

SISTEMAS DISTRIBUIDOS

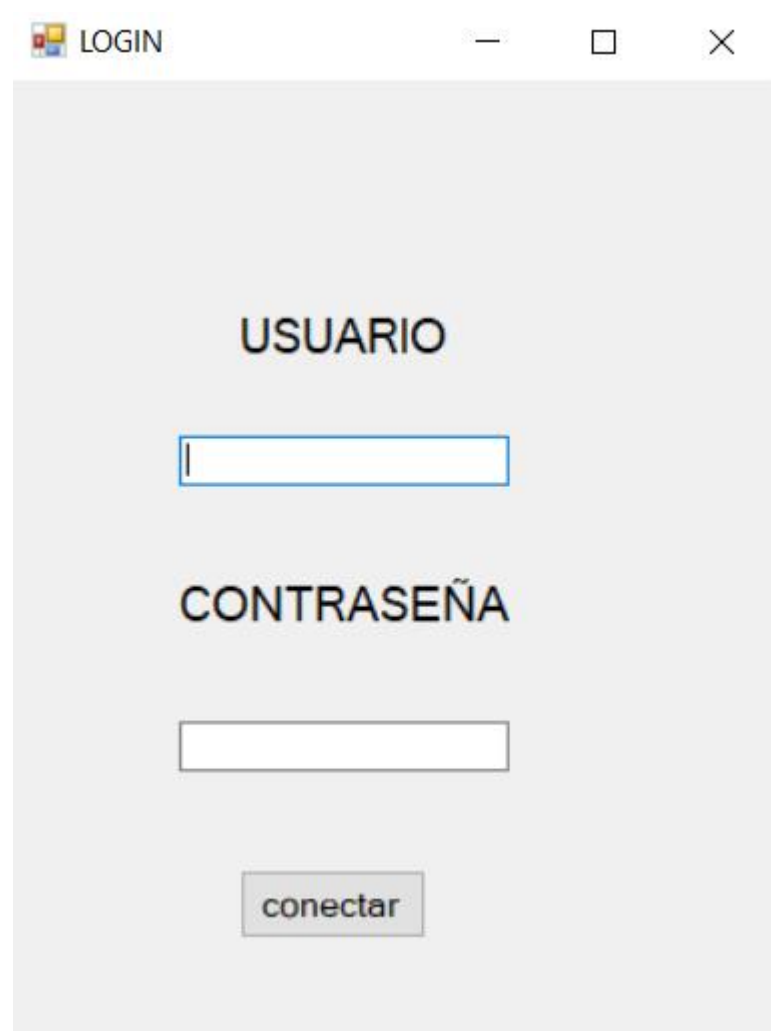
PRÁCTICA 2: CONTROL ESTACIONES METEOROLÓGICAS SERVICIOS WEB

Eduardo Correal Botero | Y2230091W | Curso 2018/2019

COMPONENTES

Aplicación .NET: es utilizada como el cliente, tiene la referencia web para obtener los servicios ofrecidos por las estaciones meteorológicas (getters y setters de humedad, temperatura, luminosidad y pantalla). Desde esta aplicación es desde donde nos conectaremos a las estaciones distribuidas y miraremos sus valores o los cambiemos como queramos. Tiene dos forms, uno usado para la parte del login, mientras otro es usado para la aplicación. Desde aquí es donde se genera la base de datos, los logs y la encriptación de los setters.

Pantalla de login



The image shows a screenshot of a Windows application window titled "LOGIN". The window has a standard Windows title bar with minimize, maximize, and close buttons. The main content area is light gray and contains the following elements:

- The word "USUARIO" in bold black text.
- A text input field with a blue border and a cursor.
- The word "CONTRASEÑA" in bold black text.
- A text input field with a gray border.
- A button labeled "conectar" in a gray box.

