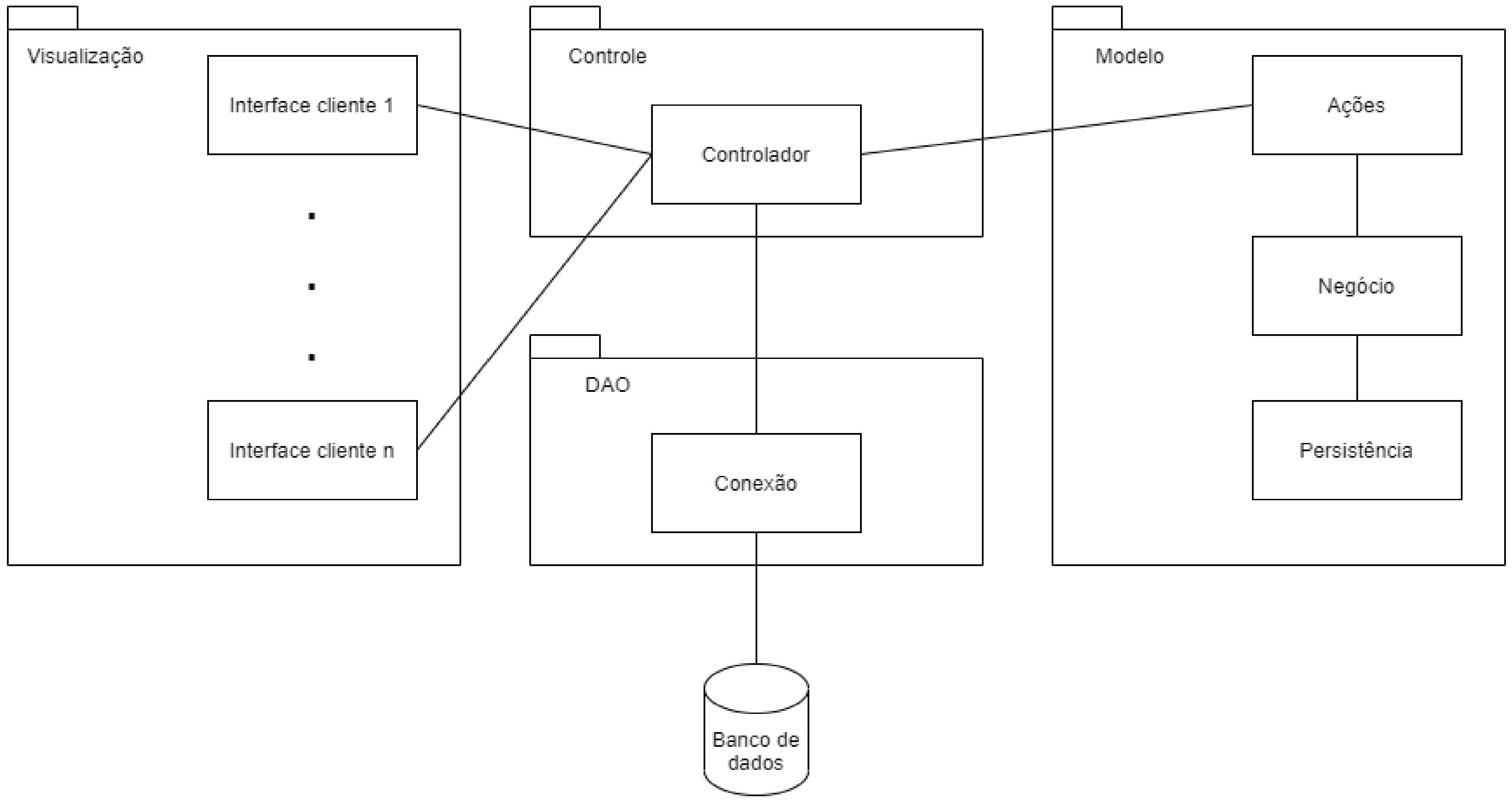
Prj MVC-DAO

Carlos Arruda Baltazar

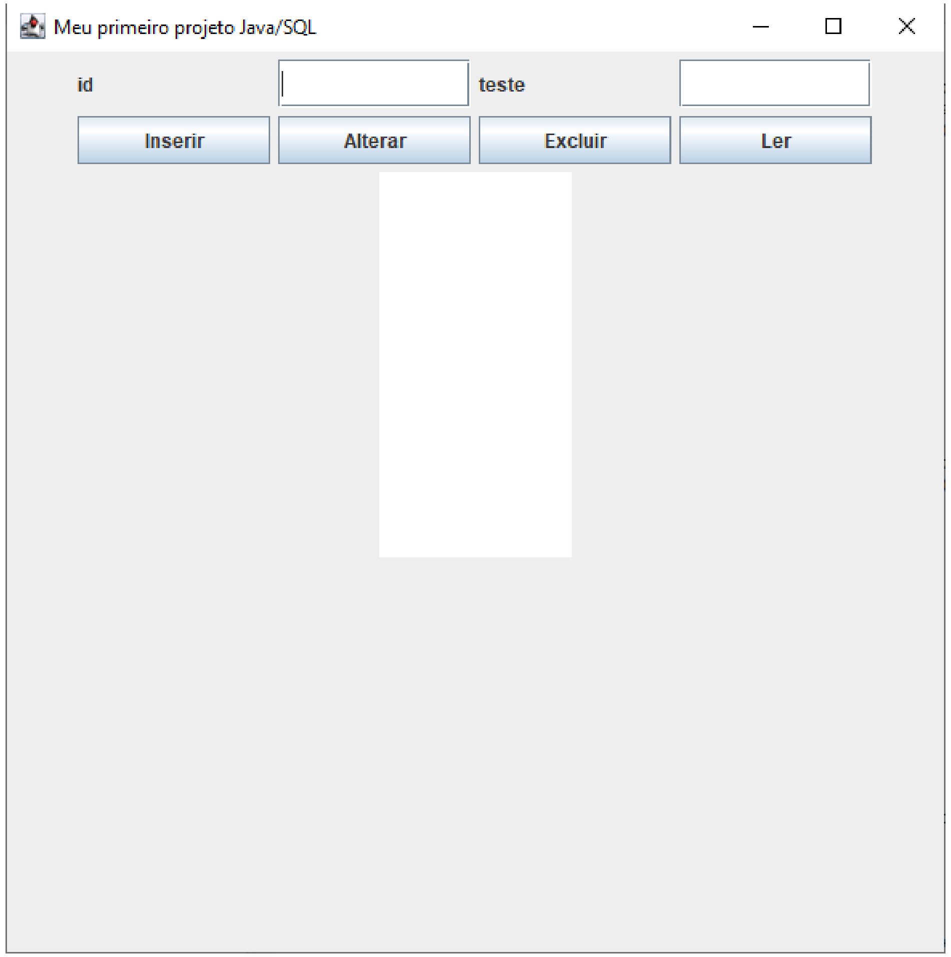
UNIP – Cidade Universitária

UNIP UNIVERSIDADE PAULISTA

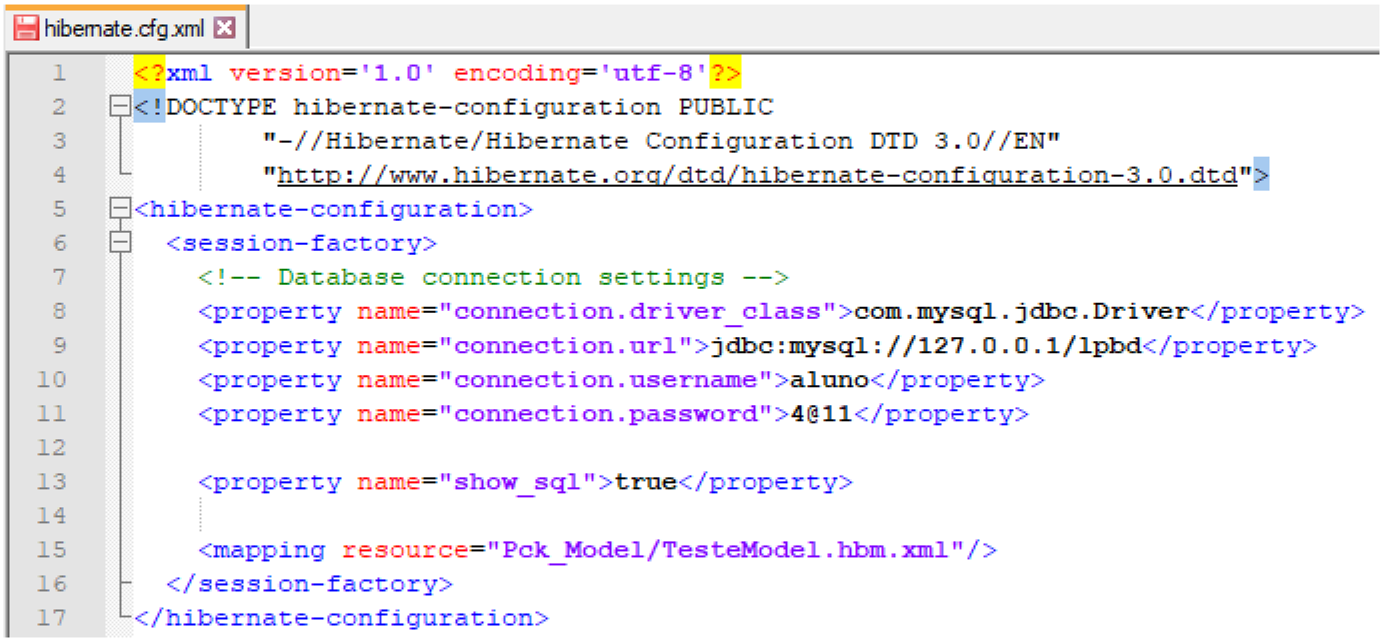
Prj MVC-DAO



Prj MVC-DAO


Meu primeiro projeto Java/SQL
id
teste
Inserir
Alterar
Excluir
Ler

Prj MVC-DAO

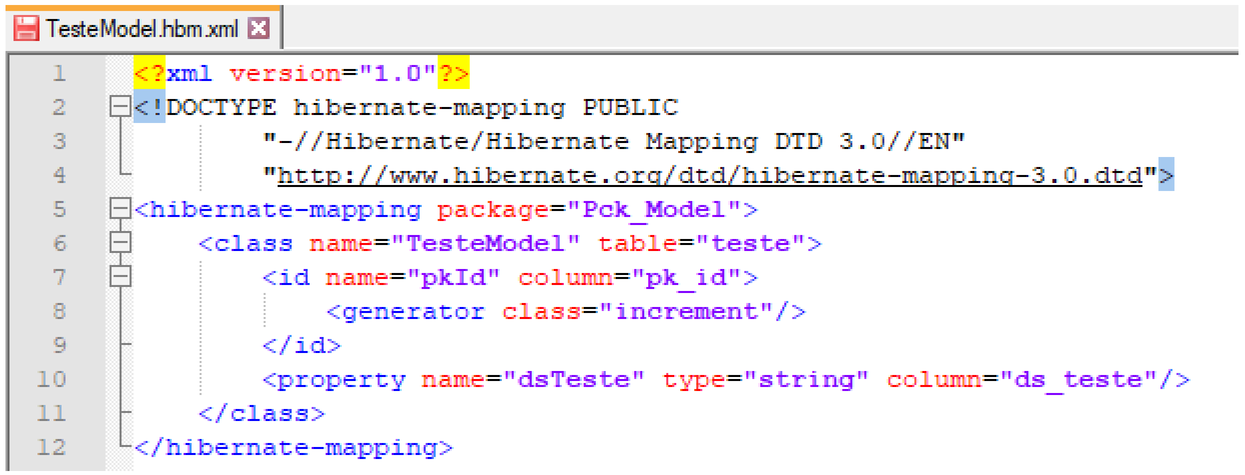


hibernate.cfg.xml

```
<?xml version='1.0' encoding='utf-8'?>
<!DOCTYPE hibernate-configuration PUBLIC
        "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
        "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
