

WChallenge - Cryptocurrencies Monitor

Bases del ejercicio

El objetivo es desarrollar una API para consumir un servicio externo y utilizar su información. Para realizar este desarrollo se tiene que utilizar NodeJS y el código debe estar alojado en un repositorio en Github o Gitlab.

Se espera que el `README.md` contenga una breve explicación de como correr el proyecto.

Ejercicio

Desarrollar una API con Express que sea un wrapper de CoinGecko (<https://www.coingecko.com/en/api>) junto con su propio modelo de datos persistido en una base de datos, los requerimientos deseados son los siguientes:

- Se debe poder crear usuarios, con los siguientes parámetros:
 - Nombre
 - Apellido
 - Username (único)
 - Contraseña (al menos 8 caracteres, alfanumérica)
 - Moneda preferida (puede ser Euro, Dólar o Peso Argentino)
- Se debe implementar autenticación mediante JWT:
 - El token se debe obtener mediante un endpoint de login (pasando un usuario y contraseña).
 - El token debe tener un tiempo de expiración.
 - Este token debe generarse desde la aplicación sin utilizar servicios externos (puede utilizarse packages que faciliten la creación del mismo pero no servicios como Auth0).
- Se debe poder listar todas las criptomonedas disponibles
 - Los datos a incluir por cada criptomoneda deben ser:
 - Simbolo.
 - Precio (en la moneda preferida del usuario).
 - Nombre.
 - Imagen.
 - Fecha última actualización.
- Se debe permitir agregar criptomonedas al usuario para hacerle seguimiento
 - Sólo un usuario puede agregarse criptomonedas a sí mismo.
 - La relación usuario-criptomoneda debe ser única.
- Se debe poder obtener el top N de criptomonedas de un usuario (comparando la cotización en la moneda preferida del usuario que realiza la consulta):

- Un usuario solo puede obtener su top N de criptomonedas.
- Debe encontrarse ordenada por cotización (ascendente/descendente, descendente por default).
- N puede ser menor o igual a 25.
- Los datos a incluir por cada criptomoneda deben ser:
 - Simbolo.
 - Precio en Pesos Argentinos.
 - Precio en Dólares.
 - Precio en Euros.
 - Nombre.
 - Imagen.
 - Fecha última actualización.

El proyecto se evaluará en términos de

- Buenas prácticas de diseño en términos de apropiada separación de responsabilidades entre los componentes de la solución.
- Nivel de desacoplamiento de los componentes de la solución.
- Uso de las capacidades del framework de express.
- Claridad y legibilidad del código.
- Buenas prácticas en la definición e implementación de APIs REST.
- Manejo y control de errores.
- Definir algún tipo de documentación de los endpoints definidos.
- Calidad y cobertura de testing.
- Justificación o aclaración de las decisiones tomadas.