API MANAGEMENT

TG3-IMPLEMENTACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS

DESARROLLO CON TECNOLOGÍAS EMERGENTES

# 

Contenido

[1. Autores del trabajo, planificación y entrega 2](#_Toc448254544)

[1.1 Autores 2](#_Toc448254545)

[1.2 Planificación 2](#_Toc448254546)

[1.3 Entrega 2](#_Toc448254547)

[2. Requisitos del prototipo a implementar 3](#_Toc448254548)

[2.1 Requisitos funcionales 3](#_Toc448254549)

[2.2 Otros requisitos 3](#_Toc448254550)

[3. Criterios de comparación en la implementación 4](#_Toc448254551)

[3.1 Criterio 1: Nombre del criterio 4](#_Toc448254552)

[3.2 Criterio 2: Nombre del criterio 4](#_Toc448254553)

[3.N Criterio N: Nombre del criterio 4](#_Toc448254554)

[4. Proyecto de implementación de un prototipo del sistema utilizando la tecnología A 5](#_Toc448254555)

[4.1 Documentación de diseño 5](#_Toc448254556)

[4.2 Documentación de construcción 5](#_Toc448254557)

[4.3 Documentación de pruebas 5](#_Toc448254558)

[4.4 Documentación de instalación 5](#_Toc448254559)

[4.5 Manual de usuario 5](#_Toc448254560)

[5. Proyecto de implementación de un prototipo del sistema utilizando la tecnología B 6](#_Toc448254561)

[5.1 Documentación de diseño 6](#_Toc448254562)

[5.2 Documentación de construcción 6](#_Toc448254563)

[5.3 Documentación de pruebas 6](#_Toc448254564)

[5.4 Documentación de instalación 6](#_Toc448254565)

[5.5 Manual de usuario 6](#_Toc448254566)

[6. Comparación de las dos implementaciones 7](#_Toc448254567)

[6.1 Evaluación de los criterios en la implementación usando la tecnología A 7](#_Toc448254568)

[6.2 Evaluación de los criterios en la implementación usando la tecnología B 7](#_Toc448254569)

[7. Comparación de la implementación de las tecnologías 8](#_Toc448254570)

[8. Conclusiones 10](#_Toc448254571)

# 1. Autores del trabajo, planificación y entrega

## 1.1 Autores

Grupo T7:

* Laura Guillén Casal 03149840J (Coordinador)
* Francisco Miguel Sáez Bravo 50644226C
* Jesús Gail Bohórquez 09072386J
* Jorge Pérez Campos 09053996T
* Sergio Llanos García 09054884Z

## 1.2 Planificación

En este apartado se debe incluir copias de pantalla de la planificación del trabajo con diagramas Gantt: o bien un enlace (URL) a la web donde esté disponible la planificación si se ha utilizado una herramienta online de diagramación Gantt (por ejemplo, [Teamweek](https://teamweek.com/free-online-gantt-chart.html), [GanttPro](https://ganttpro.com/), [tomsplanner](https://plan.tomsplanner.es/), [sinnaps](https://www.sinnaps.com/), u otra).

Hay que tener en cuenta que cada participante del grupo debe tener asignadas tareas que sumen al menos 45 horas. El peso de este trabajo en la calificación total de la asignatura es de un 30%, por tanto requiere de una dedicación de 45 horas del total de 150 horas de la asignatura.

## 1.3 Entrega

En este apartado debe incluirse un enlace (URL) a un repositorio en GitHub creado para el trabajo.

En dicho repositorio debe encontrarse, al menos los siguientes archivos en la rama máster:

* Informe del trabajo: con el nombre TG3\_final.docx
* Presentación del trabajo: TG3\_final.pptx
* Prototipos obtenidos implementando cada una de las tecnologías (deben incluir el código fuente y todos los archivos necesarios para la instalación y uso de cada prototipo):
  + PrototipoTecnologiaA\_final.zip (o .rar)
  + PrototipoTecnologiaB\_final.zip (o .rar).

Dichos archivos serán los que se tendrán en cuenta para la calificación del trabajo.

# 2. Requisitos del prototipo a implementar

## 2.1 Requisitos funcionales

A continuación, encontramos una tabla con los requisitos funcionales comunes para las dos tecnologías utilizadas. Cabe destacar que la selección de estos requisitos funcionales se encuentra claramente condicionada por las funcionalidades presentes en las versiones de prueba y por lo tanto accesibles para nosotros

| **REQ.** | **DESCRIPCIÓN** |
| --- | --- |
| RF01- Creación/Publicación de una API | Capacidad de crear una API y realizar su posterior implementación |
| RF02- Suscripción a un API | Posibilidad de suscribirse a una API, pudiendo invocarla posteriormente |
| RF03- Políticas de seguridad | Introducción de métodos de autenticación |
| RF04- Edición de la API | Capacidad de modificar y adaptar el código y diseño de la API que hayas creado anteriormente |
| RF05- Control de acceso | Métodos de gestión de acceso concurrente a las APIs |
| RF06- API Store | Portal donde se alojan las APIs y se presentan al usuario para su futura suscripción |
| RF07- Servidor Proxy | Creación de un servidor encargado de recibir las peticiones y procesarlas |
| RF08- Herramientas de Análisis | Conjunto de herramientas que ofrece la plataforma para analizar una API, su rendimiento, funcionamiento, etc. |

## 2.2 Otros requisitos

En la siguiente tabla se muestran los requisitos no funcionales que se tendrán en cuenta para el estudio de ambas tecnologías:

| **REQ.** | **DESCRIPCIÓN** |
| --- | --- |
| R01- Portal de desarrolladores | Apartado de comunicación entre los desarrolladores y miembros de la comunidad |
| R02- Soporte de la compañía | Existencia de un servicio permanente de ayuda a los usuarios por parte de la compañía |
| R03- Documentación | Información suministrada por la compañía para el uso de la plataforma |
| R04- Personalización | Capacidad de customizar tu API |
| R05- Testeabilidad | Posibilidad de probar el código y funcionalidad de la API |

# 3. Criterios de comparación en la implementación

Se trata de criterios del tipo” “horas empleadas en el desarrollo del sistema”, “velocidad de funcionamiento del sistema”, “recursos necesarios”, etc.

## 3.1 Criterio 1: Dificultad del proceso de creación y publicación

*Nombre del criterio: Dificultad del proceso de creación y publicación.*

*Descripción: Grado de complejidad a la hora de crear y publicar una API.*

*Tipo de valor: Numérico (Escala 1-5).*

## 3.2 Criterio 2: Horas empleadas en la construcción del prototipo

*Nombre del criterio: Horas empleadas en la construcción del prototipo.*

*Descripción: Números de horas utilizadas para llevar a cabo la construcción de los prototipos.*

*Tipo de valor: Numérico (Horas).*

## 3.3 Criterio 3: Recursos necesarios

*Nombre del criterio: Recursos necesarios.*

*Descripción: Número de recursos que deben utilizarse para la utilización de las tecnologías.*

*Tipo de valor: Texto libre.*

## 3.4 Criterio 4: Configuración del servidor Proxy

*Nombre del criterio: Configuración del servidor Proxy.*

*Descripción: Complejidad y rapidez en la puesta en funcionamiento del servidor proxy.*

*Tipo de valor: Numérico (Escala 1-5).*

## 3.5 Criterio 5: Grado de personalización

*Nombre del criterio: Grado de personalización.*

*Descripción: Capacidad para modificar a gusto del usuario la interfaz de la API creada por el mismo.*

*Tipo de valor: Numérico (Escala 1-5).*

## 3.6 Criterio 6: Calidad del soporte

*Nombre del criterio: Calidad del soporte.*

*Descripción: Facilidad de contacto con el soporte de la plataforma y eficiencia en la respuesta de estos.*

*Tipo de valor: Numérico (Escala 1-5).*

## 3.7 Criterio 7: Calidad de la documentación

*Nombre del criterio: Calidad de la documentación.*

*Descripción: Nivel de detalle y claridad de la información que brinda la plataforma.*

*Tipo de valor: Numérico (Escala 1-5).*

## 3.8 Criterio 8: Facilidad de uso

*Nombre del criterio: Facilidad de uso.*

*Descripción: Grado de facilidad de uso y acceso a las distintas funcionalidades de las plataformas.*

*Tipo de valor: Numérico (Escala 1-5).*

## 3.9 Criterio 9: Implementación de seguridad y control de concurrencia

*Nombre del criterio: Implementación de seguridad y control de concurrencia.*

*Descripción: Capacidad de implementar seguridad y control de acceso a la API creada.*

*Tipo de valor: Numérico (Escala 1-5).*

## 3.10 Criterio 10: Profundidad del análisis

*Nombre del criterio: Profundidad del análisis.*

*Descripción: Cantidad y variedad de parámetros que ofrece la plataforma para el análisis de las APIs.*

*Tipo de valor: Numérico (Escala 1-5).*

## 3.11 Criterio 11: Capacidad de prueba

*Nombre del criterio: Capacidad de prueba.*

*Descripción: Posibilidad de realizar un test de prueba del código de la API.*

*Tipo de valor: Booleano (Si/No)*

# 4. Proyecto de implementación de un prototipo del sistema utilizando la tecnología A

Se trata de incluir en este apartado la documentación del desarrollo del proyecto de implementación, utilizando la tecnología A, del sistema cuyos requisitos funcionales se enumeraron en el apartado 2.

## 4.1 Documentación de diseño

Hay que incluir la descripción del diseño del prototipo, incluyendo diagramas, y el diseño de la interfaz de usuario.

## 4.2 Documentación de construcción

Hay que incluir una descripción de la construcción del prototipo, incluyendo algún extracto de código fuente. No es necesario todo el código. Sólo algún extracto para ver cómo se ha comentado.

## 4.3 Documentación de pruebas

Casos de prueba establecidos y resultados de las pruebas y acciones de corrección. No es creíble que no hayan aparecido errores en los caso de prueba.

## 4.4 Documentación de instalación

Descripción suficiente para que una persona que no ha participado en el proyecto pueda instalar el prototipo.

## 4.5 Manual de usuario

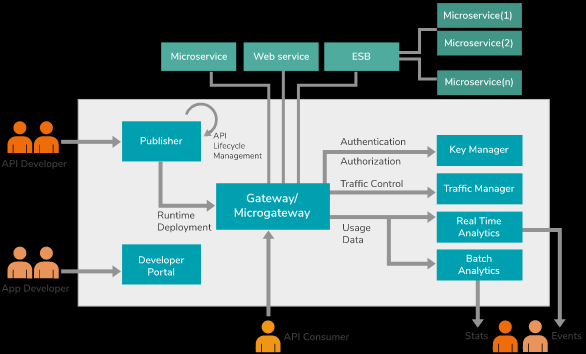
Descripción suficiente para que una persona que no ha participado en el proyecto pueda utilizar toda la funcionalidad que ofrece el prototipo. Que debe coincidir con los requisitos funcionales incluidos en el apartado 2.

# 5. Proyecto de implementación de un prototipo del sistema utilizando la tecnología WSO2

Partiendo de la base establecida por los requisitos funcionales y no funcionales definidos en el apartado anterior, y en base a las funcionalidades disponibles en las versiones de prueba que se nos han proporcionado, se plantea elaborar una simulación que recoja las características básicas de una plataforma de API Management, haciendo uso en este apartado concreto de la herramienta WSO2.

