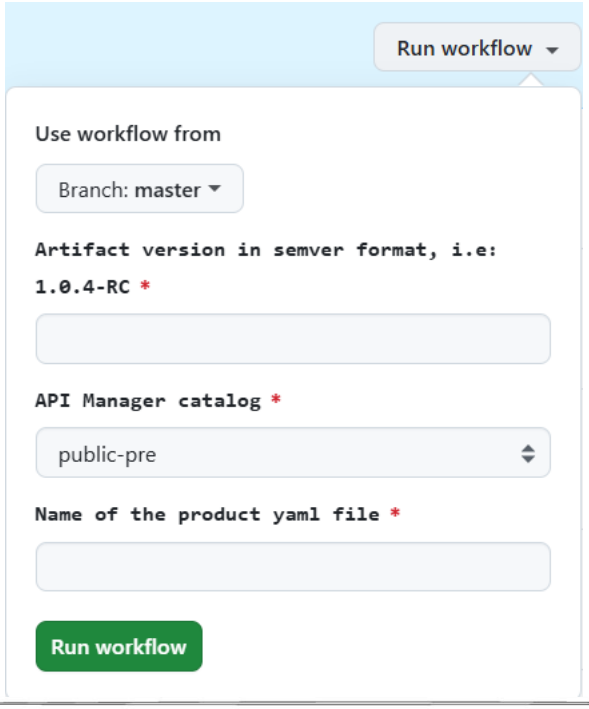
      En canvi, si la PR es fes entre les branques ***develop-release***, ***release-master***, ***hotfix-master***, s'ometrien aquests steps i es realitzaria un fast-forward, ja que tots ells haurien estat executats i validats prèviament, donat que teòricament el codi no rep més canvis des que entra en la branca develop en endavant.

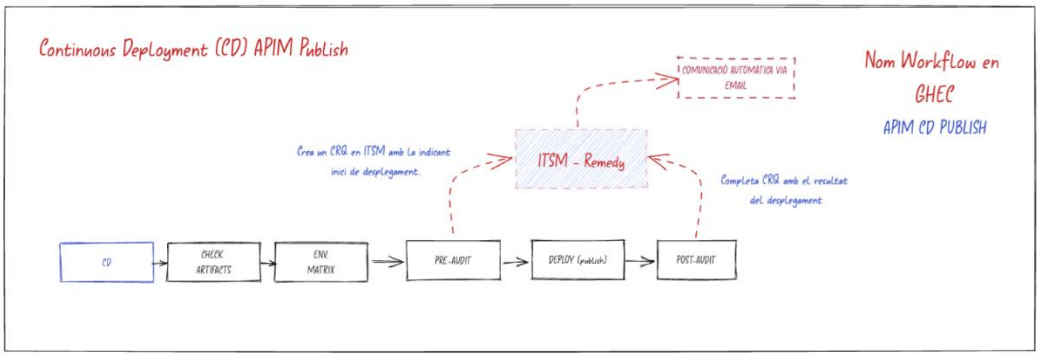


    - ***APIM CI on Commit.*** Si estem en temps de commit, i partint de la base que el paquet no ha de ser mutable entre els diferents entorns, es comprova el tag generat en temps de PR, es publica l'artefacte a ***Github Artifacts*** i es torna a fer el tag del repositori.



- ***Workflows de CD:*** Executats sota demanda a través de la interfície web de GHEC.
    - ***APIM CD PUBLISH.*** El workflow farà la publicació d'una nova versió d'un producte i APIs associades. El sistema permet redesplegar versions als catàlegs preproductius sempre que no hagin arribat a producció.



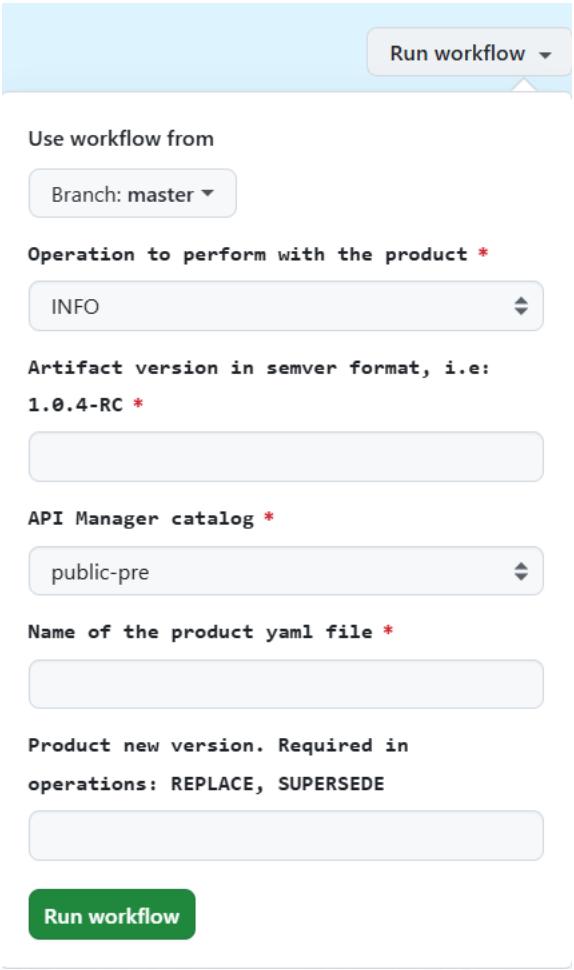


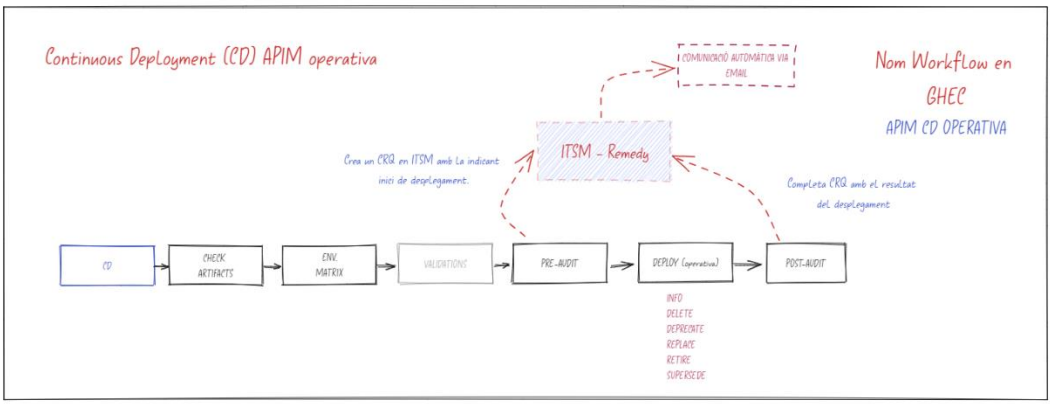
on:

