

# Control Parking

SERVICIOS WEB

## 1. Introducción

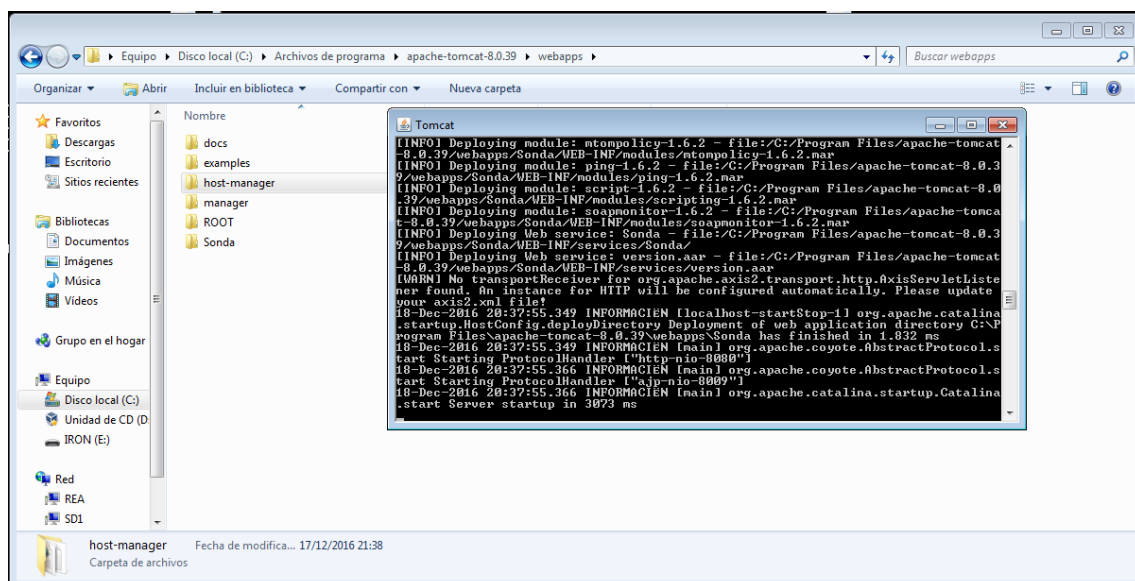
El presente documento tiene como objetivo presentar un despliegue de la aplicación Control Parking mediante el uso de la tecnología de Servicios web. La primera parte del documento consistirá en una explicación detallada del despliegue de la aplicación en un sistema distribuido. La segunda parte del documento explicará brevemente las funcionalidades de porciones de código determinantes durante el desarrollo de la aplicación.

## 2. Guía de despliegue

Para realizar la simulación en un entorno distribuido se ejecutarán los programas ya compilados en las siguientes máquinas:

- [Windows 1: Sonda 1 \(192.168.1.132\)](#)

Primero se dará de alta la primera Sonda ejecutando el servidor Apache Tomcat en el que se encuentra el documento wsdl con toda la información necesaria para poder acceder a esta Sonda. La sonda tendrá los valores predeterminados indicados en el enunciado de la práctica.



La sonda ya se encuentra activa y esperando peticiones.

- [Windows 2: Sonda 2 \(127.0.0.1\)](#)

