

SEGURIDAD JAVA EE:

ASIGNAR ACCESO POR ROLES:

```
package sv.com.gm.sga.servicio;

import java.util.ArrayList;
import java.util.List;
import javax.annotation.Resource;
import javax.annotation.security.*;
import javax.ejb.SessionContext;
import javax.ejb.Stateless;
import javax.inject.Inject;
import javax.jws.WebService;
import sv.com.gm.sga.domain.Persona;
import sv.com.gm.sga.datos.PersonaDao;

@Stateless
@WebService(endpointInterface = "sv.com.gm.sga.servicio.PersonaServiceWS")
@DeclareRoles({"ROLE_ADMIN", "ROLE_USER"})
@RolesAllowed({"ROLE_ADMIN", "ROLE_USER"})
public class PersonaServiceImpl implements PersonaServiceRemote, PersonaService, PersonaServiceWS {

    @Inject
    private PersonaDao personaDao;

    @Resource
    private SessionContext contexto;

    @Override
    public List<Persona> listarPersonas() {
        return personaDao.findAllPersonas();
    }
}
```

localhost:8080/sv.com.gm.sga\_sga-jee-web\_war\_1.0/faces/index.xhtml

Facebook Nueva pestaña 12x^2 -x -1 <0 - W... Inicio Twitch Como instalar direc... Sensor de temperat... 74ls32 - Buscar con... EL PER

HTTP Status 500 - Internal Server Error

**type** Exception report

**message** Internal Server Error

**description** The server encountered an internal error that prevented it from fulfilling this request.

**exception**

javax.servlet.ServletException: WELD-000049: Unable to invoke public void sv.com.gm.sga.web.PersonaBean.inicializar() on sv.com.gm.sga.web.Pers

**root cause**

org.jboss.weld.exceptions.WeldException: WELD-000049: Unable to invoke public void sv.com.gm.sga.web.PersonaBean.inicializar() on sv.com.gm.sga

**root cause**

java.lang.reflect.InvocationTargetException

**root cause**

javax.ejb.EJBAccessException

**root cause**

javax.ejb.AccessLocalException: Client not authorized for this invocation

**note** The full stack traces of the exception and its root causes are available in the GlassFish Server Open Source Edition 5.0 logs.

GlassFish Server Open Source Edition 5.0

CLIENTE WEB:

Start Page x PersonaServiceImpl.java x web.xml x index.xhtml x

Source General Servlets Filters Pages References Security History

```
16 <servlet-class>org.glassfish.jersey.servlet.ServletContainer</servlet-class>
17
18 <init-param>
19   <param-name>jersey.config.server.provider.packages</param-name>
20   <param-value>sv.com.gm.sga.servicio</param-value>
21 </init-param>
22 <load-on-startup>1</load-on-startup>
23 </servlet>
24 <servlet-mapping>
25   <servlet-name>JerseyWebApplication</servlet-name>
26   <url-pattern>/webservice/*</url-pattern>
27 </servlet-mapping>
28 <!--Seguridad-->
29 <login-config>
30   <auth-method>BASIC</auth-method>
31   <realm-name>Login</realm-name>
32 </login-config>
33 <security-constraint>
34   <web-resource-collection>
35     <web-resource-name>AplicacionWebJSF</web-resource-name>
36     <url-pattern>/*</url-pattern>
37   </web-resource-collection>
38   <auth-constraint>
39     <role-name>ROLE_ADMIN</role-name>
40     <role-name>ROLE_USER</role-name>
41     <role-name>ROLE_GUEST</role-name>
42   </auth-constraint>
43 </security-constraint>
</web-app>
```

