FF Operation

Anleitung

bis v1.16.x

Open Source Software
entwickelt durch JK Effects
von Julian Krauser
16. November 2026

Inhaltsverzeichnis

Inhaltsverzeichnis II
1 Einleitung 1
2 Installation 2
2.1 Docker 2
2.2 Docker-Compose 3
2.3 Git 7
2.4 Datenbank 9
2.5 Konfiguration 10
2.6 Serverzustand 15
2.7 Secret-Rotation 15
2.8 Logging 16
2.9 Health-Checks 17
2.10 API-Dokumentation (OpenAPI) 17
2.11 Dateiablage 18
2.12 Update der Version 19
2.12.1 Support-Linien 19
2.13 WebApp 20
2.14 Einrichtung 20
3 Konzepte 22
3.1 Stammdaten 22
3.2 Berechtigungen 22
3.3 Zeitformate 23
3.3.1 Gültigkeitsdauer 23
3.3.2 Zeitintervalle 23
3.4 Rechtschreibprüfung 24
3.5 Dateien und Bilder 24
3.6 Engines 25
3.6.1 Notification-Engine 25
4 Module 26
4.1 Einsatzdaten und Dokumentation 26
4.1.1 Eintragsdaten 27
4.1.2 eingesetztes Material 28
4.1.3 Anwesenheit 29
4.1.4 Protokoll 29

4.1.5 Karte ... 30
4.1.6 Sperren, Freischalten und Löschen ... 32
4.1.7 Synchronisierung mit FF Webpage ... 32
4.2 Incident-Management ... 34
4.2.1 Einsatzdaten von außen ... 34
4.2.2 Einsatzdaten-Schnittstelle ... 35
4.2.3 Verwendung der Schnittstelle ... 36
4.2.4 Einstellungen zur Verwendung ... 38
4.3 Eintragsart ... 39
4.4 Kräfte ... 40
4.5 Ausrüstung ... 41
4.6 Fahrzeuge ... 42
4.7 Kleidung ... 43
4.8 Marker und Markertypen ... 44
4.8.1 Kartentyp und Offline-Nutzung ... 45
4.8.2 Erstellung von Markern ... 46
4.9 Statistik ... 47
4.9.1 Verfügbare Auswertungen ... 47
4.9.2 Abfrage und Darstellung ... 48
4.10 Modulspezifische Einstellungen ... 49
4.11 Anwendungsspezifische Einstellungen ... 51
4.12 Modulbezeichnungen ... 52
4.12.1 Bezeichnungen ... 53
4.12.2 Module deaktivieren ... 53
4.12.3 Abhängigkeiten ... 53
4.12.4 Gesperrte Module ... 54
4.13 Benutzerverwaltung ... 54
4.13.1 Benutzer einladen ... 55
4.13.2 Verwaltete Konten ... 56
4.13.3 Konten deaktivieren ... 57
4.13.4 Details ... 57
4.13.5 Berechtigungen ... 58
4.13.6 Rollen ... 58
4.14 Rollenverwaltung ... 59
4.14.1 Details ... 60
4.14.2 Berechtigungen ... 60
4.15 WebApi ... 61
4.15.1 Details ... 62

4.15.2 Berechtigungen ... 63
4.15.3 WebApi Zugriffsablauf ... 63
4.15.4 API-Clients ... 64
4.16 Backups ... 65
4.16.1 Backup wiederherstellen ... 66
4.16.2 Daten exportieren ... 69
4.16.3 Backup hochladen ... 70
4.17 Logs ... 71
4.17.1 Ansicht ... 72
4.17.2 Protokollierung ändern ... 72
4.18 Anwendungsinformationen und Versionsmanagement ... 73
4.18.1 Anwendungsinformationen ... 73
4.18.2 Versionsmanagement ... 74
5 Benutzerbereich und Accounteigenschaften ... 76
5.1 Zugang und Anmeldedaten ... 76
5.1.1 Anmeldeoptionen ... 76
5.1.2 Änderung der Anmeldedaten ... 77
5.2 Benachrichtigungen ... 78
5.2.1 Benachrichtigungsanzeige ... 78
5.2.2 Benachrichtigungseinstellungen ... 78
5.2.3 Geplante Benachrichtigungen ... 79
5.3 Push-Benachrichtigungen ... 80
5.3.1 Funktionsweise ... 80
5.3.2 Push aktivieren ... 80
5.3.3 Übersicht und Verwaltung ... 81
5.3.4 Deaktivieren und Löschen ... 81
5.3.5 Lebenszyklus einer Push-Subscription ... 82
5.3.6 Mehrere Benutzer auf einem Gerät ... 82
5.4 Sessionverwaltung ... 82
5.4.1 Geräteerkennung und Datenschutz ... 83
5.4.2 Verwaltung aktiver Sessions ... 83
5.5 Passkeys ... 83
5.5.1 Erstellung eines Passkeys ... 84
5.5.2 Login mit Passkeys ... 84
5.6 Übertragung Administration ... 85
5.6.1 Durchführung ... 85
5.7 Wissensartikel ... 86
5.7.1 Anlegen und Bearbeiten ... 86

5.7.2 Freigaben 86
5.7.3 Löschen 87
6 Ökosystem FF Admin 88
7 Roadmap 89

1 Einleitung

FF Operation - Flexibles Einsatz- und Übungsmanagement für Feuerwehren und Vereine

FF Operation ist eine leistungsfähige Verwaltungssoftware und ein zentraler Bestandteil des FF-Ökosystems. Neben Feuerwehreinsätzen oder Übungen können auch Arbeitseinsätze von Vereinen erfasst werden. Durch den modularen Aufbau und frei definierbare Stammdaten kann die Software flexibel an unterschiedliche Organisationen angepasst werden.

FF Operation ermöglicht darüber hinaus die kollaborative Erstellung von Einsatzberichten, sowie die Synchronisierung von Daten, wenn ein Gerät zwischenzeitlich offline war.

Wird FF Admin eingesetzt, kann es als zentrale Datenquelle für Basisinformationen der Anwendung dienen.

2 Installation

FF Operation kann über mehrere Wege betrieben werden. Zum einen werden Docker-Images versioniert zur Verfügung gestellt. Weiterhin kann auch das Release Projekt heruntergeladen und verwendet werden.

2.1 Docker

Disclaimer: Die Anleitung zum Betrieb von FF Operation mit Docker setzt Kenntnisse mit Docker und Docker-Compose voraus.

Die Docker-Images können gemeinsam über eine Compose-File konfiguriert und gestartet werden. Auch können die Images einzeln gestartet werden.

Die Docker-Images sind versioniert. Für den `<tag>` des Images stehen folgende Varianten zur Verfügung:

- `vX` (z.B. `v1`): neueste Version der Hauptversion. **Empfohlen**, da alle neuen Funktionen und Fehlerbehebungen ohne grundlegende Umstellungen enthalten sind.
- `vX.Y` (z.B. `v1.6`): neueste Patch-Version einer Support-Linie (Abschnitt 2.12.1). Geeignet, um bewusst auf einem Funktionsstand zu bleiben und trotzdem Sicherheits- und Fehlerbehebungen zu erhalten.
- `vX.Y.Z` (z.B. `v1.6.4`): genau diese Version, z.B. für ein Rollback.
- `latest`: höchste veröffentlichte Version insgesamt, auch über eine neue Hauptversion hinweg.

Jeder Tag existiert zusätzlich mit den Endungen `-arm` und `-amd` für eine einzelne Architektur. Die Versionen können auch in den Releases der Repositories der Anwendungen nachgeschlagen werden. Dort lassen sich auch auch Informationen zu neuen Funktionen, Änderungen oder Fehlerbehebungen der jeweiligen Funktion finden.

2.2 Docker-Compose

App

```yaml
ff-operation-app:
  image: code.jk-effects.cloud/ff-admin/ff-operation/app:<version | latest>
  container_name: ff_operation
  restart: unless-stopped
  ports:
    - "80:80"
  environment:
    - SERVERADDRESS=<backend_url> # optional, bei abweichender URL des Backends
    - FRAME_ANCESTORS=<erlaubte Einbett-Origins, leerzeichengetrennt> # optional
```

Die Verwendung der Werte des Typs Environment werden unter dem Punkt Konfiguration (Abschnitt 2.5) erklärt.
Die Environment-Variable `SERVERADDRESS` ist optional und kann für eine Abweichende URL des Backends verwendet werden. Ist die Serveradresse nicht angegeben, wird versucht das Backend unter der selben URL mit dem PathPrefix `/api` zu erreichen.

Die Environment-Variable `FRAME_ANCESTORS` ist optional und standardmäßig leer. Dadurch darf FF Operation ausschließlich von sich selbst per `<iframe>` eingebettet werden (Content-Security-Policy `frame-ancestors 'self'`). Um das Einbetten auf anderen Webseiten zu erlauben, können die erlaubten Origins mit vollständigem Schema (`https://...`), ohne Pfad und leerzeichengetrennt angegeben werden, z.B. `https://verein.de https://www.verein.de`.

Eine erweiterte Personalisierung der App mit eigenem Logo der Feuerwehr oder des Vereins ist bei der ersten Einrichtung oder unter dem Modul Einstellungen möglich. Hiervon betroffen ist das Icon im Browser-Tab, jede Anzeige des FF Operation Logos innerhalb der App und das Icon, wenn die WebApp auf einem Gerät installiert wird.

Server

```yaml
ff-operation-server:
  image: code.jk-effects.cloud/ff-admin/ff-operation/server:<version | latest>
  container_name: ff_operation_server
  restart: unless-stopped
  ports:
    - "5000:5000"
  environment:
    - DB_HOST=<database host>
    - DB_PORT=<database port> # optional, da default 5432
    - DB_SCHEMA=<database schema> # optional, da default public
    - DB_NAME=<database name>
    - DB_USERNAME=<database username>
    - DB_PASSWORD=<database password>
    - APPLICATION_SECRET=<application_secret> # einmalig generieren, Wechsel nur über den Rotationsmodus (siehe Handbuch)
    - APPLICATION_SALT=<application_salt> # einmalig generieren, Wechsel nur über den Rotationsmodus (siehe Handbuch)
    - SERVER_URL=<server_url>
    - CLIENT_URL=<client_url>
    # \/ optional \/
    - JWT_SECRET=<jwt_secret> # signiert Sitzungen und WebAPI-Tokens statt APPLICATION_SECRET
    - APPLICATION_STORAGE=<application_storage>
    - USE_SECURITY_STRICT_LIMIT=<boolean>
    - SECURITY_STRICT_LIMIT_WINDOW=<time window>
    - SECURITY_STRICT_LIMIT_REQUEST_COUNT=<strict_request_count>
    - USE_SECURITY_LIMIT=<boolean>
    - SECURITY_LIMIT_WINDOW=<time window>
    - SECURITY_LIMIT_REQUEST_COUNT=<request_count>
    - TRUST_PROXY=<proxy config>
    - LOG_LEVEL=<error|warn|info|debug>
    - LOG_MAX_FILES=<count | days>
    - LOG_MAX_SIZE=<size>
    - LOG_SLOW_QUERY_MS=<milliseconds>
    - FILE_ORPHAN_REPORT_ONLY=<true|false>
    - FILE_KEY_VERSION=<number>
    - BACKUP_UPLOAD_MAX_SIZE=<megabytes>
    - API_DOCS=<off|public>
  volumes:
    - <volume|local path>:/app/files
    - <volume|local path>:/app/logs # optional, ohne Mount gehen die Logs beim Neuerstellen des Containers verloren
```

Die Verwendung der Werte des Typs Environment werden unter dem Punkt Konfiguration (Abschnitt 2.5) erklärt.

Environment Werte können optional sein oder haben Standard-Werte.
Das Fehlen einer geforderten Variable oder die falsche Angabe eines Variablen-Werts verhindert das Starten des der Anwendung.

Innerhalb dem Ordner, der dem Volume zugeordnet ist, werden Backups und Ausdrucke der geschriebenen Protokolle, Newsletter und alle weiteren Dokumente abgelegt, die hochgeladen oder erstellt werden können. Unter `/app/logs` schreibt der Server seine Log-Dateien (siehe Abschnitt 2.8). Der Mount dieses Ordners ist optional: Ohne Volume funktioniert das Logging ebenfalls, die Dateien gehen aber verloren, sobald der Container neu erstellt wird (z.B. bei einem Update).

Datenbank
Als Datenbank wird Postgres verwendet:

```yaml
ff-db:
  image: postgres:<version (bsp 18)>
  container_name: ff_db
  restart: unless-stopped
  ports:
    - "5432:5432"
  environment:
    - POSTGRES_DB=<database name>
    - POSTGRES_USER=<username>
    - POSTGRES_PASSWORD=<user password>
  volumes:
    - <volume|local path>:/var/lib/postgresql/data
```

`POSTGRES_DB` erstellt direkt eine Datenbank, die durch einen angelegten `POSTGRES_USER` verfügbar ist.

Hinweis Wenn eine Docker-Compose Datei verwendet wird, kann zusätzliche ein Netzwerk angelegt werden. Dadurch ist das Veröffentlichen der Datenbank-Ports nicht mehr notwendig. Das Entfernern der Port-Exposes verhindert den direkten Zugriff auf die Ports von außerhalb.
Hierfür muss das verwendete `network` dem Backend und dem Datenbank-Container bekanntgegeben werden:

1. Ergänzung zu Server und Datenbank Container:

```yaml
#volumes:
#  ...
networks:
  - ff_internal
```

2. Ergänzung zur Docker-Compose:

```yaml
networks:
  ff_internal:
```

3. Neue Host-URL der Datenbank des Servercontainers:

```yaml
environment:
  - DB_HOST: ff-db
  # ...
```

Wenn die Datenbank über ein Netzwerk in einer Compose-Datei freigegeben wird, kann als Host der Service-Name der Datenbank angegeben werden. In der angegebenen Datenbank-Konfiguration wäre das `ff-db`.

Optionale Ergänzung zum Servercontainer:

```yaml
#volumes:
#  ...
#networks:
#  - ff_internal
depends_on:
  - ff-db
```

Hierdurch kann der Server nicht starten, wenn die verwendete Datenbank nicht läuft.

Proxy

Damit die App und der Server aus dem Internet über URLs erreichbar sind, kann Traefik oder Nginx als Reverse-Proxy verwendet werden.
Bei der Konfiguration des Proxies ist Folgendes zu beachten:

- Die Tatsächliche IP sollte an den Server weitergeleitet werden
- Alle HTTP-Header müssen an den Server weitergeleitet werden
- Die maximale Größe für Dateiuploads sollte nicht zu stark eingeschränkt werden

Die Weiterleitung der Header kann im Serverlog überprüft werden. API-Anfragen werden im Format `:remote-addr :method :url :status - :response-time ms | isPWA: :headers["x-pwa-client"] - device: :headers["x-device-name"]` protokolliert. Fehlen die Werte nach „isPWA:“ oder „device:“, werden die entsprechenden Header nicht weitergeleitet.

Hinweis: Bei Anfragen von externen APIs oder beim Zugriff auf Logo bzw. Favicon existieren keine benutzerdefinierten Header, da diese nur von der WebApp über den Browser gesendet werden.

2.3 Git

Eine Alternative zu Docker ist die direkte Ausführung der Anwendungen auf dem Server oder einem Desktop Gerät.
Hierzu müssen die App und der Server als Quellcode auf das ausführende System geladen und dort direkt verwendet werden. Das System muss NodeJs 24 und Postgres installiert haben.

Wie bei den Docker-Tags kann auch im Git ein Versionsstand gewählt werden, der mit `git checkout <branch/tag>` ausgecheckt wird:
- `main`: neueste veröffentlichte Version, entspricht dem Docker-Tag `latest`.
- `release/X.Y` (z.B. `release/1.6`): neueste Patch-Version einer älteren Support-Linie (Abschnitt 2.12.1), entspricht dem Docker-Tag `vX.Y`. Die aktuelle Linie liegt in `main`.
- `vX.Y.Z` (z.B. `v1.6.4`): Git-Tag für genau diese Version, z.B. für ein Rollback.

Der Branch `develop` enthält den Entwicklungsstand und ist nicht für den Betrieb gedacht.

Um die Konfiguration mittels ENV-Variablen an die Anwendungen weitergeben zu können, müssen `.env` Dateien erstellt werden. Hierzu kann die `.env.example` Datei kopiert und die definierten Werte ausgefüllt werden. Nicht benötigte Einträge sollten entfernt werden.
Die env-Datei im Frontend muss vor dem build-Prozess erstellt sein, da dort die Werte fest in den Code übernommen werden. Weiterhin muss die Datei im Frontend `.env.production` heißen. Die bestehende Datei kann modifiziert werden. Bei einer Änderung muss die App neu gebaut werden.
Wird der Server unter der gleichen URL wie die App verwendet, muss in der `.env.production` Datei des Frontends der Wert zum Eintrag von `SERVERADDRESS` auf den leeren String gesetzt werden.
Die env-Datei im Backend muss vor der Ausführung von `npm run start` angelegt sein. Bei einer Änderung der Einträge muss der Server lediglich neu gestartet werden.

Die NodeJs Prozesse können auch durch Tools wie pm2 verwaltet werden.
Damit die App und der Server aus dem Internet erreichbar sind, muss das Routing gesondert eingerichtet werden.

App

```shell
git clone https://code.jk-effects.cloud/ff-admin/ff-operation.git
cd ff-operation
npm install
npm run build
```

Der mit `npm run build` erstellte dist Ordner kann mit Apache oder Nginx zur veröffentlicht werden.

Für das Hosting der statischen Inhalte der App kann Apache oder Nginx verwendet werden. Eine Konfiguration für Nginx ist im Repo der App enthalten:

```conf
worker_processes 4;

events { worker_connections 1024; }

http {
  include mime.types;
  server_tokens off;

  server {
    root /usr/share/nginx/html; # Pfad zum dist Ordner des Frontends
    index index.html;

    # __SERVERADDRESS__ env var used for the app's own API base URL
    # __SERVERADDRESS_WS__ the wss:// (or ws://) equivalent of __SERVERADDRESS__
    # __FRAME_ANCESTORS__ space-separated origins allowed to embed this app in <iframe>
    add_header Content-Security-Policy "default-src 'self'; script-src 'self' 'wasm-unsafe-eval'; style-src 'self' 'unsafe-inline'; img-src 'self' data: blob: __SERVERADDRESS__; font-src 'self' data:; connect-src 'self' __SERVERADDRESS__ __SERVERADDRESS_WS__ https://*.tile.openstreetmap.org https://server.arcgisonline.com https://tiles.ff-admin.de https://tiles.openrailwaymap.org https://lang.ff-admin.de; frame-src 'self' blob:; manifest-src 'self' __SERVERADDRESS__; object-src 'none'; base-uri 'self'; form-action 'self'; frame-ancestors 'self' __FRAME_ANCESTORS__" always;
    add_header X-Content-Type-Options "nosniff" always;
    add_header X-Frame-Options "SAMEORIGIN" always;
    add_header Referrer-Policy "strict-origin-when-cross-origin" always;
    # Only enable if this app is always served over HTTPS
    # add_header Strict-Transport-Security "max-age=31536000; includeSubDomains" always;

    location / {
      try_files $uri $uri/ /index.html;
    }
  }
}
```

Da die Nginx Konfiguration den Zugriff und die Einbettung einschränken, müssen die Variablen `__SERVERADDRESS__` und `__FRAME_ANCESTORS__` manuell angepasst werden.

Server

```Shell
git clone https://code.jk-effects.cloud/ff-admin/ff-operation-server.git
cd ff-operation-server
npm install
npm run build
npm run start
```

Die Log-Dateien werden im Ordner `logs` innerhalb des Arbeitsverzeichnisses des Servers abgelegt (siehe Abschnitt 2.8).

Proxy
Damit die App und der Server aus dem Internet über URLs erreichbar sind, kann Traefik oder Nginx als Reverse-Proxy verwendet werden.
Bei der Konfiguration des Proxies ist Folgendes zu beachten:
- Die Tatsächliche IP sollte an den Server weitergeleitet werden
- Alle HTTP-Header müssen an den Server weitergeleitet werden
- Die maximale Größe für Dateiuploads sollte nicht zu stark eingeschränkt werden

Die Weiterleitung der Header kann im Serverlog überprüft werden. API-Anfragen werden im Format `:remote-addr :method :url :status - :response-time ms | isPWA: :headers["x-pwa-client"] - device: :headers["x-device-name"]` protokolliert. Fehlen die Werte nach „isPWA:“ oder „device:“, werden die entsprechenden Header nicht weitergeleitet.

Hinweis: Bei Anfragen von externen APIs oder beim Zugriff auf Logo bzw. Favicon existieren keine benutzerdefinierten Header, da diese nur von der WebApp über den Browser gesendet werden.

2.4 Datenbank

Die Anwendung FF Operation unterstützt ausschließlich PostgreSQL als Datenbank, da diese Vorteile bei der Migrationsverwaltung und Rollbacks in Transaktionen mit sich bringt. Es wird empfohlen die Datenbank stets auf der neuesten Version zu halten, um auch deren neueste Features nutzen zu können.
Mit Version 1.6.x der Core-Packages wurde eine datenbank-versionsabhängige Funktionalität eingeführt, welche die Verfügbarkeit der lokalen UUID Funktionen betrifft. So wird ab PostgreSQL v18 die Generierung von UUIDv7 ermöglicht, welches Optimierungen bei dem Einfügen von Daten mit sich bringt. FF Operation prüft bei jedem Start des Servers die Datenbankversion und passt die Generierung der UUIDs intern an.

2.5 Konfiguration

Folgende Werte können zu einem Container konfiguriert werden:

Variable	Zweck	Default	Optional
⬇️ App-Variablen			
SERVERADDRESS	URL, über welche das Backend erreicht werden kann. Die URL muss mit http:// oder https:// starten. Jeder angegebene Pfad wird automatisch entfernt. Die App versucht das Backend unter <SERVERADDRESS>/api zu erreichen. Laufen Backend und App auf der gleichen URL, kann diese Variable weggelassen werden.		✅
FRAME_ANCESTORS	Leerzeichengetrennte Liste erlaubter Origins (vollständiges Schema, ohne Pfad), die die App zusätzlich zu sich selbst per <iframe> einbetten dürfen (CSP frame-ancestors).	leer (nur 'self')	✅
⬇️ Server-Variablen			
DB_HOST	URL zur Datenbank		✳️
DB_PORT	Port der Datenbank	5432	✅
DB_SCHEMA	Schema der Datenbank	public	✅
DB_NAME	Name der Datenbank		✳️
DB_USERNAME	Nutzername für Zugang zu Datenbank		✳️
DB_PASSWORD	Passwort zum Zugang zur Datenbank		✳️
APPLICATION_SECRET	Zufällige Zeichenkette zur Validierung der Session-Tokens und Kodierung von Passwörtern und Zugangsdaten.		✳️

Variable	Zweck	Default	Optional
APPLICATION_SALT	Zufällige Zeichenkette zur Kodierung von Zugangsdaten.		✴️
JWT_SECRET	Zufällige Zeichenkette zum Signieren der Session- und WebAPI-Tokens. Ist sie nicht gesetzt, wird das APPLICATION_SECRET verwendet. Ein Wechsel meldet alle Nutzer ab und macht WebAPI-Tokens ungültig.		✅
SECRET_ROTATION	Startet den Rotationsmodus mit true oder dessen Rückabwicklung mit rollback (siehe @secret-rotation).	false	✅
APPLICATION_SECRET_OLD	Bisheriges APPLICATION_SECRET, nur für den Rotationsmodus.		✅
APPLICATION_SALT_OLD	Bisheriges APPLICATION_SALT, nur für den Rotationsmodus.		✅
APPLICATION_STORAGE	Angabe des verfügbaren Speichers, alternativ zur berechneten Größe des freien Festplattenspeichers. Fromat: [0-9]*(B|KB|MB|GB|TB)		✅
SERVER_URL	URL, über welche das Backend erreicht werden kann. Die URL muss mit http:// oder https:// starten. Ein angegebener Pfad verursacht eine Fehlermeldung beim Start.		✴️
CLIENT_URL	URL, über welche der Client erreicht werden kann. Die URL muss mit http:// oder https:// starten. Ein angegebener Pfad verursacht eine Fehlermeldung beim Start.		✴️
USE_SECURITY_STRICT_LIMIT	Soll ein Anfrage-Limit für Login, Reset und Co gesetzt werden?	true	✅

Variable	Zweck	Default	Optional
	In diesem Fall ist der Nutzer nicht angemeldet, sondern versucht es.		
SECURITY_STRICT_LIMIT _WINDOW	Über welches Zeitfenster soll das Limit angewandt werden? Format: [0-9]*(y|d|h|m|s)	15m	✅
SECURITY_STRICT_LIMIT _REQUEST_COUNT	Wie viele fehlerhafte Anfragen müssen gesendet werden, bis das Limit aktiviert ist?	15	✅
USE_SECURITY_LIMIT	Soll ein Anfrage-Limit für Anfragen innerhalb der App gesetzt werden? In diesem Fall ist der Nutzer angemeldet.	true	✅
SECURITY_LIMIT_WINDOW	Über welches Zeitfenster soll das Limit angewandt werden? Format: [0-9]*(y|d|h|m|s)	1m	✅
SECURITY_LIMIT_REQUEST _COUNT	Wie viele fehlerhafte Anfragen müssen gesendet werden, bis das Limit aktiviert ist?	500	✅
TRUST_PROXY	Wird der Server hinter einem Proxy betrieben und Rate-Limit verwendet? Ist dieser Wert nicht gesetzt, wird davon ausgegangen, dass kein Proxy verwendet wird. Folgende Werte können gesetzt werden: true / false Anzahl der Proxies: [0-9]* IP-Adresse des Proxy: ip IP-Adressen der Proxy: ip1,ip2,...		✅
VAPID_PUBLIC_KEY	Base64url-kodierter öffentlicher VAPID-Schlüssel für Web-Push-Benachrichtigungen.		✅

Variable	Zweck	Default	Optional
VAPID_PRIVATE_KEY	Base64url-kodierter privater VAPID-Schlüssel (Gegenstück zu VAPID_PUBLIC_KEY).		✅
LOG_LEVEL	Detailgrad der Logs: error, warn, info oder debug. Jede Stufe enthält die vorherigen. SQL-Abfragen werden nur bei debug protokolliert.	info	✅
LOG_MAX_FILES	Aufbewahrung der Log-Dateien als Anzahl an Dateien oder Tagen. Format: [0-9]+(d)	14d	✅
LOG_MAX_SIZE	Maximale Größe einer Log-Datei, bevor eine neue Datei begonnen wird. Format: [0-9]+(k|m|g)	20m	✅
LOG_SLOW_QUERY_MS	Datenbankabfragen, die länger als dieser Wert in Millisekunden dauern, werden als Warnung protokolliert. 0 deaktiviert die Prüfung.	1000	✅
FILE_ORPHAN_REPORT_ONLY	Gespeicherte Dateien ohne Bezug zu einem Datenbank-Eintrag werden bei true nur im Log gemeldet, bei false täglich gelöscht (siehe @file-storage).	true	✅
VAPID_SUBJECT	Kontakt-URI für Push-Dienste (Format: mailto:admin@example.com). Wird benötigt, wenn VAPID-Schlüssel gesetzt sind.		✅
FILE_KEY_VERSION	Aktive Version der Dateischlüssel (siehe @file-storage).	1	✅
BACKUP_UPLOAD_MAX_SIZE	Maximale Größe hochgeladener Backups und Exporte in Megabyte.	200	✅
API_DOCS	Mit `public` liefert der Server die OpenAPI-Spezifikation und die API-	off	✅

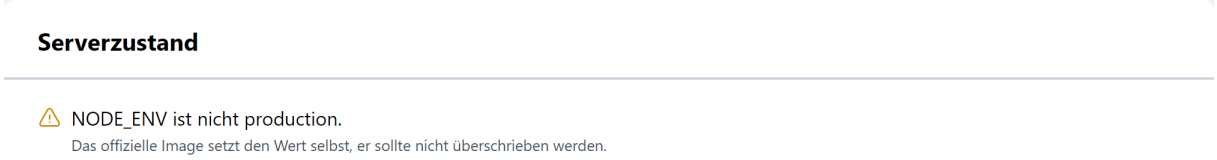
Variable	Zweck	Default	Optional
	Dokumentation aus (siehe Abschnitt 2.10). Mit `off` sind beide nicht erreichbar.		
⬇️ Database-Variablen			
POSTGRES_DB	Name der Datenbank, die bei Erstellung direkt angelegt wird.		✴️
POSTGRES_USER	Benutzername des Users, der bei Erstellung direkt angelegt wird.		✴️
POSTGRES_PASSWORD	Passwort zum User, das bei Erstellung gesetzt wird.		✴️

✴️: Ein Fehlen dieser Variable verhindert das Starten der Anwendung!

Hinweis: Eine fehlerhafte Konfiguration der optionalen oder geforderten Variable verhindert das Starten der Anwendung.
Hinweis: Ist eine Datenbank leer, wird das neueste prüfbare Backup automatisch wiederhergestellt, sollte ein solches existieren.
Hinweis: APPLICATION_SECRET und APPLICATION_SALT werden zur Verschlüsselung von Zugangsdaten in der Datenbank und von Dateien (Abschnitt 2.11) verwendet. Beide Werte dürfen nur über den Rotationsmodus gewechselt werden (Abschnitt 2.7). Werden sie einfach geändert, sind Zugangsdaten und verschlüsselte Dateien nicht mehr lesbar, bis die bisherigen Werte wieder eingetragen sind. Bekannte Platzhalter wie `<application_secret>` oder `mysecret` verhindern das Starten der Anwendung.
Daten können entsprechend über ein Backup wiederhergestellt werden, jedoch werden dabei Zugangsdaten (Credentials) nicht überschrieben oder entfernt. Diese können nur über die Benutzeroberfläche geändert werden, da spezielle Zugänge wie der E-Mail-Versand bei Änderungen validiert werden müssen. Nutzer-Passwörter werden nach einem Backup-Restore nicht zurückgesetzt, um Fremdzugriffe zu verhindern.
In einem Fall, bei dem Zugangsdaten nicht mehr stimmen und der Mailversand nicht klappt, muss die Anwendung neu aufgesetzt werden.
Hinweis: Die SERVER_URL wird benötigt, damit die Backend-Anwendung Links zu sich selbst korrekt setzen kann. Sie schränkt nicht ein, über welche Domains das Backend erreichbar ist.
Hinweis: Die CLIENT_URL wird benötigt, damit die Backend-Anwendung Informationen zur Version des Clients auslesen kann. Zusätzlich ist die CLIENT_URL die einzige erlaubte CORS-Origin für die REST-API und die WebSocket-Verbindung: Der Wert muss exakt der Origin entsprechen, unter welcher die App im Browser aufgerufen wird (Protokoll, Host und ggf. Port), da sonst Anfragen des

Browsers an das Backend blockiert werden. Server-zu-Server-Aufrufe, z.B. über die API-Clients (Abschnitt 4.15.4), sind von dieser Einschränkung nicht betroffen.
Hinweis: Sind keine VAPID-Schlüssel (`VAPID_PUBLIC_KEY`, `VAPID_PRIVATE_KEY`) konfiguriert, generiert der Server bei jedem Start ein neues Schlüsselpaar und gibt dieses im Serverlog aus. Diese generierten Schlüssel können aus dem Log kopiert und als Umgebungsvariablen gesetzt werden. Ohne persistente Schlüssel werden bestehende Push-Subscriptions bei jedem Neustart ungültig, da sich der öffentliche Schlüssel ändert.

2.6 Serverzustand



Unter Verwaltung → Serverzustand erklärt die Anwendung Administratoren und dem Besitzer Probleme der Konfiguration, z.B. zu schwache Secrets, `TRUST_PROXY=true`, fehlende VAPID-Schlüssel, einen nicht erreichbaren Mailserver oder WebAPI-Zugänge mit Admin-Rechten ohne Ablaufdatum. Bei Fehlern erscheint zusätzlich ein Hinweis im Admin-Bereich. Auch die Berichte des Rotationsmodus werden hier angezeigt und quittiert.

2.7 Secret-Rotation

Der Rotationsmodus wechselt APPLICATION_SECRET und/oder APPLICATION_SALT und schlüsselt alle davon abhängigen Daten um, z.B. nach einem Verdacht auf Kompromittierung oder um einen schwachen Wert zu ersetzen.