- ***CHECK ARTIFACTS:*** Comprova l'existència i validesa de l'artefacte en el repositori.
- ***ENVIROMENT MATRIX:*** Valida si l'artefacte pot ser desplegat en l'entorn especificat.

- ***ITSM PRE AUDIT:*** Realitza una auditoria prèvia en ITSM creant una CRQ per a la preparació del desplegament (només en entorns diferents de dev).
- ***PUBLISH PRODUCT TO IBM API MANAGER (Deploy):*** Publica el producte a IBM API Manager segons els paràmetres configurats.
- ***ITSM POST AUDIT:*** Completa l'auditoria en ITSM després del desplegament, registrant l'estat final i completant la CRQ.

- ***APIM CD OPERATIVA.*** El workflow realitzarà una de les següents operatives: ***INFO***, ***DELETE***, ***DEPRECATE***, ***RETIRE***, ***REPLACE***, ***SUPERSEDE***.





on:

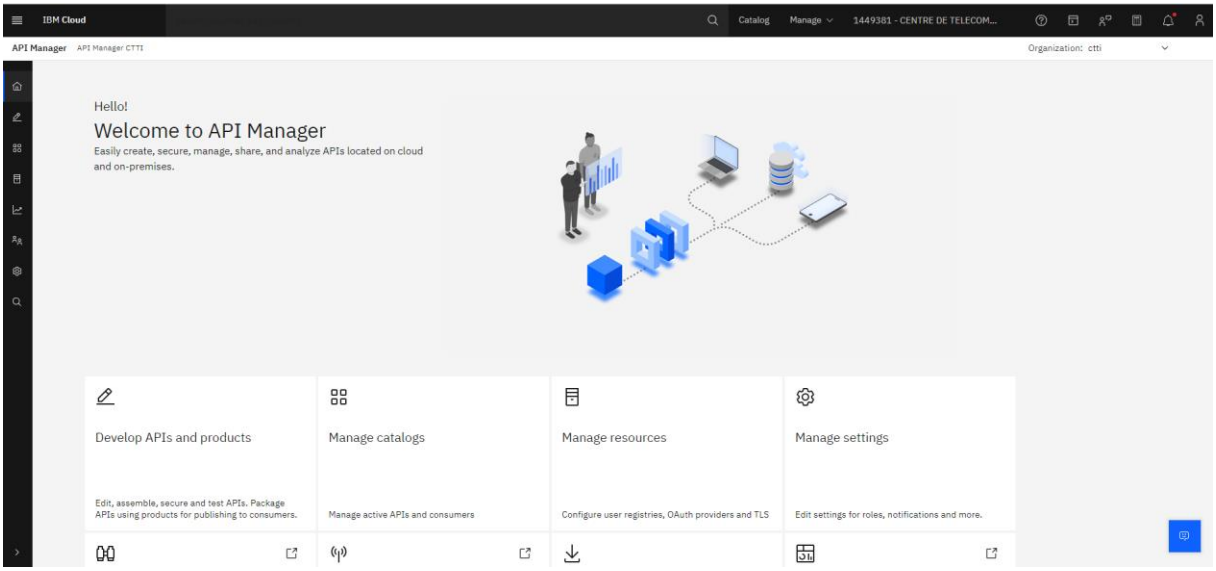
- ***CHECKOUT:*** Realitza la verificació de codi en el repositori.
- ***CHECK ARTIFACTS:*** Verifica l'existència i validesa de l'artefacte al repositori.
- ***ENVIROMENT MATRIX:*** Valida si l'artefacte pot ser desplegat en l'entorn especificat.
- ***VALIDATIONS:*** Step preparat per a inserir les futures validacions a realitzar prèvies a l'execució de les operatives.
- ***ITSM PRE AUDIT:*** Realitza una auditoria prèvia a ITSM, creant una CRQ per al desplegament (només en entorns diferents de dev).
- ***PRODUCT OPERATION INTO IBM API MANAGER (Deploy):*** Executa operacions a IBM API Manager segons l'operació especificada.
- ***ITSM POST AUDIT:*** Completa l'auditoria a ITSM després del desplegament, registrant l'estat final i completant la CRQ.

Es recomana accedir a la següent URL de Canigó, https://canigo.ctti.gencat.cat/plataformes/apim/desplegament/, on es troba tant el resum de totes les configuracions a realitzar per poder desplegar APIs i productes com la ***Guia de desplegament***, que conté tota la informació detallada. Aquesta guia també es pot trobar a la mateixa secció de documentació a ***Canigó*** que aquest document. Al seu torn, es pot trobar tota la informació sobre el model de ***SIC+*** a la url https://canigo.ctti.gencat.cat/plataformes/ghec .

