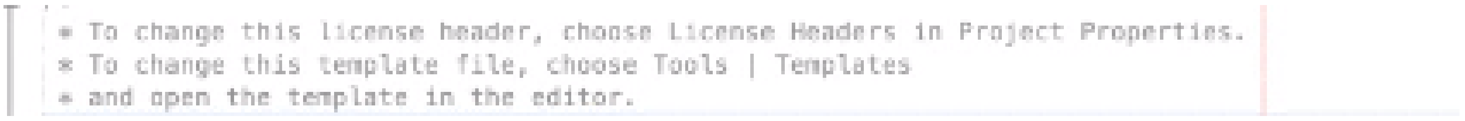
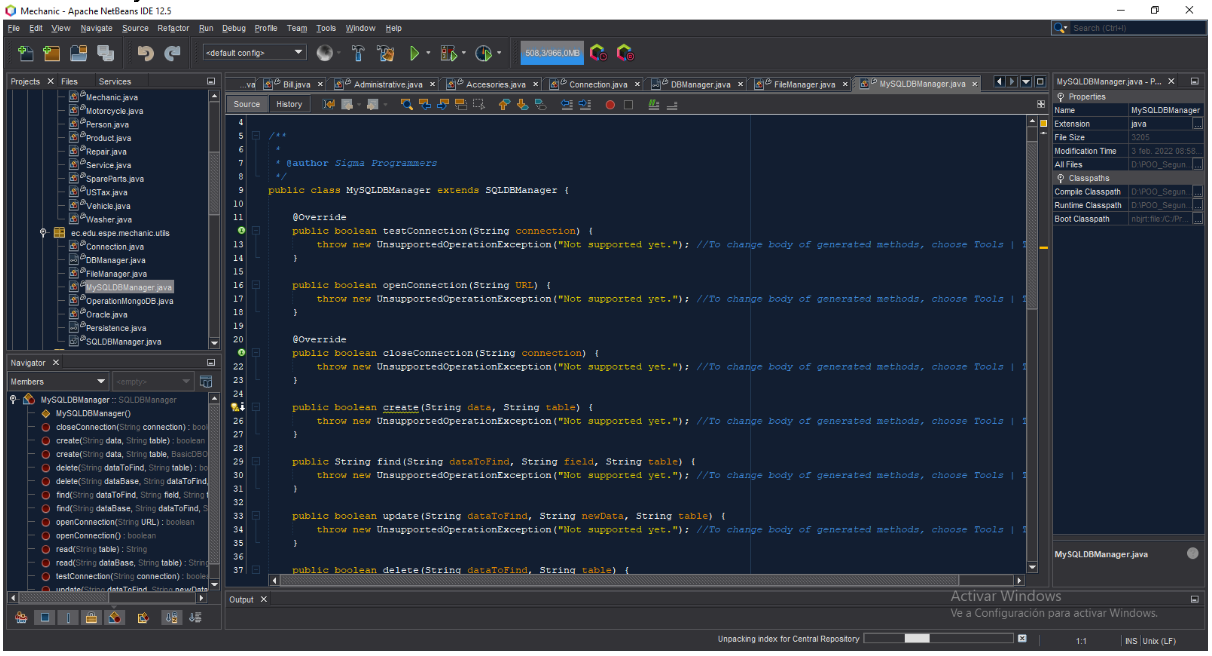
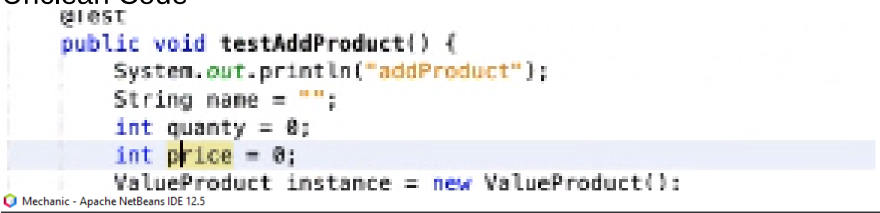
TEAM 1:
Salma Villegas
Alexander Ruano
Daniel Lincango
Mateo Maldonado
Leandro Quinga
Joel Zeas

