



Anleitung

bis v1.7.x



Open Source Software
entwickelt durch JK Effects
von Julian Krauser
08. September 2026

Inhaltsverzeichnis

Inhaltsverzeichnis	II
Abkürzungsverzeichnis / Glossar	III
1 Einleitung	1
2 Installation	2
2.1 Docker	2
2.2 Docker-Compose	2
2.3 Git	5
2.4 Konfiguration	6
2.5 Update der Version	7
3 Strapi	9
3.1 Einrichtung	9
3.2 Modelle	11
3.2.1 Global	11
3.2.2 Design-Farben	15
3.2.3 Startseite	16
3.2.4 Sammlungs Referenz	16
3.2.5 Seiten	19
3.2.6 Artikel	20
3.2.7 Einsätze	21
3.2.8 Fahrzeuge	22
3.2.9 Termine:	23
3.2.10 Blog-Beiträge	24
3.2.11 Anleitungen	25
3.2.12 User	25
3.3 Verwendung	26
3.3.1 Dynamic Zone	26
3.3.2 Shared	31
4 Ökosystem FF Admin	33

Abkürzungsverzeichnis / Glossar

Collection Type: Speicher einer Auflistung an Einträgen eines bestimmten Typs.

Single Type: Speicher eines einzelnen Eintrags eines bestimmten Typs.

Slug: Identifikation eines Eintrags über den Wert eines Feldes, welcher einzigartig ist. Slugs können automatisiert erstellt werden.

1 Einleitung

FF Webpage - Der flexible Webseitenbaukasten für Feuerwehren und Vereine

FF Webpage ist ein modularer Webseitenbaukasten, bestehend aus einem Nuxt-Frontend und Strapi als CMS. Struktur und Navigation der Seite können über das Strapi CMS individuell konfiguriert werden, was eine flexible Gestaltung ermöglicht.

Speziell entwickelte Module erleichtern die Darstellung von Listen, wobei vordefinierte Listen für Einsätze, Termine, Artikel, Fahrzeuge, Blog-Beiträge und Anleitungen zur Verfügung stehen. Ein besonderes Feature ist die direkte Übernahme von Terminen aus FF Admin in die Website, wodurch eine nahtlose Integration in das FF-Ökosystem gewährleistet ist.

2 Installation

FF Webpage 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 Webpage 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. Der `<tag>` des Images kann entweder `latest` für die neueste Version oder `vx.y.z` für eine bestimmte Version sein. Die Versionen können auch in den Releases der Repositories der Anwendungen nachgeschlagen werden. Dort lassen sich auch Informationen zu neuen Funktionen, Änderungen oder Fehlerbehebungen der jeweiligen Funktion finden.

2.2 Docker-Compose

Webseite

```
1 ff-landingpage:
2   image: code.jk-effects.cloud/ff-admin/ff-webpage/frontend:<version>
3   container_name: ff-landingpage
4   restart: unless-stopped
5   environment:
6     - NUXT_STRAPI_URL=<cms_url>
7     - NUXT_PUBLIC_STRAPI_URL=<cms_url>
8     - NUXT_PUBLIC_APP_TITLE=<website_title>
9   ports:
10    - "3000:3000"
11   volumes:
12    - <volume|local path>/favicon.png:/app/.output/public/favicon.png
```

YAML

Die Verwendung der Werte des Typs Environment werden unter dem Punkt Konfiguration erklärt.

Das Volume dient zur erweiterten Personalisierung der Webseite mit eigenem Logo der Feuerwehr

Anleitung zu FF Webpage bis v1.7.x – Installation

oder des Vereins. Hiervon betroffen ist nur Icon im Browser-Tab.

Die Konfiguration der Volumes ist optional, falls Sie die Standard-Logos verwenden wollen. Die Dateien müssen exakt gleich geschrieben sein. Achten Sie deshalb auf Schreibfehler und Groß-/Kleinschreibung.

CMS - Inhaltsverwaltung

```
1  ff-cms:
2    image: code.jk-effects.cloud/ff-admin/ff-webpage/cms:<version>
3    container_name: ff_cms
4    restart: unless-stopped
5    environment:
6      - DATABASE_CLIENT=mysql
7      - DATABASE_HOST=ff-db
8      - DATABASE_NAME=ffcms
9      - DATABASE_USERNAME=cms
10     - DATABASE_PASSWORD=dbuserpasswd
11     - JWT_SECRET=<random string>
12     - ADMIN_JWT_SECRET=<random string>
13     - APP_KEYS=<random string> # oder<random string>,<random string>,...
14     - API_TOKEN_SALT=<random string>
15     - TRANSFER_TOKEN_SALT=<random string>
16    volumes:
17      - <volume|local path>:/app/public/uploads
18    ports:
19      - "1337:1337"
```

Die Verwendung der Werte des Typs Environment werden unter dem Punkt Konfiguration 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 Uploads abgelegt.

Datenbank

Als Datenbank können MySQL, Postgres und SQLite verwendet werden. Postgres wird für den Produktiven Einsatz empfohlen.

Konfiguration von MySQL:

```
1 ff-db:
2   image: mariadb:<version (bsp 11.2)>
3   container_name: ff_db
4   restart: unless-stopped
5   ports:
6     - "3306:3306"
7   environment:
8     - MYSQL_DATABASE=<database name>
9     - MYSQL_USER=<username>
10    - MYSQL_PASSWORD=<user password>
11    - MYSQL_ROOT_PASSWORD=<root password>
12  volumes:
13    - <volume|local path>:/var/lib/mysql
```

MYSQL_USER und MYSQL_PASSWORD sind optional. Werden diese nicht gesetzt, kann der Server entweder mit dem Nutzer root und dem gesetzten MYSQL_ROOT_PASSWORD Zugang zur Datenbank erhalten, oder es wird im nachhinein ein Nutzerzugang erstellt, der Zugriff auf die erstellte Datenbank hat. MYSQL_DATABASE erstellt direkt eine Datenbank, die durch einen angelegten MYSQL_USER verfügbar ist.

Konfiguration von Postgres:

```
1 ff-db:
2   image: postgres:<version (bsp 16)>
3   container_name: ff_db
4   restart: unless-stopped
5   ports:
6     - "5432:5432"
7   environment:
8     - POSTGRES_DB=<database name>
9     - POSTGRES_USER=<username>
10    - POSTGRES_PASSWORD=<user password>
11  volumes:
12    - <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 zusätzlich ein Netzwerk angelegt werden. Dadurch ist das Veröffentlichen der Datenbank-Port-Exposes nicht mehr notwendig. Das Entfernen der port-Exposes verhindert den direkten Zugriff auf die Ports von außerhalb. Ergänzt muss hierfür das network und die Teilhabe des Backend-Containers am Netzwerk:

Anleitung zu FF Webpage bis v1.7.x – Installation

1. Ergänzung zu Server und Datenbank:

```
1 networks:
2   - ff_internal
```

2. Ergänzung zur finalen Compose:

```
1 networks:
2   ff_internal:
```

3. Optionale Ergänzung zum Server:

```
1 depends_on:
2   - ff-db
```

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

2.3 Git

Eine Alternative zu Docker ist die direkte Ausführung der Anwendungen auf dem Server oder Desktop Gerät.

Hierzu müssen die App und der Server als Quellcode auf das System geladen und dort direkt verwendet werden.

Die Veröffentlichung der App und des Servers, damit diese aus dem Internet erreichbar sind, muss gesondert eingerichtet werden.

Das System muss NodeJs und die bevorzugte Datenbank installiert haben.

Die NodeJs Prozesse können auch durch Tools wie pm2 verwaltet werden.

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-Dateien müssen vor der Ausführung von `npm run start` angelegt sein. Bei einer Änderung der Einträge müssen die Anwendungen lediglich neu gestartet werden.

Webseite

```
1 git clone https://code.jk-effects.cloud/ff-admin/ff-webpage.git
2 cd ff-webpage
3 npm install
4 npm run build
5 npm run start
```

Shell

Anleitung zu FF Webpage bis v1.7.x – Installation

`npm run start` nutzt das eingebaute Hosting von Nuxt, welches SSR ermöglicht, um die App zu betreiben.

Strapi

