**Ejemplo Proyecto EJB**

**1.- Proyecto Web**

Crear un proyecto web dinámico. Para este ejemplo lo llamaremos EJBTestWeb

Seleccionar WildFly como Target Runtime

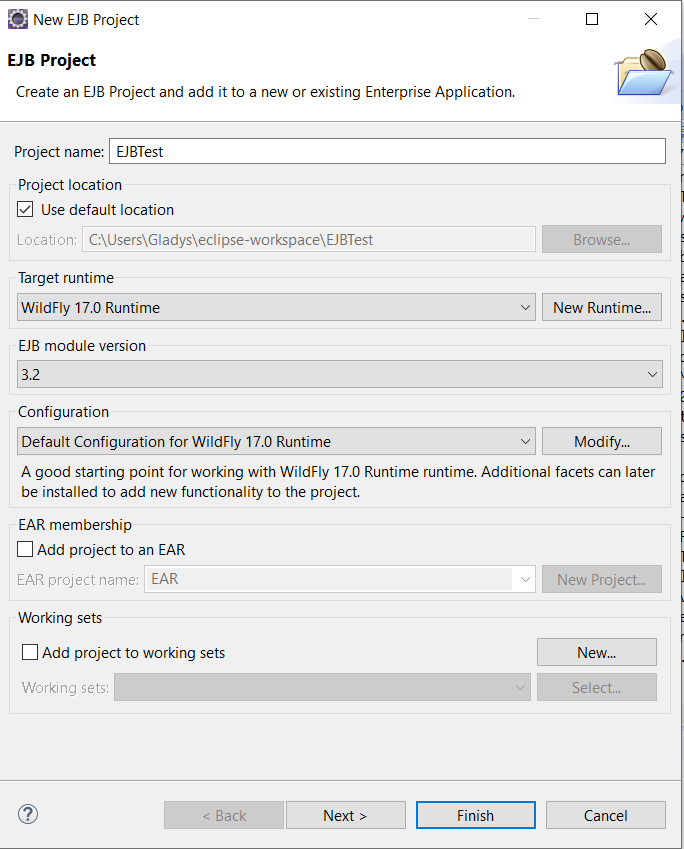
**2.- Proyecto EJB**

File-> New EJB Project

En nombre escribir EJBTest.

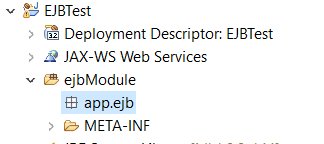
En Target runtime seleccionar WildFly

En configuration Default Configuration for WildFly 17.0

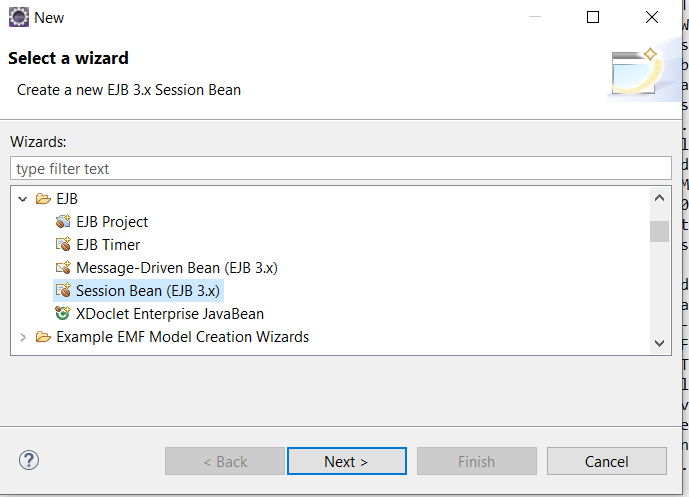


Dar click en Finish

A continuación se crear[a un EJB de sesión. Primero creamos un paquete en ejbModule con nombre app.ejb

