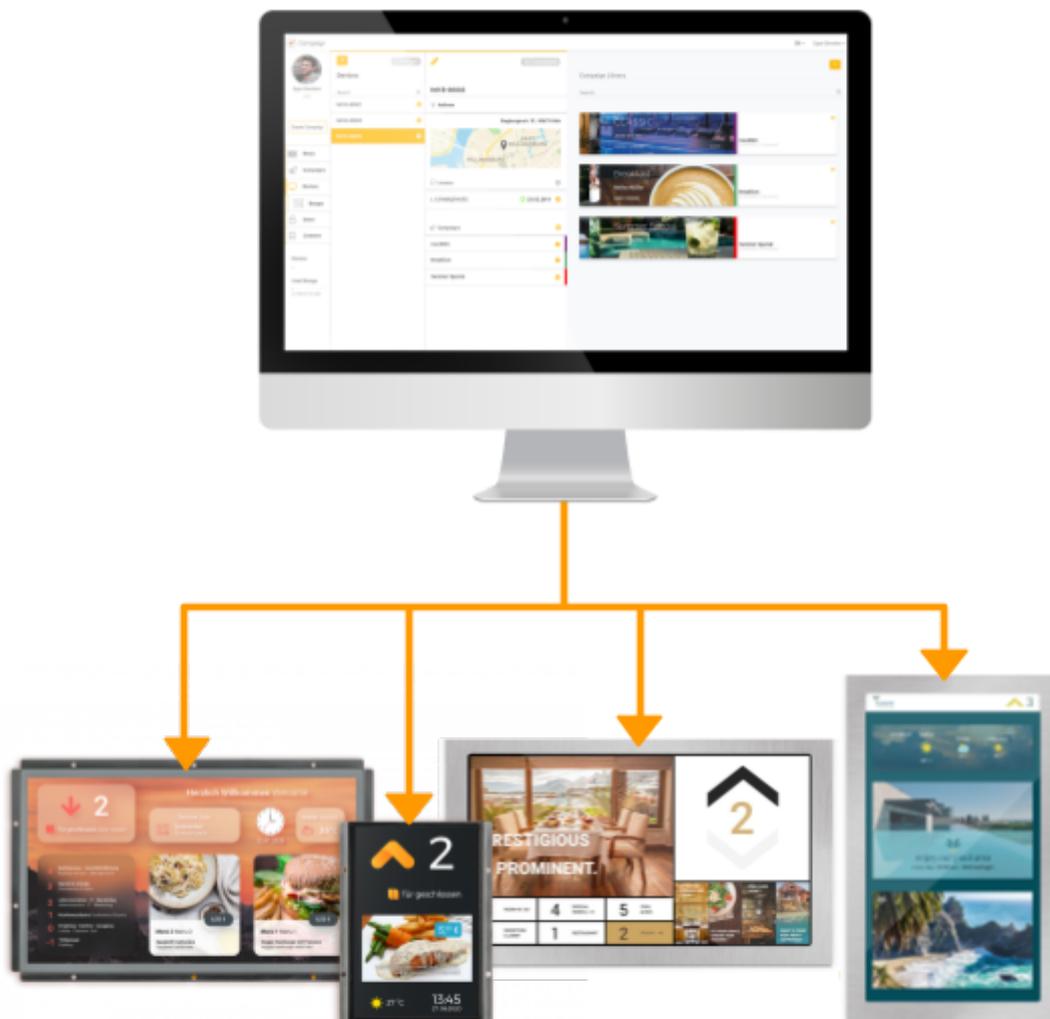


Protocol definition

flexyPage Campaign

for centralized distribution of multi media information

Version 0.2 - Draft proposal



Inhalt

A brief overview	3
Notes on flexyPage documentation	4
Protocol Definition flexyPage Campaign	5
Overview	5
JSON Document Format	5
view	5
playlist	5
layout	6
trigger	6
asset	7
media	7
sound asset	7
sound media	7
Document history	8
Your contact persons	8

A brief overview

flexyPage is a modern, flexible system for the simultaneous display of lift information and multimedia presentations inside and outside elevators and as a digital signage system in buildings.

The flexyPage displays are available in different sizes (from 7" to 65") and designs. The integrated CAN interface can be used to receive the elevator's field bus data. Various touch sensors or buttons can be connected for the input of commands by the user. Configuration, connection to the Internet and networking with building management systems is done via the LAN interface or an integrated LTE module.

The displays can be used in all elevators for a wide variety of applications, both in new buildings and in retrofitting.



flexyPage Campaign is a content management system for the flexyPage displays. It enables centralized creation and distribution of multimedia campaigns on flexyPage Displays. The functionality of the displays and flexyPage Campaign is described in the user manuals.

In this document you will find the protocol definition of the flexyPage Campaign server.



Do you have questions or suggestions? Please contact us

sales@flexyPage.de.



Notes on flexyPage documentation



This manual describes the installation and server setup of the flexyPage Campaign Server System on a local server for in-house use. The manual helps administrators to check the requirements, installation and maintenance of the server software. It is only a part of the complete product documentation and is subject to continuous updating.

