# Instrucciones

## Contenido de los programas fuentes

Este código fuente contiene:

1. Proyectos C# con el modelo en 3 capas:

* Capa de presentación BuenosAires.BodegaBA se ha terminado de programar VentanaProducto y VentanaLogin usando la Capa de Servicio. **Se ha quitado la referencia a BusinessLayer en BodegaBA y la Capa de Presentación quedó funcionando sólo mediante la Capa de Servicios con Webservices SOAP.**
* **Creada la Capa de Servicio con el Webservice SOAP WsAutenticacion y WsProducto.**
* Capa de negocio BuenosAires.BusinessLayer se ha terminado de programar BcProducto.
* Capa de datos BuenosAires.DataLayer se ha terminado de programar DcProducto.
* Capa de modelo BuenosAires.Model completa con las clases que representan las tablas de la BD.

1. Proyecto AppWebBA en Python/Django se ha terminado de programar API Rest Autenticar.
2. Proyecto BuenosAires.VentaBA en Java con las funcionalidades mínimas.
3. RECUERDA SIEMPRE DESCOMPRIMIR EL PROYECTO EN C:\BuenosAires y CORREGIR LAS CADENAS DE CONEXIÓN A LA BASE DE DATOS (NO ES LA MISMA EN TU COMPUTADOR QUE EN EL LABORATORIO).
4. Próximo paso: conectar proyecto Java de BuenosAires.VentaBA con Capa de Servicios usando Webservices SOAP desarrollados en C# .NET

**Correcciones 12/05/2025**

* Se cambió en todos lados para que Django no arranque en el puerto 8000 por defecto sino que en 8001, dado que en Duoc el puerto 8000 está tomado por otro programa.
* Para compatibilidad con los laboratorios de Duoc, se cambiaron estas 2 líneas en launch.json:

"type": "debugpy",

"args": ["runserver", "8001"],

* Para pago con WebPay se actualizó la línea: return\_url = request.build\_absolute\_uri('/pago\_exitoso/') para que tome la URL en que se está ejecutando el sistema y no la URL en duro http://127.0.0.1:8000
* En Capa de Servicios se deberá cambiar por el puerto 8000 por 8001

## Se realizará la tarea de programar el programa Java de BuenosAires.VentaBA conectando las ventanas de esta capa de presentación a la Capa de Servicio usando Web Service SOAP (en C#)

**Abrir proyecto Java y ejecutar Django y Capa de Servicios**

1. Abrir IDE de Apache NetBeans y abrir el proyecto C:\BuenosAires\VentaBA-Java\BuenosAires.VentaBA.
2. Ejecutar proyecto Django y Capa de Servicio en C#, ambos deben estar funcionando, de lo contrario no se podrá realizar la integración con Java.

**Agregar biblioteca Jar de Java para deserializar archivos Json**

1. En el proyecto Java, hacer clic derecho en Libraries y seleccionar Add Jar/Folder… luego seleccionar:

C:\BuenosAires\VentaBA-Java\Libraries\json-simple-1.1.1.jar

También se pueden descargar desde:

<https://code.google.com/archive/p/json-simple/downloads>

1. En el proyecto Java, hacer clic derecho en Libraries y seleccionar Add Jar/Folder… luego seleccionar:

C:\BuenosAires\VentaBA-Java\Libraries\ gson-2.13.1.jar

También se pueden descargar desde: <https://search.maven.org/artifact/com.google.code.gson/gson/2.13.1/jar?eh=>

**Agregar referencia a Webservice SOAP WsAutenticacion**

1. Hacer clic derecho en el proyecto BuenosAires.VentaBA y seleccionar New 🡪 Web Service client…
2. Selecciona el radio button “WSDL URL” y pegar el link del WSDL de WsAutenticacion, ¡CUIDADO que este link dependerá del dominio y puerto donde Visual Studio haya ejecutado la Capa de Servicio. En mi caso es:

<http://localhost:54096/WsProducto.svc?singleWsdl>

1. Seleccionar paquete buenosaires.proxy y aceptar.
2. Se agregará la carpeta “Web Service References” y dentro de ella aparecerá WsAutenticacion.
3. **Posteriormente deberá programar la API Rest Django y Web Service SOAP que muestran los productos de la bodega y presentarlos en una ventana Java como se indica en los Anexos del Caso de Estudio (esto es parte del desarrollo que debe hacer para el Control 2).**

**Crear clases del Webservice**

1. Abrir una consola CMD y ejecutar los siguientes comandos:

cd C:\BuenosAires\VentaBA-Java\BuenosAires.VentaBA

mkdir build\generated-sources\jax-ws

"C:\Program Files\Java\jdk-1.8\bin\wsimport.exe" -keep ^

-s build/generated-sources/jax-ws ^

-d build/classes ^

-p buenosaires.proxy ^

-Xnocompile ^

http://localhost:54096/WsAutenticacion.svc?singleWsdl

1. Estas instrucciones crearán las clases Java del Webservice SOAP

**Crear paquete Java para almacenar las clases de Service Client**

1. Hacer clic derecho en el proyecto BuenosAires.VentaBA y seleccionar New 🡪 Java package…
2. Dar el nombre buenosaires.proxy al paquete.

**Crear clase para serializar y deserializar en Json**

Clic derecho en buenosaires.proxy, seleccionar New 🡪 Java class…, crear JsonSerializer y pegar este código:

package buenosaires.proxy;

import com.google.gson.Gson;

import java.text.ParseException;

import java.util.Map;

import org.json.simple.JSONObject;

import org.json.simple.parser.JSONParser;

public class JsonSerializer {

    private static final Gson gson = new Gson();

    /\*\*

     \* Serializa un objeto Java simple (por ejemplo, Map, List, String, Integer...) a un String JSON.

     \*/

    public static String toJson(Object object) {

        if (object instanceof Map) {

            return JSONObject.toJSONString((Map<?, ?>) object);

        } else {

            throw new IllegalArgumentException("Sólo se pueden serializar objetos tipo Map con json-simple.");

        }

    }

    /\*\*

     \* Deserializa un String JSON a un Map<String, Object>

     \*/

    @SuppressWarnings("unchecked")

    public static Map<String, Object> fromJsonToMap(String json) throws ParseException, org.json.simple.parser.ParseException {

        JSONParser parser = new JSONParser();

        Object obj = parser.parse(json);

        if (obj instanceof JSONObject) {

            return (Map<String, Object>) obj;

        } else {

            throw new IllegalArgumentException("El JSON no representa un objeto tipo JSONObject.");

        }

    }

    /\*\*

     \* Serializa un objeto Java a una cadena JSON.

     \*/

    public static String serialize(Object object) {

        return gson.toJson(object);

    }

    /\*\*

     \* Deserializa una cadena JSON a un objeto del tipo especificado.

