

10/6/2019



Manual Técnico Juego Pokémon.

Programación



Andrés Ceballos Rodríguez

Manual Técnico Juego Pokémon.

Programación

Índice

1. Introducción	2
2. Base de Datos	2
3. Librerías.....	5
4. Requisitos e Instalación.....	5
5. Clases Java:	7
7. Bibliografía.....	60

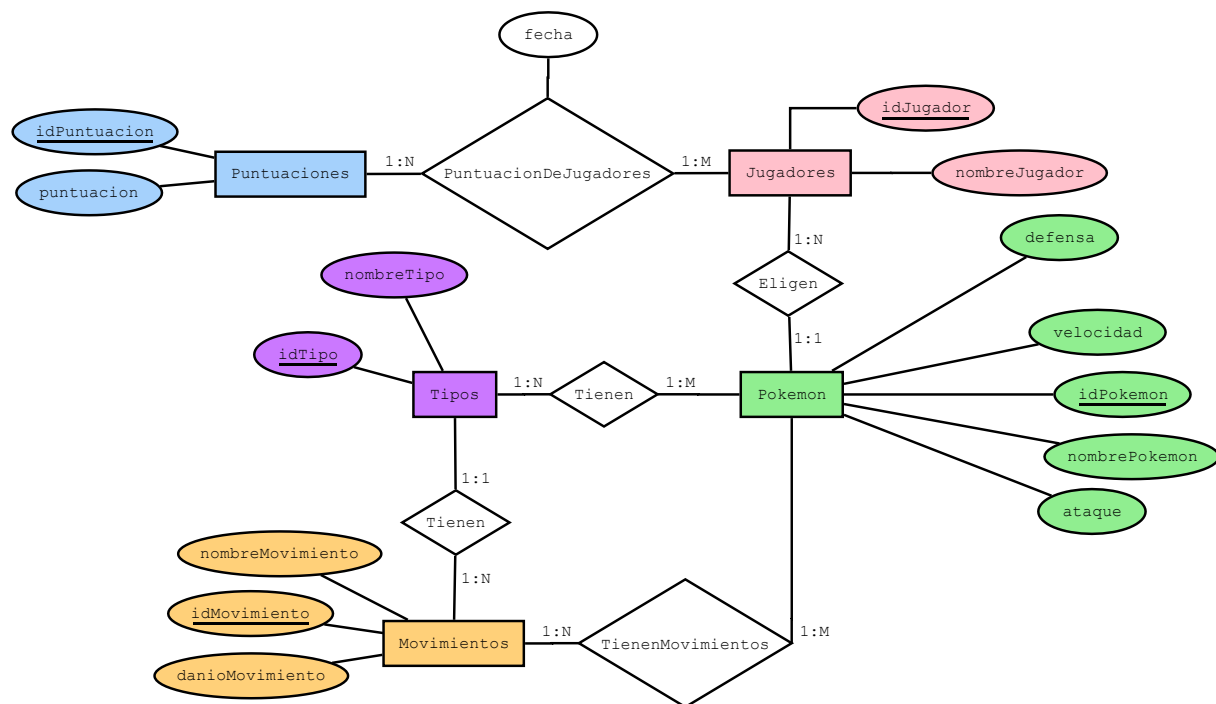
1. Introducción

En el siguiente manual, vamos a exponer los detalles técnicos de Juego Pokémon, así como los detalles de su base de datos (diagrama e-r, esquema relacional, sentencias sql...), como de las librerías externas que hemos usado para la realización del programa, los requisitos para poder utilizar el programa y la explicación de las clases Java que hemos desarrollado para la realización del mismo.

También, describiremos el proceso de la creación del ejecutable de nuestro programa.

2. Base de Datos

Modelo ERD



Modelo E/R

JUGADORES (idJugador, nombreJugador)

PUNTUACIONES (idPuntuacion, puntuacion)

PUNTUACIONESDEJUGADORES (idPuntuacionDeJugador, idPuntuacionFK, idJugadorFK, fecha)

TIPOS (idTipo, nombreTipo)

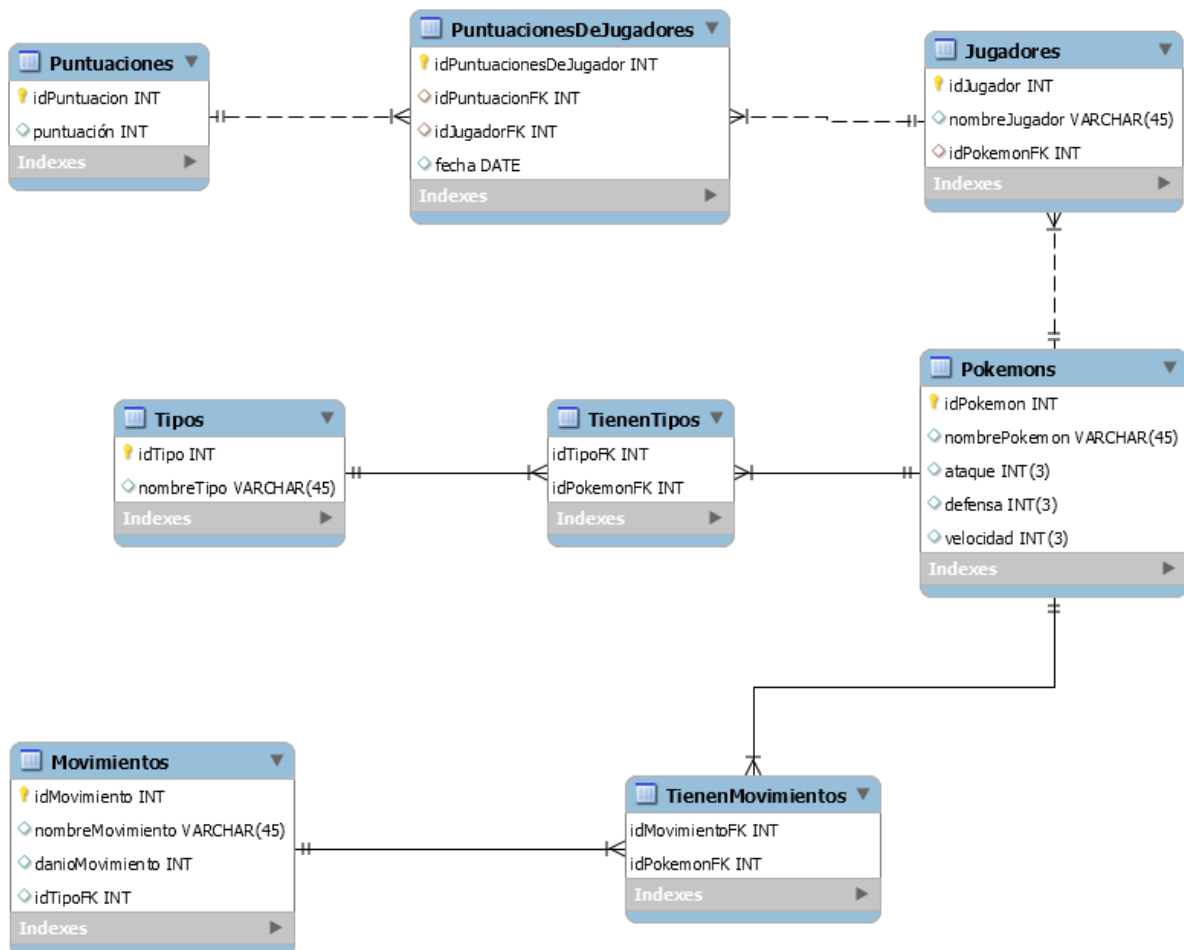
POKEMONS (idPokemon, nombrePokemon, ataque, defensa, velocidad, idTipoFK)

MOVIMIENTOS (idMovimiento, nombreMovimiento, danioMovimiento, idTipoFK)

TIENENTIPOS(idTipoFK, idPokemonFK)

TIENEN MOVIMIENTOS(idMovimientoFK, idPokemonFK)

Modelo Workbench



Sentencias SQL

Creación de la Base de datos:

```
create database JuegoPokemon char set utf8mb4 collate utf8mb4_spanish2_ci;
```

Creación de las Tablas:

Tipos:

```
CREATE TABLE `tipos` (
  `idTipo` int(11) NOT NULL AUTO_INCREMENT,
  `nombreTipo` varchar(30) CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci DEFAULT NULL,
  PRIMARY KEY (`idTipo`)
) ENGINE=InnoDB AUTO_INCREMENT=18 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_spanish2_ci;
```

Pokémons:

```
CREATE TABLE `pokemons` (
  `idPokemon` int(11) NOT NULL AUTO_INCREMENT,
  `nombrePokemon` varchar(30) CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci DEFAULT NULL,
  `puntosSalud` int(3) DEFAULT NULL,
  `ataque` int(3) DEFAULT NULL,
  `defensa` int(3) DEFAULT NULL,
  `idTipoFK` int(11) DEFAULT NULL,
  `imagen` varchar(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci DEFAULT NULL,
  PRIMARY KEY (`idPokemon`),
  KEY `idTipoFK` (`idTipoFK`),
  CONSTRAINT `pokemons_ibfk_1` FOREIGN KEY (`idTipoFK`) REFERENCES `tipos` (`idtipo`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_spanish2_ci;
```

Movimientos:

```
CREATE TABLE `movimientos` (
  `idMovimiento` int(11) NOT NULL AUTO_INCREMENT,
  `nombreMovimiento` varchar(30) CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci DEFAULT NULL,
  `danioMovimiento` int(3) DEFAULT NULL,
  `idTipoFK` int(11) DEFAULT NULL,
  PRIMARY KEY (`idMovimiento`),
  KEY `idTipoFK` (`idTipoFK`),
  CONSTRAINT `movimientos_ibfk_1` FOREIGN KEY (`idTipoFK`) REFERENCES `tipos` (`idtipo`)
) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_spanish2_ci;
```

Línea Movimientos:

```
CREATE TABLE `lineamovimientos` (
  `idMovimientoFK` int(11) NOT NULL,
  `idPokemonFK` int(11) NOT NULL,
  PRIMARY KEY (`idMovimientoFK`, `idPokemonFK`),
  KEY `idPokemonFK` (`idPokemonFK`),
  CONSTRAINT `lineamovimientos_ibfk_1` FOREIGN KEY (`idPokemonFK`) REFERENCES `pokemons` (`idpokemon`),
  CONSTRAINT `lineamovimientos_ibfk_2` FOREIGN KEY (`idMovimientoFK`) REFERENCES `movimientos` (`idmovimiento`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_spanish2_ci;
```

