

Plan de Trabajo

Descripción general del trabajo y objetivos

Se trata de un proyecto completo de integración y gestión de ciclo de vida de software cuyo principal objetivo es demostrar la capacidad de la gestión de API's (API Management) mediante el uso de herramientas diseñadas para este fin.

Para cumplir el objetivo se estudiarán diferentes soluciones de API Management que ofrecen algunas compañías líderes en el sector de las nuevas tecnologías como IBM, Axway o CA Technologies. Después de un estudio previo se escogerá la solución que más se ajuste a las necesidades del proyecto y que permita probar todo su potencial y el del resto de herramientas que participen en el sistema.

Durante el proyecto se desarrollarán, cuatro APIs de distintas características y dificultad, se probarán y se optimizarán. Además, estas APIs se expondrán para que cualquier aplicación externa pueda consumirlas mediante peticiones controladas al sistema.

La API de mayor peso consistirá en el diseño de una API REST basada en una serie de bancos donde poder simular alta de clientes, cuentas asociadas a los clientes, transacciones entre bancos y entre un mismo banco, ingreso y retirada de dinero de una cuenta, etc. La lógica del API se realizará sobre varios esquemas de Bases de Datos MySQL (un esquema por cada banco), y formará el backend. Esta BBDD estará conectada a la herramienta de API Management escogida, desde donde se implementará el resto de configuraciones adicionales, como pueden ser añadir seguridad, monitorización de rendimiento, control de parámetros, etc.

Así mismo, se utilizaran otras herramientas que ayudarán a probar el objetivo principal y cumplir otros objetivos. Algunas de estas herramientas pueden ser:

- Clientes REST como Postman o advanced REST Client para poder realizar pruebas mediante peticiones REST a nuestra API.
- Jmeter o SoapUI para realizar peticiones e inserciones masivas sobre nuestra API y así poder probar su eficiencia.
- Jenkins o Restman para cambiar la forma de autenticación del API de manera automática o para automatizar pruebas.

Otros objetivos que se pretenden conseguir durante el desarrollo de este proyecto son los siguientes:

- Realización de planes de pruebas y rendimiento completos para el Sistema.
- Dar a conocer las soluciones de API Management que permitirán cubrir los nuevos retos que demandan los modelos de negocio en la era digital.
- Presentar como la solución escogida acelera la creación de APIs, implicando en ello su gestión, securización y monitorización
- Demostrar como la solución escogida puede ayudar a empresas de cualquier volumen a expandir su negocio haciendo uso de APIs

Lista de tareas

A continuación se describe la lista de tareas previstas a realizar durante el trabajo:

- **Tarea 1: Estudio del arte.** En esta primera tarea se realizará un primer estudio sobre los conceptos más importantes del proyecto. Se responderá a las siguientes preguntas: ¿Qué es API?, ¿Qué es API Management?, ¿Cuál es la situación actual de las empresas respecto a este concepto?, y otros conceptos y aspectos generales. **(Fase de análisis)**
- **Tarea 2: Estudio previo de soluciones API Management.** Se estudiarán tres soluciones de API Management distintas y se escogerá la que más se ajuste a las características y dimensiones de este proyecto, justificando la elección. **(Fase de análisis)**
- **Tarea 3: Creación de APIs sencillas.** Se implementarán tres APIs sencillas directamente desde el sistema de API Management elegido utilizando exclusivamente las herramientas que éste ofrezca. Cada una de ellas explotará de diferente forma las posibilidades del sistema. **(Fase de construcción)**
- **Tarea 4: Pruebas de funcionamiento y rendimiento de APIs sencillas.** Se Harán pruebas de funcionamiento y rendimiento de las APIs creadas en la tarea 3 desde una aplicación externa al sistema como puede ser Jmeter o SoapUI. Se justificará la elección de la herramienta de pruebas utilizada para este caso. **(Fase de pruebas)**
- **Tarea 5: Diseño del API – Gestión de bancos.** Diseño funcional de la API principal del proyecto. Esta tarea consistirá en realizar los diagramas pertinentes, como pueden ser diagrama Entidad-Relación (E/R) o Modelo Relacional. **(Fase de diseño)**
- **Tarea 6: Creación de API completa – Gestión de Bancos.** Una vez diseñada la API, se desarrollará con una herramienta de Gestión de BBDD. Para facilitar esta tarea se utilizará un cliente de BBDD como puede ser Toad MySQL o MySQL Workbench (en caso de que la el tipo de Base de Datos escogida sea MySQL). Además, la BBDD estará conectada con la herramienta del sistema dedicada a la creación de APIs para poder completar su implementación. **(Fase de construcción)**
- **Tarea 7: Pruebas de funcionamiento y rendimiento de la API – Gestión de bancos.** Se harán pruebas de funcionamiento y rendimiento de la API creada en la tarea 6. Para ello se usarán herramientas externas al Sistema y la BBDD, como pueden ser JMeter o SoapUI. Esta tarea de pruebas se realizará en dos fases: la primera contra la Base de Datos, y la segunda contra la herramienta de creación de APIs del sistema. **(Fase de pruebas)**
- **Tarea 8: Exposición de API – Gestión de bancos en el Sistema.** Una vez expuesta y configurada correctamente en el Sistema, ya tendremos las 4 APIs disponibles. **(Fase de publicación)**
- **Tarea 9: Pruebas de rendimiento del Sistema.** Últimas pruebas de rendimiento de la API (Gestión de Bancos) expuesta en el Sistema. Esto supondría realizar una petición contra el Sistema -> Herramienta de creación -> BBDD (Backend). **(Fase de pruebas)**
- **Tarea 10: Securitización de APIs.** Mejorar la seguridad de las APIs añadiendo el protocolo HTTPS en las peticiones. Comprobar posible pérdida de rendimiento tras la securización. **(Fase de operación)**
- **Tarea 11: Exposición de APIs en el Portal.** Exposición de las APIs para que puedan ser consumidas por aplicaciones externas al sistema. Exposición con diferentes