     \*/

    public static <T> T deserialize(String json, Class<T> clazz) {

        return gson.fromJson(json, clazz);

    }

}

**Crear clase para serializar y deserializar en XML**

Clic derecho en buenosaires.proxy, seleccionar New 🡪 Java class…, crear XmlSerializer y pegar este código:

package buenosaires.proxy;

import java.beans.XMLDecoder;

import java.beans.XMLEncoder;

import java.io.ByteArrayInputStream;

import java.io.ByteArrayOutputStream;

import java.io.StringReader;

import java.io.StringWriter;

import java.util.ArrayList;

import javax.xml.bind.JAXBContext;

import javax.xml.bind.JAXBException;

import javax.xml.bind.Marshaller;

import javax.xml.bind.Unmarshaller;

import javax.xml.stream.XMLInputFactory;

import javax.xml.stream.XMLStreamConstants;

import javax.xml.stream.XMLStreamException;

import javax.xml.stream.XMLStreamReader;

public class XmlSerializer {

    public static String serialize(Object object) throws JAXBException {

        JAXBContext context = JAXBContext.newInstance(object.getClass());

        Marshaller marshaller = context.createMarshaller();

        marshaller.setProperty(Marshaller.JAXB\_FORMATTED\_OUTPUT, true);

        StringWriter writer = new StringWriter();

        marshaller.marshal(object, writer);

        String xml = writer.toString();

        return xml;

    }

    public static <T> T deserialize(String xml, Class<T> clazz) throws JAXBException {

        JAXBContext context = JAXBContext.newInstance(clazz);

        Unmarshaller unmarshaller = context.createUnmarshaller();

        StringReader reader = new StringReader(xml);

        return clazz.cast(unmarshaller.unmarshal(reader));

    }

    public static <T> T deserialize(String xml, Class<T> clazz, String contextPath) throws JAXBException {

        JAXBContext context = JAXBContext.newInstance(contextPath);

        Unmarshaller unmarshaller = context.createUnmarshaller();

        StringReader reader = new StringReader(xml);

        return clazz.cast(unmarshaller.unmarshal(reader));

    }

    public static String toXMLString(Object obj) {

        ByteArrayOutputStream baos = new ByteArrayOutputStream();

        XMLEncoder e = new XMLEncoder(baos);

        e.writeObject(obj);

        e.close();

        return new String(baos.toByteArray());

    }

    public static Object fromXMLString(String str) {

        XMLDecoder d = new XMLDecoder(new ByteArrayInputStream(str.getBytes()));

        Object obj = (Object) d.readObject();

        d.close();

        return obj;

    }

    public static ArrayList<String> getXmlElelemnts(String xml, String xmlElement) throws XMLStreamException {

        ArrayList<String> productoStrings = new ArrayList<>();

        XMLInputFactory xmlInputFactory = XMLInputFactory.newInstance();

        XMLStreamReader xmlStreamReader = xmlInputFactory.createXMLStreamReader(new StringReader(xml));

        String productoXml = null;

        while (xmlStreamReader.hasNext()) {

            int event = xmlStreamReader.next();

            switch (event) {

                case XMLStreamConstants.START\_ELEMENT:

                    if (xmlStreamReader.getLocalName().equals(xmlElement)) {

                        productoXml = "<" + xmlElement + ">";

                    } else if (productoXml != null) {

                        productoXml += "<" + xmlStreamReader.getLocalName() + ">";

                    }

                    break;

                case XMLStreamConstants.END\_ELEMENT:

                    if (xmlStreamReader.getLocalName().equals(xmlElement)) {

                        productoXml += "</" + xmlElement + ">";

                        productoStrings.add(productoXml);

                        productoXml = null;

                    } else if (productoXml != null) {

                        productoXml += "</" + xmlStreamReader.getLocalName() + ">";

                    }

                    break;

                case XMLStreamConstants.CHARACTERS:

                    if (productoXml != null) {

                        productoXml += xmlStreamReader.getText();

                    }

                    break;

                default:

                    break;

            }

        }

        return productoStrings;

    }

}

**Crear clase para el Service Client del Webservice SOAP WsAutenticacion**

Clic derecho en buenosaires.proxy, seleccionar New 🡪 Java class…, crear ScAutenticacion y pegar este código:

package buenosaires.proxy;

import org.json.simple.JSONObject;

import org.json.simple.JSONValue;

public class ScAutenticacion {

    private boolean autenticado;

    private String nombreUsuario;

    private String tipoUsuario;

    private String mensaje;

    public ScAutenticacion() {

        this.autenticado = false;

        this.nombreUsuario = "";

        this.tipoUsuario = "";

        this.mensaje = "";

    }

    public boolean isAutenticado() {

        return autenticado;

    }

    public void setAutenticado(boolean autenticado) {

        this.autenticado = autenticado;

    }

    public String getNombreUsuario() {

        return nombreUsuario;

    }

    public void setNombreUsuario(String nombreUsuario) {

        this.nombreUsuario = nombreUsuario;

    }

    public String getTipoUsuario() {

        return tipoUsuario;

    }

    public void setTipoUsuario(String tipoUsuario) {

        this.tipoUsuario = tipoUsuario;

    }

    public String getMensaje() {

        return mensaje;

    }

    public void setMensaje(String mensaje) {

        this.mensaje = mensaje;

    }

    public void ejecutarPruebas() {

        //Acceder al Web Service

        WsAutenticacion ws = new WsAutenticacion();

        IWsAutenticacion port = ws.getBasicHttpBindingIWsAutenticacion();

        // Probar con una cuenta de Vendedor

        Respuesta resp1 = port.autenticar("Vendedor", "bgates", "123");

        String json1 = resp1.getJsonAutenticado().getValue();

        System.out.println(json1);

        // Probar con una cuenta de Administrador

        Respuesta resp2 = port.autenticar("Administrador", "emusk", "123");

        String json2 = resp2.getJsonAutenticado().getValue();

        System.out.println(json2);

        // Probar con una cuenta de Bodeguero

        Respuesta resp3 = port.autenticar("Bodeguero", "emusk", "123");

        String json3 = resp3.getJsonAutenticado().getValue();

        System.out.println(json3);

        //Probar ScAutenticacion

        var bc = new ScAutenticacion();

        bc.autenticar("Vendedor", "bgates", "123");

        System.out.println(bc.toString());

    }

    public void autenticar(String tipousu, String username, String password) {

        var ws = new WsAutenticacion();

        var port = ws.getBasicHttpBindingIWsAutenticacion();

        Respuesta resp = port.autenticar(tipousu, username, password);

        String jsonAutenticado = resp.getJsonAutenticado().getValue();

        Object jsonObject = JSONValue.parse(jsonAutenticado);

        JSONObject jsonAutenticacion = (JSONObject) jsonObject;

        this.setAutenticado((Boolean)jsonAutenticacion.get("Autenticado"));

        this.setNombreUsuario((String)jsonAutenticacion.get("NombreUsuario"));

        this.setTipoUsuario((String)jsonAutenticacion.get("TipoUsuario"));

        this.setMensaje((String)jsonAutenticacion.get("Mensaje"));

    }

    @Override

    public String toString() {

        return "ScAutenticacion{"

                + "autenticado=" + autenticado

                + ", nombreUsuario=" + nombreUsuario

                + ", tipoUsuario=" + tipoUsuario

                + ", mensaje=" + mensaje + '}';

    }

}

**Crear código Java para abrir ventana de login**

En el archivo ventaba.java pegar este código:

package buenosaires.ventaba;

public class ventaba {

    public static void main(String[] args) {

        //new buenosaires.proxy.ScAutenticacion().ejecutarPruebas();

        VentanaLogin ventana = new VentanaLogin();

        ventana.setLocationRelativeTo(null);

        ventana.setVisible(true);

    }

}

**Crear VentanaLogin.java**

En la carpeta C:\BuenosAires\VentaBA-Java\BuenosAires.VentaBA\src\buenosaires\ventaba\VentanaLogin.java

Crear el archivo VentanaLogin.java y pegar este código:

package buenosaires.ventaba;

import buenosaires.proxy.ScAutenticacion;

import javax.swing.JOptionPane;

public class VentanaLogin extends javax.swing.JFrame {

    public VentanaLogin() {

        initComponents();

    }

    /\*\*

     \* This method is called from within the constructor to initialize the form.