Start Page x PersonaServiceImpl.java x web.xml x index.xhtml x

Source History

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE html>
3 <html xmlns="http://www.w3.org/1999/xhtml"
4       xmlns:f="http://java.sun.com/jsf/core"
5       xmlns:h="http://java.sun.com/jsf/html"
6       xmlns:p="http://primefaces.org/ui">
7   <h:head>
8     <title>Sistema SGA</title>
9   </h:head>
10  <h:body>
11    <h2>Sistema SGA</h2>
12    Bienvenido: <h:outputText value="#{p:userPrincipal()}" />
13    <h:form>
14      <p:messages />
15      <p:panelGrid columns="1" rendered="#{p:ifGranted('ROLE_ADMIN')} || p:ifGranted('ROLE_USER')}">
16        <h:commandButton value="Listar Personas" action="listarPersonas" />
17      </p:panelGrid>
18    </h:form>
19  </h:body>
20 </html>
```

Users en glasfish:

Source Edition

File Users

Manage user accounts for the currently selected security realm.

Configuration Name: server-config

Realm Name: file

File Users (3)

New

Delete

Select	User ID	Group List:
<input type="checkbox"/>	admin	ROLE_ADMIN ROLE_USER ROLE_GUEST
<input type="checkbox"/>	guest	ROLE_GUEST
<input type="checkbox"/>	user	ROLE_USER

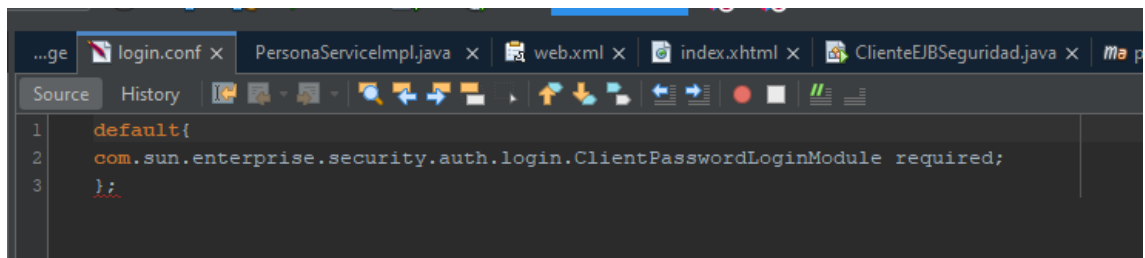
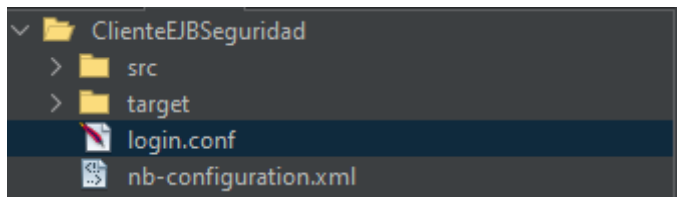
## Sistema SGA

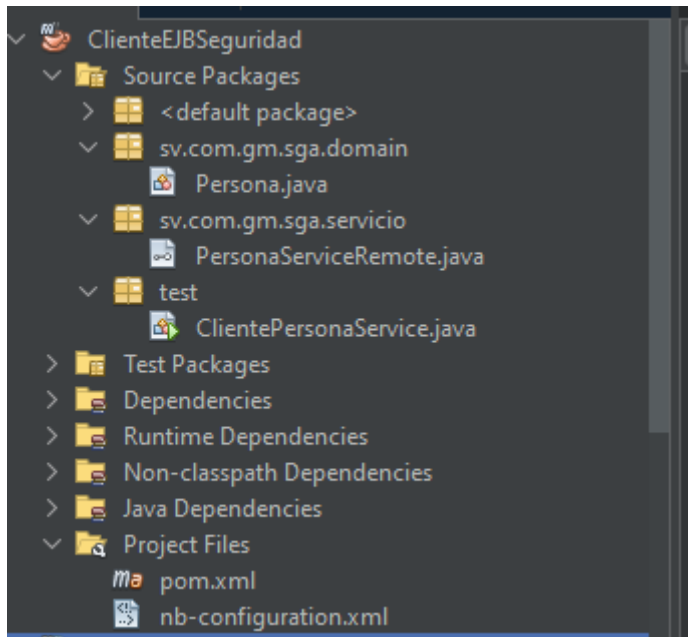
Bienvenido: admin

Listar Personas

Listado de Personas					
Id	Nombre	Apellido	Email	Telefono	Opcion
1	Juan	Higuain	jperez@mail.com	22577777	✓
2	Karla	Gomez	kgomez@mail.com	25252525	✓
3	Maria	Gutierrez	mgutierrez@mail.com	88991199	✓
4	Pedro	Luna	pedro@gmail.com	13132153	✓
6	Hugo	Hernandez	hhernandez@mail.com	55778822	✓
7	pepe	problemas	pepe@hotmail.com	79447751	✓
8	Carlos	Miranda	cmiranda3@mail.com	99887700	✓
9	Carlos	Ramirez	cmiranda3@mail.com	99887700	✓
		Regresar	Agregar		

### CLIENTE EJB CON SEGURIDAD:





```
package test;

import com.sun.enterprise.security.ee.auth.login.ProgrammaticLogin;
import java.util.List;
import javax.naming.*;
import sv.com.gm.sga.domain.Persona;
import sv.com.gm.sga.servicio.PersonaServiceRemote;

public class ClientePersonaService {
    public static void main(String[] args) {
        System.out.println("Iniciando llamada al EJB desde el cliente");

