



Die Solution besteht aus 4 Projekten:

- ClientGUI
- ServiceData
- ServiceREST
- ServiceSOA

ServiceData ist mein DataContract und wird in jedem Projekt verwendet.

Den Client hab ich so gelöst, dass ich ein ServiceFactory Pattern habe. Somit kann ich auf die gewünschte Art auf mein Service zugreifen, ohne dabei direkt auf die GUI Elemente zuzugreifen.

## GET CUSTOMERS

### SOA

```
<xs:element name="GetCustomers">
  <xs:complexType>
    <xs:sequence/>
  </xs:complexType>
</xs:element>
<xs:element name="GetCustomersResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="GetCustomersResult"
        nillable="true" type="q1:ArrayOfCustomer"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

### REST

```
Status Code: 200 OK
Cache-Control: private
Content-Length: 64
Content-Type: application/json; charset=utf-8
Date: Fri, 10 Jan 2014 19:04:51 GMT
Server: Microsoft-IIS/8.5
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
```

```
[{"Name": "Customer1"}, {"Name": "Customer2"}, {"Name": "Customer3"}]
```