Jugadores:

```
CREATE TABLE `jugadores` (  
  `idJugador` int(11) NOT NULL AUTO_INCREMENT,  
  `nombreJugador` varchar(15) CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci DEFAULT NULL,  
  `idPokemonFK` int(11) DEFAULT NULL,  
  PRIMARY KEY (`idJugador`),  
  KEY `idPokemonFK` (`idPokemonFK`),  
  CONSTRAINT `jugadores_ibfk_1` FOREIGN KEY (`idPokemonFK`) REFERENCES `pokemons` (`idpokemon`)  
) ENGINE=InnoDB AUTO_INCREMENT=41 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_spanish2_ci;
```

Puntuaciones:

```
CREATE TABLE `puntuaciones` (  
  `idPuntuacion` int(11) NOT NULL AUTO_INCREMENT,  
  `puntuacion` int(15) DEFAULT NULL,  
  `idJugadorFK` int(11) DEFAULT NULL,  
  PRIMARY KEY (`idPuntuacion`),  
  KEY `idJugadorFK` (`idJugadorFK`),  
  CONSTRAINT `puntuaciones_ibfk_1` FOREIGN KEY (`idJugadorFK`) REFERENCES `jugadores` (`idjugador`)  
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_spanish2_ci;
```

3. Librerías

Para desarrollar el programa, hemos usado la librería externa siguiente:

MYSQL Connector 5.1.46 Utilizada para proporcionarnos las utilidades para conectarnos a la Base de Datos.

4. Requisitos e Instalación








Requisitos

Los requisitos para poder correr el programa son los siguientes:

- MySQL Server
- Java Versión 8
- Archivos de instalación de nuestro programa.

Instalación del programa

Para la instalación del programa, deberemos tener la carpeta de nuestro programa con los siguientes archivos:

	imagenes	10/06/2019 9:06	Carpeta de archivos	
	AyudaJuego.chm	07/06/2019 9:57	Archivo de Ayuda ...	313 KB
	instalador.bat	10/06/2019 17:37	Archivo por lotes ...	2 KB
	JuegoPokemon.jar	10/06/2019 9:15	Executable Jar File	1.016 KB
	JuegoPokemon.sql	10/06/2019 9:38	SQL Text File	8 KB
	Manual Tecnico Juego Pokemon.pdf	10/06/2019 18:27	Chrome HTML Do...	369 KB
	Manual Usuario Juego Pokemon.pdf	10/06/2019 18:27	Chrome HTML Do...	369 KB

En esta carpeta, hemos guardado todo lo necesario para el correcto funcionamiento del programa, también hemos creado un instalador.bat en el que tenemos creadas las líneas de código para poder instalar el programa.

```

@echo off
REM Habilitar tildes y ñ
chcp 65001
echo Bienvenido al programa de instalación
REM Crear Ubicaciones
mkdir "C:\Program Files\JuegoPokemon"
mkdir "C:\Program Files\JuegoPokemon\imagenes"
REM Mover los ficheros
copy "C:\Users\Andres Ceballos\Desktop\JuegoPokemon\*.*)" "C:\Program Files\JuegoPokemon"
copy "C:\Users\Andres Ceballos\Desktop\JuegoPokemon\imagenes\*.*)" "C:\Program Files\JuegoPokemon\imagenes"

cd "C:\Program Files\JuegoPokemon"
REM Montar la BD
echo Montando la base de datos: Introduzca clave de root
mysql -u root -p -e "CREATE DATABASE JuegoPokemon CHARACTER SET utf8 COLLATE utf8_spanish2_ci;"
echo Importando la base de datos: Introduzca clave de root
mysql -u root -p JuegoPokemon<JuegoPokemon.sql
echo Creando el usuario: Introduzca clave de root
mysql -u root -p -e "CREATE USER 'usuarioJuego'@'localhost' IDENTIFIED BY 'Stadium2018;'"
echo Dando permisos al usuario: Introduzca clave de root
mysql -u root -p -e "GRANT ALL ON JuegoPokemon.* TO 'usuarioJuego'@'localhost'"
REM Iniciar
echo Iniciando el programa
JuegoPokemon.jar

```

Para instalar el programa deberemos pulsar sobre el instalador.bat y a continuación seguiremos los pasos que nos va indicando el instalador.

Cuando tengamos creado el instalador, nos dirigiremos a la carpeta donde hayamos instalado el programa y lo ejecutaremos.

5. Clases Java:

Lo siguiente que vamos a exponer es una breve explicación de cada clase Java que hemos desarrollado que componen nuestro programa.

Ayuda.java: En esta clase, ejecutamos la ayuda de nuestro Juego.

```
package es.studium.Juego;

import java.io.IOException;

public class Ayuda {

    public Ayuda()
    {
        try
        {
            Runtime.getRuntime().exec("hh.exe
AyudaJuego/AyudaJuego.chm");
        }
        catch (IOException e)
        {
            e.printStackTrace();
        }
    }
}
```

BaseDatos.java: En esta clase, tenemos todas las funciones necesarias en las que hacemos inserts, selects, etc.

```
package es.studium.Juego;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

import javax.swing.JOptionPane;

public class BaseDatos {
```



```
public Connection conectar(String baseDatos, String usuario,
String clave)
{
    String driver = "com.mysql.jdbc.Driver";
    String url
="jdbc:mysql://localhost:3306/"+baseDatos+"?autoReconnect=true&useSSL=false";
    String login = usuario;
    String password = clave;
    Connection connection = null;
    try
    {
        Class.forName(driver);
        connection = DriverManager.getConnection(url,
login,password);
    }
    catch (ClassNotFoundException cnfe)
    {
        JOptionPane.showMessageDialog(null,cnfe.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
    }
    catch (SQLException sqle)
    {
        JOptionPane.showMessageDialog(null,sqle.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
    }
    return connection;
}
public void desconectar(Connection c)
{
    try
    {
        if(c!=null)
        {
            c.close();
        }
    }
    catch (SQLException e)
    {
        JOptionPane.showMessageDialog(null,e.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
    }
}
```

```

    public ResultSet ejecutarSelect(String sentencia, Connection c)
    {
        try
        {
            Statement statement = c.createStatement();
            ResultSet rs= statement.executeQuery(sentencia);
            return rs;
        }
        catch(SQLException e)
        {
            JOptionPane.showMessageDialog(null,e.getMessage(),"Error",
            JOptionPane.ERROR_MESSAGE);
            return null;
        }
    }
    public void ejecutarIDA(String sentencia, Connection c)
    {
        try
        {
            Statement statement = c.createStatement();
            statement.executeUpdate(sentencia);
        }
        catch(SQLException e)
        {
            JOptionPane.showMessageDialog(null,e.getMessage(),"Error",
            JOptionPane.ERROR_MESSAGE);
        }
    }
}

```

EstadisticaMovimiento.java: En esta clase, podemos ver las estadísticas de un movimiento de un Pokémon elegido anteriormente.

```

package es.studium.Juego;

import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import java.sql.ResultSet;

```

```

import java.sql.SQLException;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextField;

public class EstadisticaMovimiento extends JFrame implements
WindowListener, ActionListener
{
    /**
     *
     */
    private static final long serialVersionUID = 1L;
    JLabel lblNombre = new JLabel("Nombre:");
    JLabel lblDanio = new JLabel("Daño:");
    JLabel lblTipo = new JLabel("Tipo:");
    JTextField txtNombre = new JTextField(10);
    JTextField txtDanio = new JTextField(10);
    JTextField txtTipo = new JTextField(10);
    JButton btnAceptar = new JButton("Aceptar");

    JPanel pnl1 = new JPanel();
    JPanel pnl2 = new JPanel();
    JPanel pnl3 = new JPanel();
    JPanel pnl4 = new JPanel();
    BaseDatos bd = new BaseDatos();

    public EstadisticaMovimiento(int idMovimiento)
    {
        this.setLayout(new GridLayout(4,1));
        this.setLocationRelativeTo(null);
        this.setSize(500,300);
        //Rellenar Estadísticas.
        ResultSet Mov = bd.ejecutarSelect("select
nombreMovimiento, danioMovimiento, nombreTipo from movimientos,
tipos where idTipoFK = idTipo and idMovimiento ="+idMovimiento+";",
bd.conectar("juegoPokemon", "usuarioJuego",
"Stodium2018;"));
        try {
            Mov.next();
            txtNombre.setText(Mov.getString("nombreMovimiento"));
            txtDanio.setText(Mov.getString("danioMovimiento"));
            txtTipo.setText(Mov.getString("nombreTipo"));
            pnl1.add(lblNombre);

```

```

        pnl1.add(txtNombre);
        txtNombre.setEditable(false);
        pnl2.add(lblDanio);
        pnl2.add(txtDanio);
        txtDanio.setEditable(false);
        pnl3.add(lblTipo);
        pnl3.add(txtTipo);
        txtTipo.setEditable(false);
        pnl4.add(btnAceptar);
        btnAceptar.addActionListener(this);
        this.add(pnl1);
        this.add(pnl2);
        this.add(pnl3);
        this.add(pnl4);
        this.setVisible(true);
    } catch (SQLException e) {

        JOptionPane.showMessageDialog(null, e.getMessage(), "Error",
JOptionPane.ERROR_MESSAGE);
    }
    bd.desconectar(bd.conectar("juegoPokemon",
"usuarioJuego", "Stodium2018;"));
}

@Override
public void actionPerformed(ActionEvent ae) {
    // TODO Auto-generated method stub
    if(btnAceptar.equals(ae.getSource())) {
        this.setVisible(false);
    }
}

@Override
public void windowActivated(WindowEvent e) {
    // TODO Auto-generated method stub
}

@Override
public void windowClosed(WindowEvent e) {
    // TODO Auto-generated method stub
}

@Override
public void windowClosing(WindowEvent e) {
    // TODO Auto-generated method stub
}

```

```
        this.setVisible(false);
    }

    @Override
    public void windowDeactivated(WindowEvent e) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowDeiconified(WindowEvent e) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowIconified(WindowEvent e) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowOpened(WindowEvent e) {
        // TODO Auto-generated method stub
    }
}
```