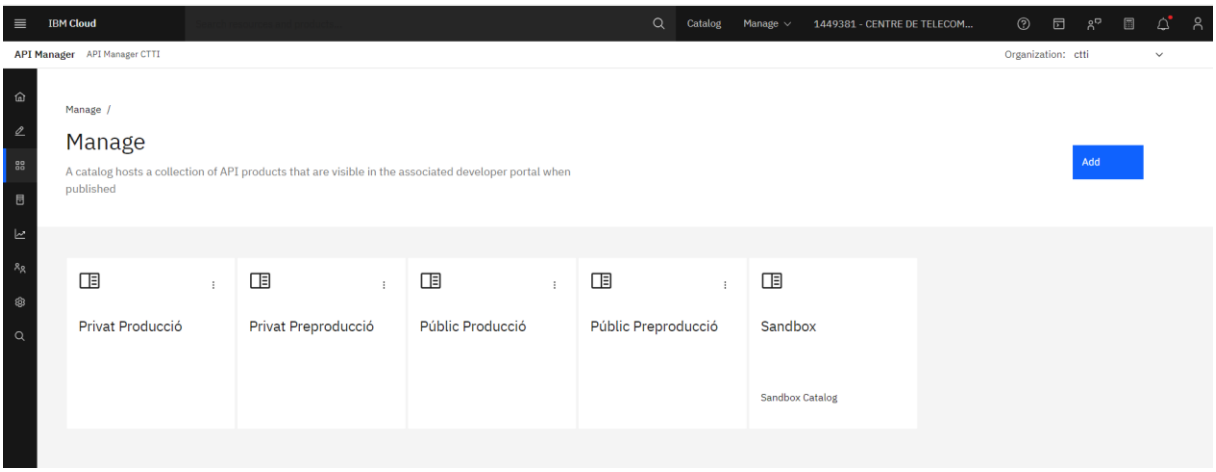
5. GESTIÓ DE SUBSCRIPCIONS

En el cas que un usuari sigui el responsable d' aprovar o denegar les subscripcions als seus productes i APIs, haurà de realitzar els següents passos per poder aprovar o denegar-ne una.

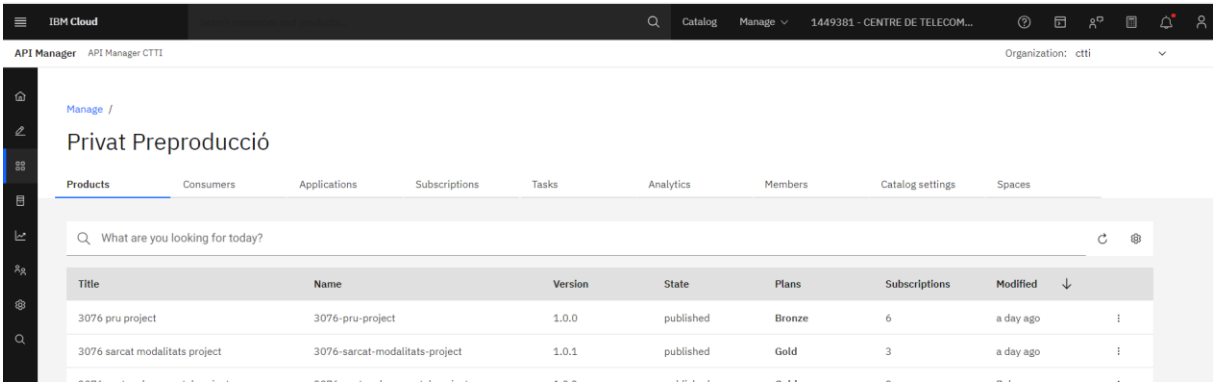
Primer, ha d' arribar a la pantalla d' inici de la consola de gestió de l' organització de CTTI (els passos per arribar-hi es troben anteriorment en aquest document, en l' apartat 1.4).



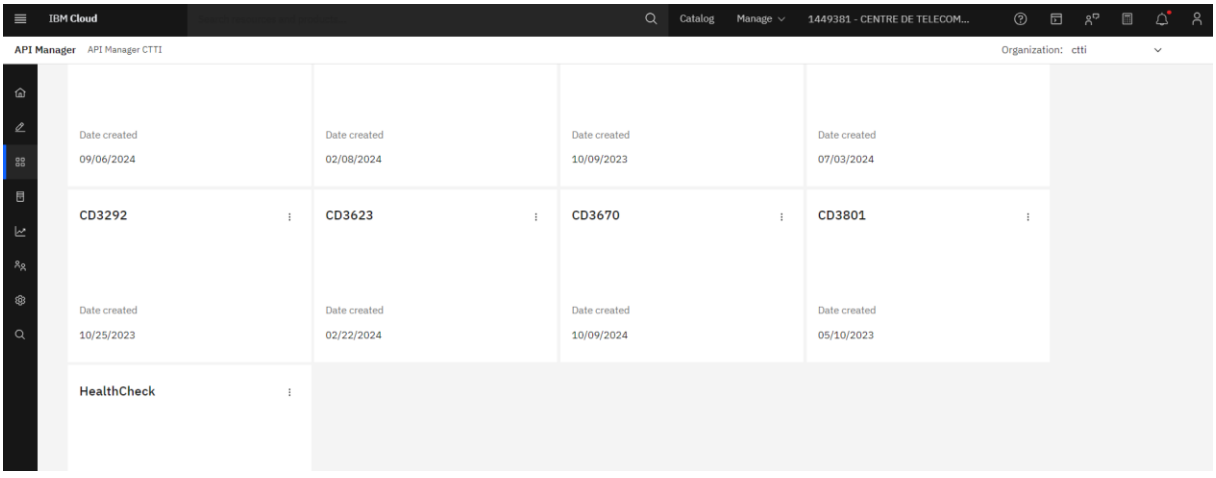
Un cop s'arriba a aquesta pantalla, polsar a ***Manage catalogs*** o a la tercera icona de la barra de l'esquerra.



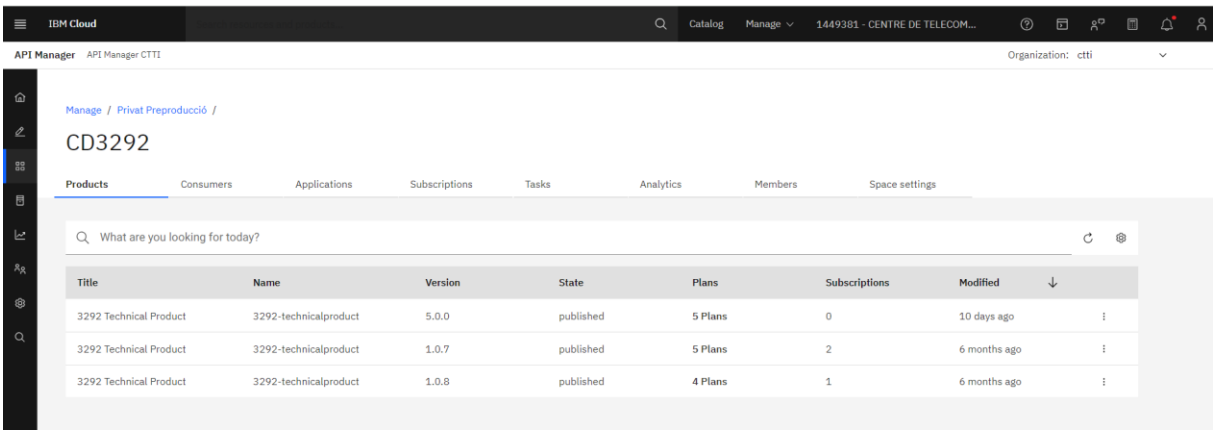
Després, s' ha de polsar en el catàleg corresponent en el qual està desplegat el producte per al qual s' aprovarà aquesta subscripció.



