

Actividad grupal : Programador web avanzado

1. Para poder desarrollar esta actividad es importante que tengas un grupo de al menos 3 personas.
2. La actividad incluye los siguientes conceptos
 - a. Creación de rutas
 - b. Armado de una base de datos (básica)
 - c. Generación de JWT
 - d. Envío de correo

La actividad consiste en emular un registro validando el mismo a través de un JWT que expira en una hora.

Proceso →

1. El usuario se registra dentro de la aplicación con (nombre, apellido, mail, usuario y password)
2. Se genera un uuid y un JWT que contenga el uuid y un tiempo de vida de 1 hora
3. Se registra al usuario dentro de una tabla "usuarios", se ingresa el token en un campo tokenCorreo y el campo uidCorreo
4. Se envía un correo al usuario que tenga un link de confirmación con la siguiente estructura : URL:PUERTO/users/confirm?token='JWT'&uid='UID'
5. Una vez que el usuario hace click dentro de ese link, debe existir una ruta que valide esa información (decodificar el token a través de una clave pública) y verificar que el uid del token matchee con el uid de la URL. En caso de ser válido, actualizar un campo habilitado en la tabla y responder un mensaje de éxito.

Hints :

- a. Recordá que para crear un token necesitas dos claves (una privada para generarlo y una pública para descifrarlo) → <https://travistidwell.com/jsencrypt/demo/>
- b. La estructura que tiene que tener el token debe ser algo así : payload : {uid : 'UUID GENERADO DINÁMICAMENTE' }
- c. Los tokenOptions deben tener las propiedades expiresIn : '1h' y algorithm : 'RS256'
- d. Para enviar el correo recuerda que debes autenticarte con usuario y password y esta debe estar guardado en variables de entorno y se debe acceder a través de process.env.VARIABLE

Los modulos a instalar para el proyecto :

```
npm i dotenv  
npm i nodemailer  
npm i jsonwebtoken  
npm i sha1 (para guardar el password)
```

El modelo de datos a crear

Nombre base de datos : ejercicio
Nombre tabla : usuarios
Campos tabla :

id (PK A_I)
nombre
apellido
correo
password
tokenCorreo
uidCorreo
habilitado → default 0
ts_create
ts_update

La actividad debe ser entregada en un plazo de 3 horas y debe estar subida a un repositorio con la siguiente estructura :

master → solo la versión productiva y testeada
registerFeature → rama creada a partir de master (se realiza todo el desarrollo)

Recordá que para crear esta estructura tenes que inicializar el proyecto con git init, crear la rama y moverte (git checkout -b registerFeature)