EstadisticasPokemon.java: En esta clase, podemos ver las estadísticas del Pokémon.

```
package es.studium.Juego;

import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.swing.*.*;

public class EstadisticasPokemon extends JFrame implements
WindowListener, ActionListener
```

```
{  
    /**  
     *  
     */  
    private static final long serialVersionUID = 1L;  
  
    BaseDatos bd = new BaseDatos();  
  
    JLabel lblNombre = new JLabel("Nombre:");  
    JLabel lblAtaque = new JLabel("Ataque:");  
    JLabel lblDefensa = new JLabel("Defensa:");  
    JLabel lblTipo = new JLabel("Tipo");  
    JLabel lblPuntosSalud = new JLabel("Puntos de Salud:");  
    JLabel lblImg = new JLabel("");  
  
    JTextField txtNombre = new JTextField(10);  
    JTextField txtAtaque = new JTextField(10);  
    JTextField txtDefensa = new JTextField(10);  
    JTextField txtPuntosSalud = new JTextField(10);  
    JTextField txtTipo = new JTextField(10);  
  
    JButton btnAceptar = new JButton("Aceptar");  
    JButton btnMovimiento1 = new JButton("");  
    JButton btnMovimiento2 = new JButton("");  
    JButton btnMovimiento3 = new JButton("");  
    JButton btnMovimiento4 = new JButton("");  
  
    String [] arrayMovimientos = new String[4];  
    int [] arrayMovimientosid = new int[4];  
  
    JPanel pnlImagen = new JPanel();  
    JPanel pnl = new JPanel();  
    JPanel pnl1 = new JPanel();  
    JPanel pnl2 = new JPanel();  
    JPanel pnl3 = new JPanel();  
    JPanel pnl4 = new JPanel();  
    JPanel pnl5 = new JPanel();  
    JPanel pnl6 = new JPanel();  
  
    public EstadisticasPokemon(int idPokemon)  
    {  
        this.setTitle("Estadísticas Pokemon");  
        this.setSize(500,700);  
        this.setLocationRelativeTo(null);  
        this.setLayout(new GridLayout(8,1));  
  
        int idTipo = 0;
```

```

        ResultSet rs = bd.ejecutarSelect("SELECT * FROM pokemons
where idPokemon="+idPokemon+";",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

        try {
            rs.next();
            lblImg.setIcon(new
ImageIcon("imagenes/"+rs.getString("imagen")));
            idTipo = rs.getInt("idTipoFK");
            txtNombre.setText(rs.getString("nombrePokemon"));
            txtPuntosSalud.setText(rs.getString("puntosSalud"));
            txtAtaque.setText(rs.getString("ataque"));
            txtDefensa.setText(rs.getString("defensa"));
            txtNombre.setEditable(false);
            txtPuntosSalud.setEditable(false);
            txtAtaque.setEditable(false);
            txtDefensa.setEditable(false);
        }

        catch (SQLException sqle)
        {

            JOptionPane.showMessageDialog(null,sqle.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
        }
        ResultSet rs2 = bd.ejecutarSelect("SELECT * FROM tipos
where idTipo="+idTipo+";",bd.conectar("juegoPokemon","usuarioJuego",
"Stodium2018;") );
        try {
            rs2.next();

            txtTipo.setText(rs2.getString("nombreTipo"));
            txtTipo.setEditable(false);

        } catch (SQLException sqle)
        {

            JOptionPane.showMessageDialog(null,sqle.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
        }

        //CONSULTA DE MOVIMIENTOS
        String consultaAtaquesJ1 = "SELECT * FROM
lineaMovimientos join movimientos on idMovimiento = idMovimientoFK
where idPokemonFK="+idPokemon+";";

```

```
ResultSet rsMovimientos =
bd.ejecutarSelect(consultaAtaquesJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
try {

    int i=0;
    rsMovimientos.next();
    for(i=0;i<4;i++)
    {

        arrayMovimientos[i]=rsMovimientos.getString("nombreMovimiento")
;

        arrayMovimientosid[i]=rsMovimientos.getInt("idMovimiento");
        rsMovimientos.next();
    }
    btnMovimiento1.setText(arrayMovimientos[0]);
    btnMovimiento2.setText(arrayMovimientos[1]);
    btnMovimiento3.setText(arrayMovimientos[2]);
    btnMovimiento4.setText(arrayMovimientos[3]);
    pnlImagen.add(lblImg);
    pnl.add(lblNombre);
    pnl.add(txtNombre);
    pnl1.add(lblPuntosSalud);
    pnl1.add(txtPuntosSalud);
    pnl2.add(lblAtaque);
    pnl2.add(txtAtaque);
    pnl3.add(lblDefensa);
    pnl3.add(txtDefensa);
    pnl4.add(lblTipo);
    pnl4.add(txtTipo);

    pnl5.setBorder(BorderFactory.createTitledBorder(BorderFactory.c
reateEtchedBorder(), "Estadísticas de movimientos"));
    pnl5.add(btnMovimiento1);
    pnl5.add(btnMovimiento2);
    pnl5.add(btnMovimiento3);
    pnl5.add(btnMovimiento4);
    pnl6.add(btnAceptar);
    btnMovimiento1.addActionListener(this);
    btnMovimiento2.addActionListener(this);
    btnMovimiento3.addActionListener(this);
    btnMovimiento4.addActionListener(this);

    this.add(pnlImagen);
    this.add(pnl);
```



```
        this.add(pnl1);
        this.add(pnl2);
        this.add(pnl3);
        this.add(pnl4);
        this.add(pnl5);
        this.add(pnl6);

        this.addWindowListener(this);
        btnAceptar.addActionListener(this);
        this.setVisible(true);
    } catch (SQLException e) {

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error 3",
        JOptionPane.ERROR_MESSAGE);
    }
}

@Override
public void windowActivated(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowClosed(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowClosing(WindowEvent arg0) {
    // TODO Auto-generated method stub
    this.setVisible(false);
}

@Override
public void windowDeactivated(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowDeiconified(WindowEvent arg0) {
    // TODO Auto-generated method stub

}
```

```
@Override
public void windowIconified(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowOpened(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void actionPerformed(ActionEvent ae) {
    // TODO Auto-generated method stub
    if(btnAceptar.equals(ae.getSource())) {
        this.setVisible(false);
    }
    //Movimientos
    else if(btnMovimiento1.equals(ae.getSource())) {
        new EstadisticaMovimiento(arrayMovimientosid[0]);
    }
    else if(btnMovimiento2.equals(ae.getSource())) {
        new EstadisticaMovimiento(arrayMovimientosid[1]);
    }
    else if(btnMovimiento3.equals(ae.getSource())) {
        new EstadisticaMovimiento(arrayMovimientosid[2]);
    }
    else if(btnMovimiento4.equals(ae.getSource())) {
        new EstadisticaMovimiento(arrayMovimientosid[3]);
    }
}
}
```

