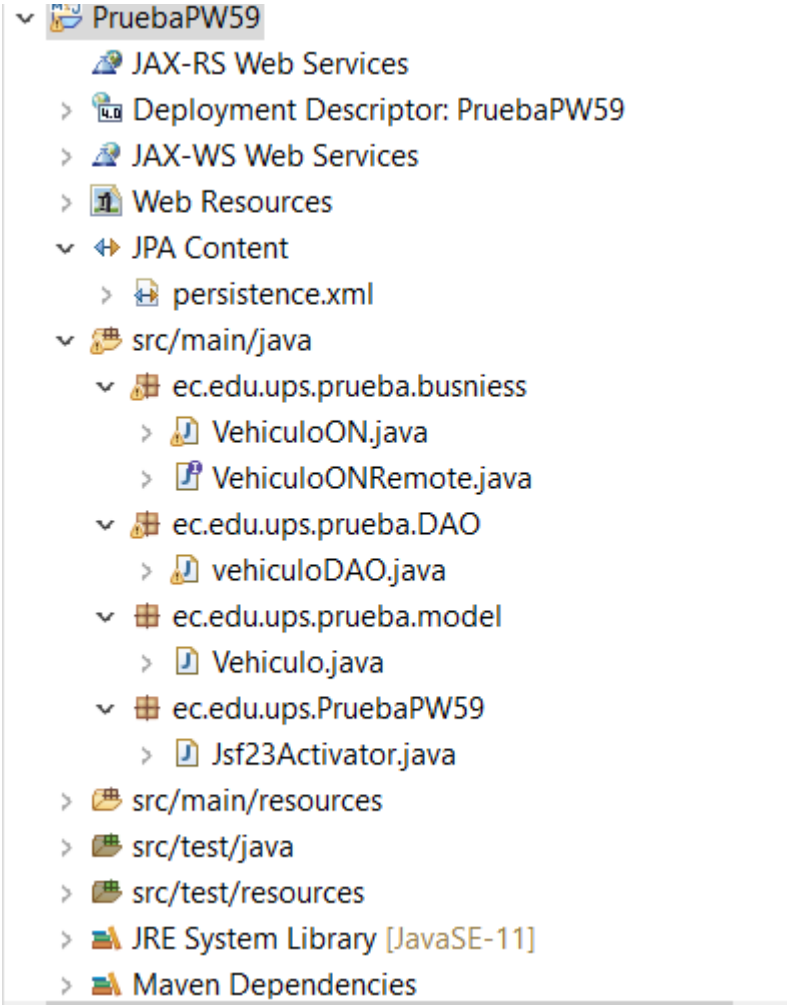
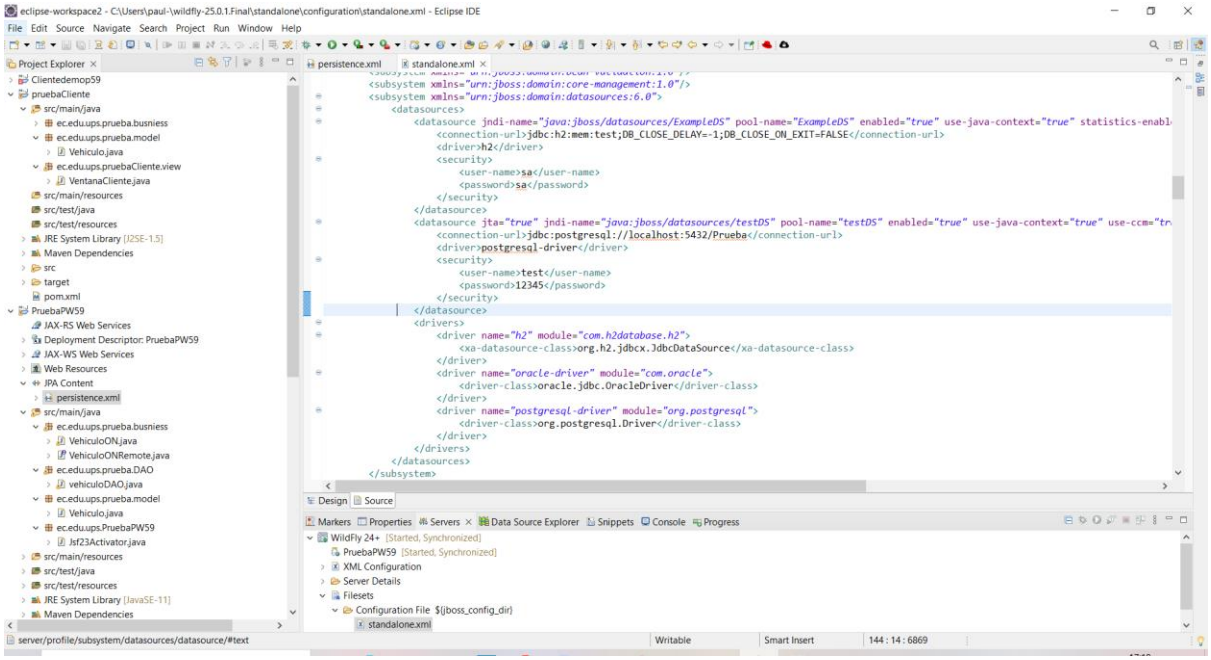
EVALUACION 2