1. Backup der Datenbank und des Ordners `/app/files` anlegen. Die Anwendung nicht im selben Schritt aktualisieren: Update und Rotation müssen getrennt erfolgen, die Anwendung sollte vorher einmal normal mit der neuen Version gestartet sein.
2. Neue Werte eintragen und die Rotation einschalten. Für Werte, die gleich bleiben, entfällt die jeweilige `_OLD`-Variable:

```yaml
- SECRET_ROTATION=true
- APPLICATION_SECRET=<neuer Wert, mindestens 32 zufällige Zeichen>
- APPLICATION_SALT=<neuer Wert, mindestens 32 zufällige Zeichen>
- APPLICATION_SECRET_OLD=<bisheriger Wert>
- APPLICATION_SALT_OLD=<bisheriger Wert>
```

3. Container neu erstellen. Die Anwendung legt einen Restore-Punkt an, schlüsselt die Datenbank in einer Transaktion und danach die Dateien um. Der Fortschritt steht im Log. Die API

bleibt währenddessen und danach gesperrt, der Prozess läuft aber weiter, damit `restart: unless-stopped` ihn nicht erneut startet.

4. Erscheint im Log `secret rotation completed`: `SECRET_ROTATION`, `APPLICATION_SECRET_OLD` und `APPLICATION_SALT_OLD` entfernen und den Container erneut erstellen.
5. Die im Serverzustand (Abschnitt 2.6) gemeldeten Schritte ausführen und den Bericht quittieren.

Stimmen die alten Werte nicht, bricht die Rotation ohne Änderung ab. Schlägt die Umschlüsselung einzelner Dateien fehl, setzt ein Neustart mit denselben Variablen sie fort. Ein wiederholter Start nach einer abgeschlossenen Rotation ändert nichts.
Der Bericht liegt als Datei unter `/app/files/secret-rotation` und enthält keine geheimen Werte.

Gewechselt	Daten umgeschlüsselt	Nutzer abgemeldet	WebAPI-Tokens ungültig
nur Salt	ja	nein	nein
Secret oder beides	ja	ja	ja, außer `JWT_SECRET` ist gesetzt

Rotation zurücknehmen: `SECRET_ROTATION=rollback` stellt Datenbank, Dateien und Backups im aktuellen Format auf die bisherigen Schlüssel zurück, Änderungen seit der Rotation bleiben erhalten. Der Ablauf entspricht der Rotation, nur sind die Werte vertauscht: `APPLICATION_SECRET` und `APPLICATION_SALT` enthalten die bisherigen Werte, `APPLICATION_SECRET_OLD` und `APPLICATION_SALT_OLD` die der Rotation. Soll auch der Inhalt zurück, wird danach der Restore-Punkt der Rotation eingespielt. Der Restore-Punkt allein reicht nicht, da er weder Dateien noch gespeicherte Zugangsdaten der Nutzer zurückstellt.
Achtung: Backups im aktuellen Format stellt die Rotation auf die neuen Schlüssel um, ausgenommen ihren eigenen Restore-Punkt. Backups im alten Format werden bei einer leeren Datenbank nicht mehr automatisch wiederhergestellt, ihre verschlüsselten Werte sind nur lesbar, solange die `_OLD`-Variablen gesetzt sind. Exporte bleiben lesbar, da sie nur durch ihr eigenes Passwort geschützt sind.
Ein Wechsel von `JWT_SECRET` allein benötigt keinen Rotationsmodus.

2.8 Logging

Der Server schreibt alle Meldungen gleichzeitig auf die Konsole und in Log-Dateien im Ordner `logs` (im Docker-Container `/app/logs`). Pro Tag wird eine Datei `app-JJJJ-MM-TT.log` angelegt; wird die mit `LOG_MAX_SIZE` festgelegte Größe überschritten, wird eine weitere Datei begonnen. Ältere Dateien werden entsprechend `LOG_MAX_FILES` automatisch gelöscht. Erfasst werden Meldungen der Anwendung, aller API-Anfragen und der Datenbank.
Stürzt der Server durch einen unerwarteten Fehler ab, wird dieser zusätzlich sofort in die Datei `crash.log` geschrieben. Bei einem Abbruch von außen (z.B. durch fehlenden Arbeitsspeicher)

können die letzten Einträge fehlen. Wird der Server ohne Docker betrieben, empfiehlt sich deshalb ein Prozessmanager (z.B. systemd, pm2 oder unter Windows NSSM), welcher die Konsolenausgabe zusätzlich speichert und den Server nach einem Absturz neu startet.
Die Logs können zusätzlich in der Anwendung unter `Verwaltung > Logs` eingesehen, gefiltert und heruntergeladen werden. Dort lässt sich der Detailgrad auch vorübergehend bis zum nächsten Neustart ändern (siehe Abschnitt 4.17).
Fehler-ID: Jede Anfrage erhält eine kurze ID, die im Log bei der Anfrage (`id:`) und beim zugehörigen Fehler steht. Bei einem Serverfehler zeigt die App diese ID in der Fehlermeldung an, z.B. „(Fehler-ID: Xk3a_9Qe)“. Mit der Suche unter `Verwaltung > Logs` lässt sich der Fehler so direkt finden. Die ID wird auch im Header `X-Request-Id` jeder Antwort mitgeschickt.

2.9 Health-Checks

Die Docker-Images von Server und App enthalten einen Health-Check, den Docker und Plattformen wie Coolify oder Portainer automatisch auswerten. Eine zusätzliche Konfiguration in der compose-Datei ist nicht nötig.

- **Server:** `GET /api/health` antwortet, solange der Server läuft, auch während eine Datensicherung wiederhergestellt wird.
- **Server:** `GET /api/ready` prüft zusätzlich die Datenbankverbindung und antwortet mit Status `503`, solange die Datenbank nicht erreichbar oder die API vorübergehend deaktiviert ist. Dieser Endpunkt eignet sich für externes Monitoring (z.B. Uptime Kuma).
- **App:** `GET /health` wird direkt von nginx beantwortet.

Die Endpunkte benötigen keine Anmeldung und erscheinen nicht im Log.

2.10 API-Dokumentation (OpenAPI)

Der Server beschreibt seine gesamte REST-API in einer OpenAPI-Spezifikation (Version 3.1). Sie wird bei jeder Version aus dem Code erzeugt und enthält alle Routen, Parameter, Anfrage- und Antwortformate, Fehlercodes sowie die benötigten Berechtigungen. Die Routen des Core sind darin enthalten.
Standardmäßig ist die Spezifikation nicht erreichbar. Mit `API_DOCS=public` liefert der Server aus:

- `GET /api/openapi.json`: die Spezifikation, `info.version` entspricht der Version des Servers.
- `GET /api/docs`: eine lesbare Dokumentation der API (Scalar). Alle Dateien dafür kommen vom eigenen Server, es werden keine externen Dienste geladen.

Beide Adressen sind ohne Anmeldung abrufbar. Daten der Anwendung enthalten sie nicht, sie verraten aber Aufbau und Version der API. Wer die Dokumentation nur für die Entwicklung eigener

Anbindungen braucht, kann sie dafür vorübergehend aktivieren oder die Spezifikation aus dem Repository des Servers (`openapi.json`) bzw. aus den API-Clients (Abschnitt 4.15.4) verwenden.
Fehlerformat: Fehler beantwortet der Server standardmäßig mit einer Meldung als Text. Sendet ein Client den Header `Accept: application/problem+json`, erhält er ein Objekt nach RFC 9457:

```json
{
  "type": "about:blank",
  "title": "Forbidden",
  "status": 403,
  "detail": "missing permission for management.user.update",
  "code": "permission.missing",
  "params": { "permission": "management.user.update" },
  "requestId": "Xk3a_9Qe"
}
```

`code` ist stabil und eignet sich zur Auswertung in eigenen Anwendungen, `detail` ist ein Text für Entwickler. Validierungsfehler haben den Code `validation.failed` und listen unter `errors` je Feld den Pfad (z.B. `body.firstname`) und eine deutsche Meldung. `requestId` ist die Fehler-ID aus dem Log (Abschnitt 2.8).

2.11 Dateiablage

Alle dauerhaft gespeicherten Dateien wie Bilder, Dokumente, Ausdrucke und Backups liegen im Ordner `files` (im Docker-Container `/app/files`). Hochgeladene Dateien werden zunächst in einen temporären Ordner des Betriebssystems geschrieben und erst nach erfolgreicher Prüfung in den Ordner `files` verschoben. Übrig gebliebene temporäre Dateien werden täglich entfernt. Ein zusätzliches Volume für den temporären Ordner ist nicht nötig.
Verschlüsselte Dateien: Dateien mit personenbezogenen oder Gesundheitsdaten werden verschlüsselt (AES-256-GCM) im Ordner `files` abgelegt. In FF Operation betrifft das die Bilder des Protokolls und künftig Dokumente wie Einsatz-PDFs.. Der Schlüssel wird aus `APPLICATION_SECRET` und `APPLICATION_SALT` abgeleitet. Eine Sicherung des Ordners `files` ist daher nur zusammen mit diesen Werten nutzbar.
Wird `FILE_KEY_VERSION` erhöht, stellt der tägliche Lauf die Dateischlüssel auf die neue Version um, ohne dass sich Secret oder Salt ändern. Alle Versionen leiten sich aus denselben Werten ab, bei einem Verdacht auf Kompromittierung hilft deshalb nur der Rotationsmodus (Abschnitt 2.7).
Durch abgebrochene Anfragen oder gelöschte Datenbank-Einträge können Dateien zurückbleiben, auf die kein Eintrag mehr verweist. Der Server sucht täglich um Mitternacht nach solchen Dateien, sofern sie älter als 24 Stunden sind. Standardmäßig werden die gefundenen Dateien nur im Log gemeldet (`FILE_ORPHAN_REPORT_ONLY=true`), damit vor dem ersten Löschen geprüft werden kann, welche Dateien betroffen wären. Mit `FILE_ORPHAN_REPORT_ONLY=false` werden diese Dateien anschließend automatisch gelöscht, leere Ordner werden ebenfalls entfernt.

2.12 Update der Version

Um eine Version auf eine Neuere zu aktualisieren, muss meist nur der Docker-Tag oder das Repo ersetzt werden.
Wer Docker mit `vX`, `vX.Y` oder `latest` nutzt, kann das neue Image direkt mit `docker pull` neu beziehen und dann den Container neustarten.

Informationen zu neuen Versionen können innerhalb der App unter `Verwaltung > Version` oder in den Release-Pages gefunden werden.
Die Releases beinhalten Informationen zu einem Update und was zu beachten ist. So enthalten die Release-Informationen beispielsweise Vorbereitungen vor einem Update.

Bei Verwendung mittels Git, müssen die Repos neu bezogen werden. Anschließend müssen die Dependencies neu installiert und die Anwendungen neu gebaut werden. Bei einem ausgecheckten Branch (`main` oder `release/X.Y`) genügt dazu `git pull`, bei einem Tag muss der neue Tag ausgecheckt werden.

2.12.1 Support-Linien

Jede Minor-Version (z.B. `1.6`) bildet eine eigene Support-Linie. Neben der aktuellen Version werden ältere Linien weiter gepflegt, sodass ein Update auf eine neue Minor-Version nicht sofort erfolgen muss.
Unterstützt werden die letzten drei Minor-Versionen der aktuellen Hauptversion sowie jeweils die neueste Minor-Version jeder älteren Hauptversion. Ist z.B. `1.7` aktuell, werden `1.7`, `1.6` und `1.5` unterstützt.
Unterstützte ältere Linien erhalten Sicherheits- und Fehlerbehebungen sowie Aktualisierungen von Abhängigkeiten und Basis-Images als Patch-Version (z.B. `1.6.5`), jedoch keine neuen Funktionen. Diese Updates erscheinen automatisch unter dem Docker-Tag `vX.Y` der Linie und im Git im Branch `release/X.Y`.
Erscheint eine neue Minor-Version, endet der Support der ältesten Linie. Sie erhält danach keine Updates mehr. Das Image der letzten Patch-Version bleibt erhalten, ältere Patch-Versionen einer beendeten Linie werden aus der Registry entfernt. Instanzen auf einer beendeten Linie sollten zeitnah auf eine unterstützte Linie aktualisiert werden.
Welche Linien aktuell unterstützt werden, ist für alle Anwendungen zentral unter `support.json` einsehbar (Abschnitt `supported`, beendete Linien unter `eol`).

2.13 WebApp

FF Operation ist als WebApp verfügbar. Dadurch lässt sich die Anwendung auf einem Smartphone oder Desktop über den Browser installieren.
Sollte die Version des FF Operation aktualisiert werden, wird dem Nutzer in der Anwendung eine Information angezeigt, dass die Anwendung aktualisiert werden kann. Wird im Client eine neue Version erkannt, wird auch eine Meldung zum manuellen Download der neuen Version angezeigt.



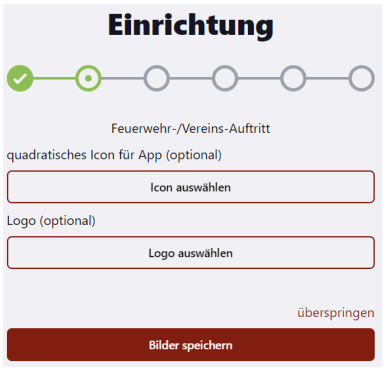
Updates werden im Hintergrund heruntergeladen, aber erst auf Klick installiert, da die Seite dabei neu geladen wird und ungespeicherte Eingaben sonst verloren gingen. Auf neue Versionen prüft die App beim Start, beim Zurückkehren in die App, nach einer Unterbrechung der Verbindung und alle 30 Minuten. Wurde das Update bereits in einem anderen Fenster installiert, weist die App darauf hin, offene Eingaben zu speichern und die Seite neu zu laden.
Kann die App eigene Dateien nicht laden oder schlägt ein Update fehl, bietet der Hinweis `App reparieren` an. Dabei werden die zwischengespeicherten App-Dateien gelöscht und neu geladen. Anmeldung, Push-Benachrichtigungen und lokal gespeicherte Daten bleiben erhalten. Ein manuelles Leeren des Browser-Caches ist dafür nicht mehr nötig.

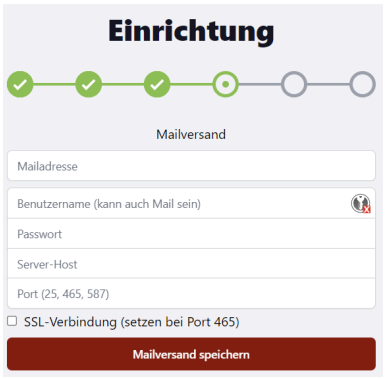
2.14 Einrichtung

Um die Anwendung nutzen zu können, kann ein erster Administrator-Account wie folgt erstellt werden:



Admin Benutzer erstellen: Um einen ersten Benutzer mit Administrator-Berechtigungen zu erstellen, muss der Einrichtungs-Assistent unter dem Pfad `/setup` aufgerufen werden. Nach der initialen Einrichtung wird der Pfad automatisch geblockt. Der Nutzer, welcher über die Einrichtung erstellt wird, wird anschließend als `Owner` gekennzeichnet und hat unabhängig der gesetzten Berechtigungen Vollzugriff auf die gesamte Anwendung.
Der Einrichtungs-Assistent ermöglicht das Setzen erster Einstellungen wie Vereinsdaten und Logos. Auch ermöglicht der Assistent das Einrichten der Mail-Adresse, die von FF Operation verwendet werden soll. Über diese Mailadresse können andere Nutzer eingeladen, Zugänge zurückgesetzt oder Newsletter versendet werden.







Rollen und Berechtigungen: Unter `Verwaltung > Rollen` können die Rollen und Berechtigungen für die Benutzer erstellt und angepasst werden.

Nutzer einladen: Unter `Verwaltung > Benutzer` können weitere Nutzer eingeladen werden. Diese erhalten dann eine E-Mail mit einem Link. Über diesen Link können die Nutzer einen Account mit Zugangsdaten erstellen. Für den Zugang können entweder TOTP oder Passwörter verwendet werden.

3 Konzepte

FF Operation basiert auf mehreren Konzepten, die das System modular und flexibel machen, so dass es an verschiedene Anwendungsfälle angepasst werden kann.

3.1 Stammdaten

Stammdaten sind grundlegende Basisdaten, die als Grundlage für weitere Einträge dienen. Dazu gehören z.B. die Eintragsart wie Einsatz/Übung... oder die auswählbaren Kräfte, Fahrzeuge und co..

Diese Daten sind frei definierbar, so dass die Benutzer ihre eigene Namensgebung festlegen und sicherstellen können, dass alle benötigten Einträge zur Verfügung stehen.

Damit Einträge wie Einsätze angelegt oder verwendet werden können, müssen **zuvor** die entsprechenden Stammdaten angelegt worden sein.

3.2 Berechtigungen

Das Berechtigungssystem ist tief in FF Operation integriert und steuert den Zugriff von Benutzern oder API-Clients auf verschiedene Sektionen und Module. Berechtigungen bestimmen, ob jemand Daten lesen, erstellen, aktualisieren oder löschen kann.

Berechtigungen werden immer summiert:
Ein Benutzer erhält alle Berechtigungen, die ihm direkt oder indirekt über Rollen zugewiesen wurden.
Ein Beispiel: Erhält ein Nutzer Leserechte zu den Mitgliedern über eine Rolle aber auch Rechte zum Erstellen direkt zugewiesen, so kann der Benutzer letztendlich Mitglieder erstellen. Die höhere Berechtigung zählt in dem Fall.

Der Berechtigungseditor ist in Sektionen und Module unterteilt. Wenn ein Benutzer eine Berechtigung für einen Abschnitt erhält, gilt diese automatisch auch für alle untergeordneten Module.

Hinweis: Um Einträge in einem Modul, das auf Stammdaten zugreift, anlegen oder bearbeiten zu können, erhält der Benutzer automatisch Leserechte auf die Stammdaten. Dies bedeutet jedoch nicht, dass der Benutzer die Stammdaten direkt in der Anwendung sehen kann - hierfür benötigt er eine explizite Berechtigung für den entsprechenden Abschnitt oder das entsprechende Modul.

3.3 Zeitformate

Es gibt verschiedene Formate für Zeitangaben. Das eine Format wird für Einstellungen wie der Session-Gültigkeitsdauer, das andere für wiederkehrende Termine und Erinnerungen verwendet.

3.3.1 Gültigkeitsdauer

Angaben im Format **<zahl>(mins|hrs|days|weeks|months|yrs|m|h|d|w|mo|y)** definieren sich durch eine Zahl gefolgt von einer Zeiteinheit und beschreiben Zeitlängen.
Die unterstützten Zeiteinheiten sind:

- **m** bzw. **mins**: Minuten - Beispiel: **15m** bzw. **15mins** für 15 Minuten gültig
- **h** bzw. **hrs**: Stunden - Beispiel: **2h** bzw. **2hrs** für 2 Stunden gültig
- **d** bzw. **days**: Tage - Beispiel: **1d** bzw. **1days** für 1 Tag gültig
- **w** bzw. **weeks**: Minuten - Beispiel: **15w** bzw. **15weeks** für 3 Wochen gültig
- **mo** bzw. **months**: Monate - Beispiel: **1mo** bzw. **1months** für 1 Monat gültig
- **y** bzw. **yrs**: Minuten - Beispiel: **15y** bzw. **15yrs** für 15 Jahre gültig

Dieses Format wird hauptsächlich für die Gültigkeitsdauer von Nutzersessions verwendet.

3.3.2 Zeitintervalle

Angaben im Format **<zahl>-(d|m|y)** oder **DD/MM** oder **DD/*** bzw. **DD/MM+** oder **DD/*+** beschreiben Zeitintervalle. Diese Angaben dienen der Berechnung eines künftigen oder vergangenen Datums, welches zum Erinnern oder als nächstes Datum eines Intervalls genutzt werden kann.

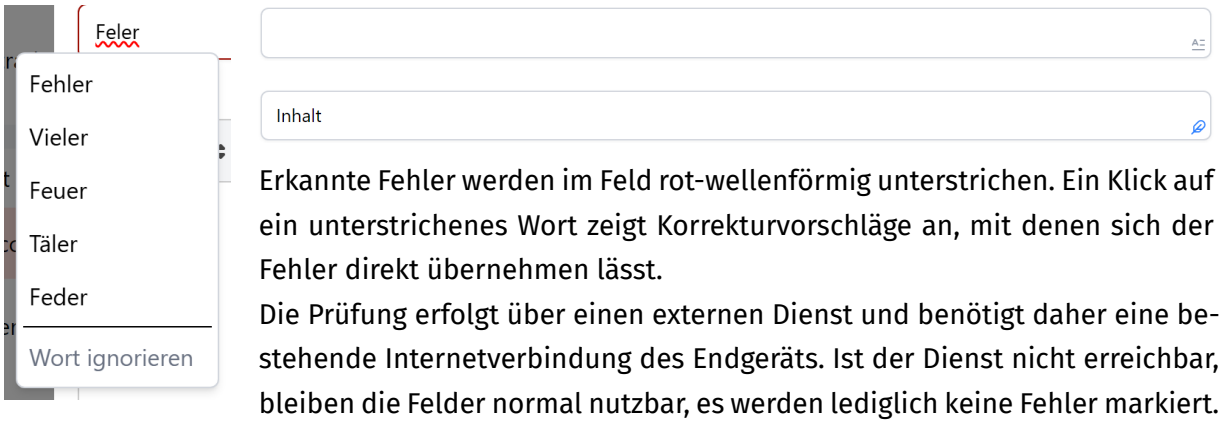
Format	**als Intervall**	**als Erinnerung**
<zahl>-(d|m|y)	Intervall für alle x Tage, alle x Monate oder alle x Jahre. z.B.: **7-d** für alle 7 Tage	Erinnerung für x Tage, x Monate oder x Jahre vor dem Fälligkeitsdatum.
DD/MM	Intervall für jedes Jahr an einem festen Tag. z.B.: **15/06** für den 15. Juni	Erinnerung an einem bestimmten Tag vor dem Fälligkeitsdatum.
DD/*	Intervall für bestimmten Tag jeden Monat. z.B.: **01/*** für den ersten Tag jeden Monat	Erinnerung an einem bestimmten Tag eines Monat vor dem Fälligkeitsdatum.

Format	als Intervall	als Erinnerung
DD/MM+ oder **DD/*+**	Erzwingt einen Mindestabstand zum Ausgangsdatum: **15/06+** liegt garantiert nicht im selben Jahr, **01/*+** nicht im selben Monat. Mit Zahl, z.B. **15/06+2**, für einen größeren Abstand (hier mind. 2 Jahre).	Erzwingt einen Mindestabstand zum Fälligkeitsdatum: **15/06+** liegt garantiert mindestens ein Jahr, **01/*** mindestens einen Monat vorher — auch wenn das Datum sonst nur wenige Tage vorher läge. Mit Zahl, z.B. **15/06+2**, für einen größeren Abstand (hier mind. 2 Jahre).

Dieses Format wird hauptsächlich für die Intervalle von Wartungen und Prüfungen und den Zeitpunkt der Erinnerung vor dem nächsten Check verwendet.

3.4 Rechtschreibprüfung

In vielen Textfeldern steht eine integrierte Rechtschreibprüfung zur Verfügung. Ein Symbol am Rand des Feldes zeigt an, ob die Prüfung für das Feld aktiv ist bzw. gerade läuft. Schreibgeschützte oder deaktivierte Felder werden nicht geprüft.



Erkannte Fehler werden im Feld rot-wellenförmig unterstrichen. Ein Klick auf ein unterstrichenes Wort zeigt Korrekturvorschläge an, mit denen sich der Fehler direkt übernehmen lässt.
Die Prüfung erfolgt über einen externen Dienst und benötigt daher eine bestehende Internetverbindung des Endgeräts. Ist der Dienst nicht erreichbar, bleiben die Felder normal nutzbar, es werden lediglich keine Fehler markiert.

3.5 Dateien und Bilder

Hochgeladene Dateien werden vom Server anhand ihres tatsächlichen Inhalts geprüft, nicht anhand der Dateiendung. Je nach Modul werden Bilder, PDF-Dokumente oder beides akzeptiert.

Bilder können als JPEG, PNG, WebP, GIF, TIFF oder AVIF hochgeladen werden. Sie werden einheitlich im WebP-Format gespeichert, entsprechend der Aufnahme-Ausrichtung gedreht und auf

maximal 2000 Pixel Kantenlänge verkleinert. Dadurch bleiben Speicherbedarf und Ladezeiten gering. HEIC-Bilder, wie sie z.B. von iPhones erstellt werden, werden nicht unterstützt und müssen vorher als JPEG oder PNG exportiert werden. Ausgenommen sind Logo und Icon in den Einstellungen, die weiterhin als PNG hochgeladen werden.

PDF-Dokumente werden unverändert gespeichert.

Dateien, die nicht öffentlich bereitgestellt werden, sind nur für angemeldete Nutzer mit entsprechender Berechtigung abrufbar. Bilder und PDFs werden direkt in der Anwendung angezeigt und können von dort heruntergeladen werden.

3.6 Engines

FF Operation stellt eine Reihe von systemweiten Funktionen zur Verfügung, die vom Benutzer konfiguriert und in verschiedenen Modulen verwendet werden können.

3.6.1 Notification-Engine

Die Notification-Engine ermöglicht es dem Benutzer, Benachrichtigungen für verschiedene Ereignisse in der Anwendung zu konfigurieren. Dies umfasst sowohl systemweite Vorgänge als auch Erinnerungen an Fristen und Fälligkeiten.
Benachrichtigungen können entweder direkt in der Anwendung angezeigt oder per E-Mail erhalten werden. Der Benutzer kann dabei selbst festlegen, welche Ereignisse auf welchem Weg mitgeteilt werden sollen.
Dabei kann ein Nutzer nur Benachrichtigungen für Module konfigurieren, für die er über entsprechende Berechtigungen verfügt. Werden einem Nutzer nachträglich Berechtigungen entzogen, prüft das System beim Versand einer Benachrichtigung automatisch, ob der Zugriff auf das betreffende Modul weiterhin besteht.

Eine detaillierte Anleitung zur Verwendung finden Sie unter **Benutzerbereich und Accounteigenschaften -> Benachrichtigungen** (Abschnitt 5.2).

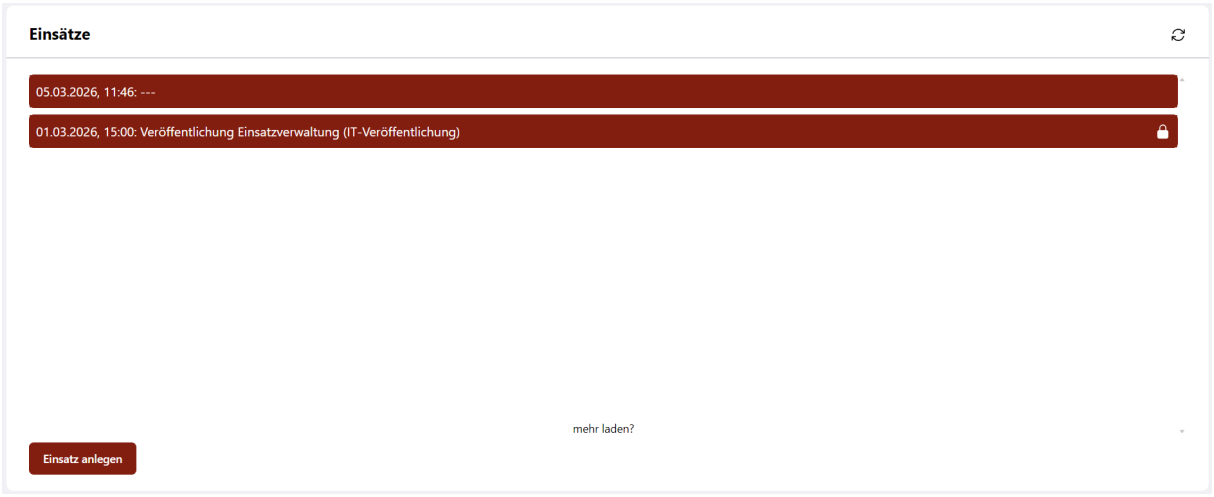
4 Module

FF Operation nutzt verschiedenste Module, um Daten zu organisieren, die Verwaltung zu vereinfachen und Berechtigungen sinnvoll setzen zu können.
Diese Module ermöglichen die Anpassung der Anwendung an die eigenen Bedürfnisse, sodass die Daten der Feuerwehr oder des Vereins optimal abgebildet werden können.



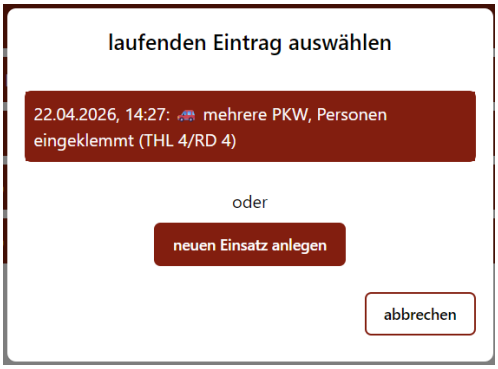
Die Module gliedern sich in drei Teilbereiche, die sich an der Navigationsleiste orientieren:
- Zum einen die Verwaltung der Einsatze mit dem Formular zur Dateneingabe.
- Die übrigen Teilbereiche dienen der Verwaltung von Basisdaten wie Gerätschaften, Fahrzeuge oder Kleidung sowie der Konfiguration und Steuerung der Anwendung selbst.

4.1 Einsatzdaten und Dokumentation



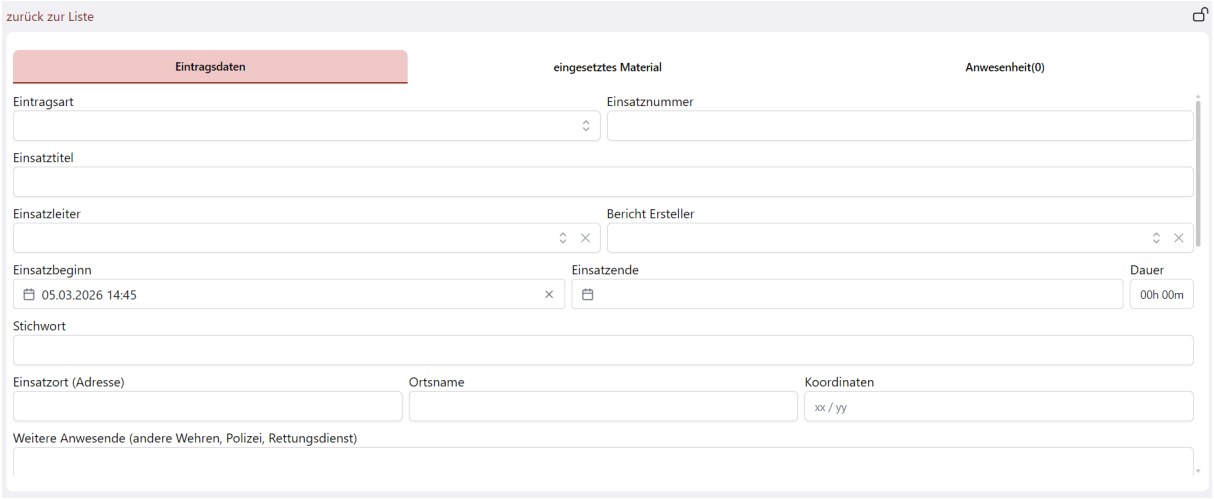
Das Modul Einsätze zeigt alle aktuell laufenden und abgeschlossenen Einsätze an. Neue angelegte Einsätze werden automatisch an alle aktiven Geräte übermittelt und die Liste aktualisiert.
Ein Reload-Button oben rechts ermöglicht das manuelle Neuladen.
Ein Schloss-Symbol kennzeichnet schreibgeschützte Einträge - diese können geöffnet, aber nicht bearbeitet werden.
Die Liste nutzt infinite Loading: Beim Scrollen nach unten werden weitere Einträge geladen. Über Berechtigungseinstellungen lässt sich die Anzahl der angezeigten Einträge und die Tiefe der Historie begrenzen.