Funciones.java: Aquí encontramos la función del cálculo del daño de un movimiento.

```
package es.studium.Juego;

import java.sql.ResultSet;
import java.sql.SQLException;

import javax.swing.JOptionPane;
```

```
public class Funciones {

    BaseDatos bd = new BaseDatos();

    public int calcularDanio (int ataque, int defensa, int
idAtaque,int tipoPkm)
    {

        String consultaMovimiento = "SELECT * FROM movimientos
WHERE idMovimiento =" +idAtaque+"";
        ResultSet rs = bd.ejecutarSelect(consultaMovimiento,
bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));

        int potencia;
        int danio = 0;
        double efectividad = 1;
        try {
            rs.next();
            potencia = rs.getInt("danioMovimiento");
            int tipoAtaque = rs.getInt("idTipoFK");

            //ACERO
            //MUY EFICAZ
            if (tipoAtaque ==1 & tipoPkm==8) {
                efectividad =2;
            } else if(tipoAtaque == 1 && tipoPkm == 13) {
                efectividad = 2;
            }

            //POCO EFICAZ
            else if(tipoAtaque == 1 && tipoPkm == 1) {
                efectividad = 0.5;
            }
            else if(tipoAtaque == 1 && tipoPkm == 2) {
                efectividad = 0.5;
            }
            else if(tipoAtaque == 1 && tipoPkm == 5) {
                efectividad = 0.5;
            }
            else if(tipoAtaque == 1 && tipoPkm == 7) {
                efectividad = 0.5;
            }
            //AGUA
            //MUY EFICAZ
            else if(tipoAtaque == 2 && tipoPkm == 7) {
                efectividad = 2;
            } else if(tipoAtaque == 2 && tipoPkm == 13) {
```

```
        efectividad = 2;
    } else if(tipoAtaque == 2 && tipoPkm == 15) {
        efectividad = 2;
    }

    //POCO EFICAZ
    else if(tipoAtaque == 2 && tipoPkm == 2) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 2 && tipoPkm == 4) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 2 && tipoPkm == 11) {
        efectividad = 0.5;
    }
    //BICHO
    //MUY EFICAZ
    else if(tipoAtaque == 3 && tipoPkm == 11) {
        efectividad = 2;
    } else if(tipoAtaque == 3 && tipoPkm == 12) {
        efectividad = 2;
    } else if(tipoAtaque == 3 && tipoPkm == 14) {
        efectividad = 2;
    }

    //POCO EFICAZ
    else if(tipoAtaque == 3 && tipoPkm == 1) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 3 && tipoPkm == 6) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 3 && tipoPkm == 7) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 3 && tipoPkm == 9) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 3 && tipoPkm == 16) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 3 && tipoPkm == 17) {
        efectividad = 0.5;
    }
    //DRAGÓN
    //MUY EFICAZ
    else if(tipoAtaque == 4 && tipoPkm == 4) {
        efectividad = 2;
    }
}
```

```
//POCO EFICAZ
else if(tipoAtaque == 4 && tipoPkm == 1) {
    efectividad = 0.5;
}
//ELÉCTRICO
//MUY EFICAZ
else if(tipoAtaque == 5 && tipoPkm == 2) {
    efectividad = 2;
}
else if(tipoAtaque == 5 && tipoPkm == 17) {
    efectividad = 2;
}
//POCO EFICAZ
else if(tipoAtaque == 5 && tipoPkm == 4) {
    efectividad = 0.5;
}
else if(tipoAtaque == 5 && tipoPkm == 5) {
    efectividad = 0.5;
}
else if(tipoAtaque == 5 && tipoPkm == 11) {
    efectividad = 0.5;
}
//NO AFECTA
else if(tipoAtaque == 5 && tipoPkm == 15) {
    efectividad = 0;
}
//FANTASMA
//MUY EFICAZ
else if(tipoAtaque == 6 && tipoPkm == 6) {
    efectividad = 2;
}
else if(tipoAtaque == 6 && tipoPkm == 12) {
    efectividad = 2;
}
//POCO EFICAZ
else if(tipoAtaque == 6 && tipoPkm == 11) {
    efectividad = 0.5;
}
//NO AFECTA
else if(tipoAtaque == 6 && tipoPkm == 10) {
    efectividad = 0;
}
//FUEGO
//MUY EFICAZ
else if(tipoAtaque == 7 && tipoPkm == 1) {
    efectividad = 2;
}
```

```
else if(tipoAtaque == 7 && tipoPkm == 3) {
    efectividad = 2;
}
else if(tipoAtaque == 7 && tipoPkm == 8) {
    efectividad = 2;
}
else if(tipoAtaque == 7 && tipoPkm == 11) {
    efectividad = 2;
}
//POCO EFICAZ
else if(tipoAtaque == 7 && tipoPkm == 2) {
    efectividad = 0.5;
}
else if(tipoAtaque == 7 && tipoPkm == 4) {
    efectividad = 0.5;
}
else if(tipoAtaque == 7 && tipoPkm == 7) {
    efectividad = 0.5;
}
else if(tipoAtaque == 7 && tipoPkm == 13) {
    efectividad = 0.5;
}

//HIELO
//MUY EFICAZ
else if(tipoAtaque == 8 && tipoPkm == 4) {
    efectividad = 2;
}
else if(tipoAtaque == 8 && tipoPkm == 11) {
    efectividad = 2;
}
else if(tipoAtaque == 8 && tipoPkm == 15) {
    efectividad = 2;
}
else if(tipoAtaque == 8 && tipoPkm == 17) {
    efectividad = 2;
}
//POCO EFICAZ
else if(tipoAtaque == 8 && tipoPkm == 1) {
    efectividad = 0.5;
}
else if(tipoAtaque == 8 && tipoPkm == 2) {
    efectividad = 0.5;
}
else if(tipoAtaque == 8 && tipoPkm == 7) {
    efectividad = 0.5;
}
```

```
        else if(tipoAtaque == 8 && tipoPkm == 8) {
            efectividad = 0.5;
        }
//LUCHA
//MUY EFICAZ
        else if(tipoAtaque == 9 && tipoPkm == 1) {
            efectividad = 2;
        }
        else if(tipoAtaque == 9 && tipoPkm == 8) {
            efectividad = 2;
        }
        else if(tipoAtaque == 9 && tipoPkm == 10) {
            efectividad = 2;
        }
        else if(tipoAtaque == 9 && tipoPkm == 13) {
            efectividad = 2;
        }
        else if(tipoAtaque == 9 && tipoPkm == 14) {
            efectividad = 2;
        }
//POCO EFICAZ
        else if(tipoAtaque == 9 && tipoPkm == 3) {
            efectividad = 0.5;
        }
        else if(tipoAtaque == 9 && tipoPkm == 12) {
            efectividad = 0.5;
        }
        else if(tipoAtaque == 9 && tipoPkm == 16) {
            efectividad = 0.5;
        }
        else if(tipoAtaque == 9 && tipoPkm == 17) {
            efectividad = 0.5;
        }
//NO AFECTA
        else if(tipoAtaque == 1 && tipoPkm == 6) {
            efectividad = 0;
        }
//NORMAL
//POCO EFICAZ
        else if(tipoAtaque == 10 && tipoPkm == 1) {
            efectividad = 0.5;
        }
        else if(tipoAtaque == 10 && tipoPkm == 13) {
            efectividad = 0.5;
        }
//NO AFECTA
        else if(tipoAtaque == 10 && tipoPkm == 6) {
```

```
        efectividad = 0;
    }

    //PLANTA
    //MUY EFICAZ
    else if(tipoAtaque == 11 && tipoPkm == 2) {
        efectividad = 2;
    }
    else if(tipoAtaque == 11 && tipoPkm == 13) {
        efectividad = 2;
    }
    else if(tipoAtaque == 11 && tipoPkm == 15) {
        efectividad = 2;
    }
    //POCO EFICAZ
    else if(tipoAtaque == 11 && tipoPkm == 1) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 11 && tipoPkm == 3) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 11 && tipoPkm == 4) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 11 && tipoPkm == 7) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 11 && tipoPkm == 11) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 11 && tipoPkm == 16) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 11 && tipoPkm == 17) {
        efectividad = 0.5;
    }
    //PSIQUICO
    //MUY EFICAZ
    else if(tipoAtaque == 12 && tipoPkm == 9) {
        efectividad = 2;
    }
    else if(tipoAtaque == 12 && tipoPkm == 16) {
        efectividad = 2;
    }
    //POCO EFICAZ
    else if(tipoAtaque == 12 && tipoPkm == 1) {
        efectividad = 0.5;
```



```
    }  
    else if(tipoAtaque == 1 && tipoPkm == 12) {  
        efectividad = 0.5;  
    }  
    //NO AFECTA  
    else if(tipoAtaque == 12 && tipoPkm == 15) {  
        efectividad = 0;  
    }  
    //ROCA  
    //MUY EFICAZ  
    else if(tipoAtaque == 13 && tipoPkm == 3) {  
        efectividad = 2;  
    }  
    else if(tipoAtaque == 13 && tipoPkm == 7) {  
        efectividad = 2;  
    }  
    else if(tipoAtaque == 13 && tipoPkm == 8) {  
        efectividad = 2;  
    }  
    else if(tipoAtaque == 13 && tipoPkm == 17) {  
        efectividad = 2;  
    }  
    //POCO EFICAZ  
    else if(tipoAtaque == 13 && tipoPkm == 1) {  
        efectividad = 0.5;  
    }  
    else if(tipoAtaque == 13 && tipoPkm == 9) {  
        efectividad = 0.5;  
    }  
    else if(tipoAtaque == 13 && tipoPkm == 14) {  
        efectividad = 0.5;  
    }  
    //SINIESTRO  
    //MUY EFICAZ  
    else if(tipoAtaque == 14 && tipoPkm == 6) {  
        efectividad = 2;  
    }  
    else if(tipoAtaque == 14 && tipoPkm == 12) {  
        efectividad = 2;  
    }  
    //POCO EFICAZ  
    else if(tipoAtaque == 14 && tipoPkm == 9) {  
        efectividad = 0.5;  
    }  
    else if(tipoAtaque == 14 && tipoPkm == 14) {  
        efectividad = 0.5;  
    }  
}
```

```
//TIERRA
    //MUY EFICAZ
    else if(tipoAtaque == 15 && tipoPkm == 1) {
        efectividad = 2;
    }
    else if(tipoAtaque == 15 && tipoPkm == 5) {
        efectividad = 2;
    }
    else if(tipoAtaque == 15 && tipoPkm == 7) {
        efectividad = 2;
    }
    else if(tipoAtaque == 15 && tipoPkm == 13) {
        efectividad = 2;
    }
    else if(tipoAtaque == 15 && tipoPkm == 16) {
        efectividad = 2;
    }
    //POCO EFICAZ
    else if(tipoAtaque == 15 && tipoPkm == 3) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 15 && tipoPkm == 11) {
        efectividad = 0.5;
    }
    //NO AFECTA
    else if(tipoAtaque == 15 && tipoPkm == 17) {
        efectividad = 0;
    }
//VENENO
    //MUY EFICAZ
    else if(tipoAtaque == 16 && tipoPkm == 11) {
        efectividad = 2;
    }
    //POCO EFICAZ
    else if(tipoAtaque == 16 && tipoPkm == 6) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 16 && tipoPkm == 13) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 16 && tipoPkm == 15) {
        efectividad = 0.5;
    }
    else if(tipoAtaque == 16 && tipoPkm == 16) {
        efectividad = 0.5;
    }
}
```

```

        //NO AFECTA
        else if(tipoAtaque == 16 && tipoPkm == 1) {
            efectividad = 0;
        }
        //VOLADOR
        //MUY EFICAZ
        else if(tipoAtaque == 17 && tipoPkm == 3) {
            efectividad = 2;
        }
        else if(tipoAtaque == 17 && tipoPkm == 9) {
            efectividad = 2;
        }
        else if(tipoAtaque == 17 && tipoPkm == 11) {
            efectividad = 2;
        }
        //POCO EFICAZ
        else if(tipoAtaque == 17 && tipoPkm == 1) {
            efectividad = 0.5;
        }
        else if(tipoAtaque == 17 && tipoPkm == 5) {
            efectividad = 0.5;
        }
        else if(tipoAtaque == 17 && tipoPkm == 13) {
            efectividad = 0.5;
        }
        else {
            efectividad = 1;
        }

        //FORMULA DAÑO:
        danio = (int)
(0.1*efectividad*100*((ataque*potencia*((0.2*100)+1))/(200*defensa)+
2));

        return danio;
    } catch (SQLException e) {
        // TODO Auto-generated catch block

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
    }
    return danio;
}

```

```
}
```

MenuPrincipal.java: Esta clase es la clase principal en la que encontramos el menú principal.

```
package es.studium.Juego;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JPanel;
import java.awt.GridBagLayout;
import java.awt.GridBagConstraints;
import javax.swing.JLabel;
import javax.swing.ImageIcon;
public class MenuPrincipal extends JFrame implements WindowListener,
ActionListener {

    private static final long serialVersionUID = 1L;
    JButton btnIniciar = new JButton("Iniciar");
    JButton btnTop10 = new JButton("Top 10");
    JButton btnAyuda = new JButton("Ayuda");

    JPanel pnlImagen = new JPanel();
    JPanel pnluno = new JPanel();
    JLabel lblImg = new JLabel("");

    public MenuPrincipal()
    {
        this.setTitle("Combate Pokemon!");
        setLocationRelativeTo(null);
        this.setSize(600,350);
        this.setLayout(new GridBagLayout());
        GridBagConstraints gbcPnlImagen = new
GridBagConstraints();

        gbcPnlImagen.gridy = 0;
        this.add(pnlImagen, gbcPnlImagen);
        lblImg.setIcon(new ImageIcon("imagenes/pokemon.png"));

        pnlImagen.add(lblImg);
```

```
GridBagConstraints gbcPnluno = new GridBagConstraints();

gbcPnluno.gridy = 1;
this.add(pnluno, gbcPnluno);
pnluno.add(btnIniciar);
pnluno.add(btnTop10);
pnluno.add(btnAyuda);
btnAyuda.addActionListener(this);
btnTop10.addActionListener(this);
btnIniciar.addActionListener(this);

this.setVisible(true);
addWindowListener(this);
}

public static void main(String[] args) {
    new MenuPrincipal();
}

@Override
public void actionPerformed(ActionEvent ae)
{
    if (btnIniciar.equals(ae.getSource()))
    {
        new NuevaPartida();
        this.setVisible(false);
    } else if (btnTop10.equals(ae.getSource())) {
        new Top10();
    } else if (btnAyuda.equals(ae.getSource())) {
        new Ayuda();
    }
}

@Override
public void windowActivated(WindowEvent arg0) {
    // TODO Auto-generated method stub
}

@Override
public void windowClosed(WindowEvent arg0) {
    // TODO Auto-generated method stub
}
```

```
    }

    @Override
    public void windowClosing(WindowEvent arg0) {
        System.exit(0);
    }

    @Override
    public void windowDeactivated(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowDeiconified(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowIconified(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowOpened(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }
}
```