## 5.1 Documentación de diseño

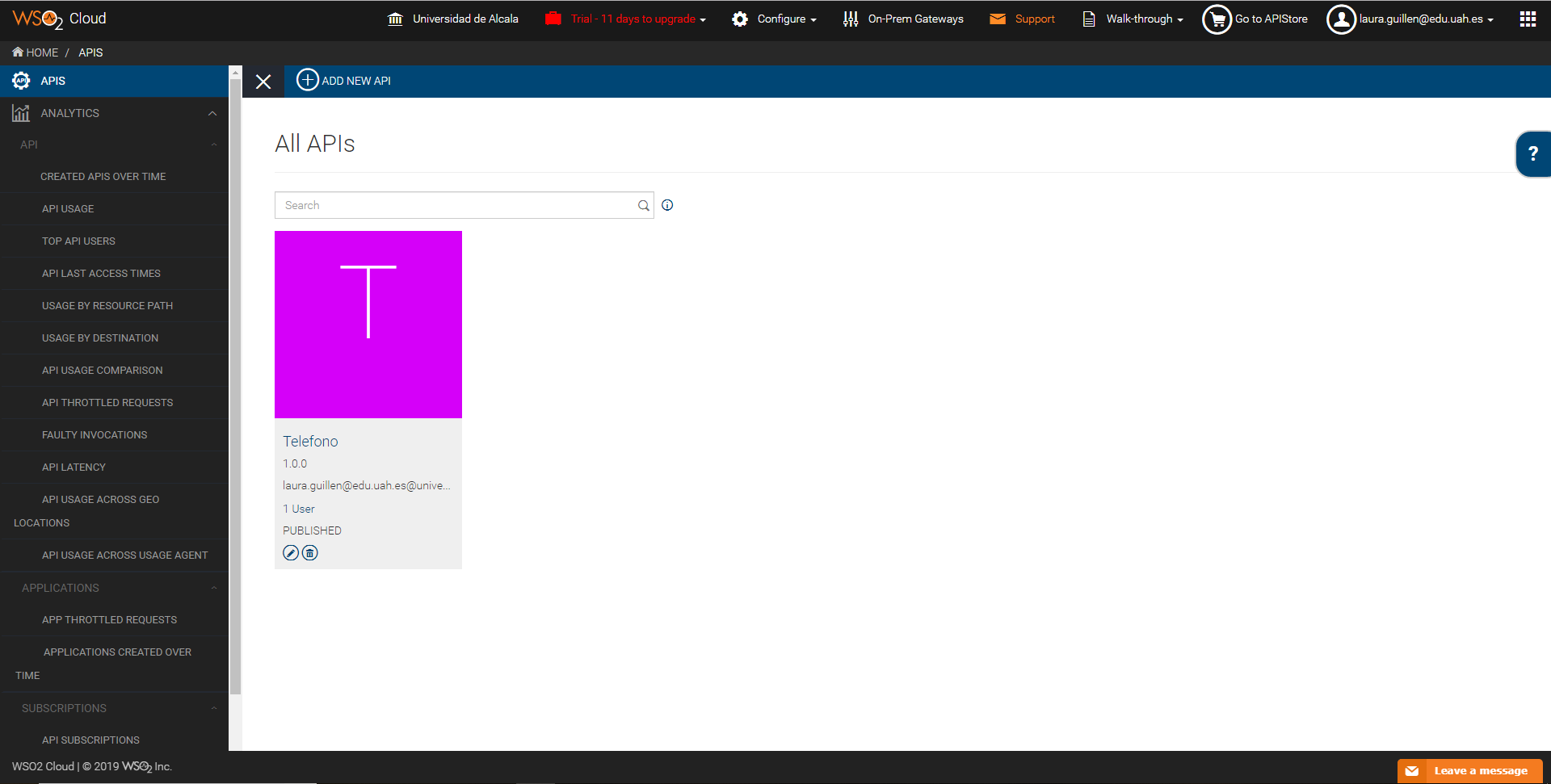
El prototipo que se va a realizar con la herramienta de API management de WSO2 consistirá en que, a partir de un fragmento de código, llevaremos a cabo la creación de una API, su gestión y su supervisión. Todo ello se realizará en la nube que nos brinda la plataforma de WSO2.



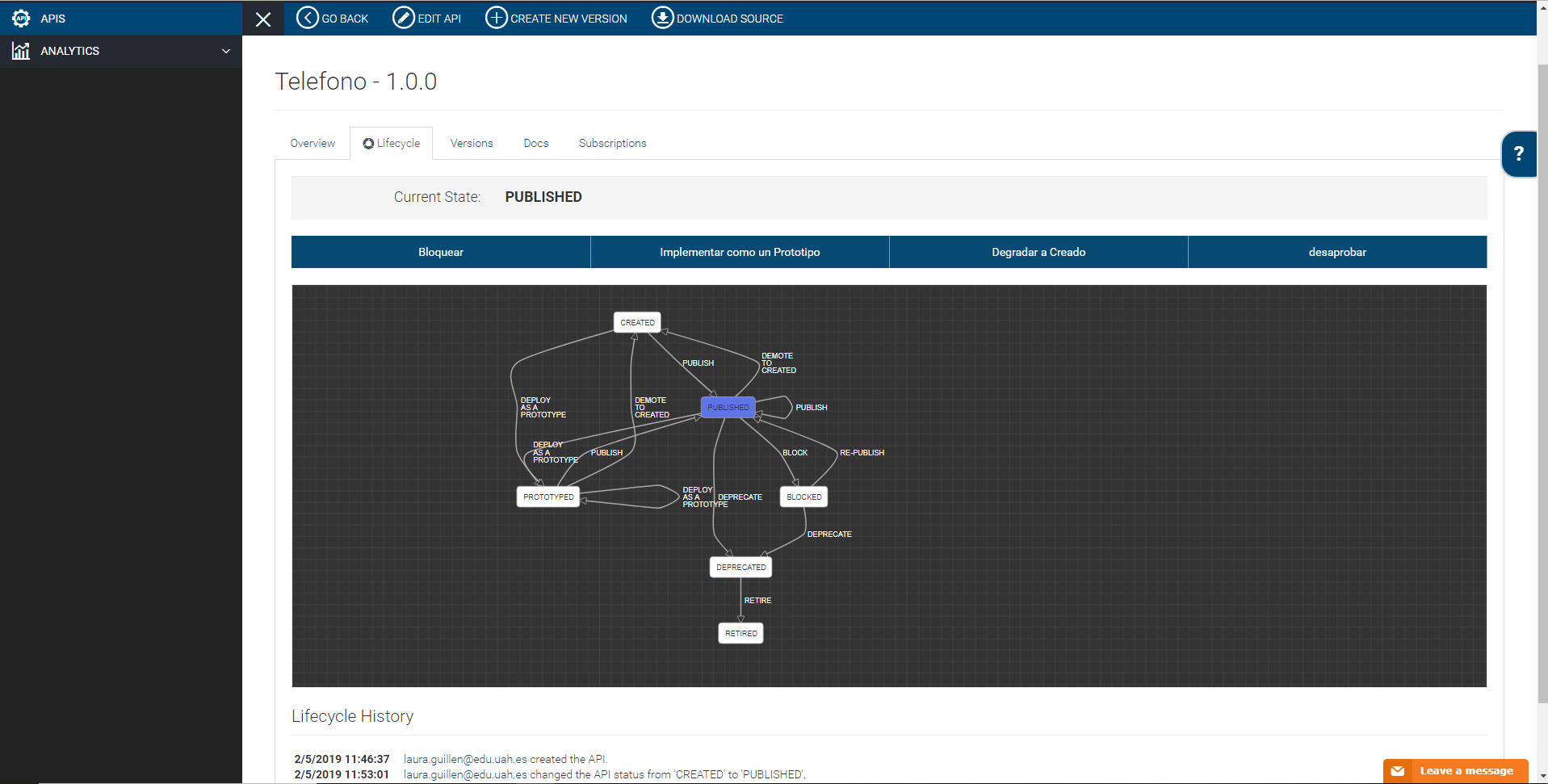
Puesto que se trata de la versión de prueba gratuita a la hora de realizar el prototipo no es posible acceder a todas las funcionalidades que nos ofrece la plataforma, como ejemplo destacable al usar la versión de prueba no es posible monetizar la API creada.

En cuanto a la interfaz de usuario que nos ofrece WSO2 se trata de una interfaz amigable para el usuario, intuitiva, pero con algunas dificultades a la hora de ser utilizada.

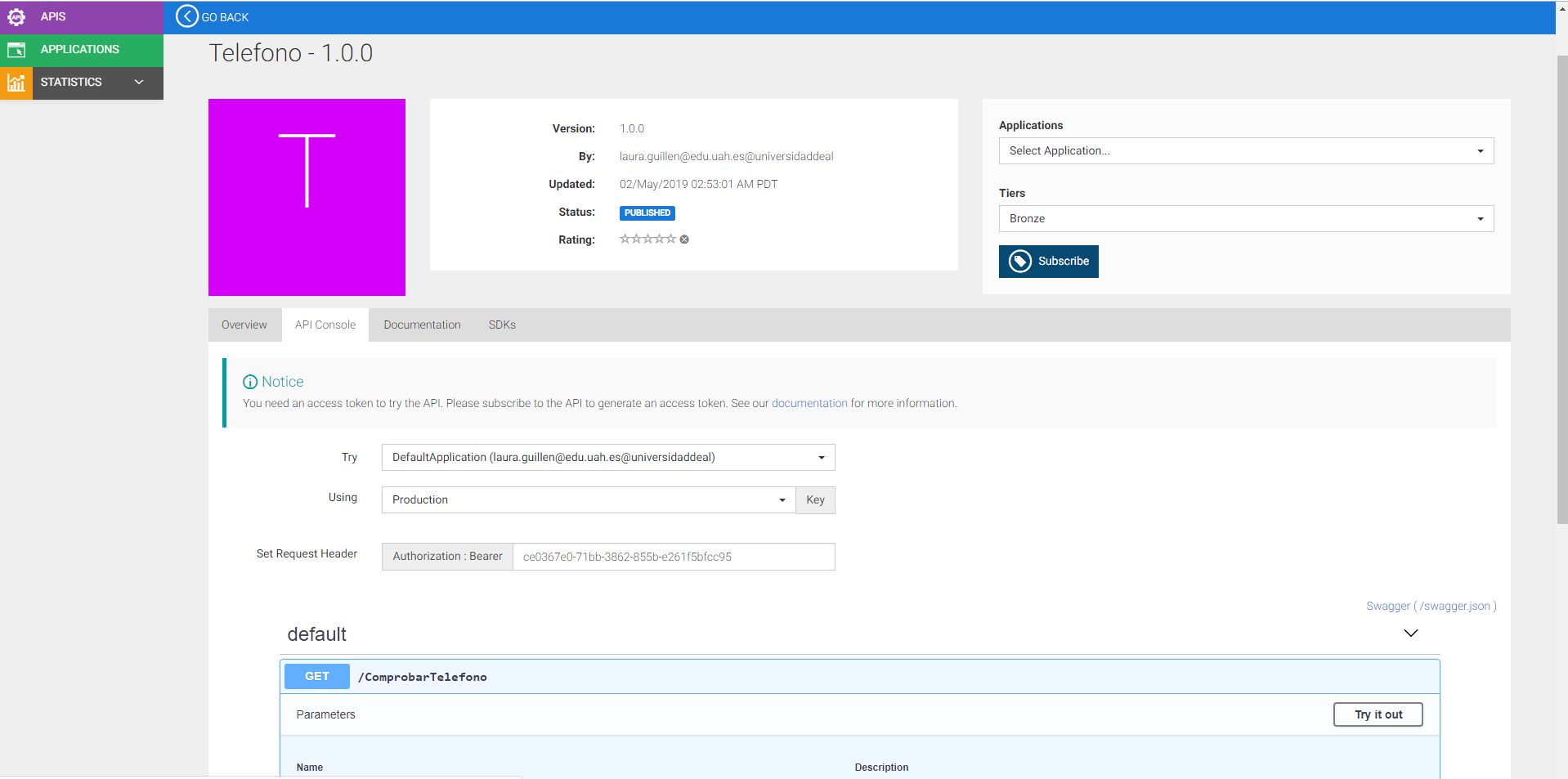
En la primera imagen podemos el diseño de la interfaz de usuario para la nube de WSO2, donde encontramos nuestras APIs creadas y las herramientas de análisis que se encuentran a disposición del creador.