This document, the photos, graphics and videos as well as hardware and software are protected by copyright. They may not be copied or otherwise circulated without written permission. Translations also require written permission.

Contact for copies, translations and all similar requests is exclusively ELFIN Technology GmbH. The documentation is prepared with great care. Nevertheless, errors cannot be excluded. We are very pleased to receive suggestions and information about errors in our documentation. Please contact our sales or support team.

ELFIN Technology GmbH is not liable for damages, possible errors and their consequences which occur in connection with the delivery or use of this user manual.

Please read the documents carefully before use!



The current version of the manuals, product data sheets and other documents and application examples can be found on the product homepage at

Introduction to the flexyPage system	flexypage.de/en/documentation
Quick start guide	flexypage.de/en/documentation
Video instructions	flexypage.de/en/videos-and-tutorials
flexypage user manual	flexypage.de/en/user-documentation
Widget descriptions	flexypage.de/en/doc/widget-descriptions
Product data sheets	flexypage.de/en/documentation
Frequently Asked Questions	flexypage.de/en/faq
Contact for sales	flexypage.de/en/sales
Contact to support	flexypage.de/en/support

Protocol Definition flexyPage Campaign

Overview

flexyPage Campaign works via a pull system. The devices query a central service for updates via HTTP GET requests. The central service decides which content is shown when. The devices query in a regular interval. If the content is still up to date the server responds with a 304. Otherwise the server responds with a JSON document and a 200 status code.

JSON Document Format

The JSON document that is returned to the client is an object with key `views` that contains an array of view objects as a value. If no view needs to be currently displayed this array is empty.

```
{
  "views": [
    <view 1>,
    <view 2>
  ],
  "background": {
    "type": "image",
    "url": "<image-url>",
    "size": "(contain|stretch|cover)"
  },
  "sounds": [
    <playlist 1>,
    <playlist 2>
  ]
}
```

view

A view is an assembly of layouted visual media content that should be displayed for a certain period of time or at a certain elevator event.

In the protocol it is an object that defines a layout in which the content should be arranged (`layout`), the trigger that decides when the view should be displayed (`trigger`), and the assets that are shown in this layout (`assets`).

```
{
  "layout": <layout>,
  "trigger": <trigger>,
  "assets": [
    <asset 1>
    <asset 2>
  ]
}
```

playlist

A playlist is the auditory equivalent to a view. It contains the sounds that should be played on the device and a trigger defining when the sound should be played.

```
{
  "trigger": <trigger>,
  "sounds": [
    <sound asset 1>
    <sound asset 2>
  ]
}
```

layout

The layout object defines a layout type. Currently supported is fullscreen but alternatives like three-column, three-rows, sidebar-left are possible. The used object is open to extension.

```
{
  "type": "fullscreen"
}
```

trigger

The trigger element defines for how long (duration) and at which time (days, date, time) or elevator state (event) the view should be shown.

duration defines the number of seconds for which the containing view should be shown.

event is a string that defines the elevator event, e.g. DirectionUp, at which the view should be shown.

days defines that days of week at which the view should be displayed. An empty array means no restriction.

date defines when the view should be displayed by date. Currently that means the full day.

time defines from which time of the day on the view should be displayed.

```
{
  "duration": <number of seconds>,
  "event": <event string>,
  "days": [ "Mon", "Tue" (, etc.) ],
  "date": [ <start date string>,
            <end date string> ],
  "time": [ <start time string>,
            <end time string> ]
}
```

asset

An asset is a visual media element that is assigned to a view in a campaign. It defines the media, the amount of time (duration) it should be shown and a zone in which it should be displayed. In a multi zone layout like three-columns zones like left, center, and right can be used.

```
{
  "zone": "center"
  "duration": <number of seconds>,
  "media": <media>
}
```

media

The visual media element that contains the path to the online media resource and a type that hints to the target device what media type it is.

```
{
  "type": "<image|video|rss>"
  "path": "<url/path to source>"
}
```

sound asset

A sound asset is an element that is assigned to a playlist. The duration it is played is inherent to the media.

```
{
  "media": <sound media>
}
```

sound media

The auditory media element that contains the path to the online media resource and a type that hints to the target device its media type. In this case just sound.

```
{
  "type": "sound"
  "path": "<url/path to source>"
}
```

Document history

Version	Release Date	Changes
0.1	2019-05-17	pre release
0.2	2020-10-27	some customer specific functions added

Your contact persons

Even an extensive documentation cannot answer all questions. Do you have questions or suggestions concerning our flexyPage system? We look forward to your requests. You can contact us at:

ELFIN Technology GmbH

Siegburger Strasse 215

50679 Cologne

Germany

Phone: +49 (221) 6778932-0

FAX: +49 (221) 6778932-2

service@elfin.de

www.elfin.de



Sales flexyPage

flexypage.de/en/sales

Phone: +49 (221) 6430816-2

FAX: +49 (221) 6778932-2

sales@flexyPage.de



innovative display solutions

Support flexyPage

flexypage.de/en/support

Phone: +49 (221) 6430816-3

support@flexyPage.de