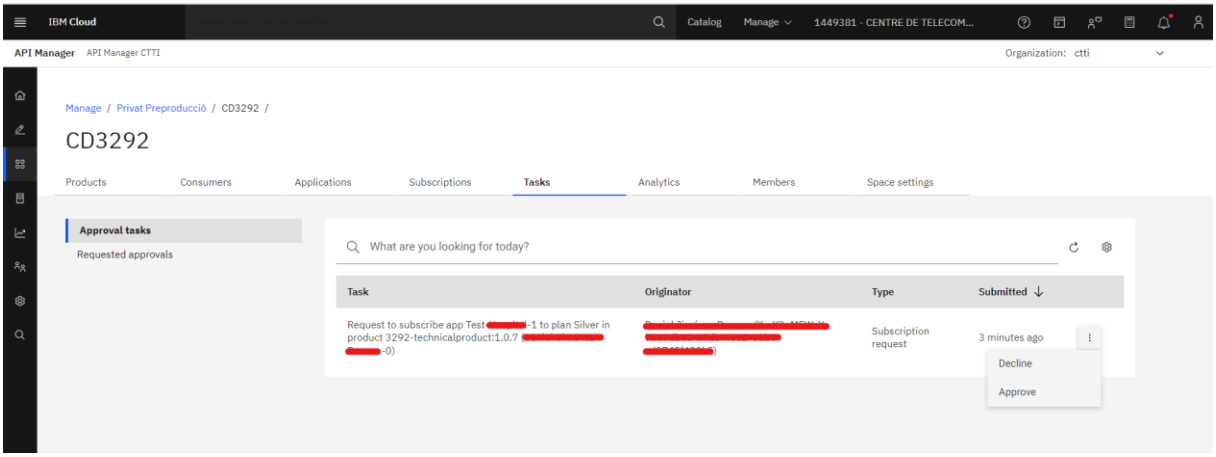
A continuació, s'ha de polsar a ***Spaces***.



Després, polsar a l'espai on està desplegat el producte per al qual s'aprovarà la subscripció.



Finalment, s'ha de polsar a ***Tasks***. Aquí es trobaran les peticions de subscripció. Polsant en els tres punts a la dreta de cada petició, es podran ***aprovar*** o ***declinar***.



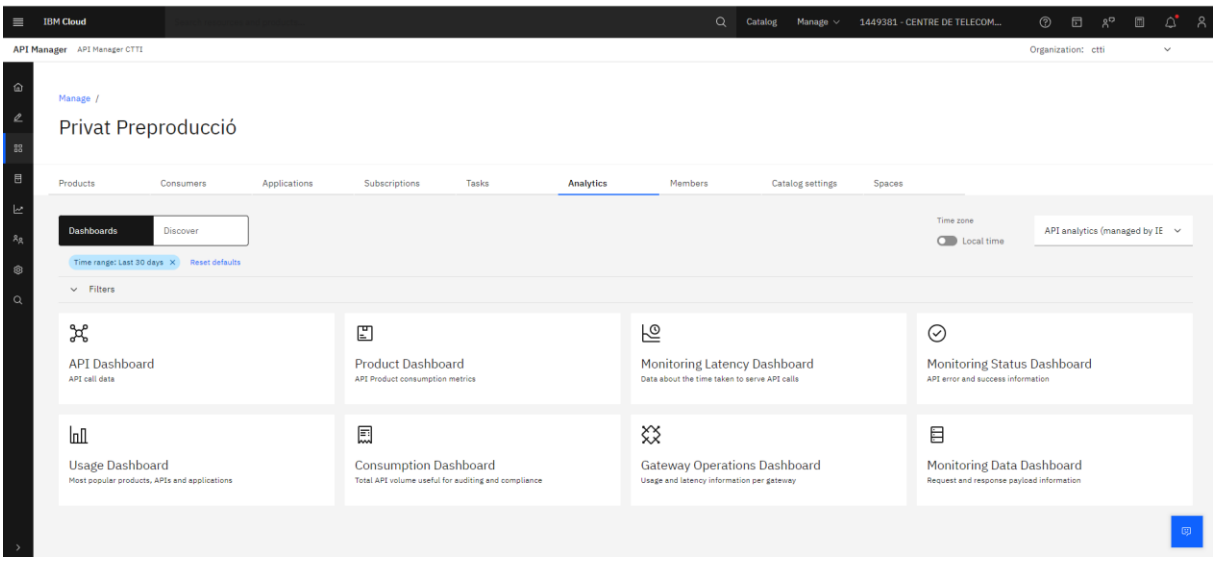
6. ANÁLISI DE CONSUMS

6.1. Descripció

Per veure les analítiques, cal entrar al portal de gestió de l'API Manager. Cal accedir a ***API Management***, ***Services*** i ***ctti***. Des del menú de la esquerra es selecciona ***Manage***, es selecciona un catàleg de treball i un espai, i premem l'opció ***Analytics***. Els passos més detallats de com arribar fins a un espai des que un es lloa al Cloud d'IBM es troben en altres punts d'aquest mateix document.