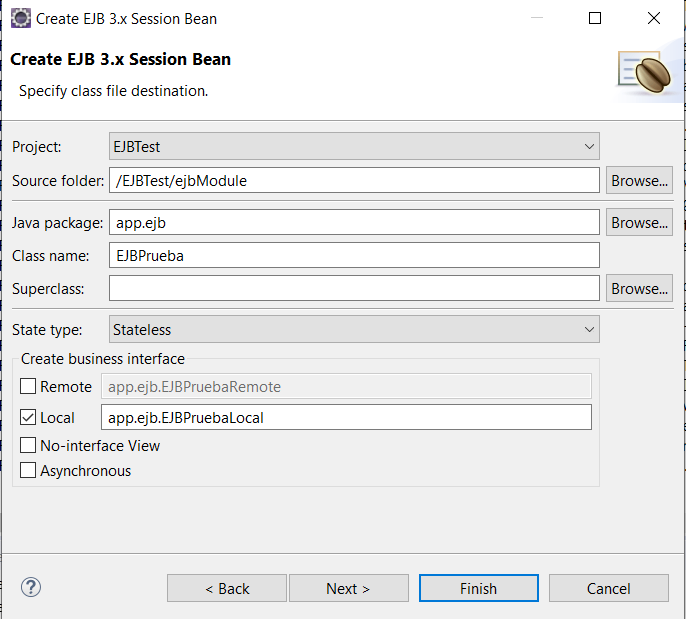


En el paquete crear dar click en New - Other EJB y seleccionar Session Bean



Dar click en Next.

En la siguiente pantalla se escribe el nombre, se selecciona el tipo y se selecciona que se creará una interface Local:



Dar click en Finish

Luego de estos se crean dos archivos. Uno de ellos es una interface donde vamos a definir el método para la clase concreta.

public String saludo(String nombre);

Ahora en la clase EJBPrueba hay que implementar el método

@Override

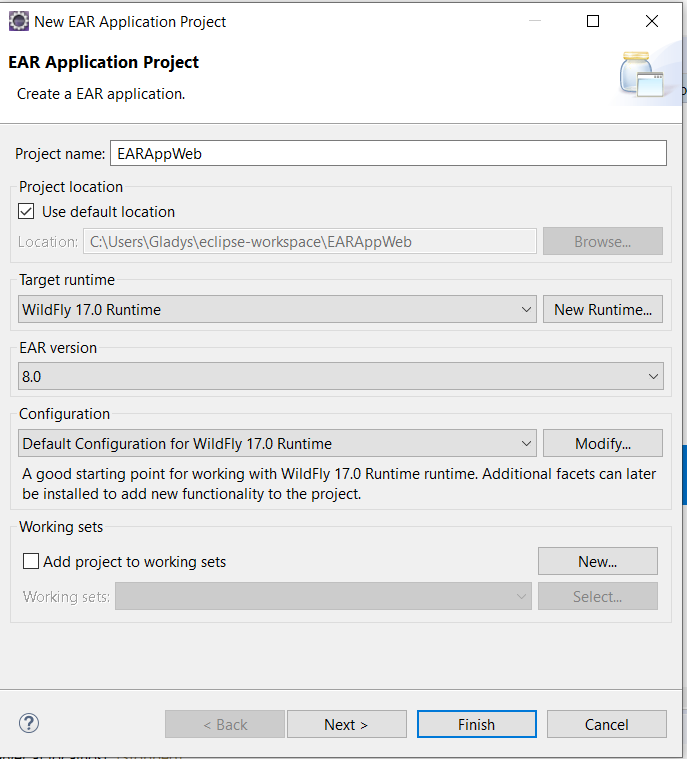
public String saludo(String nombre) {

return "Hola "+ nombre + " desde un EJB";