Die Daten zu den Einträgen sind auf die Tabs Eintragsdaten, eingesetztes Material, Anwesenheit, Protokoll und Karte aufgeteilt. Der gewählte Tab wird in der URL gespeichert, sodass ein Link oder ein Neuladen der Seite wieder den gleichen Tab öffnet. Alle Daten der Einträge sind kollaborativ verfügbar. Daten werden auch nachsynchronisiert, wenn ein Gerät zwischenzeitlich die Internetverbindung verliert.
Wer Einträge bearbeiten darf, dessen Stand speichert die Anwendung dafür zusätzlich lokal im Browser. Änderungen ohne Verbindung überstehen so auch ein Neuladen der Seite, das Verlassen des Eintrags oder den Standby eines Geräts. Der lokale Stand gilt 12 Stunden, gehört zum angemeldeten Konto und wird beim Abmelden gelöscht. Passt er nicht mehr zum Stand des Servers, etwa nach dem Einspielen eines Backups, verwirft die Anwendung ihn und lädt den Eintrag neu. Noch nicht übertragene Änderungen gehen dabei verloren, ein Hinweis informiert darüber.
Soll ein neuer Eintrag erstellt werden, kann dies automatisch über die Incident-API (siehe Abschnitt 4.2) passieren oder manuell über den Button in der Listenansicht. Werden beim manuellen Erstellen nichtabgeschlossene Einträge gefunden, so wird ein Dialog angezeigt, um auf laufende Einträge hinzuweisen und so Doppelanlagen zu verhindern.



4.1.1 Eintragsdaten

Wird ein Einsatz neu angelegt, beinhaltet dieser bis auf das Startdatum keine weiteren Daten. Anhand des gesetzten Enddatums werden alle verfügbaren Kräfte, Geräte etc. ermittelt.



Das Datenformular beinhaltet Felder zur Eingabe von typischerweise benötigten Feldern. Sind am Feld hinten Pfeile, so wird von diesen DropDowns auf konfigurierten Basisdaten zugegriffen. Für

die Auswahl in diesen DropDowns genügt die Leseberechtigung für Einsätze. Nutzer benötigen keine Berechtigungen für die Module Kräfte, Fahrzeuge, Geräte, Kleidung oder Eintragsarten, um diese im Einsatz auswählen zu können. Dabei werden nur die zur Auswahl nötigen Daten (z.B. Name oder Kennung) übermittelt. Die Eingabe zum Datum kann durch tippen oder über das angezeigte Dialogfeld erfolgen. Auf mobilen Geräten wird die Eingabe für das Datum aus Usabilitygründen zentral angezeigt.

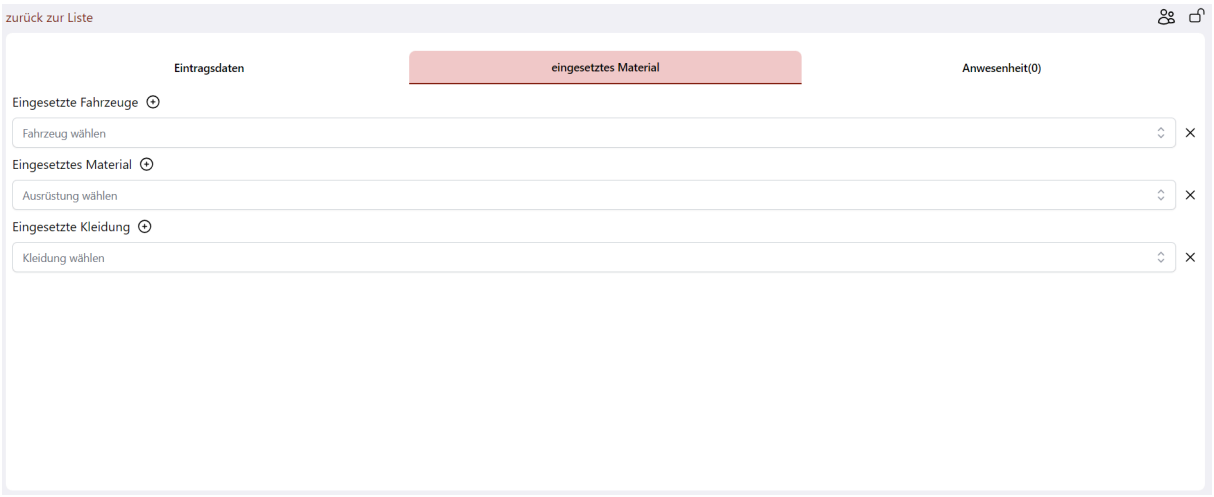


Das Feld Einsatznummer, welches zur Identifikation über die Einträge hinweg verwendet wird, zeigt die Einzigartigkeit an. Die „Einzigartigkeit“ ist eine visuelle Indikation und verhindert nicht das Speichern eines Eintrags.



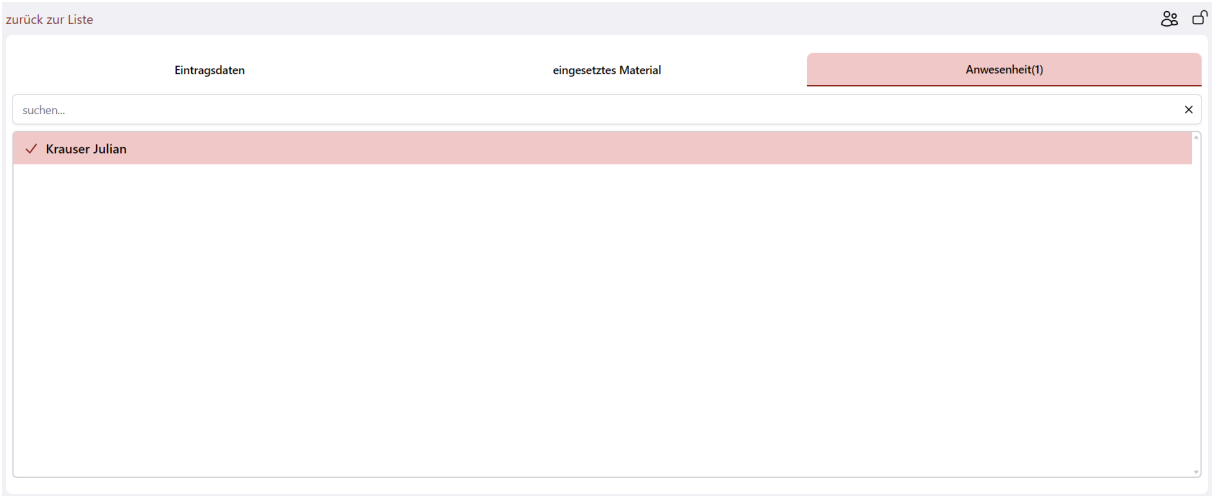
Das Formular hat außerdem ein Freitexteditor, welcher Daten zum Eintrag erfassen kann. Auch gibt es einen separaten Abschnitt zur Erfassung von Kontaktpersonen mit deren Beteiligung und Daten.

4.1.2 eingesetztes Material



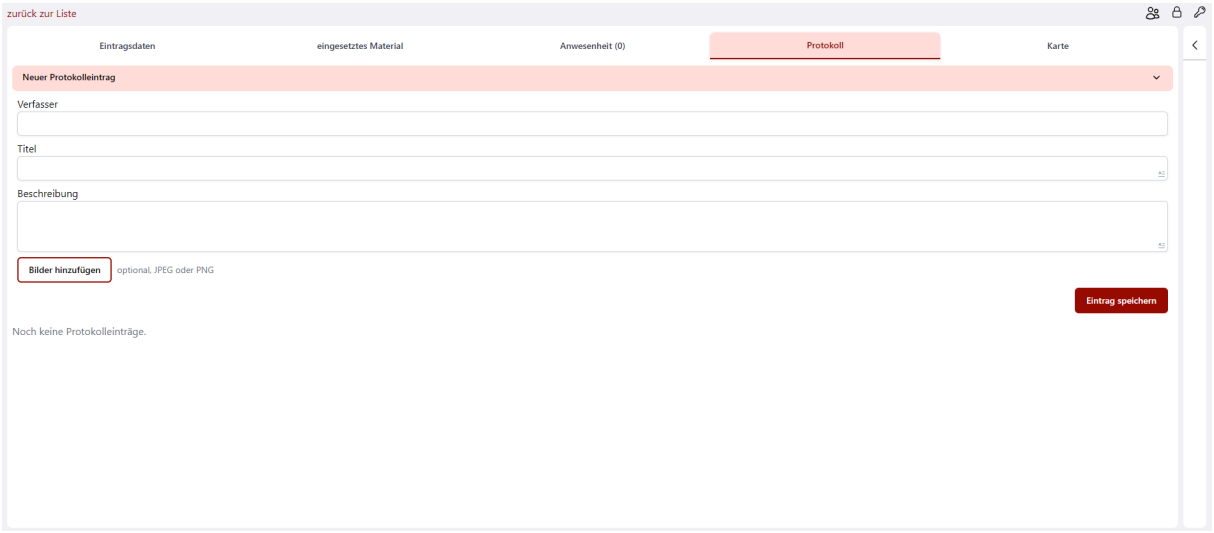
Der Tab „eingesetztes Material“ erfasst Daten zu den verwendeten Gerätschaften, Fahrzeugen und Kleidung. Je nach Eintrag werden unterschiedliche Daten abgefragt. So zum Beispiel, den Fahrer oder von wem etwas verwendet wurde. Alle Einträge können mit Notizen versehen werden.

4.1.3 Anwesenheit



Im Tab „Anwesenheit“ kann die Anwesenheit der verfügbaren Kräfte erfasst werden. Über die Suche kann zusätzlich die Übersicht gefiltert werden. Wird eine Kraft irgendwo im Formular ausgewählt, wird diese automatisch auch in der Anwesenheit angehackt.

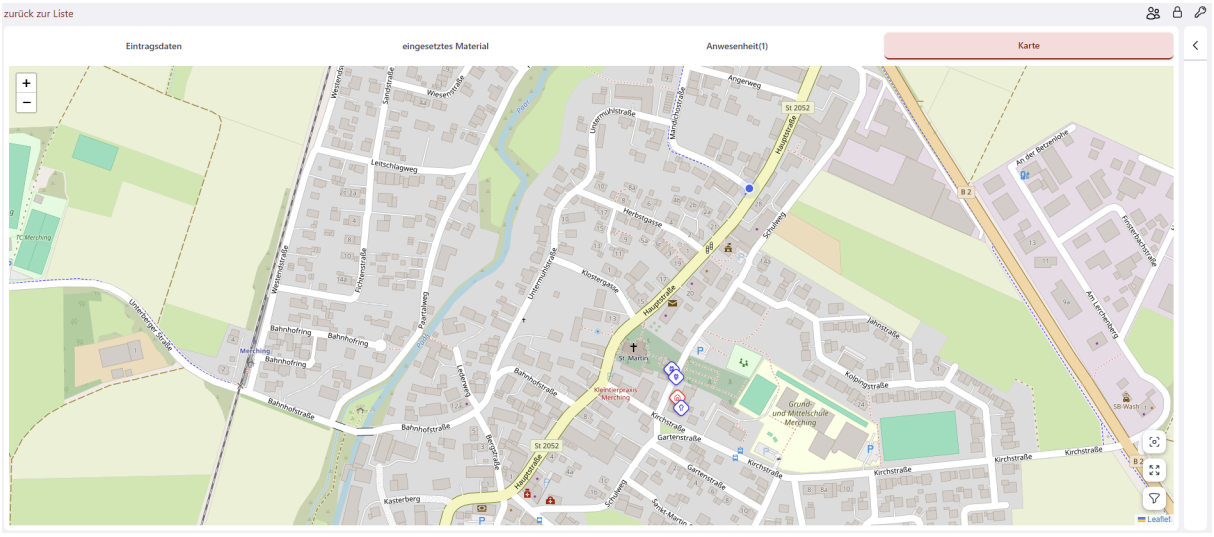
4.1.4 Protokoll



Im Tab „Protokoll“ werden während eines Einsatzes Entscheidungen, Lagemeldungen oder sonstige Ereignisse festgehalten. Im Gegensatz zu den übrigen Eintragsdaten sind Protokolleinträge unveränderbar: Ein gespeicherter Eintrag kann weder bearbeitet noch gelöscht werden. Die Einträge werden chronologisch mit Zeitpunkt der Erstellung und Verfasser angezeigt. Neue Einträge erscheinen ohne Neuladen bei allen Nutzern, die den Einsatz geöffnet haben.

Über den Bereich **Neuer Protokolleintrag** wird ein Eintrag erstellt. Verfasser, Titel und Beschreibung sind Pflichtfelder. Der Verfasser bleibt nach dem Speichern erhalten, sodass mehrere Einträge nacheinander erfasst werden können. Das Formular wird nur Nutzern mit Bearbeitungsrechten für Einsätze angezeigt. Nutzer mit Leserechten können das Protokoll einsehen. Auch bei einem gesperrten Eintrag (siehe Abschnitt 4.1.6) können weiterhin Protokolleinträge ergänzt werden.
Zu jedem Protokolleintrag können optional bis zu 10 Bilder mit jeweils maximal 20 MB angehängt werden. Unterstützt werden JPEG und PNG, HEIC-Bilder müssen vorher umgewandelt werden. Sind in den Bildern Aufnahmezeitpunkt und GPS-Koordinaten hinterlegt, werden diese ausgelesen und am Bild angezeigt. Ein Klick auf ein Bild öffnet dieses in der Anwendung, von dort kann es auch heruntergeladen werden. Während des Hochladens wird der Fortschritt angezeigt.
Protokolleinträge können nur mit bestehender Verbindung zum Server gespeichert werden. Schlägt das Speichern fehl, bleiben die Eingaben im Formular erhalten und können erneut gesendet werden.
Hinweis zur Datensicherung: Die Bilder des Protokolls werden verschlüsselt im Ordner `files/protocol` (im Docker-Container `/app/files/protocol`) abgelegt. Der Schlüssel wird aus den Umgebungsvariablen `APPLICATION_SECRET` und `APPLICATION_SALT` abgeleitet. Die Bilder sind nicht Teil des Datenbank-Backups und müssen gemeinsam mit dem Volume gesichert werden. Gehen `APPLICATION_SECRET` oder `APPLICATION_SALT` verloren oder werden diese ohne den Rotationsmodus (Abschnitt 2.7) geändert, sind die gespeicherten Bilder nicht mehr lesbar. Auch der zwischengespeicherte Bearbeitungsstand offener Einsätze wird verschlüsselt in der Datenbank abgelegt. Er ist nicht Teil eines Backups: Überschreibt eine Wiederherstellung die Einträge, wird er verworfen und alle geöffneten Einträge werden vom Server neu geladen.

4.1.5 Karte



Im Tab „Karte” wird eine OpenStreetMap-Karte mit den Koordinaten des Eintrags angezeigt. Zusätzlich werden die eigene Position (falls vorhanden) und die erstellten Marker dargestellt. Informationen zu den Markern, Typen und der Karte finden Sie in Abschnitt 4.8.

4.1.6 Sperren, Freischalten und Löschen



Die Icons im rechten oberen Eck ermöglichen das Sperren, Freischalten oder Löschen eines Eintrags. Diese Icons werden abhängig von denBerechtigungen und dem aktuellen Status des Eintrags angezeigt.

Schloss offen: Der aktuell geöffnete Eintrag kann für die Bearbeitung gesperrt werden. Für alle Nutzer werden die Felder auf Readonly umgestellt. Jegliche Änderungen werden vom Server blockiert.

Schloss geschlossen: Der aktuelle Eintrag ist im Modus Readonly. Dies ist der Fall, wenn ein Nutzer keine Berechtigungen zum Bearbeiten hat oder der Eintrag selbst Readonly ist.

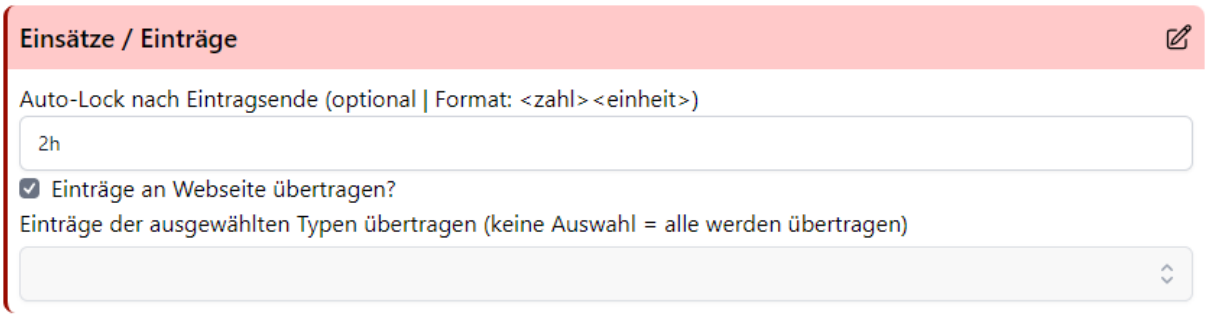
Schlüssel: Über den Schlüssel kann ein Eintrag wieder zum Bearbeiten freigeschalten werden. Dazu muss ein Nutzer Admin-Rechte haben oder zusätzlich zu den Bearbeitungsrechten für Einsätze über die weitere Berechtigung „erlaubt Einträge wieder zu entsperren„ verfügen (siehe Abschnitt 4.14). Ist in den Moduleinstellungen (Abschnitt 4.10) ein **Manueller Unlock nach Eintragsende** hinterlegt, verschwindet der Schlüssel für alle Nutzer - auch für Admins - sobald diese Frist nach dem Eintragsende abgelaufen ist.

Ergänzung hinzufügen: Ist ein Eintrag gesperrt, können Nutzer mit Bearbeitungsrechten für Einsätze unterhalb der Einsatzbeschreibung dennoch einen ergänzenden Text nachtragen, ohne den Eintrag entsperren zu müssen. Der Text wird mit Zeitstempel und Namen des Nutzers automatisch an die bestehende Einsatzbeschreibung angehängt und ist danach nicht mehr entfernbar.

Mülltonne: Der aktuelle Eintrag kann über die Mülltonne gelöscht werden. Das Icon wird nur angezeigt, wenn der Nutzer die Berechtigung zum Löschen hat und der Eintragsbeginn geleert ist. Da ein Eintrag standardmäßig mit Beginn erstellt wird, wird das Icon standardmäßig nicht angezeigt.

4.1.7 Synchronisierung mit FF Webpage

In den Modulspezifischen Einstellungen (Abschnitt 4.10) kann ein automatischer sync mit FF Webpage eingerichtet werden.



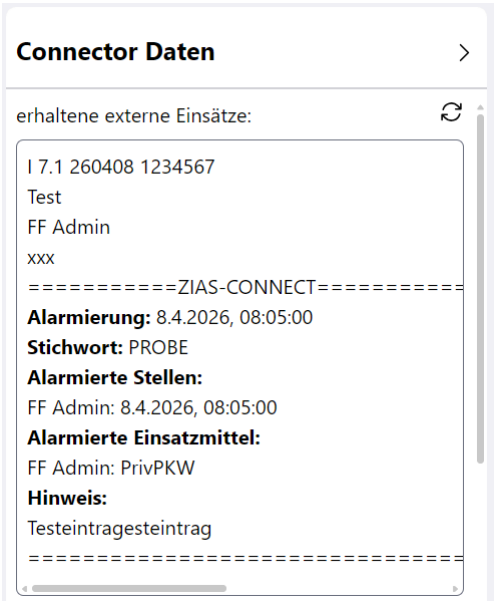
Für die Synchronisierung muss zusätzlich die Moduleinstellung `Webpage WebAPI Connect` eingerichtet sein. Wird die Synchronisierung für die Einträge aktiviert, kann zusätzlich noch ausgewählt werden, ob alle angelegten oder nur ausgewählte Eintragsarten synchronisiert werden sollen.

4.2 Incident-Management

Das Incident Management ermöglicht die Verarbeitung von Einsatzdaten aus externen Systemen. Diese Daten werden über eine generische Schnittstelle an FF Operation übergeben.
Für die Anbindung können Connectoren verwendet oder erstellt werden, die eingehende Einsatzdaten an FF Operation weiterleiten. Ein Beispiel ist das GO-CLI-Tool ff-operation-zias-connect, das auf einem Raspberry Pi laufen und neue JSON- oder PDF-Dateien an FF Operation weiterleiten kann.

Wichtig: Aktuell werden nur Einsatzdaten verarbeitet. Eine Erweiterung zur Verarbeitung bzw. Erstellung von Einsatz-PDFs ist in Arbeit.

4.2.1 Einsatzdaten von außen



Wenn die Berechtigung `Incident` für einen Nutzer freigeschaltet ist, wird in den Eintragsdetails eine zusätzliche Seitenleiste angezeigt. Dort sind die empfangenen Incident-Daten sichtbar und können zum Ausfüllen des Formulars verwendet werden.

Wenn im Formular noch keine externe ID eingetragen ist, werden alle übergebenen Incidents ohne Zuordnung angezeigt. Ist eine externe ID eingetragen, werden nur Incidents mit dieser externen ID angezeigt.

Die Incident-Daten werden in folgender Reihenfolge verarbeitet:

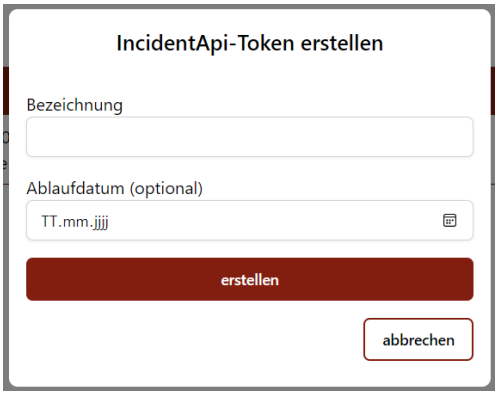
1. übergebene Incidents werden in der Datenbank abgelegt und dort maximal 5 Tage gespeichert.
2. Wenn in den Moduleinstellungen ein Regex hinterlegt ist, wird die übergebene Incident-ID gegen diesen Regex geprüft. Passt die Incident-ID nicht, wird der Incident abgelehnt.
3. Wenn kein offener Eintrag ohne externe ID vorhanden ist, erstellt das System einen neuen Eintrag automatisch und übernimmt die übergebenen Daten. Einzelne Felder werden dabei in das interne Format umgewandelt.
4. Die Startzeit wird aus `incidentStart` übernommen. Falls keine gültige Startzeit vorliegt, wird die aktuelle Uhrzeit verwendet. Ist in den Moduleinstellungen ein Name definiert, wird zusätzlich

in den informierten Stellen nach einer passenden Quelle gesucht und die dortige Startzeit bevorzugt.

5. Wenn eine Standard-Eintragsart konfiguriert ist, wird der neue Eintrag mit dieser Eintragsart angelegt. Der Abgleich mit den offenen Einträgen und deren externe ID wird dann auf diese Eintragsart beschränkt.

4.2.2 Einsatzdaten-Schnittstelle





IncidentApi-Tokens ermöglichen externen Systemen die Kommunikation mit der Incident-Schnittstelle der Anwendung. Die Schnittstelle ist nur für das Senden von Daten vorgesehen, nicht für das Abfragen.
Das Recht auf das Modul IncidentApi ist privilegiert: Da die Tokens von außen Einträge anlegen können, vergeben und entziehen es nur Administratoren. Das gilt auch für Rollen, die dieses Recht enthalten.



Unter dem Fingerabdruck-Symbol kann das Token der IncidentApi abgerufen und bei Bedarf auch erneuert werden.
Diese Tokens sind JWTs und können ohne Kenntnis der Serverkonfiguration nicht gefälscht werden. Nach einem Wechsel des Secrets (Abschnitt 2.7) müssen sie neu erzeugt und in den Connectoren eingetragen werden.
Zur schnelleren Konfiguration eines Connectors zeigt der Dialog mit dem Token zusätzlich die Domain des Server der Anwendung an.

4.2.3 Verwendung der Schnittstelle

Externe Daten werden per PUT-Request an die FF-Operation-API übergeben. Wenn der Key `incidentId` fehlt oder nicht zum konfigurierten Regex passt, wird der Request abgelehnt.

Authentifizierung

Die Authentifizierung erfolgt über ein IncidentApi-Token im Header:

```
1 Authorization: Bearer <incident-api-token>
2 Content-Type: application/json
```

Beispiel

```
PUT <https|http>://<operation-server-adress>/api/incident/generic

Authorization: Bearer <incident-api-token>
Content-Type: application/json

{
  incidentId: string;
  incidentTitle?: string;
  incidentStart?: string; // ISO-8601, z. B. 2026-04-08T12:34:56Z
  incidentCategory?: string;
  incidentLocation?: string;
  incidentLocationPin?: string;
  incidentLatitude?: string;
  incidentLongitude?: string;
  incidentInformed?: Array<{
    department?: string;
    informedAt?: string; // ISO-8601
    vehicles?: string[];
  }>;
  incidentDescription?: string;
}
```

Feldregeln

- `incidentId` ist ein Pflichtfeld.
- `incidentStart` sollte im ISO-8601-Format mit Zeitzone übergeben werden. Das Feld ist optional.
- Alle weiteren Felder sind optional, werden aber für eine bessere Vorbelegung des Eintrags empfohlen.
- Der Server prüft die Anfrage bewusst nachsichtig, damit bestehende Einsatzleitsysteme weiter senden können: Zahlen werden auch als Text angenommen, `null` gilt wie ein fehlendes Feld.

Für eigene Anbindungen enthält das Paket `@ff-admin/ff-operation-api` (Abschnitt 4.15.4) zusätzlich den `IncidentApiClient`. Er nutzt das IncidentApi-Token direkt und bietet nur die Routen der Incident-Schnittstelle an:

```typescript
import { IncidentApiClient } from "@ff-admin/ff-operation-api";

const incident = new IncidentApiClient({
  server: "https://einsatz.ff-beispiel.de",
  incidentApiToken: "<incident-api-token>",
});
await incident.api.receiveIncident({ body: { incidentId: "E-2026-0815", incidentTitle: "Brand Wohngebäude" } });
```

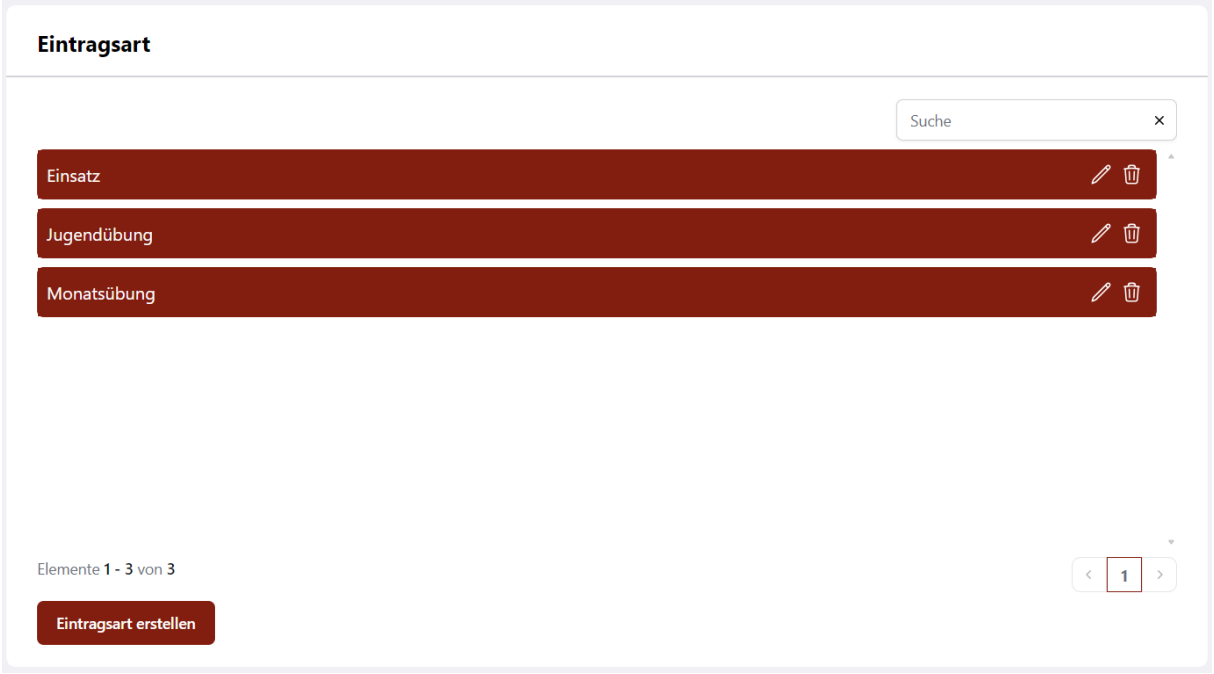
4.2.4 Einstellungen zur Verwendung

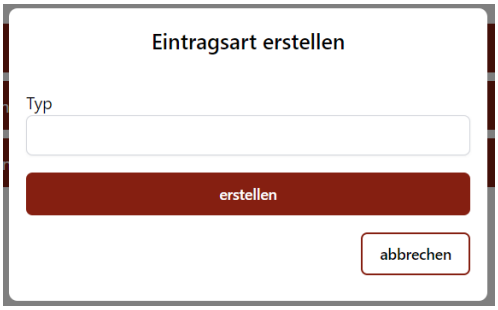
In den modulspezifischen Einstellungen (Abschnitt 4.10) wird die Verwendung der externen Incident-Schnittstelle konfiguriert.



Hier wird die oben beschriebene Verarbeitungslogik bei der Übergabe von Incidents eingestellt.

4.3 Eintragsart



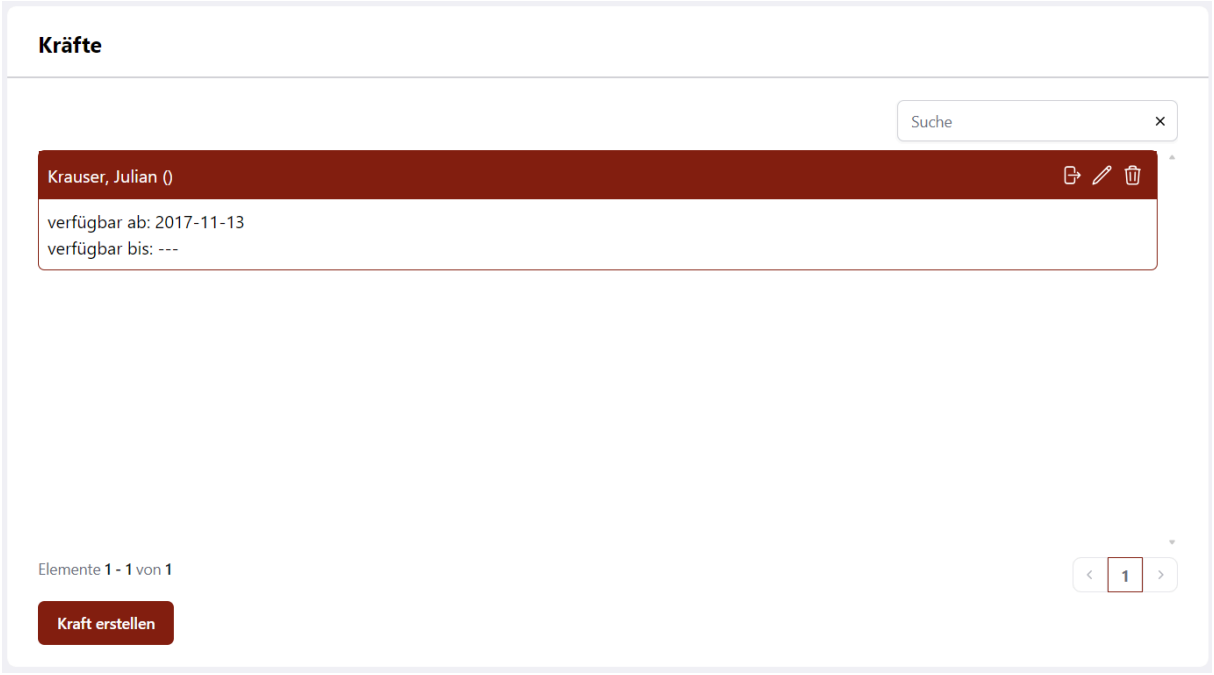


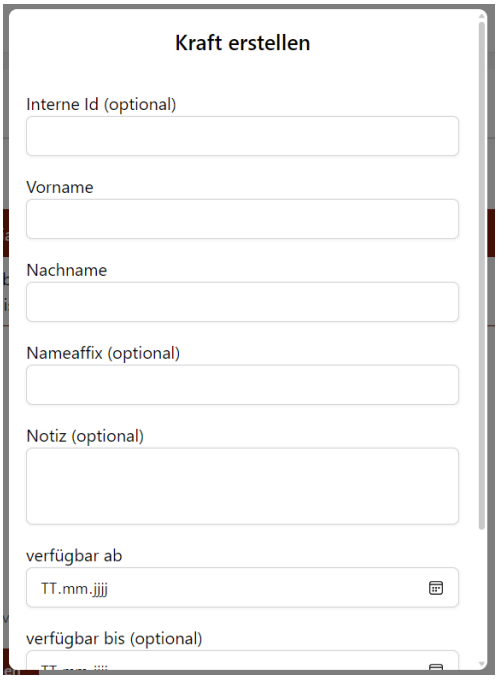
Eintragsarten kategorisieren die Einträge. So kann hierüber zwischen z.B. Einsätzen, Übungen und co unterschieden werden.
Eintragsarten bestehen nur aus einer Bezeichnung.

