

1 Datenstrukturen

Damit dieses Projekt realisiert werden kann, wird die notwendige Strukturierung der Daten ausführlich besprochen.

Die unten dargestellte Tabelle zeigt um welche Daten es sich handelt:

Funktion	Daten
Benutzer	<ul style="list-style-type: none">• User Gruppe• Vorname, Nachname• E-Mail• Passwort• SecCode• MailChecked
User Gruppe	<ul style="list-style-type: none">• ID• Gruppen Name
Parkplätze	<ul style="list-style-type: none">• ID• SecCode• Anzahl Parkplätze• Anzahl freie Parkplätze• Geodaten (Längen-/Breitengrad)• Name Parkplatz• Entfernung zu TH• Adresse• Öffnungszeiten• Kosten• Anzeige von Frauenparkplatz• Anzeige von Behindertenparkplatz

Table 1: Daten innerhalb der Anwendung

1.1 XML-Dateien

Für dieses Projekt wurde eine Android-Applikation entwickelt. Zur Übermittlung von Daten eignet sich das Repräsentationsformat XML. Dieses wird in diesem Projekt in 3 Fällen genutzt.

Beispielhaft wird das XML Dokument für einen neuen Nutzer betrachtet.

Zunächst wird die XML-Dateien präsentiert und in der Folge durch das zugehörigen XML-Schemata dargestellt und erläutert.

```
<newuser>
  <email>selin.oetztuerk@mail.th-koeln.de</email>
  <password>vollSicher</password>
  <usergroup>3</usergroup>
</newuser>
```

Table 2: XML-Datei "Registrierung"

```
<list>
<list>
  <parkingPlace>
    <id>1</id>
    <parkingPlaces>111</parkingPlaces>
    <freeParkingPlaces>112</freeParkingPlaces>
    <longitude>6.78363318</longitude>
    <latitude>4.13645645</latitude>
    <name>Testparkplatz 1</name>
    <distanceToFH>2.7</distanceToFH>
    <addressStreet>Teststraße</addressStreet>
    <addressNumber>74b</addressNumber>
    <addressPostalCode>55555</addressPostalCode>
    <addressCity>Kummersbach</addressCity>
    <openingtimes>
      <mondayOpening>07:00:00</mondayOpening>
      <mondayClosing>20:00:00</mondayClosing>
    <tuesdayOpening>07:00:00</tuesdayOpening>
    <tuesdayClosing>20:00:00</tuesdayClosing>
    <wednesdayOpening>07:00:00</wednesdayOpening>
    <wednesdayClosing>20:00:00</wednesdayClosing>
    <thursdayOpening>07:00:00</thursdayOpening>
    <thursdayClosing>20:00:00</thursdayClosing>
    <fridayOpening>07:00:00</fridayOpening>
    <fridayClosing>20:00:00</fridayClosing>
    <saturdayOpening>08:00:00</saturdayOpening>
    <saturdayClosing>22:00:00</saturdayClosing>
    <sundayOpening>09:00:00</sundayOpening>
    <sundayClosing>21:00:00</sundayClosing>
  </openingtimes>
  <freeTime>00:15:00</freeTime>
  <priceFirstHour>2.5</priceFirstHour>
  <priveFurtherHour>1.5</priveFurtherHour>
  <parkingPlacesForWomen>true</parkingPlacesForWomen>
  <parkingPlacesForHandicapped>true</parkingPlacesForHandicapped>
  <availableForProfessors>true</availableForProfessors>
  <availableForWorkers>true</availableForWorkers>
  <availableForStudents>true</availableForStudents>
  <availableForGuests>false</availableForGuests>
</parkingPlace>
</list>
</list>
```

Table 3: XML-Datei "Park Liste"

1.2 XML-Schemata

Die XML-Schemata verdeutlichen die Strukturierung der Daten noch einmal vor.

```
<?xml:stylesheet type="text/xsl" href="style.xsl" />
<?xml:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="newuser">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="email"/>
        <xs:element type="xs:string" name="password"/>
        <xs:element type="xs:int" name="usergroup"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Table 4: XML-Schemata "Registrierung"