Al clicar en nuestra API creada podemos encontrar la información de nuestra API, su ciclo de vida, sus versiones, la documentación que aportemos como creadores para los futuros usuarios de la API y el número de suscripciones que ha conseguido nuestra API. En la imagen se aprecia esta interfaz en la pestaña del ciclo de vida.

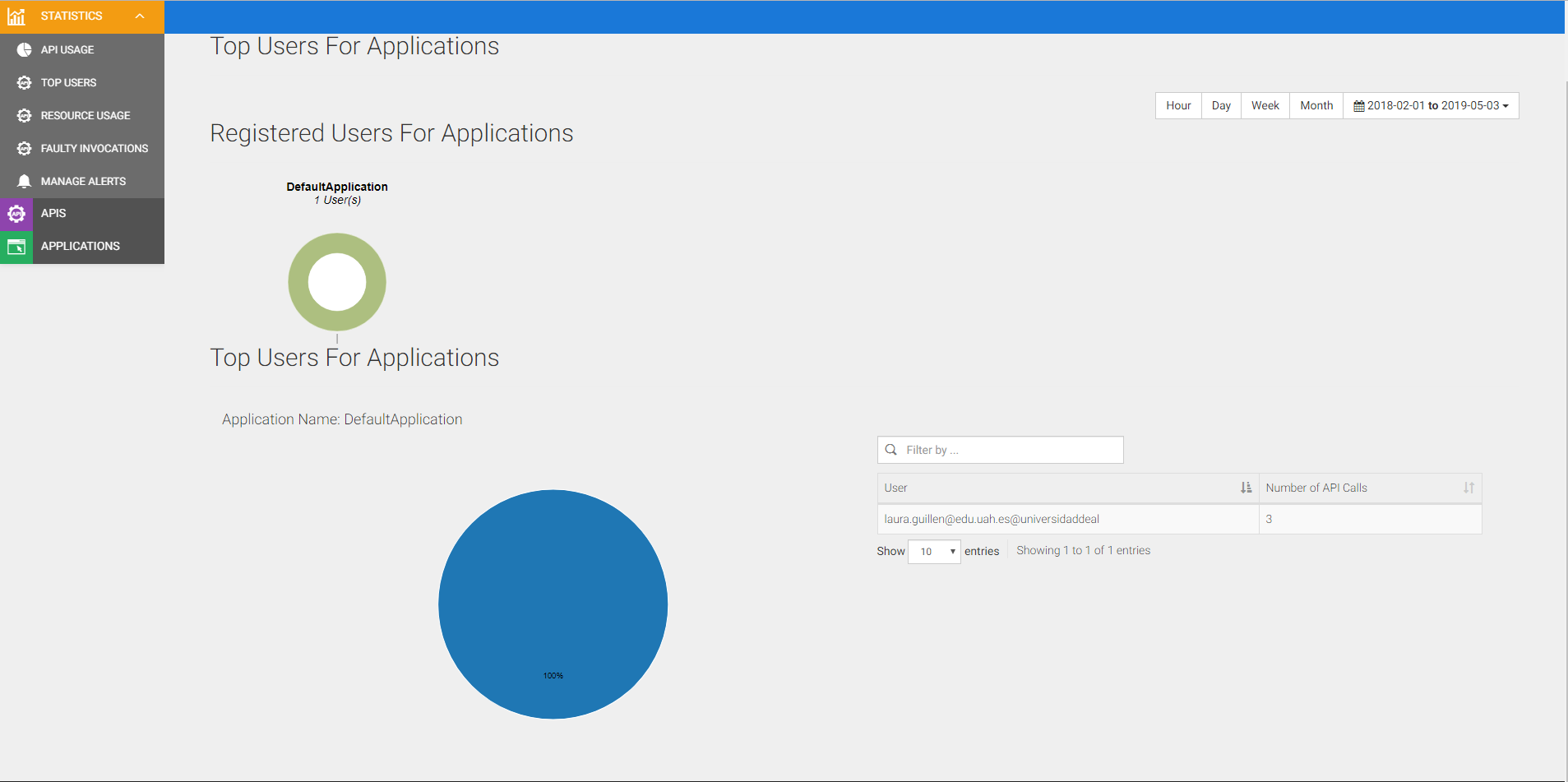


Por último, se mostrará la interfaz de usuario relativa a la API Store, lugar donde se puede llevar a cabo la suscripción a las distintas APIs así como la gestión de la API que hayas creado. Se puede editar, testear, añadir documentación y analizar dicha API a través de las herramientas de análisis que nos ofrece esta sección de la plataforma.



En la imagen nos encontramos en la sección de testeo de la API.

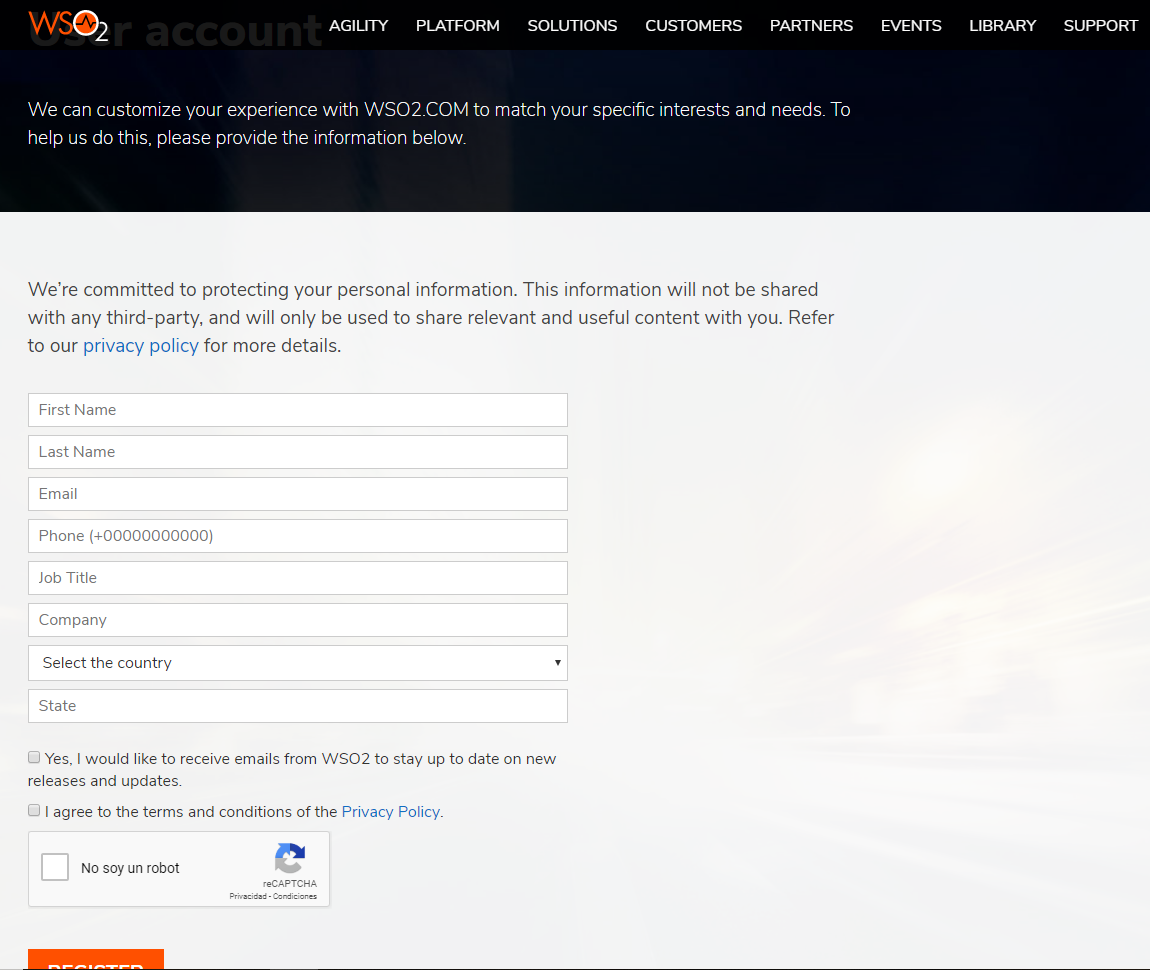
Por otro lado, la interfaz de estadísticas sería:



## 5.2 Documentación de construcción

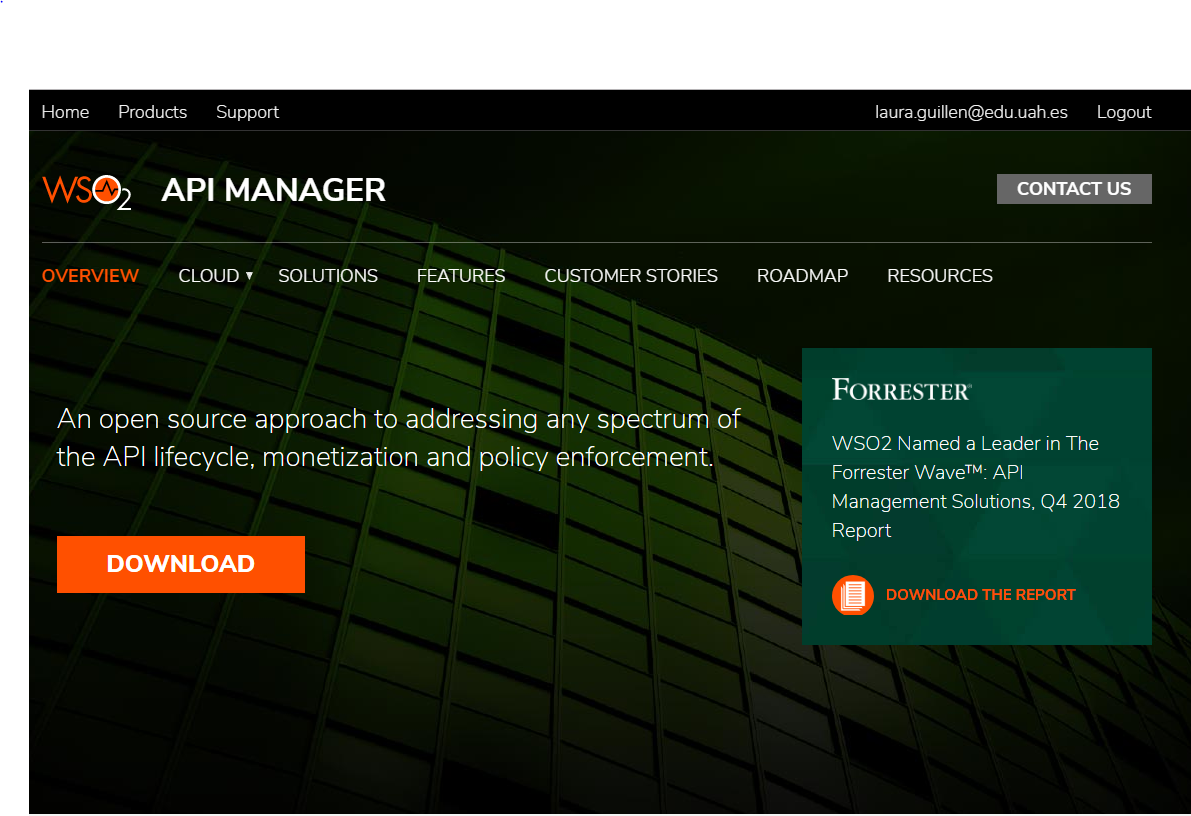
Para la construcción del prototipo con WSO2 debemos destacar que se realiza a través de la versión gratuita que nos permite la utilización de la nube que ofrece la plataforma de WSO2, esto se hace para equiparase con el prototipo que se realizará en la plataforma de APIGEE.

