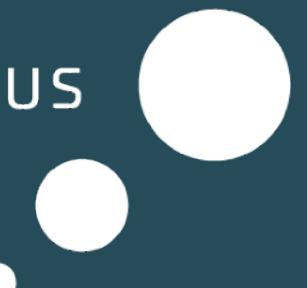


# Headless CMS

Yet another Data Source



# Headless CMS

Your own data & content available through an  
API





# What's a Headless CMS?



A CMS without a head 🤔

# I'm not kidding

It's a CMS without any frontend



# Headless CMS

...is a system to control the content that is separated from the display layer or the front-end user experience.

What  
Is Headless CMS?



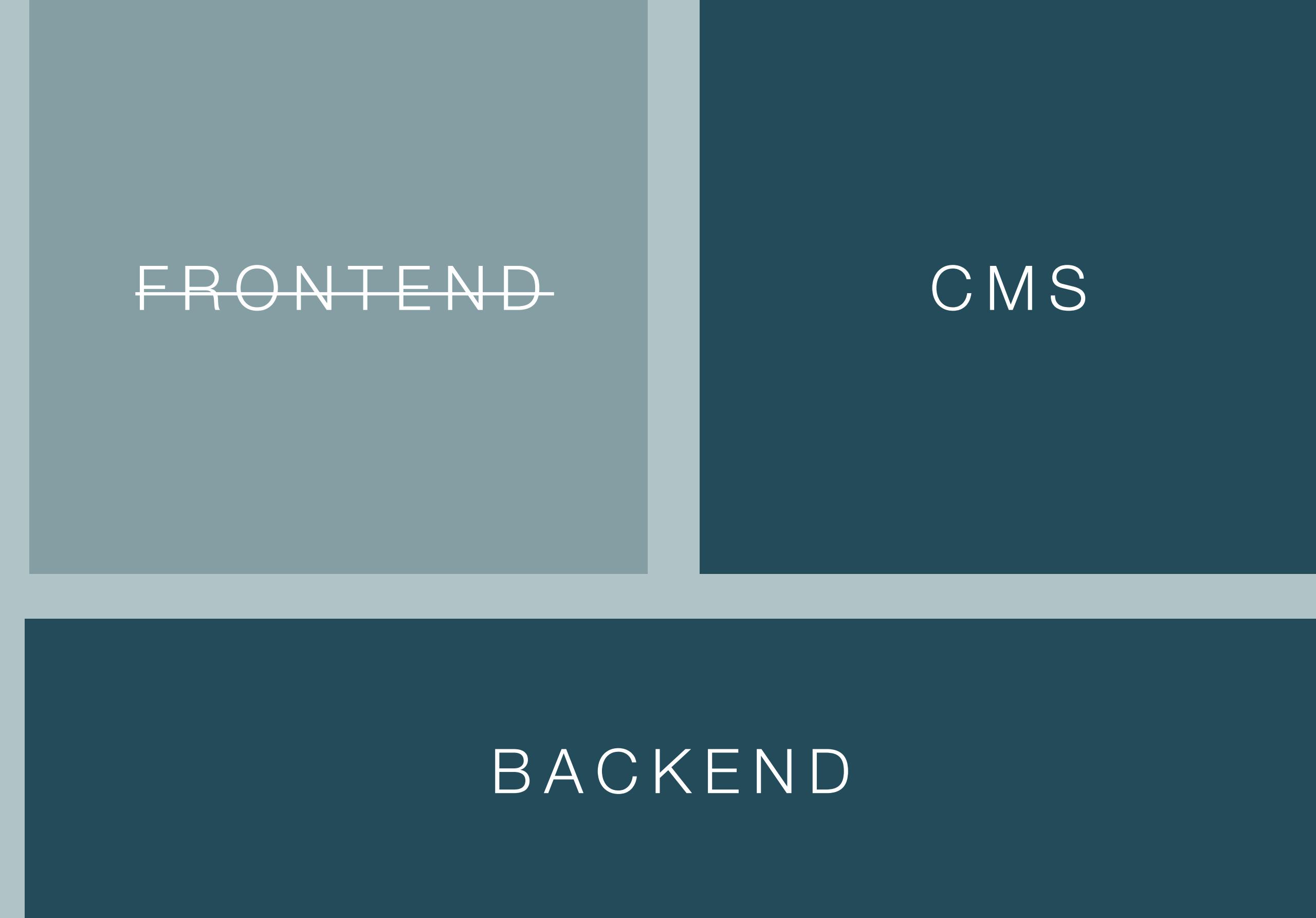
# Traditional CMS

FRONTEND

CMS

BACKEND

# Headless CMS

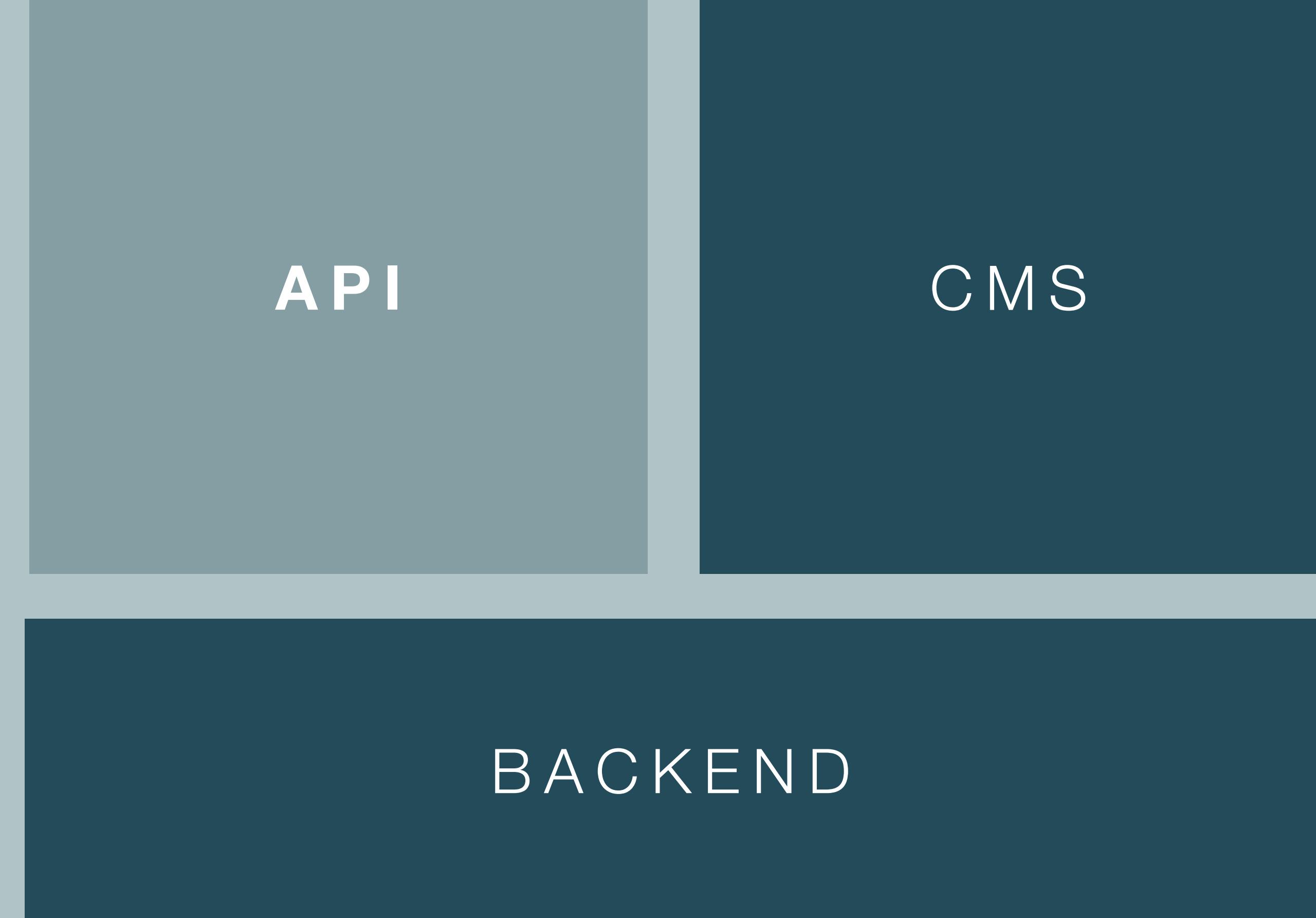


FRONTEND

CMS

BACKEND

# Headless CMS



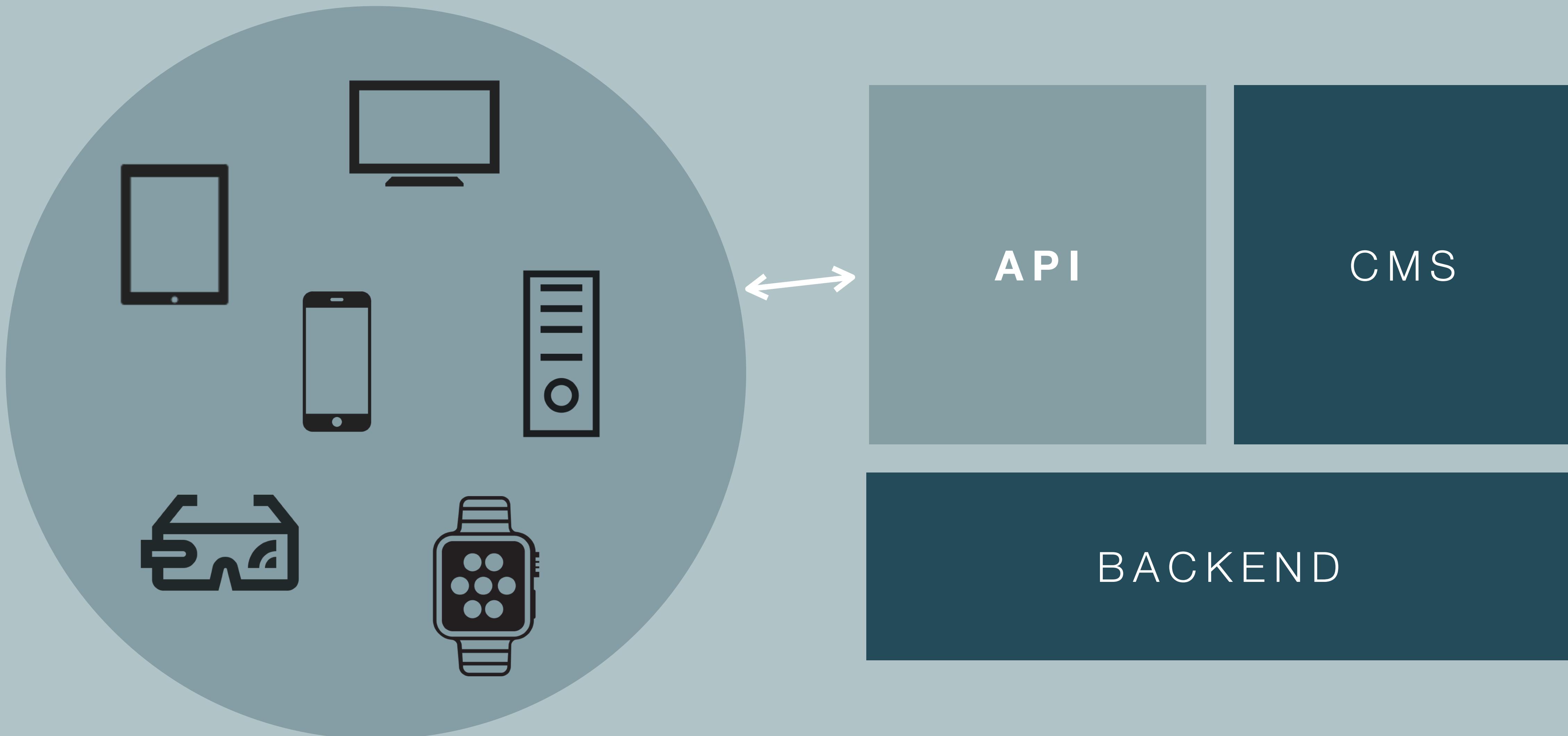
The diagram illustrates the architecture of a Headless CMS. It features three main components arranged vertically: a light gray rectangle labeled "API" at the top, a dark blue rectangle labeled "CMS" in the middle, and a dark teal rectangle labeled "BACKEND" at the bottom.

