

Cloud Computing

SOAP REST Service

Florian Grill

Inhalt

SOAP Service.....	2
REST Service	2
Server Design Entscheidung	2
SOAP Server.....	2
REST Server.....	9
Client Design Entscheidung	11
SOAP client	11
REST client	12
Client Funktionsbeschreibung	12

SOAP Service

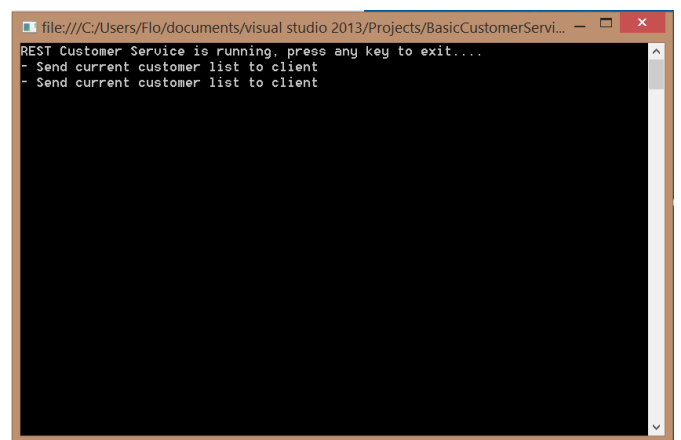
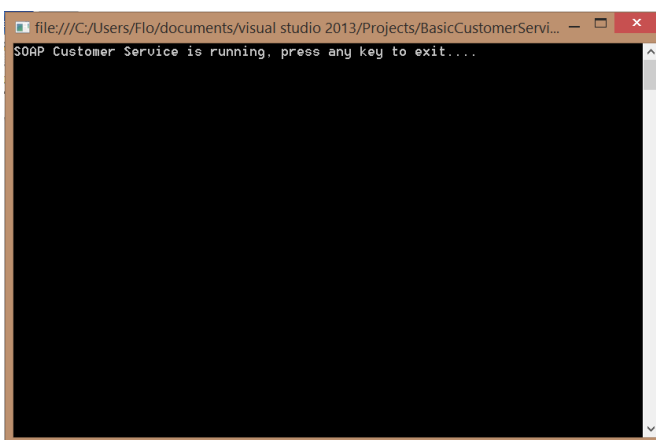
Soap ist ein Netzwerkprotokoll mit dessen Hilfe Remote Procedure Calls ausgeführt werden können und so einfach Daten zwischen Endgeräten ausgetauscht werden können. Soap verwendet zum Repräsentieren der ausgetauschten Daten XML.

REST Service

REST bezeichnet ein Programmierparadigma für Webanwendungen. REST fordert, dass eine Web-Adresse (URI) genau einen Seiteninhalt repräsentiert, und dass ein Web-/REST-Server auf mehrfache Anfragen mit demselben URI auch mit demselben Webseiteninhalt antwortet. REST stellt eine einfache Alternative zu ähnlichen Verfahren wie SOAP und WSDL und dem verwandten Verfahren RPC dar

Server Design Entscheidung

Der Prototyp Server wurde aufgrund von eigener Erfahrung in C# umgesetzt und verwendet dazu die von .Net zur Verfügung gestellten WCF Bibliotheken.



SOAP Server

Aufgerufen werden kann das Service unter der URL <http://localhost:8180/customerservice> und das wsdl File findet man unter <http://localhost:8180/customerservice?wsdl>. Das Service wurde in einer simplen Consolen Anwendung gehostet und kann einfach mit dem „CustomerServiceHost.exe“ gestartet werden. Wichtig ist das man für das Ausführen der Anwendung Administrator Rechte auf den PC besitzt muss.

Das Service nimmt folgende Anfragen entgegen:

URL	Methoden name	Erwartete Parameter im C# Service
1: http://grill.com/webservices/CustomerService/AddCustomer	AddCustomer	string customerName
2: http://grill.com/webservices/CustomerService/GetCustomers	GetCustomers	string customerName
3: http://grill.com/webservices/CustomerService/DeleteCustomer	DeleteCustomer	string customerName
4: http://grill.com/webservices/CustomerService/DeleteOrder	DeleteOrder	string customerName, int orderIndex
5: http://grill.com/webservices/CustomerService/GetOrders	GetOrders	string customerName
6: http://grill.com/webservices/CustomerService/AddOrder"	AddOrder	string customerName, List<string> order

Soap Messages:

1: Request Client:

