

## Datenstrukturen:

Im Folgenden werden die Datenstrukturen der Ressourcen, welche zwischen Client und Server ausgetauscht werden sollen, spezifiziert. Im Anschluss wird eine REST-Tabelle angelegt, welche diese Ressourcen mit ihren möglichen Verben, ihrer Bedeutung und Datentypen für die Anfragen bzw. Antworten und möglichen Statuscodes angegeben.

### **Ressource: User**

```
{  
  "Id" : "Einzigartige EintragsID"  
  "uid" : "Einzigartige Benutzerid"  
  "Token" : "Firebase Token eines Benutzers"  
  "Vorname" : "Vorname des Benutzers"  
  "Nachname" : "Nachname des Benutzers"  
  "Straße" : "Straße des Benutzers"  
  "Hausnummer" : "Hausnummer des Benutzers"  
  "Stadt" : "Stadt des Benutzers"  
  "Latitude" : "Breitengrad des Standorts"  
  "Longitude" : "Längengrad des Standorts"  
  "Bild" : "Bild des Benutzers"  
  "email" : "E-Mail Adresse des Benutzers"  
  "Einschränkungen" : "Liste von Einschränkungen des Benutzers"  
  "Einträge" : "Liste von Lebensmitteleinträgen"  
  "Angebote" : "Liste von Angeboten eines Benutzers"  
  "Reservierungen" : "Liste von Reservierungen eines Benutzers"  
}
```

Die User Ressource ist eine Primärressource, welche zunächst einmal alle Eigenschaften eines Benutzers enthält. Viele dieser Eigenschaften sind jedoch von der Datenstruktur her umfangreicher und es ist zu erwarten, dass oftmals Änderungen vorkommen. Um nicht immer die gesamte Primärressource zu verändern, wodurch viel Aufwand entstehen würde, wurden weitere Subressourcen definiert. Zu diesen Ressourcen gehören: „Angebote“, „Einträge“, „Einschränkungen“, „Angebote“, „Reservierungen“ und „Token“. Auf diese Subressourcen wird im Folgenden weiter eingegangen. Die Id ist die einzigartige ID für die Datenbank, während die uid, die ID ist, welche von Firebase vergeben wird, um den Benutzer, während der Authentifizierung eindeutig zu identifizieren. Die Eigenschaften Vorname, Nachname, Straße, Hausnummer, Stadt, Bild und Email sind selbsterklärend. Die Eigenschaften Latitude und Longitude bilden den Standort ab, an dem sich der Benutzer befindet.

### **Subresource: User/Einträge**

```
{  
  "Id" : "Einzigartige EintragsID"  
  "Name" : "Name des Lebensmittels"  
  "Anzahl" : "Anzahl der Einkäufe"  
}
```

In dieser Subresource werden die Lebensmittel, welche der Benutzer Abholt oder selbst Kauft festgehalten. Diese Information soll im Weiteren Verlauf dabei helfen, geeignete Angebote für den Benutzer zu finden. Diese Liste soll sich in regelmäßigen Abständen automatisch leeren, damit immer die aktuellsten Interessen des Benutzers berücksichtigt werden. Bei neu eingefügten Einträgen oder beim Löschen soll sich, falls bereits ein Eintrag vorhanden ist, die Anzahl erhöhen bzw. verkleinern.

### **Subresource: User/Einschränkungen**

```
{  
  "id" : "Einzigartige EintragsID"  
  "name" : "Name der Einschränkung"  
}
```

In dieser Subresource werden die Esseneinschränkungen des Benutzers festgehalten, welche bei der Auswahl von geeigneten Angeboten relevant sind. Bei diesen Einschränkungen kann der Benutzer zwischen: Nierenfreundlich, Vegan, Vegetarisch, Molkefrei, Glutenunverträglichkeit, Eifrei, Laktoseintoleranz, Nussallergie, Scheinefleischfrei, Alkoholfrei wählen. Die Eigenschaften des Lebensmittel werden anhand des externen Dienstgebers Edamam Nutrition Analysis durchgeführt.

