**Introduction:**

Blue API includes a catalog of APIs available to all of IBM, a continuously available platform to build and manage APIs

Roles:

1)Application Developers:

### Application Developers are the web developers who can easily log into the App Developer Hub, search for APIs and start building applications from exposed APIs.

### 2)API Providers:

### API Providers can quickly configure and test APIs from start to finish. Once the APIs are published, you can easily onboard developers, manage support and provide an easy on-ramp to the API Economy. You will also be able to get real business insights at a glance that show how customers are utilizing your APIs.

**There are three roles that are included in the API Provider function:**

**API Developers, also called API Authors, utilize the API Manager to define, unit test and debug managed APIs.**

Rapidly assemble APIs via configuration, not coding.

Easily version, apply security and publish APIs.

Leverage analytic capabilities to improve your APIs.

**API Product Manager (PM) is responsible for approving or rejecting App Developer subscription requests to API plans. In addition, the PM may run reports of API usage.**

Allows for maximized revenue for your APIs by making it easy for adopters to find, consume and analyze new solutions.

Allows you to change pricing as business needs change.

Provides detailed analytics on usage by your adopters.

**Deployment Manager (DM) is the technical lead from the business unit, responsible for the Provider Organization and all APIs developed there.**

On-board plans and manage support of APIs quickly and easily.

Make changes quickly and easily.

Improve lifecycle management by integrating APIs through a DevOps approach

### API Manager:

API Manager is the access point for API Developers to design, control, secure, publish, manage, analyze and scale your API in minutes with an easy-to-use configuration platform that does not require coding.

The API Manager user interface tool provides a solution to manage APIs for IBM internal business applications. This tool provides the capabilities that allow you to externalize and manage your REST APIs.

With API Manager, API Developers can:

* Define APIs
* Test and deploy APIs
* Configure APIs (throttle, etc.)

API Plans contain a collection of APIs only made available to consumers through the process of promotion. The process involves 3 different environments: Development, Test, and Run.

* Development – Open environment for API developers to create and test APIs.
* Test – Environment used for functional, performance or quality assurance testing.
* Run – Environment to expose APIs in API Catalog for operational use.

Promotion from Dev to Test

API developers have full access and control within Development to construct and change any aspect of the API and its Plans before promotion. Once developers are ready for their APIs to be promoted to Test, they must contact their Deployment Manager (DM) for assistance. DM should review the API and its Plans for the business need of its organization as configured by API developers as well as compliance with [API management standards](https://w3.apihub.ibm.com/blueapi/api_standards.html). DM can initiate promotion to Test from the Dashboard, API Lifecycle or scripts (see [Get Started](https://w3.apihub.ibm.com/blueapi/get_started.html) page for more details).