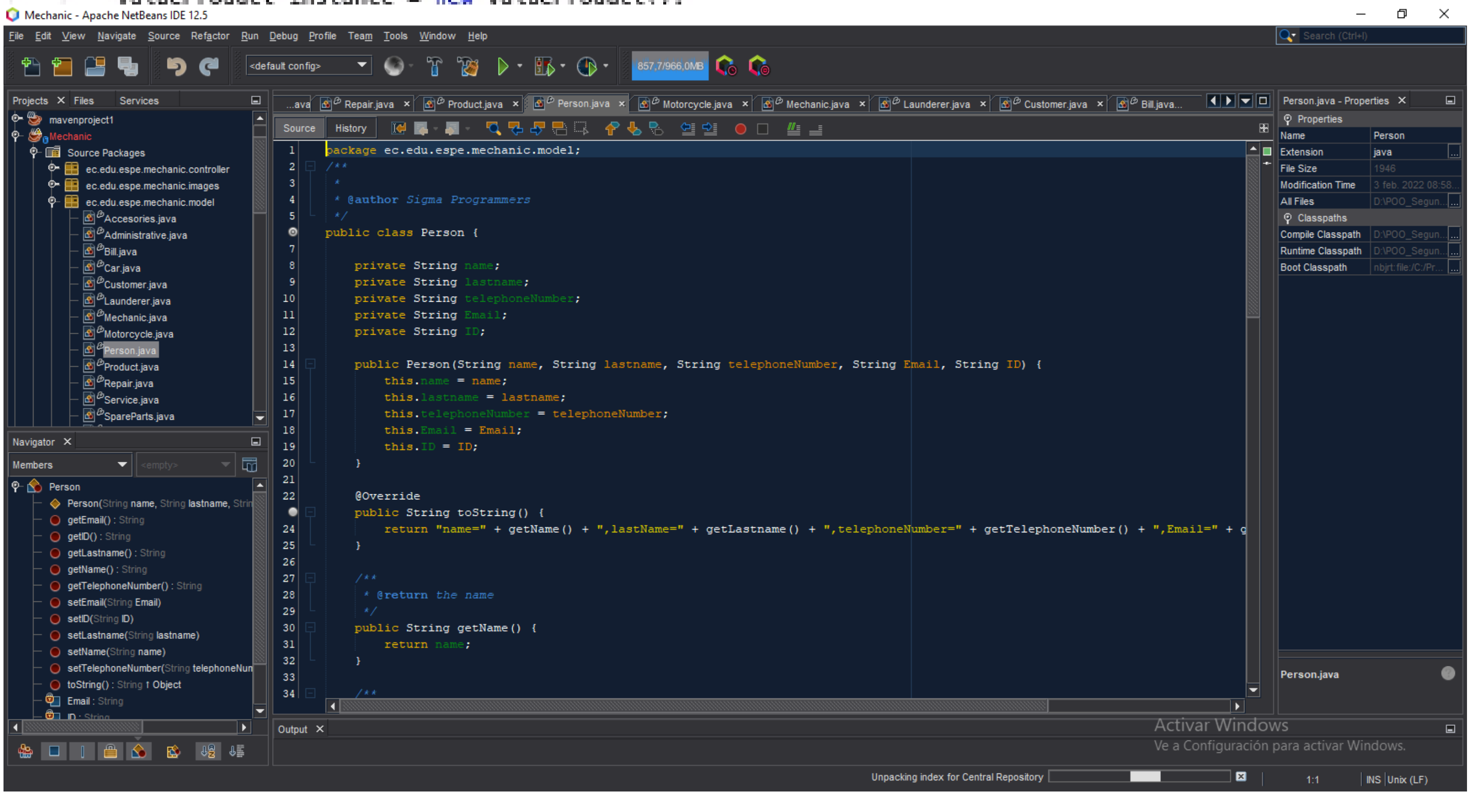

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
```

Innecesary comments, unclean code

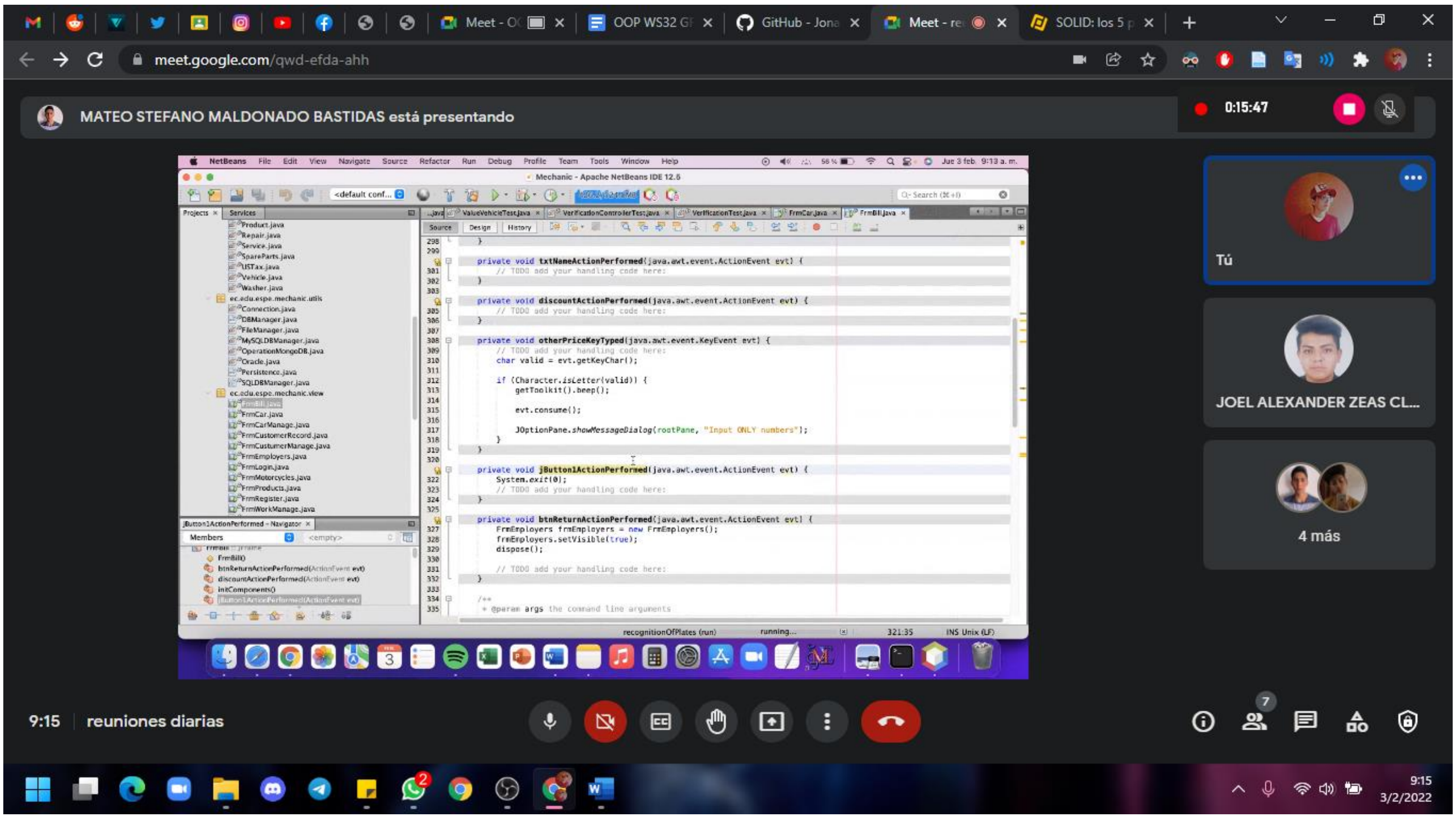

```java
/**
 *
 * @author Sigma Programmers
 */
public class MySQLDBManager extends SQLDBManager {

    @Override
    public boolean testConnection(String connection) {
        throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | T
    }

    public boolean openConnection(String URL) {
        throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | T
    }

    @Override
    public boolean closeConnection(String connection) {
        throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | T
    }

    public boolean create(String data, String table) {
        throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | T
    }

    public String find(String dataToFind, String field, String table) {
        throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | T
    }

    public boolean update(String dataToFind, String newData, String table) {
        throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | T
    }

    public boolean delete(String dataToFind, String table) {
```

Unclean Code


```java
    @Test
    public void testAddProduct() {
        System.out.println("addProduct");
        String name = "";
        int quanty = 0;
        int price = 0;
        ValueProduct instance = new ValueProduct();
```


```java
package ec.edu.espe.mechanic.model;
/**
 *
 * @author Sigma Programmers
 */
public class Person {

    private String name;
    private String lastname;
    private String telephoneNumber;
    private String Email;
    private String ID;

    public Person(String name, String lastname, String telephoneNumber, String Email, String ID) {
        this.name = name;
        this.lastname = lastname;
        this.telephoneNumber = telephoneNumber;
        this.Email = Email;
        this.ID = ID;
    }

    @Override
    public String toString() {
        return "name=" + getName() + ",lastName=" + getLastname() + ",telephoneNumber=" + getTelephoneNumber() + ",Email=" + g
    }

    /**
     * @return the name
     */
    public String getName() {
        return name;
    }