        String authFile = "login.conf";
        System.setProperty("java.security.auth.login.config", authFile);
        ProgrammaticLogin programmaticLogin = new ProgrammaticLogin();
        programmaticLogin.login("admin", "admin".toCharArray());

        try {
            Context jndi = new InitialContext();
            PersonaServiceRemote personaService = (PersonaServiceRemote) jndi.lookup("java:global/sga-jee-web-leccion12/PersonaServiceImpl!sv.com.");

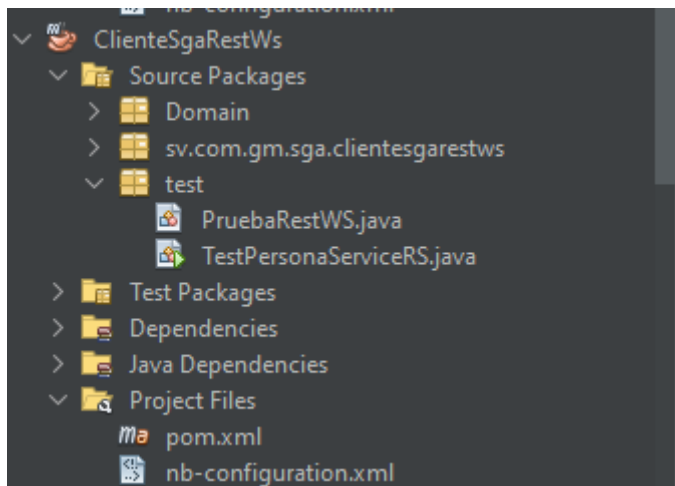
            List<Persona> personas = personaService.listarPersonas();

            for(Persona p : personas){
                System.out.println("persona: " + p);
            }
            System.out.println("Fin llamada al EJB desde el cliente");

        } catch (NamingException ex) {
            ex.printStackTrace(System.out);
        }
    }
}
```

```
--- exec:3.1.0:exec (default-cli) @ ClienteEJBSeguridad ---
Iniciando llamada al EJB desde el cliente
nov 03, 2023 11:26:45 PM com.sun.enterprise.v3.server.CommonClassLoaderServiceImpl findDerbyClient
INFORMACIÓN: Cannot find javadb client jar file, derby jdbc driver will not be available by default.
persona:Persona{idPersona=1, apellido=Higuain, email=jperez@mail.com, telefono=22577777}
persona:Persona{idPersona=2, apellido=Gomez, email=kgomez@mail.com, telefono=25252525}
persona:Persona{idPersona=3, apellido=Gutierrez, email=mgutierrez@mail.com, telefono=88991199}
persona:Persona{idPersona=4, apellido=Luna, email=pedro@gmail.com, telefono=13132153}
persona:Persona{idPersona=6, apellido=Hernandez, email=hhernandez@mail.com, telefono=55778822}
persona:Persona{idPersona=7, apellido=problemas, email=pepe@hotmail.com, telefono=79447751}
persona:Persona{idPersona=8, apellido=Miranda, email=cmiranda3@mail.com, telefono=99887700}
persona:Persona{idPersona=9, apellido=Ramirez, email=cmiranda3@mail.com, telefono=99887700}
```

## Cliente jaxrs:



```
package test;

import domain.Persona;
import javax.ws.rs.client.*;
import javax.ws.rs.core.MediaType;
import org.glassfish.jersey.client.ClientConfig;
import org.glassfish.jersey.client.authentication.HttpAuthenticationFeature;

public class PruebaRestWS {
    public static void main(String[] args) {

        HttpAuthenticationFeature feature = HttpAuthenticationFeature.basicBuilder().nonPreemptive().credentials(username: "admin", password: "admin").build();

        ClientConfig clientConfig = new ClientConfig();
        clientConfig.register(provider: feature);

        Client cliente = ClientBuilder.newClient(configuration: clientConfig);