  <session-factory>
    <!-- Database connection settings -->
    <property name="connection.driver_class">com.mysql.jdbc.Driver</property>
    <property name="connection.url">jdbc:mysql://127.0.0.1/lpbd</property>
    <property name="connection.username">aluno</property>
    <property name="connection.password">4@11</property>

    <property name="show_sql">true</property>

    <mapping resource="Pck_Model/TesteModel.hbm.xml"/>
  </session-factory>
</hibernate-configuration>
```

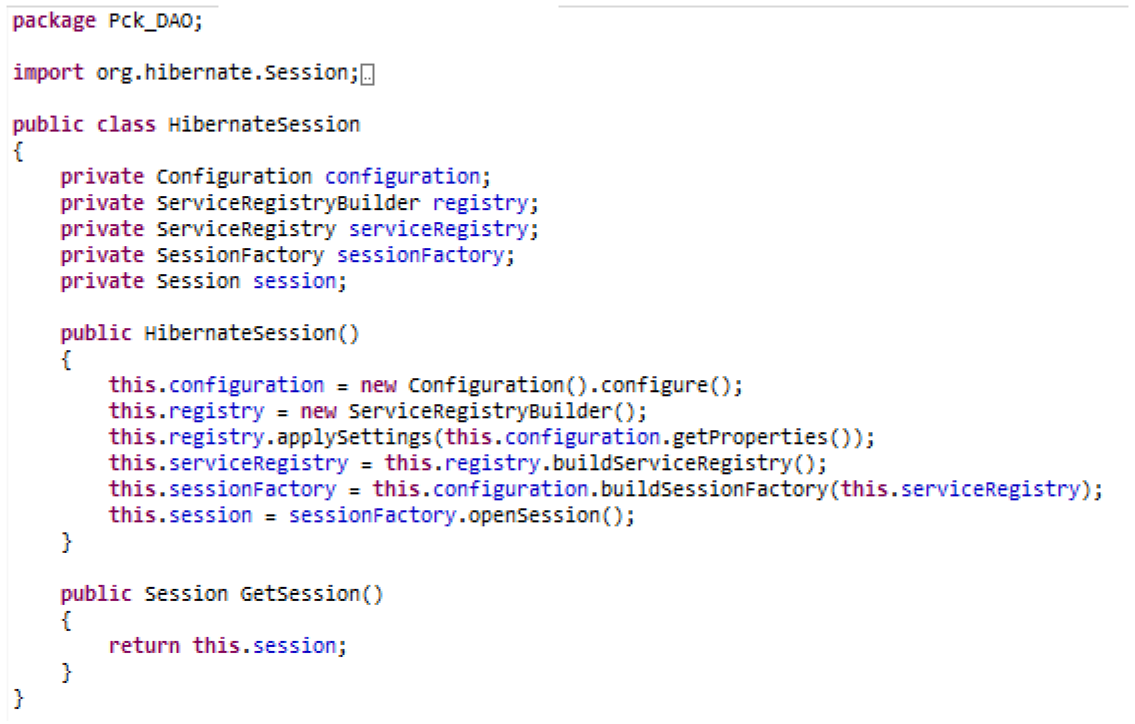
Model

```java
package Pck_Model;

public class TesteModel
{
    private int pkId;
    private String dsTeste;

    public TesteModel()
    {

    }
    public TesteModel(int id, String teste)
    {
        this.pkId = id;
        this.dsTeste = teste;
    }

    public TesteModel(String teste)
    {
        this.dsTeste = teste;
    }

    public TesteModel(int id)
    {
        this.pkId = id;
    }
    public int getPkId() {
        return pkId;
    }
    public void setPkId(int pkId) {
        this.pkId = pkId;
    }
    public String getDsTeste() {
        return dsTeste;
    }
    public void setDsTeste(String dsTeste) {
        this.dsTeste = dsTeste;
    }
}
```



