

Manual de Usuario del Sistema de Autenticación en Dos Pasos (API)

1. Introducción

Esta API proporciona los endpoints necesarios para simular un proceso de autenticación en dos pasos. La API está diseñada para manejar la creación de usuarios, la autenticación inicial y la confirmación de registro a través de un token enviado por correo electrónico. Una vez autenticado, el usuario podrá acceder a los endpoints relacionados con la gestión de productos.

2. Requisitos Previos

- Tener un entorno de desarrollo configurado con .NET y acceso a MySQL para la base de datos.
 - Contar con Github instalado.
 - Contar con un correo electrónico válido para el proceso de autenticación.
 - Conexión a internet para ejecutar el proyecto y recibir correos electrónicos.
-

3. Configuración Inicial

1. Clonacion del Repositorio:

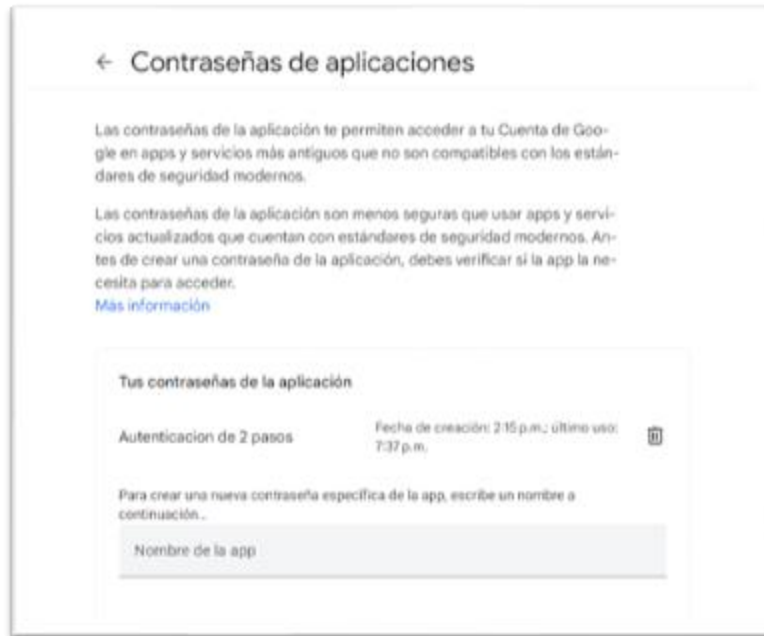
- Clona el repositorio con el comando:
git clone https://github.com/MiguelSamayoa/2FA_API.git

2. Creación de la Base de Datos:

- Utilice el script SQL proporcionado para crear las tablas necesarias en la base de datos.
- Asegúrese de que la base de datos esté funcionando y accesible desde su entorno local o servidor.
- Agregue la cadena de conexión en las variables de entorno dentro del archivo .env.

3. Configuración de cuenta de correo

- Debe disponer de un correo electrónico útil, desde este correo se enviarán los mensajes por lo que no puede ser el mismo que se utilice para el registro de usuarios
- **Esta configuración está pensada para el uso de los servicios de Google**
- Debe ingresar a la configuración de su cuenta y en el apartado de seguridad verificar que esté activada la verificación en 2 pasos (Activarla en caso que no esté activa)
- Desde la barra de búsqueda, buscar y acceder a la opción “Contraseñas de Aplicación”
- En el apartado de “Contraseñas de Aplicación” ingresar un nombre a la app y crear la contraseña



- Al crear la contraseña, esta se mostrara en un cuadro de dialogo en el que tendrá que copiar y asegurarse de guardar esa contraseña, no podrá volver a acceder a esta información



- Ingrese a la carpeta del proyecto, ubicar el archivo "EmailService.cs" en la carpeta "Service", en este proyecto debe reemplazar el campo "TuEmail" por el email que se usó para la configuración y el campo "TuContraseña" por la contraseña de aplicación generada anteriormente

```
Application1 | WebApplication1.Services.EmailService | SendEmail
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53

try
{
    var newEmail = new MimeMessage();
    newEmail.From.Add(MailboxAddress.Parse("TuEmail"));
    newEmail.To.Add(MailboxAddress.Parse(email));
    newEmail.Subject = subject;

    // Crear el cuerpo del correo con formato HTML
    newEmail.Body = new TextPart("html")
    {
        Text = message
    };

    using var smtp = new SmtpClient();

    // Ignorar la validación del certificado en entornos de desarrollo
    smtp.ServerCertificateValidationCallback = (sender, certificate, chain, sslPolicyErrors) => true;

    smtp.Connect(
        "smtp.gmail.com",
        Convert.ToInt32(587),
        SecureSocketOptions.StartTls
    );

    smtp.Authenticate(
        "TuEmail",
        "TuContraseña"
    );
}
```