API

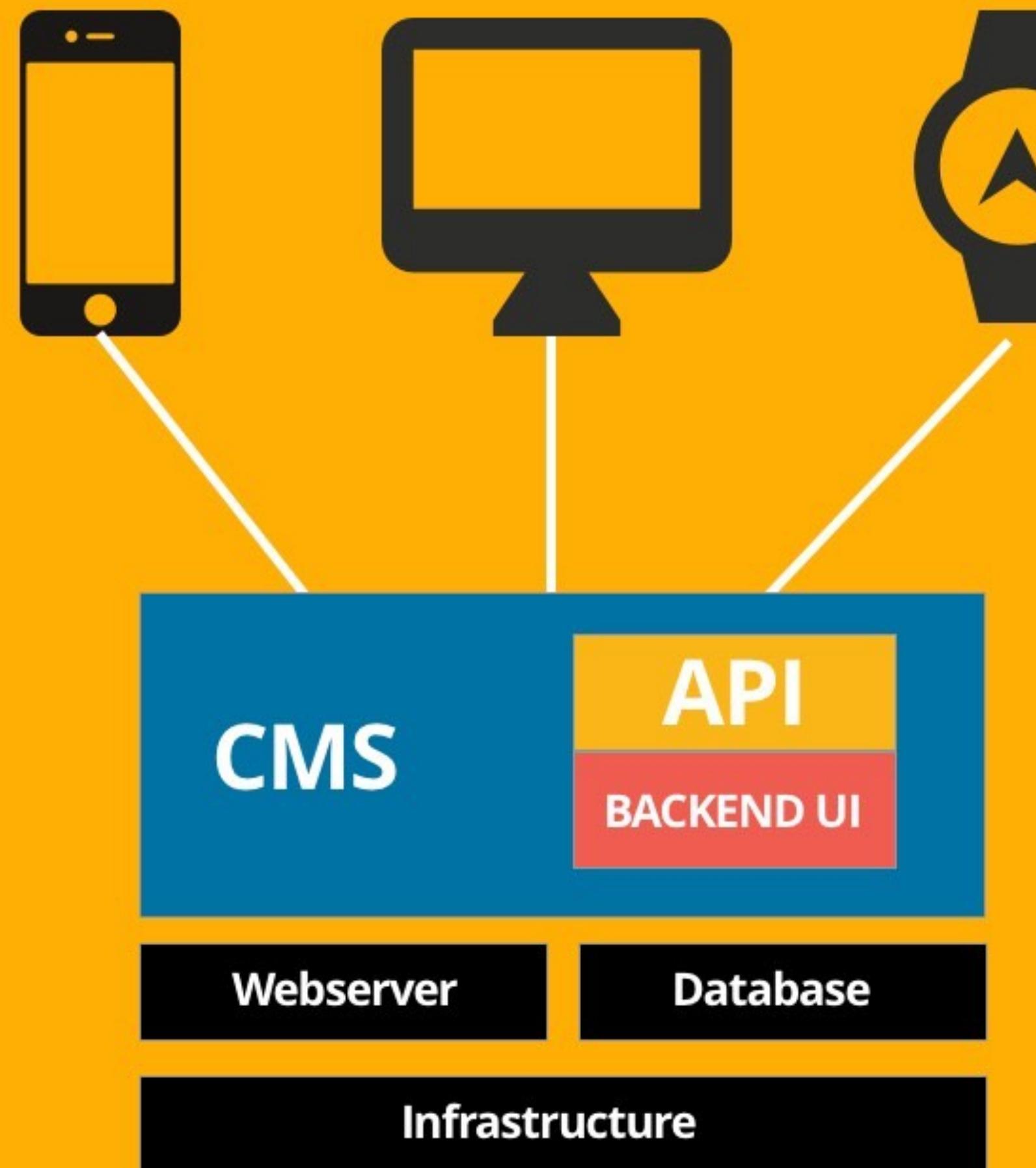
CMS

BACKEND

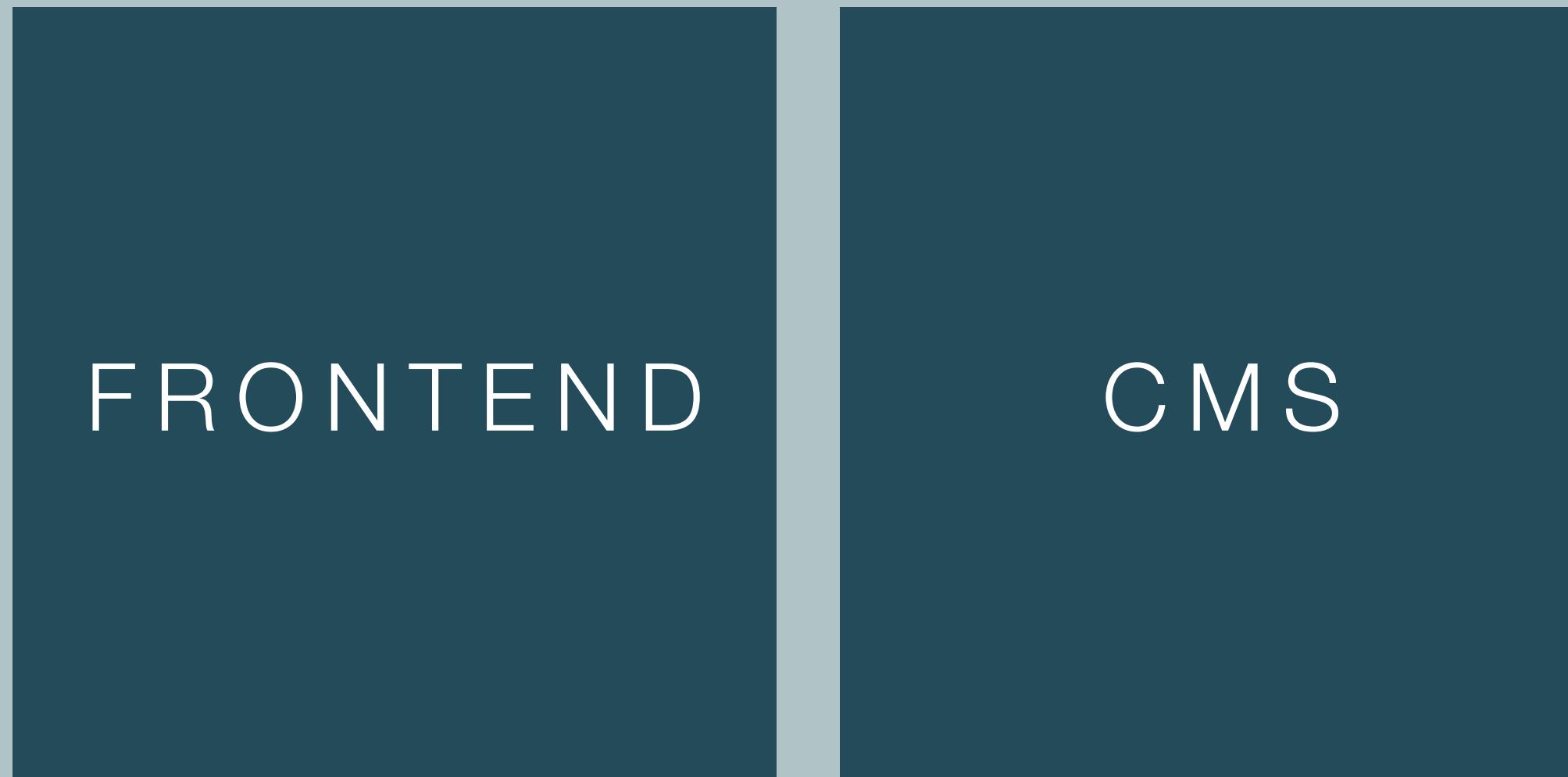
# Headless CMS



# Headless CMS



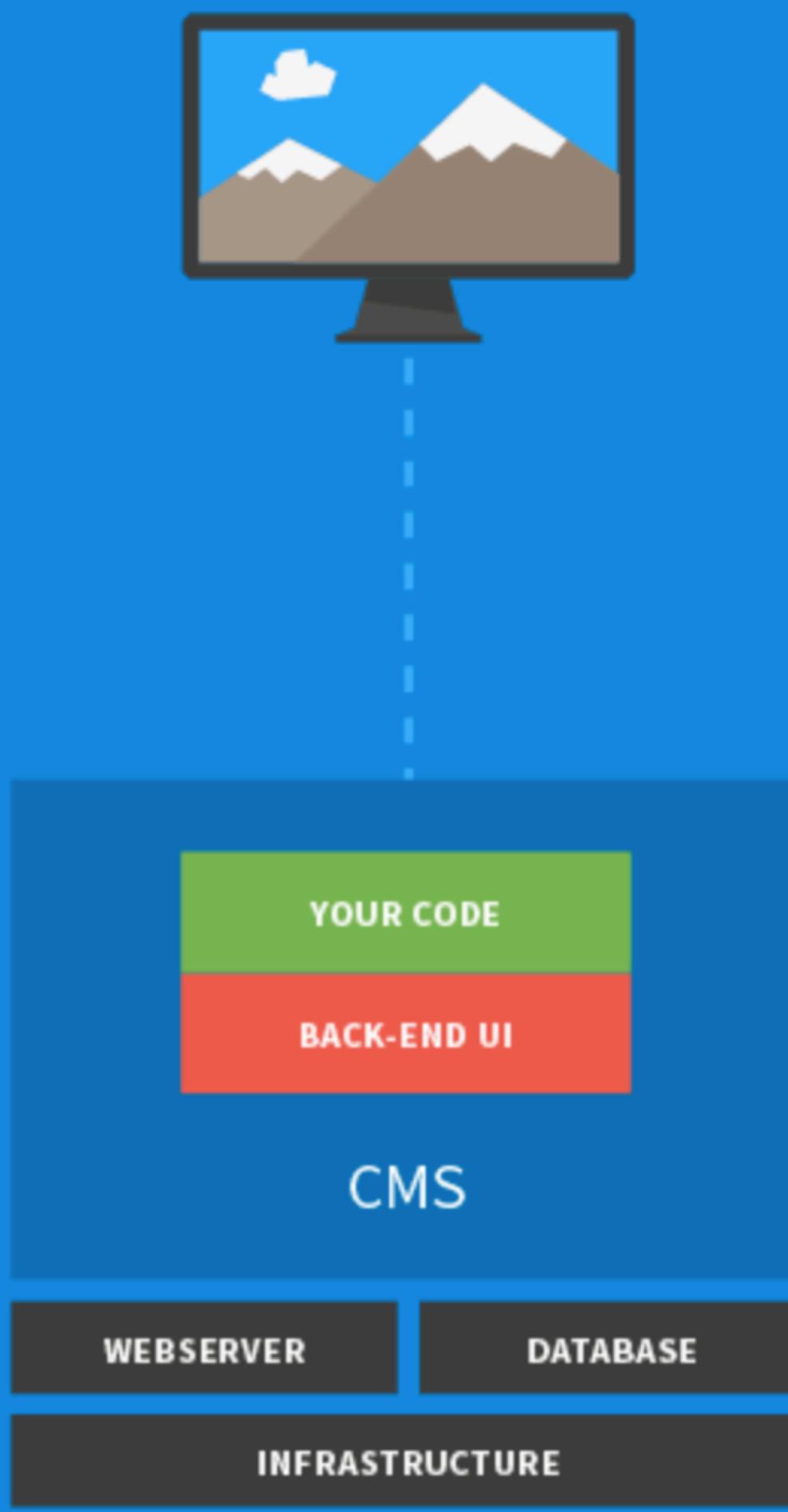
# Traditional CMS



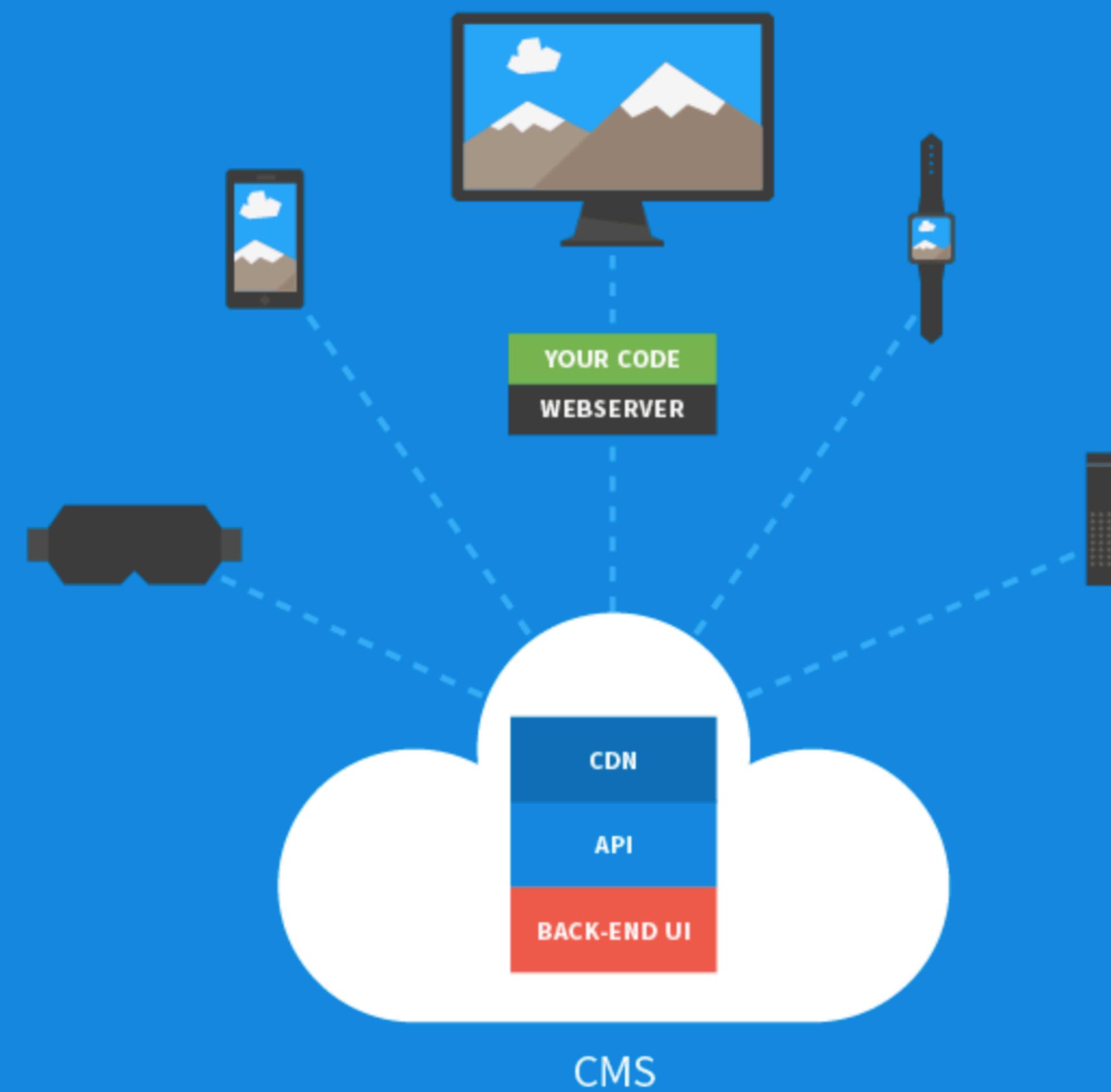
# Headless CMS



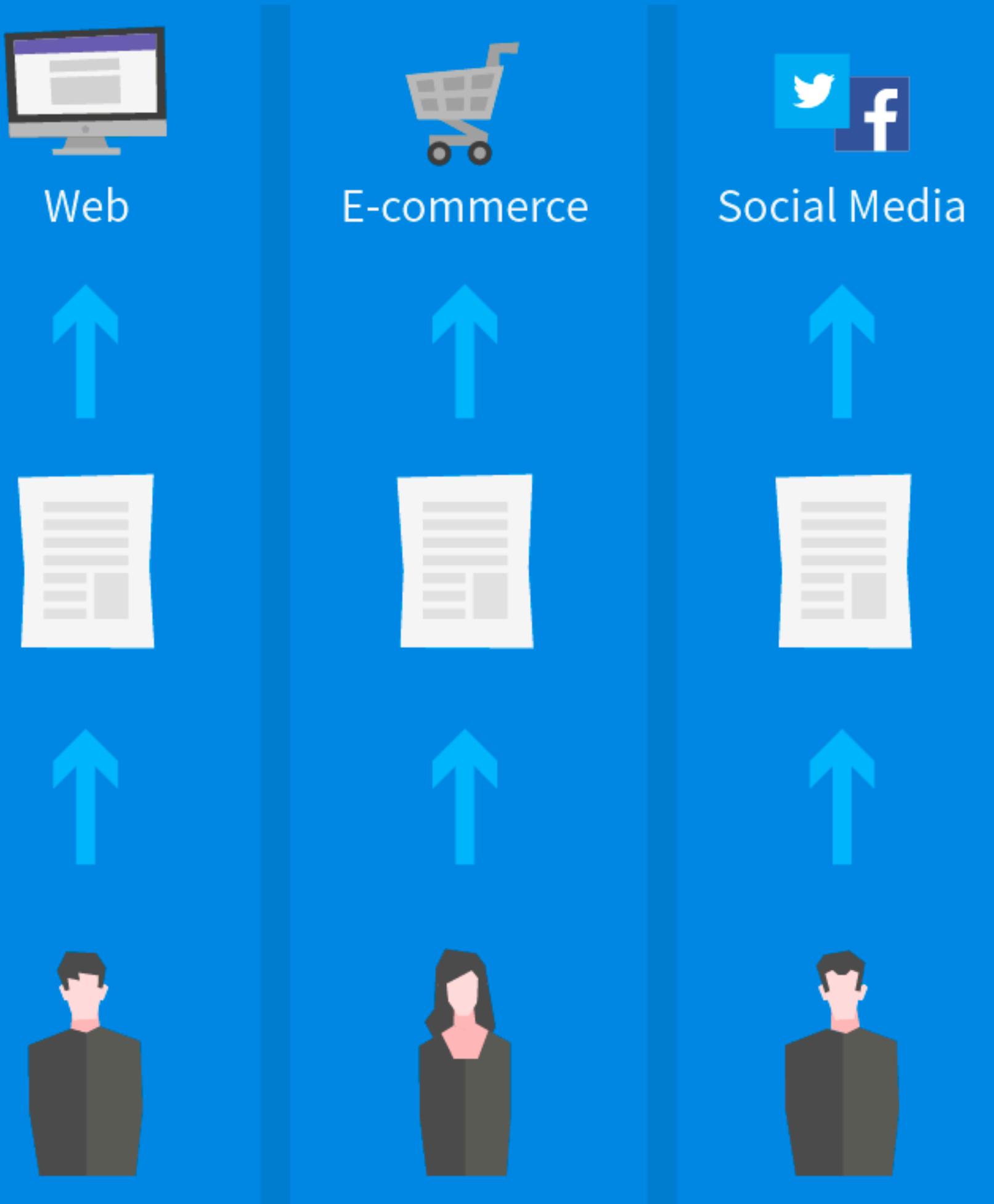
# TRADITIONAL CMS



# HEADLESS CMS



## TRADITIONAL CMS



## HEADLESS CMS



# TRADITIONAL CMS

A WAY TO STORE DATA

A CRUD UI

A WAY TO DISPLAY THE DATA

CRUD: CREATE, READ, UPDATE & DELETE

