

Bootcamp Full Stack

Proyecto Integrador: Etapa 3

Consideraciones generales

En la última etapa del proyecto, se va a conectar el proyecto de frontend realizado en Javascript vanilla o ReactJS con un servicio de Backend desarrollado en NodeJS con Express y MongoDB NodeJS Driver (mongodb) como driver para interactuar con MongoDB.

Para ello, es necesario implementar el código del servidor siguiendo el patrón de arquitectura MVC (Modelo-Vista-Controlador), dividiéndolo en capas y creando las carpetas y archivos correspondientes para cada una de ellas.

Se debe configurar el servidor para permitir realizar un CRUD sobre el caso de uso de productos en la base de datos utilizando los métodos GET, POST, PUT y DELETE, según corresponda. También se pueden adicionar otros casos de uso como sucursales, promociones o medios de pago.

La base de datos a utilizar debe hacer uso del servicio de MongoDB Atlas. Se deberá disponer de un archivo para realizar la configuración del servidor, como el puerto de escucha, string de conexión hacia la base de datos, etc. vinculado con su archivo .env respectivo.

Se debe añadir una ruta para subir, mediante el método post, y haciendo uso del formato JSON, un array con los productos contenidos en el carrito del frontend. La información recibida en el backend deberá mostrarse en la consola del servidor. Optativamente, se puede almacenar dicha información en un archivo o base de datos. Al ser enviada la información del carrito, el mismo deberá vaciarse en el frontend.

Finalmente, se debe crear un repositorio en Github, al cual se subirá el código fuente completo del proyecto. Se deberá subir el proyecto a Glitch, quedando funcionando de forma pública en la nube. Se deberá entregar el código fuente total en un zip (sin los node_modules), y las URL de Glitch y Github.

Información adicional

Para cumplir con la implementación de un CRUD para el caso de uso de productos en la base de datos, utilizando los métodos GET, POST, PUT y DELETE, se deben definir las siguientes rutas en el servidor:

- Ruta GET '/api/productos': Esta ruta permitirá obtener todos los productos existentes en la base de datos.
- Ruta GET '/api/productos/:id': Esta ruta permitirá obtener un producto específico a través del ID proporcionado en la URL.
- Ruta POST '/api/productos': Esta ruta permitirá agregar un nuevo producto a la base de datos.
- Ruta PUT '/api/productos/:id': Esta ruta permitirá actualizar un producto existente en la base de datos, identificado a través del ID proporcionado en la URL.
- Ruta DELETE '/api/productos/:id': Esta ruta permitirá eliminar un producto específico de la base de datos, identificado a través del ID proporcionado en la URL.

Además se debe implementar una ruta POST '/api/carrito' para almacenar el pedido que ha sido validado por el frontend.

Funcionalidades Extras.

Se solicita agregar la funcionalidad de pago mediante la implementación de Mercado Pago en el proyecto.

Para ello, se deberá realizar lo siguiente:

- Se deberá generar una cuenta en Mercado Pago y obtener las credenciales necesarias para realizar la integración.
- Se deberá añadir un botón de pago en la página del producto, y llevar a cabo el checkout mediante la API de Mercado Pago.
- Se deberá configurar la página de éxito y la de cancelación, según corresponda, para notificar al cliente sobre el resultado de la transacción.

La implementación de Mercado Pago es opcional y no afecta la entrega del proyecto en caso de no ser realizada.