



PRUEBA TECNICA PARA DESARROLLADOR ABANK

Introducción.

El propósito de este desafío es evaluar la capacidad de un desarrollador de backend para interpretar e implementar los requerimientos planteados. En este documento, se le presenta un escenario que está destinado a ayudar a mostrar sus talentos de desarrollo de software y despertar el pensamiento creativo y la resolución de problemas.

¿Qué se busca evaluar?

1. Un API Rest usando NET 8 (Requerido)
2. Una copia del código en GitHub o Gitlab. (Requerido)
3. Documentación del proyecto en markdown (Opcional)
4. Adjuntar al repositorio las indicaciones para correr y probar su proyecto. (Requerido)
5. Docker (Opcional)
6. JWT (Requerido)
7. Test unitarios (Opcional)
8. Despliegue de aplicación (Opcional)
9. Base de datos PostgreSQL, MySQL, SQLite, MongoDB (PostgreSQL preferiblemente.)
10. Para la persistencia de datos usar dapper.

Se le calificará

1. Calidad del Código:
 - Estructura y Organización: El código debe estar bien estructurado y organizado.
 - Buenas Prácticas: Uso de convenciones de codificación, comentarios claros y significativos, y adherencia a principios como DRY (Don't Repeat Yourself) y SOLID.
2. Eficiencia de los Algoritmos:
 - Complejidad Temporal y Espacial: Evaluar la eficiencia del código en términos de tiempo de ejecución y uso de memoria.
 - Optimización: Capacidad para optimizar el código y mejorar su rendimiento.
3. Creatividad y Resolución de Problemas:
 - Soluciones Innovadoras: Capacidad para encontrar soluciones creativas y efectivas a los problemas planteados.
 - Adaptabilidad: Flexibilidad para ajustar y mejorar las soluciones según los requisitos cambiantes.
4. Familiaridad con Frameworks y Herramientas:
 - Conocimiento Técnico: Dominio de frameworks y herramientas relevantes para el desarrollo backend.



- Integración y Uso: Habilidad para integrar y utilizar estas herramientas de manera efectiva en el proyecto.

5. Pruebas y Depuración:

- Testing: Capacidad para escribir y ejecutar pruebas unitarias y de integración.
- Depuración: Habilidad para identificar y corregir errores en el código.

6. Documentación:

- Claridad y Detalle: La documentación debe ser clara, detallada y fácil de seguir.
- Mantenimiento: Facilitar el mantenimiento y la escalabilidad del código a través de una buena documentación.

ENTREGABLES

Se debe enviar el link público del repositorio a los siguientes correos electrónicos:

rafael.cuchilla@abank.com.sv y david.martinez@abank.com.sv adjuntando las indicaciones para levantar y probar las aplicaciones, tienes 4h para realizar la prueba a partir desde la recepción del correo.

La solución de la prueba se debe subir a GitHub o Gitlab siguiendo la siguiente estructura de carpetas:



En la base de datos se deben colocar todos los scripts de creación de objetos y los insert a las tablas, y en la carpeta Backend, colocar el proyecto de código.

Ejercicio.

Los Data Type y Length marcados con * deben ser definidos de acuerdo con el criterio y experiencia del candidato.

Nuestro equipo tiene la tarea de generar una API para gestionar los usuarios y el acceso a la aplicación. Se define para la capa de datos la siguiente estructura:

Name	Data Type	Length/Precision	Not Null
id	int	*	Yes
nombres	varchar	*	Yes
apellidos	varchar	*	Yes
fechanacimiento	*	*	Yes
direccion	*	*	Yes
password	*	120	yes



telefono	varchar	*	yes
email	varchar	*	yes
fechacreacion	*	*	Yes
fechamodificacion	*	*	NO

Teniendo en cuenta la tabla anterior usted debe desarrollar los siguientes endpoints:

Algunos endpoints marcados con **(security)** deben estar protegidos por un token de acceso.

End-point: Login

POST /api/v1/auth/login

Body

```
{
  "telefono": "prueba",
  "password": "prueba"
}
```

Response 200 OK

```
{
  "user": {
    "id": 53,
    "nombres": "prueba",
    "apellidos": "prueba",
    "session_active": true,
    "fechanacimiento": "1988-03-17",
    "email": "prueba@yopmail.com",
    "telefono": "020202",
    "password": "$2b$12$HGSW3f6EKIE8zf7JP.gGUeoagaYNnC.6NQA6k8zHQ0.rDBph8Wsq",
    "address": "cl 50 b sur"
  },
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJjb2JpbGVfcGhvbmUiOiIwMjAyMDIi",
  "token_type": "bearer"
}
```

End-point: GetUsers

GET /api/v1/users

Response 200 OK

```
[
  {
    "id": 1,
    "nombres": "test",
    "apellidos": "test",
    "fechanacimiento": "1996-05-10",
    "telefono": "213212656",
    "email": "test@gmail.com",
    "direccion": null
  },
  {
    "id": 1,
    "nombres": "test",
    "apellidos": "test",
    "fechanacimiento": "1996-05-10",
    "telefono": "213212656",
    "email": "test@gmail.com",
    "direccion": null
  }
]
```



End-point: GetUser

(security) GET /api/v1/users/{id_user}

Response 200 OK

```
{
  "id": 1,
  "nombres": "test",
  "apellidos": "test",
  "fechanacimiento": "1996-05-10",
  "telefono": "213212656",
  "email": "test@gmail.com",
  "direccion": null
}
```

End-point: CreateUser

(security) POST /api/v1/users

Body

```
{
  "nombres": "test",
  "apellidos": "test",
  "fechanacimiento": "1996-05-10",
  "telefono": "213212656",
  "email": "test@gmail.com",
  "password": "123",
  "direccion": "direccion"
}
```

Response 201 Created

```
{
  "id": 1,
  "nombres": "test",
  "apellidos": "test",
  "fechanacimiento": "1996-05-10",
  "telefono": "213212656",
  "email": "test@gmail.com",
  "password": "123",
  "direccion": "direccion"
}
```

End-point: UpdateUser

(security) PUT /api/v1/users/{id_user}

Body



```
{
  "nombres": "test",
  "apellidos": "test",
  "fechanacimiento": "1996-05-10",
  "telefono": "213212656",
  "email": "test@gmail.com",
  "password": "123",
  "direccion": "direccion"
}
```

Response 200 OK

```
{
  "id": 1,
  "nombres": "test",
  "apellidos": "test",
  "fechanacimiento": "1996-05-10",
  "telefono": "213212656",
  "email": "test@gmail.com",
  "password": "123",
  "direccion": "direccion"
}
```

End-point: DeleteUser

(security) DELETE /api/v1/users/{id_user}

Response 200 OK

```
{
  "id": 1,
  "nombres": "test",
  "apellidos": "test",
  "fechanacimiento": "1996-05-10",
  "telefono": "213212656",
  "email": "test@gmail.com",
  "password": "123",
  "direccion": "direccion"
}
```

¡¡MUCHA SUERTE!!