### **Subresource: User/Token:**

```
{  
  "Id" : "Einzigartige EintragsID"  
  "Token" : "Firebase Token"  
  "uid" : "Benutzerid für die Zuordnung"  
}
```

Mit Hilfe dieses Tokens wird die asynchrone Kommunikation mit Firebase realisiert. Dieser Token identifiziert das mobile Endgerät des Benutzers und wird sich ggf. falls verändern, falls der Benutzer sich in einem anderen Gerät einloggt. Bei jedem Login des Benutzers muss dieser auf dem Client abgefragt werden und auf dem Server hinterlegt sein.

### **Subresource: User/Nachrichten:**

```
{  
  "id" : "Einzigartige EintragsID"  
  "sender" : "Einzigartige BenutzerID des Senders"  
  "empfaenger" : "Einzigartige BenutzerID des Empfängers"
```

*"inhalt" : "Inhalt der Nachricht"*

*"zeit" : "Zeitpunkt an dem die Nachricht versendet wurde"*

}

Diese Subressource ist für den Nachrichtenaustausch zwischen Anholder und Anbieter verantwortlich. Die id ist für die Speicherung der Nachricht innerhalb der Datenbank notwendig. Die Attribute Sender und Empfänger sind selbstbeschreibend, an dieser Stelle ist die einzigartige BenutzerID nötig, um entsprechend den notwendigen Token für die Kommunikation zu ermitteln, welche zuvor beschrieben wurde.

#### **Primärressource: Angebote**

{

*"id" : "Einzigarte EintragsID"*

*"uid" : "BenutzerID des Erstellers"*

*"status" : "Status des Angebots"*

*"Zeit" : "Erstellungszeitpunkt des Angebots"*

*"Titel" : "Titel des Angebots"*

*"Lebensmittel" : "Lebensmittel, welches Angeboten wird"*

*"Lebensmittelliste" : "Lebensmittelliste, welche das Angebot enthält"*

*"Offizielles Ablaufdatum" : "Ablaufdatum des Lebensmittels"*

*"Geschätztes Ablaufdatum" : "Geschätztes Ablaufdatum des Lebensmittels"*

*"Beschreibung" : "Zusätzliche Beschreibung des Angebots"*

*"Latitude" : "Breitengrad des erstellen Angebots"*

*"Longitude" : "Längengrad des erstellen Angebots"*

*"Einschränkungen" : "Ermittelten Einschränkungen für das Angebot"*

*"Entfernung" : "Ermittelte Entfernung des Angebots zum potentiellen Käufer"*

*"Abholwert" : "Abholwert, welcher Beschreibt, wie gut sich das Angebot für den Käufer eignet"*

}

Die Ressource bildet die angebotenen Lebensmittel der Benutzer ab. Ein Angebot hat wie auch die anderen Ressourcen eine ID und selbstbeschreibende Attribute wie: „Titel“, „Lebensmittel“, „Zeit“ und „Beschreibung“. Jedes Angebot muss ebenfalls die BenutzerID des Erstellers enthalten. Die Attribute Latitude, Longitude werden automatisch von dem jeweiligen Benutzer übernommen und wie Einschränkungen werden wie zuvor beschrieben mit Hilfe eines externen Dienstes automatisch übernommen. Die Entfernung und der Abholwert sind Attribute, welche erst dann vorliegen können, wenn ein Angebot von einem konkreten Benutzer (potentieller Abholder) angefragt wird. Diese werden benutzerbezogen serverseitig berechnet. Eine detaillierte Beschreibung der Ermittlung dieser Werte findet im Abschnitt: „Beschreibung der Anwendungslogik“ statt.

### **Primärressource: Reservierungen**

```
{  
  "id" : "Einzigartige EintragsID"  
  "aid" : "Einzigartige AngebotsID"  
  "kid" : "Einzigartige BenutzerID des Abholers"  
  "eid" : "Einzigartige BenutzerID des Abholers"  
  "zeit" : "Erstellungszeitpunkt der Reservierung"  
}
```

Die User Ressource ist eine Primärressource, welche Reservierungen eines Benutzers innerhalb des Systems abbildet. Hat ein Benutzer ein Angebot für interessant empfunden und möchte dieses entgegennehmen, so kann er eine Reservierung für dieses erstellen und sich mit dem Käufer in Kontakt setzen. Wenn ein Benutzer ein Angebot reserviert, so muss darauf ein Statusänderung des Angebots erfolgen, damit andere Benutzer dieses Angebot nicht ebenfalls reservieren können.