

# **Documentación Proyecto**

## **MENSAJERIA ELECTRONICA**

**NRC: 2097**

**Equipo No. 08**

**Integrantes:**

*Antonio Gil*

*Franco Pantoja*

*Santiago Espitia*

*Steven Parra*

*Wendy Sanabria*

## Métodos controladores del Back-End (Sprint 3)

Nombre	Ruta	Métodos HTTP permitidos	Lógica algorítmica
validarUsuario	/validarUsuario	<u>GET/POST</u>	
registrarUsuario	/registrarUsuario	GET/POST	
enviarMail	/enviarMail	GET/POST	
actualizacionPassword	/actualizacionPassword	GET/POST	
activarUsuario	/activarUsuario	GET/POST	
historialEnviados	/historialEnviados	GET/POST	
historialRecibidos	/historialRecibidos	GET/POST	

- Nombre método controlador 1
  - Ruta: /validarUsuario
  - Métodos HTTP permitidos: GET/POST
  - Lógica algorítmica:

*Valida el usuario a través del formulario y el tab correcto este permite verificar los registros correspondientes*

- Nombre método controlador 2
  - Ruta: /registrarUsuario
  - Métodos HTTP permitidos: GET/POST
  - Lógica algorítmica:

- Nombre método /enviarMail
  - Ruta:/enviarMail
  - Métodos HTTP permitidos: GET/POST
  - Lógica algorítmica:

- /actualizacionPassword
  - Ruta: /actualizacionPassword
  - Métodos HTTP permitidos: GET/POST
  - Lógica algorítmica:

*Inserte una breve descripción de la lógica algorítmica del método controlador.*

- Nombre método /activarUsuario
  - Ruta: /activarUsuario
  - Métodos HTTP permitidos: **GET/POST**
  - Lógica algorítmica:  
*Inserte una breve descripción de la lógica algorítmica del método controlador.*
- Nombre método /historialEnviados
  - Ruta: /historialEnviados
  - Métodos HTTP permitidos: **GET/POST**
  - Lógica algorítmica:  
*Inserte una breve descripción de la lógica algorítmica del método controlador.*
- Nombre /historialRecibidos
  - Ruta: /historialRecibidos
  - Métodos HTTP permitidos: **GET/POST**
  - Lógica algorítmica:  
*Inserte una breve descripción de la lógica algorítmica del método controlador.*

## Base de datos (Sprint 3)



*Se tienen dos tablas una tabla usuario que contiene el id que es auto incrementable, el nombre de usuario que es único al igual que el correo la contraseña que será encriptada y el estado que nos indicará si el usuario esta activo o no. La tabla mensajería tiene como cabeceras el asunto, mensaje, la fecha y hora de envió*

## Prácticas de programación segura (Sprint 3)

*Para la contraseña se utilizó encriptación hash para poder encriptarla y que sea mucho mas segura de almacenar, también se toma el moto try exception para poder tener registros los cuales no son iguales tanto como en nombre de usuario como en correo, por ultimo se controla la inyección sql a través de la eliminación de palabras reservadas como WHERE,INSERT,UPDATE,entre otras.*

## Link repositorio en github:

<https://github.com/Santi980923/SPRINT3-equipo8.git>