Eintragsarten werden aber an zwei weiteren Stellen aktiv verwendet:

- Bei der **Synchronisierung mit FF Webpage** (Abschnitt 4.1.7) kann die Übertragung auf ausgewählte Eintragsarten eingeschränkt werden. Ohne Auswahl werden Einträge aller Eintragsarten übertragen.
- In den Einstellungen der Einsatzdaten-Schnittstelle (Abschnitt 4.2) kann eine Eintragsart als **Standard-Eintragsart** hinterlegt werden. Neue Einträge, die automatisiert über die Schnittstelle angelegt werden, erhalten dann diese Eintragsart, und der Abgleich mit bereits offenen Einträgen wird auf diese Eintragsart beschränkt.

4.4 Kräfte





Kräfte, auch Mitglieder der Wehr, können für die Anwesenheit oder belegten Positionen ausgewählt werden. Kräfte besitzen Basisinformationen und einen Verfügbarkeitszeitraum.

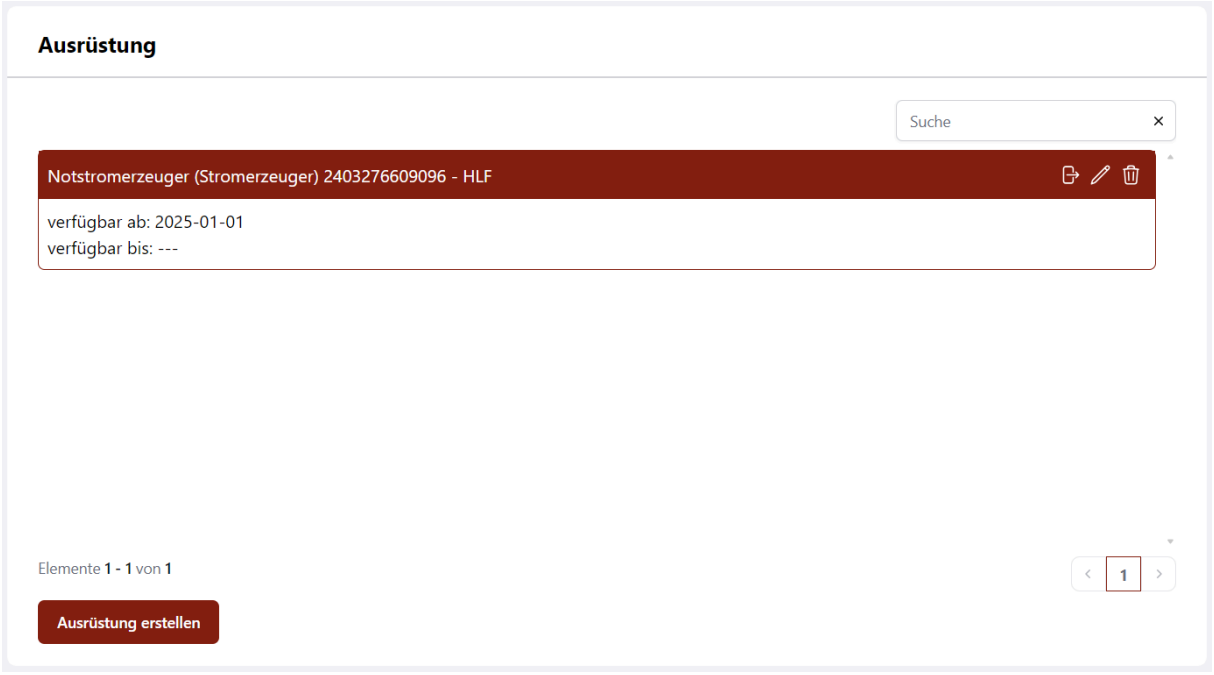
Kräfte können entweder automatisiert aus FF Admin synchronisiert oder manuell direkt in FF Operation angelegt werden.

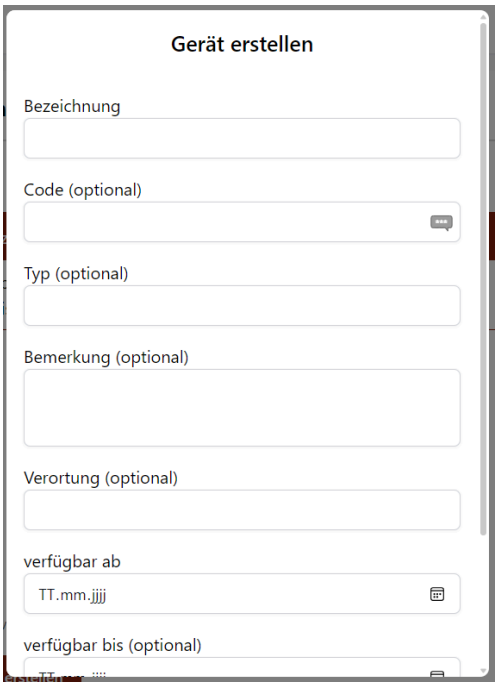
Für die Synchronisierung wird in den Moduleinstellungen (Abschnitt 4.10) eine gespeicherte Query aus dem FF-Admin Query-Builder sowie das Feld ausgewählt, das die Mitglieds-ID im Query-Ergebnis liefert. Beim Sync wird diese Query in FF Admin ausgeführt: Alle Mitglieder, deren ID im Ergebnis enthalten ist, werden als Kraft übernommen. Ein Mitglied gilt dabei so lange als verfügbar, wie seine ID im Ergebnis der Query auftaucht - fällt ein Mitglied beim nächsten Sync aus dem Ergebnis heraus, gilt es ab diesem Zeitpunkt nicht mehr als verfügbar. Der Beginn der Verfügbarkeit wird aus dem Beginn der aktuellen Mitgliedschaft in FF Admin übernommen.

Werden Kräfte manuell erfasst, können diese über das Symbol (siehe rechts) auf verfügbar bis heute gesetzt.



4.5 Ausrüstung



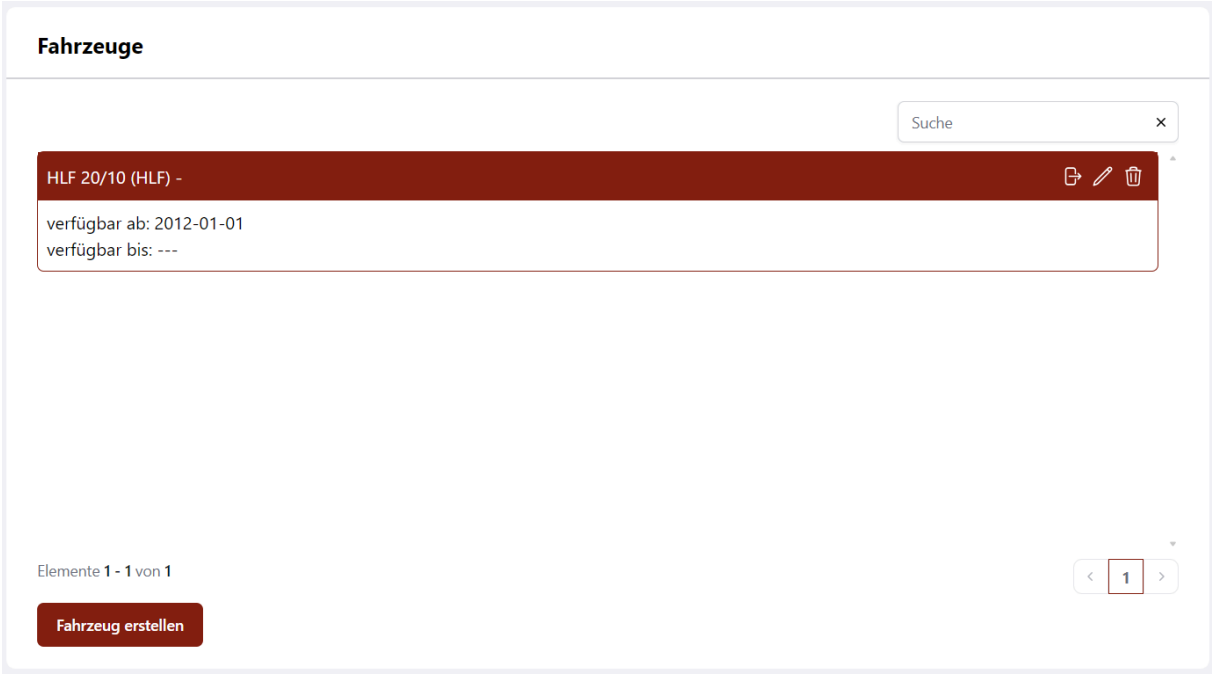


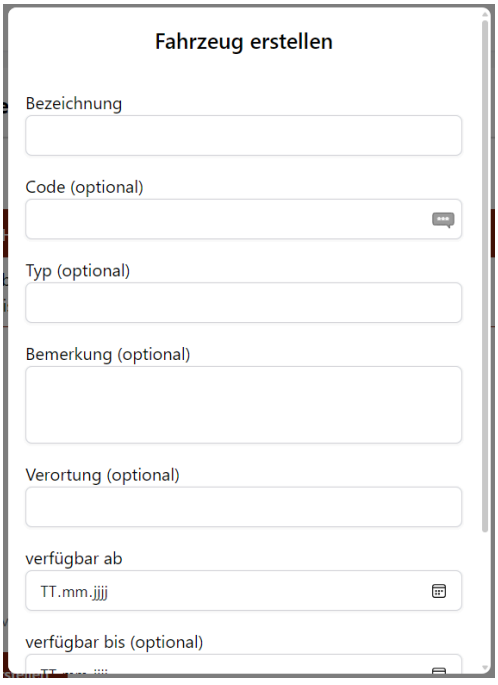
Ausrüstung kann bei eingesetztes Material ausgewählt werden. Ausrüstung besitzt Basisinformationen und einen Verfügbarkeitszeitraum.
Ausrüstung kann entweder automatisiert aus FF Admin synchronisiert oder manuell direkt in FF Operation angelegt werden.
Anders als bei den Kräften erfolgt die Synchronisierung hier ohne Query: Es wird immer die gesamte in FF Admin erfasste Ausrüstung geladen. Ob ein Gerät aktuell verfügbar ist, wird dabei anhand des Indienststellungs- und eines optionalen Außerdienststellungsdatums aus FF Admin bestimmt - nur Geräte, deren Außerdienststellungsdatum leer ist oder noch nicht erreicht wurde, werden übernommen. Wird ein Gerät in FF Admin außer Dienst gestellt, entfällt es entsprechend beim nächsten Sync.

Wird Ausrüstung manuell erfasst, können diese über das Symbol (siehe rechts) auf verfügbar bis heute gesetzt.



4.6 Fahrzeuge



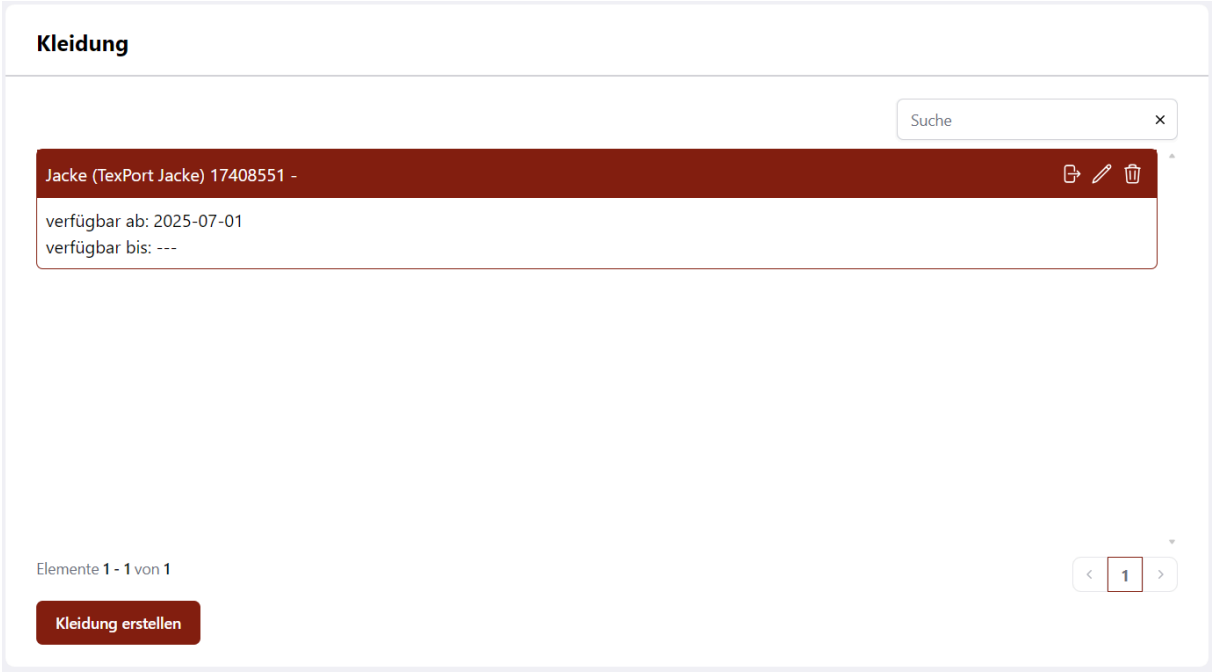


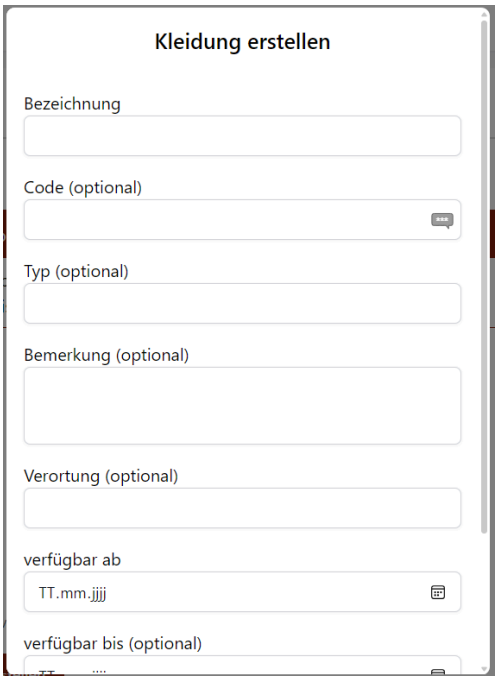
Fahrzeuge können bei eingesetztes Material ausgewählt werden. Ein Fahrzeug besitzt Basisinformationen und einen Verfügbarkeitszeitraum.
Fahrzeuge können entweder automatisiert aus FF Admin synchronisiert oder manuell direkt in FF Operation angelegt werden.
Anders als bei den Kräften erfolgt die Synchronisierung hier ohne Query: Es werden immer alle in FF Admin erfassten Fahrzeuge geladen. Ob ein Fahrzeug aktuell verfügbar ist, wird dabei anhand des Indienststellungs- und eines optionalen Außerdienststellungsdatums aus FF Admin bestimmt - nur Fahrzeuge, deren Außerdienststellungsdatum leer ist oder noch nicht erreicht wurde, werden übernommen. Wird ein Fahrzeug in FF Admin außer Dienst gestellt, entfällt es entsprechend beim nächsten Sync.

Wird ein Fahrzeug manuell erfasst, können diese über das Symbol (siehe rechts) auf verfügbar bis heute gesetzt.



4.7 Kleidung





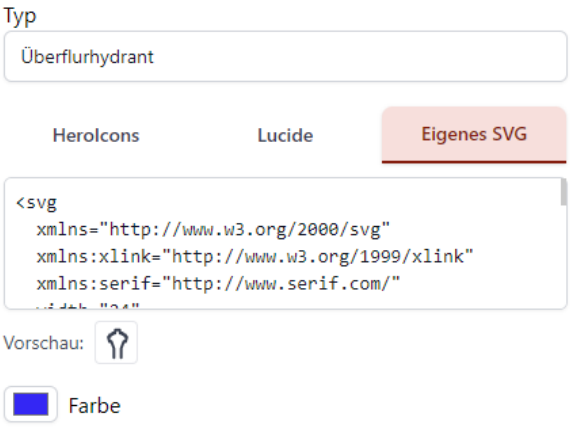
Kleidung kann bei eingesetztes Material ausgewählt werden. Kleidung besitzt Basisinformationen und einen Verfügbarkeitszeitraum.
Kleidung kann entweder automatisiert aus FF Admin synchronisiert oder manuell direkt in FF Operation angelegt werden.
Anders als bei den Kräften erfolgt die Synchronisierung hier ohne Query: Es wird immer die gesamte in FF Admin erfasste Kleidung geladen. Ob ein Kleidungsstück aktuell verfügbar ist, wird dabei anhand des Indienststellungs- und eines optionalen Außerdienststellungsdatums aus FF Admin bestimmt - nur Kleidung, deren Außerdienststellungsdatum leer ist oder noch nicht erreicht wurde, wird übernommen. Wird ein Kleidungsstück in FF Admin außer Dienst gestellt, entfällt es entsprechend beim nächsten Sync.
Wird Kleidung manuell erfasst, können diese über das Symbol (siehe rechts) auf verfügbar bis heute gesetzt.

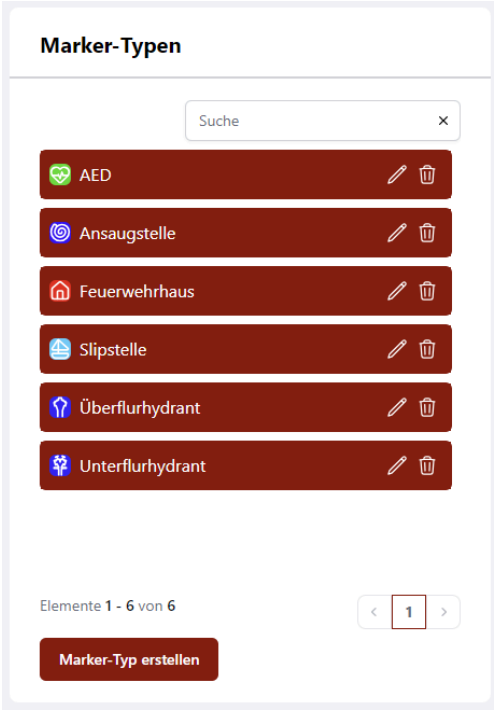


4.8 Marker und Markertypen

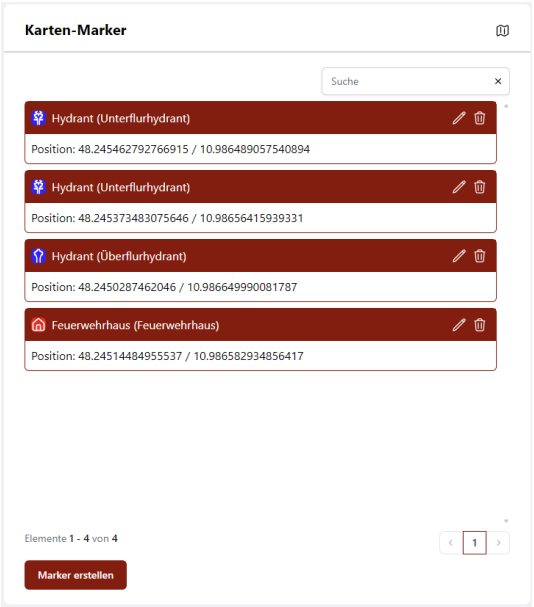
Die Marker ermöglichen eine digitale Repräsentation von Orten oder Punkten auf einer Karte, welche in Einsätzen interessant sein könnten. Speziell wurden die Marker für Hydraten, Ansaugstellen oder Defibrillatoren entwickelt.

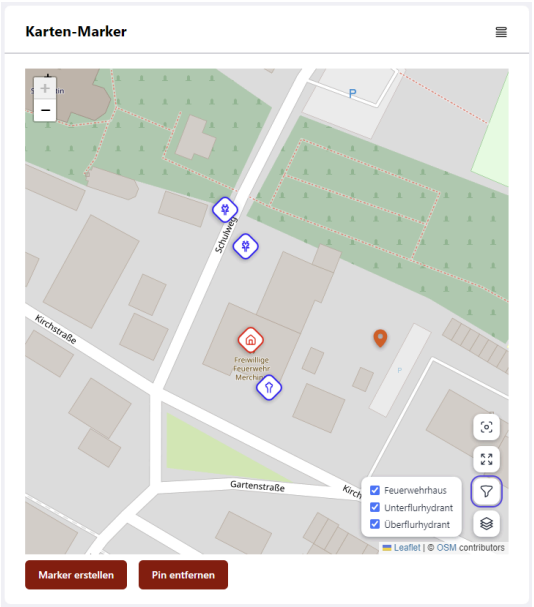
Es können eigene Markertypen mit bestehenden oder eigenen Icons und Farben angelegt werden. In der Karte werden die Marker dann mit dem gewählten Icon als Raute dargestellt.





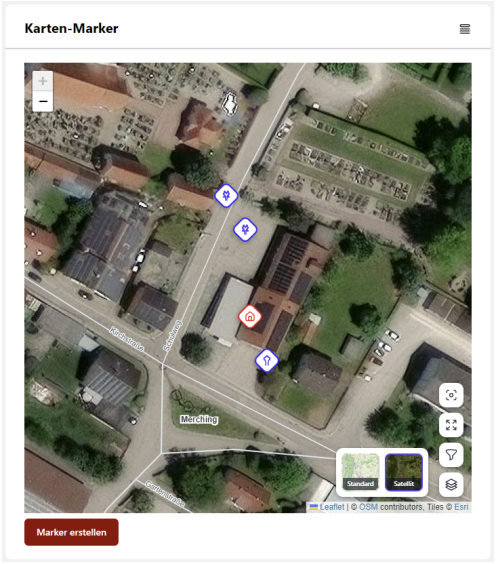
Die Marker selbst können als Liste oder als Karte angezeigt werden.





Die Karte ermöglicht zudem die räumliche Darstellung der gespeicherten Marker. Durch Klicken auf die Icons können darüber hinaus weitere Details angezeigt werden. Im Detail-Popup eines Markers befinden sich neben dem Navigationslink zusätzlich Icons zum Bearbeiten und Löschen des Markers. Diese werden nur angezeigt, wenn der Nutzer die entsprechende Berechtigung für Karten-Marker besitzt. So können Marker direkt aus der Karte heraus angepasst werden, ohne in die Listenansicht zu wechseln. Durch Klicken auf die Karte wird ein Pin gesetzt, an dem ein Marker erstellt werden kann. Es gibt auch einen Filter, um die einzelnen Typen anzuzeigen, sowie Zoom-Buttons, um zu einer Übersicht zu zoomen oder zur eigenen Position zu navigieren.

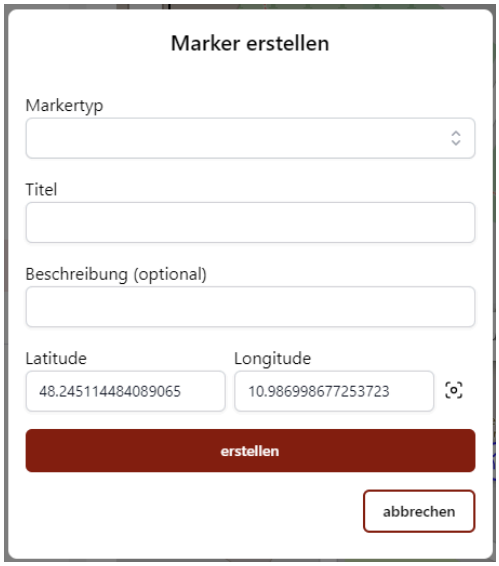
4.8.1 Kartentyp und Offline-Nutzung



Über das Kartentyp-Icon in der Werkzeugleiste kann zwischen der Standardkarte (OpenStreetMap), einer Satellitenansicht (ESRI) und einer Bahnstrecken-Ansicht (OpenRailwayMap) gewechselt werden. In der Satellitenansicht werden zusätzlich Straßen und Ortsnamen eingeblendet, da Luftbilder allein zur Orientierung oft nicht ausreichen. Die Bahnstrecken-Ansicht blendet zusätzlich zur Standardkarte Gleisanlagen ein und eignet sich z.B. für Einsätze im Bereich von Bahnstrecken.

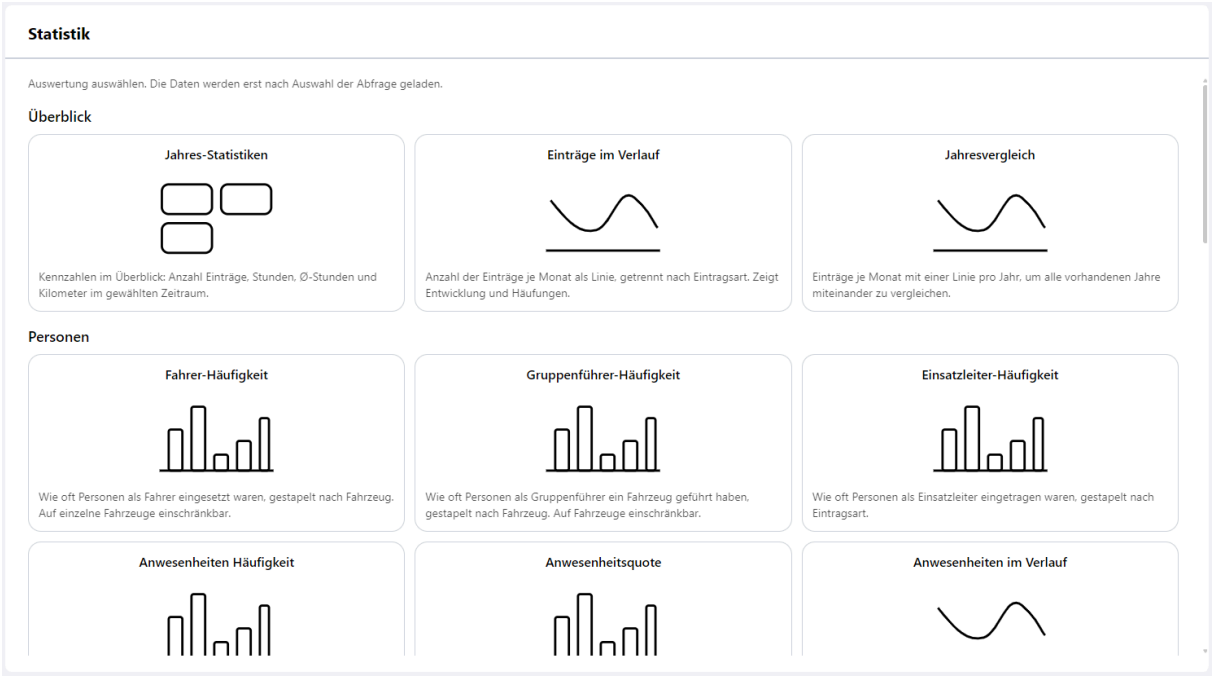
Einmal angezeigte Kartenausschnitte – sowohl die Standardkarte als auch die Straßen-Overlay der Satellitenansicht – werden automatisch auf dem Gerät zwischengespeichert und bleiben dadurch auch ohne Internetverbindung nutzbar, etwa im Einsatzgebiet ohne Netzabdeckung. Bereiche, die zuvor noch nie geladen wurden, können dabei fehlen. Sobald wieder eine Verbindung besteht, wird der Kartenspeicher automatisch im Hintergrund aktualisiert.

4.8.2 Erstellung von Markern



Um einen Marker zu erstellen, sind Koordinaten erforderlich. Diese können manuell, über die Karte oder anhand der aktuellen Position ausgewählt werden. Wenn GPS aktiviert ist, ermöglicht das Icon neben den Koordinaten das Setzen der eigenen Position.
Typ, Titel und Beschreibung sollten eindeutig gewählt werden, da diese auch auf der Karte angezeigt werden. Es empfiehlt sich, im Titel oder in der Beschreibung auch die Adresse des Markers anzugeben.

4.9 Statistik



Das Modul Statistik ermöglicht die Auswertung der erfassten Einträge. Die Übersicht zeigt alle verfügbaren Auswertungen als Kacheln mit kurzer Beschreibung, gruppiert nach Themen. Die Daten werden erst geladen, wenn eine Auswertung geöffnet und die Abfrage gestartet wird.
Für den Zugriff wird die Berechtigung für das Modul Statistik benötigt. Die Einschränkungen der Einsatzliste über Anzahl der angezeigten Einträge und Tiefe der Historie gelten für die Statistik nicht. Ausgewertet werden immer alle Einträge, die zur Abfrage passen.

4.9.1 Verfügbare Auswertungen

Überblick

- **Jahres-Statistiken:** Kennzahlen zu Anzahl Einträge, Stunden, durchschnittlichen Stunden und Kilometern im gewählten Zeitraum.
- **Einträge im Verlauf:** Anzahl der Einträge je Monat, getrennt nach Eintragsart.
- **Jahresvergleich:** Einträge je Monat mit einer Linie pro Jahr, um die Jahre miteinander zu vergleichen.

Personen

- **Fahrer-, Gruppenführer- und Einsatzleiter-Häufigkeit:** Wie oft Personen in der jeweiligen Funktion eingetragen waren. Fahrer und Gruppenführer werden nach Fahrzeug gestapelt und können auf einzelne Fahrzeuge eingeschränkt werden.

- **Anwesenheiten Häufigkeit** und **Anwesenheitsquote:** Anwesenheiten je Person als Anzahl bzw. als Anteil an allen Einträgen.
- **Anwesenheiten im Verlauf:** Anzahl anwesender Personen je Eintrag in zeitlicher Reihenfolge.
- **Stunden je Person:** Summe der Stunden aller Einträge, bei denen eine Person anwesend war.

Einträge

- **Einträge je Eintragsart**, **Stichwort-Häufigkeiten** und **Orts-Häufigkeit:** Anzahl der Einträge je Eintragsart, Stichwort oder Ort. Unterschiedliche Schreibweisen eines Stichworts werden zusammengefasst.
- **Ø Personenstärke je Eintragsart** und **je Stichwort:** durchschnittliche Anzahl anwesender Personen.

Fahrzeuge

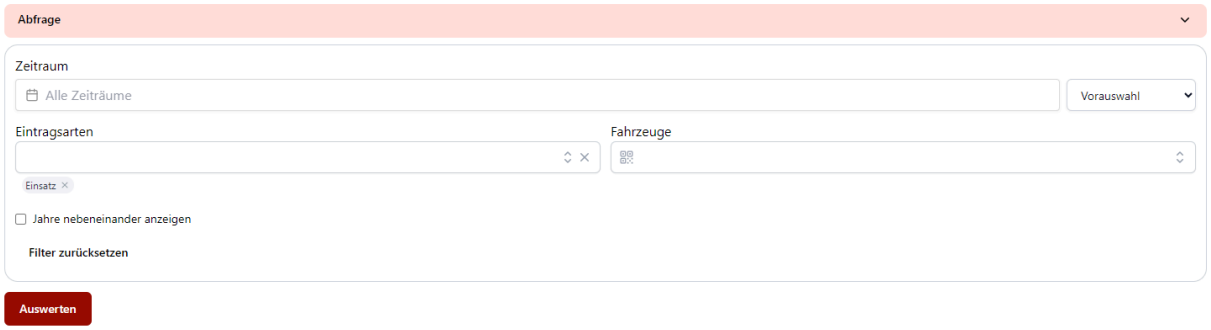
- **Kilometerstrecke:** Gefahrene Kilometer je Fahrzeug als Anteil am Gesamten.
- **Einträge je Fahrzeug** und **Stunden je Fahrzeug:** Einsatzhäufigkeit und -dauer der Fahrzeuge.