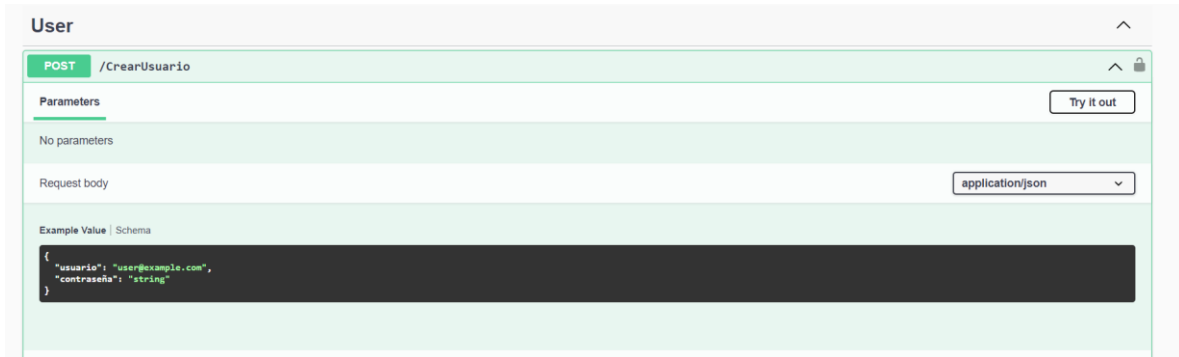
4. Ejecución del Proyecto:

- Ejecute el proyecto desde su entorno de desarrollo. Asegúrese de que la API esté corriendo correctamente y accesible desde la URL configurada.
-

4. Proceso de Uso de la API

1. Registro de Usuario:

- El primer paso es crear una cuenta de usuario enviando los datos necesarios a través del endpoint de creación de usuario.
- Encuentra la sección “User” y despliega la opción “CrearUsuario”



User

POST /CrearUsuario

Parameters

No parameters

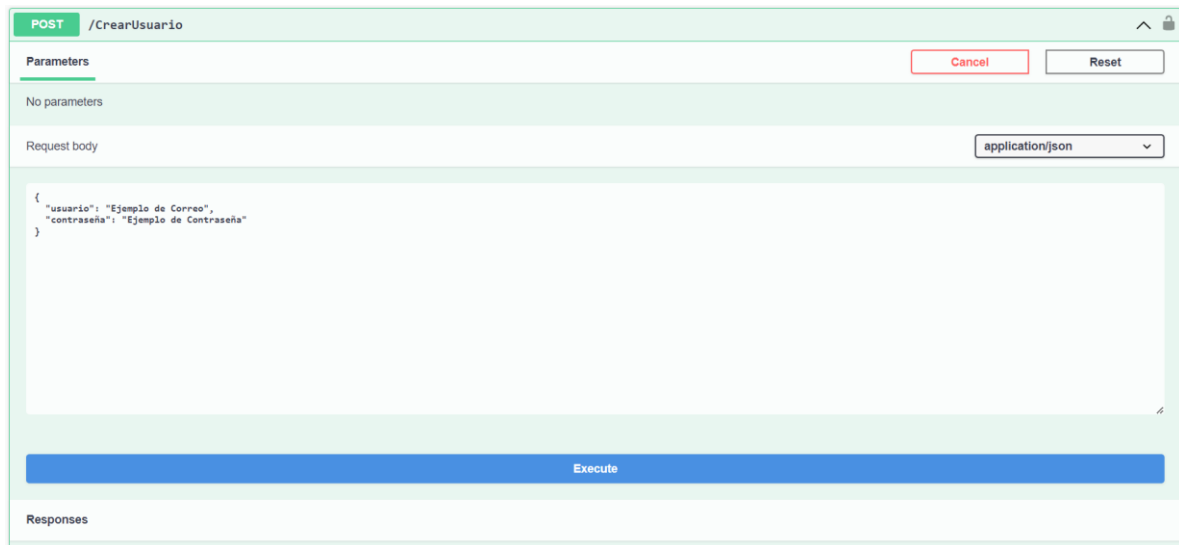
Request body

application/json

Example Value | Schema

```
{  "usuario": "user@example.com",  "contraseña": "string"}
```