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

<v:Envelope xmlns:v="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:c="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:d="http://www.w3.org/2001/XMLSchema"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">

    <v:Header />

    <v:Body>

        <AddCustomer xmlns="http://grill.com/webservices/" id="o0" c:root="1">

            <customerName>Test</customerName>

        </AddCustomer>

    </v:Body>

</v:Envelope>
```

1: Response Server:

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

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">

    <s:Body>

        <AddCustomerResponse xmlns="http://grill.com/webservices/">

            <AddCustomerResult>true</AddCustomerResult>

        </AddCustomerResponse>

    </s:Body>

</s:Envelope>
```

```
    </s:Body>

</s:Envelope>
```

2: Request Client:

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

<v:Envelope xmlns:v="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:c="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:d="http://www.w3.org/2001/XMLSchema"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">

    <v:Header />

    <v:Body>

        <GetCustomers xmlns="http://grill.com/webservices/" id="o0" c:root="1" />

    </v:Body>

</v:Envelope>
```

2: Response Server:

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

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">

    <s:Body>

        <GetCustomersResponse xmlns="http://grill.com/webservices/">

            <GetCustomersResult
xmlns:a="http://schemas.datacontract.org/2004/07/BasicCustomerServiceInterfaces.Models" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">

                <a:Customer>

                    <a:CustomerName>Daniel Kienböck</a:CustomerName>

                    <a:CustomerOrders
xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">

                        <b:ArrayOfstring>

                            <b:string>DVD</b:string>

                            <b:string>Handy</b:string>

                            <b:string>Laptop</b:string>

                        </b:ArrayOfstring>

                    </a:CustomerOrders>

                </a:Customer>

            </GetCustomersResult>

        </GetCustomersResponse>

    </s:Body>

</s:Envelope>
```

```

        <b:ArrayOfstring>

            <b:string>Buch</b:string>

            <b:string>Stofftier</b:string>

            <b:string>Jacke</b:string>

        </b:ArrayOfstring>

    </a:CustomerOrders>

</a:Customer>

<a:Customer>

    <a:CustomerName>Mario Murrent</a:CustomerName>

    <a:CustomerOrders
xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays">

        <b:ArrayOfstring>

            <b:string>Gameboy</b:string>

            <b:string>Handy</b:string>

            <b:string>PC-Spiel</b:string>

        </b:ArrayOfstring>

    </a:CustomerOrders>

</a:Customer>

<a:Customer>

    <a:CustomerName>Bernhard Stöckl</a:CustomerName>

    <a:CustomerOrders
xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays" />

        <b:ArrayOfstring>

            <b:string>DVD</b:string>

```

```

        <b:string>Spiele-Konsole</b:string>

    </b:ArrayOfString>

</a:CustomerOrders>

</a:Customer>

</GetCustomersResult>

</GetCustomersResponse>

</s:Body>

</s:Envelope>

```

3: Request Client:

```

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

<v:Envelope xmlns:v="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:c="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:d="http://www.w3.org/2001/XMLSchema"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">

    <v:Header />

    <v:Body>

        <DeleteCustomer xmlns="http://grill.com/webservices/" id="o0" c:root="1">

            <customerName>Test</customerName>

        </DeleteCustomer>

    </v:Body>

</v:Envelope>

```

3: Response Server:

```

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

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">

    <s:Body>

        <DeleteCustomerResponse xmlns="http://grill.com/webservices/">

            <DeleteCustomerResult>true</DeleteCustomerResult>

        </DeleteCustomerResponse>

    </s:Body>

```

```
</s:Envelope>
```

4: Request Client:

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

<v:Envelope xmlns:v="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:c="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:d="http://www.w3.org/2001/XMLSchema"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">

  <v:Header />

  <v:Body>

    <DeleteOrder xmlns="http://grill.com/webservices/" id="o0" c:root="1">

      <customerName>Daniel Kienböck</customerName>

      <orderIndex>1</orderIndex>

    </DeleteOrder>

  </v:Body>

</v:Envelope>
```

4: Response Server:

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

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">

  <s:Body>

    <DeleteOrderResponse xmlns="http://grill.com/webservices/">

      <DeleteOrderResult>true</DeleteOrderResult>

    </DeleteOrderResponse>

  </s:Body>

</s:Envelope>
```

5: Request Client:

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

<v:Envelope xmlns:v="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:c="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:d="http://www.w3.org/2001/XMLSchema"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">