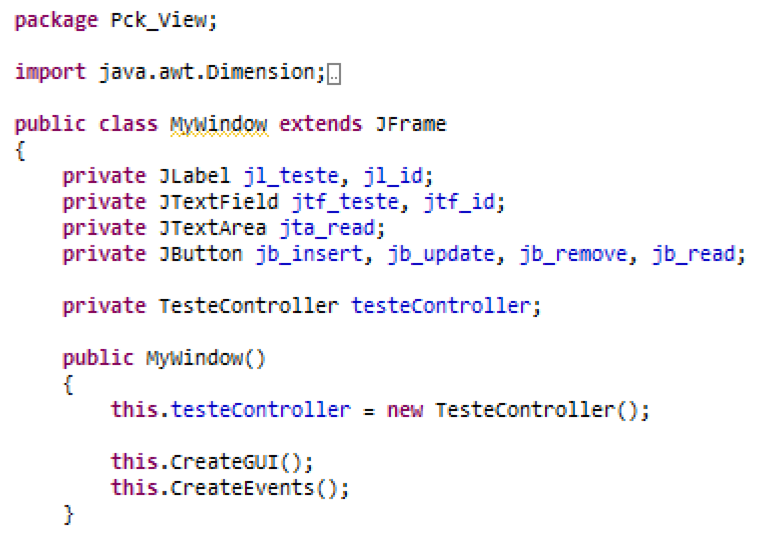
```xml
<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping PUBLIC
        "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
        "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">
<hibernate-mapping package="Pck_Model">
    <class name="TesteModel" table="teste">
        <id name="pkId" column="pk_id">
            <generator class="increment"/>
        </id>
        <property name="dsTeste" type="string" column="ds_teste"/>
    </class>
</hibernate-mapping>
```