NOMBRE: Paul Guzhñay

Creación del Proyecto Servidor

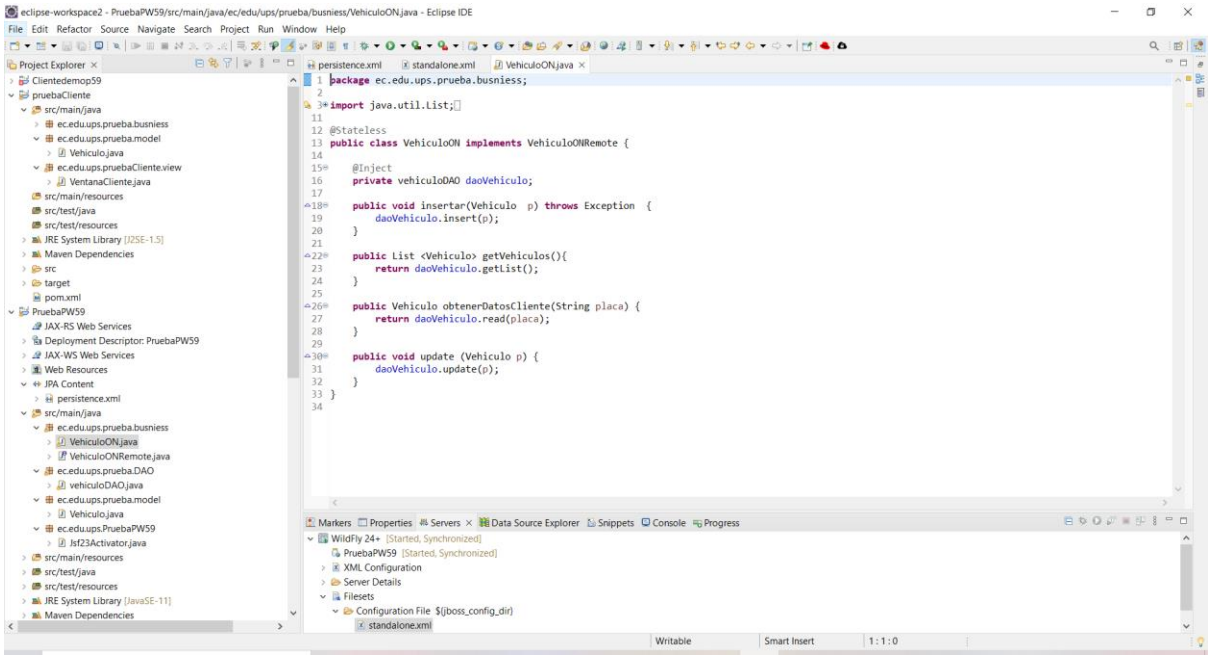


Conexión con la Base de Datos

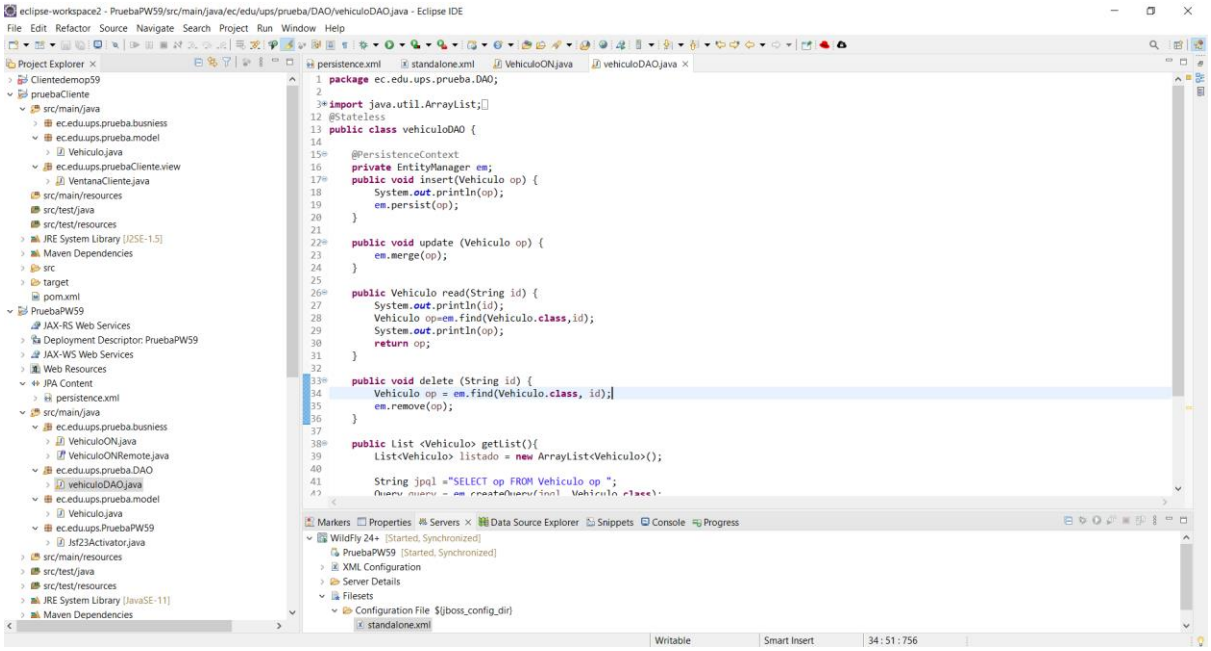


Creación de los paquetes business, model, DAO y view

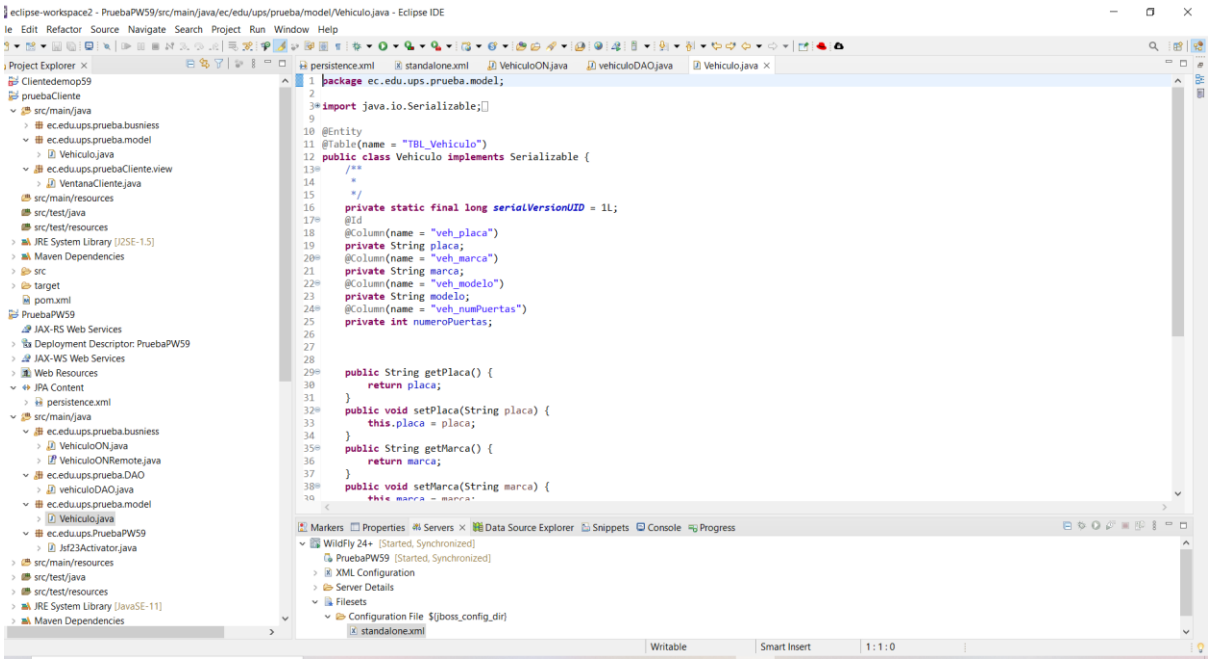
Business



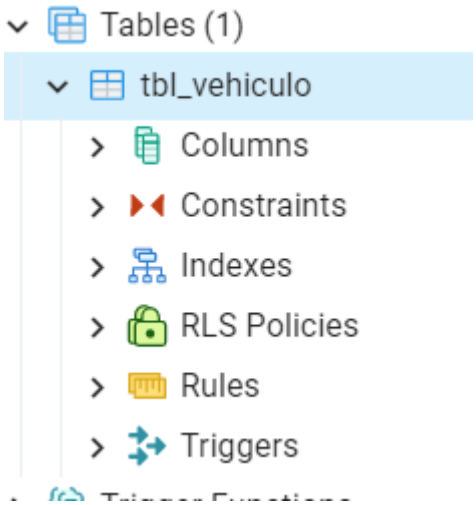
DAO


eclipse-workspace2 - PruebaPW59/src/main/java/ec/edu/ups/prueba/DAO/vehiculoDAO.java - Eclipse IDE
File Edit Refactor Source Navigate Search Project Run Window Help
Project Explorer
Clientedemop59
pruebaCliente
src/main/java
ec.edu.ups.prueba.busniess
ec.edu.ups.prueba.model
Vehiculo.java
ec.edu.ups.pruebaCliente.view
VentanaCliente.java
src/main/resources
src/test/java
src/test/resources
JRE System Library [J2SE-1.5]
Maven Dependencies
src
target
pom.xml
PruebaPW59
JAX-RS Web Services
Deployment Descriptor: PruebaPW59
JAX-WS Web Services
Web Resources
JPA Content
persistence.xml
src/main/java
ec.edu.ups.prueba.busniess
VehiculoON.java
VehiculoONRemote.java
ec.edu.ups.prueba.DAO
vehiculoDAO.java
ec.edu.ups.prueba.model
Vehiculo.java
ec.edu.ups.PruebaPW59
Jsf23Activator.java
src/main/resources
src/test/java
src/test/resources