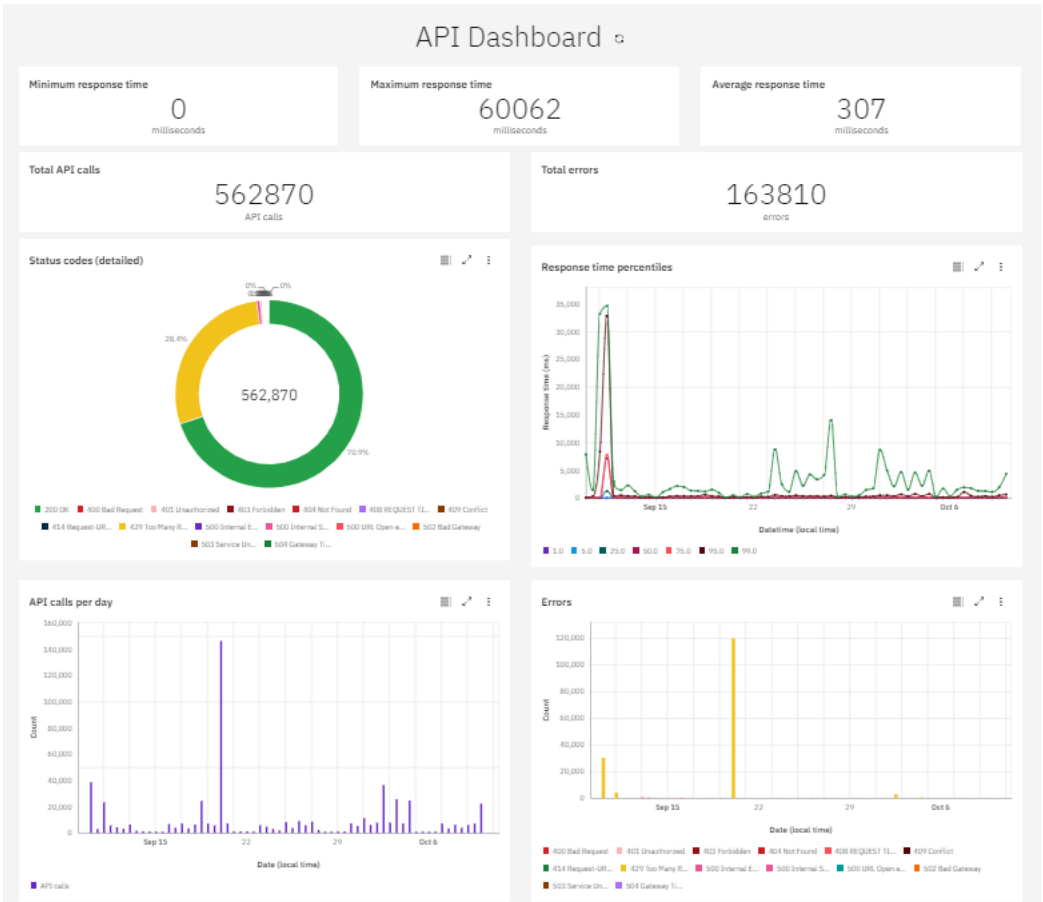
6.2. Analítiques

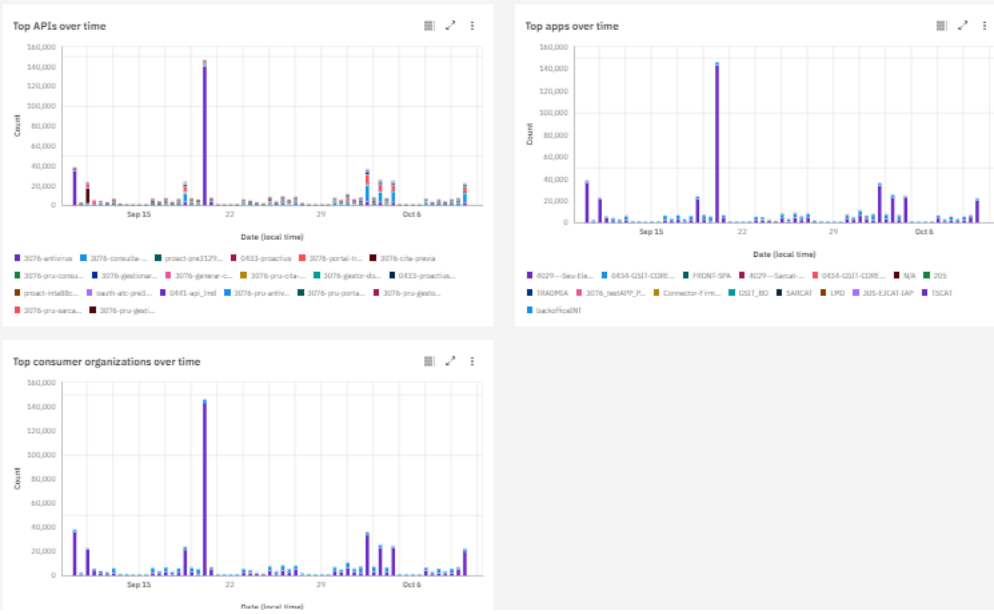
Aquest opció permet visualitzar anàlitiques a més d' exportar-les.



