

UNIVERSIDAD DEL QUINDÍO
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

Información general	
Actualizada por:	Einer Zapata
Duración estimada en minutos:	90
Docente:	Christian Andrés Candela
Guía no.	11
Nombre de la guía:	EJB de Sesión sin Estado

Información de la Guía

Objetivo

Comprender que son los EJB de sesión y aprender a crearlos y configurarlos para el desarrollo de la lógica de negocio (capa de negocio de la aplicación).

CONCEPTOS BÁSICOS

Manejo de Eclipse, Java, Bases de Datos, JDBC, DataSource, Entidades y Glassfish.

PRECAUCIONES Y RECOMENDACIONES

Recuerde verificar que el servidor de aplicaciones soporte el motor de base de datos que usará, de igual forma debe verificar que eclipse este asiendo uso del JDK y no del JRE y recuerde adicionar al workspace el servidor de aplicaciones Glassfish antes de crear cualquier proyecto. También puede ser importante verificar que los puertos usados por Glassfish no estén ocupados (Para ello puede hacer uso del comando **netstat -npl** o **netstat -a**)

ARTEFACTOS

Se requiere tener instalado el JDK y un IDE para el desarrollo de aplicaciones (Eclipse JEE en su última versión), un servidor de aplicaciones que cumpla con las especificaciones de JEE, para esta práctica Glassfish y el motor de base de datos Mysql.

EVALUACIÓN O RESULTADO

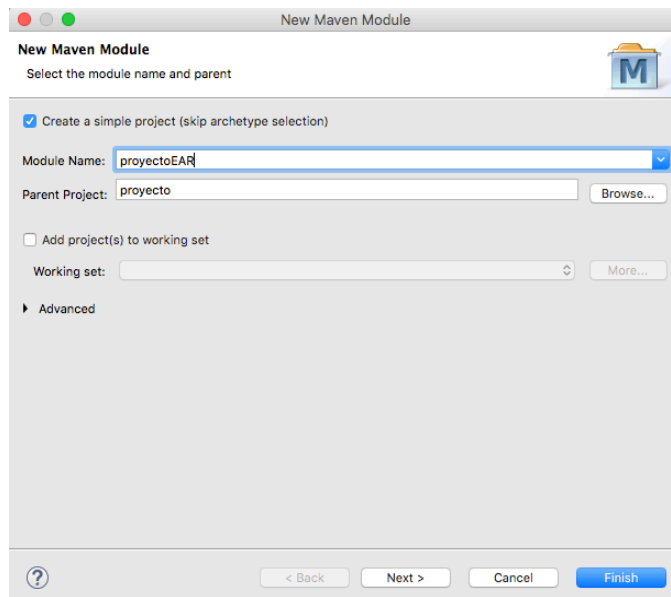
Se espera que el alumno pueda crear y usar exitosamente un EJB de sesión para la creación de la lógica de negocio de las aplicaciones.

Procedimiento

1. Para el desarrollo de esta guía necesitara una base de datos en mysql, un proyecto de tipo maven – pom, el cual contenga un proyecto maven con soporte para el uso de JPA y otro proyecto maven configurado para

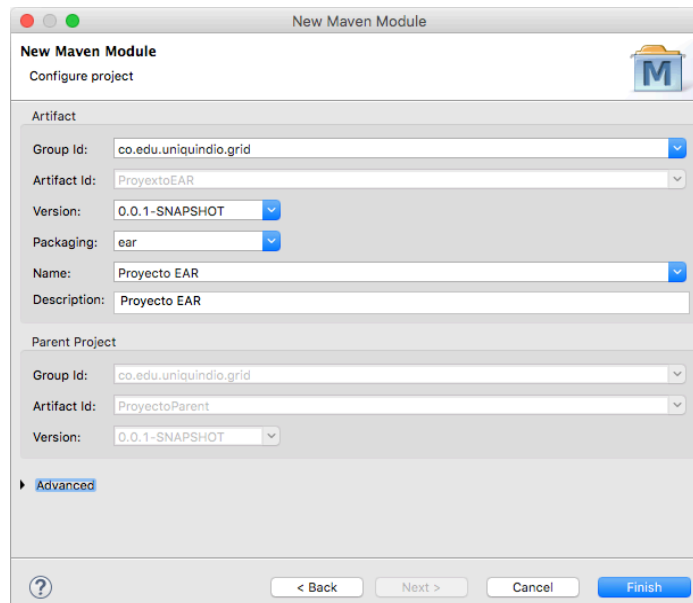
la realización de pruebas. Y una conexión a dicha base de datos para ser usada en la generación de las tablas.