        WebTarget webTarget = cliente.target(uri: "http://localhost:8080/sv.com.gm.sga.sga-jee-web-war-1.0/webservice").path(path: "/personas");
        //Proporcionamos un idPersona valido
        Persona persona = webTarget.path(path: "/2").request(acceptedResponseTypes: MediaType.APPLICATION_XML).get(responseType: Persona.class);
        System.out.println("Persona recuperada:" + persona);
    }
}
```

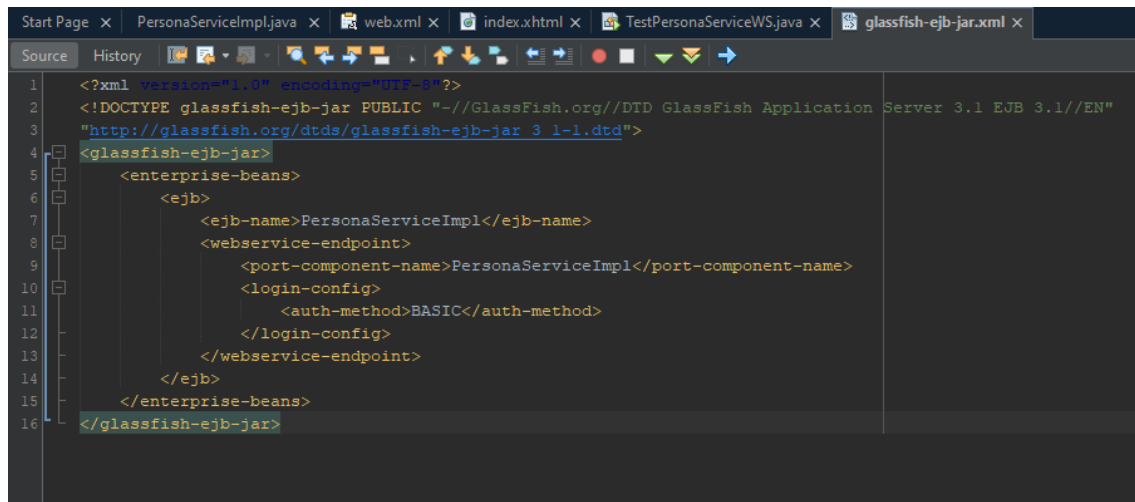
```
--- exec:3.1.0:exec (default-cli) @ ClienteSgaRestWs ---
Persona recuperada:Persona{idPersona=2, nombre=Karla, apellido=Gomez, email=kgomez@mail.com, telefono=25252525}