     \* WARNING: Do NOT modify this code. The content of this method is always

     \* regenerated by the Form Editor.

     \*/

    @SuppressWarnings("unchecked")

    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents

    private void initComponents() {

        lblNombreSistema = new javax.swing.JLabel();

        lblTituloVentana = new javax.swing.JLabel();

        lblCuenta = new javax.swing.JLabel();

        lblPassword = new javax.swing.JLabel();

        txtCuenta = new javax.swing.JTextField();

        btnIngresar = new javax.swing.JButton();

        txtPassword = new javax.swing.JPasswordField();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

        setTitle("Iniciar sesión");

        lblNombreSistema.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N

        lblNombreSistema.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

        lblNombreSistema.setText("Sistema Buenos Aires");

        lblTituloVentana.setFont(new java.awt.Font("Segoe UI", 1, 14)); // NOI18N

        lblTituloVentana.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

        lblTituloVentana.setText("Ingresar al Sistema de Ventas");

        lblCuenta.setText("Cuenta");

        lblPassword.setText("Contraseña");

        btnIngresar.setText("Ingresar");

        btnIngresar.addActionListener(new java.awt.event.ActionListener() {

            public void actionPerformed(java.awt.event.ActionEvent evt) {

                btnIngresarActionPerformed(evt);

            }

        });

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

        getContentPane().setLayout(layout);

        layout.setHorizontalGroup(

            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addGroup(layout.createSequentialGroup()

                .addGap(95, 95, 95)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

                    .addComponent(lblNombreSistema, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

                    .addComponent(lblTituloVentana, javax.swing.GroupLayout.DEFAULT\_SIZE, 216, Short.MAX\_VALUE)

                    .addGroup(layout.createSequentialGroup()

                        .addComponent(lblPassword)

                        .addGap(18, 18, 18)

                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

                            .addComponent(txtCuenta, javax.swing.GroupLayout.DEFAULT\_SIZE, 138, Short.MAX\_VALUE)

                            .addComponent(txtPassword)))

                    .addGroup(javax.swing.GroupLayout.Alignment.LEADING, layout.createSequentialGroup()

                        .addComponent(lblCuenta)

                        .addGap(2, 2, 2)

                        .addComponent(btnIngresar, javax.swing.GroupLayout.PREFERRED\_SIZE, 150, javax.swing.GroupLayout.PREFERRED\_SIZE)))

                .addContainerGap(111, Short.MAX\_VALUE))

        );

        layout.setVerticalGroup(

            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addGroup(layout.createSequentialGroup()

                .addGap(47, 47, 47)

                .addComponent(lblNombreSistema)

                .addGap(18, 18, 18)

                .addComponent(lblTituloVentana)

                .addGap(26, 26, 26)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(lblCuenta)

                    .addComponent(txtCuenta, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(lblPassword)

                    .addComponent(txtPassword, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

                .addGap(29, 29, 29)

                .addComponent(btnIngresar)

                .addContainerGap(56, Short.MAX\_VALUE))

        );

        pack();

    }// </editor-fold>//GEN-END:initComponents

    private void btnIngresarActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_btnIngresarActionPerformed

        var bc = new ScAutenticacion();

        bc.autenticar("Vendedor", txtCuenta.getText(), txtPassword.getText());

        if (bc.isAutenticado()) {

            JOptionPane.showMessageDialog(null, "AUTENTICADO"

                , "Buenos Aires", JOptionPane.INFORMATION\_MESSAGE);

        }

        else {

            JOptionPane.showMessageDialog(null, bc.getMensaje()

                , "Buenos Aires", JOptionPane.INFORMATION\_MESSAGE);

        }

    }//GEN-LAST:event\_btnIngresarActionPerformed

    /\*\*

     \* @param args the command line arguments

     \*/

    public static void main(String args[]) {

        /\* Set the Nimbus look and feel \*/

        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

        /\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

         \* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

         \*/

        try {

            for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

                if ("Nimbus".equals(info.getName())) {

                    javax.swing.UIManager.setLookAndFeel(info.getClassName());

                    break;

                }

            }

        } catch (ClassNotFoundException ex) {

            java.util.logging.Logger.getLogger(VentanaLogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

        } catch (InstantiationException ex) {

            java.util.logging.Logger.getLogger(VentanaLogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

        } catch (IllegalAccessException ex) {

            java.util.logging.Logger.getLogger(VentanaLogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

            java.util.logging.Logger.getLogger(VentanaLogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

        }

        //</editor-fold>

        //</editor-fold>

        /\* Create and display the form \*/

        java.awt.EventQueue.invokeLater(new Runnable() {

            public void run() {

                new VentanaLogin().setVisible(true);

            }

        });

    }

    // Variables declaration - do not modify//GEN-BEGIN:variables

    private javax.swing.JButton btnIngresar;

    private javax.swing.JLabel lblCuenta;

    private javax.swing.JLabel lblNombreSistema;

    private javax.swing.JLabel lblPassword;

    private javax.swing.JLabel lblTituloVentana;

    private javax.swing.JTextField txtCuenta;

    private javax.swing.JPasswordField txtPassword;

    // End of variables declaration//GEN-END:variables

}

En la carpeta C:\BuenosAires\VentaBA-Java\BuenosAires.VentaBA\src\buenosaires\ventaba\VentanaLogin.java

Crear el archivo VentanaLogin.form y pegar este código:

<?xml version="1.0" encoding="UTF-8" ?>

<Form version="1.3" maxVersion="1.9" type="org.netbeans.modules.form.forminfo.JFrameFormInfo">

  <Properties>

    <Property name="defaultCloseOperation" type="int" value="3"/>

    <Property name="title" type="java.lang.String" value="Iniciar sesi&#xf3;n"/>

  </Properties>

  <SyntheticProperties>

    <SyntheticProperty name="formSizePolicy" type="int" value="1"/>

    <SyntheticProperty name="generateCenter" type="boolean" value="false"/>

  </SyntheticProperties>

  <AuxValues>

    <AuxValue name="FormSettings\_autoResourcing" type="java.lang.Integer" value="0"/>

    <AuxValue name="FormSettings\_autoSetComponentName" type="java.lang.Boolean" value="false"/>

    <AuxValue name="FormSettings\_generateFQN" type="java.lang.Boolean" value="true"/>