Zeiten

- **Stunden je Monat**, **Uhrzeit-**, **Tages-** und **Monats-Häufigkeit:** Zu welchen Zeiten Einträge beginnen bzw. wie viele Stunden je Monat anfallen.
- **Dauer-Häufigkeiten:** Dauer der Einträge in Klassen von 0,5 Stunden.
- **Verteilung der Einträge:** Jeder Punkt steht für einen Tag zu einer Uhrzeit mit Einträgen. Alle Tage des Zeitraums werden angezeigt, so fallen auch Pausen auf.
- **Wochentag × Uhrzeit:** Heatmap, an welchen Wochentagen und Uhrzeiten sich Einträge häufen.

Die meisten Auswertungen sind zusätzlich nach Eintragsart unterteilt, z.B. Einsatz und Übung.

4.9.2 Abfrage und Darstellung



Beim Öffnen einer Auswertung wird zunächst der Bereich **Abfrage** angezeigt. Je nach Auswertung stehen folgende Filter zur Verfügung:

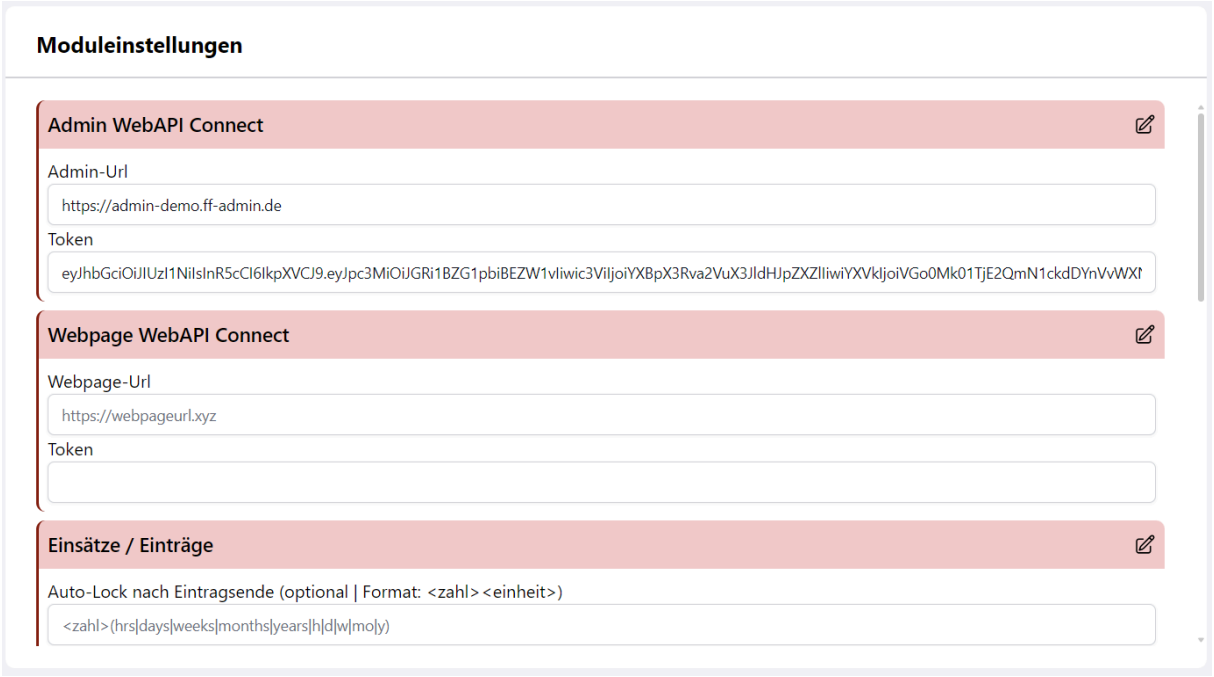
- **Zeitraum:** Frei wählbarer Datumsbereich oder eine Vorauswahl wie dieser Monat, letzter Monat, dieses Jahr, letztes Jahr, die letzten 12 Monate oder alle Zeiträume. Ohne Angabe werden alle Einträge ausgewertet.
- **Eintragsarten:** Einschränkung auf eine oder mehrere Eintragsarten.

- **Fahrzeuge:** Einschränkung auf einzelne Fahrzeuge (nur bei Fahrer- und Gruppenführer-Häufigkeit).
- **Jahre nebeneinander anzeigen:** Bei Balkendiagrammen können die Werte zusätzlich je Jahr getrennt dargestellt werden.

Über **Auswerten** wird die Abfrage gestartet, anschließend klappt der Abfrage-Bereich ein und das Ergebnis wird angezeigt. Die gewählten Filter bleiben beim Wechsel zwischen den Auswertungen erhalten und können über **Filter zurücksetzen** geleert werden.

Beim Überfahren oder Antippen der Diagramme werden die genauen Werte angezeigt. Über die Legende lassen sich einzelne Reihen ein- und ausblenden. Bei vielen Werten kann der angezeigte Bereich über den Schieberegler unterhalb des Diagramms eingegrenzt werden.

4.10 Modulspezifische Einstellungen



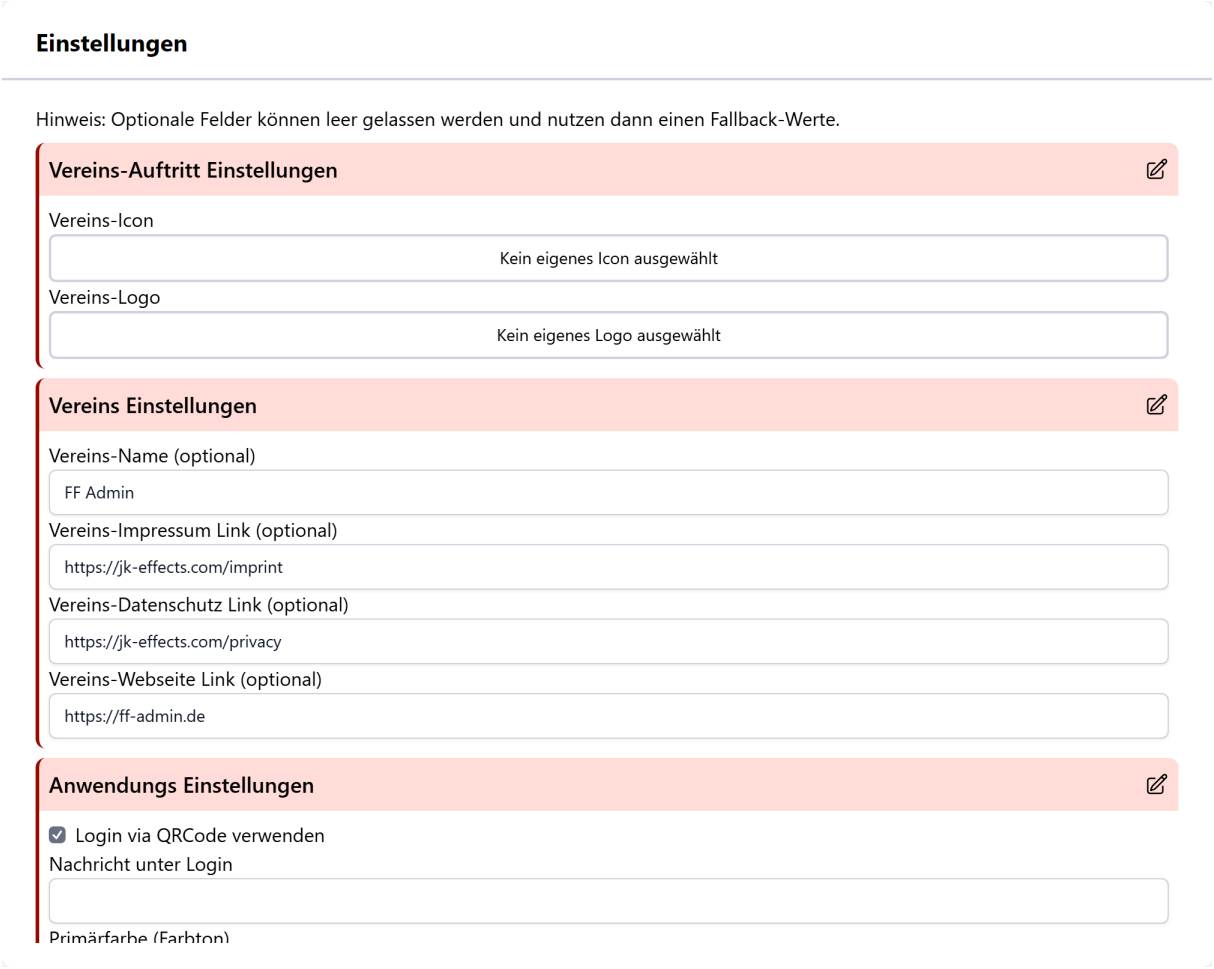
Das Modul Moduleinstellungen erweitert das bestehende Einstellungssystem um spezifische Optionen, die ausschließlich einzelne Module betreffen. Beispielsweise kann festgelegt werden, wie lange nach einem Eintrag diese automatisch gesperrt oder ob und welche Daten aus FF Admin Synchronisiert werden sollen.

Die Einstellungen zu den Modulen sind nur sichtbar, wenn ein Nutzer sowohl die Berechtigung für die Moduleinstellungen als auch für das jeweilige Modul besitzt. Dadurch wird sichergestellt, dass nur autorisierte Personen modulbezogene Anpassungen vornehmen können.

Mit den Moduleinstellungen lassen sich Funktionen und Darstellungen einzelner Module gezielt steuern, ohne das Gesamtsystem zu beeinflussen.

Admin WebAPI Connect: Damit Daten aus FF Admin synchronisiert werden können, muss ein WebAPI Token aus dem Admin angegeben werden. Die URL ist nur die URL, unter der das Backend läuft. Ein angegebener Pfad wird ignoriert. Bei der Speicherung der Einstellung wird direkt die Gültigkeit geprüft. Das Token wird verschlüsselt gespeichert und danach nicht mehr angezeigt. Bleibt das Feld leer, bleibt das gespeicherte Token erhalten. Wird in FF Admin das Secret gewechselt (Abschnitt 2.7), muss dort ein neues Token erzeugt und hier eingetragen werden.
Webpage WebAPI Connect: Damit Daten zu FF Wbpage synchronisiert werden können, muss ein WebAPI Token aus dem CMS angegeben werden. Die URL ist die, die zum aufrufen der CMS-Ansicht verwendet wird. Ein angegebener Pfad wird ignoriert. Das Token wird verschlüsselt gespeichert und danach nicht mehr angezeigt. Bleibt das Feld leer, bleibt das gespeicherte Token erhalten. Wird in FF Admin das Secret gewechselt (Abschnitt 2.7), muss dort ein neues Token erzeugt und hier eingetragen werden.
Auto-Lock: Einträge können automatisch auf gesperrt gesetzt werden, wenn eine Bestimmte Dauer seit dem gesetzten Ende des Eintrags vergangen ist. Die Dauer kann x Stunden, Tage, Wochen, Monate oder Jahre betragen. Der Job zum Sperren läuft alle 30 Minuten. Dadurch kann sich das Sperren um 30 min verzögern.
Manueller Unlock nach Eintragsende: Optional lässt sich zusätzlich festlegen, wie lange nach dem gesetzten Ende eines Eintrags dieser über den Schlüssel (Abschnitt 4.1.6) überhaupt noch manuell entsperrt werden darf. Nach Ablauf dieser Frist ist das Entsperren für niemanden mehr möglich, auch nicht für Admins. Ist keine Dauer hinterlegt, gilt keine zeitliche Begrenzung für das manuelle Entsperren.
Synchronisierung: Die Datensynchronisierung aus dem Admin muss gezielt für verfügbare Module aktiviert werden. Ist die Synchronisierung aktiviert, kann in den Einstellungen gestartet werden. Die Synchronisierung läuft immer nachts um 0Uhr.
IncidentApi: Die Incident-Schnittstelle kann in den Einstellungen aktiviert werden. Dabei kann auch eingestellt werden, wie anhand des Incidents ein neuer Eintrag angelegt werden soll.

4.11 Anwendungsspezifische Einstellungen



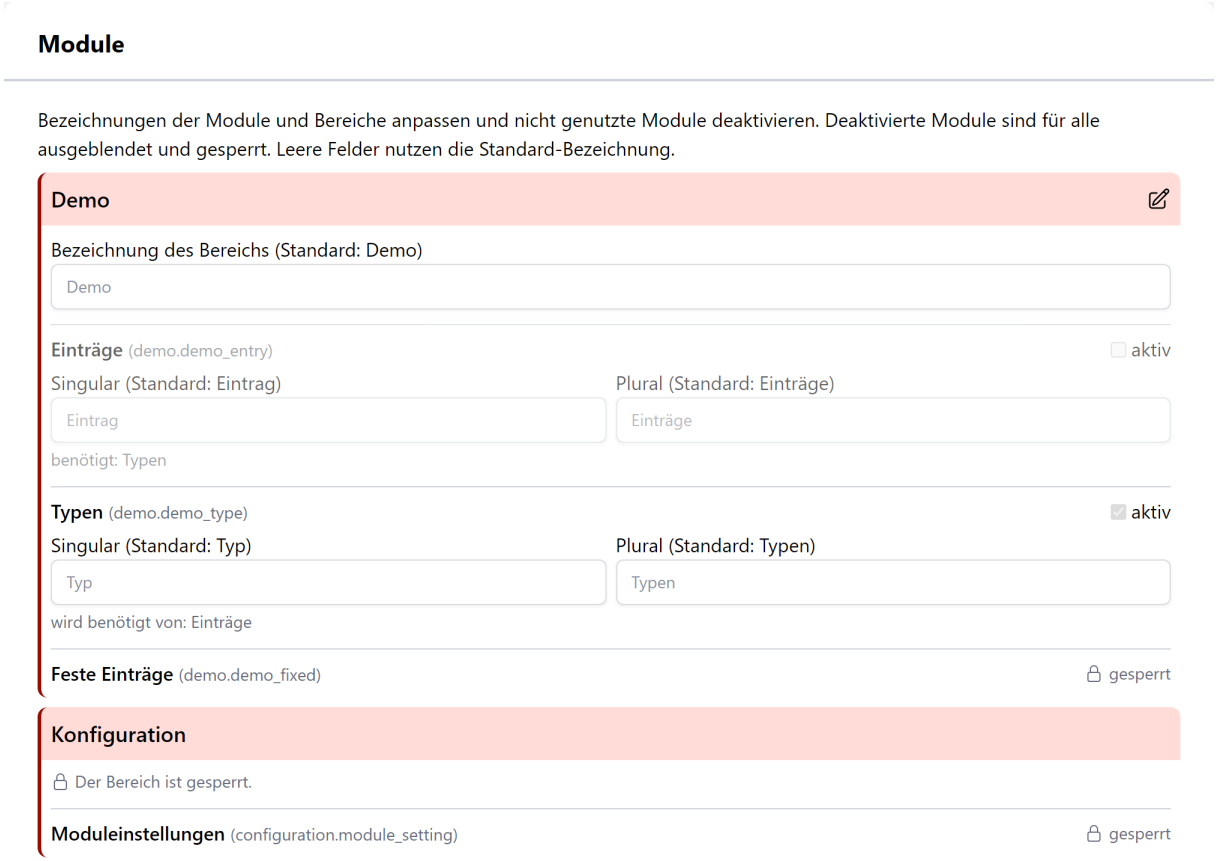
Das Modul Einstellungen ermöglicht eine flexible globale Konfiguration der Anwendung - ohne Neustart des Systems. Alle Anpassungen können direkt zur Laufzeit vorgenommen werden. Dadurch lassen sich beispielsweise Vereinslogos, Icons oder Texte im Login-Bereich individuell anpassen, ohne in den laufenden Betrieb einzugreifen.

Neben dem optischen Vereinsauftritt können auch grundlegende Informationen wie Links zum Impressum, zur Datenschutzerklärung oder zur Vereinswebseite hinterlegt werden. Diese Daten fließen dann automatisiert an verschiedenen Stellen im System ein.

Darüber hinaus ersetzt das Einstellungsmodul Teile der früheren, fest codierten Umgebungsvariablen (ENV). Werte wie etwa das Backup-Intervall oder E-Mail-Konten für den Versand können direkt in der Oberfläche geändert werden. Ein gespeichertes Passwort des E-Mail-Kontos wird dabei nicht angezeigt.

Insgesamt erlaubt das Modul eine Anpassung des Systems ohne Zugriff auf die Serverkonfiguration und Neustarts. Jedoch erfordern manche Änderungen das Neuladen der Webseite / WebApp.

4.12 Modulbezeichnungen



Unter `Verwaltung > Modulbezeichnungen` lassen sich die Bezeichnungen der Module und Bereiche an den eigenen Sprachgebrauch anpassen und nicht genutzte Module deaktivieren. Die Ansicht ist nach Bereichen gegliedert, jeder Bereich wird über das Stift-Symbol bearbeitet und gemeinsam gespeichert.

Für die Ansicht wird die Leseberechtigung des Moduls `Modulbezeichnungen` im Abschnitt Verwaltung benötigt, für Änderungen die Berechtigung zum Bearbeiten. Da ein deaktiviertes Modul für alle Nutzer gesperrt ist, handelt es sich um ein privilegiertes Modul: Die Berechtigung kann nur von Administratoren vergeben werden.

4.12.1 Bezeichnungen

Jedes Modul hat eine Standard-Bezeichnung im Singular und im Plural, jeder Bereich eine Standard-Bezeichnung. Sie stehen in Klammern über den Eingabefeldern. Ein ausgefülltes Feld ersetzt die Standard-Bezeichnung, ein leeres Feld nutzt wieder den Standard.

- **Plural:** Navigation, Seitentitel der Listen, Berechtigungen und Benachrichtigungseinstellungen, z.B. `Mitglieder`.
- **Singular:** Erstellen und Bearbeiten einzelner Einträge, z.B. `Mitglied erstellen`.

Die Bezeichnung wird ohne Artikel in die Texte eingesetzt. Bei einer Umbenennung von `Mitglied` in `Person` heißt die Schaltfläche danach `Person erstellen`.

Änderungen erscheinen sofort in allen geöffneten Sitzungen, ein Neuladen ist nicht nötig. Der Browser merkt sich die zuletzt geladenen Bezeichnungen, damit sie beim Start der Anwendung auch bei langsamer Verbindung direkt angezeigt werden.

4.12.2 Module deaktivieren

Über `aktiv` wird ein Modul ein- oder ausgeschaltet. Ein deaktiviertes Modul ist für alle Nutzer ausgeblendet, auch für Administratoren und den Besitzer:

- Es erscheint nicht mehr in der Navigation, in den Berechtigungen und in den Benachrichtigungseinstellungen.
- Aufrufe über einen gespeicherten Link führen zur Startseite zurück.
- Der Server lehnt alle Anfragen an das Modul ab, auch über WebApi-Token. Benachrichtigungen des Moduls werden nicht versendet.

Sind alle Module eines Bereichs deaktiviert, wird auch der Bereich ausgeblendet.

Die Daten und vergebenen Berechtigungen des Moduls bleiben erhalten. Nach dem erneuten Aktivieren steht das Modul wieder mit allen Einträgen zur Verfügung.

4.12.3 Abhängigkeiten

Manche Module setzen andere voraus, weil deren Einträge zum Erstellen benötigt werden, z.B. Mitglieder eine Anrede oder Fahrzeuge einen Fahrzeugtyp. Diese Beziehungen werden unter dem Modul angezeigt:

- **benötigt:** Module, die aktiv sein müssen, damit dieses Modul aktiviert werden kann.
- **wird benötigt von:** Module, die zuerst deaktiviert werden müssen, bevor dieses Modul deaktiviert werden kann.

Ist eine Änderung deshalb nicht möglich, ist der Schalter `aktiv` gesperrt, der Grund erscheint beim Darüberfahren mit der Maus. Module eines Bereichs, die voneinander abhängen, können in

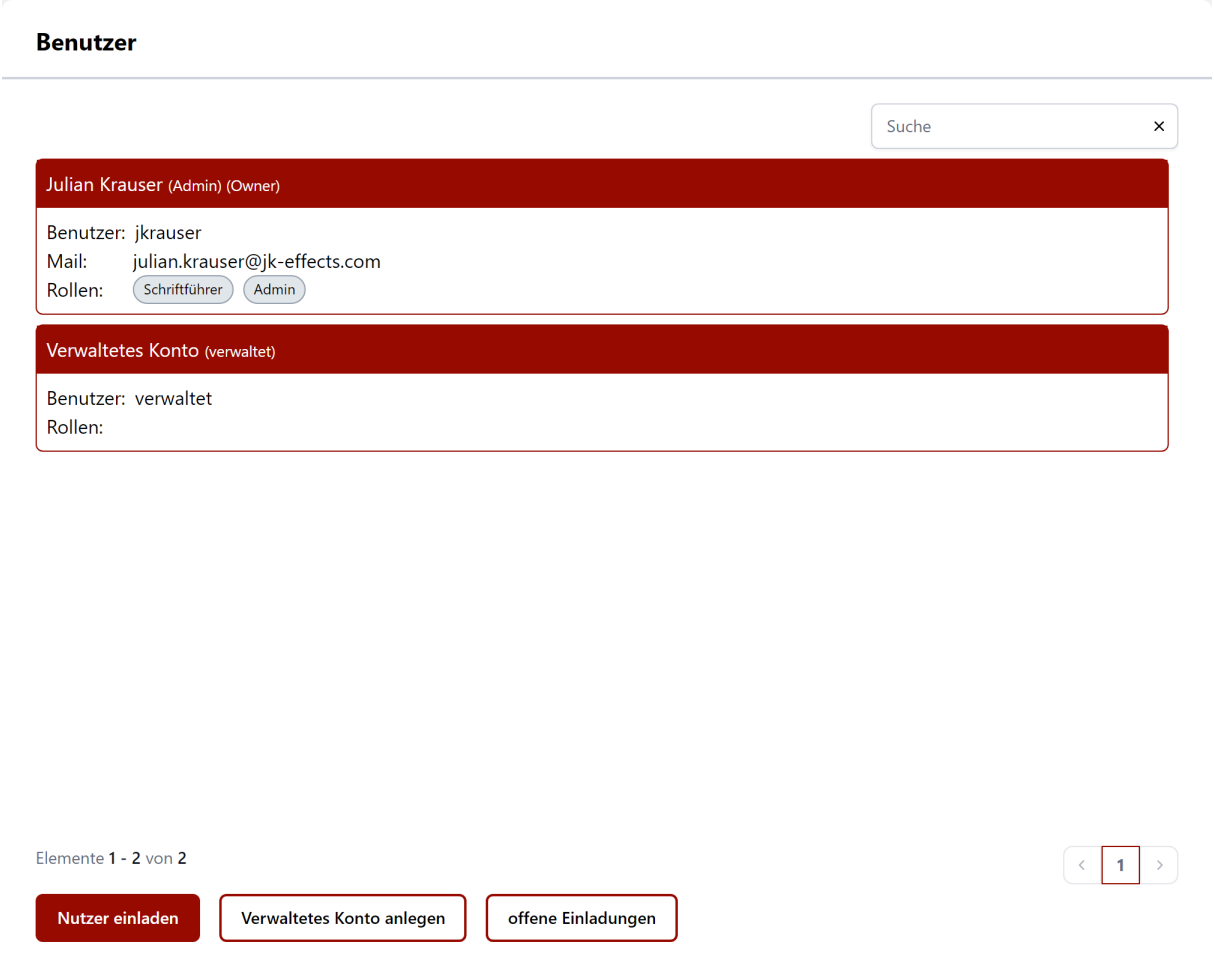
einem Schritt gemeinsam aktiviert oder deaktiviert werden. Liegt das benötigte Modul in einem anderen Bereich, muss zuerst dieser Bereich gespeichert werden.

Setzt ein aktives Modul ein deaktiviertes voraus, z.B. nach dem Einspielen eines Backups, meldet der Serverzustand (Abschnitt 2.6) einen Fehler.

4.12.4 Gesperrte Module

Mit einem Schloss gekennzeichnete Module und Bereiche können weder umbenannt noch deaktiviert werden. Dazu gehören die Bereiche Konfiguration und Verwaltung mit den Modulen dieser Anleitung, da sie für den Betrieb der Anwendung notwendig sind. Weitere Module können von der jeweiligen Anwendung gesperrt sein.

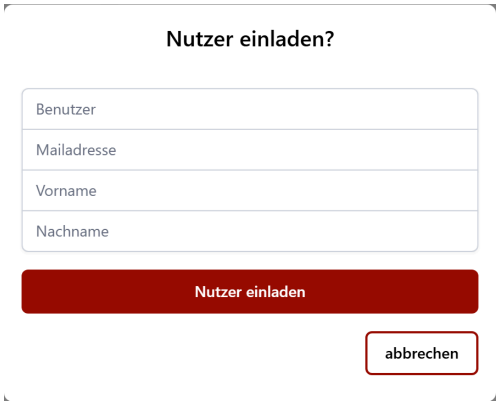
4.13 Benutzerverwaltung



Daten können nur von angemeldeten Benutzern geändert werden. Diese Benutzer melden sich mit einem TOTP oder Passwort an. Ein TOTP (Time-based One-time Password) ist ein zeitbasiertes Passwort, das sich jede Minute ändert. Nutzer können außerdem Passkeys (Abschnitt 5.5) für einen schnelleren Login erstellen. Jeder Benutzer hat Lese-, Bearbeitungs- und Löschrechte. Es können auch Rollen erstellt und Benutzern zugewiesen werden.

Benutzer mit dem Status `Owner` haben unabhängig der gesetzten Berechtigungen Vollzugriff auf die gesamte Anwendung.

4.13.1 Benutzer einladen



Benutzer mit eigener E-Mail-Adresse werden per Einladung hinzugefügt. Konten ohne E-Mail-Adresse lassen sich direkt anlegen (siehe „Verwaltete Konten“).
Einladungen sind nur solange gültig, bis sie angenommen oder zurückgezogen werden.
Bei einer Einladung wird ein personalisierter Link an die angegebene E-Mail-Adresse gesendet, über den sich der Benutzer registrieren kann. Nach der Registrierung hat der Benutzer noch keine Rollen oder Berechtigungen. Diese müssen nach der Registrierung zugewiesen werden.
Ein Benutzername oder eine E-Mail-Adresse darf nur einmal existieren.



Wenn ein Benutzer keine Berechtigungen hat oder diese ihm entzogen wurden, wird die Meldung auf der linken Seite angezeigt.
Nach der Registrierung wird diese Meldung standardmäßig angezeigt.
Mit einem Klick auf die Schaltfläche „Jetzt Berechtigungen neuladen“ werden die aktuellen Berechtigungen vom Server abgefragt. Bleibt die Ansicht unverändert, wurden dem Benutzer noch keine Berechtigungen zugewiesen. Andernfalls erfolgt eine Weiterleitung zur Admin-Oberfläche.

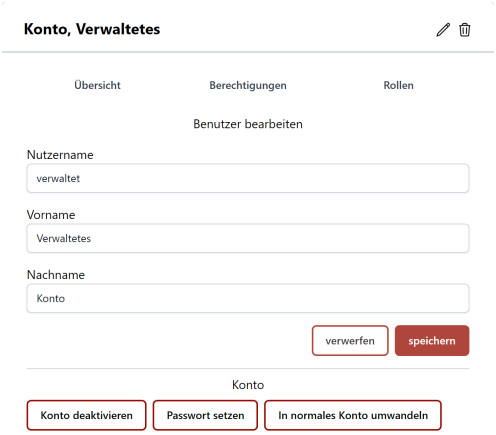
4.13.2 Verwaltete Konten

Verwaltete Konten haben keine E-Mail-Adresse. Sie sind für Personen gedacht, denen du kein Passwort weitergeben möchtest: Du legst das Konto unter `Verwaltung > Benutzer` mit Benutzername, Name und Passwort an, meldest es auf dem Gerät an und richtest dort einen Passkey ein. In der Benutzerliste ist es mit `(verwaltet)` gekennzeichnet.



- Ein Zurücksetzen per E-Mail und TOTP sind nicht möglich, Benachrichtigungen kommen nur in der App oder als Push.
- Benutzername und Passwort lassen sich mit dem aktuellen Passwort ändern. Passkeys lassen sich nur mit dem Passwort einrichten oder löschen.
- Wer Benutzer bearbeiten darf, kann das Passwort neu setzen. Dabei enden alle Sitzungen und alle Passkeys des Kontos werden gelöscht. Bei Konten mit Administratorrechten ist das nicht möglich, bei Konten mit privilegierten Modulen nur für Administratoren.
- Administratoren können ein verwaltetes Konto in ein normales umwandeln und umgekehrt, außer bei Konten mit Administratorrecht. In beiden Fällen enden alle Sitzungen und Passkeys des Kontos.
  - **In ein normales Konto:** Das Konto erhält eine E-Mail-Adresse, das bisherige Passwort bleibt gültig. An die Adresse geht ein Link, mit dem der Benutzer sein Passwort ersetzen kann.
  - **In ein verwaltetes Konto:** Die E-Mail-Adresse wird entfernt, die Anmeldung läuft danach nur noch über das dabei vergebene Passwort.

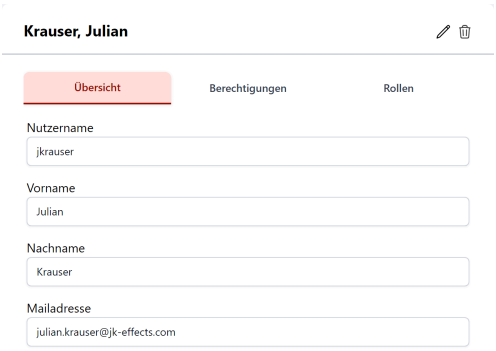
4.13.3 Konten deaktivieren



Wer Benutzer bearbeiten darf, kann ein Konto über das Stift-Symbol in der Detailansicht deaktivieren. Alle Sitzungen enden sofort, eine Anmeldung ist bis zur erneuten Aktivierung nicht mehr möglich. In der Benutzerliste ist das Konto mit `(deaktiviert)` gekennzeichnet. Das eigene Konto und das des Owners lassen sich nicht deaktivieren, Konten mit Administratorrechten nur durch Administratoren.

Im selben Bereich liegen auch das Setzen des Passworts und das Umwandeln verwalteter Konten.

4.13.4 Details

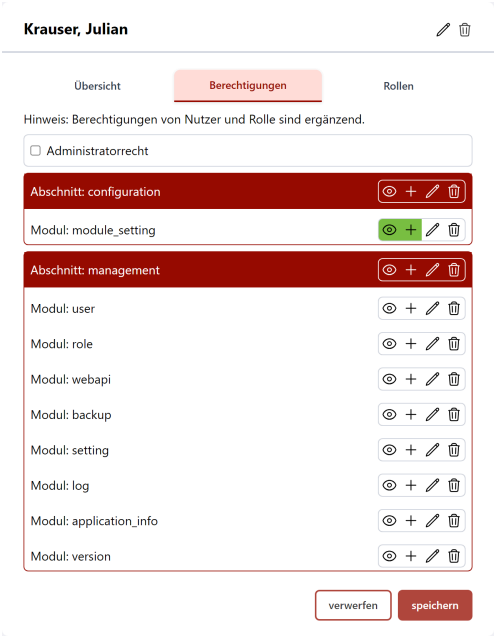


Jeder Benutzer verfügt über einen Benutzernamen, eine E-Mail-Adresse, Vor- und Nachname sowie zugewiesene Berechtigungen und Rollen.

In der Detailansicht können diese Daten eingesehen und bearbeitet werden. Benutzername und E-Mail-Adresse anderer Nutzer können dort nur Administratoren ändern. Das eigene Konto kann jeder Nutzer selbst ändern (siehe „Zugang und Anmeldedaten“).

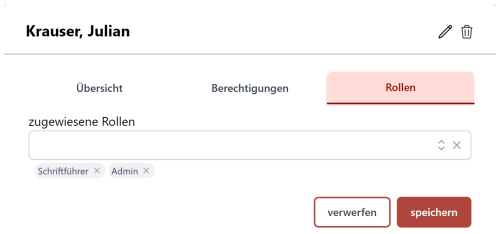
Wichtig: Bei einer Änderung des Benutzernamens ist anschließend eine Anmeldung per Passwort oder TOTP nicht mehr möglich, sofern der neue Nutzernamen nicht weitergegeben wird. Der Zugang bleibt jedoch über Passkeys erhalten. Noch offene Links zum Zurücksetzen des Zugangs werden bei einer Änderung von Benutzername oder E-Mail-Adresse ungültig.