Para comenzar la construcción de nuestro prototipo lo primero que debemos hacer es registrarnos en WSO2 y solicitar la versión de prueba de 15 días.

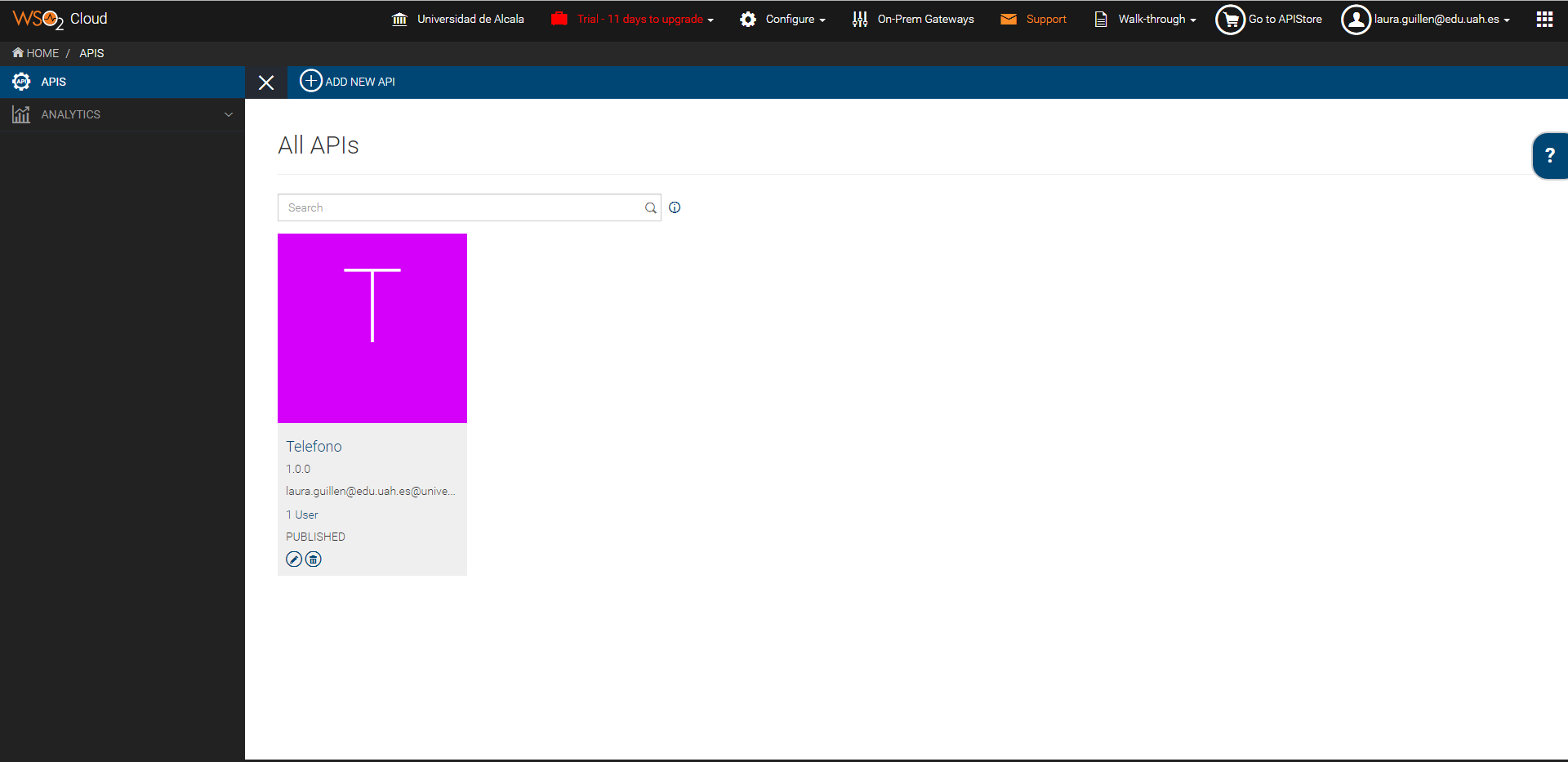


Una vez realizado este registro, WSO2 nos permitirá usar la versión de prueba por lo que deberemos iniciar sesión en la plataforma.

Tras haber iniciado sesión deberemos seleccionar el apartado de API Manager en el apartado llamado PLATAFORM.



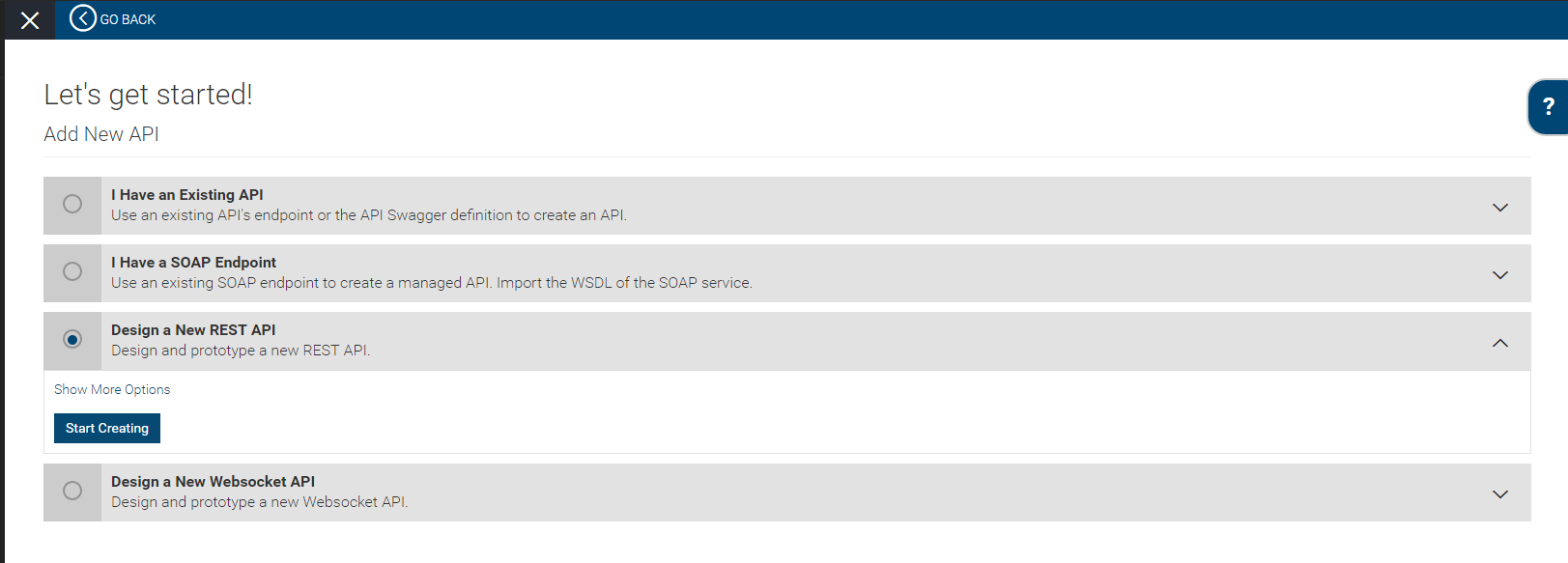
Debido a ello ya nos encontramos en la sección de API Manager que de todas las que hay en la plataforma es la que nos interesa para la construcción de nuestro prototipo. A continuación, seleccionaremos la pestaña de cloud y deberemos iniciar sesión de nuevo con la misma cuenta que se había hecho anteriormente, por lo que nos encontraremos en la siguiente pantalla:



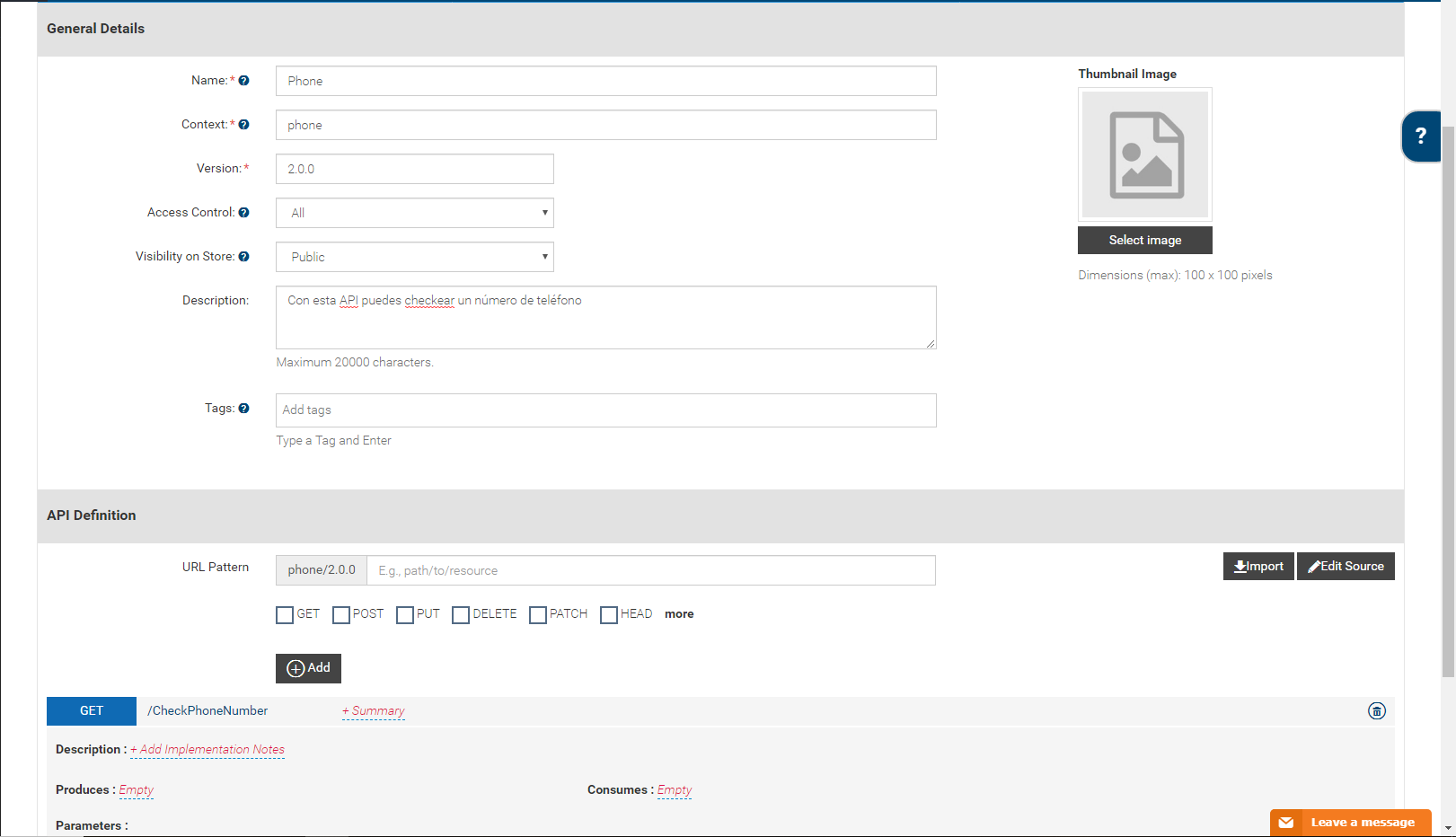
Una vez aquí, comenzaremos con la creación de una nueva API.