NuevaPartida.java: En esta clase, elegimos los nombres de los jugadores.

```
package es.studium.Juego;

import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.TextEvent;
import java.awt.event.TextListener;
import java.awt.event.WindowEvent;
```

```
import java.awt.event.WindowListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JTextField;

public class NuevaPartida extends JFrame implements WindowListener,
ActionListener, TextListener {

    private static final long serialVersionUID = 1L;
    JLabel lblJugadores = new JLabel("Introduce el nombre de los
Jugadores: ");
    JLabel lblJugador1 = new JLabel("Jugador 1:");
    JLabel lblJugador2 = new JLabel("Jugador 2:");

    JTextField txtJugador1 = new JTextField (10);
    JTextField txtJugador2 = new JTextField (10);

    JButton btnSiguiente = new JButton ("Siguiente");

    JPanel pnluno = new JPanel();
    JPanel pndos = new JPanel();
    JPanel pnltres = new JPanel();
    JPanel pnlcuatro = new JPanel();

    public NuevaPartida() {

        this.setTitle("Nueva Partida");
        setLocationRelativeTo(null);
        this.setSize(300,250);
        setLayout(new GridLayout(4,1));

        pnluno.add(lblJugadores);
        this.add(pnluno);

        pndos.add(lblJugador1);
        pndos.add(txtJugador1);
        this.add(pndos);

        pnltres.add(lblJugador2);
        pnltres.add(txtJugador2);
        this.add(pnltres);

        pnlcuatro.add(btnSiguiente);
        btnSiguiente.addActionListener(this);
```

```
        this.add(pnlcuatro);

        this.setVisible(true);
        addWindowListener(this);
    }

    @Override
    public void actionPerformed(ActionEvent ae) {
        if(btnSiguiente.equals(ae.getSource()))
        {
            new Seleccion1(txtJugador1.getText(),
txtJugador2.getText());
            this.setVisible(false);
        }
    }

    @Override
    public void windowActivated(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowClosed(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowClosing(WindowEvent arg0) {
        // TODO Auto-generated method stub
        System.exit(0);
    }

    @Override
    public void windowDeactivated(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowDeiconified(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }
}
```



```
@Override
public void windowIconified(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowOpened(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void textValueChanged(TextEvent arg0) {
    // TODO Auto-generated method stub

}

}
```

Partida.java: Esta es la clase en la que transcurre la batalla.

```
package es.studium.Juego;

import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.swing.*;
import javax.swing.border.Border;
import javax.swing.border.EtchedBorder;
import javax.swing.border.TitledBorder;

public class Partida extends JFrame implements WindowListener,
ActionListener
{
    private static final long serialVersionUID = 1L;

    int turno = 1;
```

```
BaseDatos bd = new BaseDatos();
Funciones f = new Funciones();

JPanel pnluno = new JPanel();
JPanel pnldos = new JPanel();
JPanel pnltres = new JPanel();
JPanel pnlcuatro = new JPanel();

Border bordejpanel = new TitledBorder(new EtchedBorder());

JPanel pnlVida1 = new JPanel();
JPanel pnlPkm1 = new JPanel();

JPanel pnlVida2 = new JPanel();
JPanel pnlPkm2 = new JPanel();

JPanel pnlMovimientos = new JPanel();
JPanel pnlMovimientosJ1 = new JPanel();
JPanel pnlMovimientosJ2 = new JPanel();

JPanel pnlOpciones = new JPanel();

JPanel pnlTranscurso = new JPanel();

JLabel lblQueHacer = new JLabel("");
JLabel lblAtaque = new JLabel("");
JLabel lblDanio = new JLabel("");

JLabel lblVida1 = new JLabel("");
JLabel lblVida2 = new JLabel("");
JLabel lblPkm1 = new JLabel("");
JLabel lblPkm2 = new JLabel("");

JProgressBar PBvida1;
JProgressBar PBvida2;

JButton btnAtacar = new JButton ("Atacar");
JButton btnRendirse = new JButton ("Rendirse");

JButton btnSi = new JButton ("Sí");
JButton btnNo = new JButton ("No");

JButton btnAtaque1J1 = new JButton ("");
JButton btnAtaque2J1 = new JButton ("");
JButton btnAtaque3J1 = new JButton ("");
JButton btnAtaque4J1 = new JButton ("");
```

```

JButton btnAtaque1J2 = new JButton ("");
JButton btnAtaque2J2 = new JButton ("");
JButton btnAtaque3J2 = new JButton ("");
JButton btnAtaque4J2 = new JButton ("");

JDialog dlgRendirse = new JDialog();
JDialog dlgFin = new JDialog();

JLabel lblRendirse = new JLabel("¿Está seguro de rendirte?");
JLabel lblFin = new JLabel("Fin de la Partida.");
int vidaRestada;
String consultaJ1 = "";
String consultaJ2 = "";

int [] arrayMovimientosid = new int[4];
int [] arrayMovimientos2id = new int[4];

String consultaAtaquesJ1;
String consultaAtaquesJ2;
int ataque, defensa, idAtaque, tipoPkm;
int idJugador1;
int idJugador2;

String nombreJugador1;
String nombreJugador2;
private final JLabel label = new JLabel("");
private final JLabel label_1 = new JLabel("");

public Partida(int idJ1,int idJ2)
{
    idJugador1=idJ1;
    idJugador2=idJ2;
    this.setTitle("Batalla");
    getContentPane().setLayout(new GridLayout(4,1));
    this.setSize(650, 400);
    this.setLocationRelativeTo(null);

    consultaJ1 = "select * from jugadores join pokemons on
idPokemonFK = idPokemon where idJugador ="+idJ1+"";
    consultaJ2 = "select * from jugadores join pokemons on
idPokemonFK = idPokemon where idJugador ="+idJ2+"";

    ResultSet rsJ1 = bd.ejecutarSelect(consultaJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

```

```

        try {
            rsJ1.next();
            nombreJugador1 = rsJ1.getString("nombreJugador");
            PBvida1 = new
JPProgressBar(0,rsJ1.getInt("puntosSalud"));

            PBvida1.setValue(rsJ1.getInt("puntosSalud"));
            pnlVida1.setLayout(new FlowLayout());
            pnlVida1.setBorder(bordejpanel);

            lblPkm1.setText(rsJ1.getString("nombrePokemon"));
            pnlVida1.add(lblPkm1);

            lblVida1.setText("PS: "+PBvida1.getValue());
            pnlVida1.add(lblVida1);
            pnlVida1.add(PBvida1);

            pnluno.add(pnlVida1);
            pnluno.add(pnlPkm1);
            label.setIcon(new
ImageIcon("imagenes/"+rsJ1.getString("imagen"))));

            pnlPkm1.add(label);
            getContentPane().add(pnluno);
            lblQueHacer.setText(nombreJugador1+", ¿Qué quieres
hacer?");
        } catch(SQLException e) {

            JOptionPane.showMessageDialog(null,e.getMessage(),"Error 1 ",
JOptionPane.ERROR_MESSAGE);
        }

        ResultSet rsJ2 = bd.ejecutarSelect(consultaJ2,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
        try {
            rsJ2.next();
            nombreJugador2 = rsJ2.getString("nombreJugador");
            PBvida2 = new
JPProgressBar(0,rsJ2.getInt("puntosSalud"));
            PBvida2.setValue(rsJ2.getInt("puntosSalud"));
            pnlVida2.setLayout(new FlowLayout());
            pnlVida2.setBorder(bordejpanel);
            lblPkm2.setText(rsJ2.getString("nombrePokemon"));
            pnlVida2.add(lblPkm2);
            lblVida2.setText("PS: "+PBvida2.getValue());
            pnlVida2.add(lblVida2);
            pnlVida2.add(PBvida2);

```

```

        pnlDos.add(pnlVida2);
        pnlDos.add(pnlPkm2);
        label_1.setIcon(new
ImageIcon("imagenes/"+rsJ2.getString("imagen"))));

        pnlPkm2.add(label_1);
        getContentPane().add(pnlDos);

        pnlMovimientos.setVisible(false);
        pnlMovimientos.setLayout(new FlowLayout());
        pnlMovimientos.setBorder(bordejpanel);
        pnlMovimientosJ1.setLayout(new GridLayout(2,2));

        consultaAtaquesJ1 = "SELECT * FROM lineaMovimientos
join movimientos on idMovimiento = idMovimientoFK where
idPokemonFK="+rsJ1.getInt("idPokemonFK")+";";
    } catch(SQLException e) {

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error 2",
JOptionPane.ERROR_MESSAGE);
    }
    ResultSet rsMovimientos =
bd.ejecutarSelect(consultaAtaquesJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    try {
        String [] arrayMovimientos = new String[4];

        int i=0;
        rsMovimientos.next();
        for(i=0;i<4;i++)
        {

            arrayMovimientos[i]=rsMovimientos.getString("nombreMovimiento")
;

            arrayMovimientosid[i]=rsMovimientos.getInt("idMovimiento");
            rsMovimientos.next();
        }

        btnAtaque1J1.setText(arrayMovimientos[0]);
        pnlMovimientosJ1.add(btnAtaque1J1);
        btnAtaque1J1.addActionListener(this);

        btnAtaque2J1.setText(arrayMovimientos[1]);
        pnlMovimientosJ1.add(btnAtaque2J1);
        btnAtaque2J1.addActionListener(this);