```
1 git clone https://code.jk-effects.cloud/ff-admin/ff-webpage-cms.git
2 cd ff-webpage-cms
3 npm install
4 npm run build
5 npm run start
```

Shell

2.4 Konfiguration

Folgende Werte können zu einem Container konfiguriert werden:

Variable	Zweck	Fallback	optional
📁 App-Variablen			
Nuxt_STRAPI_URL	URL, unter der das CMS erreichbar ist.		✖
Nuxt_PUBLIC_STRAPI_URL	URL, unter der das CMS erreichbar ist.		✖
Nuxt_PUBLIC_APP_TITLE	Text im Browser-Tab	FF Webpage	✓
📁 Server-Variablen			
DB_TYPE	Folgende Datenbanktypen sind verfügbar: mysql, postgres		✖
DB_HOST	URL zur Datenbank		✖
DB_PORT	Port der Datenbank		✖
DB_NAME	Name der Datenbank in welcher die Tabellen erstellt werden.		✖
DB_USERNAME	Nutzername für Zugang zu Datenbank		✖
DB_PASSWORD	Passwort zum Zugang zur Datenbank		✖
JWT_SECRET			✖
ADMIN_JWT_SECRET			✖
APP_KEYS			✖
API_TOKEN_SALT			✖
TRANSFER_TOKEN_SALT			✖
SYNC_SOFT	Verhindert das Überschreiben abweichender Konfigurationen beim Import von <code>`config/sync/`</code> .	false	✓

Variable	Zweck	Fallback	optional
SYNC_ON_START	Importiert die versionierte Konfiguration aus `config/sync/` automatisch bei jedem Start (nur im Production Modus wirksam).	true	✓
📡 Database-Variablen			
MYSQL_DATABASE	Name der Datenbank, die bei Erstellung direkt angelegt wird.		✳
MYSQL_USER	Benutzername des Users, der bei Erstellung direkt angelegt wird.		✓
MYSQL_PASSWORD	Passwort zum User, das bei Erstellung gesetzt wird.		✓
MYSQL_ROOT_PASSWORD	Passwort für den User root, das bei Erstellung gesetzt wird.		✳
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!

🔔 : Bei Verwendung von SQLite sind diese Variablen nicht notwendig!

Hinweis: Eine fehlerhafte Konfiguration der optionalen oder geforderten Variable verhindert das Starten der Anwendung.

Hinweis: Eine Änderung der Datenbank übernimmt nicht automatisch die Daten in die neue Datenbank.

2.5 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 `latest` nutzt, kann das neue Image direkt mit `docker pull neu` beziehen und dann den Container neustarten.

Anleitung zu FF Webpage bis v1.7.x – Installation

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.

3 Strapi

Disclaimer: Fehler, die auf der Webseite angezeigt werden, können auf eine fehlerhafte Konfiguration in Strapi hinweisen.

3.1 Einrichtung

Um die Webseite nutzen zu können, müssen die Datenspeicher Global mit der Navigation sowie die Startseite ausgefüllt sein.

Wichtig ist, dass ein paar Einstellungen vor der ersten Verwendung vorgenommen werden müssen. Dazu zählen:

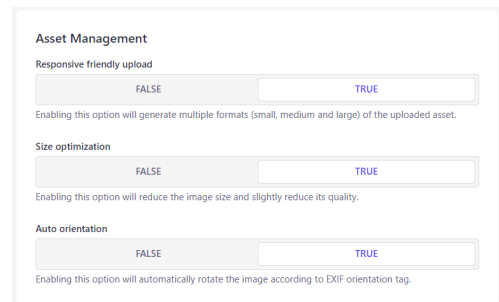
1. **Admin Benutzer erstellen:** Erstellen Sie einen Admin-Benutzer, um auf das CMS zugreifen zu können. Dazu muss die URL zum Content Management System geöffnet werden. Dort kann direkt der erste Benutzer als Admin angelegt werden.
2. **Einstellungen anpassen:** In den Einstellungen müssen unter dem Punkt Users & Permissions Plugin die Rollen und Berechtigungen angepasst werden. Unter Roles > Public müssen die Berechtigungen find, findOne und falls vorhanden distinctYears und findByYear für bestimmte Vorlagen aktiviert werden. Dadurch kann die Website später ohne weitere Einstellungen lesend auf die veröffentlichten Inhalte zugreifen. Die Leseberechtigungen müssen für Article, Blog-post, Collection-lookup, Event, Global, Homepage, How-to, Operation, Page, Theme-color und Vehicle aktiviert werden.

The screenshot shows the 'Article' role configuration in the Strapi Users & Permissions plugin. The title is 'Article' with a subtitle 'Define all allowed actions for the api:article plugin.' Below this is a search bar labeled 'ARTICLE' and a 'Select all' button. A table of permissions is displayed with checkboxes for 'create', 'delete', 'distinctYears', 'find', 'findOne', 'findByYear', and 'update'. The permissions 'distinctYears', 'find', 'findOne', and 'findByYear' are checked. Below the table are three expandable sections: 'Collection-lookup', 'Event', and 'Global', each with a subtitle 'Define all allowed actions for the api:collection-lookup plugin.', 'Define all allowed actions for the api:event plugin.', and 'Define all allowed actions for the api:global plugin.' respectively.

Permission	Active
create	☐
delete	☐
distinctYears	☒
find	☒
findOne	☒
findByYear	☒
update	☐

Diese Rechte werden beim Start von Strapi automatisch ergänzt - müssen also nur einmalig geprüft und nicht händisch nachgepflegt werden.

3. **Medienverwaltung:** Es wird empfohlen, unter Global Settings > Media Library alle Optionen auf true zu setzen. Dadurch wird z.B. auch die automatische Bilddrehung aktiviert. Dadurch werden die Bilder automatisch in die richtige Ausrichtung gedreht und in der Größe optimiert.



Asset Management

Responsive friendly upload

FALSE TRUE

Enabling this option will generate multiple formats (small, medium and large) of the uploaded asset.

Size optimization

FALSE TRUE

Enabling this option will reduce the image size and slightly reduce its quality.

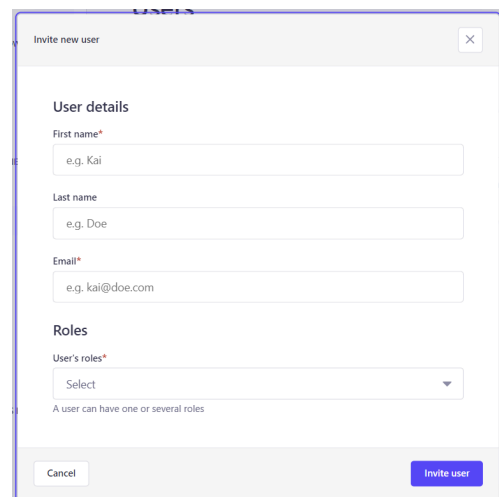
Auto orientation

FALSE TRUE

Enabling this option will automatically rotate the image according to EXIF orientation tag.

4. **Rollen und Berechtigungen:** Unter Administration Panel > Roles können die Rollen und Berechtigungen für die Benutzer angepasst werden. Es wird empfohlen, die Standardrollen zu übernehmen und nur die Berechtigungen anzupassen. Diese können jedoch beliebig erweitert werden. Dieser Punkt betrifft nur die Erstellung von Inhalten in der Administrationsoberfläche.

5. **Nutzer einladen:** Unter Administration Panel > Users können weitere Benutzer eingeladen werden. Diese erhalten dann eine E-Mail mit einem Link, über den sie ein Passwort festlegen können. Die eingeladenen Benutzer erhalten dann eine E-Mail, über die sie ihr Passwort setzen können. Rollen können hier direkt bei der Einladung vergeben werden, so dass nach der Registrierung direkt mit der Bearbeitung begonnen werden kann. Theoretisch können Benutzer auch eingeladen und direkt editiert werden, um ein Passwort zu setzen. Dies kann verwendet werden, wenn die E-Mail-Adresse nicht existiert.