Pantalla del controlador

Controlador de Estaciones

Conectar

Operar

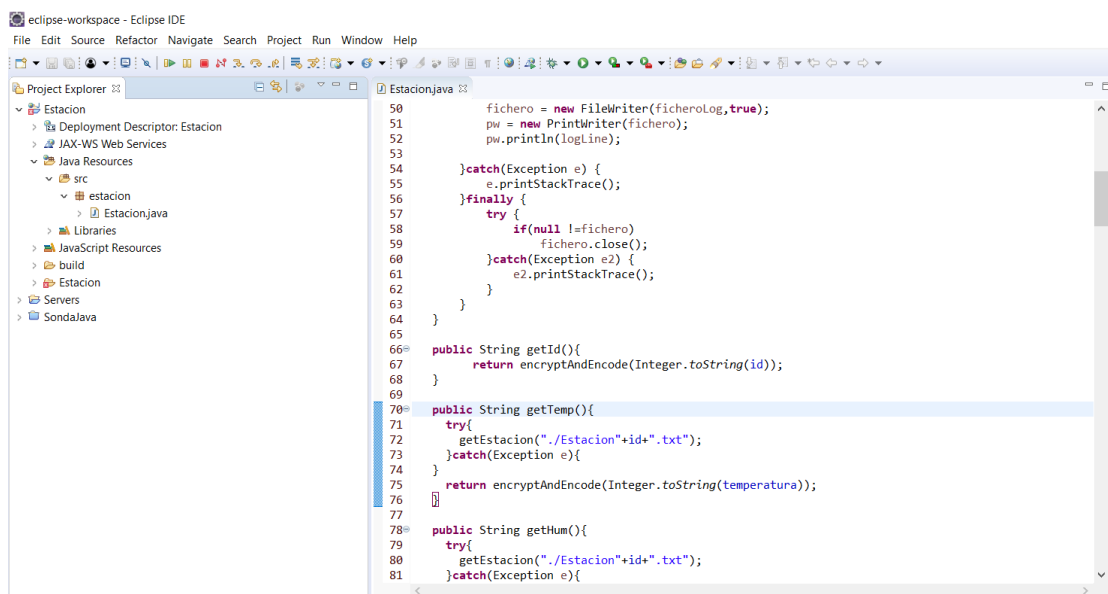
Código para consultar:

```

private void buttonConsultar_Click(object sender, EventArgs e)
{
    if(richTextBox1.Text != "" && comboBoxEstacion.Text.Contains("Estacion"))
    {
        Estacion.Estacion estacion = obtenerEstacion();
        int variable;
        string operacion=""; //qué -> coger estacion a la que fue y su operacion hecha
        Int32.TryParse(DecodeAndDecrypt(estacion.getId()), out int id);
        if (comboBoxVariable.Text.Contains("Humedad"))
        {
            operacion = "getHum";
            variable = Int32.Parse(DecodeAndDecrypt(estacion.getHum()));
            richTextBox2.Text += "La humedad de la estacion "+ id + " es de: " + variable.ToString() + "\n";
        }
        if (comboBoxVariable.Text.Contains("Temperatura"))
        {
            operacion = "getTemp";
            variable = Int32.Parse(DecodeAndDecrypt(estacion.getTemp()));
            richTextBox2.Text += "La temperatura de la estacion "+id+" es de: " + variable.ToString() + "\n";
        }
        if (comboBoxVariable.Text.Contains("Luminosidad"))
        {
            operacion = "getLum";
            variable = Int32.Parse(DecodeAndDecrypt(estacion.getLum()));
            richTextBox2.Text += "La luminosidad de la estacion "+id+" es de: " + variable.ToString() + "\n";
        }
    }
}

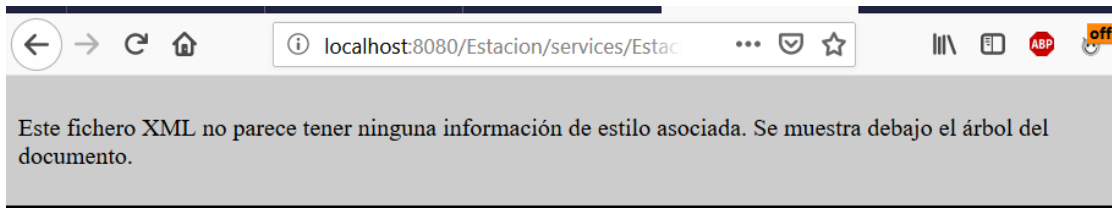
```

Aplicación JAVA:



Es el servicio web al que accederemos, tiene todos los métodos getters/setter como públicos para que los demás puedan acceder a ellos, el tipo de proyecto que debemos crear es un Dynamic web Project, y debemos crear el web service dando click derecho en Estacion.java, ir a web services-> créate web service. El único cambio que tenemos respecto a la estación de la práctica 1 es que el constructor es sin parámetros, ya que el servicio no es persistente.

Podemos consultar el wsdl del servicio publicado yendo a
localhost:8080/Estacion/services/Estacion?wsdl



```
-<wsdl:definitions targetNamespace="http://estacion">
  <wsdl:documentation> Please Type your service description here </wsdl:documentation>
  <wsdl:types>
    <xs:schema attributeFormDefault="qualified" elementFormDefault="qualified"
      targetNamespace="http://estacion">
      <xs:element name="getId">
        <xs:complexType>
          <xs:sequence/>
        </xs:complexType>
      </xs:element>
      <xs:element name="getIdResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" name="return" nillable="true" type="xs:string"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="setLog">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" name="logLine" nillable="true" type="xs:string"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="setLCD">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" name="encryptado" nillable="true" type="xs:string"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="getLum">
        <xs:complexType>
          <xs:sequence/>
        </xs:complexType>
      </xs:element>
    </xs:schema>
  </wsdl:types>
</wsdl:definitions>
```

DESPLIEGUE

Para desplegar la aplicación primero que todo debemos meter los .war de las estaciones en apache-tomcat -> webapps. E iniciamos el servidor tomcat desde apache-tomcat->bin y abrimos un terminal aquí y ejecutamos el comando startup.

Mientras que para el cliente .net, tenemos que generar su ejecutable

Y probamos la aplicación:

LOGIN

USUARIO

edu

CONTRASEÑA

conectar

Controlador de Estaciones

Conectar

Estacion5 ->localhost:8080

Operar

La luminosidad de la estacion 5 es de: 450
La humedad de la estacion 5 es de: 95

El valor ha sido cambiado