4.13.5 Berechtigungen



Berechtigungen werden immer addiert.
Das heißt: Wenn ein Benutzer direkt zugewiesene Berechtigungen und Rollen erhält, werden die Berechtigungen addiert und der Benutzer kann alles, was diesem direkt zugewiesen wurde und was die Rolle kann.
Das Berechtigungsmodell unterscheidet zwischen Lese-, Erstellungs-, Änderungs- und Löschberechtigungen. Die Berechtigungen sind abgestuft. Das heißt, wer anlegen darf, darf auch lesen. Wer bearbeiten darf, darf auch anlegen und dementsprechend auch lesen usw.
Die Berechtigungen sind entsprechend der Bereiche in der Navigationsleiste gruppiert. Wenn die Berechtigung für einen Abschnitts auf Lesen gesetzt ist, kann der Benutzer mit dieser Berechtigung Daten in allen Modulen des Abschnitts lesen.
Erweiterte Berechtigungen können pro Modul gesetzt werden. Dabei gelten die Berechtigungen des Abschnitts immer mehr als die eines Moduls innerhalb des Abschnitts.
Wer kein Administrator ist, sieht das Administratorrecht und privilegierte Module ausgegraut und kann sie nicht ändern.

4.13.6 Rollen

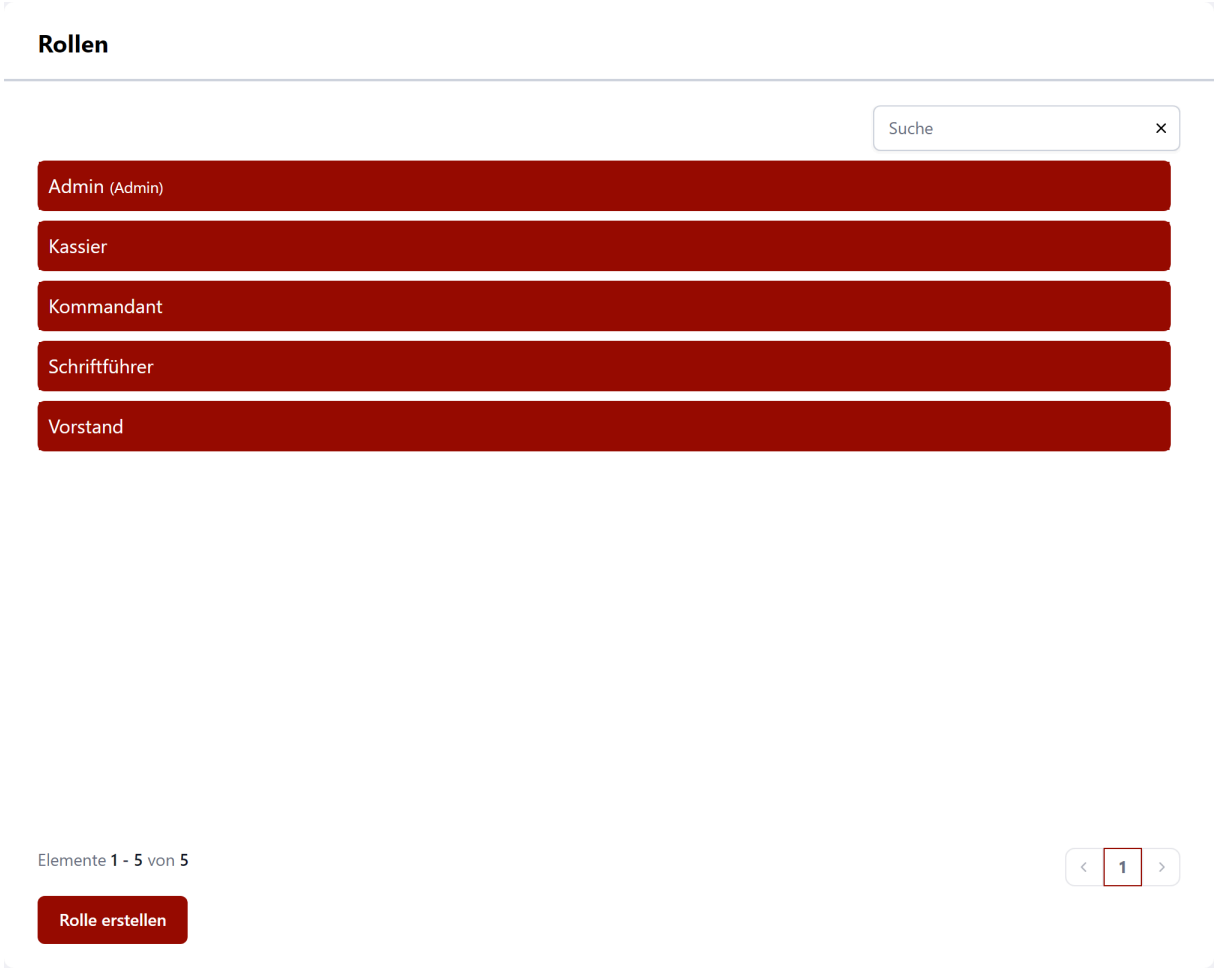


Rollen und Berechtigungen vergibt, wer Benutzer bzw. Rollen bearbeiten darf. Das Administratorrecht und privilegierte Module (zum Beispiel Backup, Webapi, Log und Einstellungen) vergeben nur Administratoren. Das gilt auch für Rollen, die solche Rechte enthalten: Nur Administratoren dürfen sie bearbeiten, löschen, zuweisen oder entziehen.
Rollen fassen Berechtigungen zusammen und ergänzen die einem Benutzer direkt zugewiesenen Berechtigungen.
Rollen können sich nicht gegenseitig ausschließen.
Einem Benutzer können mehrere Rollen zugewiesen werden. Muss aber keine Rollen haben.

Hinweis: Änderungen an Berechtigungen oder Rollen wirken sich unmittelbar auf die Benutzer

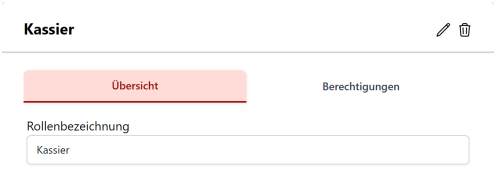
aus. Sie werden sofort an die Anwendung übertragen und dort angezeigt, sofern diese beim betroffenen Benutzer geöffnet ist. Ist der Nutzer offline, werden die Änderungen beim nächsten Öffnen der Anwendung übernommen.

4.14 Rollenverwaltung



Rollen dienen der Abstraktion von Berechtigungen. Dadurch können spezifische Berechtigungsschemata direkt einem oder mehreren Benutzern zugewiesen werden, ohne dass diese mehrfach manuell direkt zugewiesen werden müssen.

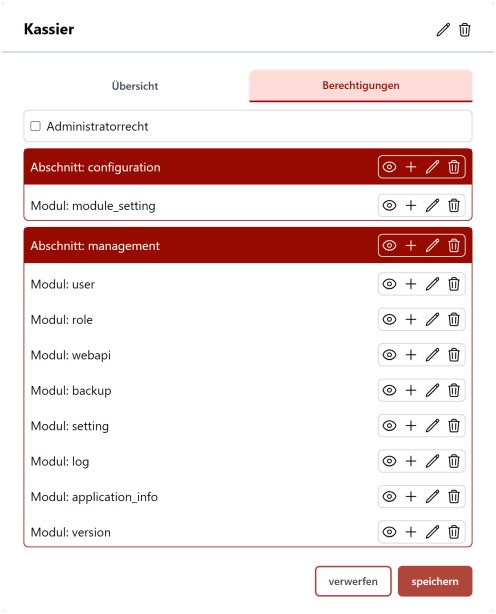
4.14.1 Details



Jede Rolle verfügt über einen Bezeichnung sowie zugewiesene Berechtigungen.
In der Detailansicht können diese Daten eingesehen und bearbeitet werden.
Rollen bearbeitet, wer das Recht zum Bearbeiten von Rollen hat. Rollen mit Administratorrecht oder privilegierten Modulen dürfen nur Administratoren bearbeiten oder löschen.

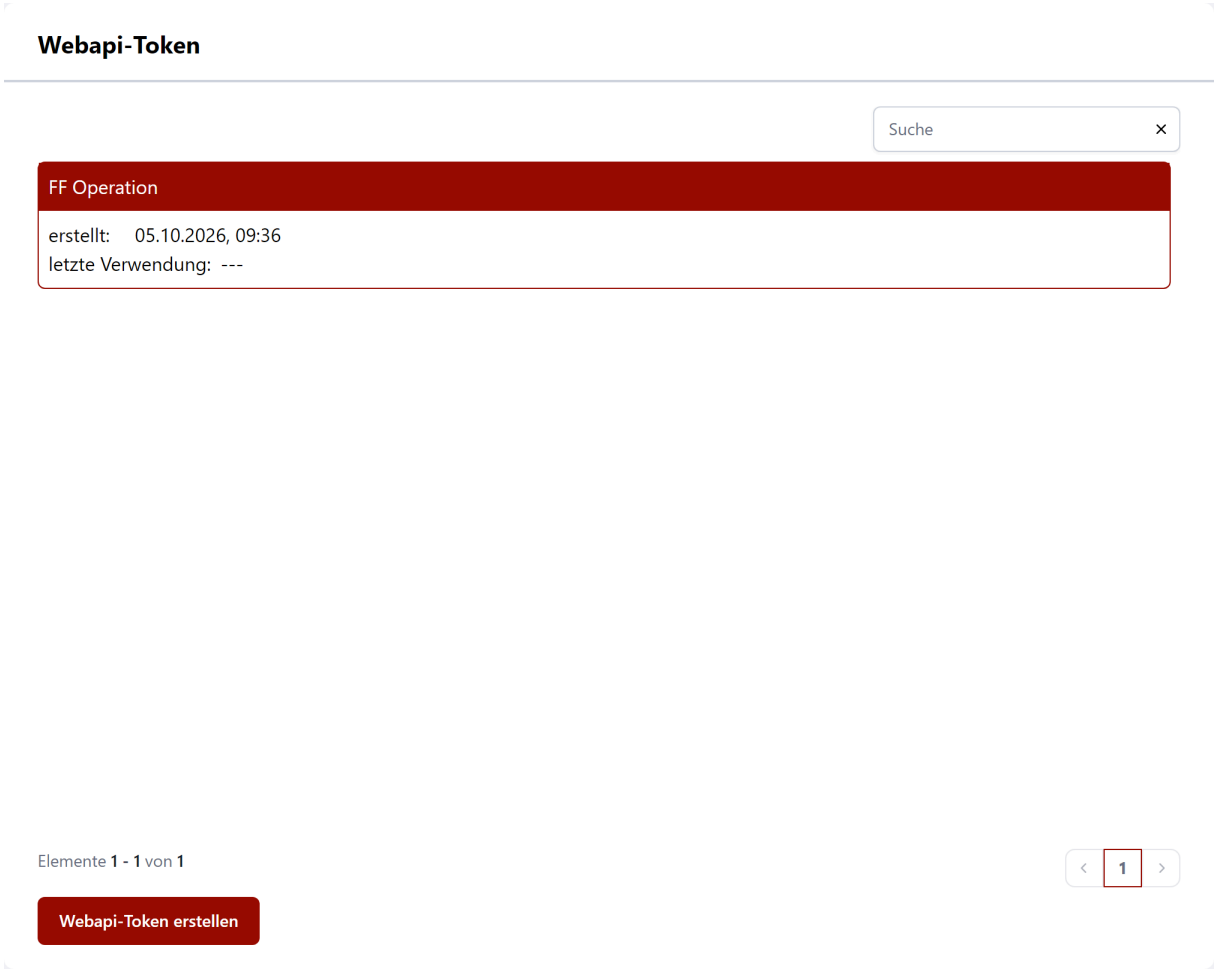
4.14.2 Berechtigungen

Berechtigungen werden immer addiert.
Das heißt: Wenn ein Benutzer mehrere Rollen zugewiesen bekommt, werden die Berechtigungen addiert und der Benutzer kann alles, was die Rollen ermöglichen. Zusätzlich werden auch die nutzerspezifischen Berechtigungen addiert.
Das Berechtigungsmodell unterscheidet zwischen Lese-, Erstellungs-, Änderungs- und Löschberechtigungen. Die Berechtigungen sind abgestuft. Das heißt, wer anlegen darf, darf auch lesen. Wer bearbeiten darf, darf auch anlegen und dementsprechend auch lesen usw.
Die Berechtigungen sind entsprechend der Bereiche in der Navigationsleiste gruppiert. Wenn die Berechtigung für einen Abschnitts auf Lesen gesetzt ist, kann der Benutzer mit dieser Berechtigung Daten in allen Modulen des Abschnitts lesen.

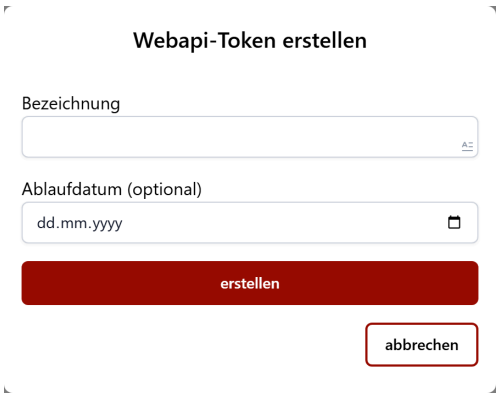


Erweiterte Berechtigungen können pro Modul gesetzt werden. Dabei gelten die Berechtigungen des Abschnitts immer mehr als die eines Moduls innerhalb des Abschnitts.
Manche Module bieten zusätzlich einzelne **weitere Berechtigungen** an, die über das Lese-/Erstellungs-/Änderungs-/Löschschema hinausgehen, z.B. um einzelnen Nutzern gezielt eine sonst nur Admins vorbehaltene Aktion zu erlauben oder um Werte wie Anzeigelimits einzuschränken. Diese werden am Ende des Berechtigungseditors gesondert aufgeführt und sind je nach Art entweder als Häkchen oder als Eingabewert gesetzt.

4.15 WebApi



WebApi-Tokens ermöglichen externen Clients oder anderen Anwendungen des FF Ökosystems oder eigenen Anwendungen die Kommunikation mit der Anwendung. Dabei haben WebApis Zugriff auf fast alle Daten der Anwendung, sofern die Berechtigungen gesetzt sind.
Kategorisch ausgeschlossen ist der Zugriff auf die Anwendungsmodule Backup, WebApi und Logs sowie Serverinformationen wie z.B. die Version. Das gilt auch für WebApis mit Administratorrecht. Diese Module werden im Berechtigungseditor der WebApi nicht angezeigt.
Auch Konten deaktivieren, verwaltete Konten anlegen und umwandeln sowie Passwörter setzen ist über eine WebApi nicht möglich. Abgerufene Benutzer können deaktiviert sein, verwaltete Konten haben keine E-Mail-Adresse.
WebApis mit Administratorrecht können nur Administratoren bearbeiten. Haben sie kein Ablaufdatum, weist der Serverzustand (Abschnitt 2.6) darauf hin.



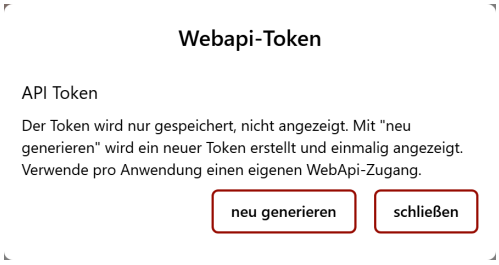
Eine WebApi kann mit einem Verfallsdatum erstellt werden. Dieses Ablaufdatum wird fest eingestellt, wenn eine WebApi Zugriff anfordert.
Zusätzlich wird für jede WebApi ein Token erstellt, das zur Authentifizierung der WebApi verwendet werden kann.

4.15.1 Details



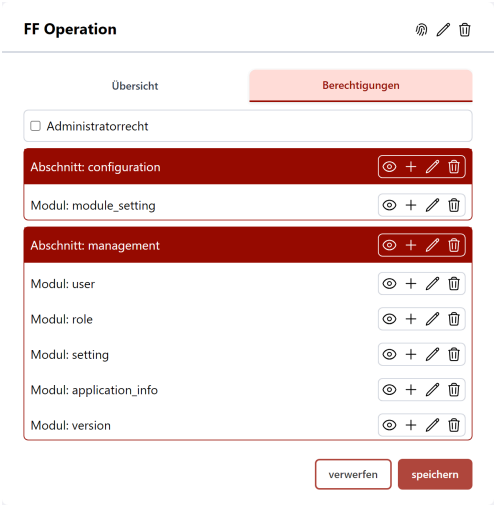
Jede WebApi verfügt über eine Bezeichnung und ein Verfallsdatum.
In der Detailansicht können diese Daten eingesehen und bearbeitet werden.
Wichtig:

Bei einer Änderung der Berechtigungen werden die Berechtigungen erst übernommen, wenn das Zugangs-Token (siehe unten Abschnitt 4.15.3) neu ausgestellt wird. Das Intervall, in dem das Token erneuert wird, ist die `JWT-Gültigkeitsdauer`, welche in den Einstellungen (Abschnitt 4.11) gesetzt werden kann.



Unter dem Fingerabdruck-Symbol kann das Token der WebApi erneuert werden.
Ein Token wird nur einmalig direkt nach dem Erzeugen angezeigt und muss dann kopiert werden. Danach lässt es sich nicht mehr abrufen. Geht es verloren, muss es mit `neu generieren` ersetzt werden. Das bisherige Token wird dabei sofort ungültig, alle Systeme, die es verwenden, müssen umgestellt werden. Pro Anwendung sollte deshalb ein eigener WebApi-Zugang verwendet werden.
Diese Token sind JWTS und können daher ohne Kenntnis der Serverkonfiguration nicht gefälscht werden.
Zur schnelleren Konfiguration in anderen Anwendungen zeigt der Dialog mit dem Token zusätzlich die Domain des Server der Anwendung an.

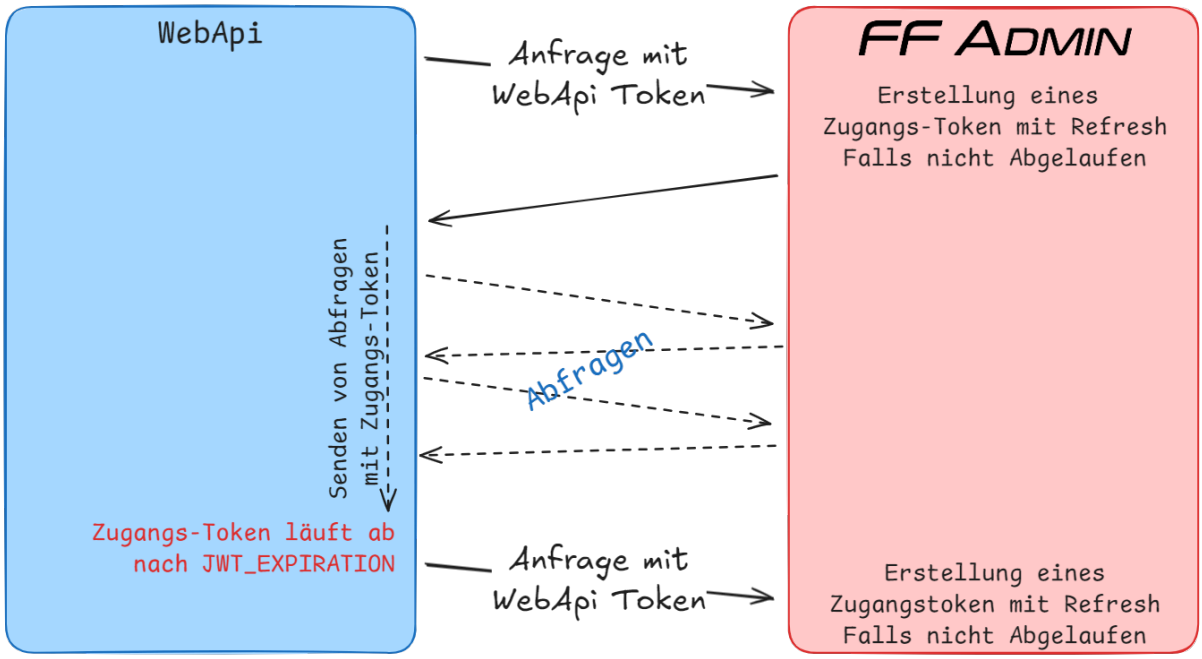
4.15.2 Berechtigungen



Einer WebApi können im Vergleich zu Benutzern nur direkte Berechtigungen zugewiesen werden (analog zu den Rollen).
Das Berechtigungsmodell unterscheidet zwischen Lese-, Erstellungs-, Änderungs- und Löschberechtigungen. Die Berechtigungen sind abgestuft. Das heißt, wer anlegen darf, darf auch lesen. Wer bearbeiten darf, darf auch anlegen und dementsprechend auch lesen usw.
Die Berechtigungen sind entsprechend der Bereiche in der Navigationsleiste gruppiert. Wenn die Berechtigung für einen Abschnitts auf Lesen gesetzt ist, kann der Benutzer mit dieser Berechtigung Daten in allen Modulen des Abschnitts lesen.
Erweiterte Berechtigungen können pro Modul gesetzt werden. Dabei gelten die Berechtigungen des Abschnitts immer mehr als die eines Moduls innerhalb des Abschnitts.

4.15.3 WebApi Zugriffsablauf

Die WebApi bezieht den Zugang wie folgt:



4.15.4 API-Clients

Zur einfacheren Verwendung der WebApi in eigenen Anwendungen gibt es fertige API-Clients für TypeScript, Python und Go. Sie werden aus der OpenAPI-Spezifikation des jeweiligen Servers erzeugt und decken damit alle Routen ab, auch die des Core.

App	TypeScript (npm)	Python	Go
FF Admin	`@ff-admin/ff-admin-api`	`ff-admin-api`	`code.jk-effects.cloud/ff-admin/ff-admin-api`
FF Operation	`@ff-admin/ff-operation-api`	`ff-operation-api`	`code.jk-effects.cloud/ff-admin/ff-operation-api`

Die Version eines Pakets folgt der Linie der Anwendung: `1.25.x` passt z.B. zu FF Admin 1.25. Für jede unterstützte Linie (Abschnitt 2.12.1) gibt es ein passendes Paket. Alle Pakete liegen in der Registry unter https://code.jk-effects.cloud/ff-admin/ff-admin-api-clients.

Alle Clients bieten dasselbe:

- `WebApiClient` für die Anmeldung mit einem WebApi-Token und `UserClient` für die Anmeldung als Benutzer (Passwort oder TOTP).
- Automatische Erneuerung des Zugangs-Tokens, wenn dieses abgelaufen ist.
- Einen Fehlertyp `ApiError` mit Status und den Fehlerdetails des Servers.
- Alle Routen der Spezifikation. Routen, die für WebApis gesperrt sind (siehe oben), schlagen mit einem WebApi-Token fehl und sind nur mit einem `UserClient` nutzbar.

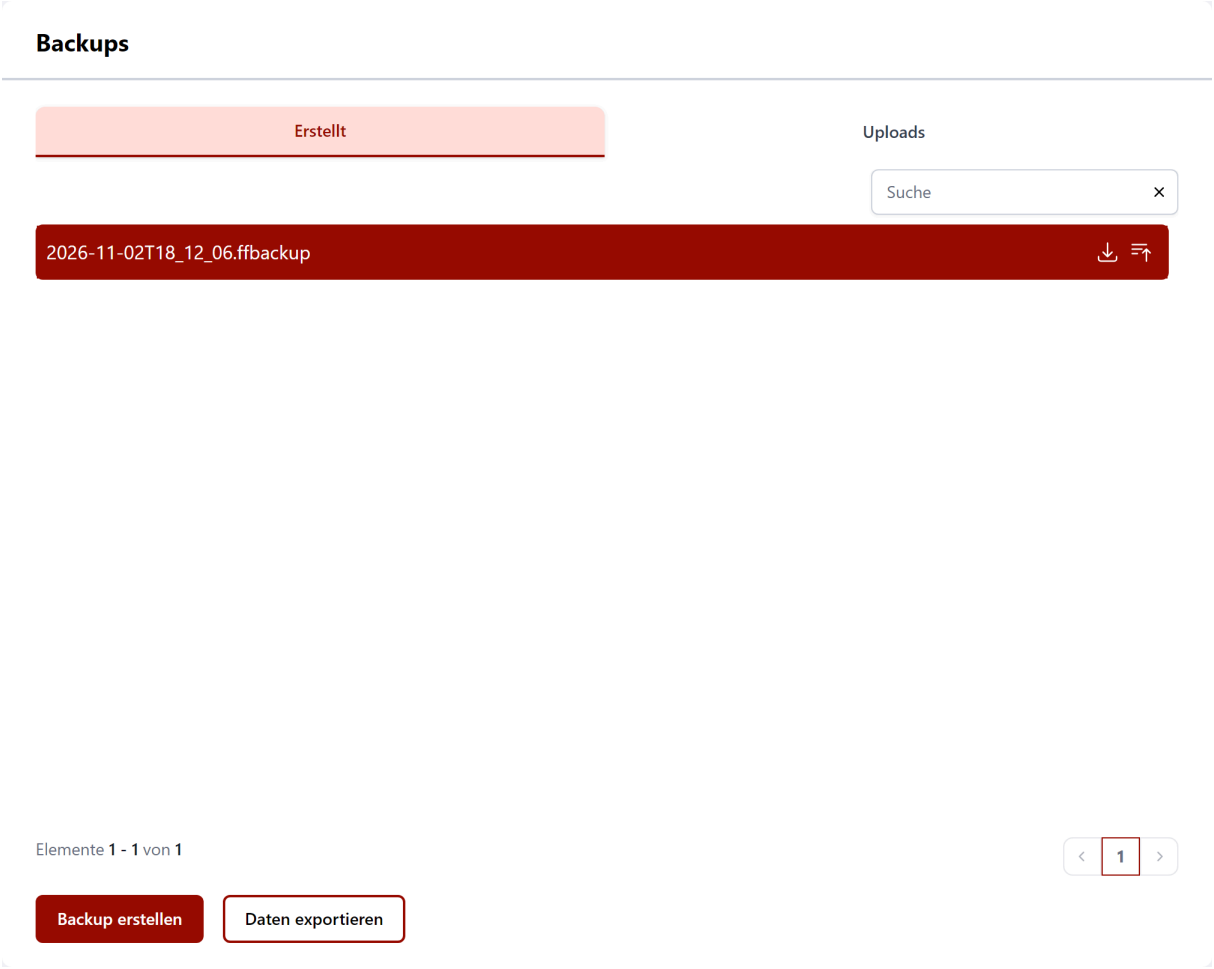
Beispiel in TypeScript:

```typescript
import { WebApiClient } from "@ff-admin/ff-admin-api";

const client = new WebApiClient({
  server: "https://admin.ff-beispiel.de",
  webapiToken: "<webapi-token>",
});
await client.initialize();

const page = await client.api.admin.member.getAllMembers({ query: { offset: 0, count: 25 } });
```

Anleitungen zur Installation und Nutzung in allen Sprachen stehen in der README des Repositories. Welche Routen und Felder es gibt, zeigt die API-Dokumentation des Servers, sofern sie aktiviert ist.

Das bisherige Paket `@ff-admin/webapi-client` (https://code.jk-effects.cloud/FF-Admin/ff-admin-webapi-client) bleibt verfügbar und enthält zusätzlich die Anbindung an das CMS von FF Webpage. Für neue Anbindungen an FF Admin und FF Operation sollten die oben genannten Clients verwendet werden.

4.16 Backups

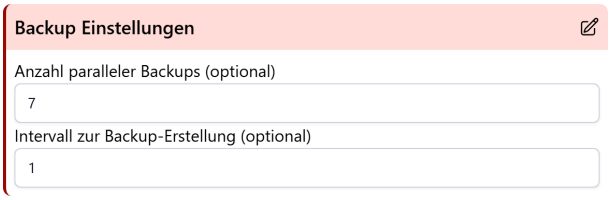


Backups sind automatisch gesicherte Datenstände der Anwendung, die jederzeit wiederhergestellt werden können. Die Dateien sind verschlüsselt und werden beim Einspielen auf Veränderungen geprüft.

Ein Backup lässt sich nur auf einem Server derselben Anwendung einspielen, der dasselbe `APPLICATION_SECRET` und `APPLICATION_SALT` verwendet. Um Daten in eine andere Instanz zu übertragen, gibt es den Export (siehe „Daten exportieren“).

Wichtig: Backups und Exporte enthalten nur die Datenbank. Dateien wie Dokumente oder Fotos sind nicht enthalten, der Ordner `/app/files` muss deshalb zusätzlich gesichert werden.

Backups herunterladen, wiederherstellen und Daten exportieren dürfen nur Administratoren.

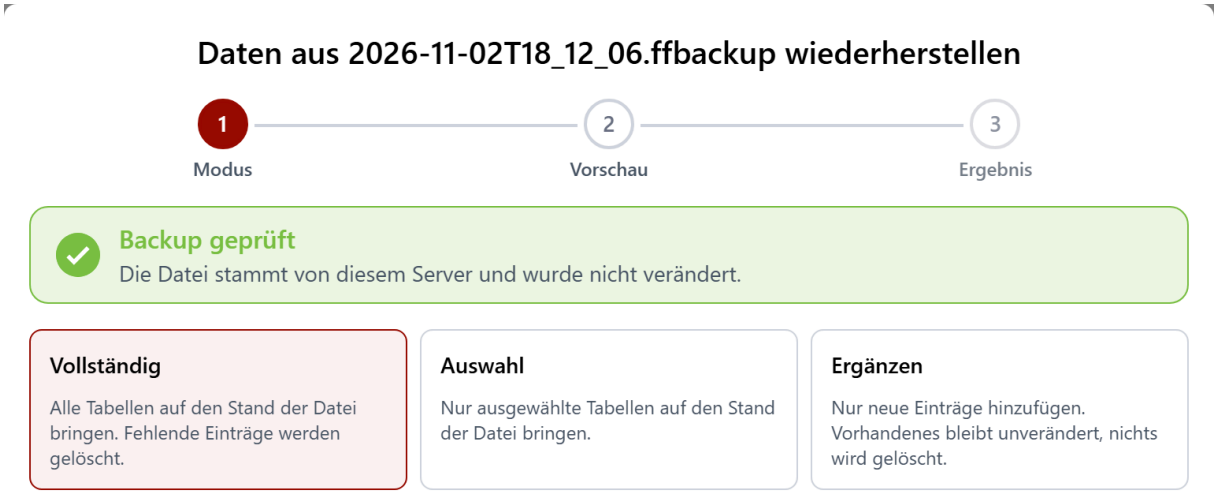


Backups werden in einem festgelegten Intervall erstellt. Das kleinste Intervall beträgt täglich. Zusätzlich kann konfiguriert werden, wie viele Backups parallel zur Verfügung stehen sollen. Standardmäßig ist eine Anzahl von 7 eingestellt, d.h. die gespeicherten Backups reichen immer eine Woche in die Vergangenheit. Die Anzahl bezieht sich dabei nicht auf die Anzahl an Backups im System, sondern auf die Anzahl Tage, an denen ein Backup gemacht wurde. Heißt, dass 2 Backups des selben Tages als eines gewertet wird.
Das Intervall und die Anzahl and Backups, welche parallel gespeichert sein sollen, können in den Einstellungen Abschnitt 4.11 gesetzt werden.

Eine Funktion, die mit den Backups eingeführt wurde, ist `AUTO RESTORE`.
`AUTO RESTORE` ermöglicht die automatische Wiederherstellung des letzten Backups - sofern vorhanden - wenn die Datenbank beim Start des Servers leer ist. Verwendet wird das neueste Backup, das sich prüfen lässt. Exporte und veränderte Dateien werden übersprungen.

