

Datenstrukturen

Die folgenden Datenstrukturen werden in einem JSON Schema, wie es als mongoose-Schema genutzt werden kann, dargestellt. Dabei enthält jedes Document automatisch eine `_id` mit dem Typ `ObjectId`, was hier der Vollständigkeit wegen ebenfalls angegeben wird. Die Keys werden zur besseren Verständlichkeit und Eindeutigkeit auf deutsch gelistet. Jedoch ist es möglich, dass sich bei der späteren Implementation um englische Keys handeln wird.

Collection: vergleichbar mit Tabelle in SQL-DBS

Document: ein Eintrag in einer Collection

Hochschule

```
{
  _id: ObjectId,
  name: String,
  position: {
    lat: Number,
    long: Number
  },
  studiengaenge: [{ type: ObjectId, ref: 'Studiengang' }]
}
```

Studiengang

```
{
  _id: ObjectId,
  name: String,
  hochschule: { type: ObjectId, ref: 'Hochschule' },
  beitraege: [{ type: ObjectId, ref: 'Beitrag' }]
}
```

Beitrag

```
{
  _id: ObjectId,
  titel: String,
  text: String,
  typ: { type: String, enum: ['NORMAL', 'ERFAHRUNG', 'ALUMNI'] },
  empfehlung: Boolean,
  benutzer: { type: ObjectId, ref: 'Benutzer' },
  studiengang: { type: ObjectId, ref: 'Studiengang' },
  kommentare: [{ type: ObjectId, ref: 'Beitrag' }]
}
```

Benutzer

```
{
  _id: ObjectId,
  name: String,
  email: String,
  typ: { type: String, enum: ['STUDENT', 'ALUMNI',
    'INTERESSENT'] },
  verifiziert: Boolean
}
```

Verifikation

```
{
  _id: ObjectId,
  code: String,
  email: String,
  benutzer: { type: ObjectId, ref: 'Benutzer' }
}
```

TLDs

```
[ Array(String) ]
```

Die Top-Level-Domains sind als Ressource nicht in der Datenbank sondern in einer automatisch erstellten JSON-Datei enthalten, die aus einem Array mit Strings besteht.

Ressourcen

Ressource	Methode	Beschreibung	Content-Type
/courses	GET	Alle Studiengänge auflisten	application/json
/courses/:id	GET	Studiengang mit ID ausgeben	application/json

Hinzufügen, bearbeiten und löschen von Studiengängen sind administrative Aufgaben.

Ressource	Methode	Beschreibung	Content-Type
/entries/:id	GET	Thread ab Element mit ID ausgeben	application/json
/entries	POST	Neues Thread-Element anlegen	application/json
/entries/:id	PUT	Thread mit ID updaten	application/json
/entries/:id	DELETE	Thread mit ID löschen	application/json

Ressource	Methode	Beschreibung	Content-Type
/users	POST	Benutzer hinzufügen	application/json
/users/:id	GET	Benutzer ausgeben	application/json

/users/:id	PUT	Benutzerdaten updaten	application/json
/users/:id	DELETE	Benutzer löschen	application/json

Ressource	Methode	Beschreibung	Content-Type
/verification	POST	Neue Benutzerverifikation beantragen	application/json
/verification/:code	GET	Benutzerverifikationsdaten anhand von Code ausgeben	application/json

Ressource	Methode	Beschreibung	Content-Type
/tlds	GET	Liste aller Hochschul Top-Level-Domains ausgeben	application/json