JRE System Library [JavaSE-11]
Maven Dependencies
persistence.xml
standalone.xml
VehiculoON.java
vehiculoDAO.java
1 package ec.edu.ups.prueba.DAO;
2
3 import java.util.ArrayList;
12 @Stateless
13 public class vehiculoDAO {
14
15 @PersistenceContext
16 private EntityManager em;
17 public void insert(Vehiculo op) {
18 System.out.println(op);
19 em.persist(op);
20 }
21
22 public void update (Vehiculo op) {
23 em.merge(op);
24 }
25
26 public Vehiculo read(String id) {
27 System.out.println(id);
28 Vehiculo op=em.find(Vehiculo.class,id);
29 System.out.println(op);
30 return op;
31 }
32
33 public void delete (String id) {
34 Vehiculo op = em.find(Vehiculo.class, id);
35 em.remove(op);
36 }
37
38 public List <Vehiculo> getList(){
39 List<Vehiculo> listado = new ArrayList<Vehiculo>();
40
41 String jpql ="SELECT op FROM Vehiculo op ";
Markers Properties Servers Data Source Explorer Snippets Console Progress
WildFly 24+ [Started, Synchronized]
PruebaPW59 [Started, Synchronized]
XML Configuration
Server Details
Filesets
Configuration File ${jboss_config_dir}
standalone.xml
Writable
Smart Insert
34 : 51 : 756

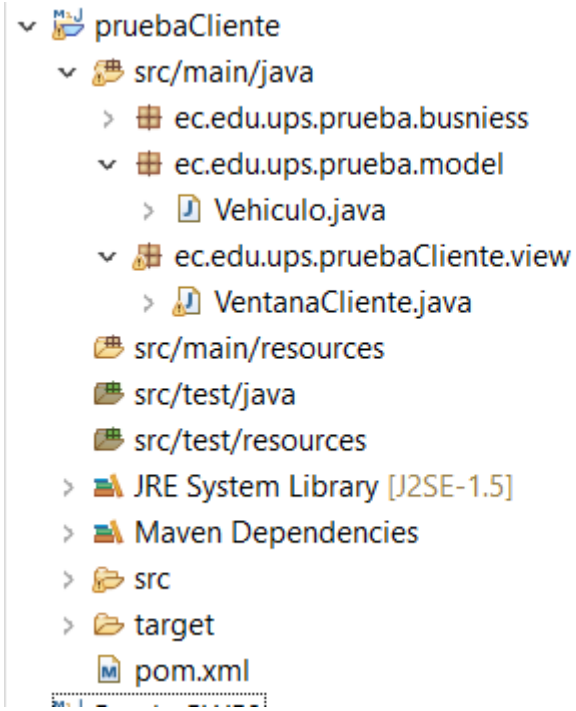
MODEL