Empezaremos seleccionando la pestaña llamada ADD NEW API que se puede apreciaren la parte superior de la imagen.

Tras esto nos aparecerá una nueva pantalla en la que seleccionaremos Desing a NEW REST API.

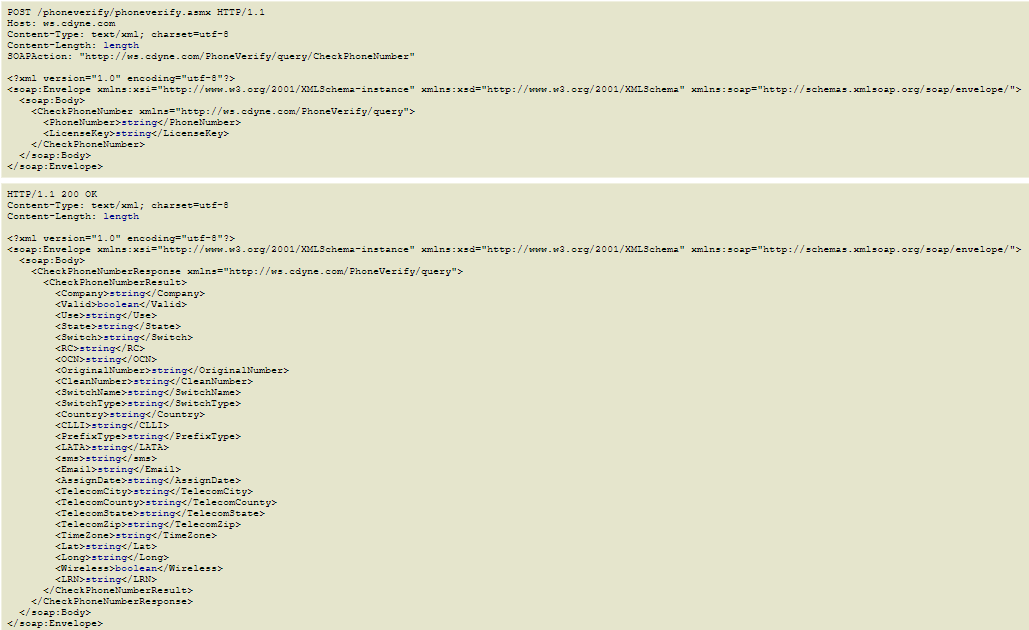


Ahora nos encontramos con una pantalla donde debemos rellenar una serie de campos, los detalles de nuestra API y su definición.



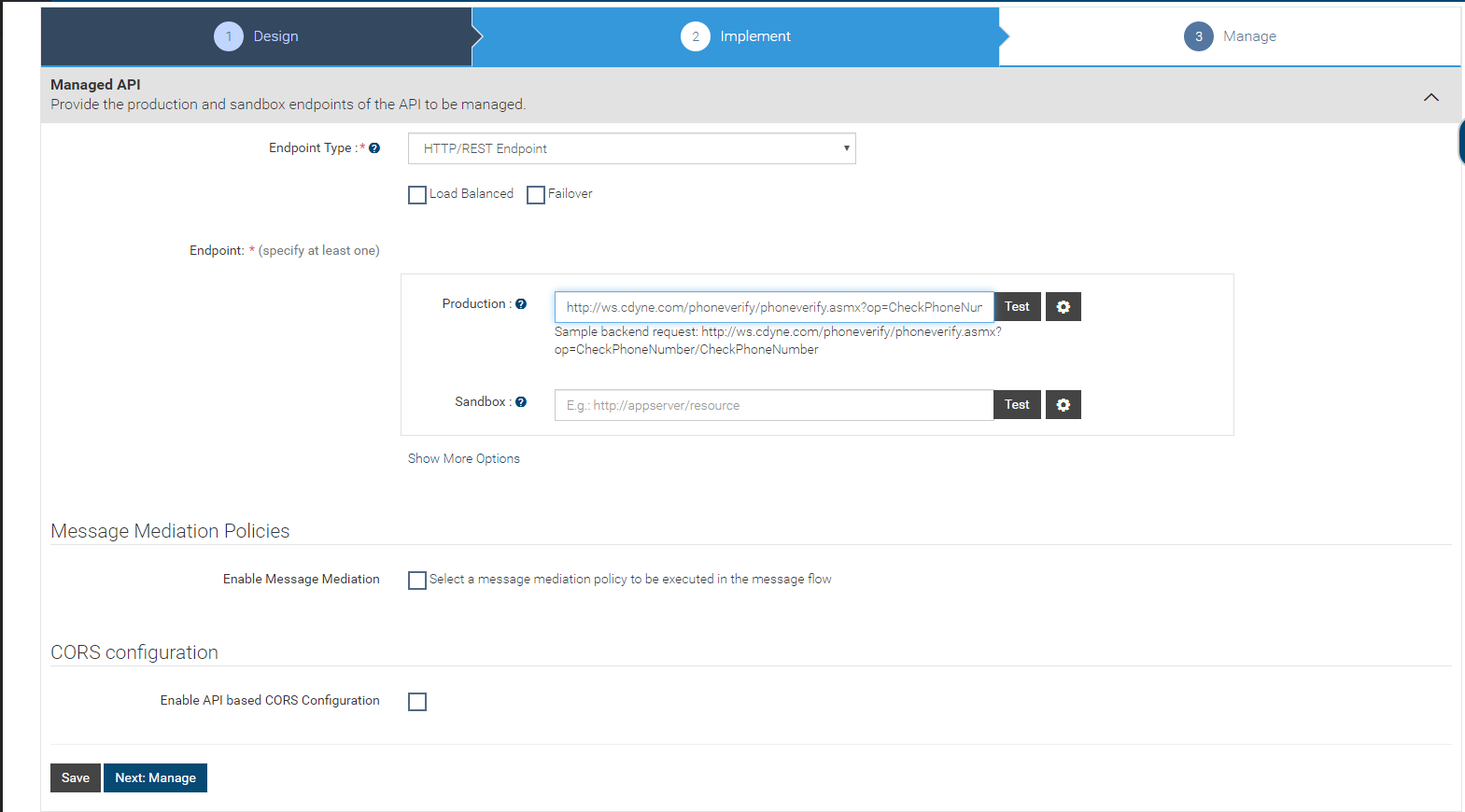
Una vez rellenados estos datos deberemos seleccionar la opción de prueba para una API, que nos permite el uso de todas las funcionalidades del API management dentro del rango de características que nos brinda la versión de prueba.

En este caso utilizaremos como ejemplo el código de una API ya creada en que si tu le introduces como parámetros un número de teléfono y un código de licencia nos verificara si ese número es real o no. Su código sería:



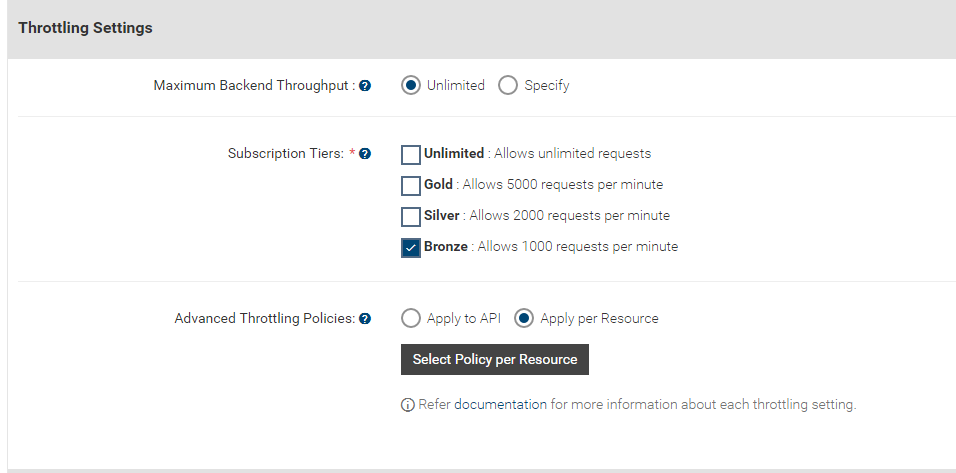
Este código ha sido obtenido de:

(<http://ws.cdyne.com/phoneverify/phoneverify.asmx?op=CheckPhoneNumber>)