    /**
```

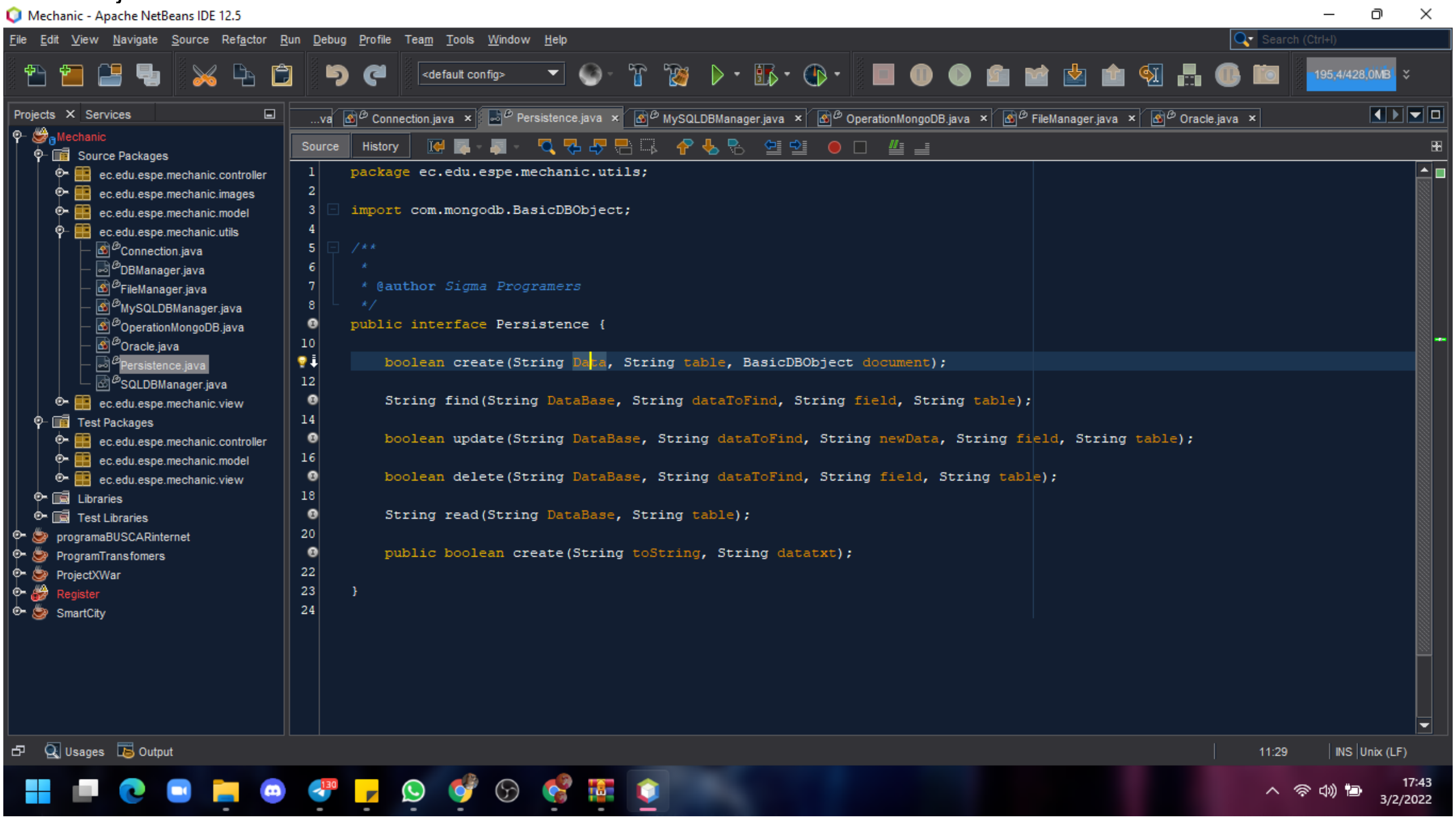
do not use the format Camelcase


MATEO STEFANO MALDONADO BASTIDAS está presentando

```java
    }

    private void txtNameActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void discountActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void otherPriceKeyTyped(java.awt.event.KeyEvent evt) {
        // TODO add your handling code here:
        char valid = evt.getKeyChar();

        if (Character.isLetter(valid)) {
            getToolkit().beep();

            evt.consume();

            JOptionPane.showMessageDialog(rootPane, "Input ONLY numbers");
        }
    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        System.exit(0);
        // TODO add your handling code here:
    }

    private void btnReturnActionPerformed(java.awt.event.ActionEvent evt) {
        FrmEmployers frmEmployers = new FrmEmployers();
        frmEmployers.setVisible(true);
        dispose();

        // TODO add your handling code here:
    }

    /**
     * @param args the command line arguments
```

9:15 reuniones diarias

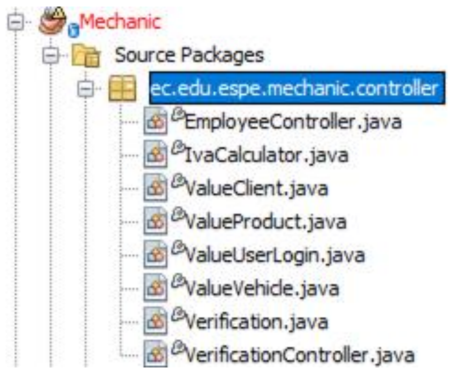
Refactor jbutton


Mechanic - Apache NetBeans IDE 12.5

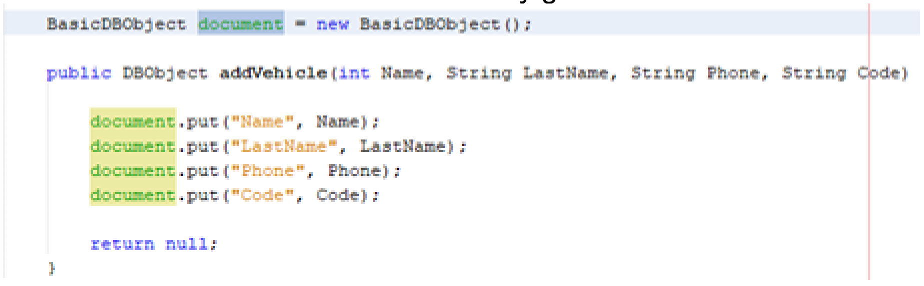
```java
package ec.edu.espe.mechanic.utils;