4.16.1 Backup wiederherstellen

Das Symbol neben einem Backup öffnet einen Assistenten. Er prüft zuerst die Datei und zeigt das Ergebnis an:



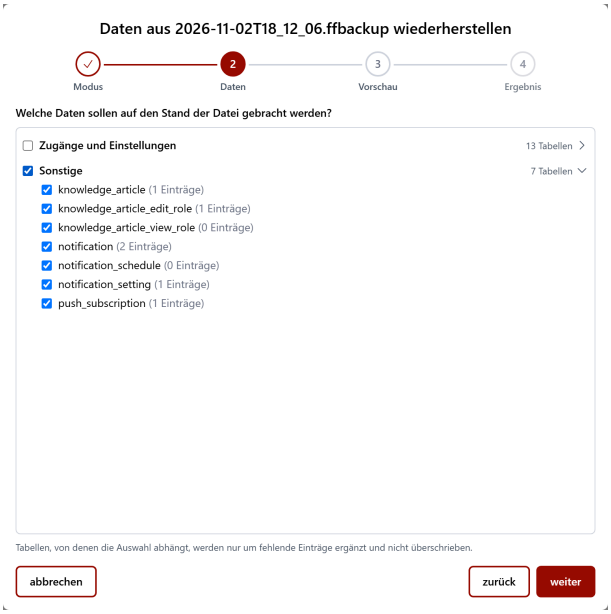
Anzeige	Bedeutung	Einspielen durch
Backup geprüft	Die Datei stammt von diesem Server und wurde nicht verändert.	Administratoren
Export geöffnet	Das Passwort eines Exports passt. Das sagt nichts darüber aus, wer die Datei erstellt hat.	Administratoren
Passwort nötig	Die Datei ist ein Export und muss erst mit ihrem Passwort geöffnet werden.	niemand
Anderer Schlüssel	Die Datei wurde auf einem Server mit anderem Secret oder Salt erstellt.	niemand
Datei verändert	Die Datei wurde nach dem Erstellen verändert.	niemand
Altes Format, nicht prüfbar	Die Herkunft einer Datei aus einer früheren Version lässt sich nicht zuverlässig prüfen.	nur Owner, nach Bestätigung

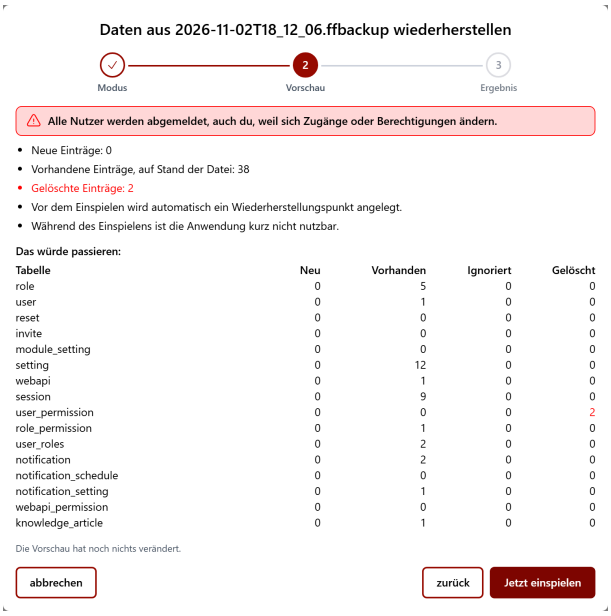
Backups liegen je nach Alter in verschiedenen Formaten vor. Aktuelle Backups haben die Endung .ffbackup. Ältere .json-Dateien lassen sich weiterhin einspielen, sofern sie bereits eine Signatur enthalten (Format 2). Noch ältere .json-Dateien (Format 1) werden abgelehnt.
Danach wird festgelegt, wie die Daten übernommen werden:

- **Vollständig:** Alle Tabellen werden auf den Stand der Datei gebracht. Einträge, die in der Datei fehlen, werden gelöscht.
- **Auswahl:** Nur die ausgewählten Tabellen werden auf den Stand der Datei gebracht. Tabellen, von denen die Auswahl abhängt, werden nur um fehlende Einträge ergänzt.
- **Ergänzen:** Es werden nur neue Einträge hinzugefügt. Vorhandenes bleibt unverändert, gelöscht wird nichts.

Eine Auswahl, die Zugänge und Einstellungen enthält, darf nur der Owner einspielen. Bei Exporten steht „Vollständig“ nicht zur Verfügung.

Bei **Auswahl** und **Ergänzen** folgt die Auswahl der Daten. Die Tabellen sind nach Bereichen gruppiert und lassen sich aufklappen, um einzelne Tabellen zu wählen.





Vor dem Einspielen zeigt eine **Vorschau** je Tabelle, was passieren würde. Die Vorschau selbst verändert nichts. Sie zeigt auch, ob alle Nutzer abgemeldet werden. Das passiert nur, wenn sich Nutzer, Rollen, Berechtigungen, Passkeys oder Sitzungen ändern. Reine Einstellungen oder Stammdaten melden niemanden ab.

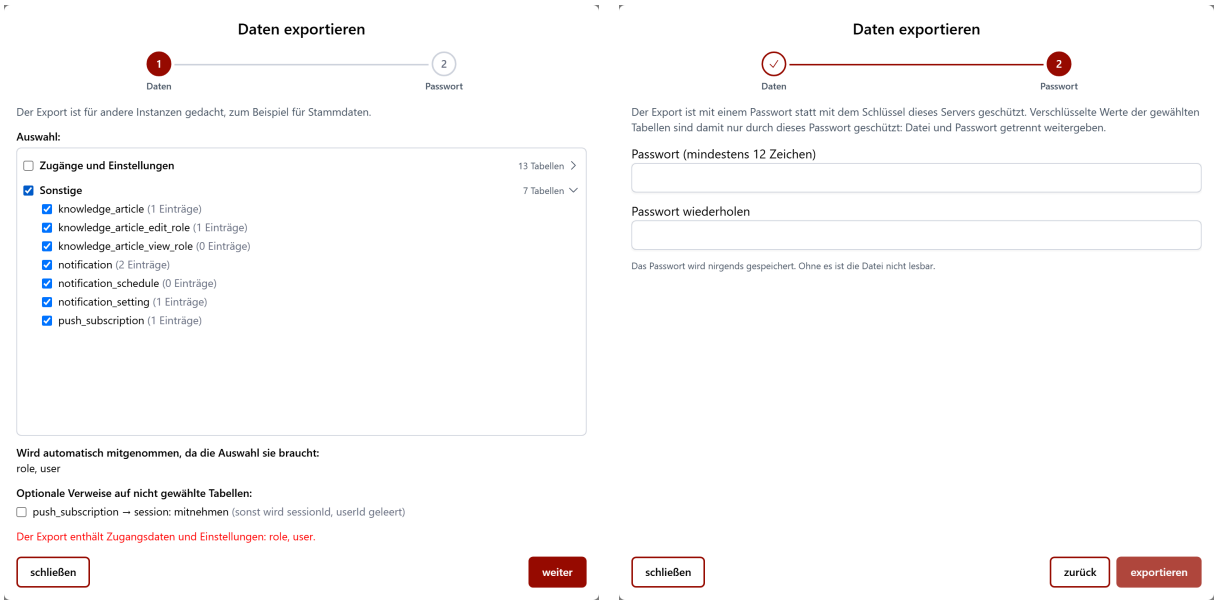
Mit `Jetzt einspielen` wird automatisch ein Wiederherstellungspunkt angelegt, mit dem sich der Vorgang zurücknehmen lässt. Während des Einspielens ist die Anwendung kurz nicht nutzbar. Das Ergebnis wird anschließend in derselben Form angezeigt.

Das sollte man beachten:

- **Zugangsdaten bleiben erhalten:** Benutzername, Passwort bzw. TOTP, E-Mail-Adresse und Owner-Status vorhandener Nutzer ändert eine Wiederherstellung nie. Solange es bereits Nutzer gibt, wird auch kein neuer Nutzer zum Owner.
- **Neuere Konten verlieren Rechte:** Ein Nutzer, der nach dem Backup angelegt wurde, bleibt bei „Vollständig“ bestehen, verliert aber Rollen, Berechtigungen und Passkeys. Das gilt auch für einen neuen Administrator, der die Wiederherstellung selbst ausführt. Die Vorschau zeigt das als gelöschte Einträge der Tabellen `user_roles`, `user_permission` und `passkey`.

- **Ignorierte Einträge:** Ist ein eindeutiger Wert schon vergeben, etwa der Name einer neu angelegten Rolle, bleibt der vorhandene Eintrag bestehen und der aus der Datei wird ignoriert. Vorschau und Ergebnis zeigen die Anzahl je Tabelle.
- **Nicht lesbare Werte:** Lässt sich ein verschlüsselter Wert der Datei nicht lesen, wird er gemeldet. Vorhandene Einträge behalten dann ihren Wert, neue bleiben an dieser Stelle leer.

4.16.2 Daten exportieren

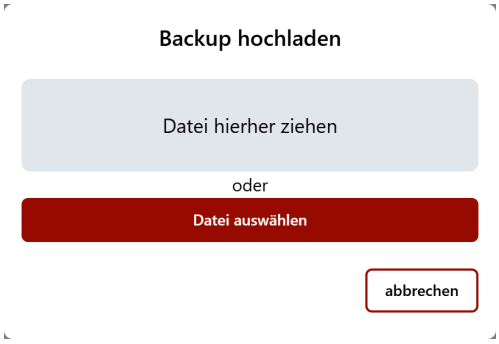


Ein Export überträgt Daten in eine andere Instanz, zum Beispiel vorbereitete Stammdaten als Ausgangspunkt für eine neue Installation. Über `Daten exportieren` werden die gewünschten Daten ausgewählt und ein Passwort mit mindestens 12 Zeichen vergeben. Die Datei wird direkt heruntergeladen und bleibt nicht auf dem Server.

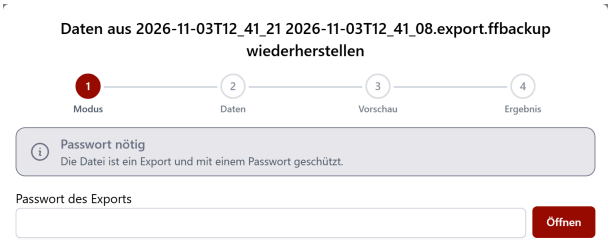
- Tabellen, die die Auswahl zwingend braucht, werden automatisch mitgenommen. Optionale Verweise auf nicht gewählte Tabellen können mitgenommen werden, sonst werden sie geleert.
- Exporte, die Zugänge und Einstellungen enthalten, darf nur der Owner erstellen und einspielen.

Achtung: Ein Export ist nur durch sein Passwort geschützt, auch nachdem Secret oder Salt des Servers gewechselt wurden. Datei und Passwort sollten getrennt weitergegeben und wie Zugangsdaten behandelt werden. Das Passwort wird nirgends gespeichert, ohne es ist die Datei nicht lesbar.

4.16.3 Backup hochladen



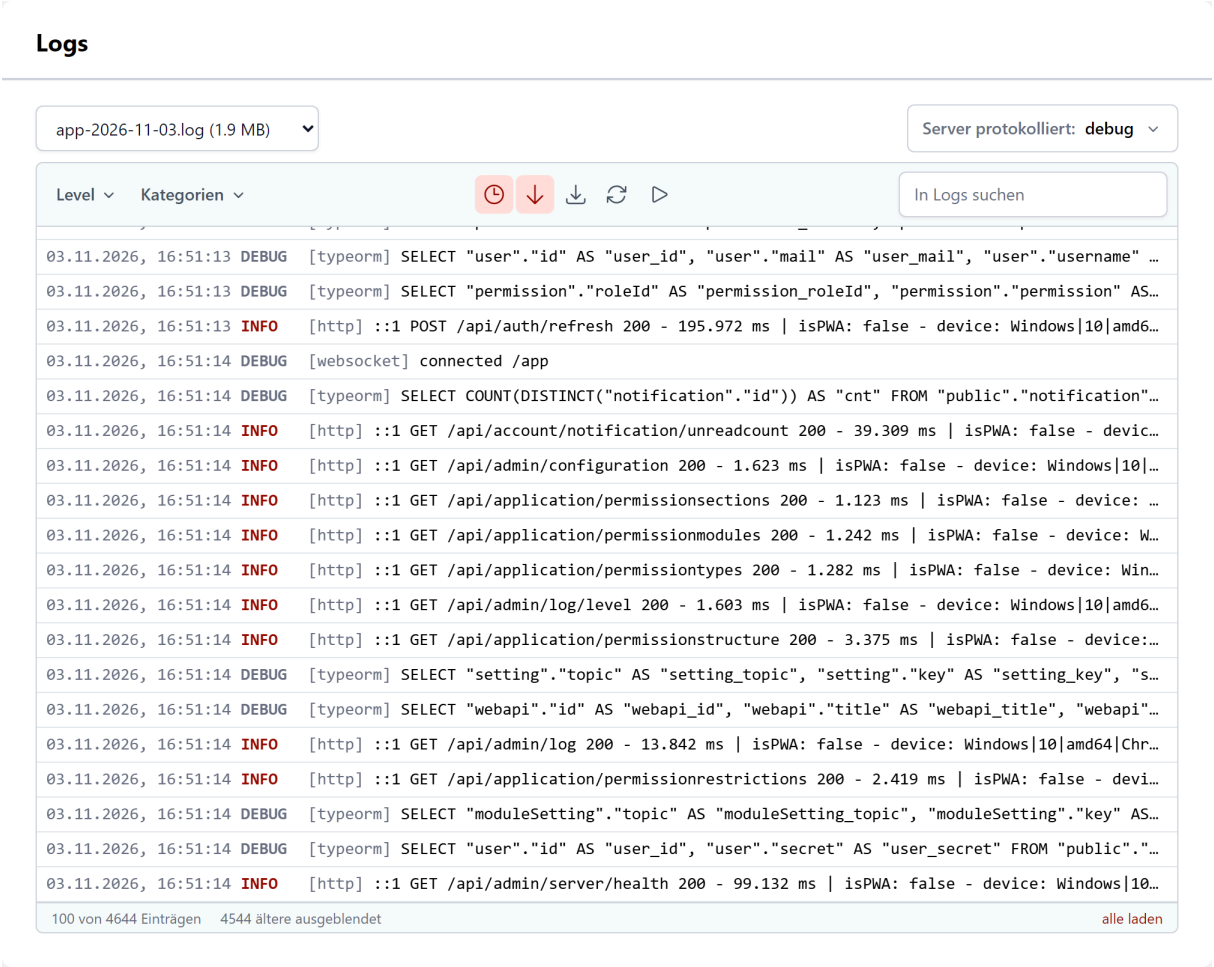
Heruntergeladene Backups und Exporte können jederzeit hochgeladen und im Admin gespeichert werden. Im Gegensatz zu den automatisch erstellten Backups sind die Slots nicht begrenzt. Es können also beliebig viele Dateien hochgeladen und gespeichert werden.
Eine Datei darf standardmäßig bis zu 200 MB groß sein, die Grenze lässt sich mit `BACKUP_UPLOAD_MAX_SIZE` anpassen. Dateien anderer Anwendungen werden abgelehnt.



Export einspielen: Die Datei hochladen, das Symbol zum Wiederherstellen wählen und mit dem Passwort öffnen. Anschließend werden die Daten üblicherweise mit „Ergänzen“ übernommen.

Instanz umziehen: Für einen vollständigen Umzug werden `APPLICATION_SECRET` und `APPLICATION_SALT` auf dem neuen Server unverändert übernommen und ein Backup zusammen mit dem Ordner `/app/files` übertragen.

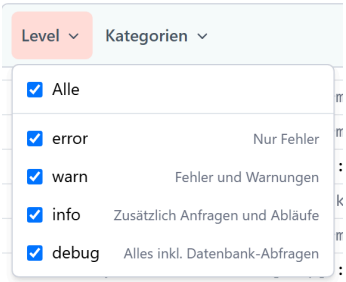
4.17 Logs



Unter `Verwaltung > Logs` können die Log-Dateien des Servers direkt in der Anwendung eingesehen werden, ohne Zugriff auf den Server oder den Docker-Container zu benötigen. Wie die Log-Dateien geschrieben und aufbewahrt werden, ist unter Logging (Abschnitt 2.8) beschrieben.

Für die Ansicht wird die Leseberechtigung des Moduls `Logs` im Abschnitt Verwaltung benötigt. Da die Logs technische Details wie IP-Adressen und Gerätenamen der Anfragen enthalten, sollte diese Berechtigung nur an Nutzer vergeben werden, die den Betrieb der Anwendung betreuen.

4.17.1 Ansicht



Über die Auswahl oben links wird die anzuzeigende Log-Datei gewählt. Neben dem Dateinamen ist die Größe der Datei angegeben. Angezeigt werden zunächst die neuesten Einträge der Datei, ältere Einträge können über `alle laden` nachgeladen werden. Ein Klick auf einen Eintrag zeigt den vollständigen Text sowie zusätzliche Details an.

Die Einträge lassen sich wie folgt eingrenzen und darstellen:

- **Level:** Anzeige nur bestimmter Stufen, z.B. ausschließlich Fehler und Warnungen.
- **Kategorien:** Anzeige nur bestimmter Bereiche des Servers, z.B. API-Anfragen (`http`) oder Datenbank (`typeorm`).
- **Suche:** Volltextsuche innerhalb der Einträge.
- **Zeitstempel:** Ein- und Ausblenden des Zeitpunkts der Einträge.
- **Neuesten Einträgen folgen:** Springt an das Ende der Datei.
- **Herunterladen:** Lädt die ausgewählte Datei vollständig herunter, z.B. um sie bei einer Fehlermeldung weiterzugeben.
- **Neu laden:** Lädt die Datei erneut vom Server.
- **Live:** Neue Einträge werden direkt am Ende angezeigt, ohne die Datei neu laden zu müssen.

4.17.2 Protokollierung ändern



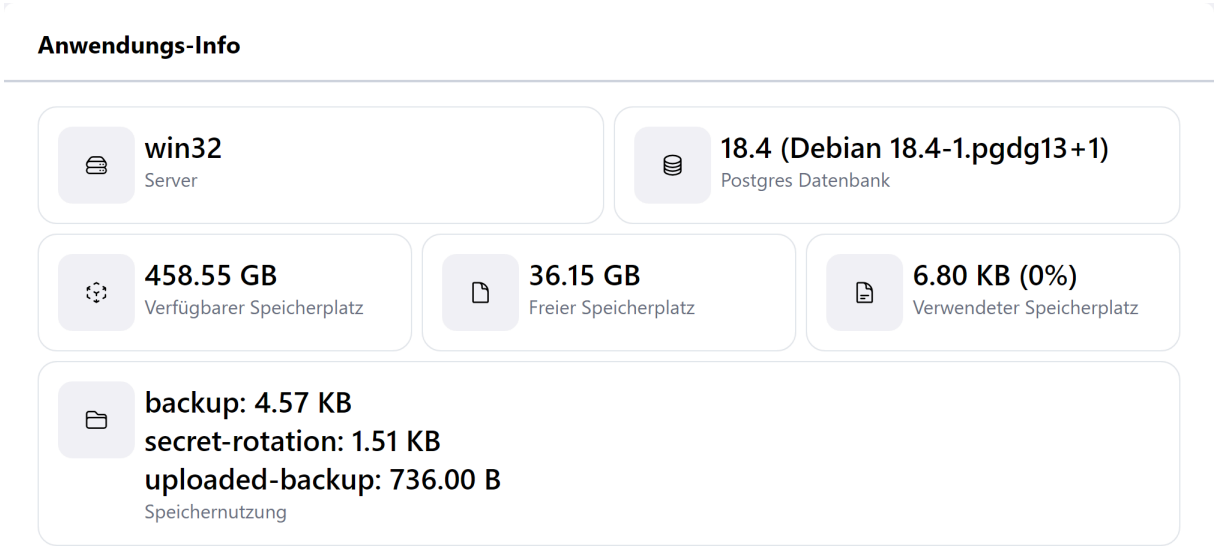
Oben rechts wird angezeigt, mit welchem Detailgrad der Server aktuell protokolliert. Mit der Berechtigung zum Bearbeiten des Moduls `Logs` kann dieser Wert geändert werden, z.B. kurzzeitig auf `debug`, um einen Fehler nachvollziehen zu können.

Die Änderung gilt sofort für alle neuen Einträge, aber nur bis zum nächsten Neustart des Servers. Solange ein abweichender Wert gesetzt ist, wird dieser als `temporär` gekennzeichnet und kann über das Menü auf den konfigurierten Wert (`LOG_LEVEL`) zurückgesetzt werden. Ein dauerhafter Wechsel erfolgt ausschließlich über die Umgebungsvariable `LOG_LEVEL` (siehe Abschnitt 2.5).

4.18 Anwendungsinformationen und Versionsmanagement

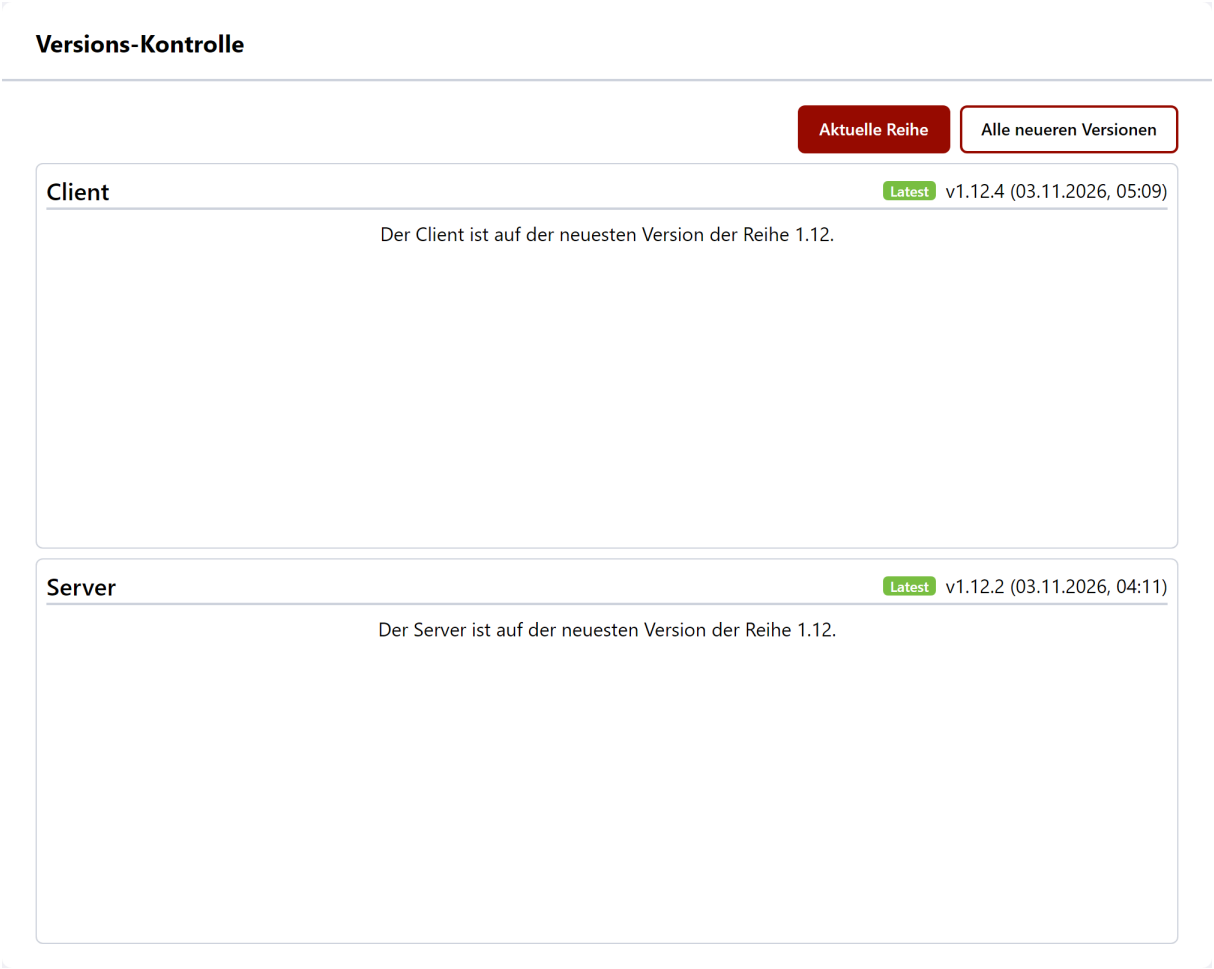
Die Anwendung stellt in der Oberfläche Informationen zum Server, der Datenbank und dem Speicherbedarf sowie der aktuellen Version und möglichen verfügbaren Updates zur Verfügung.

4.18.1 Anwendungsinformationen



Die Anwendungsinformationen geben einen aktuellen Status der Anwendung und dem Server aus. Angezeigt werden das Betriebssystem des Servers und die aktuelle Version der Datenbank. Die Version der Datenbank bestimmt unter anderem die Generierung von Ids.
Weiterhin wird der aktuelle Speicherbedarf und der verfügbare oder der zugewiesene Speicherplatz angezeigt. Die Speicherinformationen werden auch mit der Nutzung der einzelnen Module im Dateisystem angegeben.

4.18.2 Versionsmanagement



Die Oberfläche stellt außerdem Informationen zur aktuell installierten Version auf dem Server bereit. Die lokal installierte PWA-Version kann sich von der Version auf dem Server unterscheiden. Die Anwendung selbst stellt Versionsinformationen unter `/_clientinfo.json` bereit. Diese Version wird auch für die Prüfung verfügbarer Updates verwendet.

Die Versions-Kontrolle zeigt für Client und Server jeweils die installierte Version mit Veröffentlichungsdatum sowie die verfügbaren neueren Versionen an. Die Informationen zu den neuen Versionen stammen aus den Git-Releases der Anwendungen.

Neben der Version zeigt eine Markierung den Support-Status der installierten Reihe (Abschnitt 2.12.1):

- **Latest:** Die installierte Version gehört zur neuesten Reihe.
- **Supported:** Die Reihe wird weiterhin mit Fehler- und Sicherheitskorrekturen versorgt. Zusätzlich werden die neueste Version der Reihe und die neueste Version insgesamt angegeben.
- **EOL:** Die Reihe wird nicht mehr unterstützt und erhält keine Sicherheitskorrekturen mehr. Ein rot hinterlegter Hinweis empfiehlt ein Update auf eine unterstützte Version.

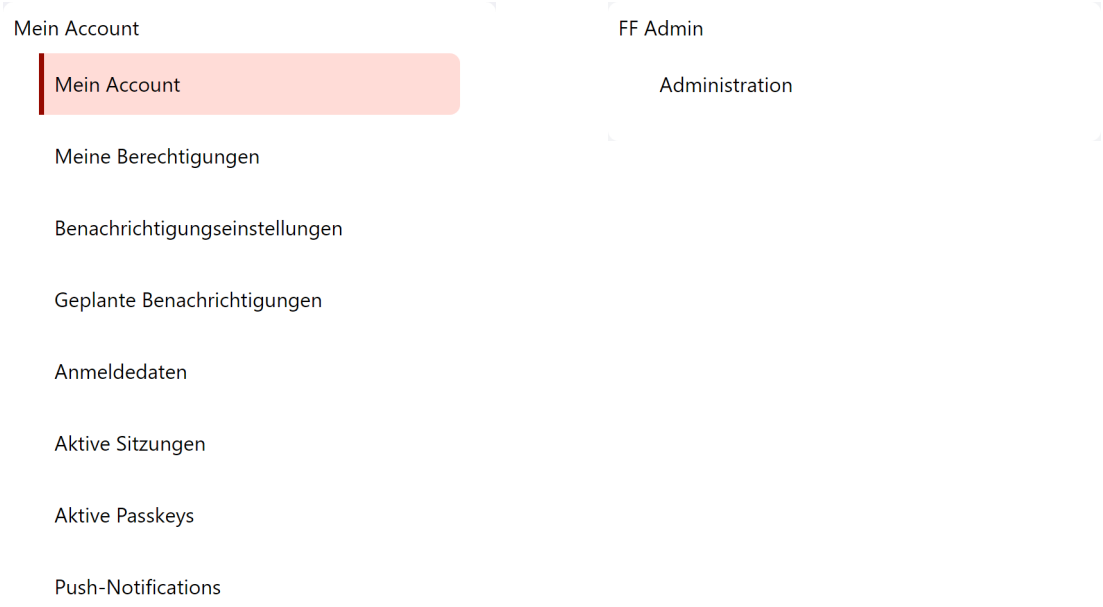
Der Support-Status wird aus der zentralen `support.json` bezogen. Ist diese nicht erreichbar, entfällt die Markierung.
Über die Auswahl oberhalb der Liste lässt sich festlegen, welche neueren Versionen angezeigt werden:

- **Aktuelle Reihe** (Standard): nur neuere Patch-Versionen der installierten Reihe, also Updates ohne neue Funktionen.
- **Alle neueren Versionen:** auch Versionen neuerer Reihen.

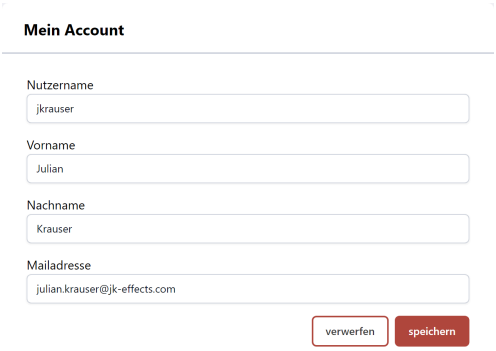
Die Versionen werden nach Versionsnummer absteigend sortiert, die neueste Version steht oben.

5 Benutzerbereich und Accounteigenschaften

Der Accountbereich ist für jeden Benutzer durchgehend zugänglich und gliedert sich in allgemeine Kontoinformationen sowie einen Administrations- bzw. Eigentümerbereich.



5.1 Zugang und Anmeldedaten



Die Kontoinformationen sind in zwei separate Bereiche unterteilt: Der Bereich **Mein Account** zeigt allgemeine Accountdaten wie Benutzername und E-Mail-Adresse an. Jeder Nutzer kann seinen Benutzernamen und seine E-Mail-Adresse nach einer erneuten Anmeldung selbst ändern, Administratoren können diese Daten auch für andere Nutzer bearbeiten. Im Bereich **Anmeldedaten** können die Einstellungen für die Authentifizierung verwaltet werden - entweder das zeitbasierte Einmalpasswort (TOTP) oder das klassische Passwort.

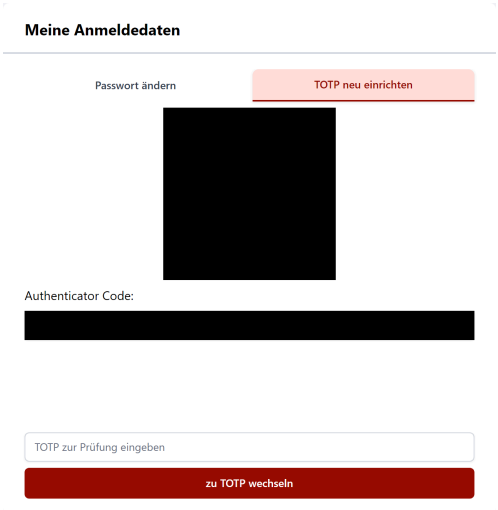
5.1.1 Anmeldeoptionen

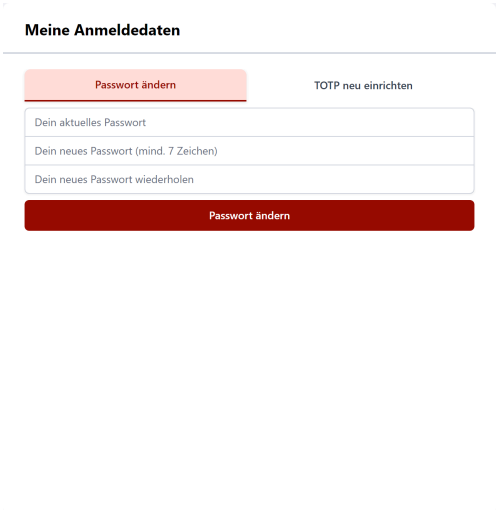
Nutzer können sich entweder mit einem zeitbasierten Einmalpasswort (TOTP) oder einem klassischen Passwort anmelden. Je nach gewählter Methode stehen folgende Optionen zur Verfügung:

- **TOTP-Anmeldung:** Der TOTP-QR-Code kann erneut angezeigt werden, um die Authentifizierung auf einem weiteren Gerät einzurichten. Alternativ kann zu einer Passwort-basierten Anmeldung gewechselt werden.
- **Passwort-Anmeldung:** Das Passwort kann geändert oder zu einer TOTP-basierten Anmeldung gewechselt werden.
- **Passkey-Anmeldung:** Passkeys können als alternative Loginmethode erstellt werden und erleichtern den Login. Passkeys ersetzen allerdings nicht das gesetzte Passwort bzw. das eingerichtete TOTP. (Abschnitt 5.5)
- **QRCode-Anmeldung:** Auch kann sich auf anderen Geräten via QR-Code angemeldet werden. Dazu muss über den Knopf `Login mit QRCode` in der Loginmaske der QRCode angezeigt werden. In der App selbst kann der Code über das User-Menü gescannt werden.