2. Asegúrese de tener al menos una entidad en su proyecto JPA, de no tenerla deberá crearla.
3. Adicione un nuevo módulo a su proyecto padre. Un módulo en un proyecto padre no es otra cosa que otro proyecto maven, el cual será contenido por el proyecto padre y heredará de él algunas de sus características. Adicione un nuevo modulo en el proyecto padre llamado EAR. El proyecto EAR es el encargado de desplegar el proyecto en el servidor.



The screenshot shows the 'New Maven Module' dialog box with the title 'New Maven Module'. Below the title is the instruction 'Select the module name and parent'. There is a checkbox 'Create a simple project (skip archetype selection)' which is checked. The 'Module Name' field contains 'proyectoEAR'. The 'Parent Project' field contains 'proyecto' with a 'Browse...' button next to it. There is an unchecked checkbox 'Add project(s) to working set' and a 'Working set:' dropdown menu with a 'More...' button. An 'Advanced' section is collapsed. At the bottom are buttons for '< Back', 'Next >', 'Cancel', and 'Finish'.

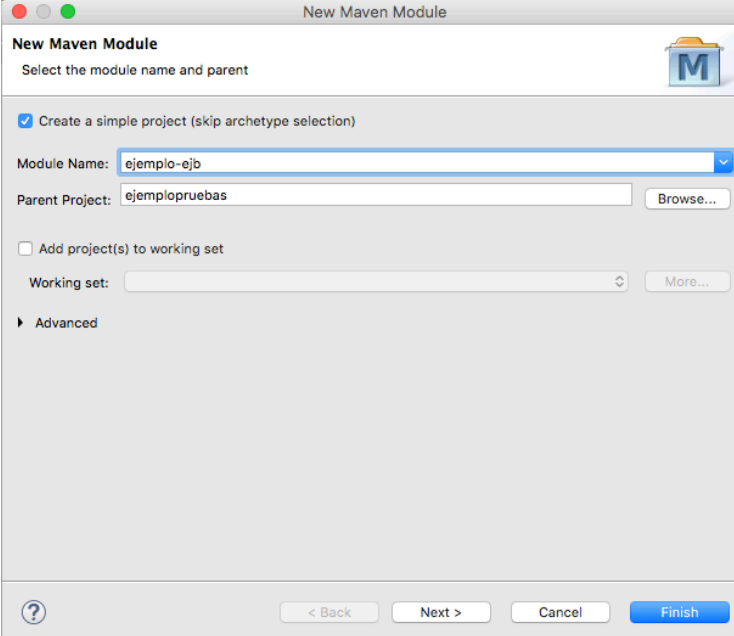
Next, Deberá seleccionar el tipo de empaquetamiento (ear) y proporcionar un nombre y una descripción.



The screenshot shows the 'New Maven Module' dialog box with the title 'New Maven Module'. Below the title is the instruction 'Configure project'. The 'Artifact' section contains fields for 'Group Id' (co.edu.uniquindio.grid), 'Artifact Id' (ProyextoEAR), 'Version' (0.0.1-SNAPSHOT), 'Packaging' (ear), 'Name' (Proyecto EAR), and 'Description' (Proyecto EAR). The 'Parent Project' section contains fields for 'Group Id' (co.edu.uniquindio.grid), 'Artifact Id' (ProyectoParent), and 'Version' (0.0.1-SNAPSHOT). An 'Advanced' section is collapsed. At the bottom are buttons for '?', '< Back', 'Next >', 'Cancel', and 'Finish'.