Invite new user

User details

First name*

e.g. Kai

Last name

e.g. Doe

Email*

e.g. kai@doe.com

Roles

User's roles*

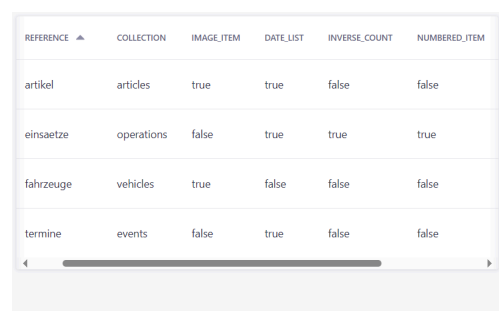
Select

A user can have one or several roles

Cancel Invite user

6. **Erstellung von Inhalten:** Nachdem die Einstellungen vorgenommen wurden, können die Inhalte erstellt werden. Dazu können die Inhalte über die Schnittstelle zu den bestehenden Modellen erstellt werden.

7. **Einrichtung Sammlungsreferenz:** Die Sammlungsreferenz ist ein eigenes Modell, das die Darstellung von Artikeln, Einsätzen, Fahrzeugen und Terminen regelt. Die Sammlungswerte sind die Verweise auf die verfügbaren Listen. Es wird empfohlen, diese Werte direkt zu definieren, damit sie in anderen Modellen als Referenz verwendet werden können.



REFERENCE	COLLECTION	IMAGE_ITEM	DATE_LIST	INVERSE_COUNT	NUMBERED_ITEM
artikel	articles	true	true	false	false
einsaetze	operations	false	true	true	true
fahrzeuge	vehicles	true	false	false	false
termine	events	false	true	false	false

3.2 Modelle

3.2.1 Global

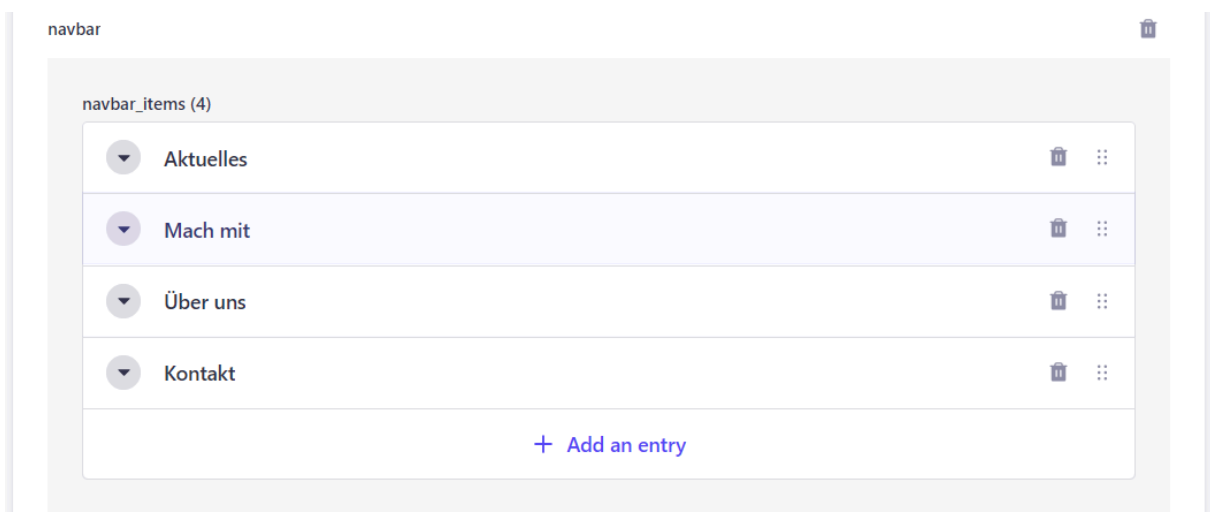
Im **Single Type Global** können Werte hinterlegt werden, die im Kontext der gesamten Website zur Verfügung stehen sollen. Diese Einträge umfassen die Navigationsleiste, die Fußzeile und SEO, welche die Anzeige in einer Suchmaschine steuern. Diese Werte müssen definiert werden, damit die Webseite funktionsfähig angezeigt werden kann.

Logo und Favicon



Das Logo und Favicon können global definiert werden und ersetzen dadurch das Fallback der Webseite. Soll irgendwo ein Bild angezeigt werden, das nicht geladen werden kann, so wird dort auch das Logo als Ersatz angezeigt. Das Favicon sollte quadratisch hochgeladen werden, damit dieses im Tab korrekt angezeigt wird.

Navigation



Die Navigationsleiste kann mit einem Logo und eigenen Links definiert werden. Die Links können entweder eigenständig oder mit Sublinks erstellt werden. Außerdem kann ein Top-Level-Link direkt auf eine Seite oder auf einen Sub-Link verweisen. Die Links erhalten immer eine Bezeichnung und einen Text, wie der Link in der URL aussehen soll. Jedem Link, der etwas anzeigen soll, muss eine Seite zugeordnet werden. Das Anlegen von Seiten wird in [Abschnitt 3.2.5](#) erklärt.

Links in der Navigation ohne untergeordneten Links:



Top-Level-Links ohne Unterlinks werden in der Hauptnavigationsleiste angezeigt. Diese Links verweisen auf eine zugeordnete Seite. Die angelegten Seiten können über das angezeigte Dropdown-Menü ausgewählt werden.

This screenshot shows the Strapi admin interface for a link named 'Mach mit'. The 'name' field is 'Mach mit' and the 'URL' field is 'mach-mit'. The 'default_active_child' field is empty. The 'page (1)' dropdown menu is open, showing 'Add relation' and 'Mach Mit' (which is highlighted). A 'Published' button is visible. The 'navbar_sub_items (0)' section is empty, with a message 'No entry yet. Click to add one.'

Links in der Navigation mit untergeordneten Links:



Top-Level-Links mit Unterlinks werden in der Hauptnavigationsleiste angezeigt. Wenn ein Link ausgewählt wird, werden die erzeugten Sublinks angezeigt. Je nach Konfiguration des Links wird dann entweder direkt eine Seite oder eine Seite des Sublinks aufgerufen. Beim Verweis auf Sublinks muss der Verweis

This screenshot shows the Strapi admin interface for a link named 'Aktuelles'. The 'name' field is 'Aktuelles' and the 'URL' field is 'aktuelles'. The 'default_active_child' field is 'einsaetze'. The 'page (1)' dropdown menu is open, showing 'Add relation' and 'Einsätze' (which is highlighted). A 'Published' button is visible. The 'navbar_sub_items (1)' section shows a sublink named 'Einsätze' with its own 'name*' and 'URL*' fields, and a 'Published' button. A '+ Add an entry' button is at the bottom.

default_active_child mit dem Wert der URL des Sublinks übereinstimmen.

Fußleiste

The screenshot shows the 'footer' configuration section in Strapi. It includes a 'layout*' dropdown menu set to 'single'. Below this, a text explanation states: '"single" = eine schlichte Link-Zeile unten (Feld "links"). "cols" = mehrspaltiger Footer mit Logo-Spalte und Link-Spalten (Felder "first_column"/"columns").'. There are two input fields: 'maintained_by' containing 'Julian Krauser' and 'copyright' containing 'JK Effects'. At the bottom, there are two lines of explanatory text: 'Optionaler Hinweis auf den Seitenbetreiber/die Pflege der Website in der Fußzeile.' and 'Copyright-Zeile in der Fußzeile (z. B. "© 2026 Freiwillige Feuerwehr").'

Die Fußzeile kann in zwei Layouts (layout) dargestellt werden:

- single: Eine einzelne, zentrierte Zeile mit Links (Feld links).
- cols: Ein mehrspaltiger Footer mit einer eigenen Logo-Spalte (Feld first_column: Logo, Vereins-/ Website-Titel, kurzer Text) und beliebig vielen weiteren Link-Spalten (Feld columns: jeweils Überschrift + Links).

The screenshot shows the 'links (4)' configuration section in Strapi. It displays a list of four links: 'Impressum', 'Datenschutz', 'Codebase', and 'Entwickler'. Each link entry has a dropdown arrow on the left and a trash icon with a three-dot menu on the right. At the bottom of the list, there is a blue '+ Add an entry' button.

Die Links können auf eine in den Seiten definierte URL ([Abschnitt 3.2.5](#)), auf externe Webseiten oder auf eine E-Mail-Adresse (mailto: ...) bzw. Telefonnummer (tel: ...) verweisen. Um auf eine definierte Seite zu verweisen, muss nur der Pfad angegeben werden. Ein Pfad kann mit einem / beginnen oder ein ganzer Pfad sein (aktuelles/einsaetze). Soll ein Link auf eine externe Seite verweisen, muss dem Link https:// oder http:// vorangestellt werden.

Anleitung zu FF Webpage bis v1.7.x – Strapi

first_column

logo



FF Admin Fire.png

Logo in der ersten Footer-Spalte (ohne Angabe wird kein Logo angezeigt).

heading

FF Admin

Website-Titel/Vereinsname in der ersten Footer-Spalte.

text

Open Source Software für Feuerwehren und Vereine. Entwickelt von JK Effects.