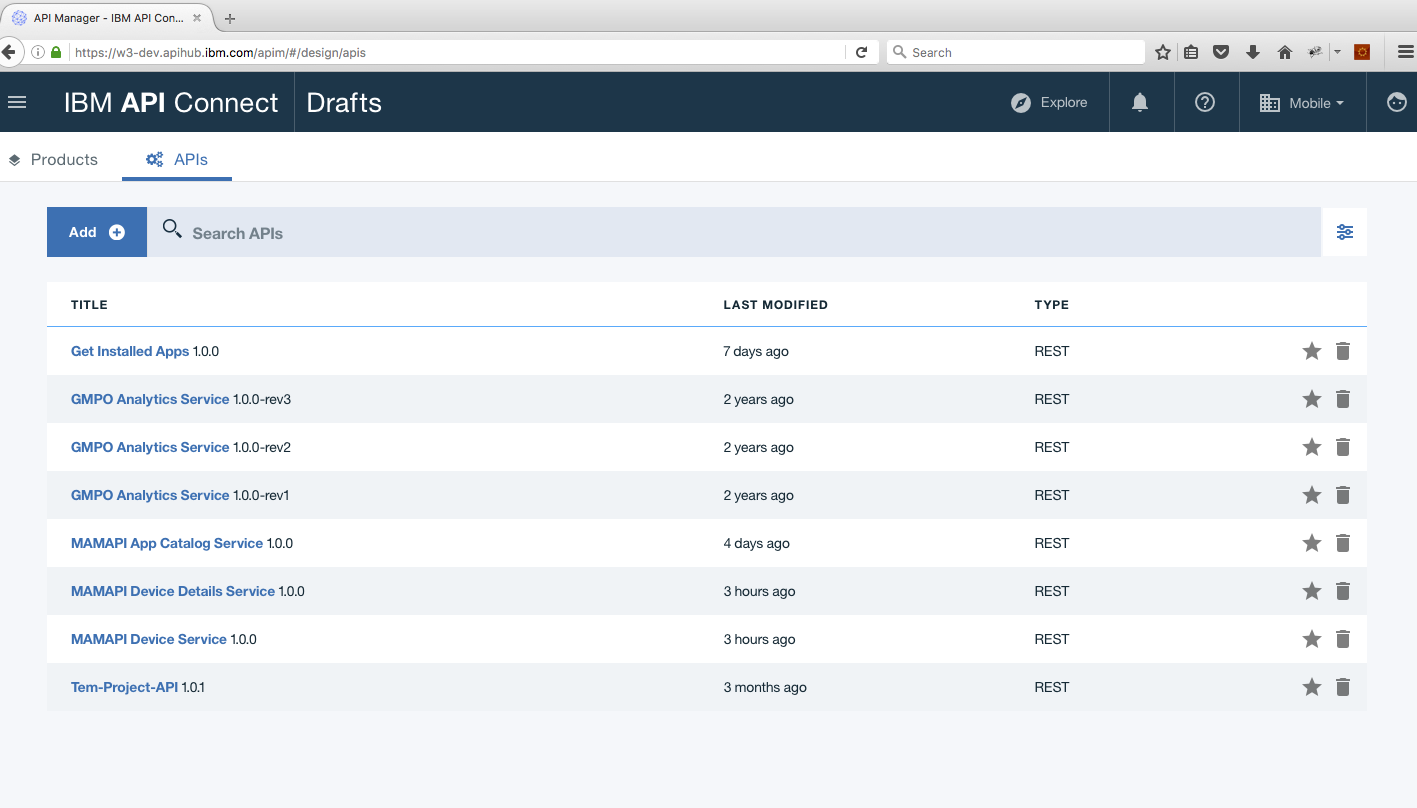
Promotion from Test to Run

API developers must not change any configuration in Test to ensure consistency of artifacts in Development and Test. (All changes to the API and Plans must be done in the Development environment.) Within Test, it is the responsibility of the Deployment Manager to maintain the integrity of the API and Plans as well as compliance with [API management standards](https://w3.apihub.ibm.com/blueapi/api_standards.html). After testing has completed and the API and Plans are ready for production (Run), the Deployment Manager can initiate promotion to Test from the Dashboard, API Lifecycle or scripts.