BUILD SUCCESS

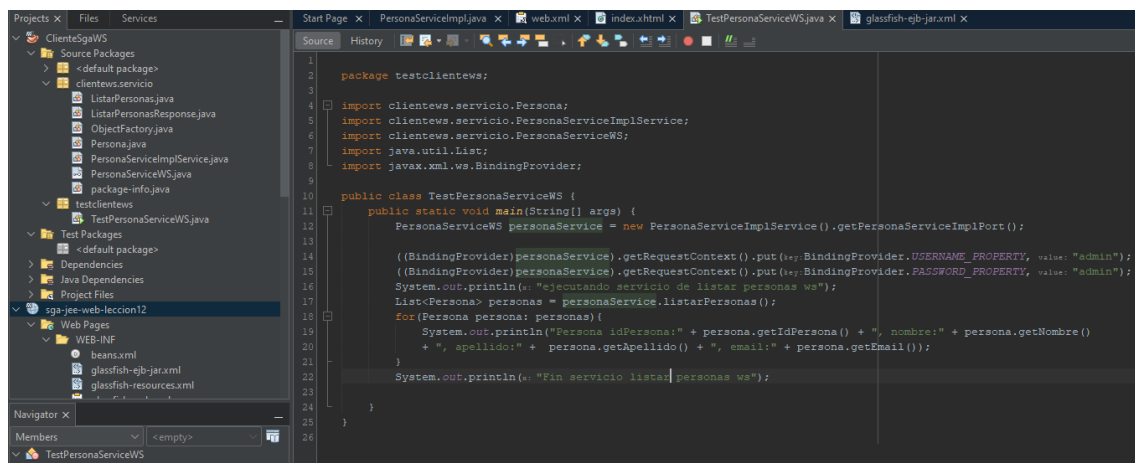
Total time: 1.283 s
Finished at: 2023-11-03T23:33:36-06:00
```

## Cliente web service jaxws

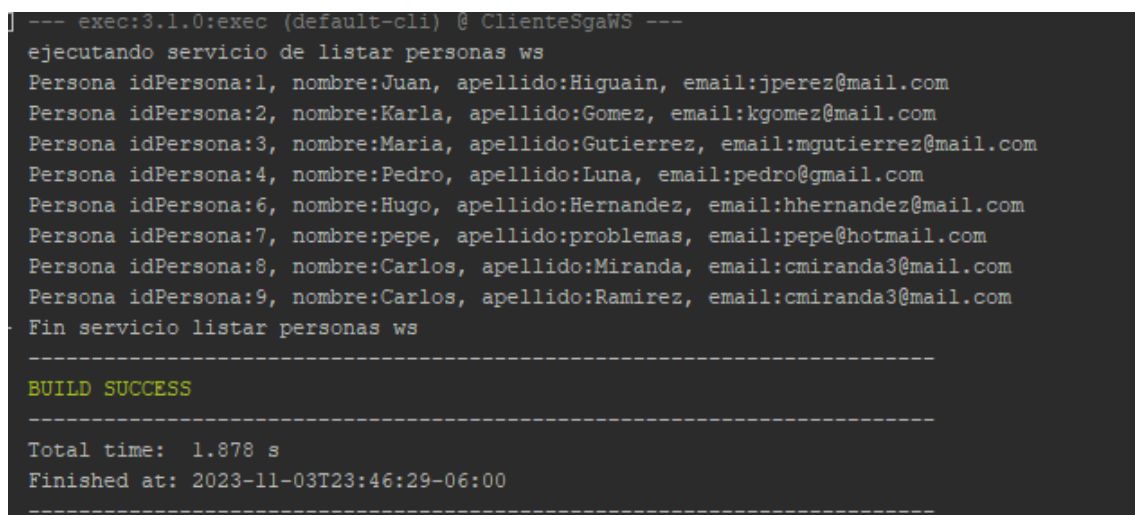
Agregar nuevo archivo al servidor:



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE glassfish-ejb-jar PUBLIC "-//GlassFish.org//DTD GlassFish Application Server 3.1 EJB 3.1//EN"
3 "http://glassfish.org/dtds/glassfish-ejb-jar 3.1-1.dtd">
4 <glassfish-ejb-jar>
5   <enterprise-beans>
6     <ejb>
7       <ejb-name>PersonaServiceImpl</ejb-name>
8       <web-service-endpoint>
9         <port-component-name>PersonaServiceImpl</port-component-name>
10        <login-config>
11          <auth-method>BASIC</auth-method>
12        </login-config>
13      </web-service-endpoint>
14    </ejb>
15  </enterprise-beans>
16 </glassfish-ejb-jar>
```



```
1 package testclientews;
2
3 import clientews.servicio.Persona;
4 import clientews.servicio.PersonaServiceImplService;
5 import clientews.servicio.PersonaServiceImplService;
6 import java.util.List;
7 import javax.xml.ws.BindingProvider;
8
9 public class TestPersonaServiceWS {
10   public static void main(String[] args) {
11     PersonaServiceImplService personaService = new PersonaServiceImplService();
12
13     ((BindingProvider) personaService).getRequestContext().put((key) BindingProvider.USERNAME_PROPERTY, value: "admin");
14     ((BindingProvider) personaService).getRequestContext().put((key) BindingProvider.PASSWORD_PROPERTY, value: "admin");
15     System.out.println("Ejecutando servicio de listar personas ws");
16     List<Persona> personas = personaService.listarPersonas();
17     for(Persona persona: personas){
18       System.out.println("Persona idPersona: " + persona.getIdPersona() + ", nombre: " + persona.getNombre()
19         + ", apellido: " + persona.getApellido() + ", email: " + persona.getEmail());
20     }
21     System.out.println("Fin servicio listar personas ws");
22   }
23 }
24
25
26
```



```
--- exec:3.1.0:exec (default-cli) @ ClienteSgaWS ---
ejecutando servicio de listar personas ws
Persona idPersona:1, nombre:Juan, apellido:Higuain, email:jperez@mail.com
Persona idPersona:2, nombre:Karla, apellido:Gomez, email:kgomez@mail.com
Persona idPersona:3, nombre:Maria, apellido:Gutierrez, email:mgutierrez@mail.com
Persona idPersona:4, nombre:Pedro, apellido:Luna, email:pedro@gmail.com
Persona idPersona:6, nombre:Hugo, apellido:Hernandez, email:hhernandez@mail.com
Persona idPersona:7, nombre:pepe, apellido:problemas, email:pepe@hotmail.com
Persona idPersona:8, nombre:Carlos, apellido:Miranda, email:cmiranda3@mail.com
Persona idPersona:9, nombre:Carlos, apellido:Ramirez, email:cmiranda3@mail.com
Fin servicio listar personas ws

BUILD SUCCESS

Total time: 1.878 s
Finished at: 2023-11-03T23:46:29-06:00
```