eclipse-workspace2 - PruebaPW59/src/main/java/ec/edu/ups/prueba/model/Vehiculo.java - Eclipse IDE
Project Explorer
persistence.xml
standalone.xml
VehiculoON.java
vehiculoDAO.java
Vehiculo.java
1 package ec.edu.ups.prueba.model;
2
3 import java.io.Serializable;
9
10 @Entity
11 @Table(name = "TBL_Vehiculo")
12 public class Vehiculo implements Serializable {
13 /**
14 *
15 */
16 private static final long serialVersionUID = 1L;
17 @Id
18 @Column(name = "veh_placa")
19 private String placa;
20 @Column(name = "veh_marca")
21 private String marca;
22 @Column(name = "veh_modelo")
23 private String modelo;
24 @Column(name = "veh_numPuertas")
25 private int numeroPuertas;
26
27
28
29 public String getPlaca() {
30 return placa;
31 }
32 public void setPlaca(String placa) {
33 this.placa = placa;
34 }
35 public String getMarca() {
36 return marca;
37 }
38 public void setMarca(String marca) {
Markers Properties Servers Data Source Explorer Snippets Console Progress
WildFly 24+ [Started, Synchronized]
PruebaPW59 [Started, Synchronized]
XML Configuration
Server Details
Filesets
Configuration File ${jboss_config_dir}
standalone.xml
Writable
Smart Insert
1 : 1 : 0