# HEADLESS CMS

A WAY TO STORE DATA

A CRUD UI

AN API TO ACCESS THE DATA

# WordPress

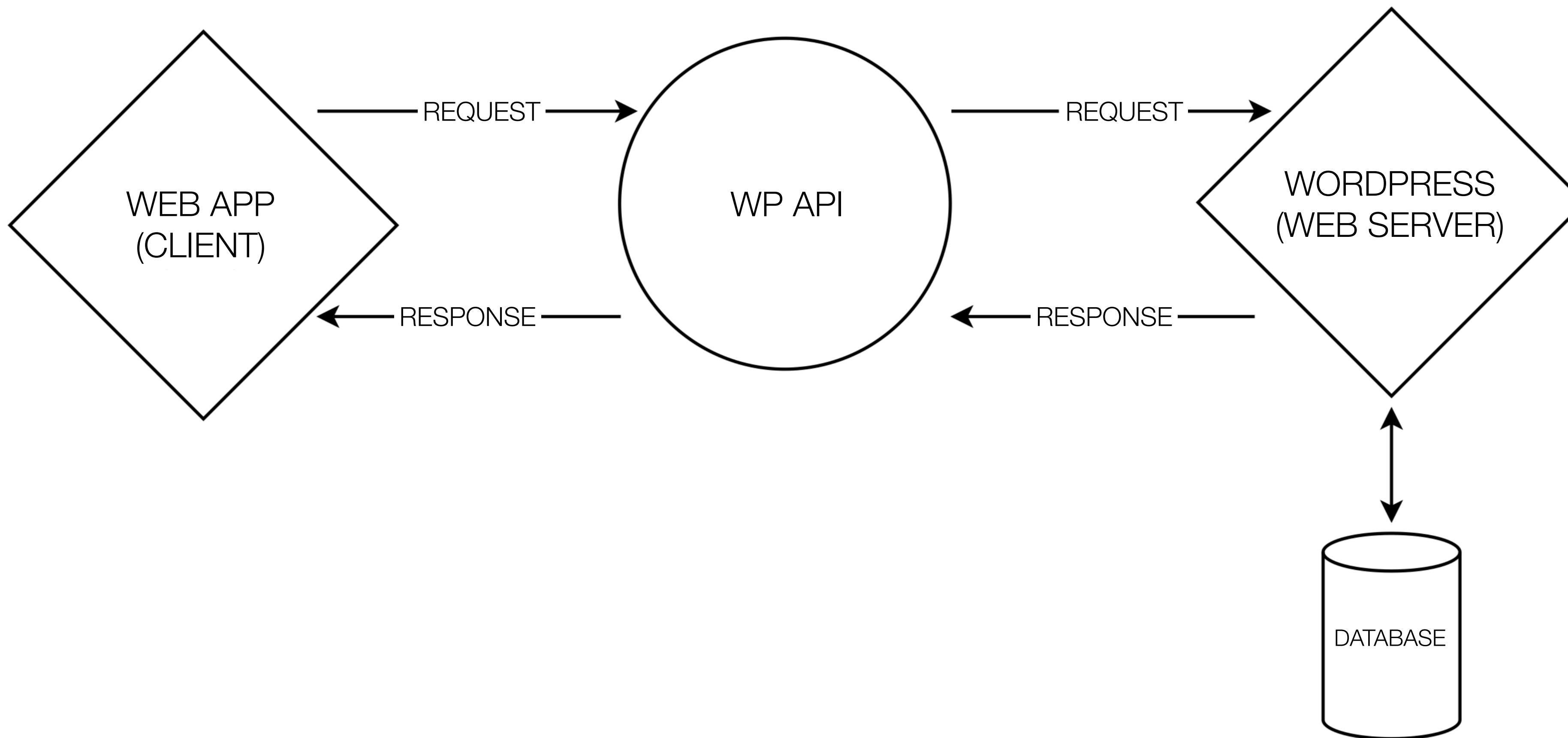
FRONTEND

API

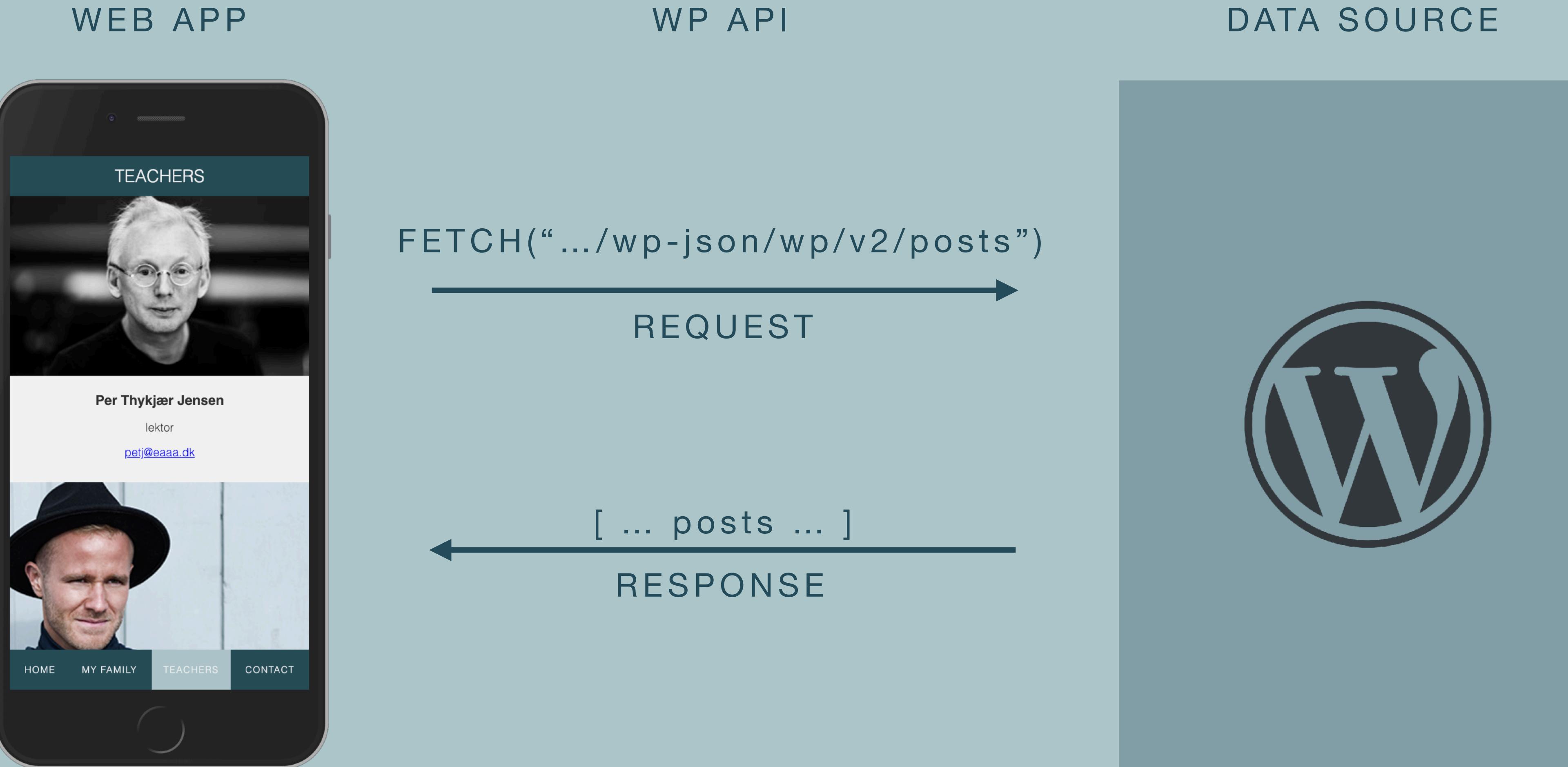
CMS

BACKEND

# APPLICATION PROGRAMMING INTERFACE



# Fetch & Headless CMS



Posts < MY HEADLESS CMS — +

← → C 🔒 headlesscms.cederdorff.com/wp-admin/edit.php

Howdy, mdu-front

Dashboard Posts All Posts Add New Categories Tags

WordPress 5.8 is available! Please notify the site administrator.