mecanismos de autenticación, como pueden ser: API Key, Usuario/Contraseña, OAUTH 2.0, etc. **(Fase de publicación)**

- **Tarea 12: Integración continua.** Integración continua de los posibles ficheros de configuración parametrizables. Automatización de pruebas. **(Fase de operación)**
- **Tarea 13: Configuración del Portal.** Adaptación del Portal mediante un Sistema de Gestión de Contenidos (o CMS en sus siglas en inglés). Creación de aplicaciones y usuarios de prueba para que consuman las APIs expuestas en el Portal. **(Fase de socialización)**
- **Tarea 14: Redacción de la memoria.**
- **Tarea 15: Elaboración y preparación de la presentación .**

Por lo tanto con todo esto, podemos obtener una aproximación al ciclo de vida del proyecto:



Figura 1: Ciclo de Vida del proyecto

NOTA: Cada tarea está colorada con su etiqueta equivalente al Diagrama de Gantt (ver siguiente apartado) y a la fase del ciclo de vida a la que corresponde.

Diagrama de Gantt

El Diagrama de Gantt que se muestra al final de este apartado, se ha obtenido a partir de las tareas descritas en el apartado anterior. Como se puede observar la mayoría de tareas son de corta y media duración, con estos conseguimos que queden bien identificadas y definidas durante todo el proyecto.

La distribución de horas ha seguido en gran medida la propuesta de trabajo inicial. A continuación se muestra la tabla de tareas, su duración y el tipo de tarea según la fase del ciclo de vida a la que corresponda.

Nombre	Fecha de inicio	Fecha de fin	Tipo Tarea
• Tarea 1	06/02/2017	09/02/2017	Análisis
• Tarea 2	10/02/2017	17/02/2017	Análisis
• Tarea 3	20/02/2017	03/03/2017	Construcción
• Tarea 4	01/03/2017	08/03/2017	Pruebas
• Tarea 5	13/03/2017	16/03/2017	Diseño
• Tarea 6	20/03/2017	07/04/2017	Construcción
• Tarea 7	03/04/2017	13/04/2017	Pruebas
• Tarea 8	17/04/2017	19/04/2017	Publicación
• Tarea 9	24/04/2017	27/04/2017	Pruebas
• Tarea 10	01/05/2017	03/05/2017	Operación
• Tarea 11	08/05/2017	10/05/2017	Publicación
• Tarea 12	15/05/2017	17/05/2017	Operación
• Tarea 13	22/05/2017	26/05/2017	Socialización
• Tarea 14	13/03/2017	02/06/2017	Documentación
• Tarea 15	12/06/2017	16/06/2017	Presentación