```

```
        btnAtaque3J1.setText(arrayMovimientos[2]);
        pnlMovimientosJ1.add(btnAtaque3J1);
        btnAtaque3J1.addActionListener(this);

        btnAtaque4J1.setText(arrayMovimientos[3]);
        pnlMovimientosJ1.add(btnAtaque4J1);
        btnAtaque4J1.addActionListener(this);

        pnlMovimientosJ2.setLayout(new GridLayout(2,2));

        consultaAtaquesJ2 = "SELECT * FROM lineaMovimientos
join movimientos on idMovimiento = idMovimientoFK where
idPokemonFK="+rsJ2.getInt("idPokemonFK")+";";
    } catch(SQLException e) {

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error 3",
JOptionPane.ERROR_MESSAGE);
    }

    ResultSet rsMovimientos2 =
bd.ejecutarSelect(consultaAtaquesJ2,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    try {
        String [] arrayMovimientos2 = new String[4];

        int j=0;
        rsMovimientos2.next();
        for(j=0;j<4;j++)
        {

            arrayMovimientos2[j]=rsMovimientos2.getString("nombreMovimiento
");

            arrayMovimientos2id[j]=rsMovimientos2.getInt("idMovimiento");
            rsMovimientos2.next();
        }
        btnAtaque1J2.setText(arrayMovimientos2[0]);
        pnlMovimientosJ2.add(btnAtaque1J2);
        btnAtaque1J2.addActionListener(this);

        btnAtaque2J2.setText(arrayMovimientos2[1]);
        pnlMovimientosJ2.add(btnAtaque2J2);
        btnAtaque2J2.addActionListener(this);
```

```
        btnAtaque3J2.setText(arrayMovimientos2[2]);
        pnlMovimientosJ2.add(btnAtaque3J2);
        btnAtaque3J2.addActionListener(this);

        btnAtaque4J2.setText(arrayMovimientos2[3]);
        pnlMovimientosJ2.add(btnAtaque4J2);
        btnAtaque4J2.addActionListener(this);
    } catch (SQLException e) {

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error 4",
        JOptionPane.ERROR_MESSAGE);
    }

    pnltres.add(pnlMovimientos);

    pnlOpciones.setLayout(new GridLayout(2,1));
    pnlOpciones.setBorder(bordejpanel);
    pnlOpciones.add(btnAtacar);
    btnAtacar.addActionListener(this);
    pnlOpciones.add(btnRendirse);
    btnRendirse.addActionListener(this);
    pnltres.add(pnlOpciones);
    getContentPane().add(pnltres);

    pnlTranscurso.setLayout(new FlowLayout());
    pnlTranscurso.setBorder(bordejpanel);
    pnlTranscurso.add(lblQueHacer);
    pnlcuatro.add(pnlTranscurso);
    getContentPane().add(pnlcuatro);

    dlgRendirse.setSize(200,100);
    dlgRendirse.setTitle("Rendirse");
    dlgRendirse.getContentPane().setLayout(new FlowLayout());
    dlgRendirse.getContentPane().add(lblRendirse);
    dlgRendirse.getContentPane().add(btnSi);
    btnSi.addActionListener(this);
    dlgRendirse.getContentPane().add(btnNo);
    btnNo.addActionListener(this);
    dlgRendirse.setLocationRelativeTo(null);

    dlgFin.setSize(200,100);
    dlgFin.setTitle("Fin de Partida");
    dlgFin.getContentPane().setLayout(new FlowLayout());
    dlgFin.setLocationRelativeTo(null);
    dlgFin.getContentPane().add(lblFin);
```

```

        this.addWindowListener(this);
        this.setVisible(true);

    }

    @Override
    public void actionPerformed(ActionEvent ae) {
        //RENDIRSE
        if (btnRendirse.equals(ae.getSource()))
        {
            dlgRendirse.setVisible(true);
        }

        if(btnSi.equals(ae.getSource())) {
            dlgRendirse.setVisible(false);
            dlgFin.setVisible(true);
            this.setVisible(false);
            new MenuPrincipal();

        } else if(btnNo.equals(ae.getSource())) {
            dlgRendirse.setVisible(false);
        }

        //PELEA
        if(btnAtacar.equals(ae.getSource())) {
            pnlMovimientos.setVisible(true);
        }
        //ATAQUES J1
        if(btnAtaque1J1.equals(ae.getSource()))
        {
            ResultSet rsJ1m = bd.ejecutarSelect(consultaJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));
            ResultSet rsJ2m = bd.ejecutarSelect(consultaJ2,
bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));
            ResultSet rsMov = bd.ejecutarSelect("SELECT * FROM
movimientos where idMovimiento =" + arrayMovimientosid[0] + ";",
bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));
            try {
                rsJ1m.next();
                rsJ2m.next();
                rsMov.next();
                pnlMovimientos.setVisible(false);
                pnlTranscurso.remove(lblQueHacer);

                lblAtaque.setText(rsJ1m.getString("nombrePokemon")+" Usó

```



```

"+btnAtaque1J1.getText());
        pnlTranscurso.add(lblAtaque);

        ataque = rsJ1m.getInt("ataque");
        defensa = rsJ2m.getInt("defensa");
        idAtaque = rsMov.getInt("idMovimiento");
        tipoPkm = rsJ2m.getInt("idTipoFK");

        vidaRestada = f.calcularDanio(ataque, defensa,
idAtaque, tipoPkm);
        PBvida2.setValue(PBvida2.getValue()-
vidaRestada);

        lblDanio.setText(rsJ2m.getString("nombrePokemon")+" pierde
"+vidaRestada+" PS");
        pnlTranscurso.add(lblDanio);
        lblQueHacer.setText(nombreJugador2+", ¿Qué
quieres hacer?");
        pnlTranscurso.add(lblQueHacer);
        turno=turno+1;
    } catch (SQLException e){

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error A1J1",
JOptionPane.ERROR_MESSAGE);
    }

}
else if (btnAtaque2J1.equals(ae.getSource()))
{
    ResultSet rsJ1m = bd.ejecutarSelect(consultaJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    ResultSet rsJ2m = bd.ejecutarSelect(consultaJ2,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

    ResultSet rsMov = bd.ejecutarSelect("SELECT * FROM
movimientos where idMovimiento =" +arrayMovimientosid[1]+";",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

    try {
        rsJ1m.next();
        rsJ2m.next();
        rsMov.next();
        pnlMovimientos.setVisible(false);
        pnlTranscurso.remove(lblQueHacer);

        lblAtaque.setText(rsJ1m.getString("nombrePokemon")+" Usó
"+btnAtaque2J1.getText());

```

```

        pnlTranscurso.add(lblAtaque);

        ataque = rsJ1m.getInt("ataque");
        defensa = rsJ2m.getInt("defensa");
        idAtaque = rsMov.getInt("idMovimiento");
        tipoPkm = rsJ2m.getInt("idTipoFK");

        vidaRestada = f.calcularDanio(ataque, defensa,
idAtaque, tipoPkm);
        PBvida2.setValue(PBvida2.getValue()-
vidaRestada);

        lblDanio.setText(rsJ2m.getString("nombrePokemon")+" pierde
"+vidaRestada+" PS");
        pnlTranscurso.add(lblDanio);
        lblQueHacer.setText(nombreJugador2+", ¿Qué
quieres hacer?");
        pnlTranscurso.add(lblQueHacer);
        turno=turno+1;
    } catch (SQLException e){
        JOptionPane.showMessageDialog(null,e.getMessage(),"Error A2J1",
JOptionPane.ERROR_MESSAGE);
    }
}
else if(btnAtaque3J1.equals(ae.getSource()))
{
    ResultSet rsJ1m = bd.ejecutarSelect(consultaJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));
    ResultSet rsJ2m = bd.ejecutarSelect(consultaJ2,
bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));

    ResultSet rsMov = bd.ejecutarSelect("SELECT * FROM
movimientos where idMovimiento =" +arrayMovimientosid[2]+";",
bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));

    try {
        rsJ1m.next();
        rsJ2m.next();
        rsMov.next();
        pnlMovimientos.setVisible(false);
        pnlTranscurso.remove(lblQueHacer);

        lblAtaque.setText(rsJ1m.getString("nombrePokemon")+" Usó
"+btnAtaque3J1.getText());
        pnlTranscurso.add(lblAtaque);

```

```

        ataque = rsJ1m.getInt("ataque");
        defensa = rsJ2m.getInt("defensa");
        idAtaque = rsMov.getInt("idMovimiento");
        tipoPkm = rsJ2m.getInt("idTipoFK");

        vidaRestada = f.calcularDanio(ataque, defensa,
idAtaque, tipoPkm);
        PBvida2.setValue(PBvida2.getValue()-
vidaRestada);

        lblDanio.setText(rsJ2m.getString("nombrePokemon")+" pierde
"+vidaRestada+" PS");
        pnlTranscurso.add(lblDanio);
        lblQueHacer.setText(nombreJugador2+", ¿Qué
quieres hacer?");
        pnlTranscurso.add(lblQueHacer);
        turno=turno+1;
    } catch (SQLException e){
        JOptionPane.showMessageDialog(null,e.getMessage(),"Error A3J1",
JOptionPane.ERROR_MESSAGE);
    }
}
else if(btnAtaque4J1.equals(ae.getSource()))
{
    ResultSet rsJ1m = bd.ejecutarSelect(consultaJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    ResultSet rsJ2m = bd.ejecutarSelect(consultaJ2,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

    ResultSet rsMov = bd.ejecutarSelect("SELECT * FROM
movimientos where idMovimiento =" + arrayMovimientosid[3] + ";",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    try {
        rsJ1m.next();
        rsJ2m.next();
        rsMov.next();
        pnlMovimientos.setVisible(false);
        pnlTranscurso.remove(lblQueHacer);

        lblAtaque.setText(rsJ1m.getString("nombrePokemon")+" Usó
"+btnAtaque4J1.getText());
        pnlTranscurso.add(lblAtaque);

        ataque = rsJ1m.getInt("ataque");
        defensa = rsJ2m.getInt("defensa");
        idAtaque = rsMov.getInt("idMovimiento");