DAO

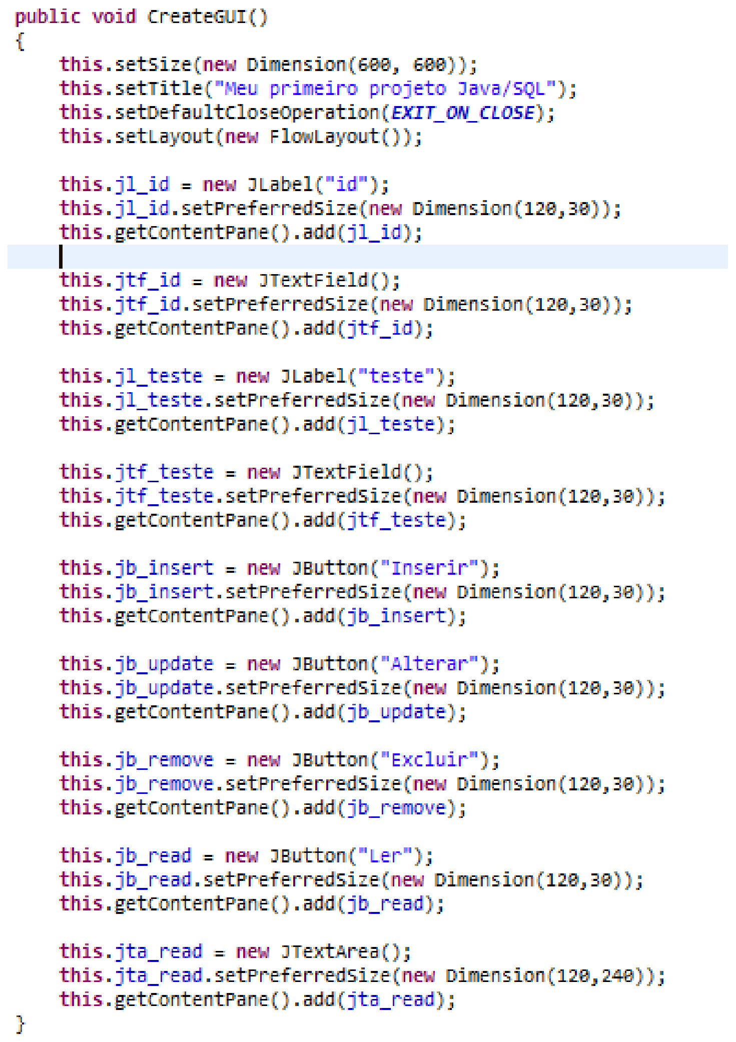


```
package Pck_DAO;

import org.hibernate.Session;

public class HibernateSession
{
    private Configuration configuration;
    private ServiceRegistryBuilder registry;
    private ServiceRegistry serviceRegistry;
    private SessionFactory sessionFactory;
    private Session session;

    public HibernateSession()
    {
        this.configuration = new Configuration().configure();
        this.registry = new ServiceRegistryBuilder();
        this.registry.applySettings(this.configuration.getProperties());
        this.serviceRegistry = this.registry.buildServiceRegistry();
        this.sessionFactory = this.configuration.buildSessionFactory(this.serviceRegistry);
        this.session = sessionFactory.openSession();
    }

    public Session GetSession()
    {
        return this.session;
    }
}
```

View

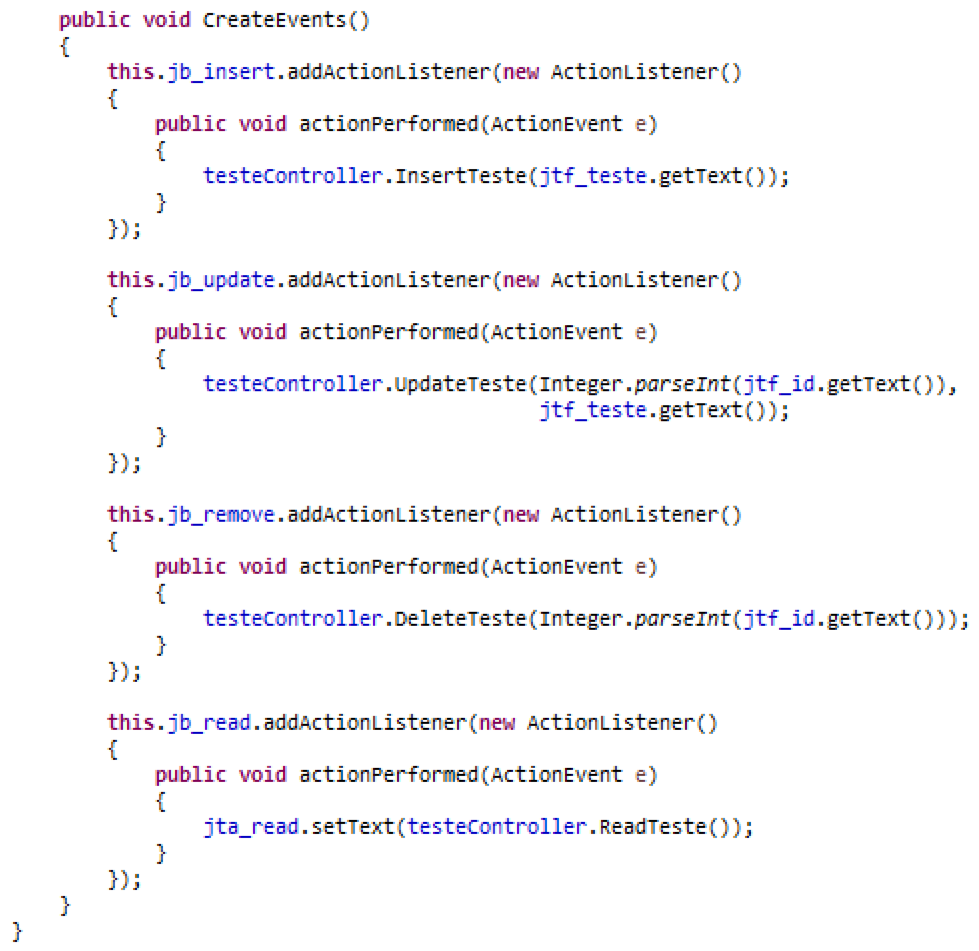


```
package Pck_View;

import java.awt.Dimension;

public class MyWindow extends JFrame
{
    private JLabel jl_teste, jl_id;
    private JTextField jtf_teste, jtf_id;
    private JTextArea jta_read;
    private JButton jb_insert, jb_update, jb_remove, jb_read;

    private TesteController testeController;

    public MyWindow()
    {
        this.testeController = new TesteController();

        this.CreateGUI();
        this.CreateEvents();
    }
```