    <AuxValue name="FormSettings\_generateMnemonicsCode" type="java.lang.Boolean" value="false"/>

    <AuxValue name="FormSettings\_i18nAutoMode" type="java.lang.Boolean" value="false"/>

    <AuxValue name="FormSettings\_layoutCodeTarget" type="java.lang.Integer" value="1"/>

    <AuxValue name="FormSettings\_listenerGenerationStyle" type="java.lang.Integer" value="0"/>

    <AuxValue name="FormSettings\_variablesLocal" type="java.lang.Boolean" value="false"/>

    <AuxValue name="FormSettings\_variablesModifier" type="java.lang.Integer" value="2"/>

  </AuxValues>

  <Layout>

    <DimensionLayout dim="0">

      <Group type="103" groupAlignment="0" attributes="0">

          <Group type="102" alignment="0" attributes="0">

              <EmptySpace min="-2" pref="95" max="-2" attributes="0"/>

              <Group type="103" groupAlignment="1" max="-2" attributes="0">

                  <Component id="lblNombreSistema" alignment="1" max="32767" attributes="0"/>

                  <Component id="lblTituloVentana" alignment="1" pref="216" max="32767" attributes="0"/>

                  <Group type="102" alignment="1" attributes="0">

                      <Component id="lblPassword" min="-2" max="-2" attributes="0"/>

                      <EmptySpace type="separate" max="-2" attributes="0"/>

                      <Group type="103" groupAlignment="0" max="-2" attributes="0">

                          <Component id="txtCuenta" pref="138" max="32767" attributes="0"/>

                          <Component id="txtPassword" max="32767" attributes="0"/>

                      </Group>

                  </Group>

                  <Group type="102" alignment="0" attributes="0">

                      <Component id="lblCuenta" min="-2" max="-2" attributes="0"/>

                      <EmptySpace min="-2" pref="2" max="-2" attributes="0"/>

                      <Component id="btnIngresar" min="-2" pref="150" max="-2" attributes="0"/>

                  </Group>

              </Group>

              <EmptySpace pref="111" max="32767" attributes="0"/>

          </Group>

      </Group>

    </DimensionLayout>

    <DimensionLayout dim="1">

      <Group type="103" groupAlignment="0" attributes="0">

          <Group type="102" alignment="0" attributes="0">

              <EmptySpace min="-2" pref="47" max="-2" attributes="0"/>

              <Component id="lblNombreSistema" min="-2" max="-2" attributes="0"/>

              <EmptySpace type="separate" max="-2" attributes="0"/>

              <Component id="lblTituloVentana" min="-2" max="-2" attributes="0"/>

              <EmptySpace min="-2" pref="26" max="-2" attributes="0"/>

              <Group type="103" groupAlignment="3" attributes="0">

                  <Component id="lblCuenta" alignment="3" min="-2" max="-2" attributes="0"/>

                  <Component id="txtCuenta" alignment="3" min="-2" max="-2" attributes="0"/>

              </Group>

              <EmptySpace type="unrelated" max="-2" attributes="0"/>

              <Group type="103" groupAlignment="3" attributes="0">

                  <Component id="lblPassword" alignment="3" min="-2" max="-2" attributes="0"/>

                  <Component id="txtPassword" alignment="3" min="-2" max="-2" attributes="0"/>

              </Group>

              <EmptySpace min="-2" pref="29" max="-2" attributes="0"/>

              <Component id="btnIngresar" min="-2" max="-2" attributes="0"/>

              <EmptySpace pref="56" max="32767" attributes="0"/>

          </Group>

      </Group>

    </DimensionLayout>

  </Layout>

  <SubComponents>

    <Component class="javax.swing.JLabel" name="lblNombreSistema">

      <Properties>

        <Property name="font" type="java.awt.Font" editor="org.netbeans.beaninfo.editors.FontEditor">

          <Font name="Segoe UI" size="18" style="1"/>

        </Property>

        <Property name="horizontalAlignment" type="int" value="0"/>

        <Property name="text" type="java.lang.String" value="Sistema Buenos Aires"/>

      </Properties>

    </Component>

    <Component class="javax.swing.JLabel" name="lblTituloVentana">

      <Properties>

        <Property name="font" type="java.awt.Font" editor="org.netbeans.beaninfo.editors.FontEditor">

          <Font name="Segoe UI" size="14" style="1"/>

        </Property>

        <Property name="horizontalAlignment" type="int" value="0"/>

        <Property name="text" type="java.lang.String" value="Ingresar al Sistema de Ventas"/>

      </Properties>

    </Component>

    <Component class="javax.swing.JLabel" name="lblCuenta">

      <Properties>

        <Property name="text" type="java.lang.String" value="Cuenta"/>

      </Properties>

    </Component>

    <Component class="javax.swing.JLabel" name="lblPassword">

      <Properties>

        <Property name="text" type="java.lang.String" value="Contrase&#xf1;a"/>

      </Properties>

    </Component>

    <Component class="javax.swing.JTextField" name="txtCuenta">

    </Component>

    <Component class="javax.swing.JButton" name="btnIngresar">

      <Properties>

        <Property name="text" type="java.lang.String" value="Ingresar"/>

      </Properties>

      <Events>

        <EventHandler event="actionPerformed" listener="java.awt.event.ActionListener" parameters="java.awt.event.ActionEvent" handler="btnIngresarActionPerformed"/>

      </Events>

    </Component>

    <Component class="javax.swing.JPasswordField" name="txtPassword">

    </Component>

  </SubComponents>

</Form>

## En este programa fuente ya se realizó la tarea de programar Web Service SOAP de Producto (en C#)

Abrir la solución C:\BuenosAires\BodegaBA-CSharp\BuenosAires.sln

**Crear Webservice de Producto en el Proyecto BuenosAires.ServiceLayer**

Haz clic derecho en el proyecto BuenosAires.ServiceLayer, luego seleccionar Agregar 🡪 Nuevo elemento y en el buscador encontrar la plantilla **“Servicio WCF”** y darle el nombre WsProducto.svc, luego presionar el botón Agregar. Luego de este paso **se habrá creado el Webservice SOAP de Producto**, que cuenta con 2 archivos: **IWsProducto.cs (que es la interface)** y **WsProducto.svc (que es la implementación)**

**Programación de Interface, Class y Web.config del Servicio en BuenosAires.ServiceLayer**

En el archivo **IWsProducto.cs** copiar el siguiente código:

using BuenosAires.Model;

using System.ServiceModel;

namespace BuenosAires.ServiceLayer

{

    [ServiceContract]

    public interface IWsProducto

    {

        [OperationContract]

        Respuesta Crear(Producto producto);

        [OperationContract]

        Respuesta LeerTodos();

        [OperationContract]

        Respuesta Leer(int id);

        [OperationContract]

        Respuesta Actualizar(Producto producto);

        [OperationContract]

        Respuesta Eliminar(int id);

    }

}

En el archivo **WsProducto.svc** copiar el siguiente código:

using BuenosAires.BusinessLayer;

using BuenosAires.Model.Utiles;

using BuenosAires.Model;

namespace BuenosAires.ServiceLayer

{

    public class WsProducto : IWsProducto

    {

        private Respuesta ObtenerRespuesta(BcProducto bc)