```

```

        tipoPkm = rsJ2m.getInt("idTipoFK");

        vidaRestada = f.calcularDanio(ataque, defensa,
idAtaque, tipoPkm);
        PBvida2.setValue(PBvida2.getValue()-
vidaRestada);

        lblDanio.setText(rsJ2m.getString("nombrePokemon")+" pierde
"+vidaRestada+" PS");
        pnlTranscurso.add(lblDanio);
        lblQueHacer.setText(nombreJugador2+", ¿Qué
quieres hacer?");
        pnlTranscurso.add(lblQueHacer);
        turno=turno+1;
    } catch (SQLException e){

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error A4J1",
JOptionPane.ERROR_MESSAGE);
    }
}

//ATAQUES J2
if(btnAtaque1J2.equals(ae.getSource()))
{
    ResultSet rsJ1m = bd.ejecutarSelect(consultaJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    ResultSet rsJ2m = bd.ejecutarSelect(consultaJ2,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

    ResultSet rsMov = bd.ejecutarSelect("SELECT * FROM
movimientos where idMovimiento =" +arrayMovimientos2id[0]+";",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    try {
        rsJ1m.next();
        rsJ2m.next();
        rsMov.next();
        pnlMovimientos.setVisible(false);
        pnlTranscurso.remove(lblQueHacer);

        lblAtaque.setText(rsJ2m.getString("nombrePokemon")+" Usó
"+btnAtaque1J2.getText());
        pnlTranscurso.add(lblAtaque);

        ataque = rsJ2m.getInt("ataque");
        defensa = rsJ1m.getInt("defensa");
        idAtaque = rsMov.getInt("idMovimiento");
        tipoPkm = rsJ1m.getInt("idTipoFK");

```

```

        vidaRestada = f.calcularDanio(ataque, defensa,
idAtaque, tipoPkm);
        PBvida1.setValue(PBvida1.getValue()-
vidaRestada);

        lblDanio.setText(rsJ1m.getString("nombrePokemon")+" pierde
"+vidaRestada+" PS");
        pnlTranscurso.add(lblDanio);
        lblQueHacer.setText(nombreJugador1+", ¿Qué
quieres hacer?");
        pnlTranscurso.add(lblQueHacer);
        turno=turno+1;
    } catch (SQLException e){

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error A1J2",
JOptionPane.ERROR_MESSAGE);
    }
}
else if (btnAtaque2J2.equals(ae.getSource()))
{
    ResultSet rsJ1m = bd.ejecutarSelect(consultaJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    ResultSet rsJ2m = bd.ejecutarSelect(consultaJ2,
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

    ResultSet rsMov = bd.ejecutarSelect("SELECT * FROM
movimientos where idMovimiento =" +arrayMovimientos2id[1]+";",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    try {
        rsJ1m.next();
        rsJ2m.next();
        rsMov.next();
        pnlMovimientos.setVisible(false);
        pnlTranscurso.remove(lblQueHacer);

        lblAtaque.setText(rsJ2m.getString("nombrePokemon")+" Usó
"+btnAtaque2J2.getText());
        pnlTranscurso.add(lblAtaque);

        ataque = rsJ2m.getInt("ataque");
        defensa = rsJ1m.getInt("defensa");
        idAtaque = rsMov.getInt("idMovimiento");
        tipoPkm = rsJ1m.getInt("idTipoFK");

        vidaRestada = f.calcularDanio(ataque, defensa,
idAtaque, tipoPkm);

```

```

        PBvida1.setValue(PBvida1.getValue()-
vidaRestada);

        lblDanio.setText(rsJ1m.getString("nombrePokemon")+" pierde
"+vidaRestada+" PS");
        pnlTranscurso.add(lblDanio);
        lblQueHacer.setText(nombreJugador1+", ¿Qué
quieres hacer?");
        pnlTranscurso.add(lblQueHacer);
        turno=turno+1;
    } catch (SQLException e){

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error A2J2",
JOptionPane.ERROR_MESSAGE);
    }
}
else if(btnAtaque3J2.equals(ae.getSource()))
{
    ResultSet rsJ1m = bd.ejecutarSelect(consultaJ1,
bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));
    ResultSet rsJ2m = bd.ejecutarSelect(consultaJ2,
bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));

    ResultSet rsMov = bd.ejecutarSelect("SELECT * FROM
movimientos where idMovimiento =" +arrayMovimientos2id[2]+"");
    bd.conectar("juegoPokemon","usuarioJuego", "Stadium2018;"));
    try {
        rsJ1m.next();
        rsJ2m.next();
        rsMov.next();
        pnlMovimientos.setVisible(false);
        pnlTranscurso.remove(lblQueHacer);

        lblAtaque.setText(rsJ2m.getString("nombrePokemon")+" Usó
"+btnAtaque3J2.getText());
        pnlTranscurso.add(lblAtaque);

        ataque = rsJ2m.getInt("ataque");
        defensa = rsJ1m.getInt("defensa");
        idAtaque = rsMov.getInt("idMovimiento");
        tipoPkm = rsJ1m.getInt("idTipoFK");

        vidaRestada = f.calcularDanio(ataque, defensa,
idAtaque, tipoPkm);
        PBvida1.setValue(PBvida1.getValue()-
vidaRestada);

```

```

        lblDanio.setText(rsJ1m.getString("nombrePokemon")+" pierde "+vidaRestada+" PS");
        pnlTranscurso.add(lblDanio);
        lblQueHacer.setText(nombreJugador1+", ¿Qué quieres hacer?");
        pnlTranscurso.add(lblQueHacer);
        turno=turno+1;
    } catch (SQLException e){

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error A3J2",
        JOptionPane.ERROR_MESSAGE);
    }
}
else if(btnAtaque4J2.equals(ae.getSource()))
{
    ResultSet rsJ1m = bd.ejecutarSelect(consultaJ1,
    bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    ResultSet rsJ2m = bd.ejecutarSelect(consultaJ2,
    bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

    ResultSet rsMov = bd.ejecutarSelect("SELECT * FROM
    movimientos where idMovimiento =" +arrayMovimientos2id[3]+"");
    bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
    try {
        rsJ1m.next();
        rsJ2m.next();
        rsMov.next();
        pnlMovimientos.setVisible(false);
        pnlTranscurso.remove(lblQueHacer);

        lblAtaque.setText(rsJ2m.getString("nombrePokemon")+" Usó "+btnAtaque4J2.getText());
        pnlTranscurso.add(lblAtaque);

        ataque = rsJ2m.getInt("ataque");
        defensa = rsJ1m.getInt("defensa");
        idAtaque = rsMov.getInt("idMovimiento");
        tipoPkm = rsJ1m.getInt("idTipoFK");

        vidaRestada = f.calcularDanio(ataque, defensa,
        idAtaque, tipoPkm);
        PBvida1.setValue(PBvida1.getValue()-
        vidaRestada);

        lblDanio.setText(rsJ1m.getString("nombrePokemon")+" pierde "+vidaRestada+" PS");
        pnlTranscurso.add(lblDanio);

```

```

        lblQueHacer.setText(nombreJugador1+", ¿Qué
quieres hacer?");
        pnlTranscurso.add(lblQueHacer);
        turno=turno+1;
    } catch (SQLException e){

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error A4J2",
JOptionPane.ERROR_MESSAGE);
    }
}
//MOSTRAR MOVIMIENTOS
if(turno%2!=0)
{
    pnlMovimientos.remove(pnlMovimientosJ2);
    pnlMovimientos.add(pnlMovimientosJ1);

} else {
    pnlMovimientos.remove(pnlMovimientosJ1);
    pnlMovimientos.add(pnlMovimientosJ2);

}
if(PBvida1.getValue()<=0)
{
    lblFin.setText(nombreJugador2+" Gana en
"+turno/2+" turnos.");
    bd.ejecutarIDA("INSERT INTO puntuaciones
VALUES (null, "+turno/2+", "+idJugador2+")",
bd.conectar("juegoPokemon", "usuarioJuego", "Stodium2018;"));

    bd.desconectar(bd.conectar("juegoPokemon", "usuarioJuego",
"Stodium2018;"));

    dlgFin.setVisible(true);
    this.setVisible(false);
    new MenuPrincipal();

} else if ( PBvida2.getValue()<=0) {
    lblFin.setText(nombreJugador1+" Gana en
"+turno/2+" turnos.");
    bd.ejecutarIDA("INSERT INTO puntuaciones
VALUES (null, "+turno/2+", "+idJugador1+")",
bd.conectar("juegoPokemon", "usuarioJuego", "Stodium2018;"));

    bd.desconectar(bd.conectar("juegoPokemon", "usuarioJuego",
"Stodium2018;"));

    dlgFin.setVisible(true);
    this.setVisible(false);
    new MenuPrincipal();
}

```



```
    }

    }

    @Override
    public void windowActivated(WindowEvent e) {
        // TODO Auto-generated method stub

    }

    @Override
    public void windowClosed(WindowEvent e) {
        // TODO Auto-generated method stub

    }

    @Override
    public void windowClosing(WindowEvent e)
    {
        int seleccion = JOptionPane.showOptionDialog(
null,"¿Desea finalizar la partida?", "Finalizar
partida", JOptionPane.YES_NO_CANCEL_OPTION, JOptionPane.QUESTION_MESSA
GE, null, new Object[] { "Finalizar", "Cancelar"}, "Cancelar");
        if (seleccion == 0)
        {
            this.setVisible(false);
        }
    }

    @Override
    public void windowDeactivated(WindowEvent e) {
        // TODO Auto-generated method stub

    }

    @Override
    public void windowDeiconified(WindowEvent e) {
        // TODO Auto-generated method stub

    }

    @Override
    public void windowIconified(WindowEvent e) {
        // TODO Auto-generated method stub

    }
}
```

```

@Override
public void windowOpened(WindowEvent e) {
    // TODO Auto-generated method stub

}

}

```

Seleccion1.java: En esta clase, el primer jugador seleccionará el Pokémon con el que va a jugar.

```

package es.studium.Juego;

import java.awt.Choice;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import java.sql.*;

import javax.swing.*.*;

public class Seleccion1 extends JFrame implements WindowListener,
ActionListener
{

    private static final long serialVersionUID = 1L;
    JLabel lblSeleccionar = new JLabel("Selecciona Pokemon: ");
    JLabel lblBUScar = new JLabel("Buscar:");
    BaseDatos bd = new BaseDatos();
    JTextField txtBuscar = new JTextField (10);
    Choice Pokemons = new Choice ();

    JButton btnAceptar = new JButton ("Aceptar");
    JButton btnEstadisticas = new JButton("Estadísticas");

    JPanel pnluno = new JPanel();
    JPanel pnldos = new JPanel();
    JPanel pnltres = new JPanel();
    JPanel pnlcuatro = new JPanel();

    String jugador1="";