Seguidamente se dará de alta una segunda Sonda de igual forma que la primera con el valor del volumen al 50%. Por falta de recursos se realizará un despliegue de la segunda sonda en el mismo equipo donde se ejecutará el cliente.

```

una completa lista de los JARs que fueron explorados y de los que nos se hallan TLDs. Saltarse JARs no necesarios durante
la exploración puede dar lugar a una mejora de tiempo significativa en el arranque y compilación de JSP.
[WARN] Web application uses org.apache.axis2.transport.http.AxisAdminServlet; please update web.xml to use org.apache.axis2.webapp.AxisAdminServlet instead
[INFO] Clustering has been disabled
[INFO] Deploying module: addressing-1.6.2 - file:/D:/Program Files/apache-tomcat-8.0.39/webapps/Sonda/WEB-INF/modules/addressing-1.6.2.mar
[INFO] Deploying module: metadataExchange-1.6.2 - file:/D:/Program Files/apache-tomcat-8.0.39/webapps/Sonda/WEB-INF/modules/metadataExchange-1.6.2.mar
[INFO] Deploying module: mtompolicy-1.6.2 - file:/D:/Program Files/apache-tomcat-8.0.39/webapps/Sonda/WEB-INF/modules/mtompolicy-1.6.2.mar
[INFO] Deploying module: ping-1.6.2 - file:/D:/Program Files/apache-tomcat-8.0.39/webapps/Sonda/WEB-INF/modules/ping-1.6.2.mar
[INFO] Deploying module: script-1.6.2 - file:/D:/Program Files/apache-tomcat-8.0.39/webapps/Sonda/WEB-INF/modules/script-1.6.2.mar
[INFO] Deploying module: soapmonitor-1.6.2 - file:/D:/Program Files/apache-tomcat-8.0.39/webapps/Sonda/WEB-INF/modules/soapmonitor-1.6.2.mar
[INFO] Deploying Web service: Sonda - file:/D:/Program Files/apache-tomcat-8.0.39/webapps/Sonda/WEB-INF/services/Sonda/version.aar
[INFO] Deploying Web service: version.aar - file:/D:/Program Files/apache-tomcat-8.0.39/webapps/Sonda/WEB-INF/services/version.aar
[WARN] No transportReceiver for org.apache.axis2.transport.http.AxisServletListener found. An instance for HTTP will be configured automatically. Please update your axis2.xml file!
18-Dec-2016 20:55:20.964 INFORMACIÓN [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory D:\Program Files\apache-tomcat-8.0.39\webapps\Sonda has finished in 1.367 ms
18-Dec-2016 20:55:20.966 INFORMACIÓN [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-apr-8080"]
18-Dec-2016 20:55:20.973 INFORMACIÓN [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["ajp-apr-8009"]
18-Dec-2016 20:55:20.977 INFORMACIÓN [main] org.apache.catalina.startup.Catalina.start Server startup in 1981 ms
  
```

- [Windows 2: Cliente](#)

Por último inicializamos el cliente que consumirá los servicios de las Sondas desde el ejecutable .exe asociado.

The screenshot shows a simple login interface. It has a title bar with 'Login' and standard window controls. Below the title bar, there are two text input fields: 'Usuario' and 'Contraseña'. Below these fields are two buttons: 'Entrar' (highlighted with a blue border) and 'Registrar usuario'.

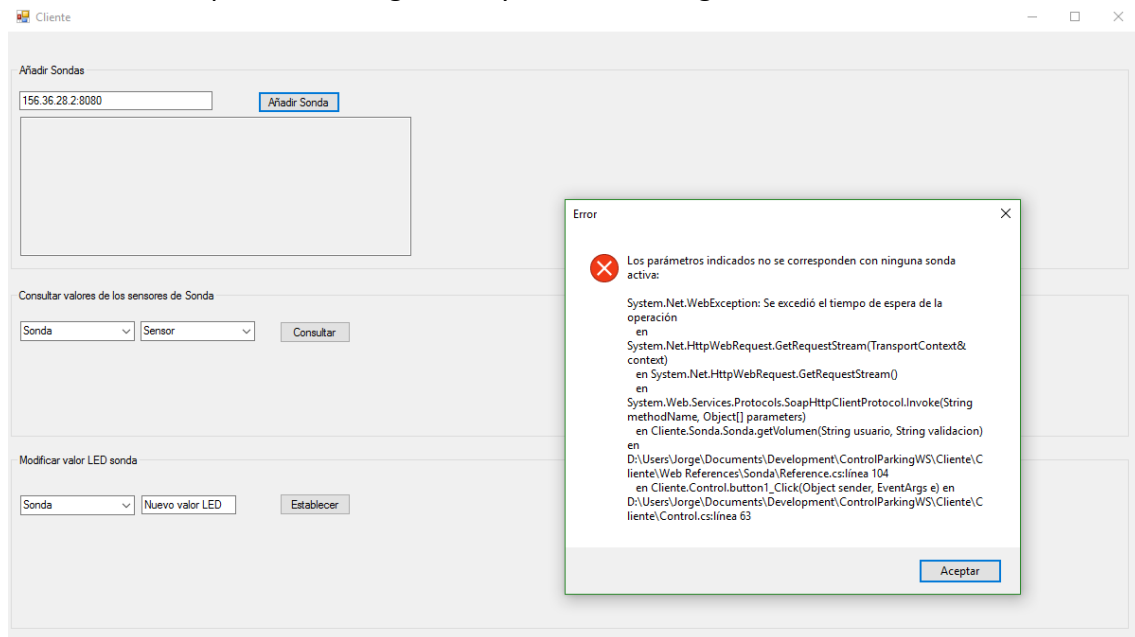
Una vez ejecutados todos los componentes pasamos a comprobar el funcionamiento de la aplicación.