  <v:Header />
```



```

<v:Body>

  <GetOrders xmlns="http://grill.com/webservices/" id="o0" c:root="1">

    <customerName>Daniel Kienböck</customerName>

  </GetOrders>

</v:Body>

</v:Envelope>

```

5: Response Server:

```

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

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">

  <s:Body>

    <GetOrdersResponse xmlns="http://grill.com/webservices/">

      <GetOrdersResult
xmlns:a="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">

        <a:ArrayOfstring>

          <a:string>DVD</a:string>

          <a:string>Handy</a:string>

          <a:string>Laptop</a:string>

        </a:ArrayOfstring>

      </GetOrdersResult>

    </GetOrdersResponse>

  </s:Body>

</s:Envelope>

```

6: Request Client:

```

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

<v:Envelope xmlns:v="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:c="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:d="http://www.w3.org/2001/XMLSchema"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">

```

```

<v:Header />

<v:Body>

    <AddOrder xmlns="http://grill.com/webservices/" id="o0" c:root="1">

        <customerName>Daniel Kienböck</customerName>

        <n0:order xmlns:n0="http://grill.com/webservices/"

            <n1:string
xmlns:n1="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
i:type="d:string">TestProduct</n1:string>

        </n0:order>

    </AddOrder>

</v:Body>

</v:Envelope>

```

6: Response Server:

```

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

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">

    <s:Body>

        <AddOrderResponse xmlns="http://grill.com/webservices/">

            <AddOrderResult>true</AddOrderResult>

        </AddOrderResponse>

    </s:Body>

</s:Envelope>

```

REST Server

Aufgerufen werden kann das Service unter der URL <http://localhost:8280>. Das Service wurde in einer simplen Consolen Anwendung gehostet und kann einfach mit dem „CustomerRestServiceHost.exe“ gestartet werden. Wichtig ist das man für das Ausführen der Anwendung Administrator Rechte auf den PC besitzt muss.

Das Service nimmt folgende Anfragen entgegen:

URL	Methoden name
1. http://localhost:8280/Customer/{customerName}	AddCustomer
2. http://localhost:8280/Customers	GetCustomers
3. http://localhost:8280/Customer/{customerName}	DeleteCustomer
4. http://localhost:8280/Order/{customerName}/orderIndex/{orderIndex}	DeleteOrder
6. http://localhost:8280/Orders/{customerName}	GetOrders

7. http://localhost:8280/Order/ {customerName}/order/{customerName}	AddOrder
--	----------

REST Messages:

1. Response Server:

true

2. Response Server:

```
[
  {
    "CustomerName": "Daniel Kienböck",
    "CustomerOrders": [
      [
        "DVD",
        "Handy",
        "Laptop"
      ],
      [
        "Buch",
        "Stofftier",
        "Jacke"
      ]
    ]
  },
  {
    "CustomerName": "Mario Murrent",
    "CustomerOrders": [
      [
        "Gameboy",
        "Handy",
        "PC-Spiel"
      ]
    ]
  },
  {
    "CustomerName": "Bernhard Stöckl",
    "CustomerOrders": [
    ]
  },
  {
    "CustomerName": "Roland Lehner",
    "CustomerOrders": [
      [
        "DVD",
        "Spiele-Konsole"
      ]
    ]
  }
]
```

3. Response Server:

true

4. Response Server:

true

5. Response Server:

```
[
  [
    "DVD",
    "Handy",
    "Laptop"
  ],
  [
    "Test|"
  ]
]
```

6. Response Server:

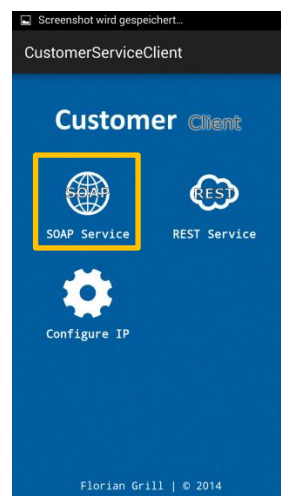
true

Client Design Entscheidung

Aufgrund eigenen Interesses wurde der Client für Android umgesetzt. Zur Umsetzung für den REST und SOAP client wurden verschiedene 3rd Party Bibliotheken verwendet. Weiter kann in diesem Client gewählt werden ob das SOAP oder das REST Service genutzt werden soll. Der client wurde als apk Datei zur Verfügung gestellt, diese kann einfach auf das Smartphone geladen werden und per Klick installiert werden. Wichtig das Gerät muss eine Android Version von 2.3.3 oder höher aufweisen.

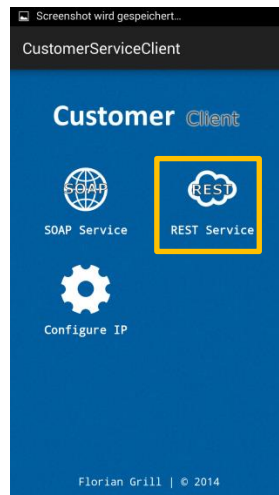
SOAP client

Leider gibt es in der Standard Android API von Google keine Bibliothek für SOAP. Aus diesem Grund wurde die 3rd Party Bibliothek von ksoap2 verwendet. Diese erweitert die Android API, und das Gerät kann als SOAP client verwendet werden.



REST client

Zur Erleichterung wurde für die REST Funktionalitäten die REST API die 3rd Party Bibliotheken von spring verwendet. Es wäre auch möglich über Standard Funktionen der Android API http requests an das REST Service zu senden.



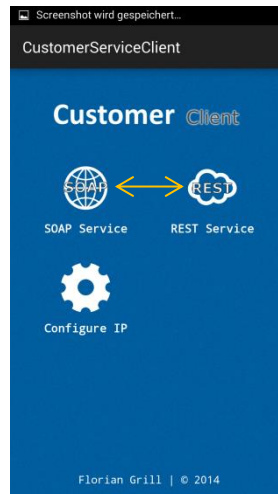
Client Funktionsbeschreibung

Die Folgenden Bilder sollen beschreiben wie der client genutzt werden kann. Von der Bedienung gibt es keinen Unterschied zwischen REST und SOAP.

1. Beim Starten sollte die IP des Servers eingetragen werden:



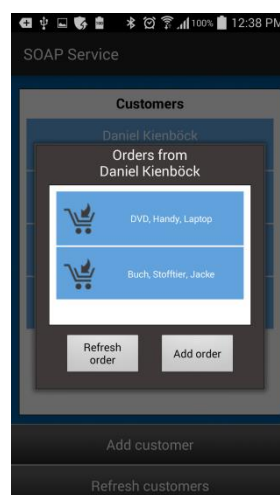
2. Danach kann je nach Belieben gewählt werden ob das SOAP oder das REST Service genutzt werden soll:



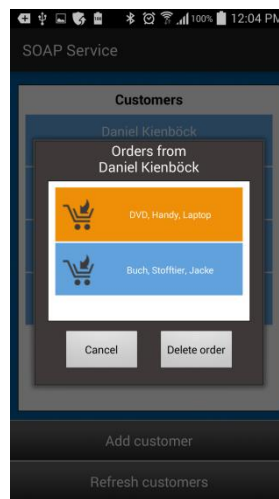
3. Beim Starten einer der beiden Dienste erscheint ein neues Fenster indem alle Customer, nach erfolgreichen Verbinden zum Service, dargestellt.



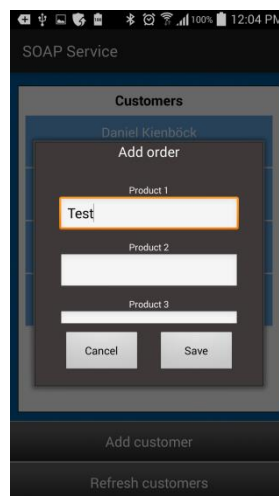
4. Beim Klicken auf „Add customer“ kann ein neuer Kunde angelegt werden. Beim Klick auf „Refresh customers“ können die Kunden vom Service neu beantragt werden.
5. Um die Bestellungen eines Kunden zu bearbeiten bzw. darzustellen, muss der entsprechende Kunde aus der Liste ausgewählt werden. Die geschieht mit einem Simplen Klick auf den Listen Eintrag.



6. Um die Liste an Bestellungen neu anzufordern genügt es auf den Button „Refresh order“ zu klicken. Soll eine Bestellung gelöscht werden, muss der gewünschte Listen Eintrag gewählt werden. Um die Bestellung für das Löschen zu aktivieren muss auf den Eintrag geklickt werden, und dieser gedrückt gehalten werden bis ein Vibrationsfeedback zu spüren ist.



7. Danach kann der Befehl abgebrochen werden oder die Bestellung mit „Delete order“ gelöscht werden.
8. Um eine Bestellung hinzuzufügen muss auf „Add order“ (Siehe Bild Nr. 5) gedrückt werden.



9. Hier können maximal 4 Produkte eingegeben werden und durch „Save“ dem Kunden hinzugefügt werden.