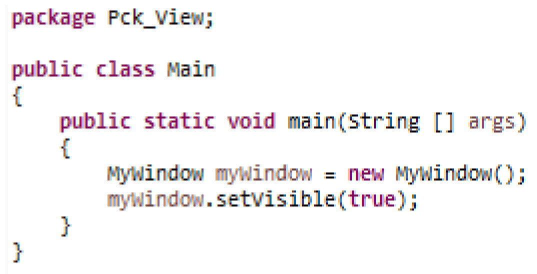


```
public void CreateGUI()
{
    this.setSize(new Dimension(600, 600));
    this.setTitle("Meu primeiro projeto Java/SQL");
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);
    this.setLayout(new FlowLayout());

    this.jl_id = new JLabel("id");
    this.jl_id.setPreferredSize(new Dimension(120,30));
    this.getContentPane().add(jl_id);

    this.jtf_id = new JTextField();
    this.jtf_id.setPreferredSize(new Dimension(120,30));
    this.getContentPane().add(jtf_id);

    this.jl_teste = new JLabel("teste");
    this.jl_teste.setPreferredSize(new Dimension(120,30));
    this.getContentPane().add(jl_teste);

    this.jtf_teste = new JTextField();
    this.jtf_teste.setPreferredSize(new Dimension(120,30));
    this.getContentPane().add(jtf_teste);

    this.jb_insert = new JButton("Inserir");
    this.jb_insert.setPreferredSize(new Dimension(120,30));
    this.getContentPane().add(jb_insert);

    this.jb_update = new JButton("Alterar");
    this.jb_update.setPreferredSize(new Dimension(120,30));
    this.getContentPane().add(jb_update);

    this.jb_remove = new JButton("Excluir");
    this.jb_remove.setPreferredSize(new Dimension(120,30));
    this.getContentPane().add(jb_remove);

    this.jb_read = new JButton("Ler");
    this.jb_read.setPreferredSize(new Dimension(120,30));
    this.getContentPane().add(jb_read);

    this.jta_read = new JTextArea();
    this.jta_read.setPreferredSize(new Dimension(120,240));
    this.getContentPane().add(jta_read);
}
```

View

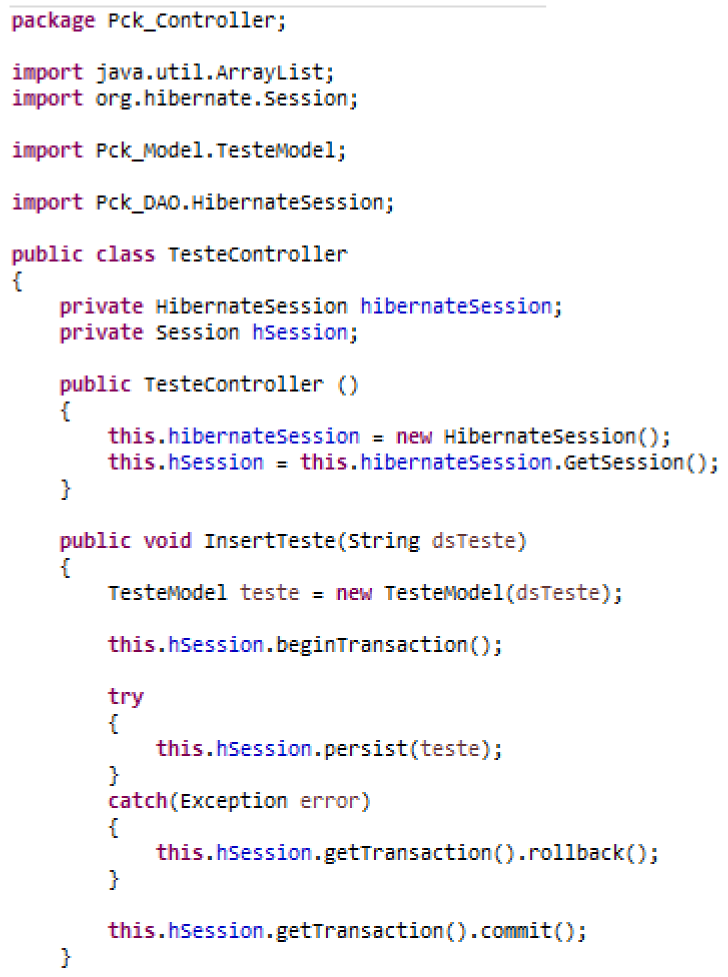


```java
        public void CreateEvents()
        {
            this.jb_insert.addActionListener(new ActionListener()
            {
                public void actionPerformed(ActionEvent e)
                {
                    testeController.InsertTeste(jtf_teste.getText());
                }
            });

            this.jb_update.addActionListener(new ActionListener()
            {
                public void actionPerformed(ActionEvent e)
                {
                    testeController.UpdateTeste(Integer.parseInt(jtf_id.getText()),
                                                jtf_teste.getText());
                }
            });

            this.jb_remove.addActionListener(new ActionListener()
            {
                public void actionPerformed(ActionEvent e)
                {
                    testeController.DeleteTeste(Integer.parseInt(jtf_id.getText()));
                }
            });

            this.jb_read.addActionListener(new ActionListener()
            {
                public void actionPerformed(ActionEvent e)
                {
                    jta_read.setText(testeController.ReadTeste());
                }
            });
        }
    }
```