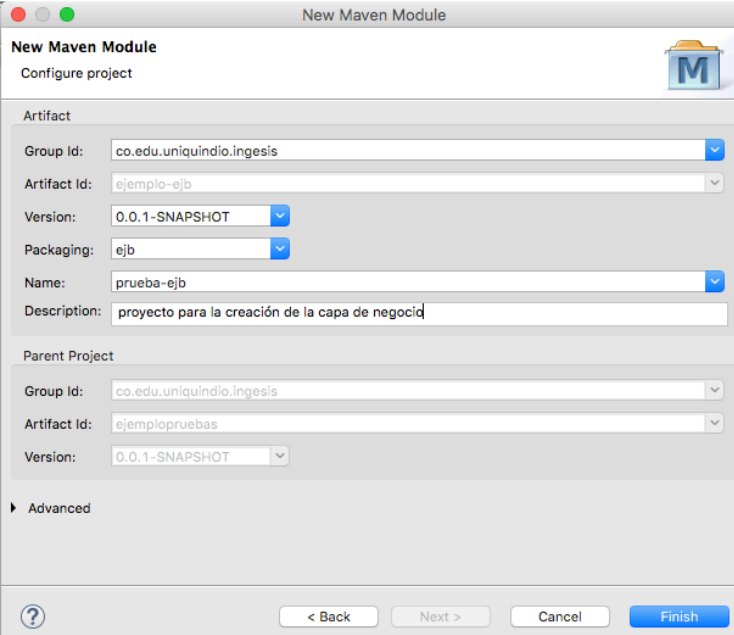
Adicione como dependencia del proyecto EAR el modulo de persistencia. Ahora debe dar clic derecho sobre el proyecto padre y en la opción maven seleccionar Update Project.

4. Adicione un nuevo módulo a su proyecto padre. El nuevo módulo maven será el proyecto EJB o Negocio. Para ello debe dar clic derecho sobre el proyecto padre y en la opción maven seleccionar new maven module Project. Para este módulo se creará la configuración por defecto, por lo que no se requiere selección de arquetipo, esto lo logrará seleccionando la opción create simple Project.



The screenshot shows the 'New Maven Module' dialog box with the title 'New Maven Module'. Below the title is the instruction 'Select the module name and parent'. There is a checkbox labeled 'Create a simple project (skip archetype selection)' which is checked. Below this, there are two text input fields: 'Module Name' with the value 'ejemplo-ejb' and 'Parent Project' with the value 'ejemplopruebas'. There is a 'Browse...' button next to the 'Parent Project' field. Below these fields is a checkbox labeled 'Add project(s) to working set' which is unchecked. Below that is a 'Working set:' dropdown menu and a 'More...' button. At the bottom, there is a section labeled 'Advanced' which is collapsed. At the very bottom, there are four buttons: '< Back', 'Next >', 'Cancel', and 'Finish'.

En la siguiente ventana deberá seleccionar como tipo de empaquetado EJB.