Kurzer Text unter dem Logo, z. B. eine kurze Beschreibung des Vereins.

columns (3)

Anwendungen

Ressourcen

Kontakt

+ Add an entry

Zusätzlich können unabhängig vom gewählten Layout ein Copyright-Hinweis (`copyright`) sowie ein Hinweis auf den Seitenbetreiber (`maintained_by`) hinterlegt werden.

SEO (Search Engine Optimization)

SEO

metaTitle*

FF Webpage Demo

max. 60 characters

metaDescription

Demo-Seite der FF Webpage mit Beispielen für Inhal

keywords

Der Abschnitt SEO ist optional und kann verwendet werden, um den Titel der Registerkarte und die Beschreibung der Seite festzulegen. Dies optimiert die Ergebnisse in Suchmaschinen, da u.a. die Beschreibung direkt angezeigt wird.

Fonts



Optional können eigene Schriftarten für Überschriften und sonstige Texte definiert werden. Das Template wurde unter Verwendung von Conthrax Semibold in den Überschriften und General Sans Medium für sonstige Texte entwickelt.

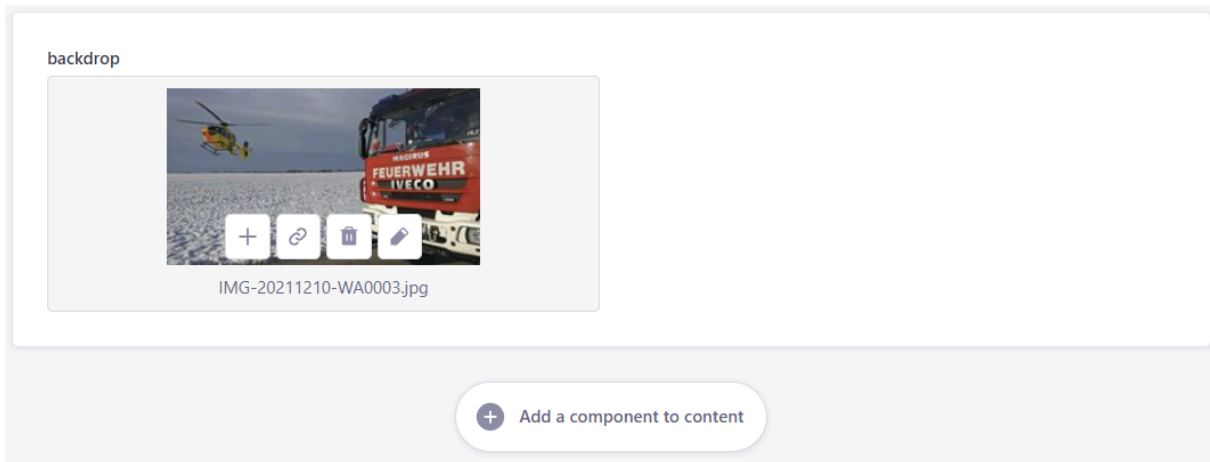
3.2.2 Design-Farben

Der [Collection Type Design-Farbe](#) legt die auf der Website verfügbaren Farben fest. Jede Farbe besteht aus einem Namen, einem Farbwert (`color`) und optional einer Rolle (`role`).

Farben mit einer der festen Rollen (`primary`, `secondary`, `accent`, `background`, `surface`) werden automatisch an den entsprechenden Stellen der Website verwendet - so bestimmt z.B. die Farbe mit der Rolle `primary` die Farbe der Navigationsleiste sowie primärer Buttons. Soll eine Farbe keiner festen Rolle zugeordnet, aber trotzdem gezielt nutzbar sein (z.B. als Hintergrund eines Bausteins „Bereich“, siehe [Abschnitt 3.3](#)), bleibt die Rolle auf `custom`.

Die Textfarbe auf einer gewählten Farbe (z.B. in der Navigationsleiste oder im Baustein „Bereich“) wird automatisch anhand der Helligkeit der Farbe zwischen Schwarz und Weiß gewählt, sodass immer ausreichend Kontrast/Lesbarkeit gewährleistet ist. Bei der Wahl einer Design-Farbe muss deshalb nicht zusätzlich auf die Lesbarkeit des Textes darauf geachtet werden.

3.2.3 Startseite



Der **Single Type** **Startseite** definiert die Startseite. Diese ist immer fest und muss konfiguriert werden, damit die Website funktionsfähig ist.

Die Startseite besteht optional aus einem Bild, das über der Navigationsleiste angezeigt werden kann. Die Funktionalität, dass etwas über der Navigationsleiste angezeigt wird, wird anhand der Füllung des Feldes `backdrop` entschieden. Ist kein Bild angegeben, wird überhalb der Navigationsleiste nichts angezeigt.

Das Logo, das auch auf der Startseite verwendet wird, ist das gleiche, das unter Global ([Abschnitt 3.2.1](#)) angegeben wurde.

Die Erstellung von Inhalten mit Hilfe von Komponenten wird unter Verwendung in [Abschnitt 3.3](#) erläutert. Die Startseite ist eine vollwertige Seite wie die Seiten ([Abschnitt 3.2.5](#)), jedoch mit erweiterten Darstellungsmöglichkeiten, um die Startseite hervorzuheben.

3.2.4 Sammlungs Referenz

reference* Artikel	collection* articles
date_use* date	list_with_date* by-year
show_image* <input type="checkbox" value="FALSE"/> <input checked="" type="checkbox" value="TRUE"/>	enable_detail* <input type="checkbox" value="FALSE"/> <input checked="" type="checkbox" value="TRUE"/>
item_index_use* number	show_future_past_break* <input type="checkbox" value="FALSE"/> <input type="checkbox" value="TRUE"/>
low_number_starts* oldest	

Der **Collection Type Sammlungs Referenz** definiert global, wie die Listen für Artikel, Einsätze, Termine, Fahrzeuge, Blog-Beiträge und Anleitungen angezeigt werden sollen. Damit wird auch festgelegt, dass der Inhalt bei jeder Verwendung einheitlich ist.

Damit eine dieser Listen auf den Seiten verwendet werden kann, muss eine Sammlungsreferenz erstellt worden sein.

Die verfügbaren Einstellmöglichkeiten werden abhängig von Werten der anderen Felder angezeigt.

- **show_image:** Soll das Element in der Liste ein Bild anzeigen.
- **grid_layout:** Nur relevant, wenn **show_image** deaktiviert ist. Ordnet die Elemente als umbrechendes Raster nebeneinander an, statt sie schlicht untereinander aufzulisten.
- **cta_text:** Optionaler Text am Ende jedes Listen-Elements, z.B. „Weiterlesen“. Es handelt sich dabei nur um einen Hinweistext, da das gesamte Element bereits klickbar ist.
- **enable_detail:** Soll das Element auf eine Detailseite weiterleiten können. Zusätzlich wird angegeben, ob eine Detailseite überhaupt verfügbar ist.
- **list_with_date:** Dieses Auswahlfeld beinhaltet mehrere Varianten.
 - **none:** Soll kein Datum zu den Einträgen angezeigt werden.
 - **date:** Soll nur das Datum zu den Einträgen angezeigt werden.
 - **date-time:** Soll das Datum mit Uhrzeit zu den Einträgen angezeigt werden.
- **list_with_date:** Dieses Auswahlfeld beinhaltet mehrere Varianten.
 - **none:** Soll die Liste ohne Gruppierung nach Zeit angezeigt werden.
 - **by-year:** Sollen die Listenelemente nach Jahren gruppiert angezeigt und navigiert werden.
 - **by-month:** Sollen die Elemente durch einen Text mit dem Monat getrennt werden.
- **item_index_use:** Dieses Auswahlfeld beinhaltet mehrere Varianten.
 - **none:** Sollen die Liste keine Zahl am Anfang besitzen.
 - **number:** Sollen die Listenelemente mit einer Nummer angezeigt werden.
 - **day:** Soll der Tag des Datums am Anfang des Eintrags angezeigt werden.
 - **day-with-month:** Soll der Tag mit Kurzform des Monats am Anfang des Eintrags angezeigt werden.
- **show_future_past_break:** Soll ein Indikator angezeigt werden, um vergangene Einträge zu kennzeichnen. Es wird immer die Referenz an vergangene abgehängt.
- **low_number_starts:** Dieses Auswahlfeld beinhaltet mehrere Varianten.
 - **oldest:** Der älteste Eintrag bekommt die kleinste Nummer.
 - **newest:** Der neueste Eintrag bekommt die kleinste Nummer.

Beispiele:

Anzeige von z.B. Artikeln als Liste mit Bild:

`list_with_date = by-year`

`item_index_use = day`



Anzeige von z.B. Einsätzen als Liste ohne Bild:

`list_with_date = by-month`

`item_index_use = number`

`low_number_start = oldest`



Anzeige von z.B. Terminen als Liste ohne Bild:

`list_with_date = by-year`

`item_index_use = day`



Anzeige von z.B. Fahrzeugen als Liste ohne Datum und Nummer:

`list_with_date = inverted`

`items_with_number = inverted`



3.2.5 Seiten

The screenshot shows the Strapi admin interface for the 'Seiten' (Pages) collection type. The form contains the following fields:

- identifier***: A text input field containing 'Einsätze'.
- slug***: A text input field containing 'einsaetze' with a refresh icon.
- ref_only_access**: A toggle switch with 'FALSE' selected (highlighted in red) and 'TRUE'.
- hero***: A section containing:
 - banner**: A media selection area showing a selected image 'IMG-20240328-WA0011.jpg' with icons for adding, linking, deleting, and editing.
 - title**: A text input field containing 'vergangene Einsätze'.

Der **Collection Type Seiten** speichert alle Seiten, die erreicht werden können. Jede Seite ist über den URL-Pfad, welche im **Slug** definiert ist, erreichbar, sofern `ref_only_access` auf `false` steht und eine Navigationseinstellung den **Slug** nicht überlagert. Standardmäßig ist die Erreichbarkeit auf `<meine Domain>/<slug>` gesetzt.

Der **Slug** kann auch vollständige Pfade beinhalten. Dabei wird ein Pfad wie `einsaetze` ZU `<meine Domain>/einsaetze` oder `mach-mit-formular` ZU `<meine Domain>/mach-mit/formular`. Ein `~` stellt dabei ein / dar.

Ist bei einer Seite `ref_only_access` auf `true` gesetzt, so ist eine Seite nur noch per Verlinkung über die Navigation erreichbar.

Der Identifier dient nur zur lesbaren Identifizierung der Seite, wenn sie einem Navigationslink zugeordnet ist. Der **Slug** speichert den Pfad, der in der URL angezeigt wird.

Eine Seite kann ein Banner (Hero) haben. Dieses Banner wird direkt unter der Navigation angezeigt. Der Hero kann mit einer Unterüberschrift versehen werden. Ist das Banner oder die Unterüberschrift nicht gesetzt, so wird der nicht erstellte Inhalt nicht angezeigt.

Der Hauptteil einer Seite kann mit definierten Komponenten erstellt werden. Die Erstellung von Inhalten mit Hilfe von Komponenten wird unter Verwendung in [Abschnitt 3.3](#) erläutert.



3.2.6 Artikel

The image shows a form for creating a new article in Strapi. The form is divided into several sections. The first section has two input fields: 'title*' with the value 'Übungsdienst' and 'description*' with the value 'Übung'. The second section has a 'slug*' field with the value 'uebungsdienst-1' and a refresh icon. The third section has an 'image' field showing a preview of a sunset over a body of water, with a file name 'IMG-20210917-WA0048.jpg' below it. The fourth section has a 'date*' field with a calendar icon and the value '07/08/2024'. Below the image field, there are four icons: a plus sign, a link, a trash can, and a pencil.

Der **Collection Type Artikel** speichert Beiträge der Website ähnlich wie Artikel in einer Zeitung. Die Art und Weise, wie ein Artikel auf der Website angezeigt wird, wird in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) festgelegt.

Jeder Artikel enthält einen Titel, einen URL-Pfad (**Slug**), ein Datum und optional ein kurzes Schlagwort (*category*), das in der Liste oberhalb des Titels angezeigt wird (z.B. „Neuigkeit“). Der Inhalt wird analog zum Inhalt von Seiten ([Abschnitt 3.2.5](#)) erstellt und wird mittels der definierten Komponenten erstellt. Die Erstellung von Inhalten mit Hilfe von Komponenten wird unter Verwendung in [Abschnitt 3.3](#) erläutert.

Die Felder wie auch der Inhalt werden je nach Konfiguration der Sammlungsreferenz ([Abschnitt 3.2.4](#)) angezeigt. Der Inhalt wird z.B. nur benötigt, wenn ein Artikel geöffnet werden

kann. Ob die Detailseite eines Artikels geöffnet werden kann, wird in der Sammlungsreferenz (Abschnitt 3.2.4) anhand des Wertes `enable_detail` eingestellt.

3.2.7 Einsätze

The screenshot shows a Strapi CMS form for creating a new 'Einsatz' (Incident). The form is divided into several sections:

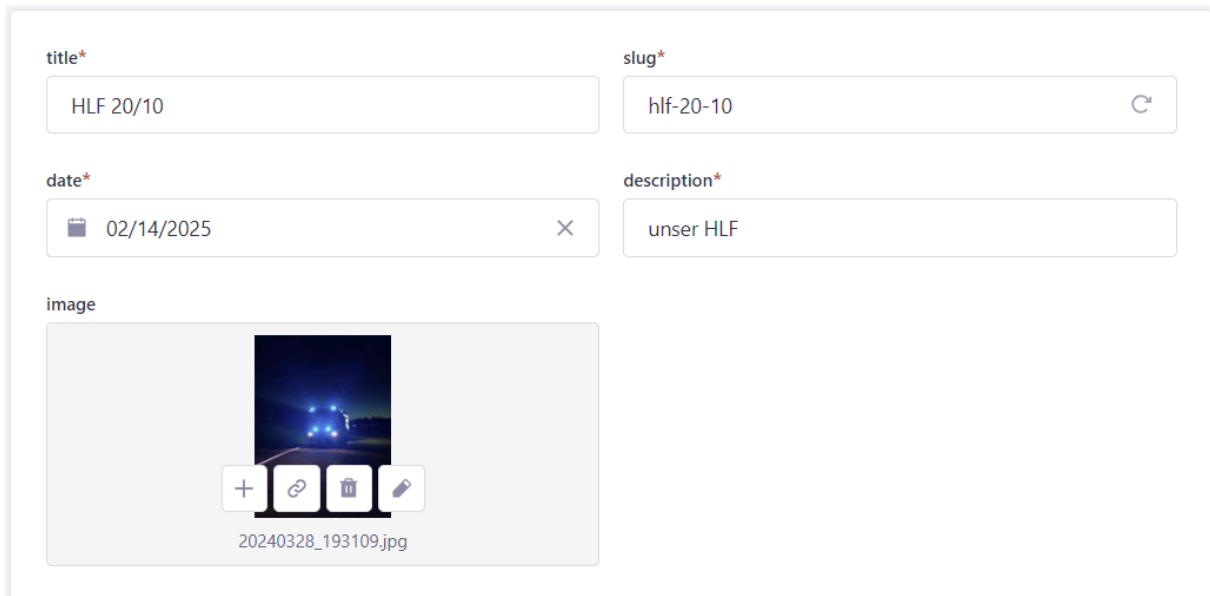
- title***: A text input field containing the value 'Brand'.
- description***: A text input field containing the value 'Feuer'.
- slug***: A text input field containing the value 'brand', with a refresh icon to its right.
- date***: A date picker field showing '04/10/2024' with a close icon.
- time***: A time picker field showing '02:00' with a close icon and a dropdown arrow.
- image**: A large rectangular area with a blue plus icon in the center and the text 'Click to add an asset or drag and drop one in this area' below it.

Der **Collection Type Einsatz** speichert vergangene Einsätze ab. Die Art und Weise, wie ein Einsatz auf der Website angezeigt wird, wird in der Sammlungsreferenz (Abschnitt 3.2.4) festgelegt.

Jeder Einsatz enthält einen Titel, einen URL-Pfad (**Slug**), ein Datum und optional ein kurzes Schlagwort (`category`), das in der Liste oberhalb des Titels angezeigt wird. Der Inhalt wird analog zum Inhalt von Seiten (Abschnitt 3.2.5) erstellt und wird mittels der definierten Komponenten erstellt. Die Erstellung von Inhalten mit Hilfe von Komponenten wird unter Verwendung in Abschnitt 3.3 erläutert.

Die Felder wie auch der Inhalt werden je nach Konfiguration der Sammlungsreferenz (Abschnitt 3.2.4) angezeigt. Der Inhalt wird z.B. nur benötigt, wenn ein Einsatz geöffnet werden kann. Ob die Detailseite eines Einsatzes geöffnet werden kann, wird in der Sammlungsreferenz (Abschnitt 3.2.4) anhand des Wertes `enable_detail` eingestellt.

3.2.8 Fahrzeuge



The screenshot shows the Strapi admin interface for the 'Fahrzeuge' collection type. The form includes the following fields:

- title***: A text input field containing 'HLF 20/10'.
- slug***: A text input field containing 'hlf-20-10' with a refresh icon on the right.
- date***: A date picker field showing '02/14/2025' with a close icon on the right.
- description***: A text input field containing 'unser HLF'.
- image**: A media library preview showing a photo of a fire truck at night. Below the image are icons for adding, linking, deleting, and editing. The filename '20240328_193109.jpg' is visible below the image.