Posts Add New

All (8) | Published (8) | Trash (2)

Bulk actions ▾ Apply All dates ▾ All Categories ▾ Filter

	Title	Author	Categories	Tags	Comment	Date
<input type="checkbox"/>	Martin	admin	Persons	—	0 1	Published 2020/08/31 at 6:37 am
<input type="checkbox"/>	Peter Cederdorff	admin	Family Members	—	—	Published 2019/02/07 at 4:14 pm
<input type="checkbox"/>	Rasmus Cederdorff	admin	Family Members	—	0 3	Published 2019/02/07 at 4:12 pm
<input type="checkbox"/>	Alicia	admin	Family Members	—	0 3	Published 2019/02/07 at 11:34 am
<input type="checkbox"/>	Per Thykær	admin	Teachers	—	0 6	Published 2019/02/06 at 7:40 am
<input type="checkbox"/>	Rasmus Cederdorff	admin	Teachers	—	0 4	Published 2019/02/05 at 6:35 pm
<input type="checkbox"/>	Michael Hvidtfeldt	admin	Teachers	—	0 2	Published 2019/02/05 at 6:34 pm
<input type="checkbox"/>	Birgitte Kirk Iversen	admin	Teachers	—	0 8	Published 2019/02/05 at 6:32 pm
	Title	Author	Categories	Tags	Comment	Date

