

Áreas a evaluar:

- Conocimiento sobre APIs con estándar REST
- Configuración de ambientes de desarrollo y deployment (Virtual Envs, Containers, etc...)
- Escalabilidad del código (que tan fácil es leerlo, mantenerlo, y probarlo).
- Resolución de requerimientos ambiguos (solución para temas con más de una respuesta correcta)

Requerimientos Técnicos

- Que el proyecto se pueda desplegar de manera local.
- Estricto uso del estándar REST.
- Uso de NestJS (backend)
- Uso de React (frontend)
- Pruebas unitarias / integración (Tests para los WS deseable).
- Obtener credenciales de autenticación para consulta de todos los servicios que dependan del Usuario. Todos los servicios relacionados a un Usuario deben estar protegidos por algún estándar de Autenticación

Requerimiento de Negocio:

Se tiene que crear una API que será consumida por un conjunto de clientes (móviles y web). El API será consumida por una base de 1000 usuarios con uso constante para a revisión de transacciones. Tiene que contener los siguientes servicios, almacenando los datos necesarios para su correcto funcionamiento.

- Lista de transacciones de pagos
- Servicio de retorne montos agrupados por categorías de gastos.
- Servicio que determine qué tan sanas son sus finanzas (análisis de ingresos y egresos)
- Servicio que sume total de ingresos y egreso de las diferentes cuentas del usuario.
- Dashboard donde se muestren graficas (Libre mínimo 3 graficas)

La Información para ser consumida la podrás obtener de <https://belvo.com> en su ambiente sandbox.

Datos a evaluar

1. Manejo de errores.
2. Validación de datos de entrada y/o salida. (congruencia de datos)
3. Manejo de y optimización de consultas.
4. Manejo de transacciones.
5. Estructura de proyecto. (buenas prácticas)
6. Conocimiento sobre APIs con estándar REST
7. Conocimiento sobre bases de datos de alto consumo y disponibilidad
8. Configuración de ambientes de desarrollo y deployment (Virtual Envs, Containers, etc...)
9. Escalabilidad del código (que tan fácil es leerlo, mantenerlo, y probarlo).
10. Resolución de requerimientos ambiguos (solución para temas con más de una respuesta correcta)