```

<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="list">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="list">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="parkingPlace">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element type="xs:int" name="id"/>
                    <xs:element type="xs:int" name="parkingPlaces"/>
                    <xs:element type="xs:int" name="freeParkingPlaces"/>
                    <xs:element type="xs:double" name="longitude"/>
                    <xs:element type="xs:double" name="latitude"/>
                    <xs:element type="xs:float" name="name"/>
                    <xs:element type="xs:float" name="distanceToFH"/>
                    <xs:element type="xs:string" name="addressStreet"/>
                    <xs:element type="xs:string" name="addressNumber"/>
                    <xs:element type="xs:int" name="addressPostalCode"/>
                    <xs:element type="xs:string" name="addressCity"/>
                    <xs:element name="openingtimes">
                      <xs:complexType>
                        <xs:sequence>
                          <xs:element type="xs:time" name="mondayOpening"/>
                          <xs:element type="xs:time" name="mondayClosing"/>
                          <xs:element type="xs:time" name="tuesdayOpening"/>
                          <xs:element type="xs:time" name="tuesdayClosing"/>
                          <xs:element type="xs:time" name="wednesdayOpening"/>
                          <xs:element type="xs:time" name="wednesdayClosing"/>
                          <xs:element type="xs:time" name="thursdayOpening"/>
                          <xs:element type="xs:time" name="thursdayClosing"/>
                          <xs:element type="xs:time" name="fridayOpening"/>
                          <xs:element type="xs:time" name="fridayClosing"/>
                          <xs:element type="xs:time" name="saturdayOpening"/>
                          <xs:element type="xs:time" name="saturdayClosing"/>
                          <xs:element type="xs:time" name="sundayOpening"/>
                          <xs:element type="xs:time" name="sundayClosing"/>
                        </xs:sequence>
                      </xs:complexType>
                    </xs:element>
                    <xs:element type="xs:time" name="freeTime"/>
                    <xs:element type="xs:double" name="priceFirstHour"/>
                    <xs:element type="xs:double" name="priceFurtherHour"/>
                    <xs:element type="xs:string" name="parkingPlacesForWomen"/>
                    <xs:element type="xs:string" name="parkingPlacesForHandicapped"/>
                    <xs:element type="xs:string" name="availableForProfessors"/>
                    <xs:element type="xs:string" name="availableForWorkers"/>
                    <xs:element type="xs:string" name="availableForStudents"/>
                    <xs:element type="xs:string" name="availableForGuests"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Table 5: XML-Datei "XML-Schemata "Park Liste"

1.3 Entity-Relationship-Modellierung

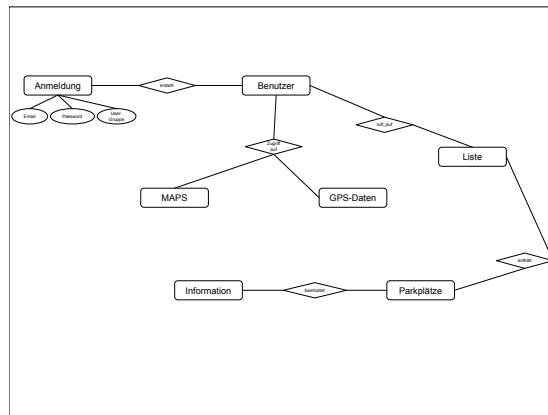


Table 6: ER-Modellierung zur Datenstruktur

Durch ein Entity-Relationship-Modellierung-Diagramm soll die gesamte Datenmodellierung dargestellt werden. Betrachtet man die Entität "Anmelden", dies beinhaltet die Attribute "Usergruppe", "Email" und "Passwort" der TH-Campus Parkplatz. Die Entität "Benutzer" hat die Möglichkeit die Entität "Map's" aufzurufen, dies enthält die Attribute "GPS Daten" und "Umgebene Parkhäuser". Außerdem kann der "Benutzer" eine Liste mit Parkhäusern und die dazugehörige Informationen mit aufrufen. Die Modellierung ist eine große Hilfe um zu zeigen welche Entitäten vorhanden sind, und wie diese in Zusammenhang stehen.