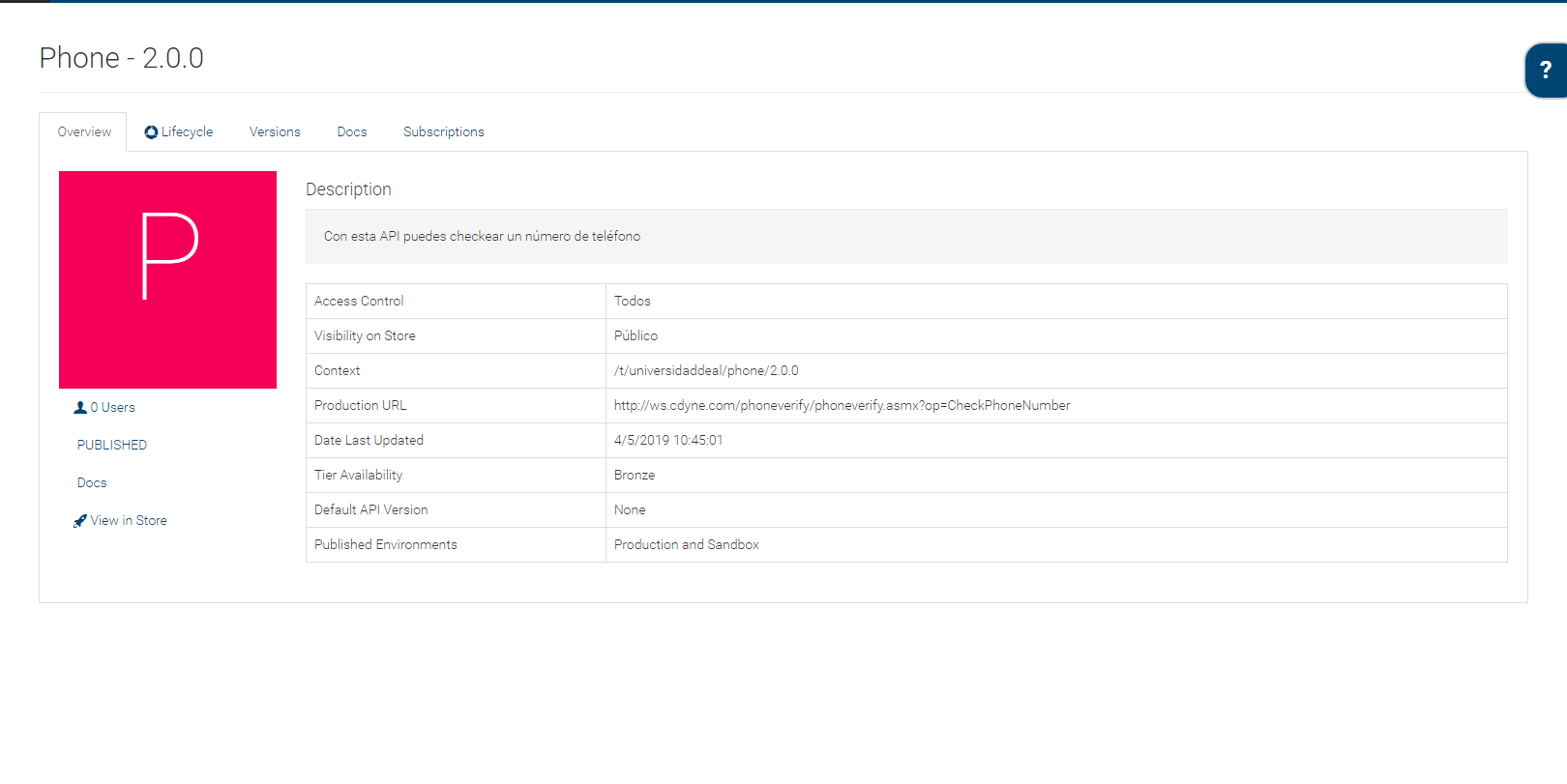


Introducimos dicha URL y continuamos.

Como último paso para la creación de nuestra API seleccionaremos el número de llamadas por minuto que se permitirá, en nuestro caso seleccionaremos la opción de bronce.



Con esto ya abríamos finalizado nuestro proceso de creación de una API, a partir de este momento nuestra API se encuentra en el API Store de WSO2, donde se puede trabajar con las distintas herramientas que ofrece el API Managment para la gestión y explotación de nuestra API. Además, otros usuarios podrán suscribirse a nuestra API para su uso a cambio de una compensación económica para el creador de la API.



## 5.3 Documentación de pruebas

Al comenzar a trabajar en el prototipo planteado de creación de una API para estudiar el funcionamiento de API Management, decidimos utilizar la nube que ofrece la plataforma de WSO2 para así equipararnos al otro prototipo que también se ha llevado a cabo en este proyecto.

El primer intento que se hizo una vez ya inscritos en la plataforma de WSO2 fue utilizar el paquete opensource que ofrece esta compañía para el API Management, cabe destacar que su instalación y puesta en funcionamiento es complicada y además no nos era de gran utilidad para este proyecto pues la comparación con el otro prototipo no sería la adecuada dado que los dos se deberían utilizar desde la nube.

Tras este primer intento conseguimos iniciar una versión de prueba de 15 días en la nube de WSO2. Una vez ya registrados comenzamos a investigar todo los que nos ofrece esta sección de API Management.

El siguiente intento consistió en crear una nueva API a partir de un código realizado por nosotros; este intento fue fallido, por nuestra nula experiencia con este tipo de herramientas y a la hora de crear APIs. Por este motivo después de un largo tiempo de intentos y de documentarnos, decidimos recurrir al soporte que nos ofrece WSO2. Tras recibir su respuesta con una serie de tutoriales y nueva información decidimos crear nuestra API a partir de un código ya existente.

Este último intento y definitivo es el que hemos utilizado como prototipo, puesto que conseguimos crear nuestra API y comenzar a trabajar con ella, en la suscripción, control de acceso, edición, etc.

## 5.4 Documentación de instalación

Nuestro prototipo como tal no necesita ningún tipo de instalación, pues se traba a través de la nube de WSO2 a la que se accede desde la página web de la misma organización (<https://wso2.com/api-management/>). Por tanto, nuestro proceso de instalación para que nuestro prototipo se encuentre operativo sería similar al proceso de creación que ha sido explicado en detalle en el apartado 5.2.

La única instalación que se ha llevado a cabo ha sido la de la herramienta y librería curl para realizar una serie de pruebas en las que se comprueba si existe conexión con el servidor. Curl actúa sobre la consola de comandos del sistema y nos proporciona una respuesta, en caso de que la conexión con el servidor se complete nos devolverá los parámetros introducidos en la prueba y en caso negativo nos devolverá un error. Un ejemplo de un comando para el uso de curl puede ser el siguiente: *curl -k -X GET "https://gateway.api.cloud.wso2.com:443/t/universidaddeal/phone/2.0.0/CheckPhoneNumber?PhoneNumber=918814042&Licencia=0" -H "accept: application/json" -H "Authorization: Bearer ".*

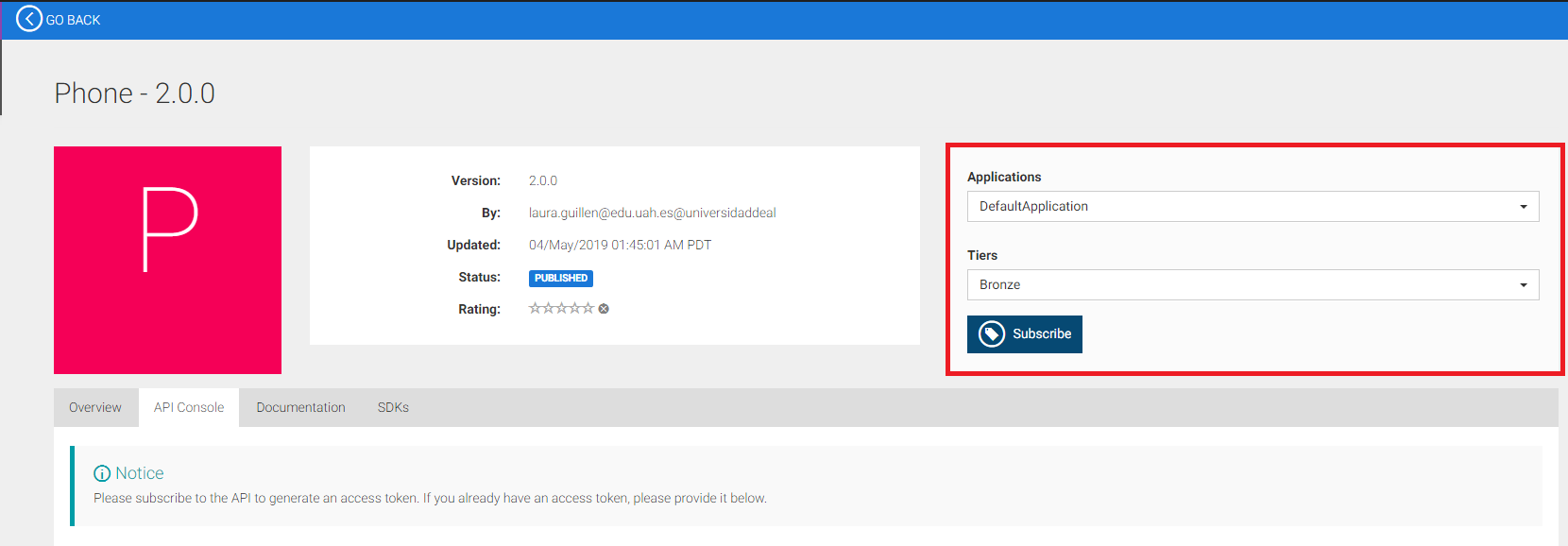
Para la instalación de esta herramienta y librería únicamente es necesario descargarse el archivo .rar que se encuentra en su página web (<https://curl.haxx.se/>). Una vez lo tengamos descargado lo recomendable es crear una carpeta en el disco C: llamada mycurl donde introduciremos el archivo curl de tipo aplicación, esto se hace para que su llamada desde el símbolo del sistema nos sea más sencilla.

## 5.5 Manual de usuario

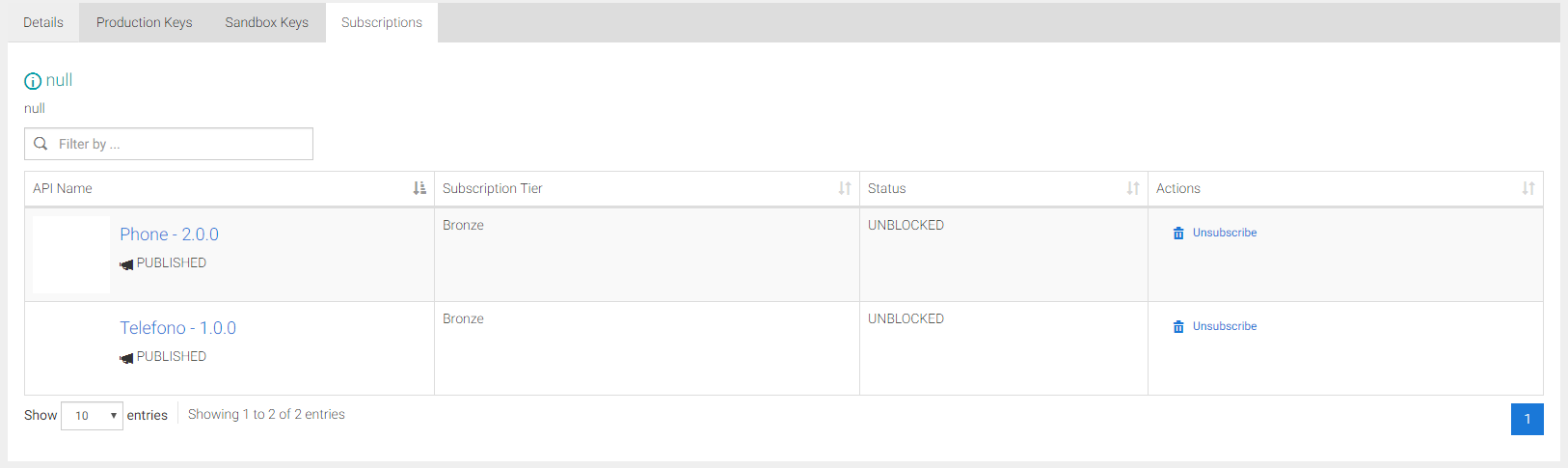
En este caso para el manual de usuario partiremos de la base de que ya tenemos creada nuestra API en la nube de WSO2 como ya ha sido comentado en el apartado 5.2. A partir de este momento se detallarán las distintas funcionalidades que nos ofrece esta plataforma para realizar un proceso de API Management con la API que hemos creado.

En primer momento comenzaremos con la suscripción a una API y su posterior invocación. En este caso realizaremos este proceso con nuestra API para que se veo mejor el proceso y el resultado de la suscripción.

