# Micro servicio de autenticación con .NET Core Web api siguiendo los principios de Clean Arquitecture y DDD.

## Visión global

El proyecto se basa en desarrollar un micro servicio de registro de usuario, confirmación de email válido, posterior inicio de sesión y recuperación de contraseña si la ha olvidado utilizando .NET Core Web API, JSON Web Token (JWT), un servicio de correo electrónico. Todo ello siguiendo el enfoque de arquitectura limpia.

## Sprints

Sprint 1: Configuración del proyecto

* Crear el proyecto en .NET Core Web API utilizando el patrón de arquitectura limpia
* Configurar la conexión a la base de datos
* Crear las entidades de Usuario y EmailConfirmation en el dominio

Sprint 2: Registro de usuario

* Crear el endpoint para el registro de usuario
* Validar los datos recibidos en el endpoint (nombre, email, contraseña, etc.)
* Encriptar la contraseña y guardar el usuario en la base de datos
* Generar un token JWT para el usuario registrado

Sprint 3: Confirmación de email

* Crear el endpoint para la confirmación de email
* Enviar un correo electrónico de confirmación al usuario registrado
* Validar el token recibido por correo electrónico para confirmar la dirección de correo electrónico del usuario
* Actualizar el estado del usuario en la base de datos para indicar que su correo electrónico ha sido confirmado

Sprint 4: Inicio de sesión

* Crear el endpoint para el inicio de sesión
* Validar los datos recibidos en el endpoint (email, contraseña)
* Verificar si el usuario está registrado y si su correo electrónico ha sido confirmado
* Generar un nuevo token JWT para el usuario autenticado

Sprint 5: Recuperación de contraseña

* Crear el endpoint para la recuperación de contraseña
* Validar el correo electrónico del usuario
* Generar un token JWT para permitir al usuario cambiar su contraseña
* Enviar un correo electrónico al usuario con un enlace para cambiar su contraseña

Sprint 6: Cambio de contraseña

* Crear el endpoint para el cambio de contraseña
* Validar el token recibido en el correo electrónico del usuario
* Cambiar la contraseña del usuario en la base de datos

## SPRINTS DETALLADOS

### Sprint 1

* Definición de las entidades de dominio Usuario y EmailConfirmation:

Usuario:

jsonCopy code

{ "id": "string", "nombre": "string", "email": "string", "contrasena": "string", "correoConfirmado": "boolean" }

* id: identificador único del usuario
* nombre: nombre completo del usuario
* email: dirección de correo electrónico del usuario
* contrasena: contraseña del usuario
* correoConfirmado: indica si la dirección de correo electrónico del usuario ha sido confirmada

EmailConfirmation:

jsonCopy code

{ "id": "string", "codigo": "string", "usuarioId": "string" }

* id: identificador único de la confirmación por correo electrónico
* codigo: código generado para la confirmación por correo electrónico
* usuarioId: identificador único del usuario asociado a la confirmación por correo electrónico