- Al registrar seleccionar “Try it out” podrá ingresar su correo y contraseña. Debe tomar en cuenta que el correo debe existir, ya que a este se enviara la confirmación. Luego de ingresar sus datos deberá presionar “Execute”. Debería recibir un código 200 (Ok) como respuesta



POST /CrearUsuario

Parameters

No parameters

Request body

application/json

```
{  "usuario": "Ejemplo de Correo",  "contraseña": "Ejemplo de Contraseña"}
```

Execute

Responses

2. Autenticación en Dos Pasos:

- Después de registrarse, el usuario debe iniciar sesión enviando su correo y contraseña. Si los datos son correctos, la API enviará un correo con un enlace de confirmación.

- En el apartado de Login deberá de ingresar los datos con los que se registró en el sistema y enviar la solicitud.

The screenshot shows a REST client interface with a POST request to the endpoint `/Login`. The request body is a JSON object containing the following data:

```
{  "usuario": "Ejemplo de Correo",  "contraseña": "Ejemplo de Contraseña"}
```

The interface includes a 'Parameters' section with 'No parameters', a 'Request body' section with a dropdown menu set to 'application/json', and an 'Execute' button at the bottom.

- Luego de lo anterior tendrá que revisar su correo, en el que llegará el correo electrónico. Seleccionara la opción de Confirmar correo.

The screenshot shows a welcome email template with the following content:

Bienvenido a Nuestro Servicio

Hola [Nombre],

Gracias por registrarte en nuestro servicio. Estamos emocionados de tenerte con nosotros.

Por favor, confirma tu dirección de correo electrónico haciendo clic en el botón de abajo:

[Confirmar Correo](#)

Si no solicitaste este registro, puedes ignorar este mensaje.

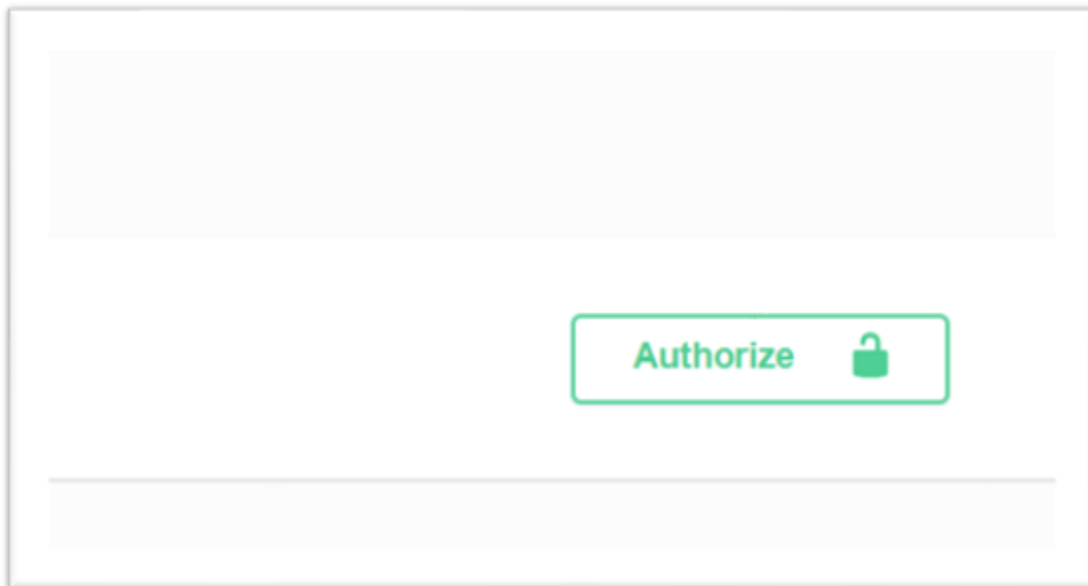
¡Gracias y que tengas un excelente día!

