

# Registro de documento WSDL en UDDI (actualización para juddi v3)

En este ejercicio se describe un ejemplo de publicación de una hoja WSDL en un registro UDDI. En concreto, para el ejemplo se utilizará la hoja WSDL adjuntada al ejercicio y la implementación del registro UDDI jUDDI.

El ejemplo se realizará utilizando la aplicación de escritorio soapUI que se adjunta en la sesión. El proceso de registro se puede resumir como se muestra en la imagen.

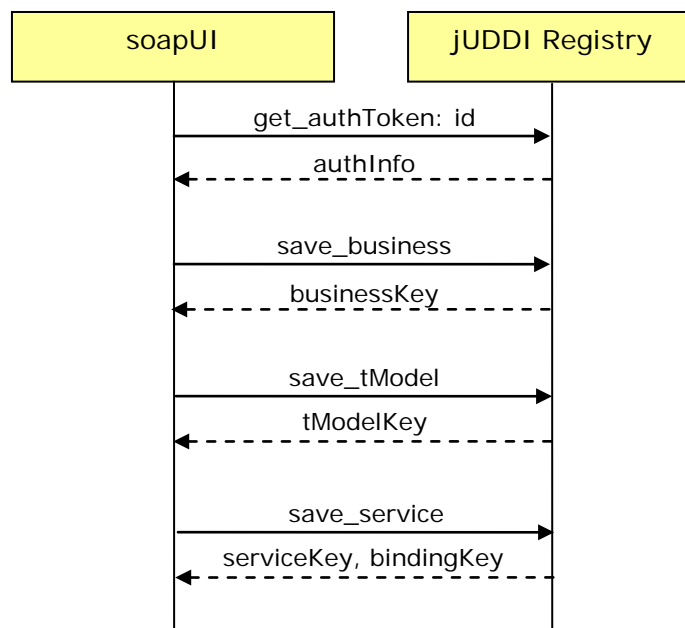
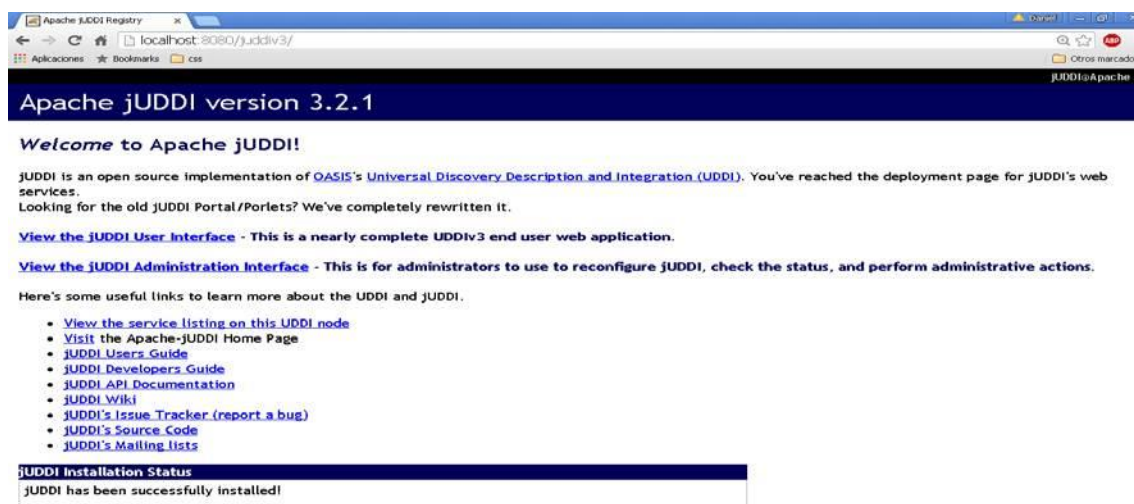


Figura 1. Proceso de registro de un servicio Web en jUDDI.