        {

            var respuesta = new Respuesta();

            respuesta.Accion = bc.Accion;

            respuesta.Mensaje = bc.Mensaje;

            respuesta.HayErrores = bc.HayErrores;

            respuesta.XmlProducto = Util.SerializarXML(bc.Producto);

            respuesta.XmlListaProducto = Util.SerializarXML(bc.Lista);

            return respuesta;

        }

        public Respuesta Crear(Producto producto)

        {

            var bc = new BcProducto();

            bc.Crear(producto);

            return ObtenerRespuesta(bc);

        }

        public Respuesta LeerTodos()

        {

            var bc = new BcProducto();

            bc.LeerTodos();

            return ObtenerRespuesta(bc);

        }

        public Respuesta Leer(int id)

        {

            var bc = new BcProducto();

            bc.Leer(id);

            return ObtenerRespuesta(bc);

        }

        public Respuesta Actualizar(Producto producto)

        {

            var bc = new BcProducto();

            bc.Actualizar(producto);

            return ObtenerRespuesta(bc);

        }

        public Respuesta Eliminar(int id)

        {

            var bc = new BcProducto();

            bc.Eliminar(id);

            return ObtenerRespuesta(bc);

        }

    }

}

En el archivo Web.config agregar <connectionString> justo debajo de <configuration>, debe quedar así:

*<?xml version="1.0" encoding="utf-8"?>*

*<configuration>*

<connectionStrings>

<add name="base\_datosEntities" connectionString="metadata=res://\*/BuenosAiresEDM.csdl|res://\*/BuenosAiresEDM.ssdl|res://\*/BuenosAiresEDM.msl;provider=System.Data.SqlClient;provider connection string=&quot;data source=z80\SQLEXPRESS;initial catalog=base\_datos;user id=sa;password=123;trustservercertificate=True;MultipleActiveResultSets=True;App=EntityFramework&quot;" providerName="System.Data.EntityClient" />

</connectionStrings>

*<appSettings>*

*... aquí continúa el resto del archivo, que debe quedar tal como estaba ...*

**Agregar Anotación para Propiedades no Serializables en BuenosAires.Model**

En la clase **Producto** (archivo Producto.cs dentro de BuenosAiresEDM.tt) se deben marcar como XmlIgnore todas las propiedades que no son serializables, de modo que cuando se transforme a XML un objeto de la clase, no considere los ICollection:

*<?xml version="1.0" encoding="utf-8"?>*

*<configuration>*

*[System.Xml.Serialization.XmlIgnore]*

*[System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]*

*public virtual ICollection<Factura> Factura { get; set; }*

*[System.Xml.Serialization.XmlIgnore]*

*[System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]*

*public virtual ICollection<GuiaDespacho> GuiaDespacho { get; set; }*

*[System.Xml.Serialization.XmlIgnore]*

*[System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]*

*public virtual ICollection<StockProducto> StockProducto { get; set; }*

Esta tarea debe ser realizada en todos los ICollection de todas las clases del Modelo que están dentro del archivo **BuenosAiresEDM.tt**

**Agregar Service Client de Producto (ScProducto) en BuenosAires.BodegaBA**

En el proyecto BuenosAires.BodegaBA crear la clase **ScProducto** y copiarle el siguiente código:

using System.Collections.Generic;

using BuenosAires.Model.Utiles;

using BuenosAires.Model;

using System;

using BuenosAires.BodegaBA.WsProductoReference;

namespace BuenosAires.ServiceProxy

{

    public class ScProducto

    {

        public string Accion = "";

        public string Mensaje = "";

        public bool HayErrores = false;

        public Producto Producto = null;

        public List<Producto> Lista = null;

        public void CopiarPropiedades(Respuesta resp)

        {

            this.Accion = resp.Accion;

            this.Mensaje = resp.Mensaje;

            this.HayErrores = resp.HayErrores;

            this.Producto = Util.DeserializarXML<Producto>(resp.XmlProducto);

            this.Lista = Util.DeserializarXML<List<Producto>>(resp.XmlListaProducto);

        }

        public WsProductoClient getWs()

        {

            var ws = new WsProductoClient();

            ws.InnerChannel.OperationTimeout = new TimeSpan(1, 0, 0);

            return ws;

        }

        public void Crear(Producto producto)

        {

            CopiarPropiedades(getWs().Crear(producto));

        }

        public void LeerTodos()

        {

            CopiarPropiedades(getWs().LeerTodos());

        }

        public void Leer(int id)

        {

            CopiarPropiedades(getWs().Leer(id));

        }

        public void Actualizar(Producto producto)

        {

            CopiarPropiedades(getWs().Actualizar(producto));

        }

        public void Eliminar(int id)

        {

            CopiarPropiedades(getWs().Eliminar(id));

        }

    }

}

**Corregir VentanaProducto en BuenosAires.BodegaBA para que use ScProducto en vez de BcProducto**

1. Quitar la referencia a BuenosAires.BusinessLayer en BuenosAires.BodegaBA, ya que ahora a capa de presentación va a comenzar a utilizar la capa de servicio en vez de la capa de negocio.
2. Cambiar todos los lugares donde aparezca BcProducto por ScProducto:

using System.Windows.Forms;

using BuenosAires.ServiceProxy;

using BuenosAires.Model;

using BuenosAires.Model.Utiles;

namespace BuenosAires.BodegaBA

{

    public partial class VentanaProducto : Form

    {

        public VentanaProducto(string cuenta)

        {

            InitializeComponent();

            this.Text = $"Mantenedor de productos - Usuario: {cuenta}";

            btnBuscar.Click += (sender, e) => Buscar();

            btnNuevo.Click += (sender, e) => Nuevo();

            btnGuardar.Click += (sender, e) => Guardar();

            btnEliminar.Click += (sender, e) => Eliminar();

            btnCargarProductos.Click += (sender, e) => CargarProductos();

            grid.SelectionChanged += (sender, e) => Seleccionar();

            grid.ConfigurarDataGridView(

                  "idprod:ID, "

                + "nomprod:Nombre, "

                + "descprod:Descripción, "

                + "precio:Precio, "

                + "imagen:Imagen"

            );

            CargarProductos();

            this.CentrarVentana();

        }

        private void Nuevo()

        {

            this.Limpiar(new TextBox[]

            {

                txtIdProd,

                txtNomProd,

                txtDescProd,

                txtPrecio,

                txtImagen

            });

        }

        public bool ValidarCamposNumericos()

        {

            if (txtIdProd.Text != "" && !txtIdProd.EsNumero())

            {

                return this.ErrEntero("ID");

            }

            if (!txtPrecio.EsNumero())

            {

                return this.ErrEntero("Precio");

            }

            return true;

        }

        private void Seleccionar()

        {

            if (grid.SelectedRows.Count <= 0) return;

            DataGridViewRow row = grid.SelectedRows[0];

            if (row.GetString("idprod") != "0")

            {

                this.AsignarValoresTextBox(row);