Creación de la tabla

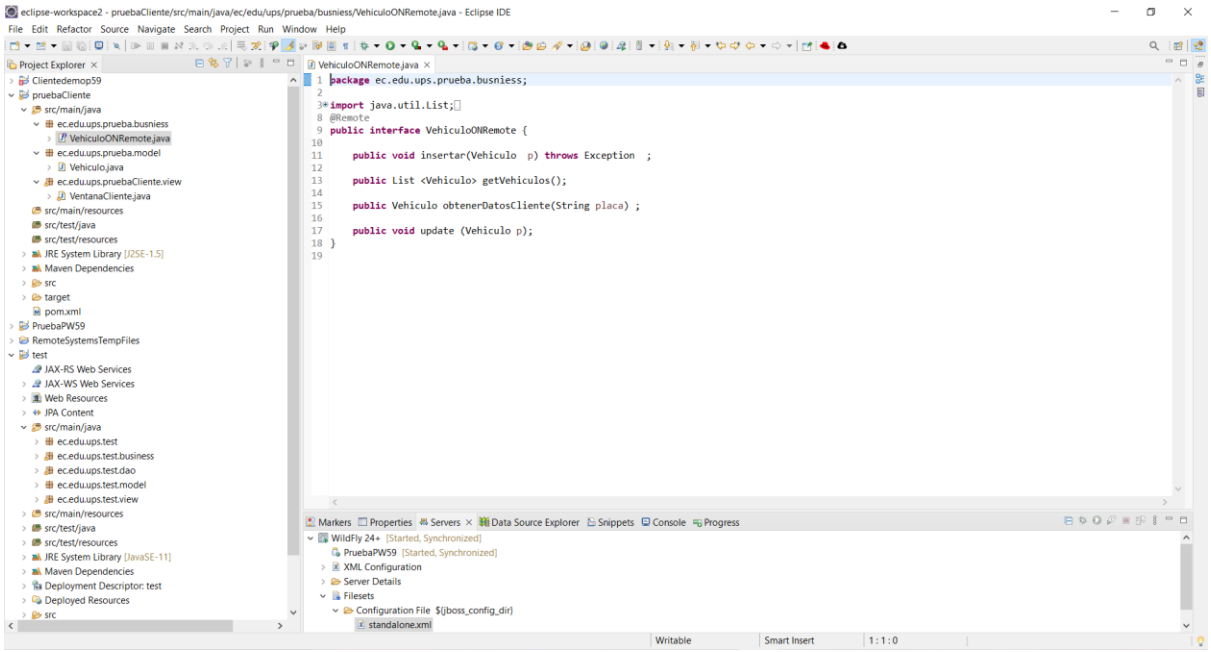


Creación del Proyector-Cliente

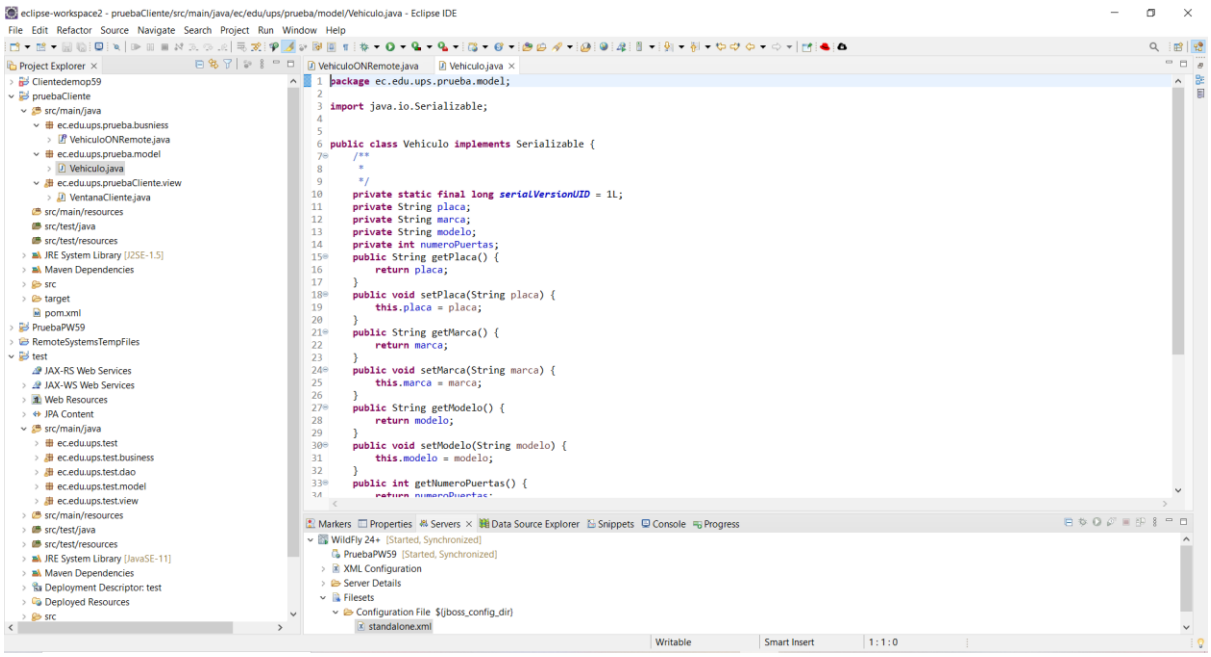


Creación de los paquetes business,model y view