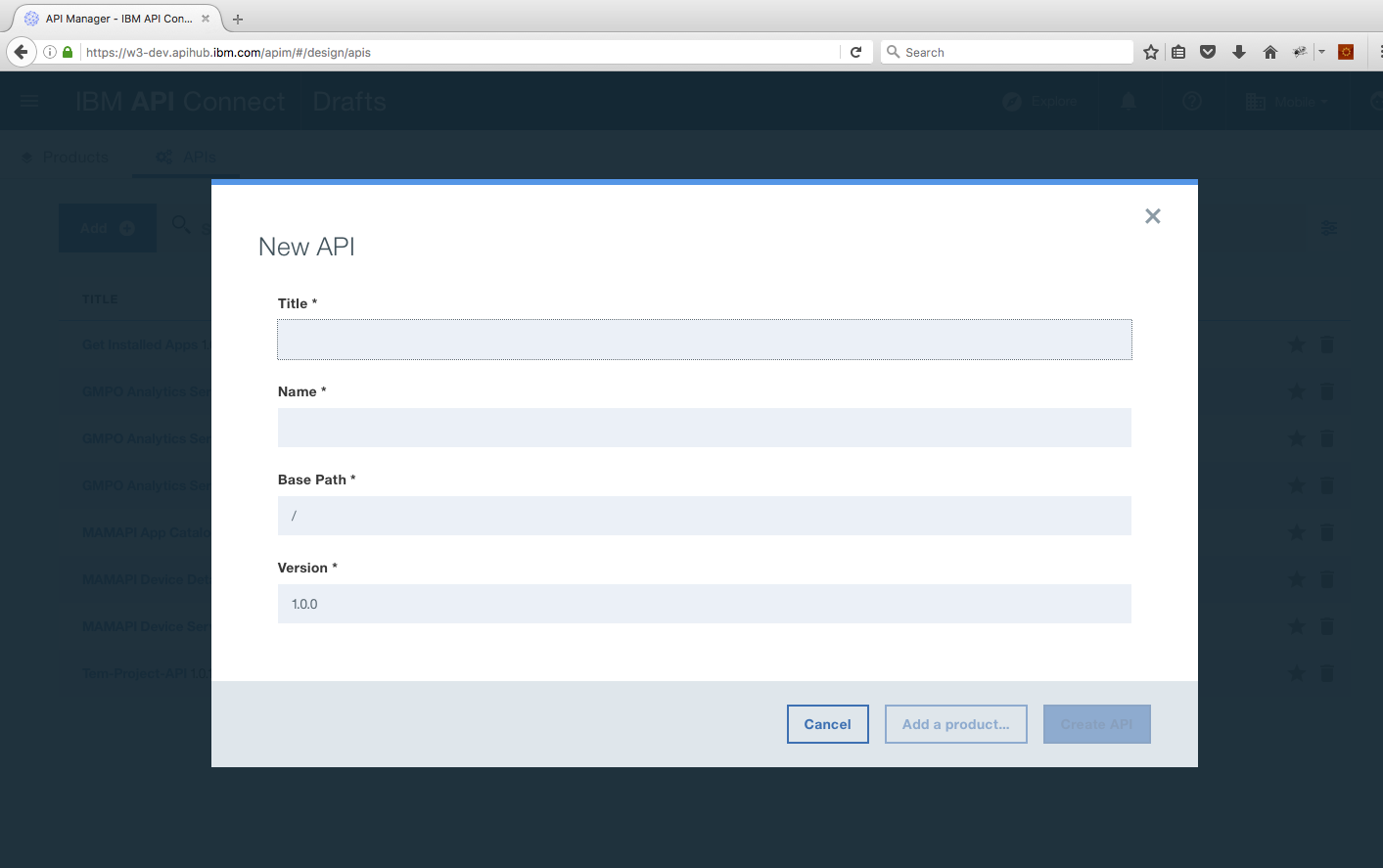
Developing API in API Manager:

1)Login to the Blue API UI https://w3-dev.apihub.ibm.com/apim with your intranet Id.

(API Developer access will be provided by Raghu)



For creating new API click on Add 🡪 New API



### Give the Title and Name of the API.API name should be in lowercase and should not contain special characters.

### Title format: MAMAPI “your service name” Service. Ex: MAMAPI App Catalog Service

### Name format: getappcatalog

### Version :0.0.1

### Creating a Product:

### Click on the Products Tab 🡪 Add 🡪 New product

### ../Screen%20Shot%202017-10-30%20at%206.10.17%20PM.png

### Title Format: MAMAPI “your API name” Team Ex: MAMAPI App Catalog Team

### Name Format: mamapi-app-catalog-team

### Version:0.0.1

### Notes:

### 1)API name and Product name will be in lowercase.

### 2)Contact field is mandatory for both API and PRODUCT.

### 3)Before promoting a particular product to TEST catalog, we should run the validation in Deployment Manager <https://w3.apihub.ibm.com:8080/blueapi/dashboard>

### 4)Version of the product and API should start with 0.0.1. For the final deployment in RUN catalog we will set the version of Product and API to “1.0.0”.

### 5)Below is the respective key for Catalog specific variable(Ex: target-url):

### Default 🡪RUN

### DEV 🡪 DEV

### TEST 🡪 TEST

### 6)In RUN catalog once the higher version product is deployed subscription will be automatically migrated.

### 7)In RUN catalog PM having RUN catalog access can’t retire the product.We need to raise a ticket for this.

### 8)App should be subscription based and there should be rate limit.

### 9)In RUN Product Manager will be able to approve and unsubscribe the subscription.

### 10)Important links:

### API Manager 🡪 <https://w3-dev.apihub.ibm.com/apim>

### App Developer Hub 🡪 <https://w3.apihub.ibm.com/developer/>

### Deployment Manager 🡪 <https://w3.apihub.ibm.com:8080/blueapi/dashboard>

### Support and Tutorial 🡪 <https://w3.apihub.ibm.com/blueapi/support.html>

### Support Ticket -> <https://ibm.support.ibmcloud.com/ics/support/mylogin.asp>