            }

            txtNomProd.FocusToEnd();

        }

        private bool Buscar()

        {

            int id = new VentanaBuscarID().MostrarVentanaModal();

            if (id == -1) return false;

            var bc = new ScProducto();

            bc.Leer(id);

            if (bc.Producto == null) return this.MensajeInfo(bc.Mensaje);

            CargarProductos();

            this.AsignarValoresTextBox(bc.Producto);

            grid.SeleccionarId("idprod", txtIdProd.ToInt());

            txtNomProd.FocusToEnd();

            return true;

        }

        private void Guardar()

        {

            if (!ValidarCamposNumericos()) return;

            var prod = new Producto();

            prod.idprod = txtIdProd.ToIntOrDefault();

            prod.nomprod = txtNomProd.Text;

            prod.descprod = txtDescProd.Text;

            prod.precio = txtPrecio.ToInt();

            prod.imagen = txtImagen.Text;

            var bc = new ScProducto();

            if (txtIdProd.Text.Trim() == "")

            {

                bc.Crear(prod);

            }

            else

            {

                bc.Actualizar(prod);

            }

            if (!bc.HayErrores)

            {

                txtIdProd.SetText(bc.Producto.idprod);

                CargarProductos();

                grid.SeleccionarId("idprod", bc.Producto.idprod);

                txtNomProd.FocusToEnd();

            }

            this.MensajeInfo(bc.Mensaje);

        }

        private bool Eliminar()

        {

            var bc = new ScProducto();

            if (txtIdProd.Text.Trim() == "")

            {

                return this.ErrAccionID("ID", "eliminar");

            }

            bc.Eliminar(txtIdProd.ToInt());

            CargarProductos();

            this.MensajeInfo(bc.Mensaje);

            return true;

        }

        public void CargarProductos()

        {

            var bc = new ScProducto();

            bc.LeerTodos();

            grid.DataSource = bc.Lista;

            grid.RefrescarYajustar();

            if (bc.HayErrores == true) this.MensajeInfo(bc.Mensaje);

        }

    }

}

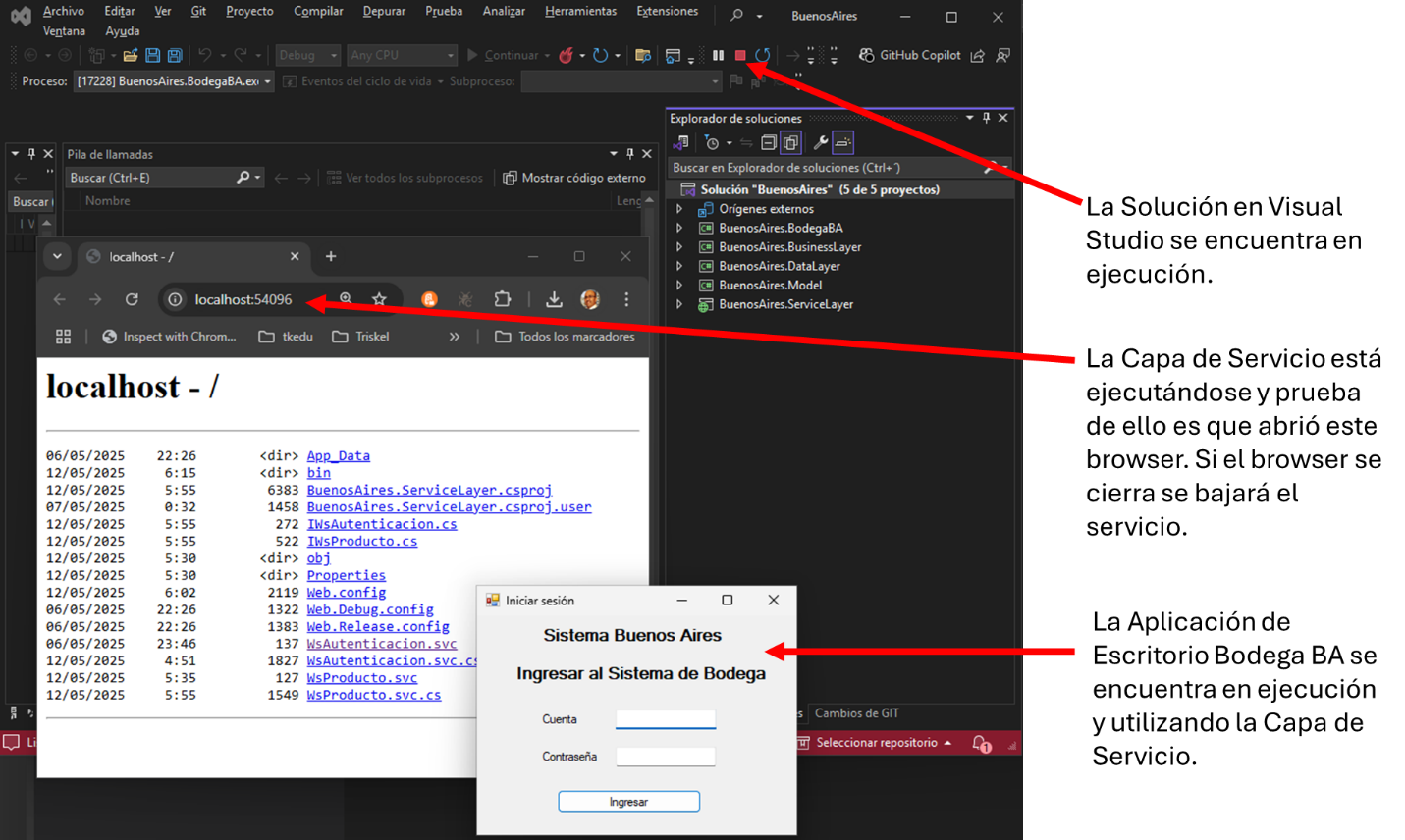
**Ejecutar proyectos de BodegaBA y Capa de Servicios**

1. Hacer clic derecho en **Solución “BuenosAires”** y seleccionar propiedades.
2. Seleccionar radio button **“Varios proyectos de inicio”.**
3. Usando los botones de flechas subir a la parte de más arriba el proyecto BuenosAires.ServiceLayer y seleccionar la columna Acción con el valor Inicio.
4. Dejar en segundo lugar el proyecto BuenosAires.BodegaBA y seleccionar la columna Acción con el valor Inicio.