- El primer paso es entrar con el usuario y la contraseña, en este caso el usuario dado de alta en todos los componentes es admin::admin, las contraseñas se guardan en un archivo usuarios.txt que almacena usuario::contraseña utilizando un método de encriptación Hash SHA-5.

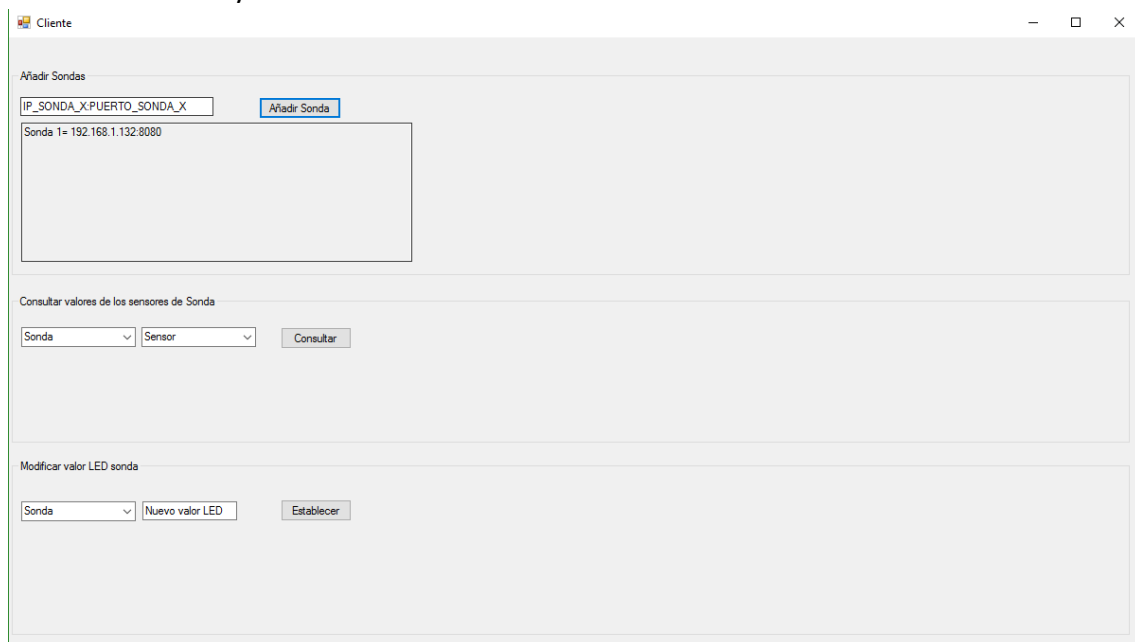
```

usuarios.txt
1
x61EY612K12gPFL56FT9weDnpSo4AV8j8+qx2AuThdRyY036xxzTTw10Wq3+4QyB+XURPwX10Nxp3Y3pB37A==:x61EY612K12gPFL56FT9weDnpSo4AV8j8+qx2AuThdRyY036xxzTTw10Wq3+4QyB+XURPwX10Nxp3Y3pB37A==
sUHQEwHj/1UnAPbRDw/7eLz5TTP3j3cWt195Klq9nbBe6L1wK02Q4ED/shRDCBpWcHtsh0JcA0niw4g==:sUHQEwHj/1UnAPbRDw/7eLz5TTP3j3cWt195Klq9nbBe6L1wK02Q4ED/shRDCBpWcHtsh0JcA0niw4g==
0TSYUo/vYE1ZkcyhsG0ZRnuHcgepy5EGZ1tmf3s2+o4A933b1/ZDZ09pqsuNbIVgnDwOm5f0yqE1P5pyMkbYnA==:0TSYUo/vYE1ZkcyhsG0ZRnuHcgepy5EGZ1tmf3s2+o4A933b1/ZDZ09pqsuNbIVgnDwOm5f0yqE1P5pyMkbYnA==
0TSYUo/vYE1ZkcyhsG0ZRnuHcgepy5EGZ1tmf3s2+o4A933b1/ZDZ09pqsuNbIVgnDwOm5f0yqE1P5pyMkbYnA==:0TSYUo/vYE1ZkcyhsG0ZRnuHcgepy5EGZ1tmf3s2+o4A933b1/ZDZ09pqsuNbIVgnDwOm5f0yqE1P5pyMkbYnA==
0TSYUo/vYE1ZkcyhsG0ZRnuHcgepy5EGZ1tmf3s2+o4A933b1/ZDZ09pqsuNbIVgnDwOm5f0yqE1P5pyMkbYnA==:0TSYUo/vYE1ZkcyhsG0ZRnuHcgepy5EGZ1tmf3s2+o4A933b1/ZDZ09pqsuNbIVgnDwOm5f0yqE1P5pyMkbYnA==
  
