

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.