}

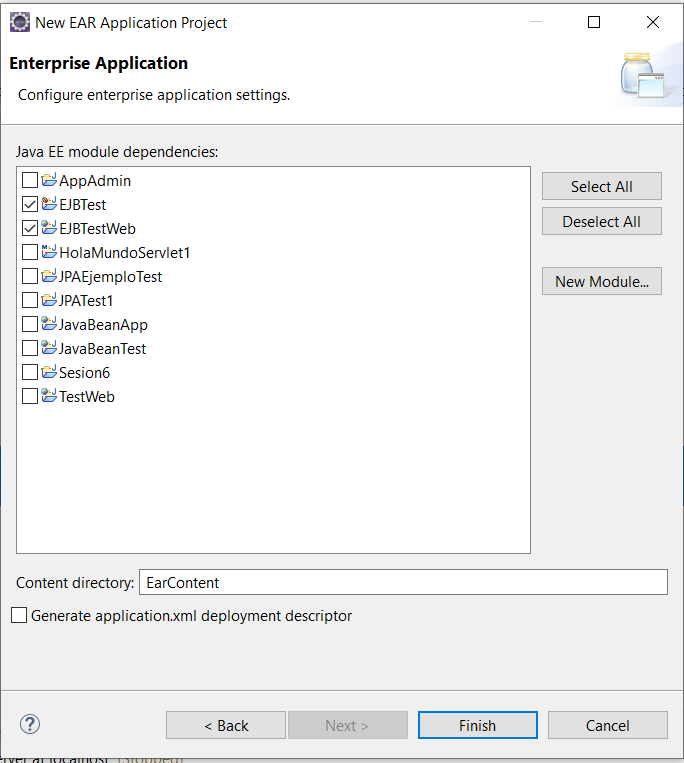
Vamos a crear un nuevo proyecto para enlazar la parte web con el EJB

New -- Enterprise Aplication Project, llamaremos al proyecto EARAppWeb

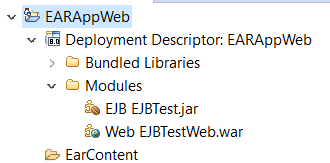


Dar click en Next

En la siguiente ventana seleccionar los módulos que serán parte de este proyecto y dar click en Finish



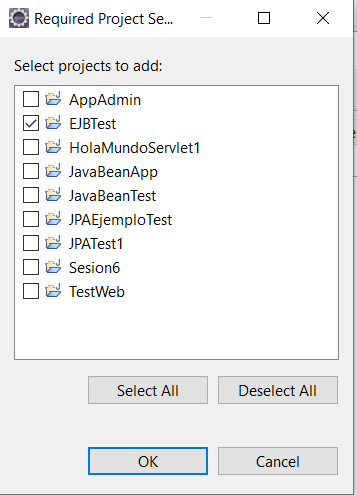
Si exploramos el proyecto vemos que los dos proyectos han sido agregados en Modules