Der **Collection Type Fahrzeuge** speichert Fahrzeuge des Fuhrparks ab. Die Art und Weise, wie ein Fahrzeug auf der Website angezeigt wird, wird in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) festgelegt.

Jedes Fahrzeug enthält einen Titel, einen URL-Pfad (**Slug**), ein Datum und optional ein kurzes Schlagwort (**category**), das in der Liste oberhalb des Titels angezeigt wird. Der Inhalt wird analog zum Inhalt von Seiten ([Abschnitt 3.2.5](#)) erstellt und wird mittels der definierten Komponenten erstellt. Die Erstellung von Inhalten mit Hilfe von Komponenten wird unter Verwendung in [Abschnitt 3.3](#) erläutert.

Die Felder wie auch der Inhalt werden je nach Konfiguration der Sammlungsreferenz ([Abschnitt 3.2.4](#)) angezeigt. Der Inhalt wird z.B. nur benötigt, wenn ein Fahrzeug geöffnet werden kann. Ob die Detailseite eines Fahrzeugs geöffnet werden kann, wird in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) anhand des Wertes `enable_detail` eingestellt.

3.2.9 Termine:

The screenshot shows the Strapi admin interface for creating a new 'Termin' (Event) entry. The form is organized into several sections:

- title***: A text input field containing 'Jahreshauptversammlung'.
- description***: A text input field containing 'Jahreshauptversammlung der Feuerwehr'.
- slug***: A text input field containing 'jahreshauptversammlung' with a refresh icon (circular arrow) to its right.
- date***: A date picker showing '01/06/2025' with a close icon (X) to its right.
- time***: A time picker showing '01:00' with a close icon (X) and a dropdown arrow to its right.
- image**: A large rectangular area with a blue plus icon in the center and the text 'Click to add an asset or drag and drop one in this area' below it.

Der **Collection Type Termin** speichert vergangene oder anstehende Termine. Die Art und Weise, wie ein Termin auf der Website angezeigt wird, wird in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) festgelegt.

Jeder Termin enthält einen Titel, einen URL-Pfad (**Slug**), ein Datum und optional ein kurzes Schlagwort (**category**), das in der Liste oberhalb des Titels angezeigt wird. Der Inhalt wird analog zum Inhalt von Seiten ([Abschnitt 3.2.5](#)) erstellt und wird mittels der definierten Komponenten erstellt. Die Erstellung von Inhalten mit Hilfe von Komponenten wird unter Verwendung in [Abschnitt 3.3](#) erläutert.

Die Felder wie auch der Inhalt werden je nach Konfiguration der Sammlungsreferenz ([Abschnitt 3.2.4](#)) angezeigt. Der Inhalt wird z.B. nur benötigt, wenn ein Termin geöffnet werden kann. Ob die Detailseite eines Termins geöffnet werden kann, wird in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) anhand des Wertes `enable_detail` eingestellt.

3.2.10 Blog-Beiträge

The screenshot shows the Strapi CMS form for creating a new Blog-Beitrag. The form is organized into several sections:

- title***: A text input field containing "Neu in Admin v1.19: Nachweismanagement". Below it, a note states: "Haupttitel, erscheint als Überschrift in der Detailansicht und in Listen."
- slug***: A text input field containing "neu-in-admin-v1-19-nachweismanagement". Below it, a note states: "Wird automatisch aus dem Titel erzeugt und ergibt die URL des Eintrags. Nur bei Bedarf manuell anpassen."
- category**: A dropdown menu showing "Release - September 2026". Below it, a note states: "Kurzes Schlagwort, das in Listen als Hinweis über dem Titel angezeigt wird (z. B. 'Release', 'Tipp')."
- date***: A date and time picker showing "09/03/2026" and "00:00". Below it, a note states: "Datum des Eintrags, bestimmt u. a. die Sortierung in Listen."
- description**: A text area containing "Nachweise und die dazugehörigen Termine lassen sich jetzt zentral zu jedem Mitglied erfassen – inklusive automatischer Erinnerung." Below it, a note states: "Kurze Zusammenfassung für die Listenansicht (nicht der ausführliche Inhalt weiter unten)."
- image**: A large empty box with a blue plus icon in the center. Below it, a note states: "Titelbild, erscheint in Listen und in der Detailansicht."

Der Collection Type **Blog-Beitrag** speichert Blogbeiträge der Website, z.B. für Neuigkeiten, Ankündigungen oder Tipps. Die Art und Weise, wie ein Blog-Beitrag auf der Website angezeigt wird, wird in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) festgelegt.

Jeder Blog-Beitrag enthält einen Titel, einen URL-Pfad (**Slug**), ein Datum und optional ein kurzes Schlagwort (**category**), das in der Liste oberhalb des Titels angezeigt wird (z.B. „Release“ oder „Tipp“). Der Inhalt wird analog zum Inhalt von Seiten ([Abschnitt 3.2.5](#)) erstellt und wird mittels der definierten Komponenten erstellt. Die Erstellung von Inhalten mit Hilfe von Komponenten wird unter Verwendung in [Abschnitt 3.3](#) erläutert.

Die Felder wie auch der Inhalt werden je nach Konfiguration der Sammlungsreferenz ([Abschnitt 3.2.4](#)) angezeigt. Der Inhalt wird z.B. nur benötigt, wenn ein Blog-Beitrag geöffnet werden kann. Ob die Detailseite eines Blog-Beitrags geöffnet werden kann, wird in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) anhand des Wertes `enable_detail` eingestellt.

3.2.11 Anleitungen

The screenshot shows the Strapi content editor interface for a 'Collection Type' entry. The form is organized into several sections:

- title***: A text input field containing 'Einsatzdokumentation'. Below it, a description reads: 'Haupttitel, erscheint als Überschrift in der Detailsicht und in Listen.'
- slug***: A text input field containing 'einsatzdokumentation'. Below it, a description reads: 'Wird automatisch aus dem Titel erzeugt und ergibt die URL des Eintrags. Nur bei Bedarf manuell anpassen.'
- category**: A text input field containing 'FF Operation'. Below it, a description reads: 'Kurzes Schlagwort, das in Listen als Hinweis über dem Titel angezeigt wird (z. B. "Release", "Tipp").'