```

```

String jugador2="";

public Seleccion1(String j1, String j2)
{
    jugador1 = j1;
    jugador2 = j2;
    this.setTitle(jugador1+" ¡Elige Pokémon!");
    this.setLocationRelativeTo(null);
    this.setSize(350,250);
    this.setLayout(new GridLayout(4,1));

    //Rellenar choice con Pokemons
    ResultSet rs = bd.ejecutarSelect("SELECT * FROM
pokemons", bd.conectar("juegoPokemon","usuarioJuego",
"Stodium2018;"));
    try {
        while(rs.next())
        {
            String
pokemon=Integer.toString(rs.getInt("idPokemon"));
            pokemon = pokemon+".-
"+rs.getString("nombrePokemon");
            Pokemons.add(pokemon);
        }
    } catch (SQLException e) {

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
    }
    bd.desconectar(
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

    pnluno.add(lblSeleccionar);
    this.add(pnluno);

    pnltres.add(Pokemons);
    this.add(pnltres);

    pnlcuatro.add(btnAceptar);
    pnlcuatro.add(btnEstadisticas);
    btnAceptar.addActionListener(this);
    btnEstadisticas.addActionListener(this);
    this.add(pnlcuatro);

    this.setVisible(true);
    addWindowListener(this);
}

```

```

    }

    @Override
    public void actionPerformed(ActionEvent ae) {
        if(btnAceptar.equals(ae.getSource()))
        {
            int idJugador = 0;
            String[] array=
Pokemons.getSelectedItemAt().toString().split("-");
            int idPokemon = Integer.parseInt(array[0]);
            bd.ejecutarIDA("INSERT INTO jugadores VALUES
(null,'"+jugador1+"','"+idPokemon+"");",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
            JOptionPane.showMessageDialog(null,"Primer Jugador
añadido!", "Jugador añadido", JOptionPane.INFORMATION_MESSAGE);
            ResultSet jugadorCreado = bd.ejecutarSelect("SELECT
idJugador FROM jugadores ORDER BY idJugador DESC;",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
            try {
                jugadorCreado.next();
                idJugador = jugadorCreado.getInt("idJugador");
            } catch (SQLException e) {
                // TODO Auto-generated catch block

                JOptionPane.showMessageDialog(null,e.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
            }

            this.setVisible(false);
            new Seleccion2(jugador2, idJugador);
        }
        if(btnEstadisticas.equals(ae.getSource())) {
            String[] array=
Pokemons.getSelectedItemAt().toString().split("-");
            int idPokemon = Integer.parseInt(array[0]);
            new EstadisticasPokemon(idPokemon);
        }
    }

    @Override
    public void windowActivated(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override

```

```
public void windowClosed(WindowEvent arg0) {  
    // TODO Auto-generated method stub  
  
}  
  
@Override  
public void windowClosing(WindowEvent arg0) {  
    this.setVisible(false);  
    new MenuPrincipal();  
  
}  
  
@Override  
public void windowDeactivated(WindowEvent arg0) {  
    // TODO Auto-generated method stub  
  
}  
  
@Override  
public void windowDeiconified(WindowEvent arg0) {  
    // TODO Auto-generated method stub  
  
}  
  
@Override  
public void windowIconified(WindowEvent arg0) {  
    // TODO Auto-generated method stub  
  
}  
  
@Override  
public void windowOpened(WindowEvent arg0) {  
    // TODO Auto-generated method stub  
  
}  
  
}
```

Seleccion2.java: En esta clase, el segundo jugador seleccionará el Pokémon con el que va a jugar.

```
package es.studium.Juego;  
  
import java.awt.Choice;  
import java.awt.GridLayout;
```

```
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.TextEvent;
import java.awt.event.TextListener;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.swing.*.*;

public class Seleccion2 extends JFrame implements WindowListener,
ActionListener, TextListener
{

    private static final long serialVersionUID = 1L;
    JLabel lblSeleccionar = new JLabel("Selecciona Pokemon: ");
    JLabel lblBUscar = new JLabel("Buscar:");
    BaseDatos bd = new BaseDatos();
    JTextField txtBuscar = new JTextField (10);
    Choice Pokemons = new Choice ();

    JButton btnAceptar = new JButton ("Aceptar");
    JButton btnEstadisticas = new JButton("Estadísticas");

    JPanel pnluno = new JPanel();
    JPanel pnldos = new JPanel();
    JPanel pnltres = new JPanel();
    JPanel pnlcuatro = new JPanel();

    String jugador2="";
    int idJugador1=0;
    int idJugador2 =0;

    public Seleccion2(String j2, int idJ1)
    {
        jugador2 = j2;
        idJugador1 = idJ1;

        this.setTitle(jugador2+" ¡Elige Pokémon!");
        this.setLocationRelativeTo(null);
        this.setSize(350,250);
        this.setLayout(new GridLayout(4,1));

        //Rellenar choice con Pokemons
        ResultSet rs = bd.ejecutarSelect("SELECT * FROM
```

```

pokemons", bd.conectar("juegoPokemon","usuarioJuego",
"Stodium2018;"));
    try {
        while(rs.next())
        {
            String
pokemon=Integer.toString(rs.getInt("idPokemon"));
            pokemon = pokemon+".-
"+rs.getString("nombrePokemon");
            Pokemons.add(pokemon);
        }
    } catch (SQLException e) {

        JOptionPane.showMessageDialog(null,e.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
    }
    bd.desconectar(
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));

    pnluno.add(lblSeleccionar);
    this.add(pnluno);

    pnltres.add(Pokemons);
    this.add(pnltres);

    pnlcuatro.add(btnAceptar);
    pnlcuatro.add(btnEstadisticas);
    btnAceptar.addActionListener(this);
    btnEstadisticas.addActionListener(this);
    this.add(pnlcuatro);

    this.setVisible(true);
    addWindowListener(this);
}

@Override
public void textValueChanged(TextEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void actionPerformed(ActionEvent ae) {
    if(btnAceptar.equals(ae.getSource()))
    {
        idJugador2 = 0;
        String[] array=

```

```

Pokemons.getSelectedItemAt().toString().split("-");
        int idPokemon = Integer.parseInt(array[0]);
        bd.ejecutarIDA("INSERT INTO jugadores VALUES
(null,'"+jugador2+"',"+idPokemon+");",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
        ResultSet jugadorCreado = bd.ejecutarSelect("SELECT
idJugador FROM jugadores ORDER BY idJugador DESC;",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
        try {
            jugadorCreado.next();
            idJugador2 =
jugadorCreado.getInt("idJugador");
            JOptionPane.showMessageDialog(null,"Segundo
Jugador añadido!", "Jugador añadido",
JOptionPane.INFORMATION_MESSAGE);
        } catch (SQLException e) {
            // TODO Auto-generated catch block

            JOptionPane.showMessageDialog(null,e.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
        }

        this.setVisible(false);
        new Partida(idJugador1, idJugador2);
    }
    if(btnEstadisticas.equals(ae.getSource())) {
        String[] array=
Pokemons.getSelectedItemAt().toString().split("-");
        int idPokemon = Integer.parseInt(array[0]);
        new EstadisticasPokemon(idPokemon);
    }

}

@Override
public void windowActivated(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowClosed(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override

```



```
    public void windowClosing(WindowEvent arg0) {
        System.exit(0);
    }

    @Override
    public void windowDeactivated(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowDeiconified(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowIconified(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }

    @Override
    public void windowOpened(WindowEvent arg0) {
        // TODO Auto-generated method stub
    }
}
```

Top10.java: En esta clase, creamos el Top 10 de mejores jugadores.

```
package es.studium.Juego;

import java.awt.BorderLayout;
import java.awt.FlowLayout;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import java.sql.ResultSet;
import java.sql.SQLException;
```

```
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.table.DefaultTableModel;

public class Top10 extends JFrame implements WindowListener,
ActionListener
{
    private static final long serialVersionUID = 1L;
    JPanel pnlPrimero = new JPanel();
    BaseDatos bd = new BaseDatos();

    JLabel lblPartidas = new JLabel("Mejores Jugadores");

    DefaultTableModel modelo = new DefaultTableModel();
    JTable tablaJugadores = new JTable(modelo);

    JButton btnOk = new JButton("Ok");

    JPanel pnlBtn = new JPanel();

    public Top10()
    {
        this.setTitle("Top 10");
        this.setLocationRelativeTo(null);
        this.setSize(549,354);
        this.setLayout(new GridLayout(3,1));
        pnlBtn.setLayout(new FlowLayout());
        modelo.addColumn("Nombre Jugador");
        modelo.addColumn("Puntuación");
        pnlPrimero.setLayout(new FlowLayout());

        ResultSet rs = bd.ejecutarSelect("SELECT nombreJugador,
puntuacion FROM jugadores JOIN puntuaciones WHERE idJugador =
idJugadorFK ORDER BY puntuacion ASC;",
bd.conectar("juegoPokemon","usuarioJuego", "Stodium2018;"));
        try {
            while (rs.next())
            {
                Object [] fila = new Object[2];
```

```

        for (int i=0;i<2;i++) {
            fila[i] = rs.getObject(i+1);
        }
        modelo.addRow(fila);
    }
} catch (SQLException e) {

    JOptionPane.showMessageDialog(null,e.getMessage(),"Error",
JOptionPane.ERROR_MESSAGE);
    }
    bd.desconectar(bd.conectar("juegoPokemon","usuarioJuego"
,"Studium2018;"));

    this.add(pnlPrimero);
    pnlPrimero.add(lblPartidas);

    this.add(new
JScrollPane(tablaJugadores),BorderLayout.CENTER);
    pnlBtn.add(btnOk);
    this.add(pnlBtn,BorderLayout.SOUTH);
    btnOk.addActionListener(this);
    this.addWindowListener(this);
    this.setVisible(true);

}

@Override
public void windowActivated(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowClosed(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowClosing(WindowEvent arg0) {
    this.setVisible(false);
    new MenuPrincipal();
}

```

```
}

@Override
public void windowDeactivated(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowDeiconified(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowIconified(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void windowOpened(WindowEvent arg0) {
    // TODO Auto-generated method stub

}

@Override
public void actionPerformed(ActionEvent ae) {
    if(btnOk.equals(ae.getSource())) {
        this.setVisible(false);
        new MenuPrincipal();
    }
}

}
```

7. Bibliografía

Temario Tema 9 Programación. (10 de 06 de 2019). Obtenido de Temario Tema 9 Programación:
http://aulastudium.com/pluginfile.php?file=%2F11026%2Fmod_resource%2Fcontent%2F2%2FPR-TEMA9%20Acceso%20a%20Bases%20de%20Datos.pdf