Bulk actions ▾ Apply

Screen Options ▾ Help ▾

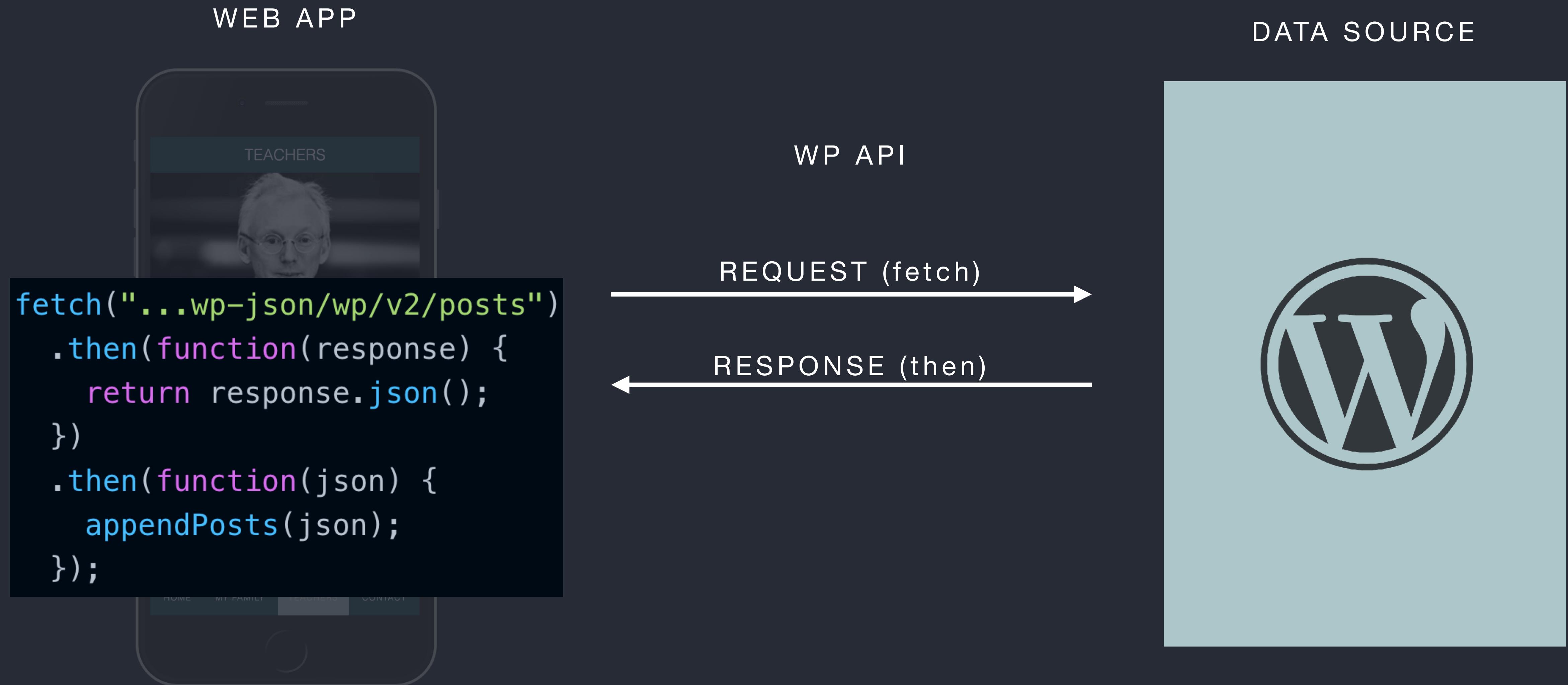
8 items

Thank you for creating with WordPress.