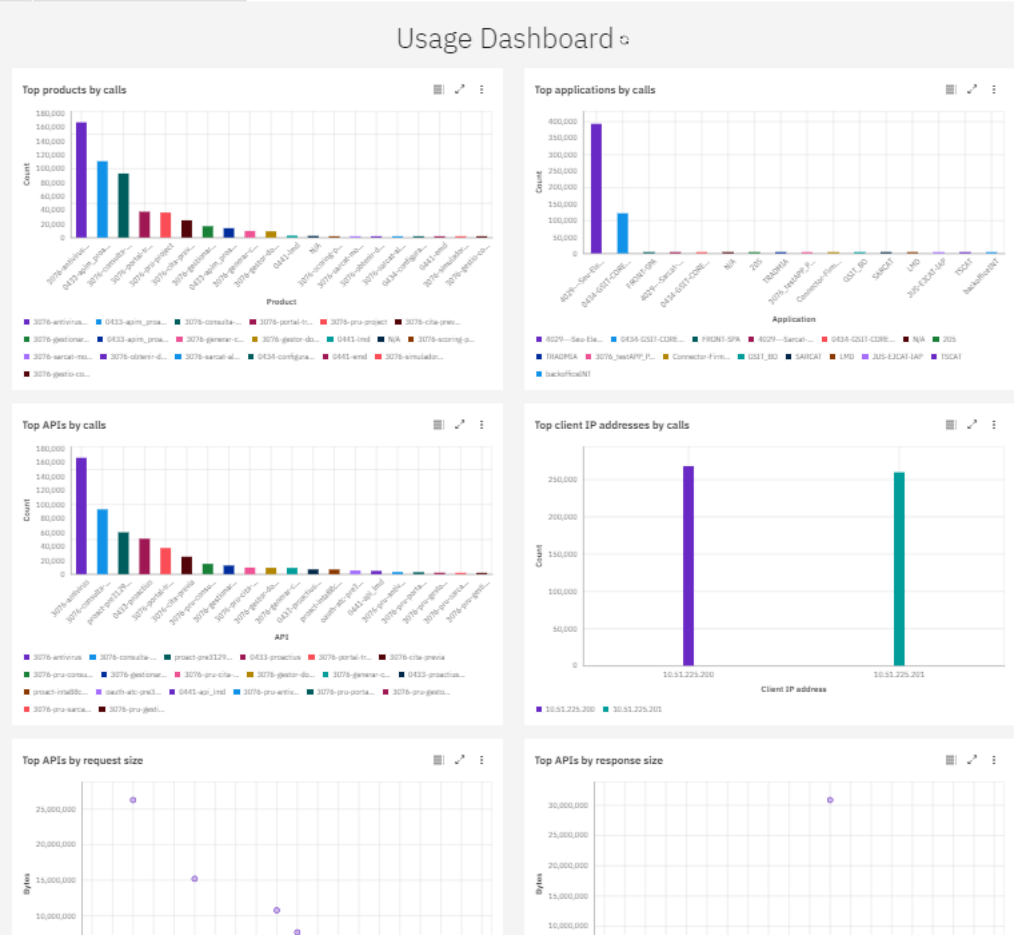
Es poden trobar diverses visualitzacions, mitjançant panells gràfics o taules, en ambdues visualitzacions es poden afegir filtres per cercar informació.

Exemple de visualització gràfica amb indicadores per defecte (Overview of API calls) amb dashboard ***API Dashboard***:





Indicadors dels accessos a productes i APIS amb el dashboard ***Usage dashboard***:



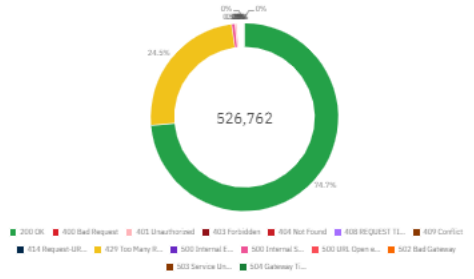
Indicadors tècnics (success rates, status codes, errors data usage, response times, data usage, etc) amb el dashboard ***Monitoring status***, ***Monitoring latency, etc***.

Monitoring Status Dashboard

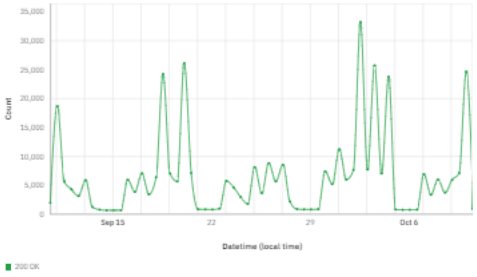
Total API calls

526762
API calls

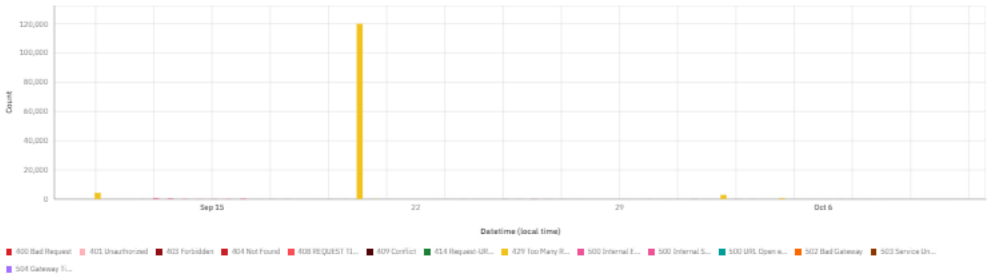
Status codes (detailed)