Figura 2: Tabla de tareas Diagrama de Gantt

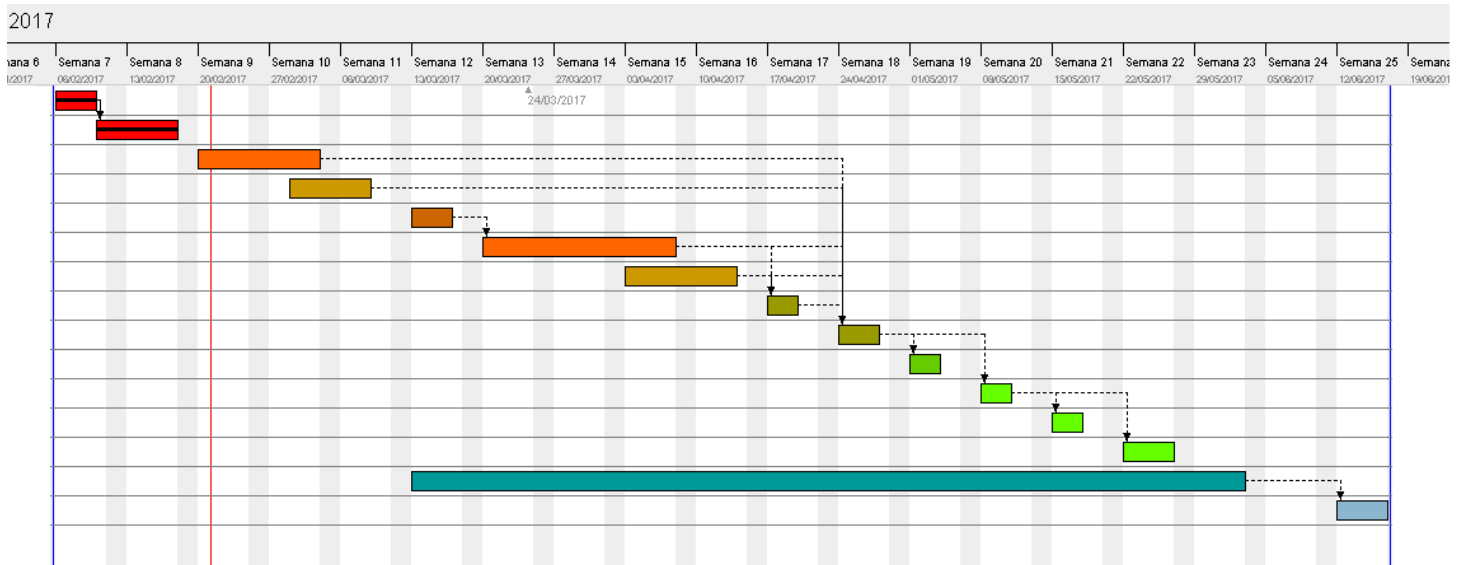


Figura 3: Diagrama de Gantt

Copia de la propuesta de trabajo escrito por el tutor

ID de la oferta	2175
Título del trabajo	Acelerando el desarrollo de API's con API Management
Resumen general del trabajo	Diseño y realización de una API Rest basada en una serie de bancos donde poder simular alta de clientes, cuentas asociadas a los clientes, transacciones entre bancos y entre un mismo banco, ingreso y retirada de dinero de una cuenta, etc.
Lista de objetivos concretos	Probar el potencial que tiene API Management para mejorar el desarrollo de API's. Esto supondría agilidad en la generación de nuevas API's, ahorrando cientos de líneas de código, por lo tanto estaríamos acelerando de forma considerable su desarrollo.
Desglose de la dedicación total del trabajo en horas	<ul style="list-style-type: none"> • Análisis: dentro de un proyecto software la fase de análisis es una de las más importantes, puesto que es donde se va a recoger toda la información necesaria para plantear el proyecto. En mi caso, voy a tener que estudiar la documentación de la solución que ofrece CA sobre API Management. Por lo tanto, he decidido dedicarle un 15% del total de horas, es decir 48h. • Diseño: montar toda la estructura del proyecto y diseñar el/los API's que voy a querer publicar, será una tarea laboriosa a la que le tendré que dedicar bastante tiempo para asegurarme poder avanzar con fluidez durante el resto del proyecto. Por lo tanto, he decidido dedicarle un 25% del total de horas, es decir 81h. • Construcción: Una vez he estudiado y comprendido cómo funcionan las herramientas que voy a utilizar, es hora de aplicar los conocimientos adquiridos durante las 2 fases anteriores y desarrollar las API's diseñadas. Dado que es una tarea laboriosa donde me pueden surgir algunas dudas o problemas, he decidido dedicarle el 25% del total de horas, es decir 81h. • Pruebas: es necesario que la fase de pruebas sea lo más precisa posible y que abarque todos los casos. Por lo tanto, dedicarle un buen número de horas es una opción inteligente. He decidido dedicarle el 20% del total de horas, es decir 65h. • Publicación/Operación: publicar las API's creadas en un entorno donde podamos gestionar algunas características (como puede ser monitorización, seguridad, etc) para su posterior operación, podría llevar dedicarle el 10% del total de horas, es decir 32h. • Socialización: por último, dar a conocer/ofrecer el API creada a través de una página web, donde además podamos probarla, nos llevará el 5% restante del proyecto, es decir 16h.
Conocimientos previos recomendados para hacer el trabajo	Programación de API.
Tutor 1	NELSON MEDINILLA MARTÍNEZ - nelson@fi.upm.es