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

1. Luego presionar Aplicar y luego Aceptar.
2. De esta forma cuando se ejecute F5 para arrancar la solución completa, primero se ejecutara IIS el servidor de aplicaciones de Microsoft para dejar activa la Capa de Servicio y luego se ejecutara la aplicación de escritorio BodegaBA.
3. La pantalla del computador se verá parecida a la siguiente:



## En este programa fuente ya se realizó la tarea de crear la Capa de Servicio con un Web Service SOAP de Autenticación (en C#) que use la API Rest de Autenticación (en Django)

Abrir la solución C:\BuenosAires\BodegaBA-CSharp\BuenosAires.sln

**Agregar Referencias y Archivos en el Proyecto BuenosAires.Model**

1. Haz clic derecho sobre el proyecto BuenosAires.Model, luego Agregar 🡪 Referencia, y seleccionar:

Desde ensamblados : System.Runtime.Serialization

1. Haz clic derecho sobre el proyecto BuenosAires.Model, luego Agregar 🡪 Elemento existente, y seleccionar:

Archivo : Respuesta.cs

**Crear Proyecto BuenosAires.ServiceLayer**

1. Haz clic derecho sobre **Solución “BuenosAires”** y seleccionar Agregar 🡪 Nuevo proyecto
2. Buscar la plantilla **Aplicación de Servicios WCF** (Un proyecto para crear una aplicación de servicio WCF que se hospeda en IIS/WAS)
3. Crear proyecto:

Nombre del proyecto : BuenosAires.ServiceLayer

Ubicación : C:\BuenosAires\BodegaBA-CSharp

Framework : .NET Framework 4.8

Proyecto se creará en "C:\BuenosAires\BodegaBA-CSharp\BuenosAires.ServiceLayer\"

1. Una vez creado el proyecto, eliminar los archivos de ejemplo IService1.cs y Service1.svc

**Agregar Referencias y Archivos en el Proyecto BuenosAires.ServiceLayer**

1. Hacer clic derecho en el proyecto BuenosAires.ServiceLayer, luego Agregar 🡪 Referencia, y seleccionar:

Desde proyectos : BuenosAires.BusinessLayer

Desde proyectos : BuenosAires.Model

Desde examinar buscar : C:\BuenosAires\BodegaBA-CSharp\packages\EntityFramework.6.5.1\lib\net45

y marcar EntityFramework.dll y EntityFramework.SqlServer.dll

Desde ensamblados : System.Net.Http

1. Haz clic derecho en el proyecto BuenosAires.ServiceLayer, luego seleccionar Agregar 🡪 Nuevo elemento y en el buscador encontrar la plantilla **“Servicio WCF”** y darle el nombre WsAutenticacion.svc, luego presionar el botón Agregar. Luego de este paso **se habrá creado el Webservice SOAP de Autenticación**, que cuenta con 2 archivos: **IWsAutenticacion.cs (que es la interface)** y **WsAutenticacion.svc (que es la implementación)**

**Programación de la Interface y la Class del Servicio**

1. En el archivo IWsAutenticacion.cs copiar el siguiente código:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Runtime.Serialization;

using System.ServiceModel;

using System.Text;

using BuenosAires.Model;

namespace BuenosAires.ServiceLayer

{

    [ServiceContract]

    public interface IWsAutenticacion

    {

        [OperationContract]

        Respuesta Autenticar(string tipousu, string username, string password);

    }

}

1. Hacer doble clic en el archivo WsAutenticacion.svc y copiar el siguiente código:

using System;

using System.Net.Http;

using BuenosAires.Model.Utiles;

using BuenosAires.Model;

using BuenosAires.BusinessLayer;

using BuenosAires.ServiceLayer;

namespace BuenosAires.ServiceLayer

{

    public class WsAutenticacion : IWsAutenticacion

    {

        public Respuesta Autenticar(string tipousu, string username, string password)

        {

            var resp = new Respuesta();

            resp.Accion = "autenticar al usuario";

            resp.Mensaje = "";

            resp.HayErrores = false;

            resp.JsonAutenticado = "";

            if (username.Trim() == "") username = "no-fue-digitado-el-username";

            if (password.Trim() == "") password = "no-fue-digitada-la-password";

            string apiUrl = "http://127.0.0.1:8001/BuenosAiresApiRest/autenticar/" + tipousu + "/" + username + "/" + password;

            try

            {

                using (HttpClient client = new HttpClient())

                {

                    HttpResponseMessage response = client.GetAsync(apiUrl).Result;

                    if (response.IsSuccessStatusCode)

                    {

                        resp.JsonAutenticado = response.Content.ReadAsStringAsync().Result;

                    }

                    else

                    {

                        resp.HayErrores = true;

                        resp.Mensaje = "No fue posible autenticar al usuario, intente nuevamente más tarde "

                            + "o comuníquese con el Administrador del Sistema";

                    }

                    return resp;

                }

            }

            catch (Exception ex)

            {

                resp.HayErrores = true;

                resp.Mensaje = Util.MensajeError(resp.Accion, "WsAutenticacion.Autenticar", ex);

                return resp;

            }

        }

    }

}

**Agregar Webservice SOAP al Proyecto BuenosAires.BodegaBA**

1. Haz clic derecho en el proyecto BuenosAires.BodegaBA, luego seleccionar Agregar 🡪 Referencia de servicio
2. Presionar el botón Detectar (que va a buscar los Webservices SOAP en los proyectos de la solución)
3. En el combobox Dirección seleccionar el servicio WsAutenticacion
4. En el Espacio de nombres reemplazar la palabra ServiceReference1 por WsAutenticacionReference y Aceptar

**Crear el Service Client ScAutenticacion en el Proyecto BuenosAires.BodegaBA**

1. Haz clic derecho en el proyecto BuenosAires.BodegaBA, luego seleccionar Agregar 🡪 Nuevo elemento y crear la clase **ScAutenticacion**
2. En la clase ScAutenticacion escribir el siguiente código:

using System.Collections.Generic;

using BuenosAires.Model.Utiles;

using BuenosAires.Model;

using System;

using BuenosAires.BodegaBA.WsAutenticacionReference;

using Newtonsoft.Json;

namespace BuenosAires.ServiceProxy

{

    public class ScAutenticacion

    {

        public class RespuestaAutenticacion

        {

            public bool Autenticado { get; set; }

            public string NombreUsuario { get; set; }

            public string TipoUsuario { get; set; }

            public string Mensaje { get; set; }

        }

        public string Accion = "";

        public string Mensaje = "";

        public bool HayErrores = false;

        public string JsonAutenticado = "";

        public bool Autenticado = false;

        public string NombreUsuario = "";

        public string TipoUsuario = "";

        public void CopiarPropiedades(Respuesta resp)

        {

            this.Accion = resp.Accion;

            this.Mensaje = resp.Mensaje;

            this.HayErrores = resp.HayErrores;

            this.JsonAutenticado = resp.JsonAutenticado;

            this.Autenticado = false;

            this.NombreUsuario = "";

            this.TipoUsuario = "";

            if (resp.JsonAutenticado != "")

            {

                RespuestaAutenticacion respAutenticacion =

                    JsonConvert.DeserializeObject<RespuestaAutenticacion>(resp.JsonAutenticado);

                this.Autenticado = respAutenticacion.Autenticado;

                this.NombreUsuario = respAutenticacion.NombreUsuario;

                this.TipoUsuario = respAutenticacion.TipoUsuario;

                this.Mensaje = respAutenticacion.Mensaje;

            }

        }

        public WsAutenticacionClient GetWs()

        {

            var ws = new WsAutenticacionClient();

            ws.InnerChannel.OperationTimeout = new TimeSpan(1, 0, 0);

            return ws;

        }

        public void Autenticar(string tipousu, string username, string password)

        {

            CopiarPropiedades(GetWs().Autenticar(tipousu, username, password));