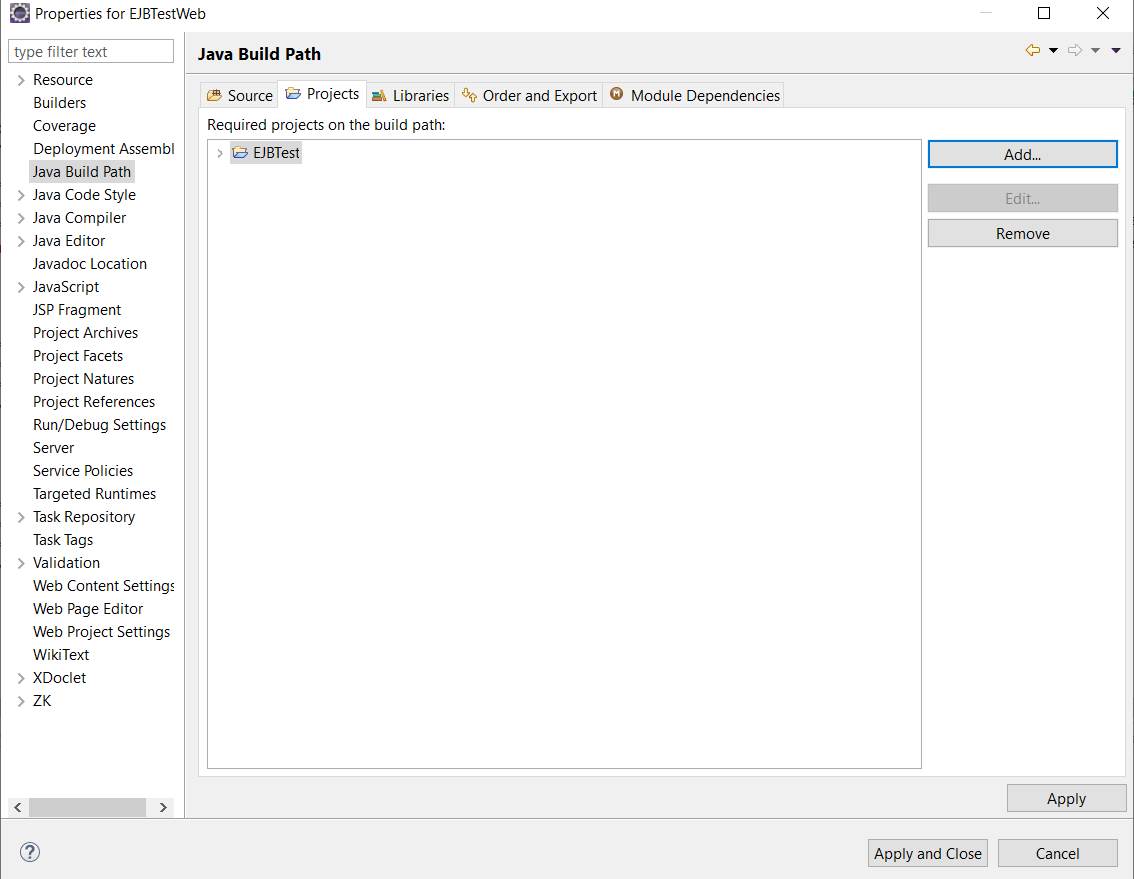
Para consumir el EJB en el proyecto Web hay que agregarlo en el BuilPath.

Vamos a las propiedades del proyecto EJBTestWeb seleccionamos Java Build Path y seleccionamos la pestaña Projects, aquí damos click en Add..

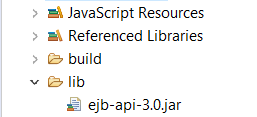
En la nueva ventana seleccionamos EJBTest y cick en Ok



En la siguiente ventana click en Apply and Close



Crear una carpeta lib en el directorio del proyecto web y agregar el jar correspondiente al ejp-api



en la carpeta src del proyecto web, crear un paquete app.servlet. En esa carpeta crear un Servlet, llamar ServletIndex

A este servlet se agrega un atributo para utilizar el EJB

@EJB

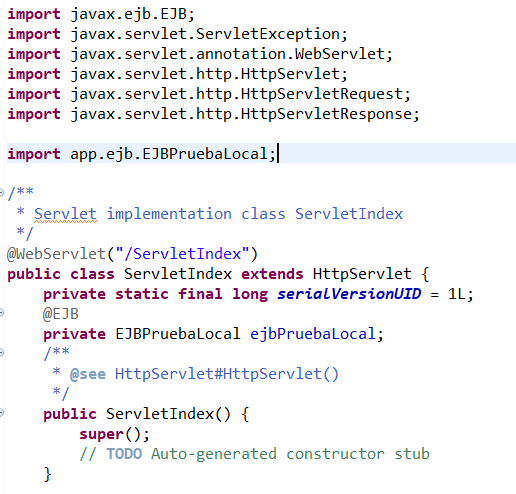
private EJBPruebaLocal ejbPruebaLocal;

Para este atributo se necesitará importar

import javax.ejb.EJB;

import app.ejb.EJBPruebaLocal;

El servlet va quedando de la siguiente forma:



En el método doGet se escribirá el código para hacer la llamada al EJB

String saludo = ejbPruebaLocal.saludo("Gladys");

request.setAttribute("saludo", saludo);

request.getRequestDispatcher("index.jsp").forward(request, response);

Se crea ahora un jsp para mostrar el mensaje

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Saludo con EJB</title>

</head>

<body>

<h1>Saludo con EJB</h1>

<%=request.getAttribute("saludo") %>

</body>

</html>

Luego de esto ejecutamos la aplicación EARAppWeb para ver la salida