import com.mongodb.BasicDBObject;

/**
 *
 * @author Sigma Programers
 */
public interface Persistence {

    boolean create(String Data, String table, BasicDBObject document);

    String find(String DataBase, String dataToFind, String field, String table);

    boolean update(String DataBase, String dataToFind, String newData, String field, String table);

    boolean delete(String DataBase, String dataToFind, String field, String table);

    String read(String DataBase, String table);

    public boolean create(String toString, String datatxt);

}
```

Don't use camelcase

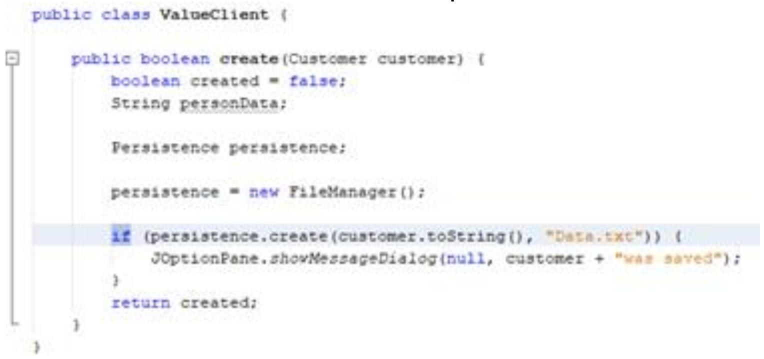
- Very redundant class names, difficult to differentiate


```
Mechanic
  Source Packages
    ec.edu.espe.mechanic.controller
      EmployeeController.java
      IvaCalculator.java
      ValueClient.java
      ValueProduct.java
      ValueUserLogin.java
      ValueVehicle.java
      Verification.java
      VerificationController.java
```

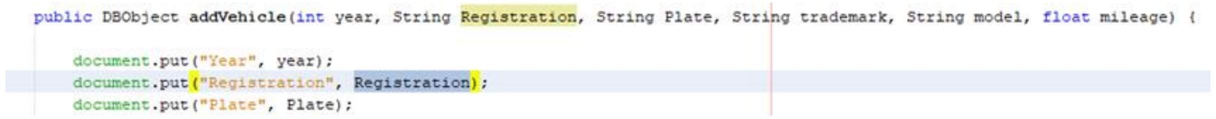
- **Single ResponsibilityPrinciples:** there is no relationship between the "document" variable and the "addVehicle" class. It is also a very general term.


```java
BasicDBObject document = new BasicDBObject();

public DBObject addVehicle(int Name, String LastName, String Phone, String Code)

    document.put("Name", Name);
    document.put("LastName", LastName);
    document.put("Phone", Phone);
    document.put("Code", Code);

    return null;
}
```

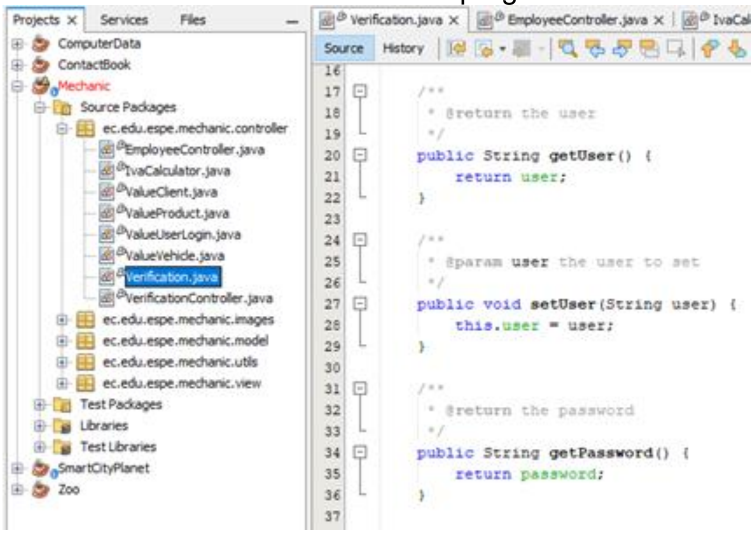
- Use an else variable that replaces the return and returns an actual data