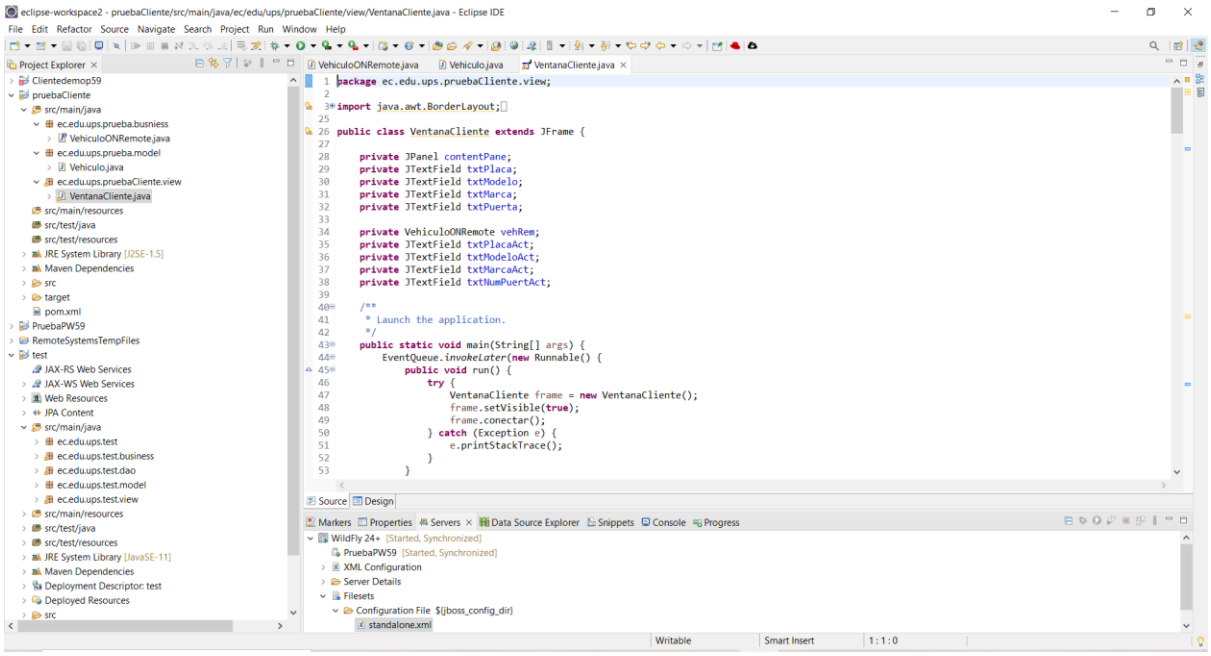
Business



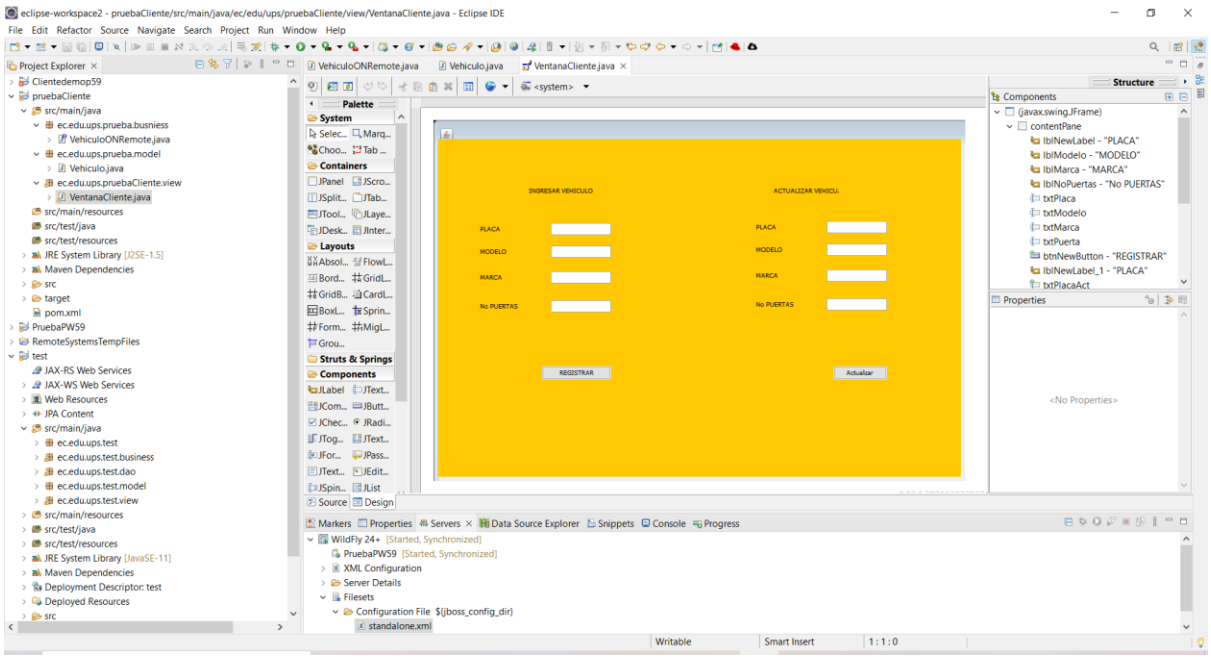
Model



View



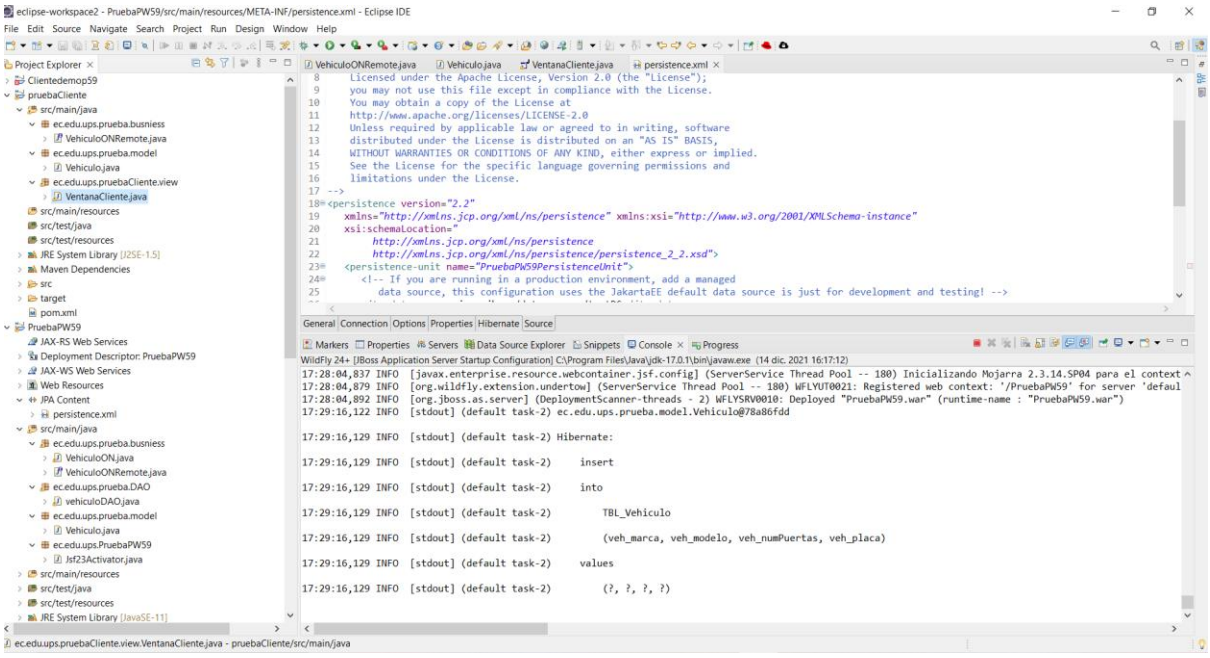
Creación de la interfaz grafica