```

- b) Una vez dentro de la aplicación vamos a probar a introducir una sonda que no esté registrada y mostrará el siguiente error:



- c) Ahora introduciremos la sonda desplegada en Windows 1 y se mostrará como registrada de la siguiente forma, cabe anotar que a cada sonda se le asignará un id automáticamente dependiendo del orden en el que se haya introducido:



- d) Ahora se mostrará el resultado de consultar cualquier valor en la Sonda, por ejemplo, el LED.

Consultar valores de los sensores de Sonda

Sonda 1 ▼ Led ▼ Consultar

4500

- e) Seguidamente cambiaremos el valor del LED a 200:

Modificar valor LED sonda

Sonda 1 ▼ 200 Establecer

El valor de la Sonda 1 se ha actualizado correctamente al valor: 200

- f) Ahora para comprobar que se ha cambiado correctamente el valor se consultará de nuevo el LED de la Sonda:

Consultar valores de los sensores de Sonda

Sonda 1 ▼ Led ▼ Consultar

200

- g) Seguidamente desplegamos la segunda Sonda en local:

Añadir Sondas

IP\_SONDA\_X:PUERTO\_SONDA\_X Añadir Sonda

Sonda 1= 192.168.1.132:8080  
Sonda 2= 127.0.0.1:8080

- h) Consultamos el volumen de la Sonda y el valor del LED

Consultar valores de los sensores de Sonda

Sonda 2 ▼ Volumen ▼ Consultar

50

Consultar valores de los sensores de Sonda

Sonda 2    Led    Consultar

4500

- i) Los archivos Log que muestran las acciones de la sesión son los siguientes:

Sonda 1:

```
18/12/2016 21:11:58, SD1/192.168.1.132, admin, Escritura en el
fichero sonda.txt, Se modifica el valor LED
18/12/2016 21:13:00, SD1/192.168.1.132, admin, Peticion del
valor del LED, Se envia correctamente el valor del LED
```

