

Relevamiento de API REST

Consignas:

- 1) Por cada pantalla de su Frontend, identificar y listar los endpoints necesarios
 - 2) Por cada endpoint, definir:
 - a) Método HTTP, URL, reglas de negocio
 - b) Identificar qué endpoints necesitan autenticación
 - c) Definir interfaces TypeScript para request/response
-

Pantalla: Perfil

- API método http get del info del perfil (foto, nombre, descripción)
GET /users/123
No necesita autenticación
- API método http patch para actualizar la información parcialmente
PATCH /users/123
Necesita autenticación
Reglas de negocio:
 - Solo el usuario autenticado puede actualizar la información
 - Maximo se permiten dos actualizaciones por mes.
- API método http get para traer los proyectos de un usuario postulante
GET /users/123/proyectos
No necesita autenticación
- API método http post para agregar proyectos a tu perfil (siendo POSTULANTE)
POST /proyectos
Necesita autenticación
Reglas de negocio:
 - Solo el usuario autenticado puede agregar proyectos a su perfil
 - Los campos no pueden estar vacíos
- API método http delete para eliminar proyectos de tu perfil
DELETE /proyectos/123
Necesita autenticación
Reglas de negocio:
 - Solo el usuario autenticado puede eliminar proyectos de su perfil y éste debe ser postulante

- API método http get para traer usuarios y mostrarlos como sugerencia.
GET /users
Necesita autenticación

Interfaz proyecto:

```
type Proyecto = {
  nombre: string
  descripción: string
  tecnologías-usadas: string
  usuario: Usuario
}
```

Pantalla: Inicio sesión

- API método HTTP POST para crear usuario.
POST /users
Reglas de negocio:
 - No puede haber dos usuarios con el mismo nombre de usuario
- API método HTTP PATCH para cambiar contraseña del usuario. PATCH /users/123
Necesita autenticación
Regla de negocio:
 - La contraseña nueva no puede ser igual que la antigua
 - La contraseña nueva no puede ser igual a una que se haya usado antes
 - La contraseña debe incluir por lo menos 1 número y un carácter especial.

Interfaz tipoUsuario:

```
type TipoUsuario = 'postulante' | 'empleador' | 'admin'
```

Interfaz Usuario:

```
type Usuario = {
  readonly id: number,
  nombre: string,
  contraseña: string,
  tipo: TipoUsuario,
  descripción: string,
  fechaNacimiento?: date
  fechaCreacion?: date
  fotoperfil?: '${string}@${string}'
  logo?: '${string}@${string}'
  proyectos?: Proyecto[ ]
  ofertas?: Ofertas [ ]
```

}

Pantalla: Home

- API método HTTP GET para filtrar por empleados.
GET /users
- API método HTTP POST para crear una nueva oferta de empleo.
POST /ofertas
Necesita Autenticación
Reglas de negocio:
 - Los campos no pueden estar vacíos.
 - Solo el usuario autenticado como empleador puede crear una oferta de empleo
- API método HTTP DELETE para eliminar una oferta de empleo.
DELETE /ofertas
Necesita Autenticación
Reglas de negocio:
 - Solo el usuario autenticado como empleador puede eliminar una oferta de empleo
- API método HTTP PUT para modificar una oferta de empleo.
PUT /ofertas
Necesita Autenticación
Reglas de negocio:
 - Los campos no pueden estar vacíos.
 - Solo el usuario autenticado como empleador puede modificar una oferta de empleo
- API método HTTP PATCH para registrar un formulario de postulación a un empleo.
PATCH /ofertas/1239
Necesita Autenticación
Reglas de negocio:
 - Solo el usuario autenticado como postulante puede postularse a una oferta de empleo
 - Un postulante podrá postularse a máximo 10 ofertas por día
- API método HTTP GET para traer todos los formularios de empleados candidatos y mostrarlos en el carousel.
GET /ofertas/formulario/
- API método HTTP GET para traer las ofertas y mostrarlas en el carousel
GET ofertas/
- API método HTTP GET para filtrar las ofertas
GET ofertas/

Interfaz oferta:

```
type Oferta = {  
  categoría: string  
  ubicacion: string  
  sueldo?: float  
  modalidad: TipoModalidad  
  horario: date [ ]  
  formularios: Formulario[ ]  
  usuarioCreador: Usuario  
}
```

Interfaz modalidad:

```
type TipoModalidad = 'presencial' | 'remoto' | 'hibrido'
```

```
type Formulario={  
  nombre:string  
  apellido:string  
  localidad:Localidad  
  pais:Pais  
  genero:Genero  
  descripcion:string  
  archivo:string  
  usuarioPostulante:Usuario  
  oferta:Oferta  
}
```