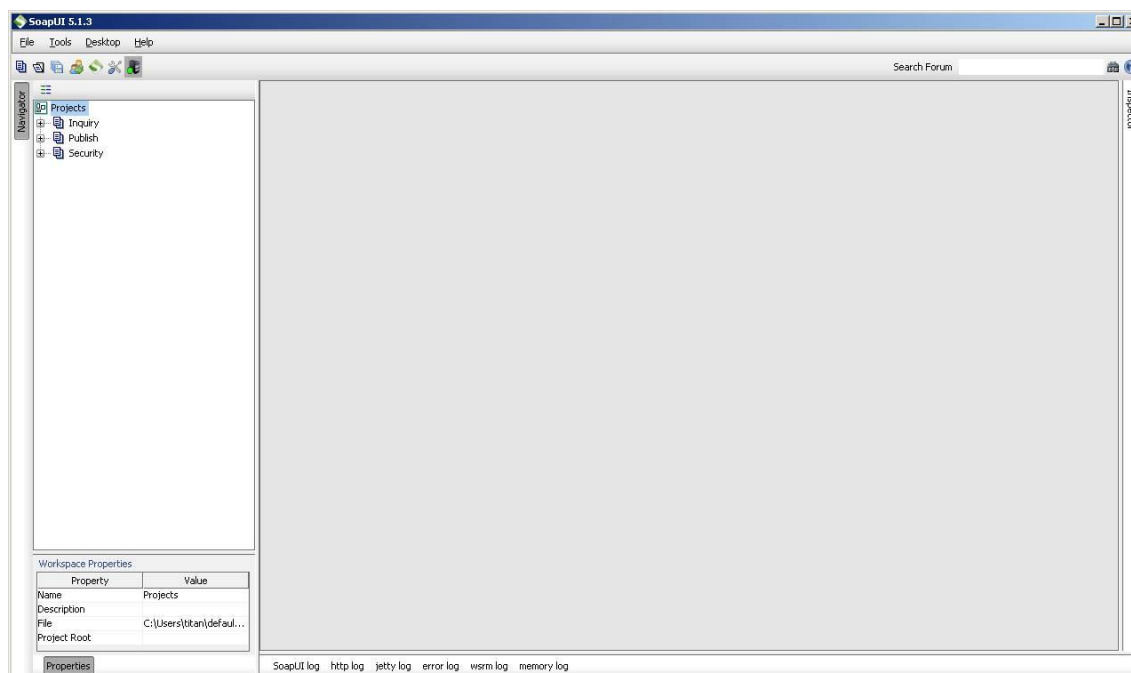
En primer lugar, se debe lanzar el registro jUDDI. Para ello debemos situarnos en la carpeta de Juddi/bin, una vez estemos en la carpeta ejecutamos en modo administrador el archivo startup.bat.



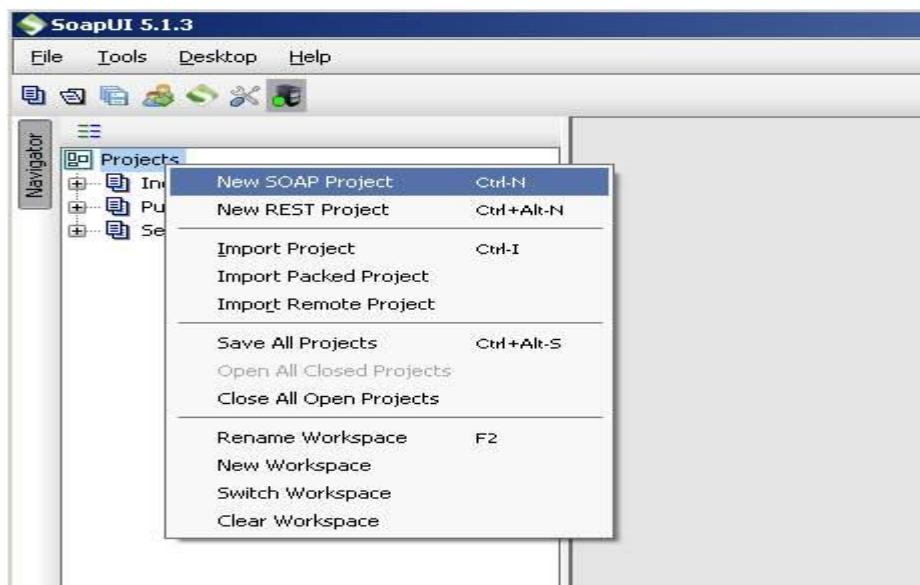
La distribución de juddi v3 no incluye, como sucedía en versiones anteriores, una interfaz Web que permita enviar mensajes SOAP para interactuar con los servicios que ofrece. En este caso utilizaremos una aplicación externa llamada soapUI en su versión 5.1.3.

Esta herramienta nos permitirá interactuar con los servicios que ofrece juddi de forma muy rápida y sencilla y familiarizarnos con la estructura de los mensajes SOAP.