- El botón para confirmar correo enviara la solicitud de confirmación y se redirigirá a otra pestaña, en la que recibirá el json con el jwt correspondiente a su sesión. Debe copiar el contenido de token (sin las comillas)

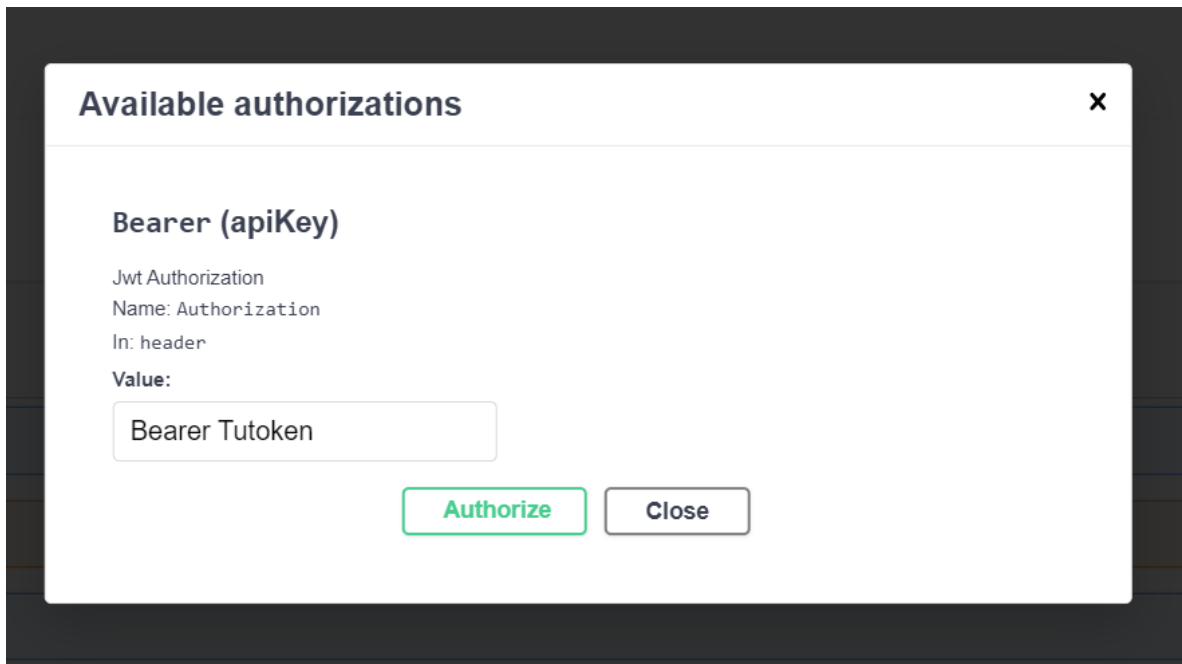
```
{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ3ZmVzZXI0IiwiaWF0IjEwNTU0NzMxMjQyOTU0NDZlbnQ.ahCSsgXQ3ToFa3q_HyNxslx8KJ1ovDW7BaHnMw4U9eY",
  "creation": "2024-08-24T21:05:31.0066615-06:00",
  "usuario": {
    "id": 4,
    "userTag": "mikysama1234@gmail.com"
  }
}
```

3. Acceso a Endpoints de Productos:

- Para acceder a los endpoints relacionados con productos, es necesario incluir el token JWT en el encabezado de las solicitudes. Puede utilizar la opción de autorización integrada en Swagger para añadir el token de manera sencilla y realizar las peticiones a los endpoints protegidos. Para esto puede localizar rápidamente la opción de autenticación proporcionada por swagger:



- Luego ingresar la palabra “Bearer”, seguida del token proporcionado:



4. Uso de Swagger para Interactuar con la API

- Swagger proporciona una interfaz visual para probar los endpoints de la API.
 - Una vez autenticado y con el token JWT, puede usar las opciones de autorización en Swagger para añadir el token y probar los endpoints relacionados con productos.
-

6. Consideraciones Finales

- El proyecto está orientado a demostrar el proceso de autenticación en dos pasos y la gestión básica de productos.
- Se recomienda familiarizarse con el uso de Swagger para la interacción con la API, ya que facilita el envío de peticiones con encabezados y tokens.

Este manual proporciona las bases para utilizar y entender el funcionamiento de la API, permitiendo realizar pruebas y validar el proceso de autenticación.