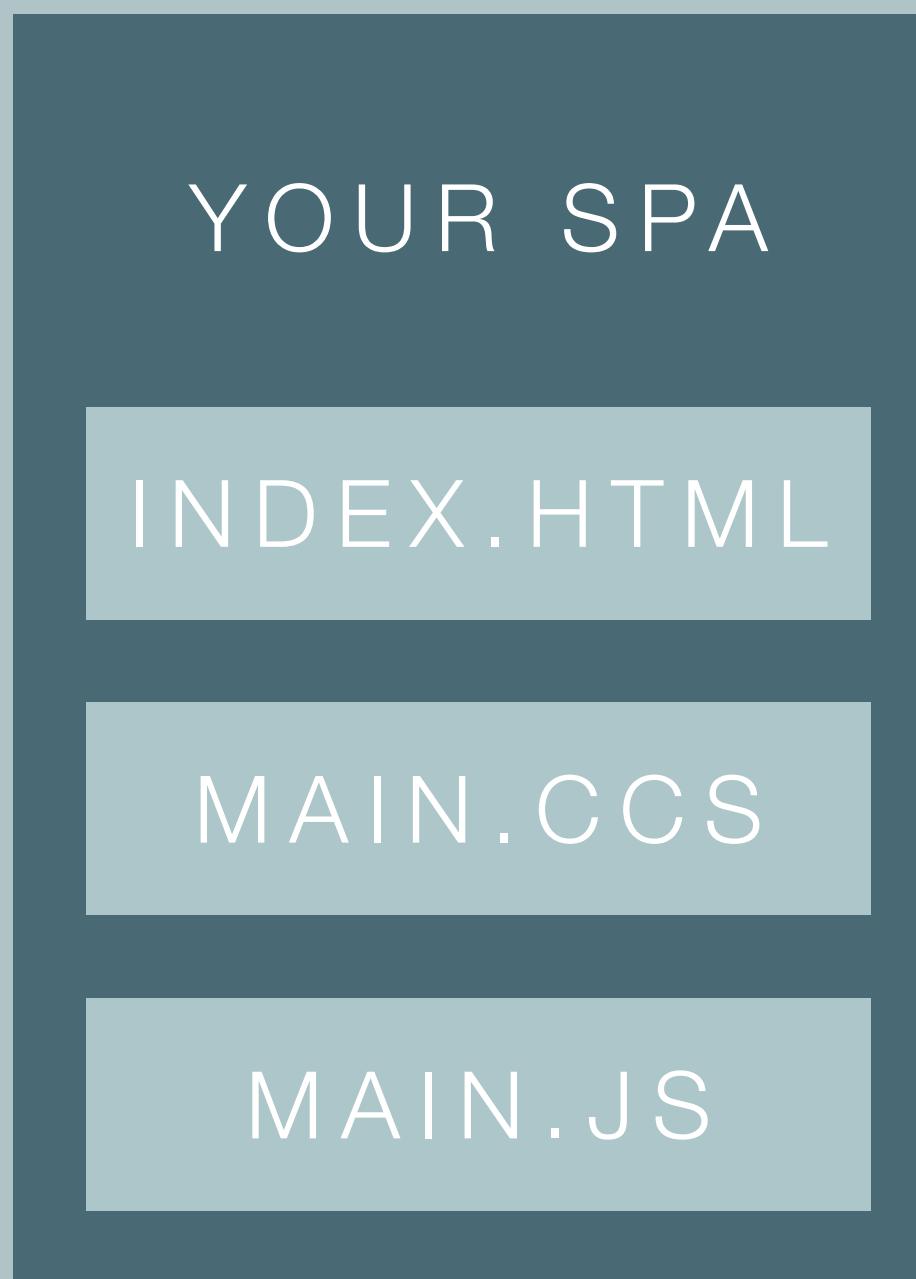
Version 5.5.5



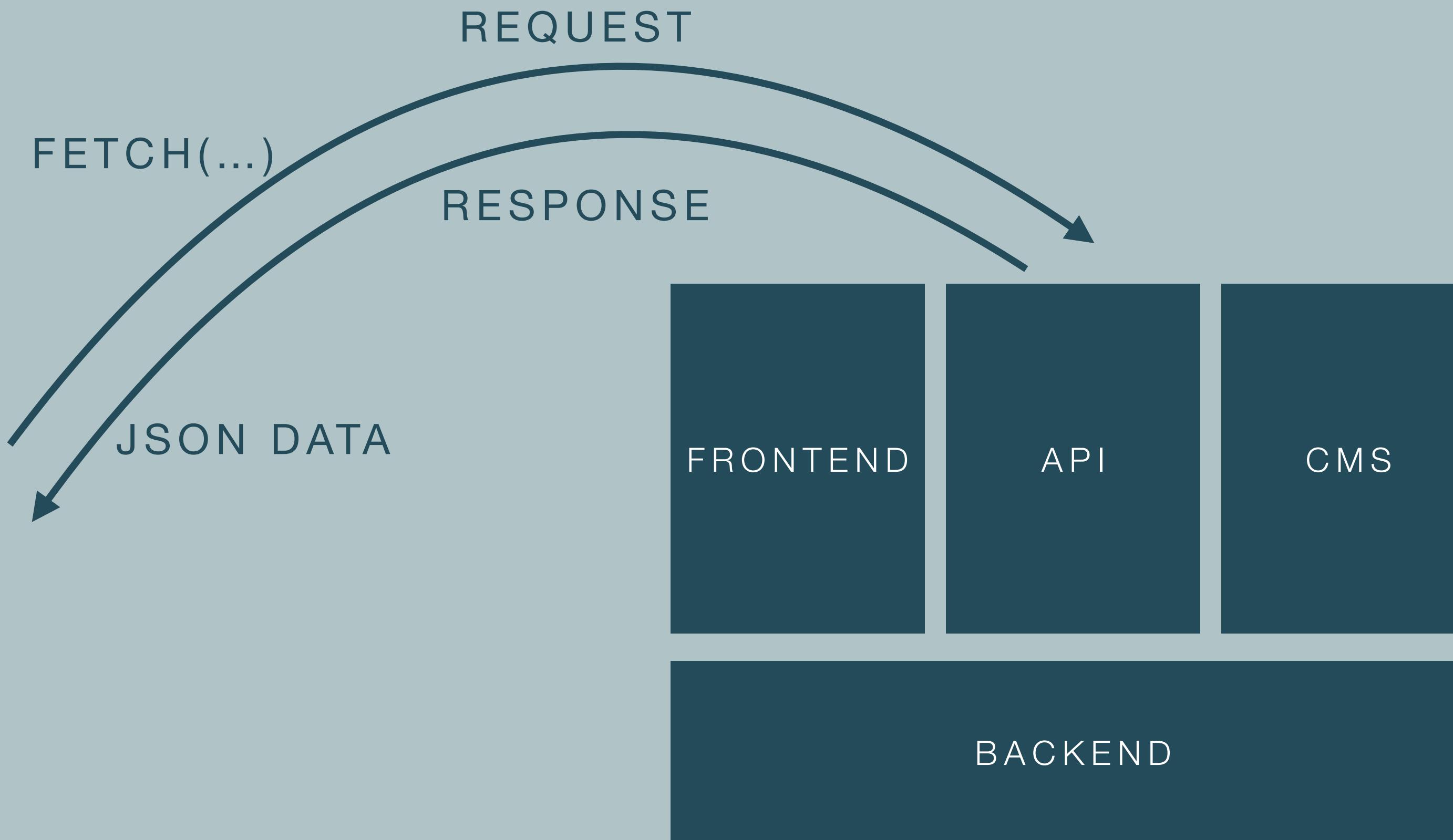
# FETCH & HEADLESS CMS



# WEB APP



# HEADLESS WP



[movie-api.cederdorff.com](http://movie-api.cederdorff.com)

# FETCH POSTS

The screenshot shows the WordPress admin dashboard with the title "Movie API" at the top. The "Posts" menu item is selected, highlighted in blue. A notification bar at the top right says "WordPress 5.3.2 is available! Please update now." Below the navigation, the "Posts" screen displays a list of three published posts: "Frozen", "Don't Let Go", and "Lion King". Each post has a checkbox next to its title, followed by the author (admin), categories (e.g., Animation, Comedy), and tags (e.g., —). The screen also includes filtering options like "Bulk Actions", "Apply", "All dates", "All Categories", and a "Filter" button.

	Title	Author	Categories	Tags
<input type="checkbox"/>	Frozen	admin	Animation, Comedy	—
<input type="checkbox"/>	Don't Let Go	admin	Drama, Horror	—
<input type="checkbox"/>	Lion King	admin	Animation	—
	Title ▲	Author	Categories	Tags

# FETCH POSTS

```
let _movies = [];

// fetch all movies from WP
async function getMovies() {
  let response = await fetch("https://movie-api.cederdorff.com/wp-json/wp/v2/posts?_embed");
  let data = await response.json();
  _movies = data;
  appendMovies(data);
  showLoader(false);
}

getMovies();
```

# HEADLESS PERSON EXERCISE

BACK

1. Copy and use the template: headless-cms-template
2. Fetch data from WP API: <https://headlesscms.cederdorff.com/wp-json/wp/v2/posts>
3. Append the fetched posts to the DOM. Display image, title and content.
4. Make sure you fetch the featured image by using `_embed` as argument in the URL
5. Add new posts in WordPress Admin and refresh your web app.

Admin: <https://headlesscms.cederdorff.com/wp-login>

Username: mdu-front

Password: headlessfront01

# WORDPRESS API

- From version 4.7 WP API is included in WordPress
- Install a fresh copy of WP
- Change Permalink Settings: POST NAME
- Test in the browser: <http://your-domain.com/wp-json/>
- Fetch JSON data from the API



# WORDPRESS API ROUTES

Meta data: <http://your-domain.com/wp-json/>

Posts: <http://your-domain.com/wp-json/wp/v2/posts>

Pages: <http://your-domain.com/wp-json/wp/v2/pages>