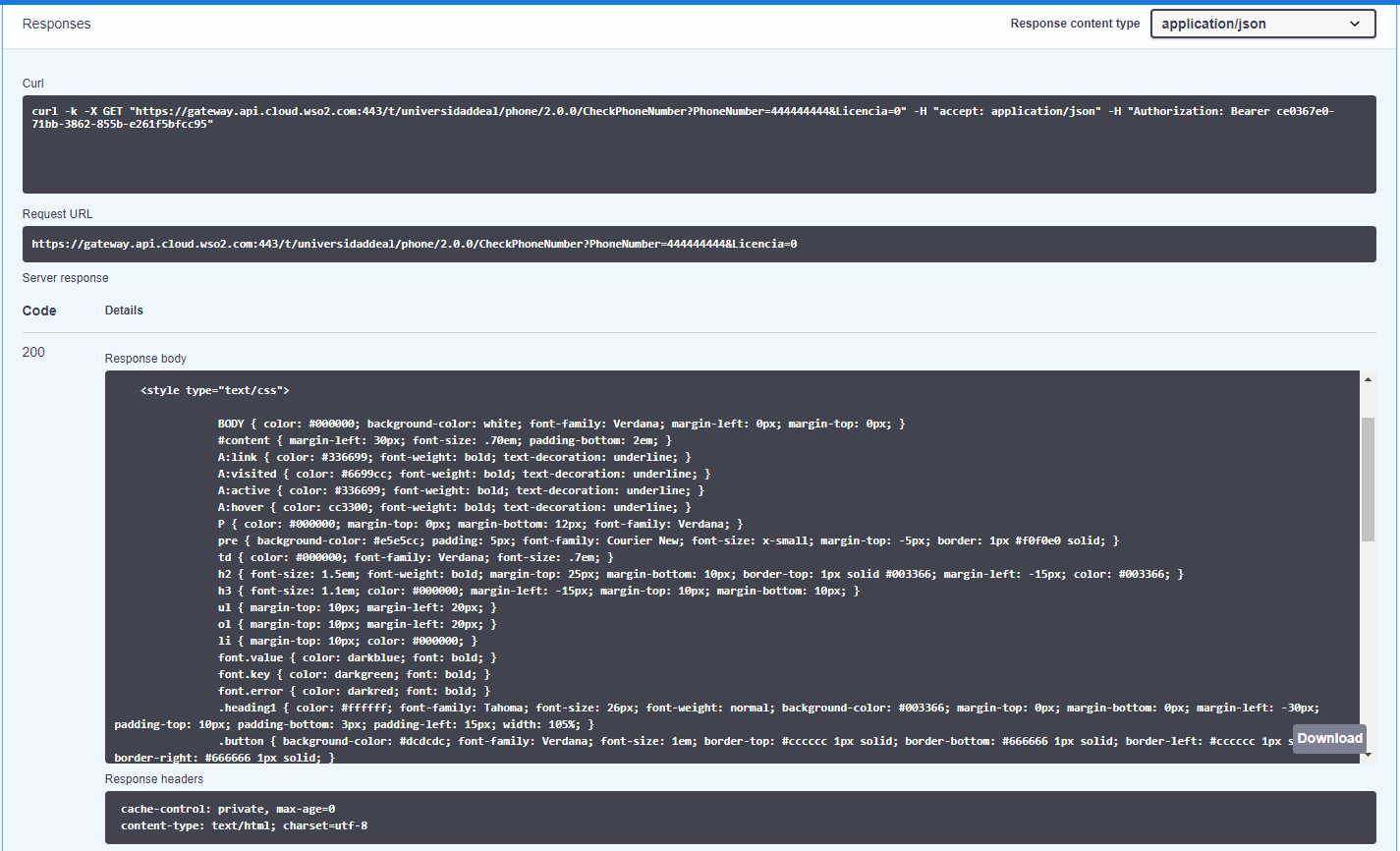
Para ello comenzaremos clicando en el API Store, una vez nos encontremos allí deberemos seleccionar la API a la que queremos suscribirnos, ir al recuadro de suscripción, seleccionar mi aplicación por defecto y el número llamadas por minuto que se realizarán. En la siguiente imagen se aprecia de una mejor forma.



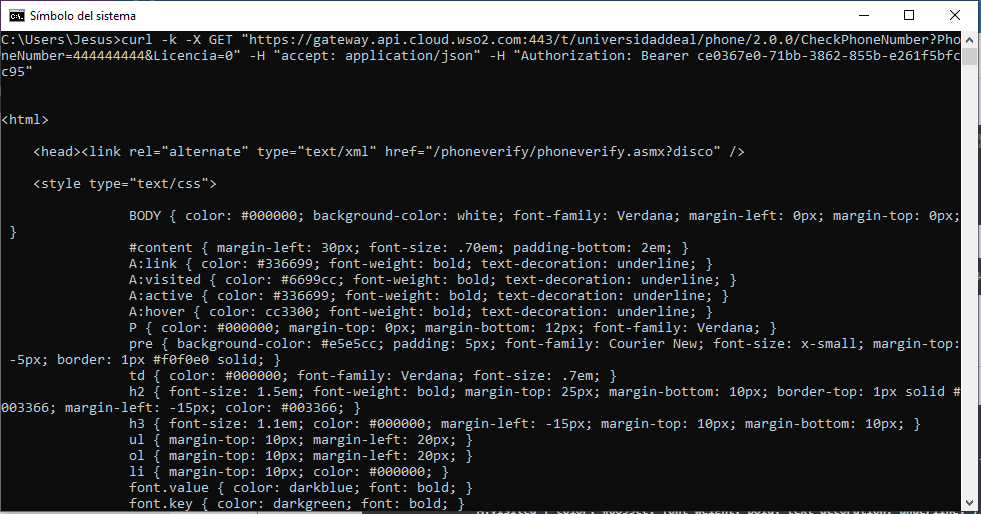
Después de aceptar la suscripción nos aparecerá una pantalla con de nuestra cuenta indicándonos a que APIs estamos suscritos.



Ahora deberemos realizar la invocación de la API, únicamente debemos encontrarnos en la pestaña de API Console donde introduciremos los parámetros que se nos pidan y el resultado sería el siguiente.



Para asegurarnos que la conexión con el servidor se realiza de una forma adecuada podemos usar la línea de curl que se nos proporciona desde el símbolo del sistema (cmd).

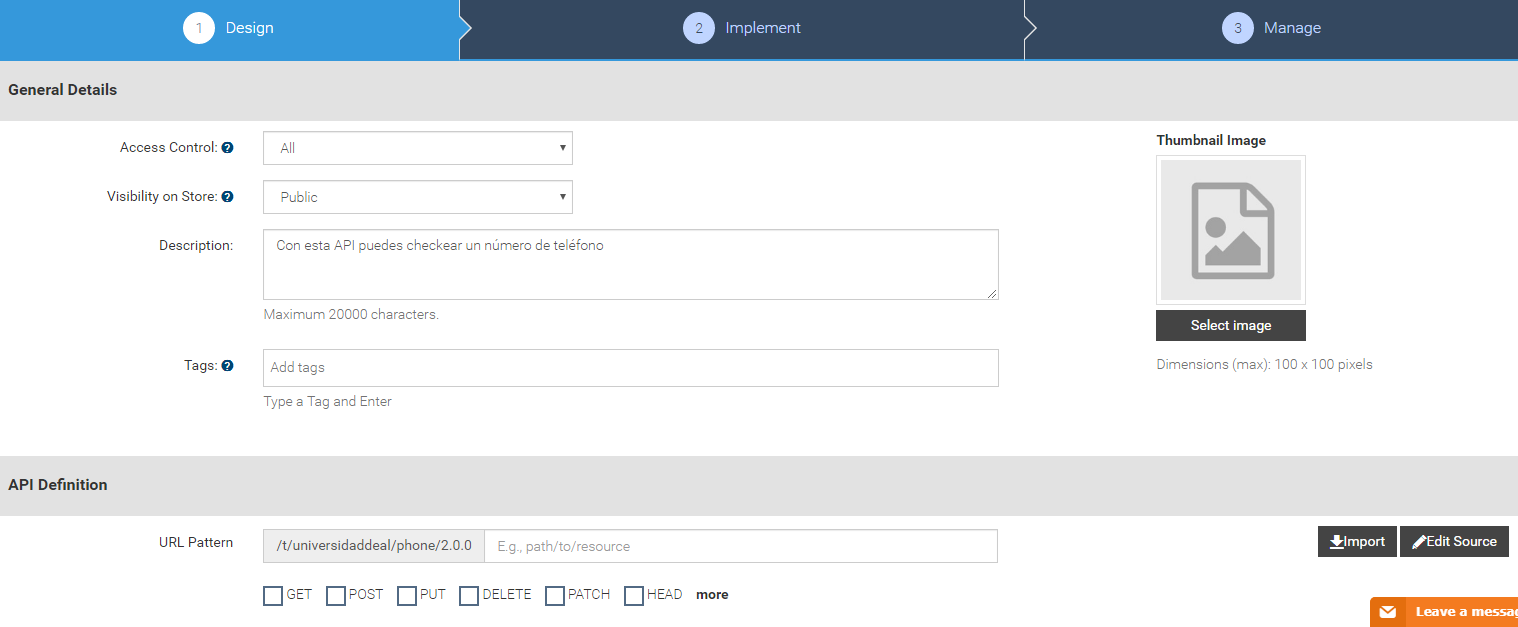


Si el resultado es óptimo como es el caso, la conexión será la adecuada y nos aparecerá el código de la API, de no ser así nos dará como resultado un error. Con todo lo anterior hemos conseguido suscribirnos a una API e invocarla.

A partir de este momento podemos realizar las distintas actividades de API Management como puede ser el control de acceso. Este control se puede seleccionar en el periodo de construcción de la API, donde tu como creador tienes la opción de elegir el acceso concurrente que deseas para tu API, hay 4 niveles a elegir, que serían Unlimited, Gold, Silver y Bronze, en nuestro caso decidimos elegir el más bajo (nivel bronce) que permite 1000 solicitudes por minuto.

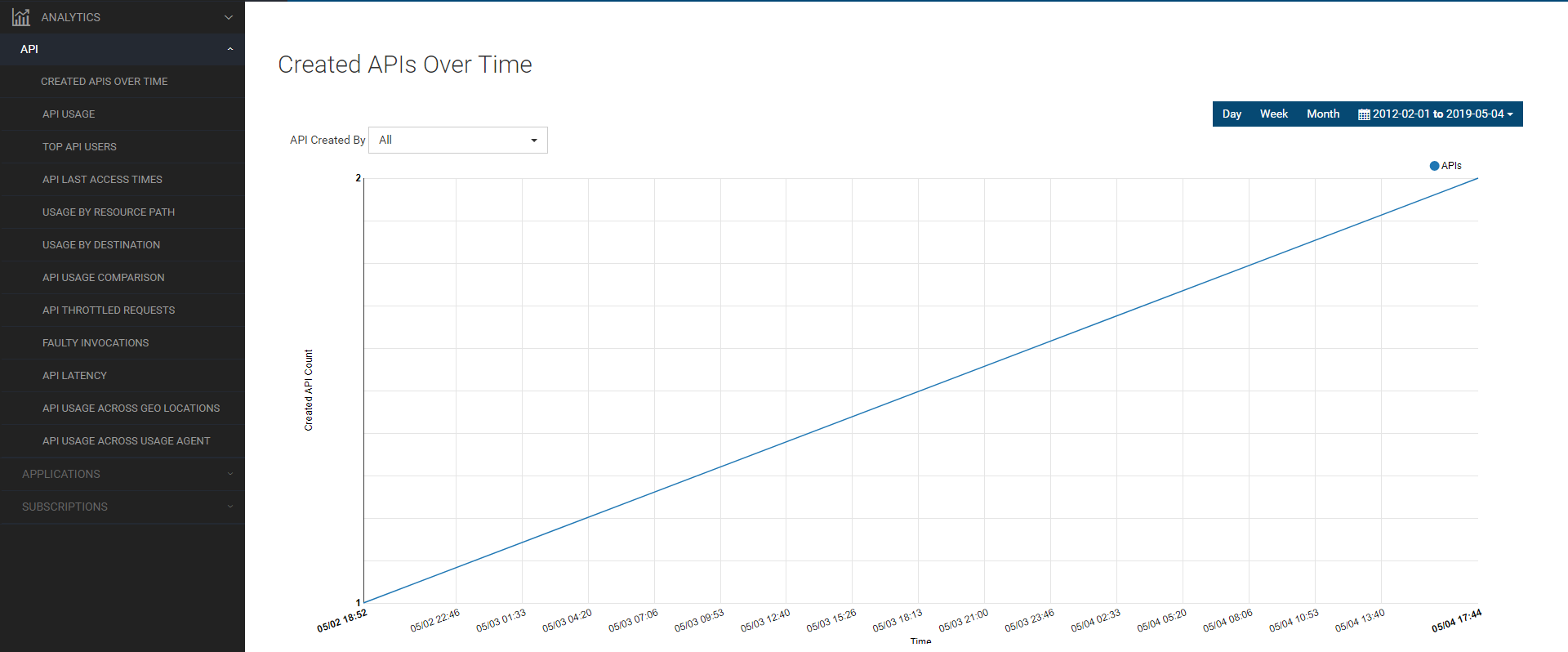
Las políticas de seguridad también se eligen en el proceso de creación, donde debes elegir entre un libre acceso para todos los usuarios o un acceso restringido para una serie de usuarios ordenador por roles. Al ser la versión de prueba en nuestro caso seleccionamos la opción de que todos los usuarios pudieran acceder al contenido de nuestra API.