```java
public class ValueClient {

    public boolean create(Customer customer) {
        boolean created = false;
        String personData;

        Persistence persistence;

        persistence = new FileManager();

        if (persistence.create(customer.toString(), "Data.txt")) {
            JOptionPane.showMessageDialog(null, customer + "was saved");
        }
        return created;
    }
}
```

- Use lower case for variables

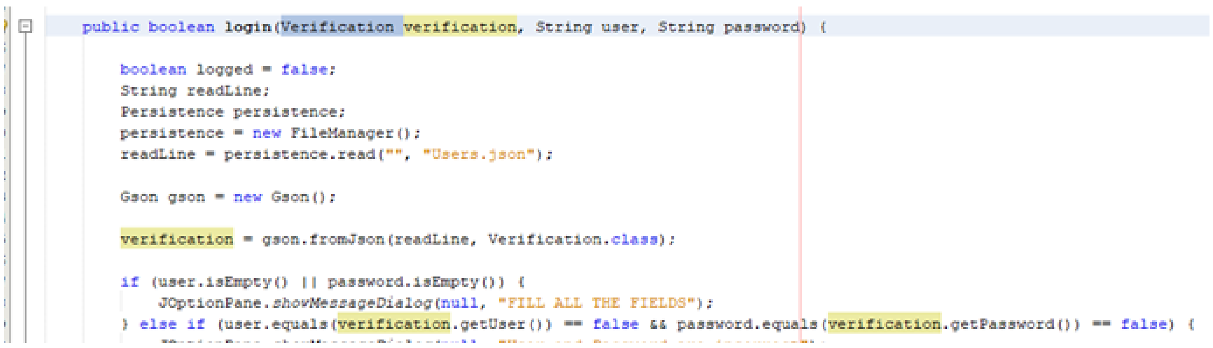

```java
public DBObject addVehicle(int year, String Registration, String Plate, String trademark, String model, float mileage) {

    document.put("Year", year);
    document.put("Registration", Registration);
    document.put("Plate", Plate);
```

- Getters and setters must be programmed in the ".model" package, no in the ".controller" package



```java
    /**
     * @return the user
     */
    public String getUser() {
        return user;
    }

    /**
     * @param user the user to set
     */
    public void setUser(String user) {
        this.user = user;
    }

    /**
     * @return the password
     */
    public String getPassword() {
        return password;
    }
```

- **Open/Closed Principle:** Very general variable and method names, also the user validation programming is usually done in the .view package and it must be open to extend it



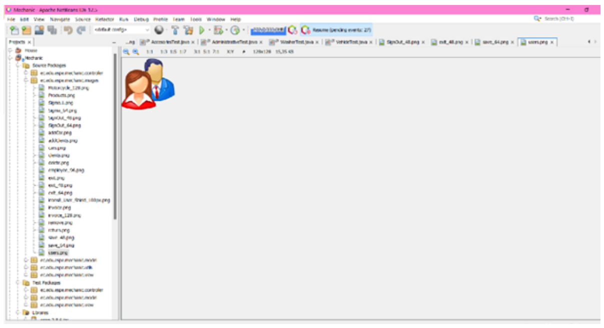
```
public boolean login(Verification verification, String user, String password) {

    boolean logged = false;
    String readLine;
    Persistence persistence;
    persistence = new FileManager();
    readLine = persistence.read("", "Users.json");

    Gson gson = new Gson();

    verification = gson.fromJson(readLine, Verification.class);