- date***: A date and time picker. The date is set to '09/02/2026' and the time to '00:00'. Below it, a description reads: 'Datum des Eintrags, bestimmt u. a. die Sortierung in Listen.'
- description**: A text input field containing 'Kollaborative Dokumentation von Übungen und Einsätzen'. Below it, a description reads: 'Kurze Zusammenfassung für die Listenansicht (nicht der ausführliche Inhalt weiter unten).'
- image**: A large rectangular area with a blue plus icon in the center. Below it, a description reads: 'Titelbild, erscheint in Listen und in der Detailsicht.'

Der Collection Type **Anleitung** speichert Schritt-für-Schritt-Anleitungen, z.B. für wiederkehrende Abläufe oder Bedienungshinweise. Die Art und Weise, wie eine Anleitung auf der Website angezeigt wird, wird in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) festgelegt.

Jede Anleitung enthält einen Titel, einen URL-Pfad (**Slug**), ein Datum und optional ein kurzes Schlagwort (**category**). Der Inhalt wird analog zum Inhalt von Seiten ([Abschnitt 3.2.5](#)) erstellt und wird mittels der definierten Komponenten erstellt. Die Erstellung von Inhalten mit Hilfe von Komponenten wird unter Verwendung in [Abschnitt 3.3](#) erläutert.

Die Felder wie auch der Inhalt werden je nach Konfiguration der Sammlungsreferenz ([Abschnitt 3.2.4](#)) angezeigt. Der Inhalt wird z.B. nur benötigt, wenn eine Anleitung geöffnet werden kann. Ob die Detailseite einer Anleitung geöffnet werden kann, wird in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) anhand des Wertes `enable_detail` eingestellt.

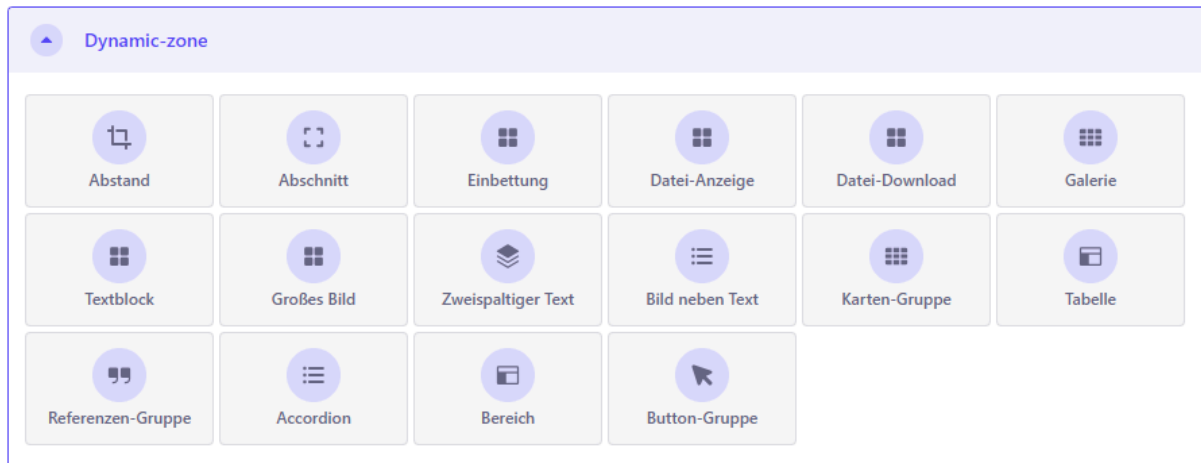
3.2.12 User

Die Verwendung des Collection Type **User** ist nicht erforderlich, da diese nur benötigt wird, wenn sich Besucher der Website auf der Website anmelden könnten.

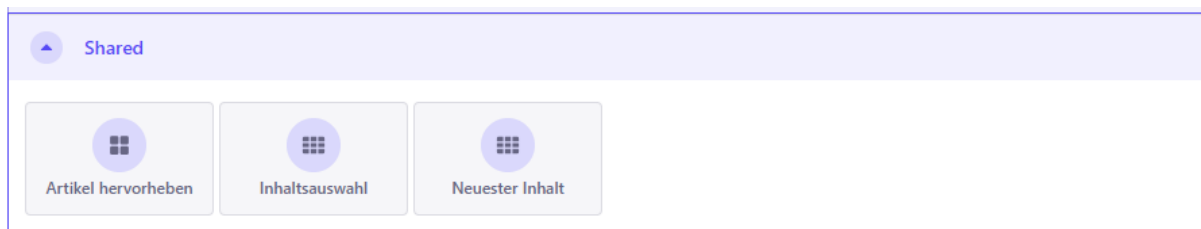
3.3 Verwendung

Es gibt 2 Arten an Komponenten, welche zur Darstellung von Inhalten verwendet werden können.

Komponenten der `Dynamic-Zone` werden verwendet, um Inhalte direkt hinzuzufügen.



Komponenten aus `shared` werden verwendet, um Inhalte aus anderen Listen verwenden zu können.



3.3.1 Dynamic Zone

Abstand

Die Komponente Abstand ermöglicht einen Abstand zwischen Elementen. Ein Abstand wird als kurze Line zwischen den anderen Komponenten angezeigt.



Abschnitt

Die Komponente Abschnitt ermöglicht das Einfügen einer Überschrift als Kennzeichnung eines Abschnitts. Optional kann eine Beschreibung mit angegeben werden. Zusätzlich lässt sich die Ausrichtung von Überschrift und Beschreibung (links, mittig, rechts) einstellen. Die Komponenten direkt darunter (z.B. Karten-Gruppe, Tabelle) benötigen dadurch keine eigene Überschrift mehr.

The screenshot shows the configuration interface for the 'Abschnitt' component. It includes a 'title*' field, a 'description' field, and a 'title_position*' dropdown menu set to 'center'. Below the dropdown, it says 'Textausrichtung von Überschrift und Untertitel.' There is also a note: 'Überschrift für die nachfolgenden Bausteine. Die Karten/Tabelle/Referenzen etc. darunter brauchen deshalb keine eigene Überschrift mehr.' and 'Optionaler Untertitel unter der Überschrift.'

Einbettung

Die Einbettung ermöglicht die Anzeige externer Links. Diese Links können Bilder, PDFs oder Webseiten sein. Ein Beispiel für eine Einbettung wäre die öffentliche Kalender-Ansicht des FF Admin.

The screenshot shows the configuration interface for the 'Einbettung' component. It has a single 'link' text input field.

Datei-Anzeige

Diese Komponente ermöglicht die Einbettung von Bildern oder PDFs. Wird ein PDF oder ein Bild angegeben, so wird dieses angezeigt. Ist eine Datei nicht darstellbar, wird nur der Download-Button angezeigt, sofern dieser nicht ausgeblendet ist.

The screenshot shows the configuration interface for the 'Datei-Anzeige' component. It features a 'file*' label and a large light gray area with a blue plus icon and the text 'Click to add an asset or drag and drop one in this area'.

Datei-Download

Es besteht auch die Möglichkeit, Dateien zum Download zur Verfügung zu stellen. Diese Komponente ermöglicht den Download einer jeden angegebenen Datei.

The screenshot shows the configuration interface for the 'Datei-Download' component. It features a 'file*' label and a large light gray area with a blue plus icon and the text 'Click to add an asset or drag and drop one in this area'.

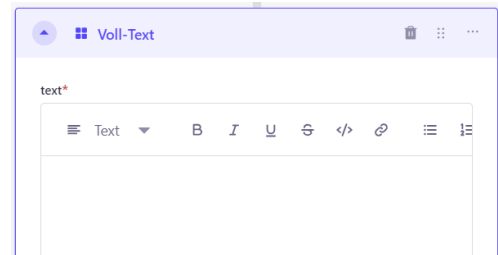
Galerie

Die Komponente Galerie ermöglicht die Anzeige mehrerer Bilder in einer Gitteransicht.

The screenshot shows the configuration interface for the 'Galerie' component. It features an 'images*' label and a large light gray area with a blue plus icon and the text 'Click to add an asset or drag and drop one in this area'.

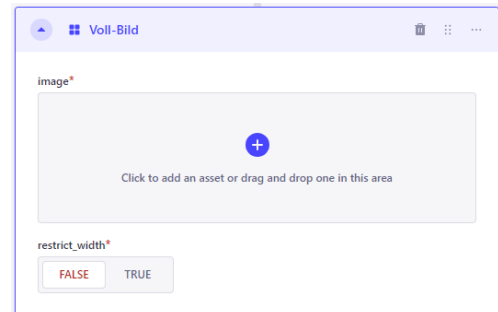
Voll-Text

Die reine Text-Komponente ermöglicht das schreiben längerer Berichte. So zum Beispiel auch das Impressum oder die Datenschutzerklärung.



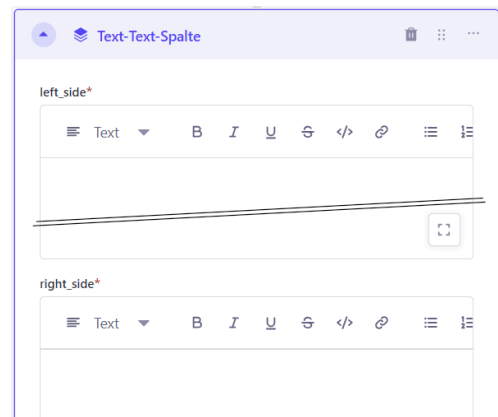
Voll-Bild

Einzelne Bilder können auch auf Größe einer ganzen Seite angezeigt werden. Zusätzlich kann eingestellt werden, ob Bilder auf großen Bildschirmen nur 50% der Breite nutzen dürfen.



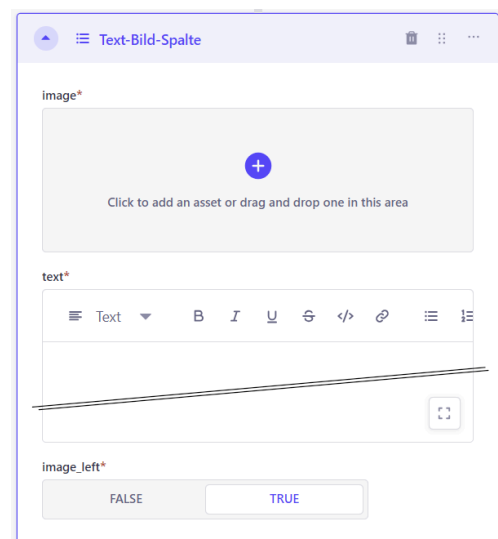
Text-Text-Spalte

Neben dem 1-spaltigen Layout gibt es auch die Möglichkeit 2 Spalten nebeneinander anzuzeigen zu können. Auf kleineren Bildschirmen werden die Texte übereinander angezeigt.



Text-Bild-Spalte

Erweitert kann auch ein Bild in einer Spalte links oder rechts angezeigt werden. Auch hier werden die Inhalte übereinander angezeigt, wenn die Webseite auf kleineren Bildschirmen angezeigt wird.



Karten-Gruppe

Die Karten-Gruppe zeigt mehrere Karten nebeneinander in einem umbrechenden Raster an. Die Anzahl der Karten pro Zeile (2, 3 oder 4) lässt sich einstellen. Jede Karte kann optional ein Schlagwort, ein Bild, einen Text und einen Link am Ende enthalten.