Para la edición de tu API una vez creada deberás encontrarte en la nube de WSO2 y clicar el botón de EDIT API, donde podrás editar todo lo relacionado con tu API y los puntos mencionados anteriormente como si se tratase de un nuevo proceso de creación.

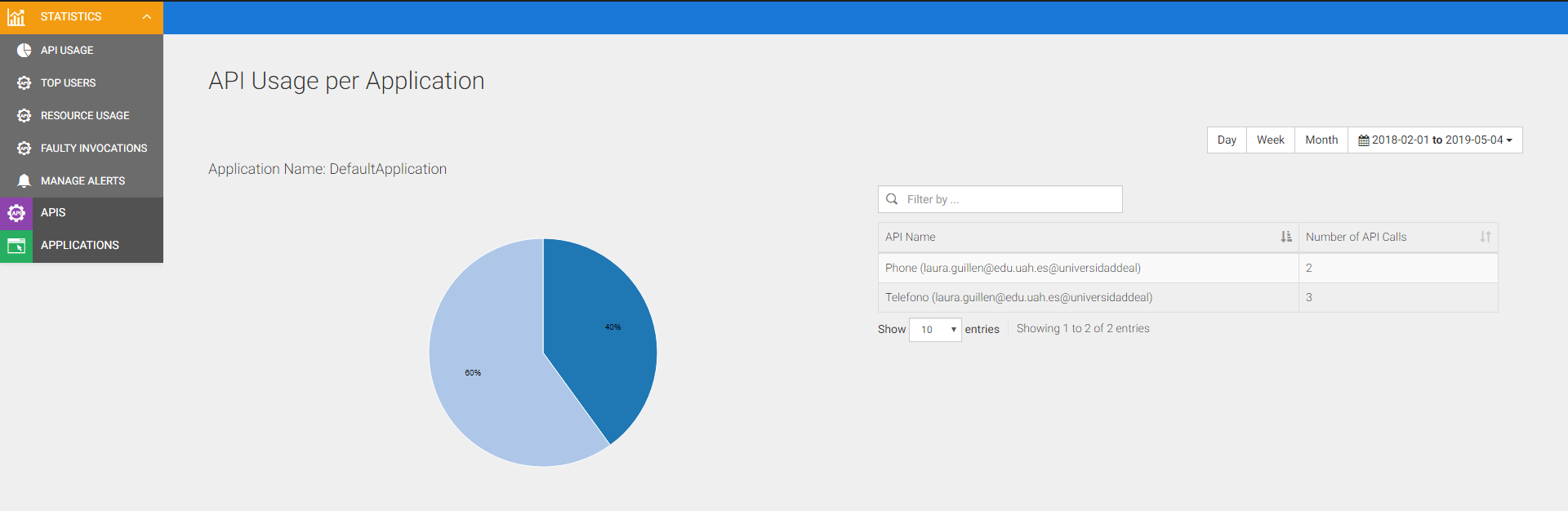


En el caso de WSO2 no es necesario instalar un servidor proxy si se va a trabajar con la nube como se ha mencionado anteriormente pues WSO2 en este caso lo tiene por defecto.

Por último, para poder acceder a las herramientas de análisis de tu API, es necesario que cliques en la pestaña de ANALYTICS, donde encontrarás un gran número de parámetros a analizar sobre el uso que se está haciendo de tu APIs, como su uso, número de llamadas, comparaciones entre tus APIs, etc.



Por otro lado, si queremos un análisis más individual para cada API y no tan general nos vemos en la obligación de clicar en el API Store donde en el apartado de STATISTICS, podremos observar nuevas estadísticas sobre nuestras APIs.



# 6. Comparación de las dos implementaciones

Se trata de dar valores a los criterios de comparación definidos en el apartado 3 sobre la implementación de cada uno de los prototipos.

## 6.1 Evaluación de los criterios en la implementación usando la tecnología Apigee

| **CRITERIO** | **EVALUACIÓN** |
| --- | --- |
| Dificultad del proceso de creación y publicación | La dificultad en el proceso de creación y publicación de nuestras APIs ha sido relativamente sencillo una vez que tenemos configurado el proxy, que es la parte más ardua.  Valoración: |
| Horas empleadas en la construcción del prototipo | Aproximadamente la construcción de este prototipo puede llevar de 2 a 3 horas debido a la dificultad de la creación el proxy, una vez sobrepasado este paso se requiere menos tiempo.  Valoración: |
| Recursos necesarios | Para la creación del prototipo basado en la tecnología APIGEE es necesario el uso de un servidor Proxy para ejecutar las peticiones del usuario.  Valoración: |
| Configuración del servidor Proxy | Una vez iniciemos sesión en la plataforma podemos crear y configurar nuestro propio proxy en el que podremos indicarle el nombre, la ruta, descripción, seguridad, etc. Y una vez creado podremos testearlo y hacer modificaciones en el futuro.  Valoración: |
| Grado de personalización | La plataforma de Google nos permite personalizar nuestras APIs de tal forma que se realiza a través del código fuente de la misma API para que cumpla la funcionalidad que nosotros queramos.  Valoración: |
| Calidad del soporte |  |
| Calidad de la documentación | La calidad de la documentación prestada por APIGEE es muy buena, ya que cuenta paso a paso lo que tienes que hacer para poner en marcha tu prototipo en formato de texto y de video.  Valoración: |
| Facilidad de uso | APIGEE es muy intuitiva debido a que con el tutorial puedes encontrar muy bien cada apartado de una forma visual y saber qué se realiza dentro de cada uno.  Valoración: |
| Implementación de seguridad y control de concurrencia | Para la implementación de seguridad y control de concurrencia también tenemos un tutorial, por lo que podemos decir que es fácil la Valoración: forma de hacerlo. |
| Profundidad del análisis | El análisis que contiene APIGEE permite hacer un análisis predeterminado de nuestras APIs y también permite obtener los resultados de un análisis personalizado a gusto del usuario.  Valoración: |
| Capacidad de prueba | Todas las APIs generadas en esta plataforma pueden ser testeadas para conocer su funcionalidad y claridad.  Valoración: |

## 6.2 Evaluación de los criterios en la implementación usando la tecnología WSO2

| **CRITERIO** | **EVALUACIÓN** |
| --- | --- |
| Dificultad del proceso de creación y publicación | Para poder publicar y crear una API en WSO2 es conveniente leer la documentación sobre ello con anterioridad pues sin saber puede resultar una tarea complicada.  Valoración:4/5 |
| Horas empleadas en la construcción del prototipo | Dado que es necesario documentarse e informarse con anterioridad sobre el proceso de creación, incluiremos dicho tiempo en la creación del prototipo por lo que es una duración aproximada de 4:30.  Valoración: 4:30 horas |
| Recursos necesarios | Puesto que estamos utilizando la versión de prueba en la nube de WSO2, únicamente es necesario el acceso a un navegador web.  Valoración: Conexión a internet |
| Configuración del servidor Proxy | La nube de WSO2 cuenta con un proxy propio ya configurado.  Valoración: 1 |
| Grado de personalización | WSO2 cuenta con la función de personalización de una API a gusto del creador, modificación de imagen, llamadas por segundo, seguridad y acceso a esta.  Valoración: 5 |
| Calidad del soporte | Resulta sencillo ponerse en contacto con el área de soporte de WSO2, además brinda una rápida atención y respuesta.  Valoración: 4 |
| Calidad de la documentación | La documentación que ofrece WSO2 es completa, pero en algunos momentos es complicada de entender, además de encontrarse por escrito también existen videotutoriales.  Valoración: 3.5 |
| Facilidad de uso | El trabajo con la nube de WSO2 no es especialmente fácil, pues es complicada de entender y de encontrar ciertas funciones, aun que por otro lado su interfaz de usuario si es amigable.  Valoración:3.5 |
| Implementación de seguridad y control de concurrencia | WSO2 permite integrar cierta seguridad en tu API a través de la creación de roles y controlar el acceso simultaneo a la API por parte del creador.  Valoración: 4 |
| Profundidad del análisis | La plataforma de WSO2 cuenta con una amplia disposición de herramientas de análisis para una API trabajando a través de una serie de parámetros. Sus resultados son de gran ayuda  Valoración: 5 |
| Capacidad de prueba | WSO2 permite realizar una prueba del código de una API ejecutando los métodos de esta.  Valoración: Si |

* Configuración del servidor proxy (WSO2): se valora como 1 dado que no existe dificultad alguna pues ya viene integrado en la plataforma.
* Dificultad del proceso de creación (WSO2): Entendemos 5 como una muy alta dificultad y 1 como muy baja dificultad.
* En los apartados que se mida la dificultad la entenderemos como se ha comentado en el punto anterior.

# 7. Comparación de la implementación de las tecnologías

Debe incluir al menos una tabla resumen, en sección de página horizontal, cruzando los criterios y los valores de cada tecnología. Con una columna de comentarios sobre la comparación

| **CRITERIOS** | **TECNOLOGÍA A** | **TECNOLOGÍA B** | **COMENTARIOS** |
| --- | --- | --- | --- |
| 1 |  |  |  |
| 2 |  |  |  |
| … |  |  |  |
| N |  |  |  |

# 8. Conclusiones

A partir de la información incluida en el apartado 7 y de la experiencia al realizar el trabajo, el grupo debe estar en condiciones de manifestar su opinión sobre la implementación del sistema utilizando ambas tecnologías, y debe plasmarla en este apartado, indicando las ventajas e inconvenientes más relevantes de utilizar una u otra tecnología para implementar el sistema.

---------------------------

(Hay que cumplir la estructura básica indicada de secciones. Pero si se desea se pueden añadir otras secciones como anexos. Por ejemplo, alguna encuesta de opinión realizada sobre las tecnologías, etc.)