    if (user.isEmpty() || password.isEmpty()) {
        JOptionPane.showMessageDialog(null, "FILL ALL THE FIELDS");
    } else if (user.equals(verification.getUser()) == false && password.equals(verification.getPassword()) == false) {
```

Source Packages

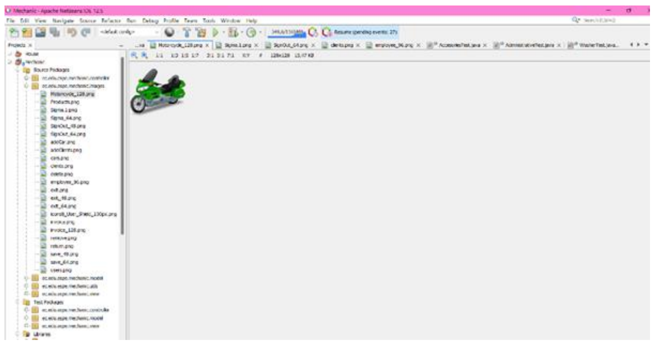
Package: ec.edu.espe.mechanic.images

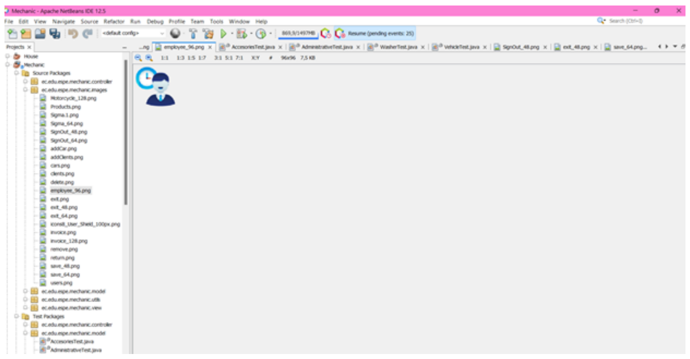
Nothing to mention, each class has only one image.

Used Principles: Single Responsibility Principle (each class have only one image)



-



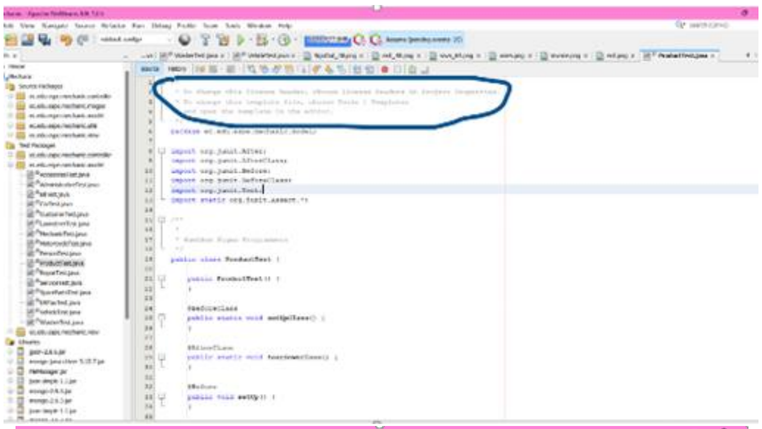


Test Packages

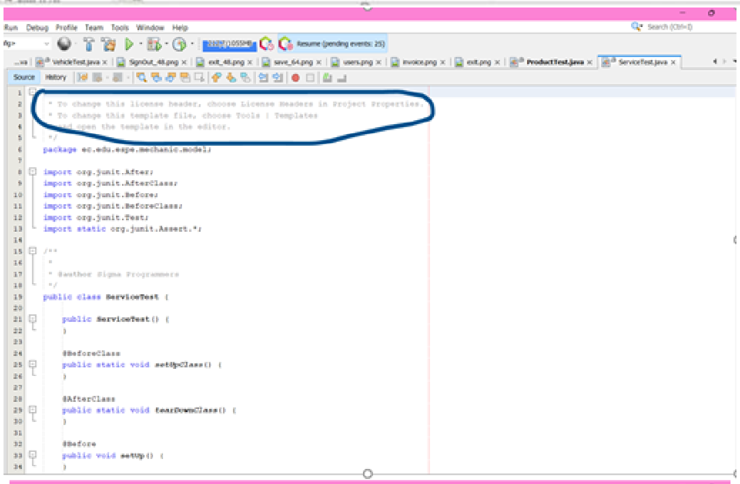
Package: ec.edu.espe.mechanic.model

Unnecessary comments, unclean code,

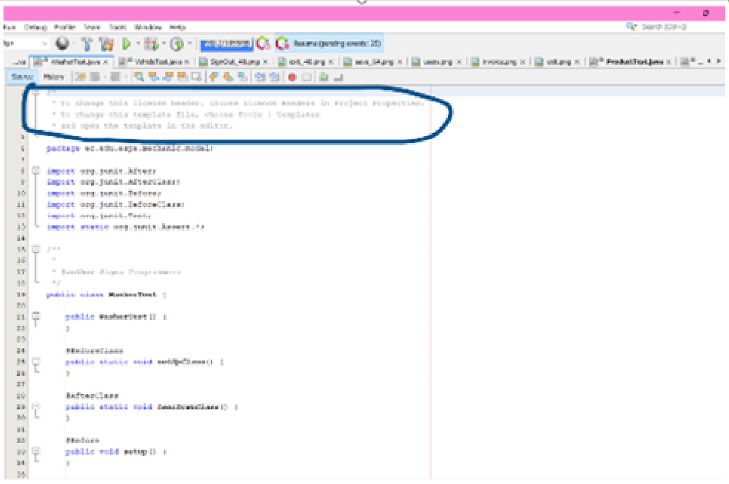
Used Principles: Single Responsibility Principle (This package is being used for testing only, so each class has a unique reason)



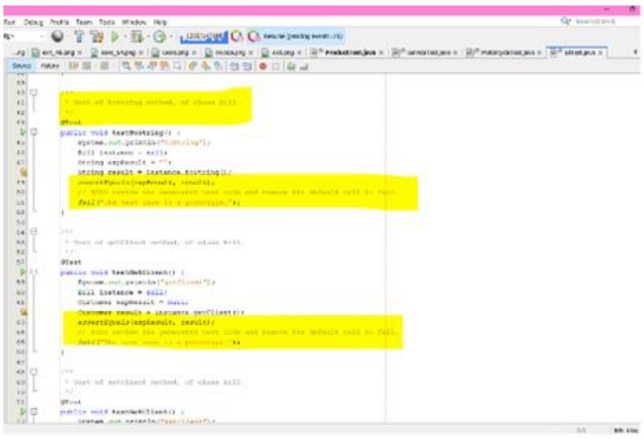
```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.espe.mechanic.model;