        }

    }

}

**Corregir la Ventana Login para que use el Cliente del WebService SOAP (ScAutenticacion)**

1. En el código fuente de VentanaLogin.cs escribir el siguiente código:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using BuenosAires.BodegaBA;

using BuenosAires.Model.Utiles;

using BuenosAires.ServiceProxy;

namespace BuenosAires.BodegaBA

{

    public partial class VentanaLogin : Form

    {

        public VentanaLogin()

        {

            InitializeComponent();

            this.AcceptButton = btnIngresar;

            this.StartPosition = FormStartPosition.CenterScreen;

        }

        private void btnIngresar\_Click(object sender, EventArgs e)

        {

            var sc = new ScAutenticacion();

            sc.Autenticar("Bodeguero", txtCuenta.Text, txtPassword.Text);

            if (sc.Autenticado)

            {

                new VentanaProducto(sc.NombreUsuario + "(" + sc.TipoUsuario + ")").Show();

                Hide();

            }

            else

            {

                this.MensajeInfo(sc.Mensaje);

            }

        }

    }

}

**Corregir la Forma en que se Inician los Proyectos**

1. Haz clic derecho en **Solución “BuenosAires”**, luego seleccionar **Propiedades**
2. Seleccionar la opción **“Varios proyectos de inicio”**
3. Subir a la primera posición el Proyecto BuenosAires.ServiceLayer. Esto es importante, pues el proyecto BodegaBA usa el Webservice SOAP del proyecto ServiceLayer y por esta razón primero debe arrancar el servidor de aplicaciones que tiene implementado el Webservice
4. El orden de los proyectos debe quedar así:

BuenosAires.ServiceLayer : Inicio

BuenosAires.BodegaBA : Inicio

BuenosAires.Model : Ninguno

BuenosAires.DataLayer : Ninguno

BuenosAires.BusinessLayer: Ninguno

1. **Estamos a punto de arrancar nuestra solución con F5, pero antes debes tener corriendo el Servidor de Aplicaciones de Django, pues el Webservice SOAP de Autenticacion va a invocar a la API Rest Autenticar de AppWebBA.**
2. **Verifica también que tu programa C# está apuntando al servidor de Base de Datos correcto, recuerda que no es el mismo servidor si estás en tu casa o en un laboratorio.**
3. Una vez levantada la aplicación web Django puedes ejecutar F5 en el proyecto BodegaBA
4. Ahora puedes escribir una cuenta y una password y verás como la ventana de login valida la cuenta usando las funciones de Django **¡HAS LOGRADO INTEGRAR UN PROGRAMA EN DJANGO CON UNO EN C# .NET!**
5. El paso siguiente será implementar WsProducto y usarlo en VentanaProducto de BodegaBA

## En este programa fuente ya se realizó la tarea de crear una API Rest de Autenticación

Los pasos que se siguieron para crear la API Rest fueron los siguientes:

1. Abrir proyecto AppWebBA y crear ambiente virtual:

* Abrir la carpeta "C:\BuenosAires\AppWebBA-Django" en Visual Studio Code
* Abrir una consola en la carpeta: C:\BuenosAires\AppWebBA-Django
* Crear el ambiente virtual ejecutando: create-venv.bat

1. Prepara la base de datos de tu computador:

* Abre SQL Server Management Studio y selecciona:

Tipo de servidor : Motor de base de datos

Nombre del servidor : Habitualmente es el nombre del computador o el nombre\SQLEXPRESS

Autenticación : Autenticación de Windows

Si te lo pide, acepta credenciales de seguridad

* Haz clic derecho sobre el nombre del servidor, selecciona Seguridad y marca "Modo de autenticación de Windows y SQL Server"
* A la izquierda en Conexiones marca "Permitir conexiones remotas con este servidor"
* Haz clic derecho sobre el nombre del servidor y selecciona "Reiniciar"
* Entra al servidor en la carpeta Seguridad->Inicios de sesión y haz doble clic en "sa", luego cambia la contraseña por "123" y quita la marca de "Exigir directivas de contraseña".
* Luego selecciona a la izquierda "Estado" y marca Inicio de sesión como "Habilitada"
* Haz clic derecho en la carpeta "Base de datos" y crea una base de datos llamada "base\_datos"
* Actualiza las cadenas de conexión a la BD ejecutando: python update-db-host.py nombre-de-servidor
* Crea la BD con sus datos de prueba ejecutando: create-bd.bat

1. Crear una nueva Aplicación Django para implementar las API Rest de AppWebBA:

* Activa el ambiente virtual y ejecuta en consola: "python manage.py startapp apirest"
* Dentro de AppWebBA en settings.py agrega "apirest" al final de INSTALLED\_APPS
* Dentro de AppWebBA en urls.py agrega en urlpatterns lo siguiente:

path('BuenosAiresApiRest/', include('apirest.urls')),

* Dentro de la carpeta "apirest" crea el archivo "urls.py" y escribe lo siguiente:

from django.urls import path

from .views import autenticar

urlpatterns = [

    path('autenticar/<tipousu>/<username>/<password>', autenticar, name="autenticar"),

    #path('obtener\_equipos\_en\_bodega', obtener\_equipos\_en\_bodega, name='obtener\_equipos\_en\_bodega'),

    #path('obtener\_productos', obtener\_productos, name='obtener\_productos'),

]

* Vamos a implementar la API Rest "autenticar" y más adelante tendrás que hacer las que están comentadas
* En views.py escribe lo siguiente:

from rest\_framework.decorators import api\_view

from django.views.decorators.csrf import csrf\_exempt

from django.http import JsonResponse

from django.contrib.auth import authenticate

from core.models import PerfilUsuario

@csrf\_exempt

@api\_view(['GET'])

def autenticar(request, tipousu, username, password):

    user = authenticate(username=username, password=password)

    if user:

        perfil = PerfilUsuario.objects.get(user=user)

        nombre = f'{user.first\_name} {user.last\_name}'

        tipo = perfil.tipousu

        if tipo in [tipousu, 'Administrador']:

            return JsonResponse({'Autenticado': True, 'NombreUsuario': nombre, 'TipoUsuario': tipo, 'Mensaje': ''})

        msg = f'La cuenta de usuario {nombre} es del perfil {tipo}, por lo que no puede ingresar al sistema'

    else:

        nombre, tipo, msg = '', '', 'La cuenta o la contraseña no coinciden con un usuario válido'

    return JsonResponse({'Autenticado': False, 'NombreUsuario': nombre, 'TipoUsuario': tipo, 'Mensaje': msg})

* Ahora puedes probar la API Rest de autenticación en un browser, con las siguientes pruebas:

http://127.0.0.1:8001/BuenosAiresApiRest/autenticar/Bodeguero/creyes/123

http://127.0.0.1:8001/BuenosAiresApiRest/autenticar/Bodeguero/creyes/password-mala

http://127.0.0.1:8001/BuenosAiresApiRest/autenticar/Bodeguero/atorres/123

http://127.0.0.1:8001/BuenosAiresApiRest/autenticar/Administrador/atorres/123

http://127.0.0.1:8001/BuenosAiresApiRest/autenticar/Administrador/emusk/123