5.1.2 Änderung der Anmeldedaten

Unter dem Menüpunkt Anmeldedaten kann der QR-Code für das aktuelle TOTP abgerufen werden. Auch können unter diesem Menüpunkt die Anmeldedaten geändert oder zwischen Anmeldung mittels TOTP bzw. Passwort gewechselt werden.
Um unberechtigten Zugriff zu verhindern wird bei Aufruf der Seite nochmals eine Authentifizierung angefordert.
Werden Passwort oder Anmeldeart geändert, bleibt nur die aktuelle Sitzung bestehen, alle anderen Geräte werden abgemeldet. Nach dem Zurücksetzen des Zugangs per E-Mail enden alle Sitzungen.
Verwalteten Konten (siehe „Verwaltete Konten“) stehen TOTP und das Zurücksetzen per E-Mail nicht zur Verfügung.





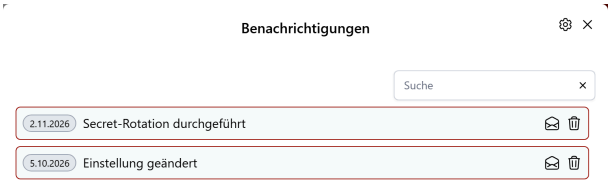
5.2 Benachrichtigungen

Benutzer können Benachrichtigungen zu Vorgängen der Anwendung einstellen und erhalten. Benachrichtigungen werden immer gespeichert und können über die Glocke in der Navigationsleiste angezeigt werden.
Für die Module, welche Benachrichtigungen versenden, kann der Versandweg eingestellt werden. Zur Verfügung stehen aktuell:

- **In-App**: Nachrichten werden direkt in der Anwendung angezeigt, wenn diese geöffnet ist
- **E-Mail**: Benachrichtigungen werden per E-Mail versendet
- **Web-Push**: Benachrichtigungen werden direkt auf das Gerät gesendet (siehe Abschnitt 5.3)

Einstellungen zu den Benachrichtigungen können nur für Module vorgenommen werden, auf die der Account Zugriff hat.

5.2.1 Benachrichtigungsanzeige



In der Benachrichtigungsanzeige werden alle Benachrichtigungen zentral aufgelistet. Für jede Benachrichtigung sind folgende Informationen ersichtlich:

- Der Nachrichteninhalt
- Das sendende Modul
- Die verwendeten Zustellwege (In-App, E-Mail, Web-Push)

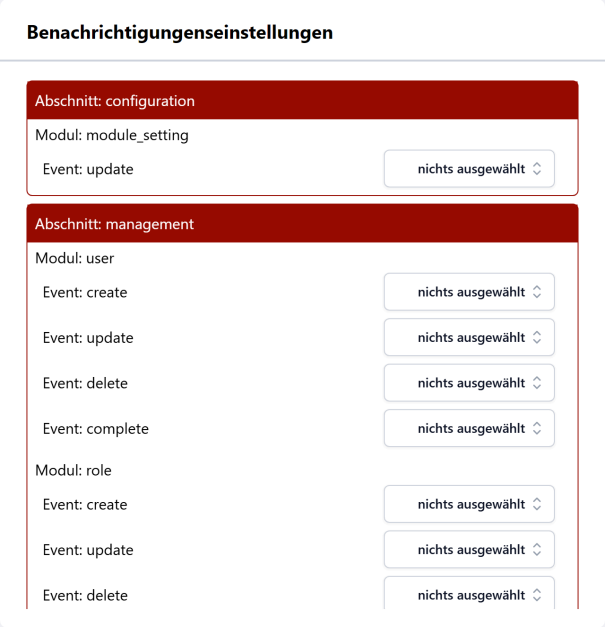
Hier können versendete Benachrichtigungen jederzeit nachgelesen werden.

5.2.2 Benachrichtigungseinstellungen

Die Benachrichtigungseinstellungen sind über das Zahnrad in der Nachrichtenanzeige oder im Usermenü unter „Mein Account“ erreichbar.
Für jedes Modul und Event können die Versandwege individuell konfiguriert werden:

- **In-App**: Direkte Anzeige in der Anwendung
- **E-Mail**: Versand per E-Mail
- **Web-Push**: Versand als Push-Benachrichtigung auf das Gerät (siehe Abschnitt 5.3)

Es können mehrere Versandarten gleichzeitig aktiviert werden.



Benachrichtigungen werden für folgende Ereignisse versendet:

- **create**: Neues Element wurde erstellt
- **update**: Element wurde aktualisiert
- **delete**: Element wurde gelöscht
- **complete**: Vorgang wurde abgeschlossen (Bsp.: Newsletter wurde fertig versandt)
- **remind**: Erinnerung an anstehende Aufgabe (Bsp.: Prüfung ist fällig)

5.2.3 Geplante Benachrichtigungen

Neben den ereignisbasierten Benachrichtigungen können Benutzer geplante Benachrichtigungen konfigurieren. Dabei handelt es sich um regelmäßig versendete Zusammenfassungen oder Erinnerungen, deren Inhalt dynamisch berechnet wird.
Die verfügbaren Benachrichtigungstypen werden von der Anwendung bereitgestellt und können je nach Instanz variieren (z.B. Erinnerungen an fällige Geräteprüfungen oder Führerscheinkontrollen).

Konfiguration

Für jeden verfügbaren Benachrichtigungstyp kann der Benutzer folgende Einstellungen vornehmen:

- **Aktiv/Inaktiv**: Einzelne Benachrichtigungen können ein- oder ausgeschaltet werden
- **Wochentage**: Auswahl der Tage, an denen die Benachrichtigung versendet werden soll (Sonntag bis Samstag)
- **Uhrzeit**: Festlegung der Sendezeit (Format HH:MM)
- **Kanäle**: Auswahl der Zustellwege — es können mehrere gleichzeitig aktiviert werden:
  - In-App: Anzeige in der Anwendung
  - E-Mail: Versand per E-Mail
  - Web-Push: Push-Benachrichtigung auf das Gerät (siehe Abschnitt 5.3)

Einige Benachrichtigungstypen sind als *täglich* markiert. Bei diesen Typen entfällt die Auswahl der Wochentage, da der Inhalt tagesaktuell ist und bei Auslassung eines Tages nicht nachgeholt werden kann (z.B. „heute fällige Prüfungen“). Für diese Benachrichtigungen kann nur die Sendezeit und die Kanäle konfiguriert werden.

Versandlogik
Das System prüft alle 5 Minuten, ob geplante Benachrichtigungen fällig sind. Eine Benachrichtigung gilt als fällig, wenn:
- sie aktiv ist
- der aktuelle Wochentag in der Konfiguration enthalten ist
- die konfigurierte Uhrzeit erreicht oder überschritten ist
- sie am heutigen Tag noch nicht versendet wurde

Ist eine Benachrichtigung fällig, wird der zugehörige Inhalt berechnet. Ergibt die Berechnung, dass es nichts zu melden gibt (z.B. keine fälligen Prüfungen), wird keine Benachrichtigung versendet. Andernfalls wird die Nachricht über alle konfigurierten Kanäle zugestellt.
Pro Benachrichtigungstyp und Benutzer wird maximal eine Nachricht pro Tag versendet, unabhängig davon, wie oft das System prüft.

5.3 Push-Benachrichtigungen

Push-Benachrichtigungen ermöglichen es, Mitteilungen direkt auf das Gerät zu erhalten — auch wenn die Anwendung nicht geöffnet ist. Sie funktionieren unabhängig davon, ob die App als PWA installiert ist oder im Browser verwendet wird.

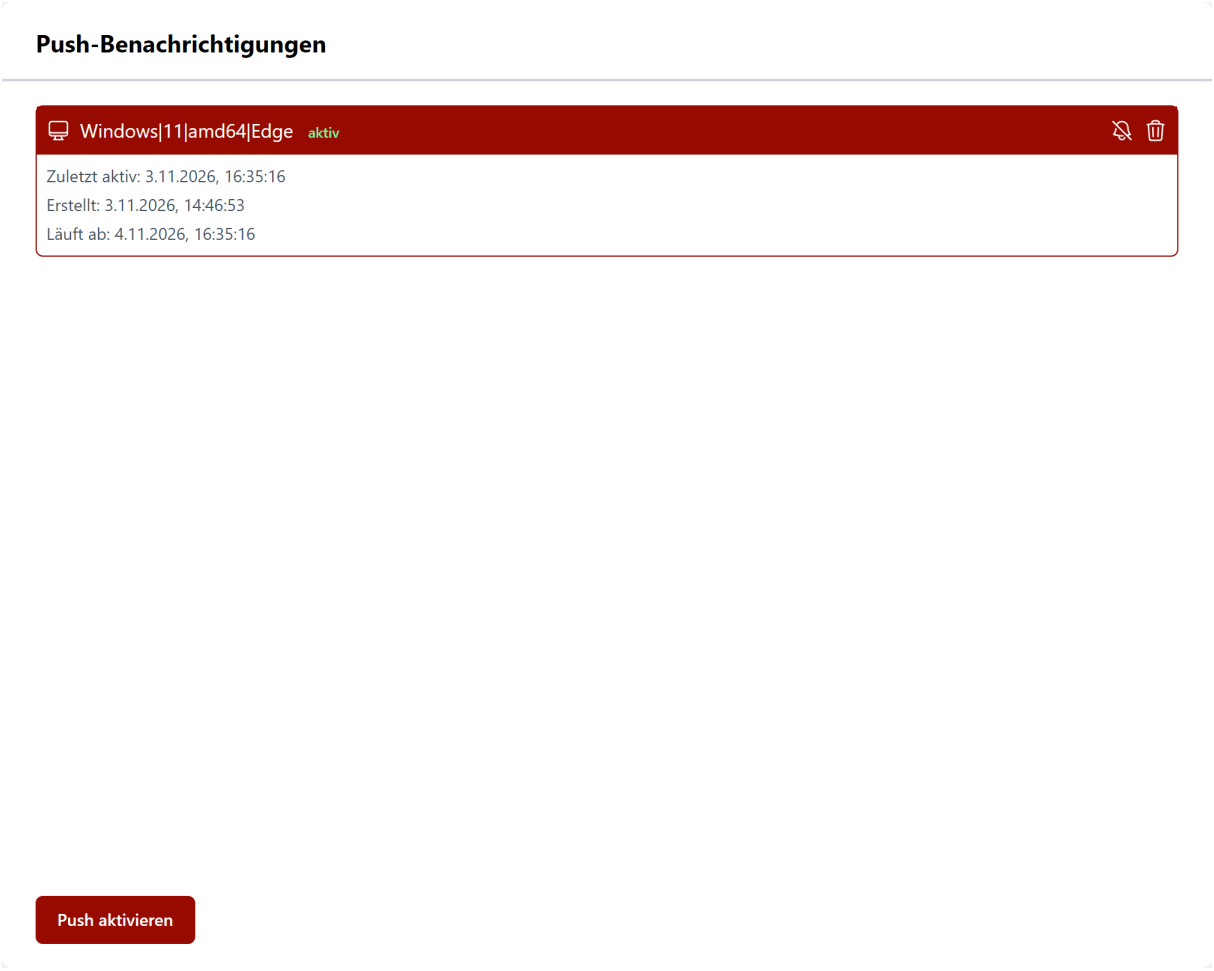
5.3.1 Funktionsweise

Push-Benachrichtigungen nutzen den Web-Push-Standard. Dabei wird eine verschlüsselte Verbindung zwischen dem Server und dem Gerät des Nutzers aufgebaut. Der Server kann darüber Nachrichten senden, die das Betriebssystem als Benachrichtigung anzeigt.
Voraussetzungen:
- Ein moderner Browser mit Push-Unterstützung (Chrome, Firefox, Edge, Safari 16+)
- Die Berechtigung für Benachrichtigungen muss im Browser erteilt sein
- Ein aktiver Service Worker muss registriert sein (wird automatisch verwaltet)

5.3.2 Push aktivieren

Die Aktivierung erfolgt über den Accountbereich unter `Push-Benachrichtigungen`. Nach Klick auf `Push aktivieren` fragt der Browser nach der Berechtigung, Benachrichtigungen anzeigen zu dürfen. Wird die Berechtigung erteilt, wird das Gerät automatisch für Push-Benachrichtigungen registriert. Der Eintrag erscheint in der Übersichtsliste.
Hinweis: Push wird auch automatisch bei jeder Anmeldung aktiviert, sofern die Berechtigung bereits erteilt wurde.

5.3.3 Übersicht und Verwaltung



In der Übersicht werden alle registrierten Geräte des Nutzers aufgelistet. Für jeden Eintrag sind folgende Informationen sichtbar:

- **Gerätetyp**: Mobilgerät, Tablet oder Desktop (mit passendem Icon)
- **Status**: aktiv oder inaktiv
- **Zuletzt aktiv**: Zeitpunkt der letzten Nutzung
- **Erstellt**: Zeitpunkt der Registrierung
- **Ablaufdatum**: Wann die Subscription automatisch ungültig wird

5.3.4 Deaktivieren und Löschen

Für jede registrierte Push-Subscription stehen zwei Aktionen zur Verfügung:
Deaktivieren (durchgestrichene Glocke):

- Die Subscription wird als inaktiv markiert
- Es werden keine Push-Nachrichten mehr an dieses Gerät gesendet

• Bei der nächsten Anmeldung auf dem betroffenen Gerät wird Push automatisch reaktiviert

Löschen (Mülleimer):

• Die Subscription wird dauerhaft aus der Datenbank entfernt
• Push wird auf dem betroffenen Gerät nicht automatisch reaktiviert
• Der Nutzer muss Push auf dem Gerät manuell erneut aktivieren

Beide Aktionen können remote ausgeführt werden — d.h. von einem anderen Gerät aus.

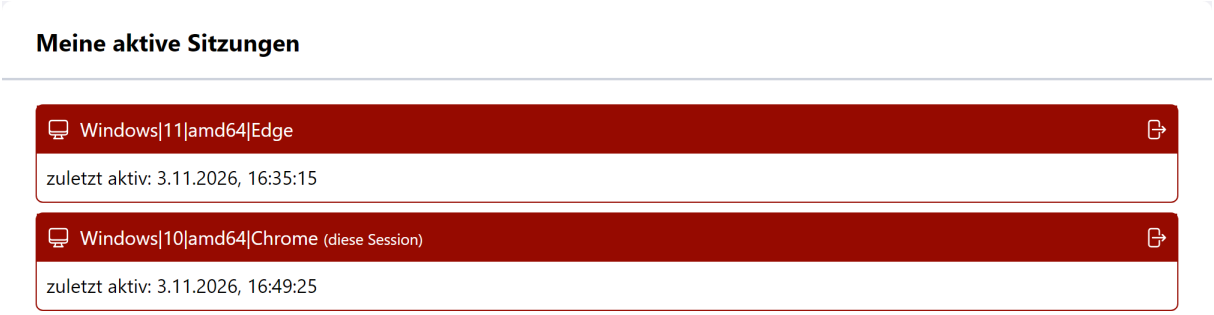
5.3.5 Lebenszyklus einer Push-Subscription

Aktion	Auswirkung
Login	Push wird aktiviert (oder reaktiviert, falls deaktiviert)
Session-Refresh	Ablaufdatum wird verlängert
Logout	Push wird deaktiviert
Remote Deaktivieren	Push wird deaktiviert, nächster Login reaktiviert
Remote Löschen	Push wird entfernt, nur manuelle Neuaktivierung möglich
Ablaufdatum erreicht	Subscription wird beim nächsten Sendeversuch ignoriert
Browser-Berechtigung entzogen	Server erhält Fehlermeldung und deaktiviert automatisch

5.3.6 Mehrere Benutzer auf einem Gerät

Push-Benachrichtigungen unterstützen die Nutzung eines Geräts durch mehrere Personen. Jeder Benutzer hat eine eigene Subscription — die Registrierung eines Nutzers beeinflusst die Push-Einstellungen anderer Nutzer auf demselben Gerät nicht.

5.4 Sessionverwaltung



Die Sessionverwaltung bietet Nutzern umfassende Kontrolle über ihre aktiven Anmeldungen. Mit dieser Funktion können Sie jederzeit einsehen, auf welchen Geräten Ihr Account derzeit aktiv ist und Details zu den entsprechenden Sitzungen abrufen.

5.4.1 Geräteerkennung und Datenschutz

Bei jeder Anmeldung erfasst das System automatisch Geräteinformationen wie Betriebssystem, Browser und Gerätetyp. Diese Daten werden direkt vom Gerät bereitgestellt und ermöglichen eine zuverlässige Geräteidentifikation – ohne dabei Fingerprinting-Techniken zur vollständigen Geräteprofilierung einzusetzen.

5.4.2 Verwaltung aktiver Sessions

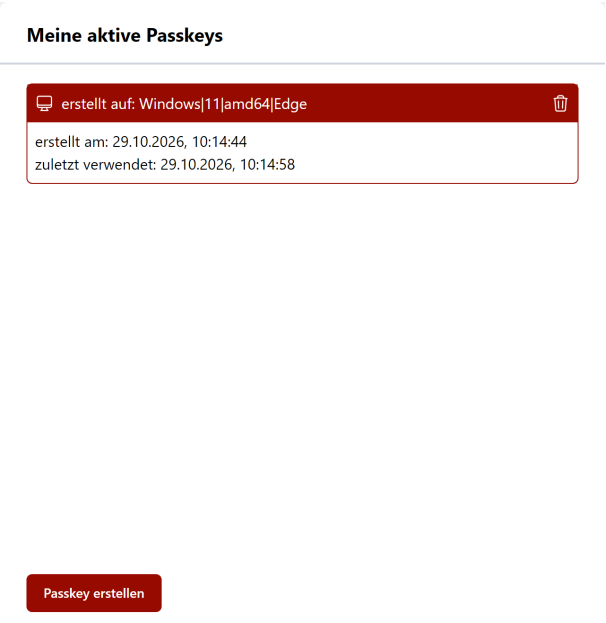
Sie haben die Möglichkeit, Sessions von anderen Geräten aus zu beenden:

- **Ist die App aktiv:** Das betroffene Gerät wird sofort abgemeldet.
- **App ist inaktiv:** Der Zugriff wird beim nächsten Öffnen der Anwendung blockiert - eine erneute Anmeldung ist erforderlich.

Diese Funktion erhöht die Sicherheit Ihres Accounts und gibt Ihnen vollständige Kontrolle über Ihre aktiven Sitzungen.

Sessions enden auch automatisch, zum Beispiel wenn das Passwort geändert oder der Zugang zurückgesetzt wird, das Konto deaktiviert wird oder eine Wiederherstellung die Zugänge ändert. Mit einer Session endet auch der Push-Empfang des jeweiligen Geräts.

5.5 Passkeys



Passkeys sind eine sichere Alternative zu Passwörtern. Sie basieren auf der FIDO2-Technologie und ermöglichen es einem Nutzer, sich mit biometrischen Daten (Fingerabdruck, Gesichtserkennung) oder einer PIN zu authentifizieren, ohne sich ein komplexes Passwort merken zu müssen.

Die Passkeys werden direkt auf dem Gerät gespeichert und ersetzen klassische Zugangsdaten komplett. Beim Login erkennt die Anwendung automatisch, dass ein Passkey vorhanden ist, und bietet diese Option direkt an - dadurch muss sich ein Nutzer nicht mehr an Passwörter erinnern.

Vorteile von Passkeys

- **Einfacherer Login**: Statt Passwort eingeben reicht Fingerabdruck oder Gesichtserkennung
- **Höhere Sicherheit**: Passkeys sind resistent gegen Phishing und Brute-Force-Attacken

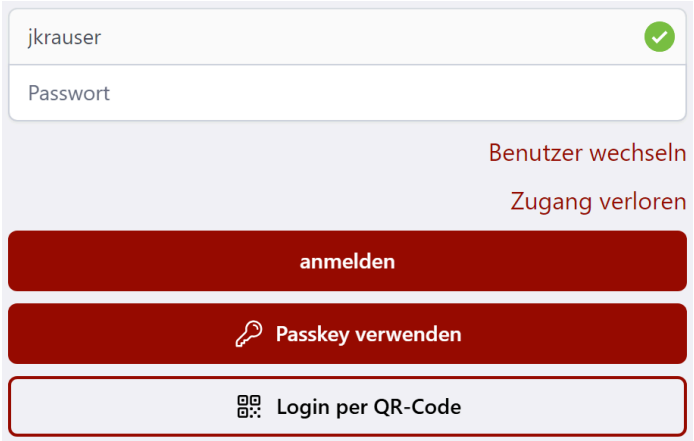
- **Gerätübergreifend**: Dein Passkey wird auf mehreren Geräten synchronisiert und kann auch auf anderen Geräten verwendet werden

5.5.1 Erstellung eines Passkeys

Ein Passkey kann nur erstellt werden, wenn der Nutzer bereits angemeldet ist. In den Kontoeinstellungen unter `Meine aktiven Passkeys` lässt sich ein neuer Passkey hinzufügen. Zum Erstellen und Löschen eines Passkeys ist eine erneute Anmeldung nötig.
Nach Klick auf `Passkey erstellen` führt das Gerät durch den Erstellungsprozess und fordert zur Bestätigung eine PIN oder biometrische Authentifizierung an.
Passkeys können auch über ein anderes Gerät erstellt werden. Nach Auswahl dieser Option wird ein QR-Code angezeigt, den man mit einem anderen Gerät scannen kann. Dadurch werden beide Geräte synchronisiert und der Passkey steht auf beiden zur Verfügung.

5.5.2 Login mit Passkeys

Passkeys können auf dem Gerät direkt nach der Erstellung verwendet werden. Ist der Nutzername schon Hinterlegt, wird ein Knopf unterhalb des Login-Formulars angezeigt:



Sollte noch kein Nutzername eingegeben sein, wird ein vorhandener Passkey zur Auswahl angezeigt, wenn ins Feld „Benutzername“ geklickt wird.
Bei Auswahl eines Passkeys erfolgt eine Bestätigung durch den Nutzer. Alternativ kann ein anderes Gerät zum Login verwendet werden: Nach Auswahl dieser Option wird ein QR-Code angezeigt, über den die Anmeldung auf einem anderen Gerät erfolgen kann.



5.6 Übertragung Administration

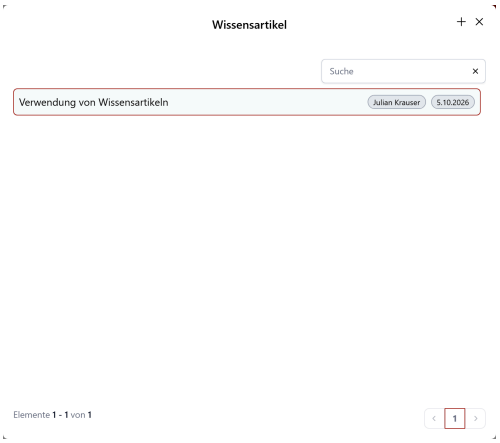


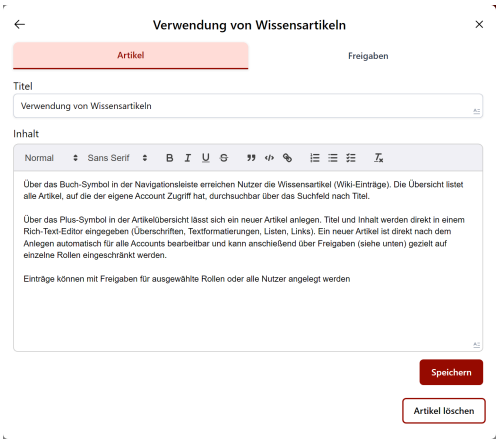
Bei der Übertragung der Administration wird ein anderer Nutzer zum Owner ernannt und erhält damit Vollzugriff auf die gesamte Anwendung, unabhängig von einzelnen Berechtigungen.
Der bisherige Administrator verfügt danach nur noch über die Berechtigungen, die ihm durch seine Rollen oder direkten Zuweisungen zuteil werden. Da der `Owner`-Status stets nur einem einzelnen Nutzer zugewiesen ist, verliert der bisherige Administrator diesen automatisch mit der Übertragung.
Der Menüpunkt **Administration** ist ausschließlich für den aktuellen Owner sichtbar und erreichbar - alle anderen Nutzer sehen diesen Punkt im Accountbereich nicht, unabhängig von ihren sonstigen Berechtigungen.

5.6.1 Durchführung

Zur Übertragung wird der gewünschte neue Owner über die Nutzersuche ausgewählt und die Auswahl mit **übertragen** bestätigt. Dafür ist eine erneute Anmeldung erforderlich (wie z.B. bei der Änderung der Anmeldedaten).
Wichtig: Die Übertragung wirkt sofort und kann nicht direkt rückgängig gemacht werden. Der bisherige Administrator kann die Administration nur zurückerhalten, wenn der neue Owner sie ihm erneut überträgt.

5.7 Wissensartikel





Über das Buch-Symbol in der Navigationsleiste erreichen Nutzer die Wissensartikel (Wiki-Einträge). Die Übersicht listet alle Artikel, auf die der eigene Account Zugriff hat, durchsuchbar über das Suchfeld nach Titel.

5.7.1 Anlegen und Bearbeiten

Über das Plus-Symbol in der Artikelübersicht lässt sich ein neuer Artikel anlegen. Titel und Inhalt werden direkt in einem Rich-Text-Editor eingegeben (Überschriften, Textformatierungen, Listen, Links). Ein neuer Artikel ist direkt nach dem Anlegen automatisch für alle Accounts bearbeitbar und kann anschließend über die Freigaben (siehe unten) gezielt auf einzelne Rollen eingeschränkt werden.
Ein bestehender Artikel öffnet sich mit zwei Reitern:

- **Artikel**: Titel und Inhalt, editierbar für jeden Account mit Bearbeitungsrecht
- **Freigaben**: Sichtbarkeit und Bearbeitbarkeit des Artikels

Ohne Bearbeitungsrecht wird der Inhalt nur lesend angezeigt, der Reiter „Freigaben“ entfällt in diesem Fall.

5.7.2 Freigaben

Sichtbarkeit und Bearbeitbarkeit werden für jeden Artikel unabhängig voneinander gesetzt:

- **Für alle sichtbar**: macht den Artikel für alle Accounts lesbar
- **Für alle bearbeitbar**: macht den Artikel für alle Accounts bearbeitbar
- **Rollen**: Rollen können für Sichtbarkeit bzw. Bearbeitbarkeit ausgewählt werden. Ein Account ohne Admin-Rechte kann dabei nur Rollen zuweisen, denen er selbst angehört; Admins sowie der Owner der Anwendung können beliebige Rollen zuweisen. Bereits am Artikel hinterlegte fremde Rollen bleiben davon unberührt und können weiterhin entfernt werden.

Jeder Account mit Bearbeitungsrecht kann die Freigaben jederzeit über den Reiter „Freigaben“ anpassen.
Hinweis: Ein Artikel muss immer für mindestens eine Rolle oder für alle bearbeitbar bleiben. Das Speichern der Freigaben wird verweigert, wenn dadurch niemand außer dem erstellenden Account mehr bearbeiten könnte. So geht das hinterlegte Wissen nicht verloren, wenn der ursprüngliche Account gelöscht wird oder ausscheidet.
Der Eigentümer (Owner) der Anwendung sieht und bearbeitet zusätzlich automatisch alle Artikel, die nicht ausschließlich für den erstellenden Account bestimmt sind - also alle Artikel mit Rollen- oder „für alle“-Freigabe.

5.7.3 Löschen

Wissensartikel lassen sich über den Button „Artikel löschen“ entfernen. Da dies unwiderruflich ist, muss die Löschung in einem zusätzlichen Dialog bestätigt werden.

6 Ökosystem FF Admin

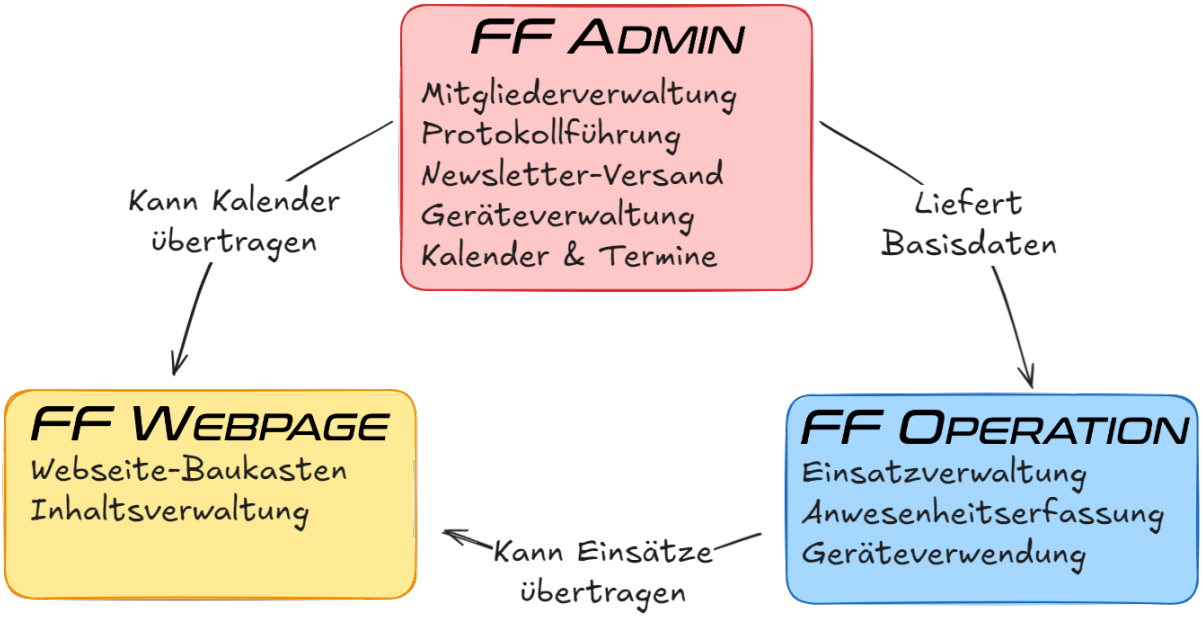


Abbildung 99: Übersicht zum Ökosystem

Das **FF Ökosystem** bietet eine zentrale Lösung für die Verwaltung von Mitgliedern, Einsätzen und Inhalten.

FF Admin ist das Herzstück mit der Mitglieder- und Geräteverwaltung sowie der Module zur Wehr- und Vereinsverwaltung. Es stellt Stammdaten bereit und ermöglicht mit allen seinen Modulen die effiziente Verwaltung einer Feuerwehr oder Vereins.

FF-Webpage erleichtert die Erstellung und Verwaltung von Webseiteninhalten. Kalender- bzw. Einsatzdaten können von FF Admin und FF Operation an die Webseite geschickt und dort veröffentlichen werden, um Mitglieder und die Öffentlichkeit auf dem Laufenden zu halten.

FF Operation unterstützt das Einsatzmanagement, die Anwesenheitserfassung und den Geräteeinsatz. Es kann die Mitgliederdaten, Gerätschaften und co aus FF Admin beziehen.

7 Roadmap

Folgende Funktionalitäten sind für FF Operation in Planung (Auszug):

- **Schnittstellen:** Implementierung von Schnittstellen wie UCRI, WDX3...
- **Dynamische Formulare:** Einstellbare Anzeige von Feldern je nach Eintragsart

Eine ausführliche Liste geplanter Funktionen kann im Git unter Issues gefunden werden:
https://code.jk-effects.cloud/FF-Admin/ff-operation/issues
https://code.jk-effects.cloud/FF-Admin/ff-operation-server/issues