**Informe Inteligencia Artificial**

**Integrantes:**

* **Andrés Carrión**
* **Christian Febres**
* **Roberth Torres**

**Comunidad:**

Desarrollo de un agente inteligente destinado al entorno virtual de aprendizaje.

**Desarrollo de los agentes**

**Se crea el agente de detección que será el encargado de obtener las palabras ingresadas por teclado para esto usamos el método ACLMessage que será el encargado de enviarle lo que se ingresa por teclado al agente de búsqueda.**

/\*

\* 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 proyecto\_jade;

import jade.core.AID;

import jade.core.Agent;

import jade.core.behaviours.CyclicBehaviour;

import jade.core.behaviours.OneShotBehaviour;

import jade.lang.acl.ACLMessage;

import java.util.Scanner;

/\*\*

\*

\* @author Andy

\*/

public class Agente\_Deteccion extends Agent {

//private gui\_jade migui;

public void setup() {

//migui = new gui\_jade();

addBehaviour(new CyclicBehaviour() {

@Override

public void action() {

ACLMessage message\_rec = receive();

if (message\_rec != null) {

String valor = message\_rec.getContent();

System.out.println(valor);

} else {

block();

}

}

});

addBehaviour(new OneShotBehaviour() {

@Override

public void action() {

System.out.println("Ingrese palabra a buscar:\n");

Scanner scan = new Scanner(System.in);

String valor = scan.next();

ACLMessage message\_env = new ACLMessage(ACLMessage.REQUEST);

message\_env.setContent(valor);

message\_env.addReceiver(new AID("busqueda", AID.ISLOCALNAME));

send(message\_env);

}

});

}

}

**Se crea un agente que en este caso es el de búsqueda que será el agente encargado de de realizar la búsqueda de acuerdo al termino ingresado, realizando la conexión a la base de datos**.

/\*

\* 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 proyecto\_jade;

import jade.core.AID;

import jade.core.Agent;

import jade.core.behaviours.CyclicBehaviour;

import jade.lang.acl.ACLMessage;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.logging.Level;

import java.util.logging.Logger;

/\*\*

\*

\* @author Andy

\*/

public class Agente\_Busqueda extends Agent {

static conexiones con = new conexiones();

static Statement st;

public void setup() {

addBehaviour(new CyclicBehaviour() {

@Override

public void action() {

ACLMessage message\_reci = receive();

String contenido = "";

if (message\_reci != null) {

try {

st = con.AbrirConexion().createStatement();

String sentencia;

System.out.println("iniciando busqueda\n");

sentencia = "Select r.recurso from tieneRecurso tr, "

+ "recurso r, materia m WHERE tr.id = r.id and "

+ "tr.id\_materia = m.id\_materia AND m.materia like '" + message\_reci.getContent() + "%';";

ResultSet rs = st.executeQuery(sentencia);

while (rs.next()) {

//System.out.println(rs.getString(1) + "\n");

contenido = contenido + rs.getString(1) + "\n";

}

System.out.println("Sugerencias de busqueda\n");

} catch (ClassNotFoundException ex) {

Logger.getLogger(Agente\_Busqueda.class.getName()).log(Level.SEVERE, null, ex);

} catch (SQLException ex) {

Logger.getLogger(Agente\_Busqueda.class.getName()).log(Level.SEVERE, null, ex);

}

} else {

block();

}

ACLMessage message\_env = new ACLMessage(ACLMessage.REQUEST);

message\_env.setContent(contenido);

message\_env.addReceiver(new AID("deteccion", AID.ISLOCALNAME));

send(message\_env);

}

});

}

}

**Se crea la clase conexión en la cual se configurara los parámetros para conectar con la base de datos.**

/\*

\* 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 proyecto\_jade;

import java.sql.\*;

import java.sql.Connection;

/\*\*

\*

\* @author Andy

\*/

public class conexiones {

public Connection con;

public Connection conectar() throws SQLException, ClassNotFoundException {

String driver = "com.mysql.jdbc.Driver";

String url = "jdbc:mysql://localhost:3306/db\_agente";

Class.forName(driver);

return DriverManager.getConnection(url, "root", "");

}

public Connection AbrirConexion() throws ClassNotFoundException, SQLException {

con = conectar();

return con;