Success rate



Errors

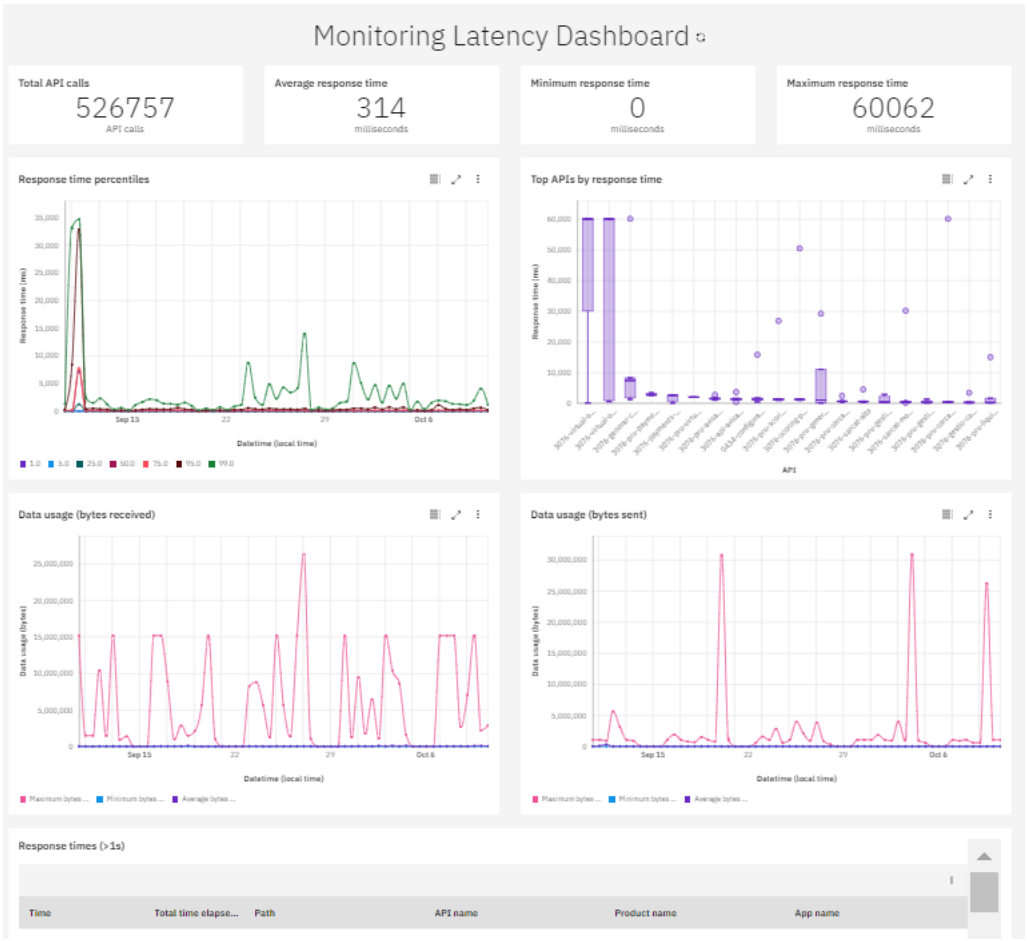


Errors data

Datetime	Status code	Path	API name	Produ

Success data

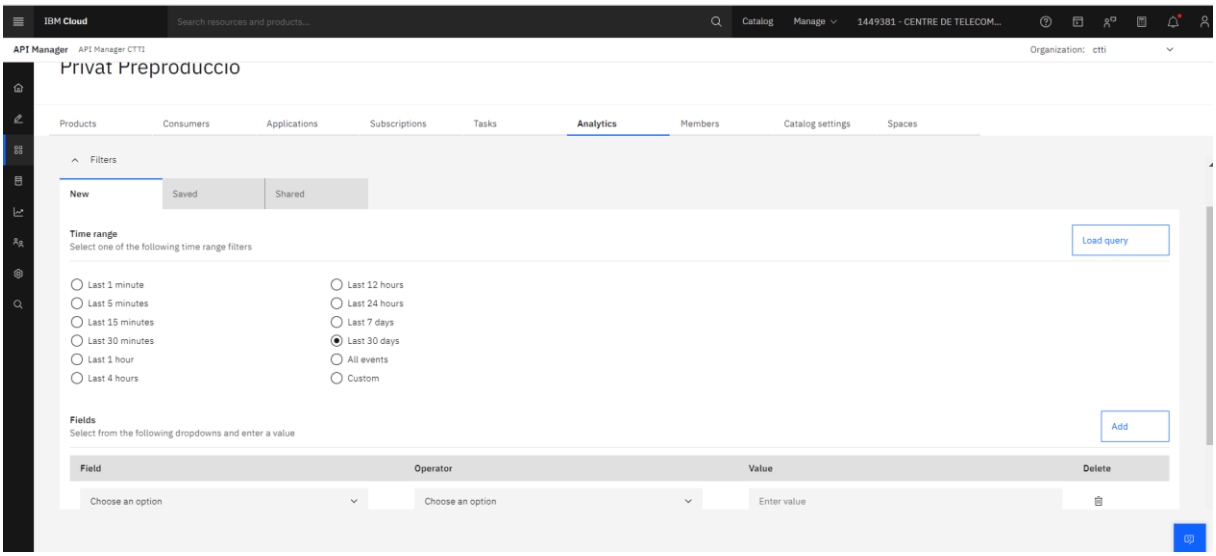
Datetime	Status code	Path	API name	Produ



6.2.1. Filtres

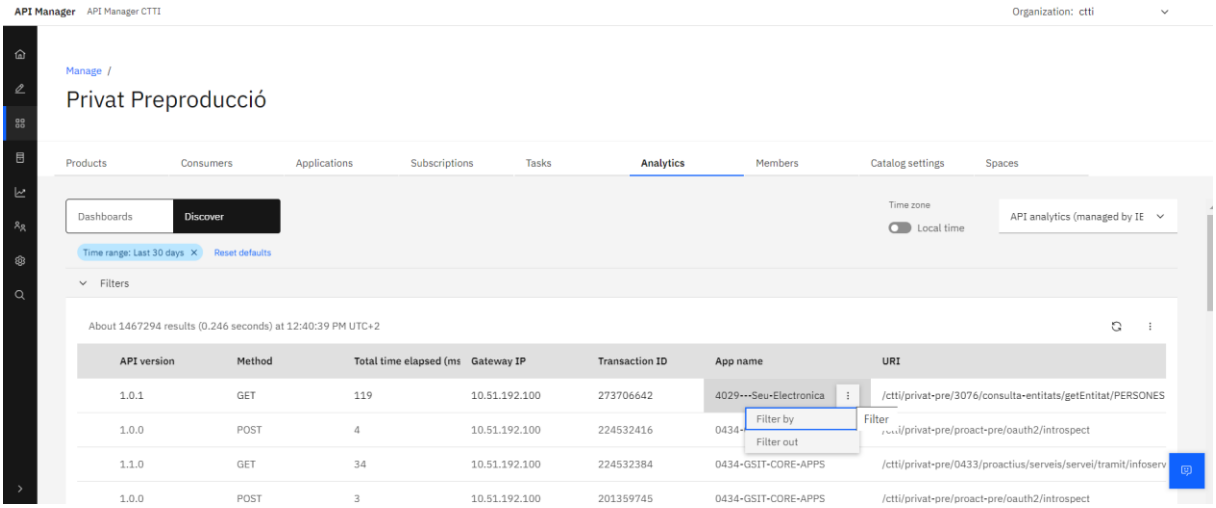
La pestaña ***Discover*** filtra tota la informació generada per totes les peticions que s'han processat als gateways, informació que esta indexada a l'elàstic per cada petició.

Aplicant filtres podem fer la recerca acotada per temps.