Sonda 2:

```
29 18/12/2016 21:15:01, MACHINA/192.168.56.1, admin, Peticion de volumen, Se envia el volumen de la Sonda
30 18/12/2016 21:16:15, MACHINA/192.168.56.1, admin, Peticion de volumen, Se envia el volumen de la Sonda
31 18/12/2016 21:16:36, MACHINA/192.168.56.1, admin, Peticion del valor del LED, Se envia correctamente el valor del LED
```

Cliente:

```
126 18/12/2016 20:02:33, 192.168.1.129, admin, Acceso al sistema
127 18/12/2016 20:04:26, 192.168.1.129, admin, Añadir sonda, Error en los parámetros Sonda: 156.36.28.2:8080
128 18/12/2016 20:04:54, 192.168.1.129, admin, Añadir sonda, Parámetros Sonda añadida: 192.168.1.132:8080
129 18/12/2016 20:06:37, 192.168.1.129, admin, Consultar sensor, Consulta del valor LED en Sonda 1
130 18/12/2016 20:07:28, 192.168.1.129, admin, Modificar LED, Error en la modificación del sensor de la Sonda 1
131 18/12/2016 20:07:33, 192.168.1.129, admin, Modificar LED, Error en la modificación del sensor de la Sonda 1
132 18/12/2016 20:07:47, 192.168.1.129, admin, Modificar LED, Error en la modificación del sensor de la Sonda 1
133 18/12/2016 20:11:58, 192.168.1.129, admin, Modificar LED, Modificación del sensor de la Sonda 1
134 18/12/2016 20:13:00, 192.168.1.129, admin, Consultar sensor, Consulta del valor LED en Sonda 1
135 18/12/2016 20:15:01, 192.168.1.129, admin, Añadir sonda, Parámetros Sonda añadida: 127.0.0.1:8080
136 18/12/2016 20:16:15, 192.168.1.129, admin, Consultar sensor, Consulta del volumen en Sonda 2
137 18/12/2016 20:16:36, 192.168.1.129, admin, Consultar sensor, Consulta del valor LED en Sonda 2
```

### 3. Descripción de los componentes Software

a) Aplicación Cliente - c#

Se compone de dos WindowsForm, el Login.cs se encarga de registrar los usuarios y guardarlos en el fichero usuarios.txt con un encriptado SHA. Al intentar acceder al sistema también comprueba que el usuario introducido exista en el archivo usuarios.txt.

```
try
{
    string input = Hash(textBox1.Text) + ":" + Hash(textBox2.Text);
    StreamWriter file = new StreamWriter(@"\usuarios.txt", true);
    file.WriteLine(input);
    file.Close();
    inputLog("Registro de usuario", "Usuario " + textBox1.Text + " registrado correctamente");
    MessageBox.Show("Usuario registrado correctamente", "Registro de usuario", MessageBoxButtons.OK, MessageBoxIcon.Information);
}
```

```

public string Hash(string password)
{
    var bytes = new UTF8Encoding().GetBytes(password);
    byte[] hashBytes;
    using (var algorithm = new System.Security.Cryptography.SHA512Managed())
    {
        hashBytes = algorithm.ComputeHash(bytes);
    }
    return Convert.ToBase64String(hashBytes);
}

```

La segunda parte contiene el panel de control de la aplicación donde se permite añadir Sondas, consultar los valores de los sensores y modificar los valores LED de las Sondas.