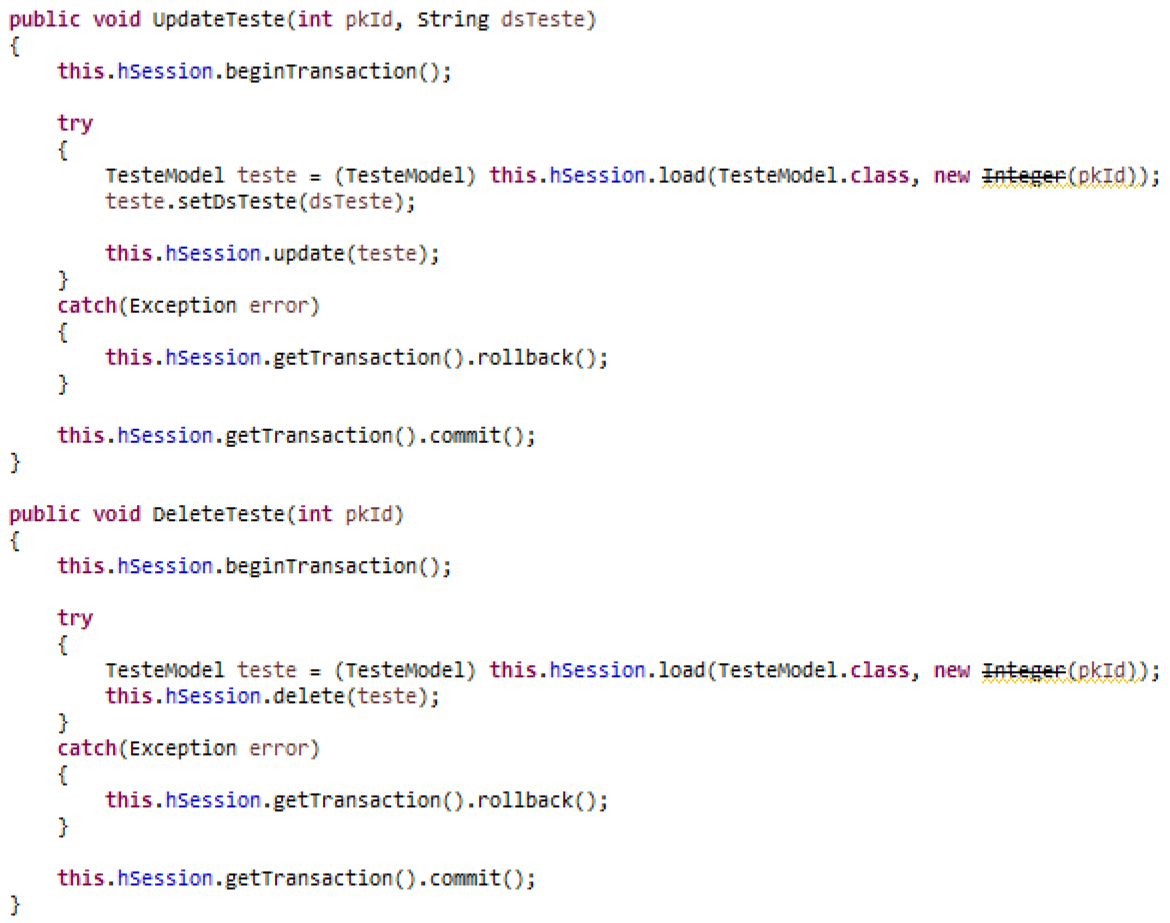


```java
package Pck_View;

public class Main
{
    public static void main(String [] args)
    {
        MyWindow myWindow = new MyWindow();
        myWindow.setVisible(true);
    }
}
```