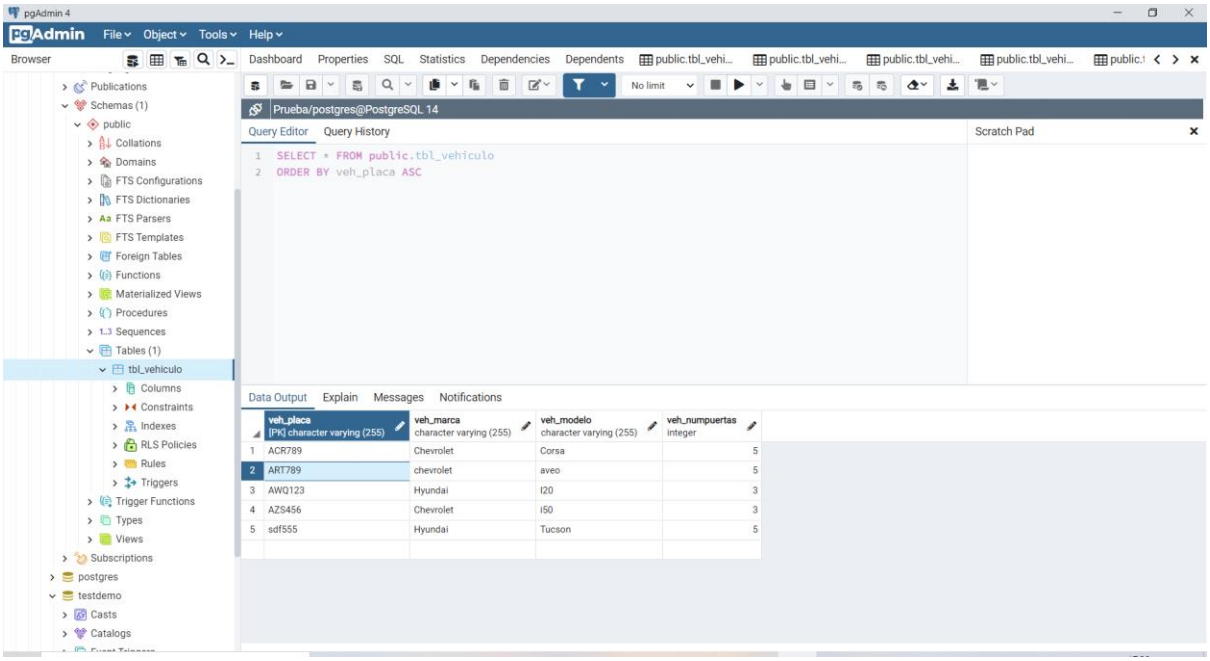
Funcionamiento del proyecto

- **INGRESO DE DATOS**

- **INGRESO EN LA BASE DE DATOS**

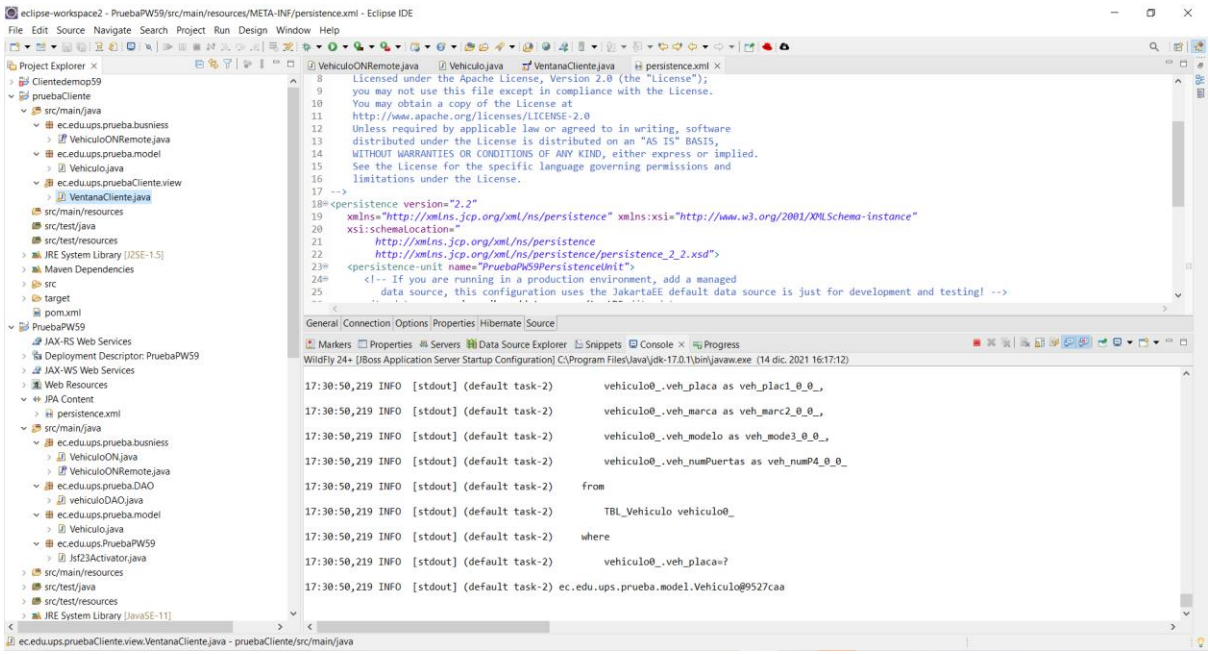