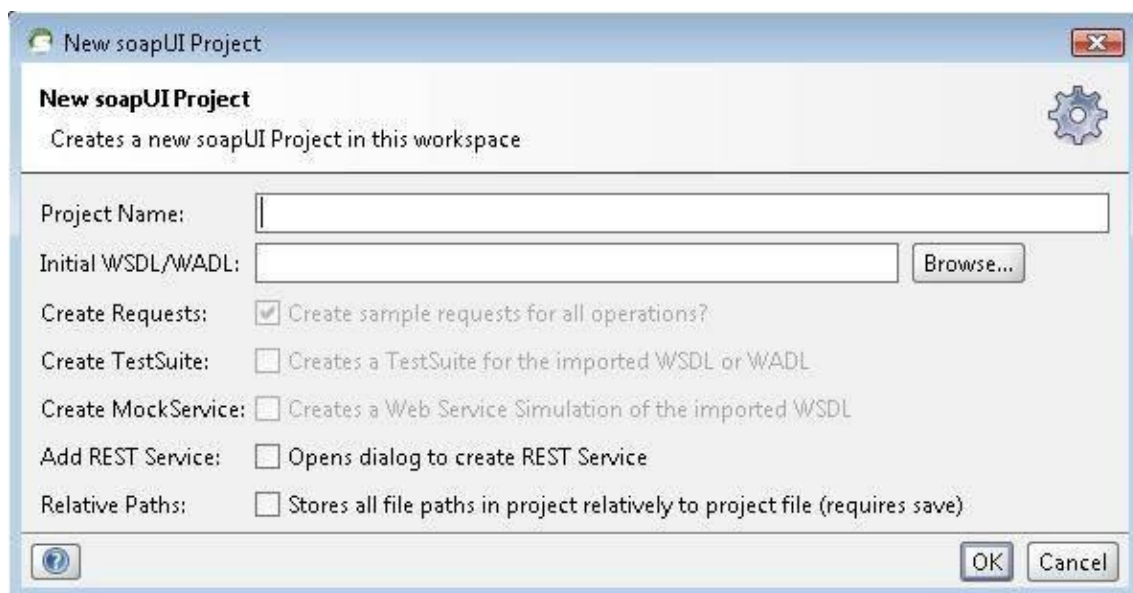
Como la aplicación soapUI no está integrada en juddi lo primero que debemos hacer es crear los proyectos en soapUI para poder acceder a los servicios que ofrece juddi. Para ello arrancamos la aplicación soapUI (en el directorio *bin* ejecutar el script *soapui.bat*) o bien mediante un acceso directo al inicio del programa.



Una vez abierta la aplicación, seleccionamos el icono *Projects* que aparece en el panel izquierdo, en la esquina superior y pulsamos con el botón derecho del ratón. Seleccionamos la opción *New soapUI Project*.



A continuación aparecerá una venta donde debemos introducir la URL del documento WSDL que describe los servicios que ofrece juddi. En este caso son tres: seguridad, publicación y búsqueda.

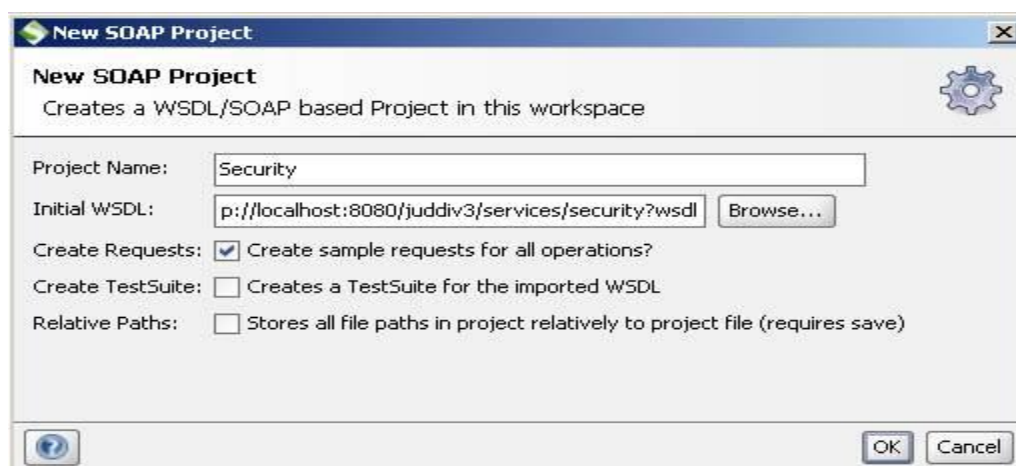


Las WSDL las podemos obtener accediendo a la opción *View service listing* de la página principal de servicio juddi que hemos lanzado o directamente a la URL <http://localhost:8080/juddiv3/services>. Por ejemplo, vamos a incluir en soapUI los servicios de seguridad de juddi y los otros dos se harán de forma similar.

En la página anterior buscamos el servicio *UDDI\_Security\_Porttype* para obtener la WSDL que describe las capacidades del servicio (los métodos).

UDDI_Security_PortType	Endpoint address: <a href="http://localhost:8080/juddiv3/services/security">http://localhost:8080/juddiv3/services/security</a>
<ul style="list-style-type: none"> <li>• discard_authToken</li> <li>• get_authToken</li> </ul>	WSDL : <a href="#">{urn:uddi-org:v3_service}UDDISecurityService</a>
	Target namespace: urn:uddi-org:v3_service

Pulsamos en el enlace que indica WSDL y copiamos la URL entera que aparece en la ventana que hemos abierto y que indica la dirección al documento WSDL correspondiente. En este caso es <http://localhost:8080/juddiv3/services/security?wsdl>. Pegamos esta URL en la ventana que veíamos anteriormente en la aplicación soapUI.



Pulsamos *ok* y aparece un nuevo proyecto *soapUI* denominado *Security* que incluye los servicios que describía la WSDL introducida.

4

Realizamos los mismos pasos para el servicio de búsqueda (*inquiry*) y para el de publicación (*publish*) con lo que ya tenemos los servicios necesarios para registrar el servicio Web de ejemplo.

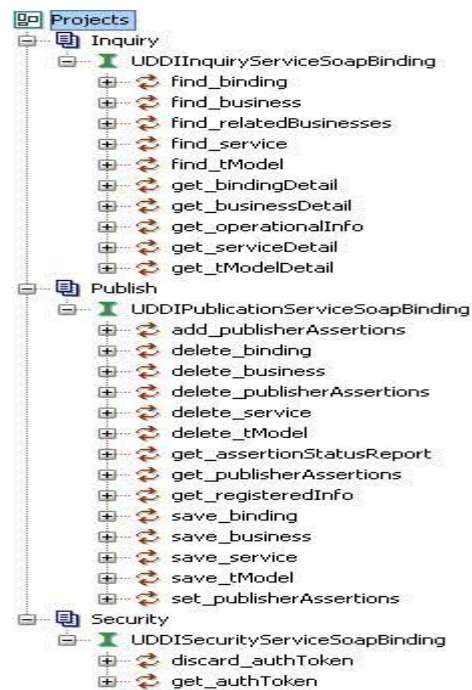
## Inquiry

UDDI_Inquiry_PortType	
<ul style="list-style-type: none"><li>• get_tModelDetail</li><li>• get_bindingDetail</li><li>• find_service</li><li>• get_serviceDetail</li><li>• find_binding</li><li>• get_businessDetail</li><li>• find_tModel</li><li>• find_relatedBusinesses</li><li>• find_business</li><li>• get_operationalInfo</li></ul>	Endpoint address: <a href="http://localhost:8080/juddiv3/services/inquiry">http://localhost:8080/juddiv3/services/inquiry</a> WSDL : <a href="#">{urn:uddi-org:v3_service}UDDIInquiryService</a> Target namespace: urn:uddi-org:v3_service

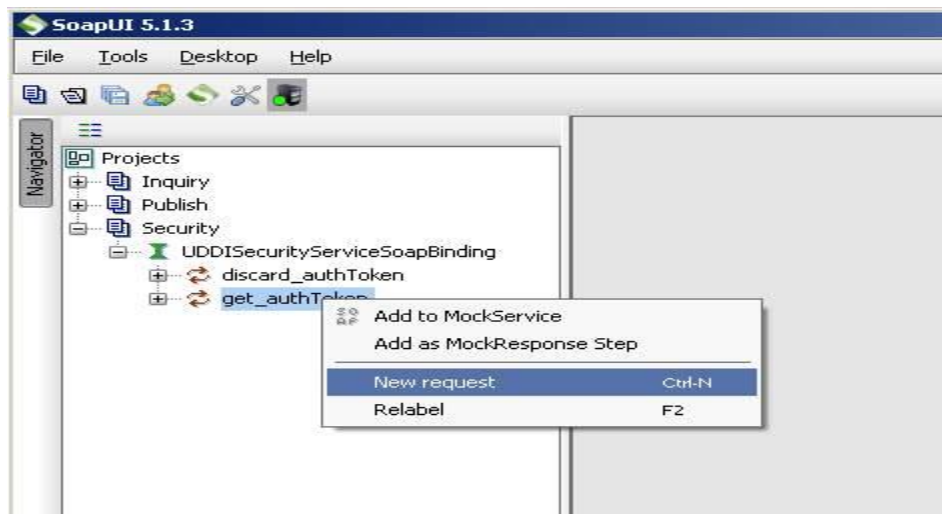
## Publish

UDDI_Publication_PortType	
<ul style="list-style-type: none"><li>• delete_business</li><li>• save_service</li><li>• delete_service</li><li>• set_publisherAssertions</li><li>• delete_binding</li><li>• get_publisherAssertions</li><li>• save_tModel</li><li>• delete_publisherAssertions</li><li>• add_publisherAssertions</li><li>• get_assertionStatusReport</li><li>• save_business</li><li>• delete_tModel</li><li>• save_binding</li><li>• get_registeredInfo</li></ul>	Endpoint address: <a href="http://localhost:8080/juddiv3/services/publish">http://localhost:8080/juddiv3/services/publish</a> WSDL : <a href="#">{urn:uddi-org:v3_service}UDDIPublicationService</a> Target namespace: urn:uddi-org:v3_service

Una vez creados los tres proyectos deberían de tener una estructura similar a las siguiente imagen.



Seleccionando cualquiera de los servicios y pulsando el botón derecho del ratón se puede generar un mensaje SOAP para enviar una petición al registro UDDI mediante la opción *New Request*.



De esta forma se genera el mensaje SOAP que se debe completar para ser enviado al registro UDDI.



Una vez descrito cómo crear peticiones SOAP con soapUI vamos a ir definiendo el orden de pasos para registrar un servicio.

El primer paso que se debe realizar para el registro consiste en la obtención de un *token* o identificador que permita las posteriores operaciones.

Para ello podemos utilizar la petición creada anteriormente, *get\_authToken*, para obtener el identificador. Como podéis observar hemos añadido para *userID = juddi* y para *cred = juddi*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:uddi-org:api_v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:get_authToken userID="juddi" cred="juddi"/>
  </soapenv:Body>
</soapenv:Envelope>>
```

En el mensaje de respuesta obtendremos el *token* que debemos usar en el resto de llamadas.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:authToken xmlns:ns11="urn:uddi-org:policy_v3_instanceParms" xmlns:ns10="urn:uddi-
org:vs_v3" xmlns:ns9="urn:uddi-org:vscache_v3" xmlns:ns8="urn:uddi-org:subr_v3"
xmlns:ns7="urn:uddi-org:sub_v3" xmlns:ns6="urn:uddi-org:custody_v3" xmlns:ns5="urn:uddi-
org:repl_v3" xmlns:ns4="http://www.w3.org/2000/09/xmldsig#" xmlns:ns3="urn:uddi-org:policy_v3"
xmlns:ns2="urn:uddi-org:api_v3">
      <ns2:authInfo>authtoken:c3bfof5e-od1f-4648-8d1c-e8f780301a56</ns2:authInfo>
    </ns2:authToken>
  </soap:Body>
</soap:Envelope>
```

Una vez tenemos el token comenzaremos registrando la parte abstracta del documento WSDL, que especifica la interfaz del servicio. Esta parte podría ser común a varios servicios ubicados en diferentes localizaciones geográficas. Esta parte se conoce como *tmodel*. Desde el proyecto soapUI que creamos con el nombre *Publish* debemos invocar la opción *save\_tmodel*.

Como se trata de un registro básico de ejemplo, se eliminarán las partes del mensaje optativas o más avanzadas.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:uddi-org:api_v3" xmlns:xd="http://www.w3.org/2000/09/xmldsig#">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:save_tModel>
      <urn:authInfo>authtoken: c3bfof5e-od1f-4648-8d1c-e8f780301a56</urn:authInfo>
      <urn:tModel tModelKey="">
        <urn:name>Conversor Temperatura</urn:name>
        <urn:description>Interfaz de conversor de temperatura</urn:description>
        <urn:overviewDoc>
          <urn:description>Parte abstracta del documento WSDL</urn:description>
          <urn:overviewURL>http://localhost:8080/MasterSW/Converter.wsdl</urn:overviewURL>
        </urn:overviewDoc>
      </urn:tModel>
    </urn:save_tModel>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    </urn:overviewDoc>
  </urn:tModel>
</urn:save_tModel>
</soapenv:Body>
</soapenv:Envelope>

```

La respuesta obtenida será

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:tModelDetail xmlns:ns11="urn:uddi-org:policy_v3_instanceParms" xmlns:ns10="urn:uddi-org:vs_v3" xmlns:ns9="urn:uddi-org:vscache_v3" xmlns:ns8="urn:uddi-org:subr_v3" xmlns:ns7="urn:uddi-org:sub_v3" xmlns:ns6="urn:uddi-org:custody_v3" xmlns:ns5="urn:uddi-org:repl_v3" xmlns:ns4="urn:uddi-org:policy_v3" xmlns:ns3="http://www.w3.org/2000/09/xmldsig#" xmlns:ns2="urn:uddi-org:api_v3">
      <ns2:tModel tModelKey="uddi:juddi.apache.org:5a6d57fc-cb95-40ao-b4de-foa2777d227e">
        <ns2:name>Conversor Temperatura</ns2:name>
        <ns2:description>Interfaz de conversor de temperatura</ns2:description>
        <ns2:overviewDoc>
          <ns2:description>Parte abstracta del documento WSDL</ns2:description>
          <ns2:overviewURL>http://localhost:8080/MasterSW/Converter.wsdl</ns2:overviewURL>
        </ns2:overviewDoc>
      </ns2:tModel>
    </ns2:tModelDetail>
  </soap:Body>
</soap:Envelope>

```

Una vez registrada la parte abstracta se registran los datos específicos de cada proveedor de servicios. Esta parte está relacionada con la información de la organización (independiente de la WSDL) y los datos técnicos de acceso al servicio (datos concretos de la WSDL como la ubicación y la forma de acceso).

Como paso previo registramos la información general de la organización invocando al método *save\_business*.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:uddi-org:api_v3" xmlns:xd="http://www.w3.org/2000/09/xmldsig#">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:save_business>
      <urn:authInfo>authtoken: c3bfof5e-od1f-4648-8d1c-e8f780301a56</urn:authInfo>
      <urn:businessEntity businessKey="">
        <urn:name>MiEmpresa</urn:name>
        <urn:description>Mi empresa</urn:description>
        <urn:contacts>
          <urn:contact>
            <urn:personName>Manolo Macia Macia</urn:personName>
            <urn:phone>916377868</urn:phone>
            <urn:email>manolo@empresa.es</urn:email>
          </urn:contact>
        </urn:contacts>
      </urn:businessEntity>
    </urn:save_business>
  </soapenv:Body>
</soapenv:Envelope>

```



La respuesta obtenida es

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:businessDetail xmlns:ns11="urn:uddi-org:policy_v3_instanceParms" xmlns:ns10="urn:uddi-
org:vs_v3" xmlns:ns9="urn:uddi-org:vscache_v3" xmlns:ns8="urn:uddi-org:subr_v3"
xmlns:ns7="urn:uddi-org:sub_v3" xmlns:ns6="urn:uddi-org:custody_v3" xmlns:ns5="urn:uddi-
org:repl_v3" xmlns:ns4="urn:uddi-org:policy_v3" xmlns:ns3="http://www.w3.org/2000/09/xmldsig#"
xmlns:ns2="urn:uddi-org:api_v3">
      <ns2:businessEntity businessKey="uddi:juddi.apache.org:c1982527-866f-4c8a-bb40-
4f63c8689919">
        <ns2:name>MiEmpresa</ns2:name>
        <ns2:description>Mi empresa</ns2:description>
        <ns2:contacts>
          <ns2:contact>
            <ns2:personName>Manolo Macia Macia</ns2:personName>
            <ns2:phone>916377868</ns2:phone>
            <ns2:email>manolo@empresa.es</ns2:email>
          </ns2:contact>
        </ns2:contacts>
      </ns2:businessEntity>
    </ns2:businessDetail>
  </soap:Body>
</soap:Envelope>
```

A continuación registramos la parte de implementación específica del servicio. Para ello invocamos al método *save\_service*. En este paso asociamos a nuestra organización a través del *business\_key* los servicios que se ofertan. Además asociamos al servicio la interfaz abstracta mediante el *tmodel\_key*. En el atributo *URLType* de la etiqueta *accessPoint* se indica la ubicación física del servicio. Para otro proveedor que utilizase la misma interfaz abstracta la ubicación física variaría. Para el ejemplo se ha usado un único documento WSDL que contiene tanto la parte abstracta como la de implementación. Si se hubiera separado estas dos partes en el *OverviewURL* del *service* y del *tmodel* se debería haber hecho referencia al documento correspondiente.

Por tanto ya tenemos los siguientes parámetros que nos ha generado el SoapUI:

```
-authToken: c3bf0f5e-0d1f-4648-8d1c-e8f780301a56
tModelKey =uddi:juddi.apache.org:5a6d57fc-cb95-40a0-b4de-f0a2777d227e
businessKey="uddi:juddi.apache.org:c1982527-866f-4c8a-bb40-4f63c8689919
```

Estos parámetros serán usados en la siguiente invocación. Recordad que el parámetro authToken tiene tiempo de expiración, y por tanto esto puede provocar un error, así que si os ocurre tendréis que generar de nuevo el authToken y por consiguiente todos los demás parámetros.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:uddi-org:api_v3" xmlns:xd="http://www.w3.org/2000/09/xmldsig#">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:save_service>
      <urn:authInfo>authToken:c3bf0f5e-0d1f-4648-8d1c-e8f780301a56</urn:authInfo>
```