Controller

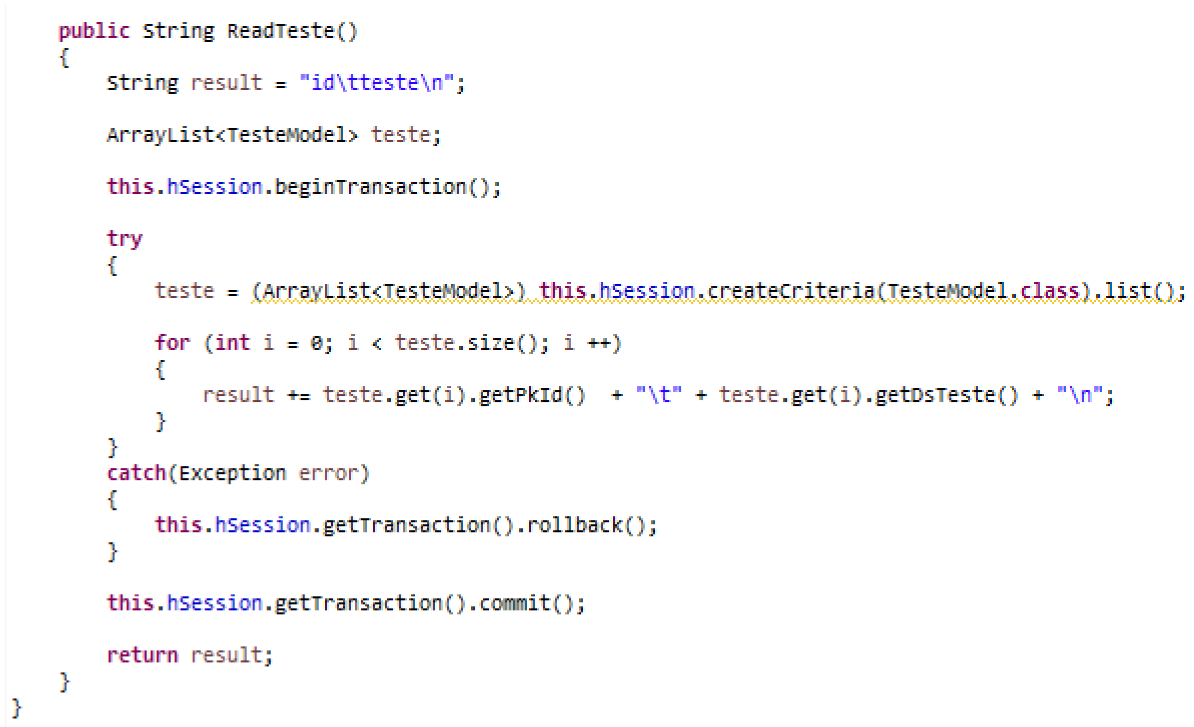


```
package Pck_Controller;

import java.util.ArrayList;
import org.hibernate.Session;

import Pck_Model.TesteModel;

import Pck_DAO.HibernateSession;

public class TesteController
{
    private HibernateSession hibernateSession;
    private Session hSession;

    public TesteController ()
    {
        this.hibernateSession = new HibernateSession();
        this.hSession = this.hibernateSession.GetSession();
    }

    public void InsertTeste(String dsTeste)
    {
        TesteModel teste = new TesteModel(dsTeste);

        this.hSession.beginTransaction();

        try
        {
            this.hSession.persist(teste);
        }
        catch(Exception error)
        {
            this.hSession.getTransaction().rollback();
        }

        this.hSession.getTransaction().commit();
    }
```



```
public void UpdateTeste(int pkId, String dsTeste)
{
    this.hSession.beginTransaction();

    try
    {
        TesteModel teste = (TesteModel) this.hSession.load(TesteModel.class, new ~~Integer~~(pkId));
        teste.setDsTeste(dsTeste);

        this.hSession.update(teste);
    }
    catch(Exception error)
    {
        this.hSession.getTransaction().rollback();
    }

    this.hSession.getTransaction().commit();
}

public void DeleteTeste(int pkId)
{
    this.hSession.beginTransaction();

    try
    {
        TesteModel teste = (TesteModel) this.hSession.load(TesteModel.class, new ~~Integer~~(pkId));
        this.hSession.delete(teste);
    }
    catch(Exception error)
    {
        this.hSession.getTransaction().rollback();
    }

    this.hSession.getTransaction().commit();
}
```

Controller



```
    public String ReadTeste()
    {
        String result = "id\tteste\n";

        ArrayList<TesteModel> teste;

        this.hSession.beginTransaction();

        try
        {
            teste = (ArrayList<TesteModel>) this.hSession.createCriteria(TesteModel.class).list();

            for (int i = 0; i < teste.size(); i ++)
            {
                result += teste.get(i).getPkId()  + "\t" + teste.get(i).getDsTeste() + "\n";
            }
        }
        catch(Exception error)
        {
            this.hSession.getTransaction().rollback();
        }

        this.hSession.getTransaction().commit();

        return result;
    }
}
```

UNIP
UNIVERSIDADE PAULISTA

UNICLUSTER

OBRIGADO