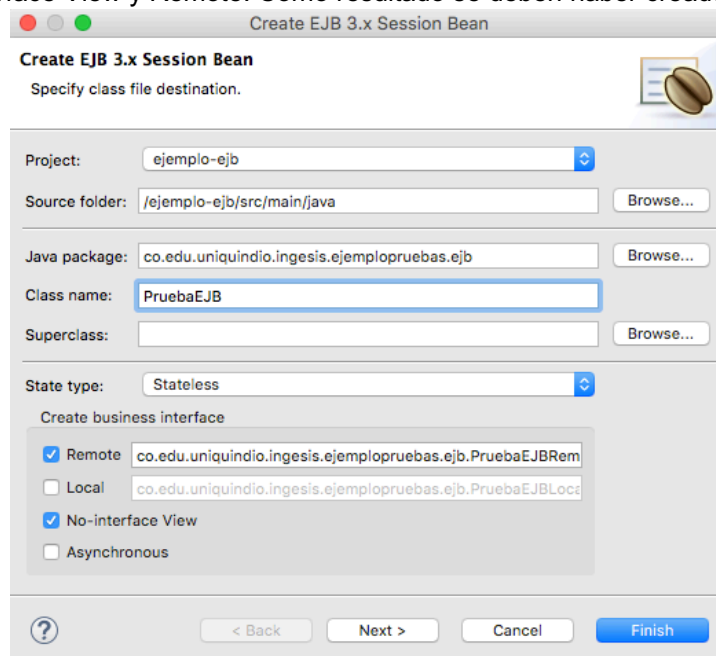


The screenshot shows the 'New Maven Module' dialog box with the title 'New Maven Module'. Below the title is the instruction 'Configure project'. There is a checkbox labeled 'Create a simple project (skip archetype selection)' which is checked. Below this, there are several text input fields and dropdown menus: 'Group Id' with the value 'co.edu.uniquindio.ingesis', 'Artifact Id' with the value 'ejemplo-ejb', 'Version' with the value '0.0.1-SNAPSHOT', 'Packaging' with the value 'ejb', 'Name' with the value 'prueba-ejb', and 'Description' with the value 'proyecto para la creación de la capa de negocio'. Below these fields is a section labeled 'Parent Project' which contains three dropdown menus: 'Group Id' with the value 'co.edu.uniquindio.ingesis', 'Artifact Id' with the value 'ejemplopruebas', and 'Version' with the value '0.0.1-SNAPSHOT'. At the bottom, there is a section labeled 'Advanced' which is collapsed. At the very bottom, there are four buttons: '?', '< Back', 'Next >', 'Cancel', and 'Finish'.

5. Adicione la siguiente dependencia a su archivo pom

```
<dependency>
  <groupId>javax</groupId>
  <artifactId>javaee-api</artifactId>
  <version>8.0</version>
  <scope>provided</scope>
</dependency>
```

6. Como dependencia del proyecto ejb adicione el proyecto de persistencia.
7. En su proyecto EJB, cree un EJB para la gestión de su capa de negocio con el nombre de su preferencia, para ello acceda al menu File – New - Other – Session Bean (EJB 3.x). Cuando se le pregunte por el paquete del EJB puede crear un paquete de su preferencia, lo mismo para el nombre del EJB. Seleccione las opciones No-Interface View y Remote. Como resultado se deben haber creado una clase y una interfaz.



8. Adicione a su EJB un atributo de tipo EntityManager, con la anotación @PersistenceContext

```
@PersistenceContext
private EntityManager entityManager;
```

9. Adicione como dependencia del proyecto EAR el modulo del proyecto EJB. Ahora debe dar clic derecho sobre el proyecto padre y en la opción maven seleccionar Update Project.
10. Para probar la correcta configuración despliegue el proyecto en el servidor de aplicaciones. Para ello de clic derecho sobre el proyecto EAR y acceda a la opción Run As – Run on Server.
11. Si todo ha salido bien, podrá dar clic derecho sobre el servidor de aplicaciones y en el menú Glassfish seleccione la opción view log file. Dentro del log deberá encontrar una línea como la siguiente:

INFO: Portable JNDI names for EJB NombreDelEJB

12. En la base de datos verifique que se hayan creado las tablas correspondientes a las entidades de su proyecto.
13. Identifique al menos 5 operaciones de negocio y desarróllelas en métodos individuales. Por ejemplo, Si debiera ingresar un nuevo usuario en una aplicación, debería no solo grabar el usuario en la base de datos, también debería verificar previamente si dicho usuario no se encuentra ya registrado, o si el login (o nombre de usuario) no está siendo usado por otro usuario. Cree sus propias excepciones de ser necesario.

```
public void registrarUsuario(Usuario usuario){  
  
    if( entityManager.find( Usuario.class,usuario.getCedula() ) != null ){  
        Throw new Exception("El usuario ya está registrado");  
    }  
    if( buscarPorLogin(usuario.getLogin()) != null ){  
        Throw new RuntimeException("El login ya está siendo usado por otro usuario");  
    }  
    try{  
        entityManager.persist(usuario);  
    } catch( Exception e){  
        // Registrar log  
        Throw new RuntimeException("Problema al registrar el usuario");  
    }  
}
```

14. Después de crear el método, debe registrar dicho método en la interfaz del EJB.
15. Repita los pasos 14 y 15 para las 5 operaciones identificadas.
NOTA: Es importante conservar este proyecto, ya que su desarrollo continuará en la siguiente guía.
16. Cree las pruebas unitarias que sean necesarias para verificar el funcionamiento de los EJB, tenga en cuenta que en las pruebas unitarias las instancias de los EJB deben declararse así:

```
@EJB  
private PersonaEJB personaEJB;
```

Y el método para el despliegue sería así:

```
@Deployment  
public static Archive<?> createDeploymentPackage() {  
    return ShrinkWrap.create(JavaArchive.class).addClass(NOMBRE_EJB.class)  
        .addPackage(Entidad.class.getPackage())  
        .addAsResource("persistenceForTest.xml", "META-INF/persistence.xml")  
        .addAsManifestResource(EmptyAsset.INSTANCE, "beans.xml");  
}
```