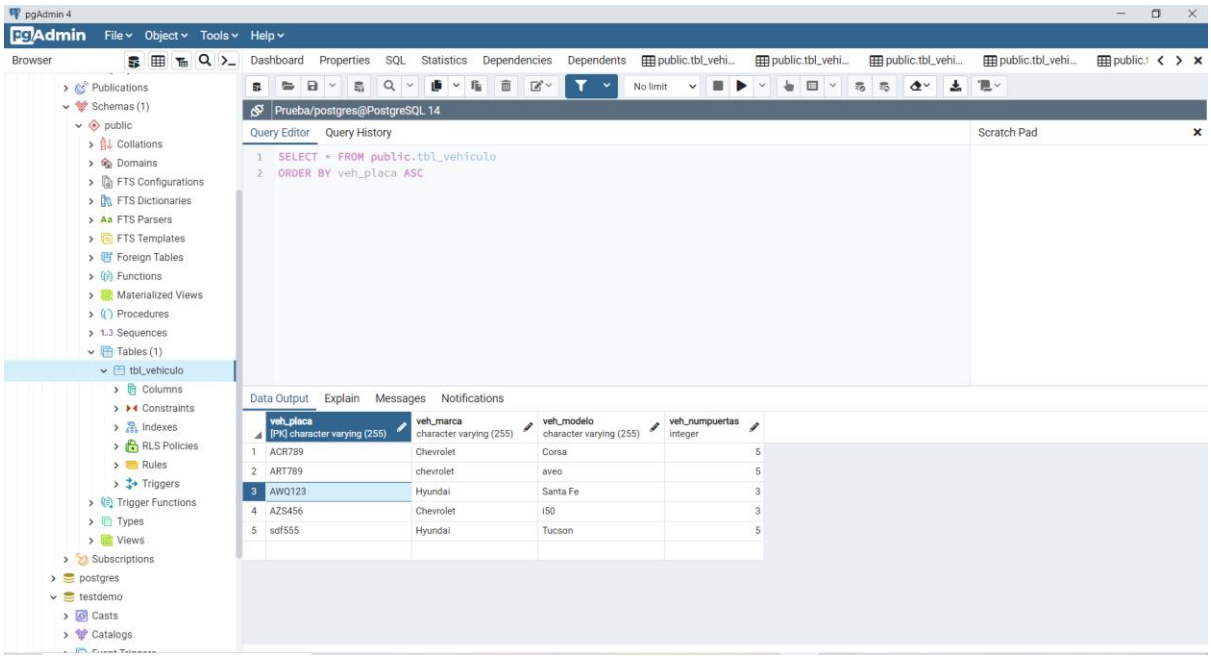


- **COMPROBACION EN LA BASE DE DATOS**



-MODIFICACION DEL AUTO

-SENTENCIA DE LA BASE DE DATOS



-COMPROBACION EN LA BASE DE DATOS