import org.junit.After;
import org.junit.AfterClass;
import org.junit.Before;
import org.junit.BeforeClass;
import org.junit.Test;
import static org.junit.Assert.*;

/**
 *
 * @author Sigma Programmers
 */
public class ProductTest {

    public ProductTest() {
    }

    @BeforeClass
    public static void setUpClass() {
    }

    @AfterClass
    public static void tearDownClass() {
    }

    @Before
    public void setUp() {
    }
```



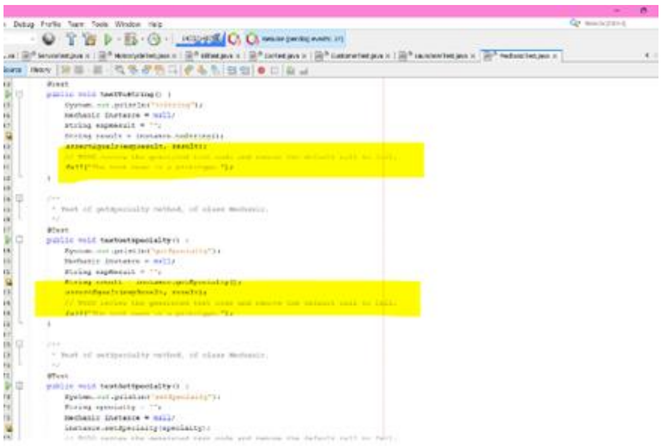
```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.espe.mechanic.model;

import org.junit.After;
import org.junit.AfterClass;
import org.junit.Before;
import org.junit.BeforeClass;
import org.junit.Test;
import static org.junit.Assert.*;

/**
 *
 * @author Sigma Programmers
 */
public class ServiceTest {

    public ServiceTest() {
    }

    @BeforeClass
    public static void setUpClass() {
    }

    @AfterClass
    public static void tearDownClass() {
    }

    @Before
    public void setUp() {
    }
```



```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.espe.mechanic.model;

import org.junit.After;
import org.junit.AfterClass;
import org.junit.Before;
import org.junit.BeforeClass;
import org.junit.Test;
import static org.junit.Assert.*;

/**
 *
 * @author Sigma Programmers
 */
public class WasherTest {

    public WasherTest() {
    }

    @BeforeClass
    public static void setUpClass() {
    }

    @AfterClass
    public static void tearDownClass() {
    }

    @Before
    public void setUp() {
    }
```



```java
    /**
     * Test of toString method, of class Bill.
     */
    @Test
    public void testToString() {
        System.out.println("toString");
        Bill instance = null;
        String expResult = "";
        String result = instance.toString();
        assertEquals(expResult, result);
        // TODO review the generated test code and remove the default call to fail.
        fail("The test case is a prototype.");
    }

    /**
     * Test of getClient method, of class Bill.
     */
    @Test
    public void testGetClient() {
        System.out.println("getClient");
        Bill instance = null;
        Customer expResult = null;
        Customer result = instance.getClient();
        assertEquals(expResult, result);
        // TODO review the generated test code and remove the default call to fail.
        fail("The test case is a prototype.");
    }

    /**
     * Test of setClient method, of class Bill.
     */
    @Test
    public void testSetClient() {
        System.out.println("setClient");
```



```java
    @Test
    public void testToString() {
        System.out.println("toString");
        Mechanic instance = null;
        String expResult = "";
        String result = instance.toString();
        assertEquals(expResult, result);
        // TODO review the generated test code and remove the default call to fail.
        fail("The test case is a prototype.");
    }

    /**
     * Test of getSpecialty method, of class Mechanic.
     */
    @Test
    public void testGetSpecialty() {
        System.out.println("getSpecialty");
        Mechanic instance = null;
        String expResult = "";
        String result = instance.getSpecialty();
        assertEquals(expResult, result);
        // TODO review the generated test code and remove the default call to fail.
        fail("The test case is a prototype.");
    }

    /**
     * Test of setSpecialty method, of class Mechanic.
     */
    @Test
    public void testSetSpecialty() {
        System.out.println("setSpecialty");
        String specialty = "";
        Mechanic instance = null;
        instance.setSpecialty(specialty);
        // TODO review the generated test code and remove the default call to fail.
```