



Tarea 2

Entrega

- **Fecha y hora:** Viernes 29 de septiembre del 2023, a las 22:00
- **Lugar:** Repositorio individual en la organización del curso en GitHub

Objetivos

- **Comprender** el funcionamiento de una API a partir de documentación.
- **Crear** un programa que pueda manejar requests y consumir una API
- **Producir** documentación efectiva y clara que permita el entendimiento del proceso realizado

Descripción

En esta tarea, tendrán la oportunidad de aprender a consumir una interfaz de programación de aplicaciones (API), lo cual es de suma utilidad en el mundo real, dado que en muchas ocasiones les evitará tener que “reinventar la rueda”. Es decir, podrán darle uso a herramientas que ya estén a tu disposición y, así, ahorrar tiempo y recursos.

Para esta tarea, se les proveerá una [API](#) creada por el equipo docente que contiene a un modelo *Product* con una serie de atributos asociados como nombre, descripción, precio, etc. Esta contiene seis *endpoints* con los que ustedes deberán interactuar. La información que reciban de la API deberá poder ser visualizada mediante un frontend en React.

En resumen, los objetivos que deberán tener en mente son: comprender qué es una API y por qué alguien querría darle uso, entender su documentación, consumirla y documentar el proceso.

Método GET

Para esta parte, tendrán que utilizar los diversos *endpoints* GET de la API dada. Se espera que puedan:

- Recibir todos los productos creados en el modelo.
- Obtener un producto en particular mediante su ID.
- Filtrar la obtención de un producto mediante dos parámetros: su categoría y su ID.

Métodos POST, PUT y DELETE

En esta sección, deberán utilizar los endpoints de la API tal que logren publicar un producto nuevo, además de modificar y eliminar un producto existente. Para ello, es sumamente importante que averigüen cuál es la diferencia entre los métodos PUT y POST en un contexto de uso en API RESTful.

Recordatorio: Es importante que tengan en consideración que, para métodos donde se está editando información sensible, no todos los usuarios deben poder tener acceso a ellos. El *token* que les dará acceso a todos los recursos reservados a utilizar es:

```
TOKEN = iicapit2
```

Interfaz gráfica

Para esta parte, tendrán que mostrar visualmente los resultados de las *requests* anteriormente descritas. Para ello, les proveeremos un código base en React que podrán encontrar en sus respectivos repositorios.

Documentación del proceso

Este ítem tiene como propósito:

- Explicar qué logra hacer su programa y también aquello que no lograron implementar.
- Indicar apropiadamente el proceso para levantar tu aplicación, incluyendo que comandos se deben correr.
- Dar a entender cómo funciona cada parte de su programa, es decir, cómo funciona cada *endpoint* implementado y qué consideraciones tuvieron que tener para que estos funcionaran correctamente.
- Indicar consideraciones particulares de su trabajo (en caso de haber).

Además, en sus respectivos repositorios encontrarán una serie de preguntas teóricas acerca de su tarea y del consumo de APIs en general. Deberán responder estas preguntas de forma más completa posible.

Para un mejor entendimiento sobre cómo escribir la documentación, se les hará entrega de una plantilla con la estructura que debe seguir, su descripción y las preguntas que deben responder.

Entregables

Cada entrega deberá incluir los siguientes archivos:

- `README.md` con la documentación de tu programa. Debes mencionar brevemente aquello que pudiste implementar y también lo que no. Debes indicar cómo levantar tu aplicación y responder las preguntas indicadas en el *template* que te facilitamos. Para finalizar, debes explicar claramente cómo funciona cada implementación de tu programa.
- Aplicación en React desarrollada en base al material entregado. Se deben considerar todos los directorios necesarios para poder levantar y correr el programa.

Rúbrica

A continuación, se describe el criterio de cada uno de los ítems de la rúbrica con la que se evaluará esta entrega. La nota se calcula al 50 % considerando un total de 23 puntos.

- **Interfaz gráfica [3 puntos]:** Se espera que la información obtenida a través del uso de la API dada se visualice de manera satisfactoria y completa en la interfaz gráfica. Esta debe ser realizada utilizando React.
- **Método GET [4 puntos]:** Se espera que se utilicen los *endpoints* necesarios para conseguir la información de todos los productos, de un producto en específico según su ID, y según su categoría y ID.
- **Método POST [2.5 puntos]:** Se espera que se haga uso del *endpoint* para publicar nuevos productos y que se haga una *request* funcional y bien construida.
- **Método PUT [3 puntos]:** Se espera que se haga uso del *endpoint* apropiado para actualizar de forma efectiva la información de un producto existente.
- **Método DELETE [2 puntos]:** Se espera que sean capaces de utilizar este método para eliminar un producto existente según su ID.
- **Escritura de documentación [8.5 puntos]:** Se espera que se explique de manera clara, ordenada y suficiente cada parte del programa. Se debe notar un claro entendimiento acerca de cómo se consume una API y de cómo se realizó (y cómo se corre) el código entregado. Además, se esperan respuestas claras y completas explicando las preguntas teóricas asociadas a la tarea.

Dudas

Cualquier duda que se presente acerca del enunciado debes consultarla en las [issues](#) del repositorio del curso. Recuerda que como equipo docente estaremos atentos para poder ayudarte :)

PD: *No se responderán dudas por correo.*

Notas

En caso que no puedan abrir el link de la API, pueden encontrar el código fuente en la organización del curso en el [repositorio en Github](#).