La comunicación se realiza mediante una encriptación AES/CBC:

```

public RijndaelManaged GetRijndaelManaged(String secretKey)
{
    var keyBytes = new byte[16];
    var secretKeyBytes = Encoding.UTF8.GetBytes(secretKey);
    Array.Copy(secretKeyBytes, keyBytes, Math.Min(keyBytes.Length, secretKeyBytes.Length));
    return new RijndaelManaged
    {
        Mode = CipherMode.CBC,
        Padding = PaddingMode.PKCS7,
        KeySize = 128,
        BlockSize = 128,
        Key = keyBytes,
        IV = keyBytes
    };
}

1 referencia | Jorge Espinosa, Hace 1 día | 1 autor, 1 cambio
public byte[] Encrypt(byte[] plainBytes, RijndaelManaged rijndaelManaged)
{
    return rijndaelManaged.CreateEncryptor()
        .TransformFinalBlock(plainBytes, 0, plainBytes.Length);
}

1 referencia | Jorge Espinosa, Hace 1 día | 1 autor, 1 cambio
public byte[] Decrypt(byte[] encryptedData, RijndaelManaged rijndaelManaged)
{
    return rijndaelManaged.CreateDecryptor()
        .TransformFinalBlock(encryptedData, 0, encryptedData.Length);
}

7 referencias | Jorge Espinosa, Hace 20 horas | 1 autor, 1 cambio
public String Encrypt(String plainText)
{
    String key = readKey();
    var plainBytes = Encoding.UTF8.GetBytes(plainText);
    return Convert.ToBase64String(Encrypt(plainBytes, GetRijndaelManaged(key)));
}

```

## b) Aplicación Sonda – JAVA

Accede a los archivos donde están guardados los datos de las Sondas y los manipula, para poder acceder a sus métodos requiere de un usuario que esté registrado para poder modificar los valores de la Sonda.

Encripta las comunicaciones con el mismo algoritmo que la aplicación Cliente AES/CBC:

```
public String getVolumen(String usuario, String validacion) {
    Encriptador crypt = new Encriptador();

    if(buscaUsuario(validacion)) {
        String mensaje = "";

        try {
            this.usuario = crypt.decrypt(usuario, key);
            mensaje = crypt.encrypt(volumen, key);
        } catch(Exception e) {
            System.out.println("Error al encriptar el volumen: " + e.toString());
            inputLog("Petición de volumen", "Error enviando el volumen de la Sonda");
        }
        inputLog("Petición de volumen", "Se envía el volumen de la Sonda");
        return mensaje;
    } else {
        inputLog("Petición de volumen", "Error de validación de usuario: se envía mensaje de error");
        try {
            return crypt.encrypt("ErrorValidacion", key);
        } catch(Exception e) {
            System.out.println("Error al encriptar el volumen: " + e.toString());
            inputLog("Petición de volumen", "Error enviando el volumen de la Sonda");
        }
    }
    return null;
}

public byte[] decrypt(byte[] cipherText, byte[] key, byte[] initialVector)
    throws NoSuchAlgorithmException, NoSuchPaddingException,
        InvalidKeyException, InvalidAlgorithmParameterException,
        IllegalBlockSizeException, BadPaddingException {
    Cipher cipher = Cipher.getInstance(cipherTransformation);
    SecretKeySpec secretKeySpec = new SecretKeySpec(key,
        aesEncryptionAlgorithm);
    IvParameterSpec ivParameterSpec = new IvParameterSpec(initialVector);
    cipher.init(Cipher.DECRYPT_MODE, secretKeySpec, ivParameterSpec);
    cipherText = cipher.doFinal(cipherText);
    return cipherText;
}

public byte[] encrypt(byte[] plainText, byte[] key, byte[] initialVector)
    throws NoSuchAlgorithmException, NoSuchPaddingException,
        InvalidKeyException, InvalidAlgorithmParameterException,
        IllegalBlockSizeException, BadPaddingException {
    Cipher cipher = Cipher.getInstance(cipherTransformation);
    SecretKeySpec secretKeySpec = new SecretKeySpec(key,
        aesEncryptionAlgorithm);
    IvParameterSpec ivParameterSpec = new IvParameterSpec(initialVector);
    cipher.init(Cipher.ENCRYPT_MODE, secretKeySpec, ivParameterSpec);
    plainText = cipher.doFinal(plainText);
    return plainText;
}
```