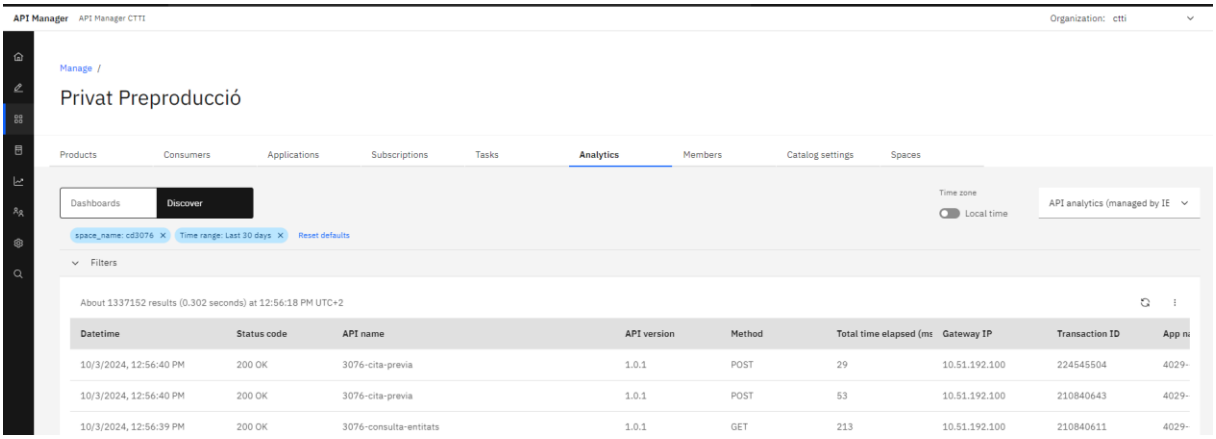
Per fer cerques més concretes, amb l'opció ***Add*** podem afegir filtres manualment.

Una altra manera que proporciona l'eina d'aplicar filtres és anant a la pestanya ***Discover*** i polsant en els tres punts que apareixen al costat de cada valor de la taula, en qualsevol fila o columna.

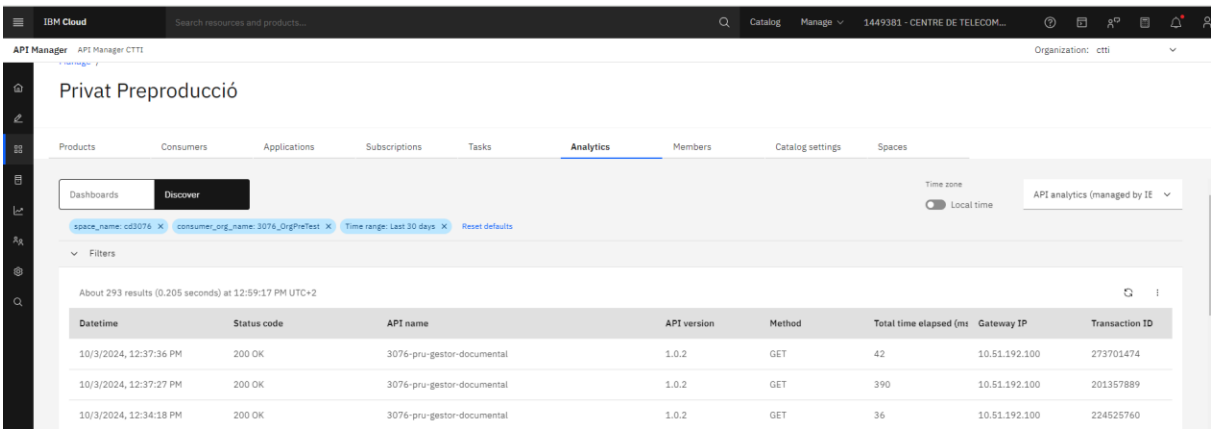


Si apliquem un filtre a qualsevol columna de la dreta apareixen concatenats els filtres del nivell superior.

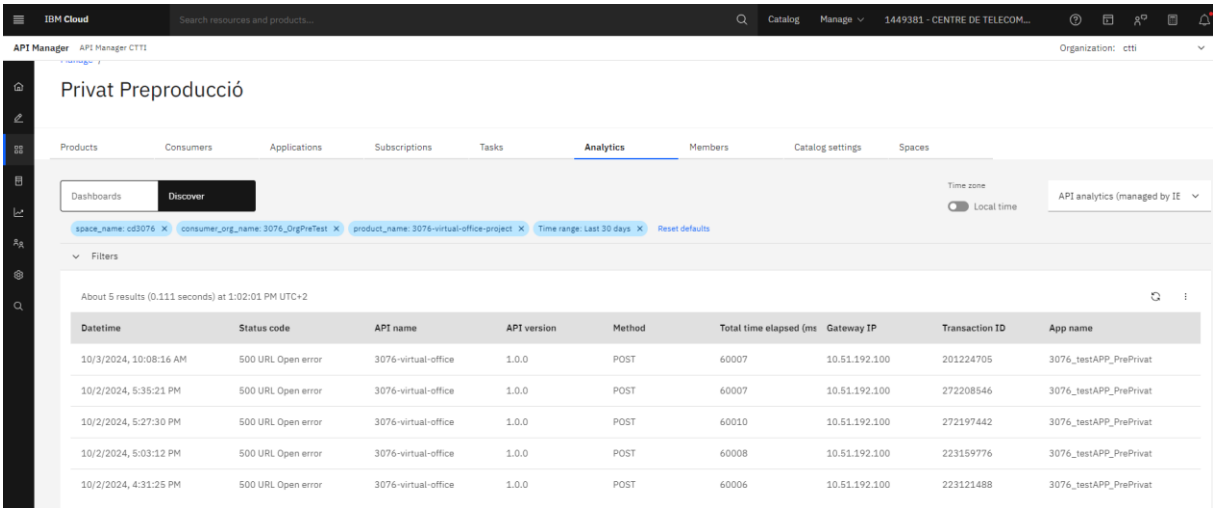
Filtre a nivell d'espai:



Filtre a nivell de consumidor:



Filtre a nivell de producte:



Filtre a nivell d'API: