

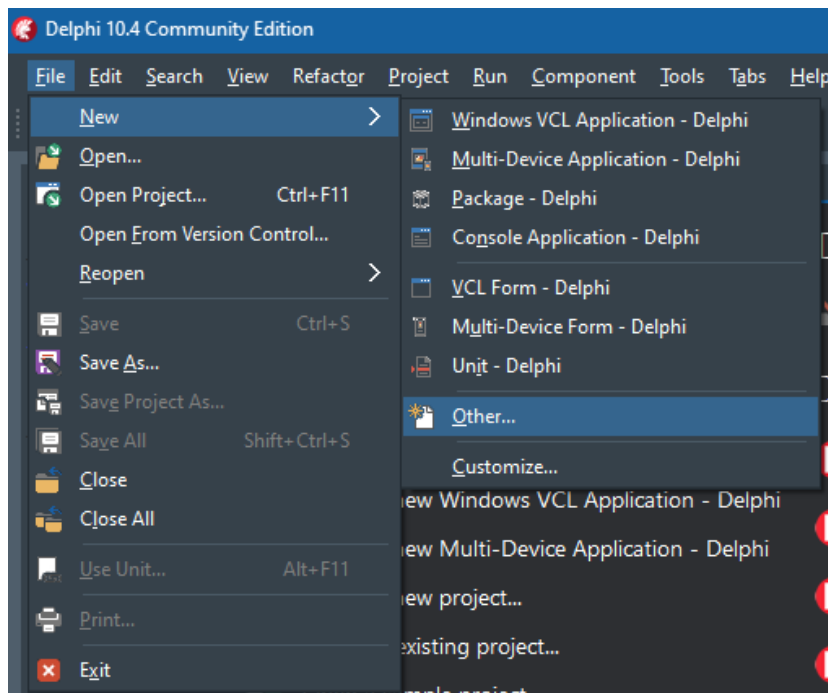
# **CRIANDO WEB SERVICES SOAP (DELPHI)**



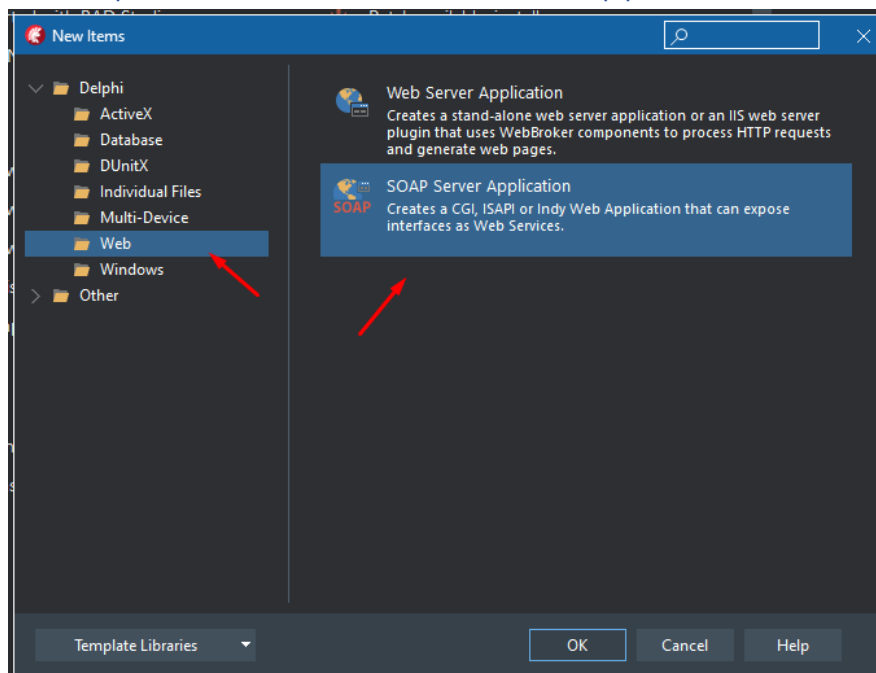
## Sumário

1. File -> New -> Other .....	3
2. Delphi -> Web -> SOAP Server Application .....	3
2.1. Escolher o ambiente Windows.....	4
2.2. Escolher o tipo de aplicação (Stand-alone GUI Application).....	4
2.3. Escolher a linguagem que será utilizada (VCL Application) .....	5
2.4. Definir a porta de comunicação (9090), Testar e Finish .....	5
2.5. Criar uma interface para o módulo SOAP (Yes) .....	5
3. File -> Save All.....	6
3.1. Compile All .....	6
3.2. Rodando e conhecendo nosso primeiro Webservice .....	6
4. Importar o WSDL no SoapUI .....	7
4.1. Testar o método echoDouble .....	7
5. Criando um novo método (Hello World).....	8
5.1. Na Unit Hello_WorldIntF criar um método novo na interface .....	8
5.2. Na Unit Hello_WorldImpl implementar o método criado na interface .....	8
5.3. Compilar, rodar e levantar o servidor .....	9
6. Recarregar as definições do WSDL no SoapUI .....	9
6.1. Rodar o novo método .....	9

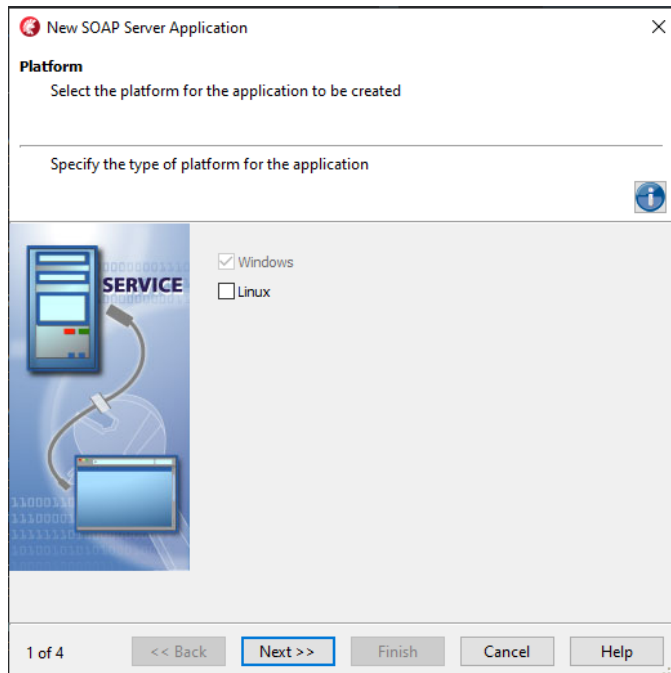
## 1. File -> New -> Other



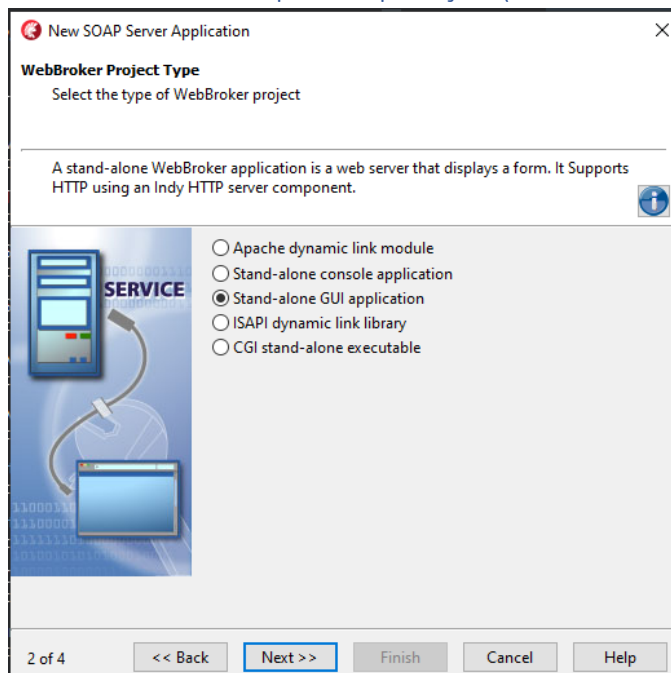
## 2. Delphi -> Web -> SOAP Server Application



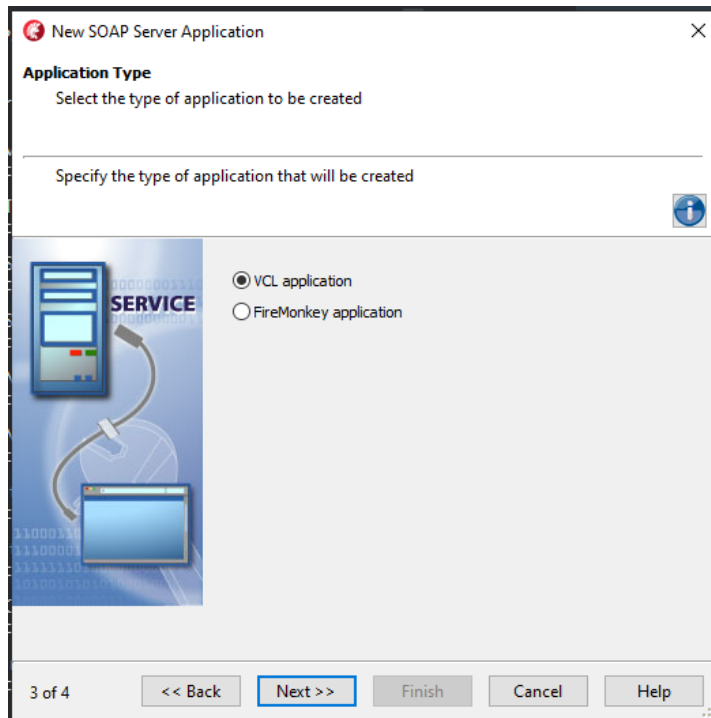
## 2.1. Escolher o ambiente Windows



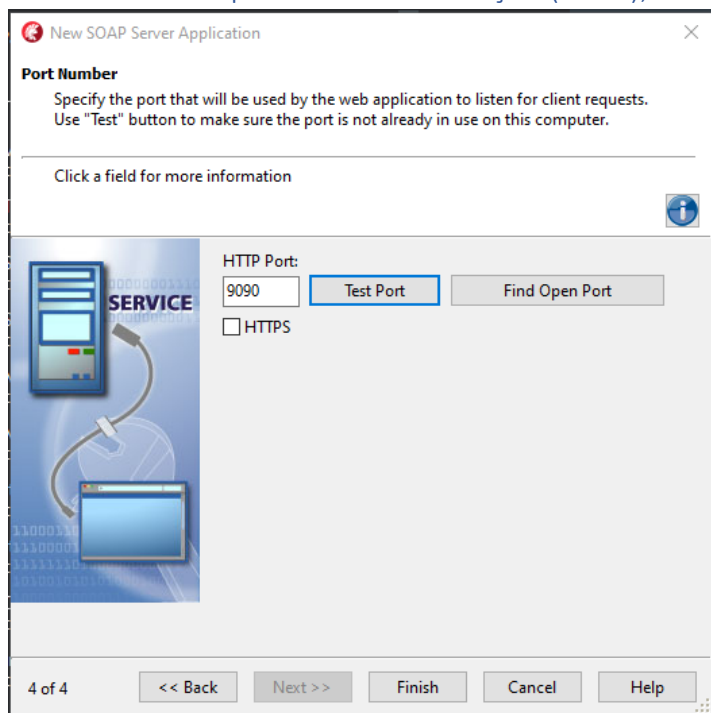
## 2.2. Escolher o tipo de aplicação (Stand-alone GUI Application)



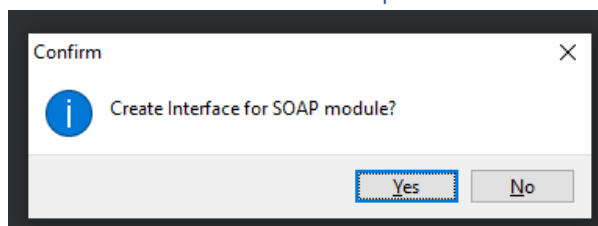
### 2.3. Escolher a linguagem que será utilizada (VCL Application)

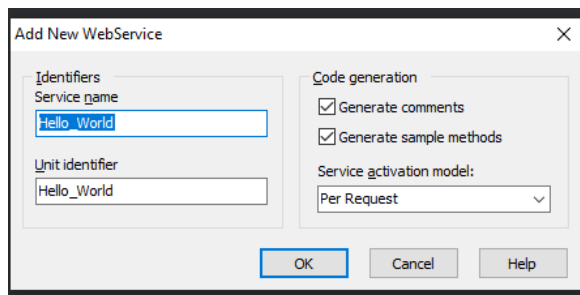


### 2.4. Definir a porta de comunicação (9090), Testar e Finish



### 2.5. Criar uma interface para o módulo SOAP (Yes)





### 3. File -> Save All

#### 3.1. Compile All

#### 3.2. Rodando e conhecendo nosso primeiro Webservice

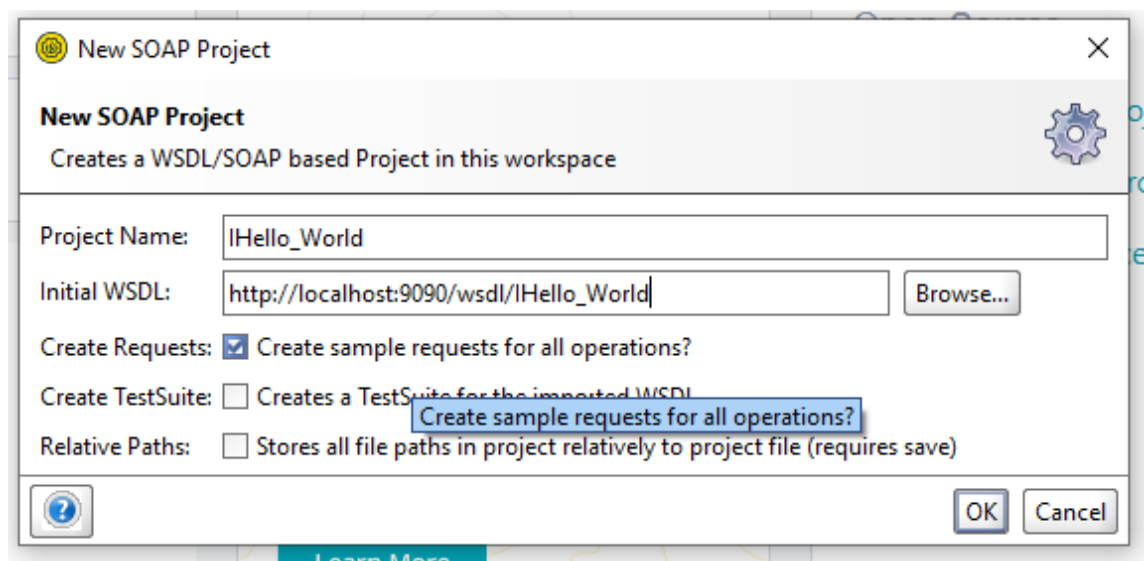
localhost:9090/wsd/IHello\_World

This XML file does not appear to have any style information associated with it. The document tree is shown below:

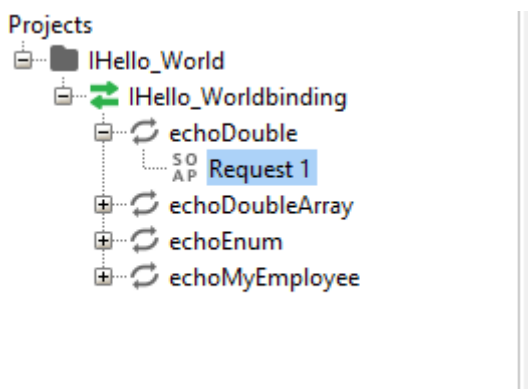
```
<?xml version='1.0' encoding='utf-8'?>
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:tns="http://tempuri.org/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:wsi="http://schemas.xmlsoap.org/wsdl/wsi/" xmlns:wsil="urn:Hello_WorldIntf" name="IHello_WorldService" targetNamespace="http://tempuri.org/">
  <types>
    <xsd:schema xmlns="urn:Hello_WorldIntf" targetNamespace="urn:Hello_WorldIntf">
      <xsd:complexType name="TEnumTest">
        <xsd:restriction base="xs:string">
          <xsd:enumeration value="etNone"/>
          <xsd:enumeration value="etAFew"/>
          <xsd:enumeration value="etSome"/>
          <xsd:enumeration value="etAllot"/>
        </xsd:restriction>
      </xsd:complexType>
      <xsd:complexType name="TDoubleArray">
        <xsd:complexContent>
          <xsd:restriction base="soap:Array">
            <xsd:sequence xmlns="http://www.w3.org/2001/XMLSchema" ref="soap:ArrayType" n1:arrayType="xs:double[]"/>
          </xsd:restriction>
        </xsd:complexContent>
      </xsd:complexType>
      <xsd:complexType name="TMyEmployee">
        <xsd:sequence xmlns="http://www.w3.org/2001/XMLSchema">
          <xsd:element name="LastName" type="xs:string"/>
          <xsd:element name="FirstName" type="xs:string"/>
          <xsd:element name="Salary" type="xs:double"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:schema>
  </types>
  <message name="echoEnumRequest">
    <part name="Value" type="tns:TEnumTest"/>
  </message>
  <message name="echoEnumResponse">
    <part name="return" type="tns:TEnumTest"/>
  </message>
  <message name="echoDoubleArrayRequest">
    <part name="Value" type="tns:TDoubleArray"/>
  </message>
  <message name="echoDoubleArrayResponse">
    <part name="return" type="tns:TDoubleArray"/>
  </message>
  <message name="echoMyEmployeeRequest">
    <part name="Value" type="tns:TMyEmployee"/>
  </message>
  <message name="echoMyEmployeeResponse">
    <part name="return" type="tns:TMyEmployee"/>
  </message>
</definitions>
```

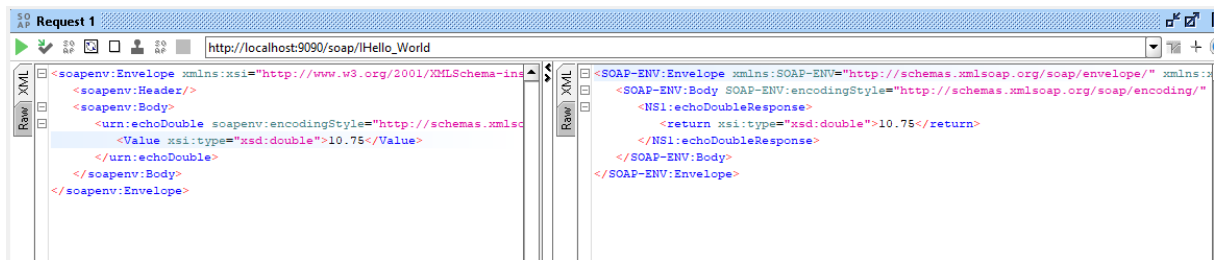
#### 4. Importar o WSDL no SoapUI

([http://localhost:9090/wsd/IHello\\_World](http://localhost:9090/wsd/IHello_World))



##### 4.1. Testar o método echoDouble





## 5. Criando um novo método (Hello World)

### 5.1. Na Unit Hello\_WorldIntF criar um método novo na interface

```
{ Invokable interfaces must derive from IInvokable }
IHello_World = interface(IInvokable)
[ '{5BE615A8-66F6-4086-BE08-5E30E677F576}' ]

{ Methods of Invokable interface must not use the de
{ calling convention; stdcall is recommended }
function echoEnum(const Value: TEnumTest): TEnumTest;
function echoDoubleArray(const Value: TDoubleArray): TDoubleArray;
function echoMyEmployee(const Value: TMyEmployee): TMyEmployee;
function echoDouble(const Value: Double): Double; stdcall;
function echoHelloWorld: String; stdcall;
end;
```

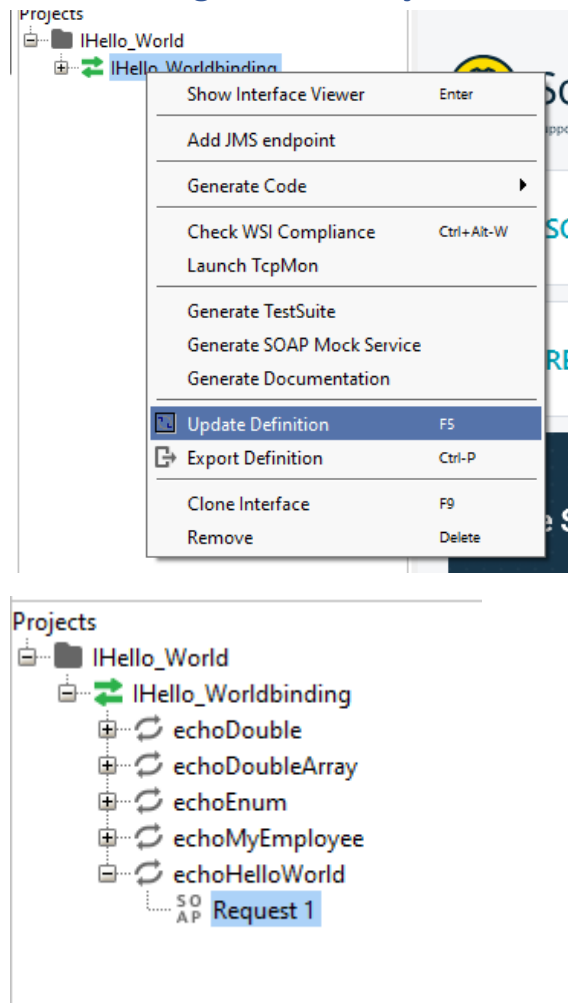
### 5.2. Na Unit Hello\_WorldImpl implementar o método criado na interface

```
{ THello_World }
THello_World = class(TInvokableClass, IHello_World)
public
function echoEnum(const Value: TEnumTest): TEnumTest;
function echoDoubleArray(const Value: TDoubleArray): TDoubleArray;
function echoMyEmployee(const Value: TMyEmployee): TMyEmployee;
function echoDouble(const Value: Double): Double;
function echoHelloWorld: String; stdcall;
end;
```



### 5.3. Compilar, rodar e levantar o servidor

## 6. Recarregar as definições do WSDL no SoapUI



### 6.1. Rodar o novo método