The screenshot shows the configuration for a 'Karten-Gruppe' (Cards Group) widget. At the top, there is a 'columns*' dropdown menu set to '3'. Below it, a note states: 'Anzahl der Karten nebeneinander pro Zeile (auf kleinen Bildschirmen automatisch weniger)'. The main configuration area is titled 'cards (1)*' and contains several fields: 'title*' (with a sub-note 'Titel der Karte.'), 'eyebrow' (with a sub-note 'Optionales kurzes Schlagwort über dem Titel (z. B. "Neu").'), 'text' (with a sub-note 'Kurzer Beschreibungstext der Karte.'), 'image' (with a sub-note 'Bild der Karte.' and a placeholder image with a '+' icon and the text 'Click to add an asset or drag and drop one in this area'), and 'link' (with a sub-note 'No entry yet. Click to add one.' and a '+' icon). At the bottom right of the configuration area is a '+ Add an entry' button.

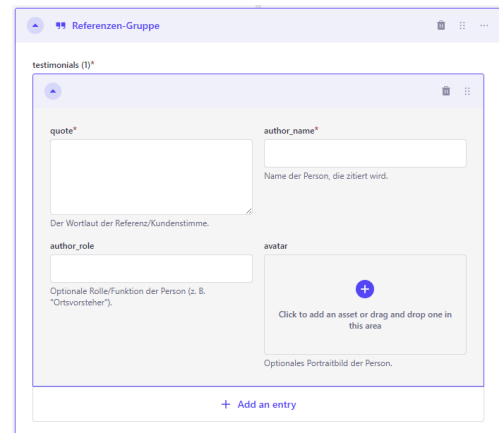
Tabelle

Die Tabelle ermöglicht die Darstellung tabellarischer Daten mit optionalen Spaltenüberschriften. Über die Einstellung „erste Spalte als Kopfzeile“ lässt sich festlegen, ob die erste Spalte (die Zeilenbezeichnung) wie eine Kopfzeile hervorgehoben wird oder eine ganz normale Datenspalte ist. Einzelne Zellen können zusätzlich hervorgehoben werden. Bei vielen Spalten kann die Tabelle auf kleinen Bildschirmen horizontal gescrollt werden.

The screenshot shows the configuration for a 'Tabelle' (Table) widget. At the top, there is a 'first_column_header*' toggle switch set to 'TRUE'. Below it, a note states: 'Wenn aktiv, wird die erste Spalte (die Zeilen-Bezeichnung) optisch hervorgehoben wie eine Kopfzeile. Deaktivieren, wenn die erste Spalte ganz normale Daten enthalten soll.' The main configuration area is titled 'columns (1)' and contains a 'label*' field with a sub-note 'Überschrift dieser Spalte in der Kopfzeile der Tabelle.' and a '+ Add an entry' button. Below this is the 'rows (1)*' section, which contains a 'label*' field with a sub-note 'Bezeichnung dieser Zeile (erste Spalte).', a 'cells (1)*' section, and a '+ Add an entry' button. The 'cells (1)*' section contains a 'value*' field with a sub-note 'Inhalt dieser Tabellenzeile.' and a 'highlighted*' toggle switch set to 'FALSE' with a sub-note 'Hebt diese Zeile optisch hervor.' At the bottom right of the configuration area is a '+ Add an entry' button.

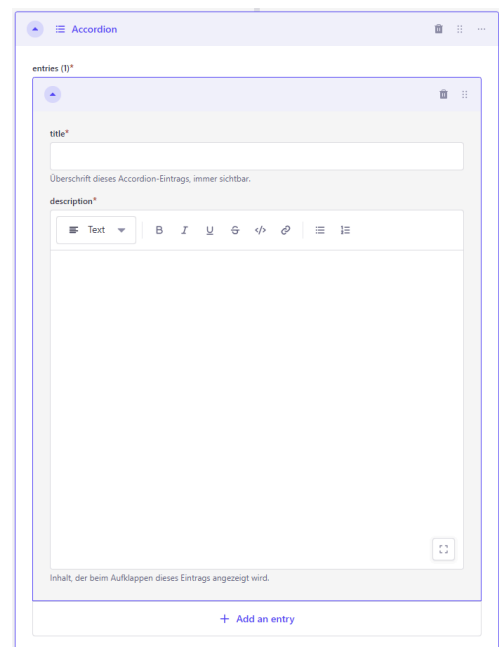
Referenzen-Gruppe

Die Referenzen-Gruppe zeigt mehrere Referenzen (Kundenstimmen) nebeneinander an. Jede Referenz besteht aus einem Zitat, dem Namen der zitierten Person, optional deren Rolle/Funktion und einem optionalen Portraitbild.



Accordion

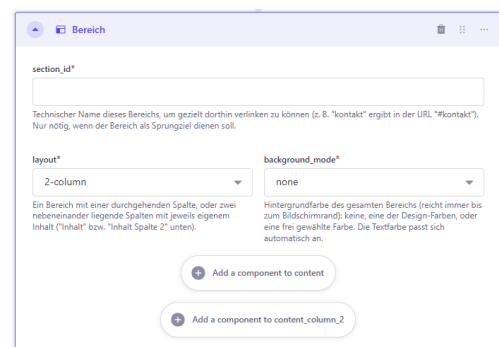
Das Accordion zeigt mehrere auf- und zuklappbare Einträge untereinander an. Jeder Eintrag besitzt eine immer sichtbare Überschrift und einen Inhalt, der erst beim Anklicken angezeigt wird - geeignet z.B. für häufig gestellte Fragen.



Bereich

Der Bereich ist ein Container für beliebig viele weitere Bausteine (ein weiterer Bereich kann nicht verschachtelt werden) und kann als 1- oder 2-spaltiges Layout dargestellt werden. Bei 2 Spalten wird auf kleineren Bildschirmen automatisch übereinander statt nebeneinander dargestellt.

Der Bereich kann zusätzlich einen Hintergrund erhalten: keinen, eine Design-Farbe (Abschnitt 3.2.2) oder eine frei gewählte Farbe. Der Hintergrund reicht dabei immer bis zum Bildschirmrand, während der Inhalt auf der normalen Breite der übrigen Seite bleibt. Die Textfarbe innerhalb des Bereichs



sowie darin verwendete Buttons (siehe Button-Gruppe) passen sich automatisch an den gewählten Hintergrund an, sodass immer ausreichend Kontrast gewährleistet ist.

Direkt aufeinanderfolgende Bereiche werden ohne zusätzlichen Abstand dazwischen dargestellt, damit farbige Flächen nahtlos ineinander übergehen.

Optional kann einem Bereich ein technischer Name (`section_id`) gegeben werden. Damit lässt sich gezielt zu diesem Bereich verlinken, z.B. über einen Link mit dem Ziel `#kontakt`, wenn `section_id` auf `kontakt` gesetzt wurde - die Seite springt dann automatisch zu diesem Bereich. Das ist z.B. nützlich für einen Link in der Navigation oder einem Button, der direkt zu einem bestimmten Bereich einer längeren Seite führen soll. Nur nötig, wenn der Bereich tatsächlich als Sprungziel dienen soll.

Button-Gruppe

Die Button-Gruppe zeigt eine Reihe von Buttons an. Jeder Button kann einzeln als hervorgehoben (gefüllt) oder zurückhaltend (nur umrandet) gestaltet werden. Die Ausrichtung der Buttons innerhalb der Zeile (links, mittig, rechts) lässt sich einstellen.

3.3.2 Shared

Artikel hervorheben

Mit dieser Komponente lassen sich Artikel hervorheben, welche besondere Aufmerksamkeit erhalten sollen. Dazu können dann auch noch ein Titel und eine Beschreibung angegeben werden.

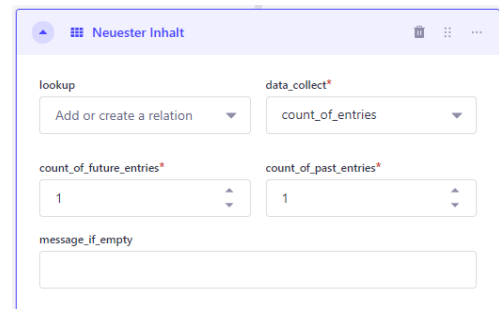
Weiterhin kann zu der URL weitergeleitet werden, unter der die Artikel-Anzeige ist.

Inhaltsauswahl

Die Inhaltsauswahl kann die Listen der Artikel, Einsätze, Fahrzeuge oder Termine anzeigen, sofern diese in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) mit der Anzeige konfiguriert wurden.

Neuester Inhalt

Neuester Inhalt zeigt eine Liste von Artikeln, Einsätzen, Fahrzeugen oder Terminen an, sofern diese in der Sammlungsreferenz ([Abschnitt 3.2.4](#)) mit der Anzeige konfiguriert wurden. Hier kann konfiguriert werden, wie viele Einträge relativ zum aktuellen Datum angezeigt werden. Die Filterung erfolgt entweder durch eine feste Anzahl an Einträgen oder durch einen Zeitraum in Wochen vor bzw. nach dem aktuellen Tag.



The screenshot shows the 'Neuester Inhalt' (Latest Content) configuration panel in Strapi. It includes the following fields:

- lookup**: A dropdown menu with the option 'Add or create a relation'.
- data_collect***: A dropdown menu with the option 'count_of_entries'.
- count_of_future_entries***: A numeric input field with the value '1'.
- count_of_past_entries***: A numeric input field with the value '1'.
- message_if_empty**: A text input field.

4 Ökosystem FF Admin

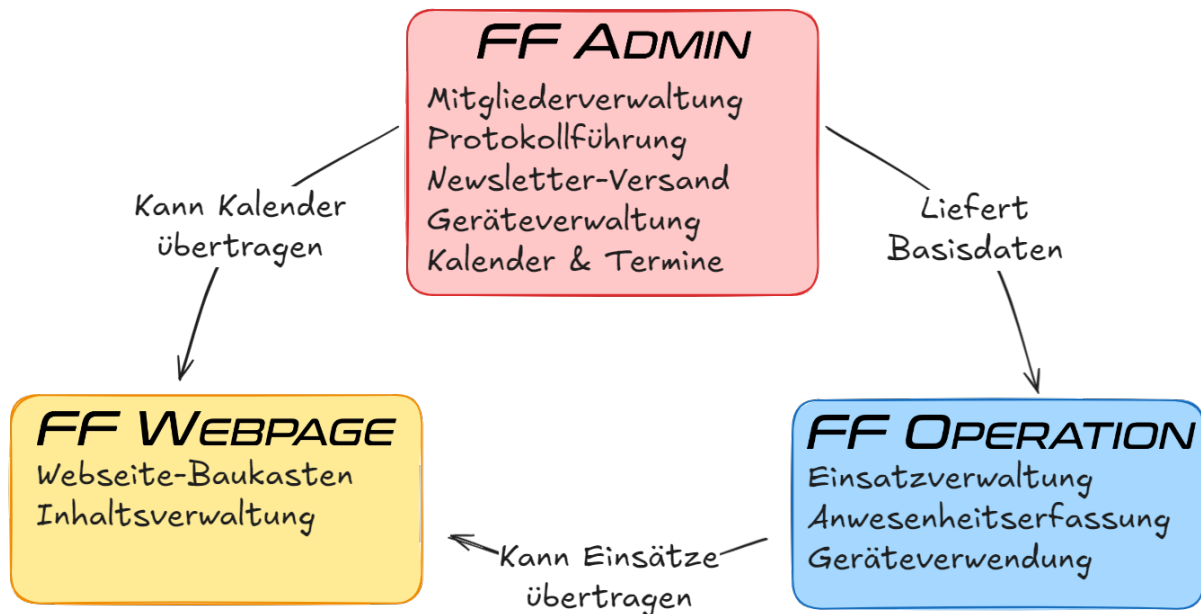


Abbildung 48: Übersicht zum Ökosystem

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 veröffentlicht 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.