```

        <urn:businessService businessKey="uddi:juddi.apache.org:c1982527-866f-4c8a-bb40-4f63c8689919" serviceKey="">
        <urn:name>ServicioConversionTemperatura</urn:name>
        <urn:description>Conversion de temperatura</urn:description>
        <urn:bindingTemplates>
        <urn:bindingTemplate bindingKey="">
        <urn:accessPoint
        useType="http">http://localhost:8080/ProyectoMaster/services/Converter</urn:accessPoint>
        <urn:tModelInstanceDetails>
        <urn:tModelInstanceInfo tModelKey="uddi:juddi.apache.org:5a6d57fc-cb95-40ao-b4de-foa2777d227e">
        <urn:instanceDetails>
        <urn:overviewDoc>
        <urn:overviewURL>http://localhost:8080/MasterSW/Converter.wsdl</urn:overviewURL>
        </urn:overviewDoc>
        </urn:instanceDetails>
        </urn:tModelInstanceInfo>
        </urn:tModelInstanceDetails>
        </urn:bindingTemplate>
        </urn:bindingTemplates>
        </urn:businessService>
    </urn:save_service>
</soapenv:Body>
</soapenv:Envelope>

```

La respuesta obtenida es

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:serviceDetail xmlns:ns11="urn:uddi-org:policy_v3_instanceParms" xmlns:ns10="urn:uddi-org:vs_v3" xmlns:ns9="urn:uddi-org:vscache_v3" xmlns:ns8="urn:uddi-org:subr_v3" xmlns:ns7="urn:uddi-org:sub_v3" xmlns:ns6="urn:uddi-org:custody_v3" xmlns:ns5="urn:uddi-org:repl_v3" xmlns:ns4="urn:uddi-org:policy_v3" xmlns:ns3="http://www.w3.org/2000/09/xmldsig#" xmlns:ns2="urn:uddi-org:api_v3">
      <ns2:businessService serviceKey="uddi:juddi.apache.org:3a9a1328-f3da-44d1-85d4-c7ac81fd80d8" businessKey="uddi:juddi.apache.org:c1982527-866f-4c8a-bb40-4f63c8689919">
        <ns2:name>ServicioConversionTemperatura</ns2:name>
        <ns2:description>Conversion de temperatura</ns2:description>
        <ns2:bindingTemplates>
          <ns2:bindingTemplate bindingKey="uddi:juddi.apache.org:8f274bff-47e8-4239-b8ca-b85081df24bc">
            <ns2:accessPoint
              useType="http">http://localhost:8080/ProyectoMaster/services/Converter</ns2:accessPoint>
            <ns2:tModelInstanceDetails>
              <ns2:tModelInstanceInfo tModelKey="uddi:juddi.apache.org:5a6d57fc-cb95-40ao-b4de-foa2777d227e">
                <ns2:instanceDetails>
                  <ns2:overviewDoc>
                    <ns2:overviewURL>http://localhost:8080/MasterSW/Converter.wsdl</ns2:overviewURL>
                  </ns2:overviewDoc>
                </ns2:instanceDetails>
              </ns2:tModelInstanceInfo>
            </ns2:tModelInstanceDetails>
          </ns2:bindingTemplate>
        </ns2:bindingTemplates>
      </ns2:businessService>
    </ns2:serviceDetail>
  </soap:Body>
</soap:Envelope>

```

```

    </ns2:bindingTemplates>
  </ns2:businessService>
</ns2:serviceDetail>
</soap:Body>
</soap:Envelope>