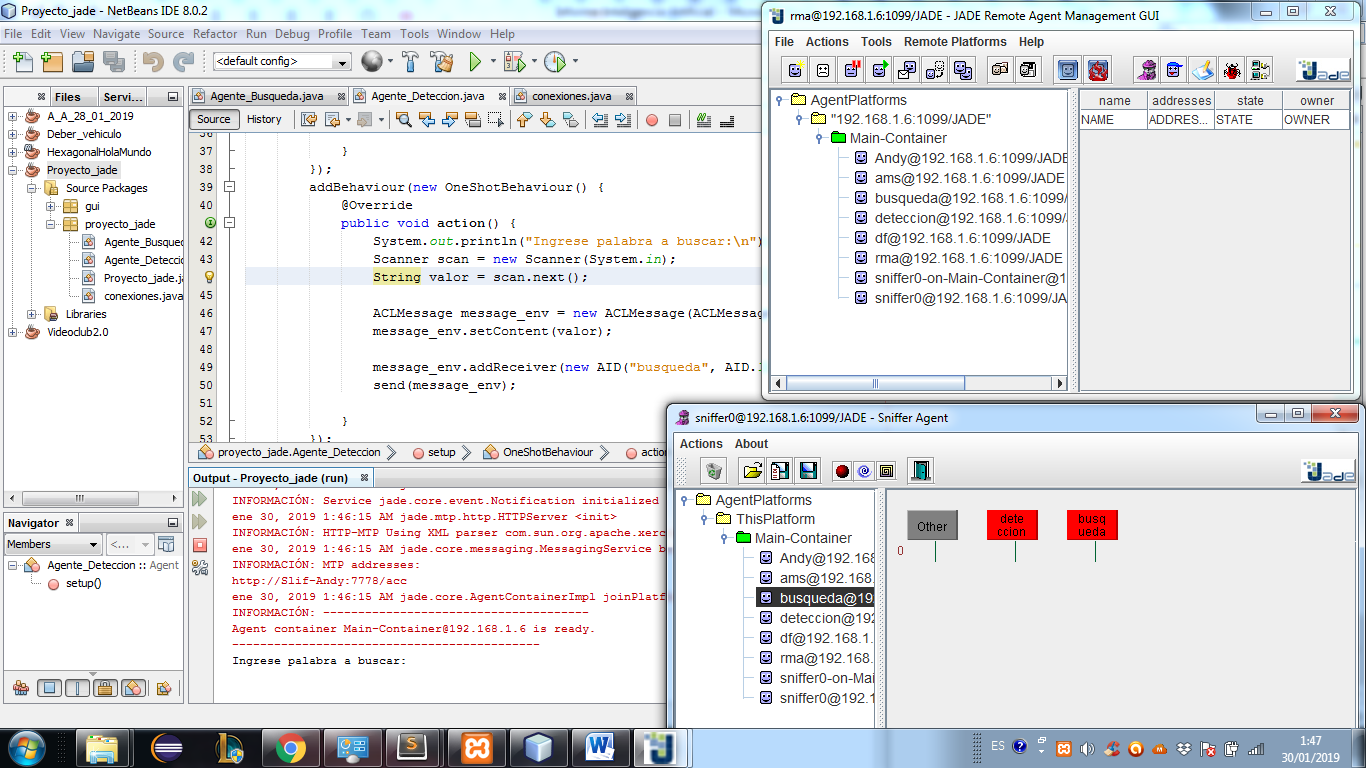
}

public void CerrarConexion() throws SQLException {

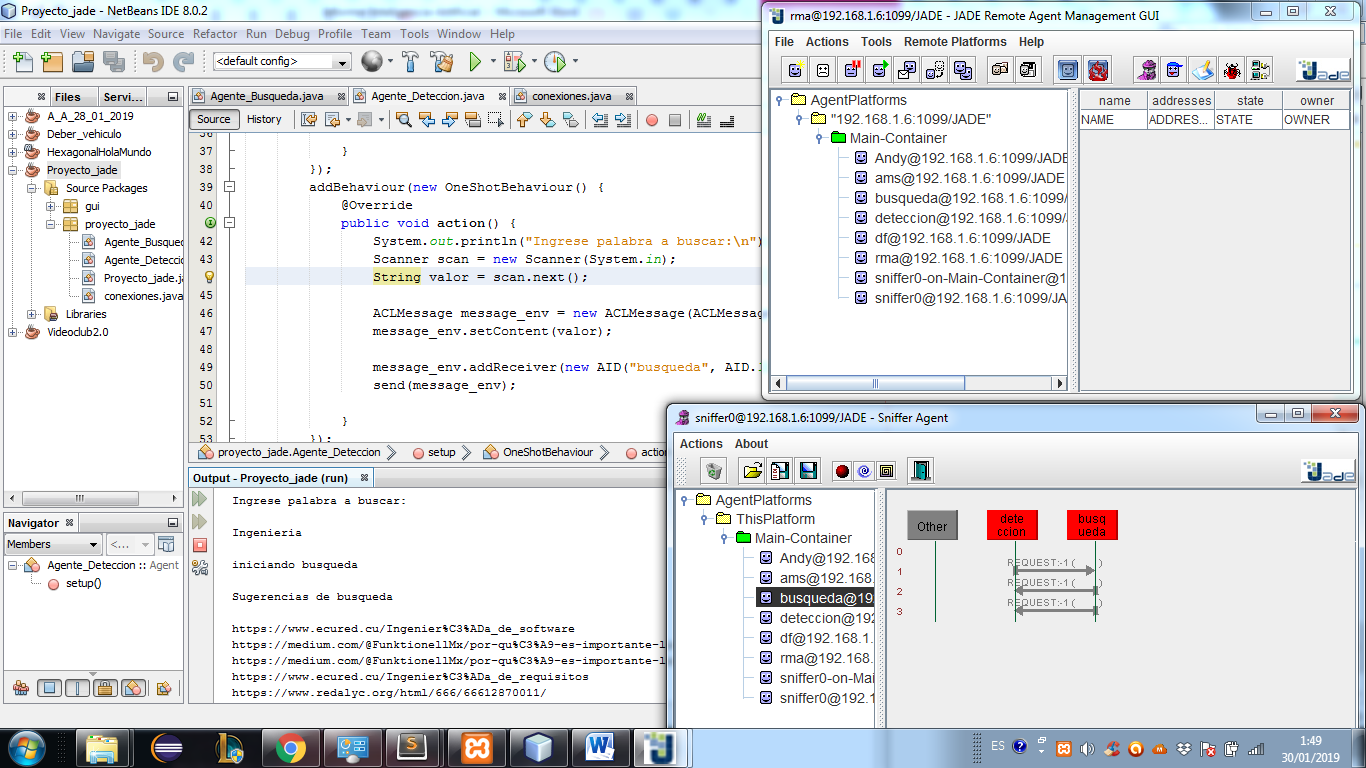
con.close();

}

}



Luego de esto observamos en la ventana de JADE el gráfico que muestra cómo están los agentes antes de la comunicación.



Observamos en la ventana de JADE el gráfico que muestra cómo están los agentes despues de la comunicación entre ellos.