```

A partir, del proyecto Inquiry de soapUI, podemos usar sus operaciones para realizar búsquedas. Los métodos *get* permiten localizar información de los elementos de registro jUDDI mediante el uso de las claves que se obtuvieron en el registro de la entidad de negocio, el servicio, el tmodel, etc. A continuación se muestra un ejemplo de búsqueda del tmodel registrado anteriormente.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:uddi-org:api_v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:get_tModelDetail>
      <!--Optional:-->
      <urn:authInfo> authToken:c3bfof5e-od1f-4648-8d1c-e8f780301a56</urn:authInfo>
      <!--1 or more repetitions:-->
      <urn:tModelKey>uddi:juddi.apache.org:5a6d57fc-cb95-40ao-b4defoa2777d227e</urn:tModelKey>
    </urn:get_tModelDetail>
  </soapenv:Body>
</soapenv:Envelope>

```

Como respuesta obtendríamos el siguiente mensaje.

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:tModelDetail xmlns:ns11="urn:uddi-org:policy_v3_instanceParms" xmlns:ns10="urn:uddi-
org:vs_v3" xmlns:ns9="urn:uddi-org:vscache_v3" xmlns:ns8="urn:uddi-org:subr_v3"
xmlns:ns7="urn:uddi-org:sub_v3" xmlns:ns6="urn:uddi-org:custody_v3" xmlns:ns5="urn:uddi-
org:repl_v3" xmlns:ns4="urn:uddi-org:policy_v3" xmlns:ns3="http://www.w3.org/2000/09/xmldsig#"
xmlns:ns2="urn:uddi-org:api_v3">
      <ns2:tModel tModelKey="uddi:juddi.apache.org:5a6d57fc-cb95-40ao-b4de-foa2777d227e"
deleted="false">
        <ns2:name>Conversor Temperatura</ns2:name>
        <ns2:description>Interfaz de conversor de temperatura</ns2:description>
        <ns2:overviewDoc>
          <ns2:description>Parte abstracta del documento WSDL</ns2:description>
          <ns2:overviewURL
useType="">http://localhost:8080/MasterSW/Converter.wsdl</ns2:overviewURL>
        </ns2:overviewDoc>
      </ns2:tModel>
    </ns2:tModelDetail>
  </soap:Body>
</soap:Envelope>

```

También podemos utilizar los métodos *find* para realizar búsqueda. Estos métodos ofrecen un mecanismo más potente de búsqueda cuando no conocemos las claves precisas del elemento a localizar o simplemente queremos realizar una búsqueda por términos más generales (por ejemplo, buscar entidades de negocio dedicadas a organización de viajes).

En los métodos *find* el carácter comodín es ‘%’ pero es necesario declarar el parámetro opcional *findQualifier* a *approximateMatch*.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:uddi-org:api_v3">
  <soapenv:Header/>
  <soapenv:Body>
    <urn:find_tModel>
      <!--Optional:-->
      <urn:authInfo>authtoken:c3bfof5e-od1f-4648-8d1c-e8f780301a56</urn:authInfo>
      <!--Optional:-->
      <urn:findQualifiers>
        <!--1 or more repetitions:-->
        <urn:findQualifier>approximateMatch</urn:findQualifier>
      </urn:findQualifiers>
      <!--Optional:-->
      <urn:name xml:lang="">Conv%</urn:name>
    </urn:find_tModel>
  </soapenv:Body>
</soapenv:Envelope>

```

Como respuesta se obtendrían todos los tmodels registrados en el catálogo que coincidan con la cadena indicada en la etiqueta *name*.

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:tModelList truncated="false" xmlns:ns11="urn:uddi-org:policy_v3_instanceParms"
xmlns:ns10="urn:uddi-org:vs_v3" xmlns:ns9="urn:uddi-org:vscache_v3" xmlns:ns8="urn:uddi-
org:subr_v3" xmlns:ns7="urn:uddi-org:sub_v3" xmlns:ns6="urn:uddi-org:custody_v3"
xmlns:ns5="urn:uddi-org:repl_v3" xmlns:ns4="urn:uddi-org:policy_v3"
xmlns:ns3="http://www.w3.org/2000/09/xmldsig#" xmlns:ns2="urn:uddi-org:api_v3">
      <ns2:listDescription>
        <ns2:includeCount>3</ns2:includeCount>
        <ns2:actualCount>3</ns2:actualCount>
        <ns2:listHead>1</ns2:listHead>
      </ns2:listDescription>
      <ns2:tModelInfos>
        <ns2:tModelInfo tModelKey="uddi:juddi.apache.org:5a6d57fc-cb95-40ao-b4de-foa277d227e">
          <ns2:name>Conversor Temperatura</ns2:name>
          <ns2:description>Interfaz de conversor de temperatura</ns2:description>
        </ns2:tModelInfo>
        <ns2:tModelInfo tModelKey="uddi:juddi.apache.org:c4e8dao8-1751-4d7e-aac2-81aac4d2e2f4">
          <ns2:name>Conversor Temperatura</ns2:name>
          <ns2:description>Interfaz de conversor de temperatura</ns2:description>
        </ns2:tModelInfo>
        <ns2:tModelInfo tModelKey="uddi:juddi.apache.org:233f5ffc-05f5-46b1-8a2b-4d6d80e23dc4">
          <ns2:name>Conversor Temperatura</ns2:name>
          <ns2:description>Interfaz de conversor de temperatura</ns2:description>
        </ns2:tModelInfo>
      </ns2:tModelInfos>
    </ns2:tModelList>
  </soap:Body>
</soap:Envelope>

```