Media: <http://your-domain.com/wp-json/wp/v2/media>

Types: <http://your-domain.com/wp-json/wp/v2/types>

Taxonomies: <http://your-domain.com/wp-json/wp/v2/taxonomies>

Categories: <http://your-domain.com/wp-json/wp/v2/categories>

Tags: <http://your-domain.com/wp-json/wp/v2/tags>

Users: <http://your-domain.com/wp-json/wp/v2/users>

Comments: <http://your-domain.com/wp-json/wp/v2/comments>

DOCS: <https://developer.wordpress.org/rest-api/>

# EMBED FEATURED IMAGE

Posts with featured image:

[http://api.cederdorff.com/wp-json/wp/v2/posts? embed](http://api.cederdorff.com/wp-json/wp/v2/posts?embed)

Pages with featured image:

[http://api.cederdorff.com/wp-json/wp/v2/pages? embed](http://api.cederdorff.com/wp-json/wp/v2/pages?embed)

# BY CATEGORY

Posts by a specific category:

<https://api.cederdorff.com/wp-json/wp/v2/posts?categories=2>

**?categories=2**

Dashboard

Posts

All Posts

Add New

Categories

Tags

Media

Pages

Comments 4

Appearance

Plugins 2

Users

Tools

Settings

Collapse menu

## Edit Category

Name

Client

*The name is how it appears on your site.*

Slug

client

*The “slug” is the URL-friendly version of the name. It is usually all lowercase and contains only letters, numbers, and hyphens.*

Parent Category

None

*Categories, unlike tags, can have a hierarchy. You might have a Jazz category, and under that have children categories for Bebop and Big Band. Totally optional.*

Description

*The description is not prominent by default; however, some themes may show it.*

# CATEGORY + FEATURED IMAGE

Posts with featured image and by specific category:

[https://api.cederdorff.com/wp-json/wp/v2/posts?  
\\_embed&categories=2](https://api.cederdorff.com/wp-json/wp/v2/posts?_embed&categories=2)

**? \_embed&categories=2**

# FILTER, SORT & ORDER ROUTES

[https://api.cederdorff.com/wp-json/wp/v2/posts? embed](https://api.cederdorff.com/wp-json/wp/v2/posts?embed)

[https://api.cederdorff.com/wp-json/wp/v2/posts?  
embed&categories=2](https://api.cederdorff.com/wp-json/wp/v2/posts?embed&categories=2)

[https://api.cederdorff.com/wp-json/wp/v2/posts?  
orderby=date&order=desc&after=2019-00-10T17:00:00](https://api.cederdorff.com/wp-json/wp/v2/posts?orderby=date&order=desc&after=2019-00-10T17:00:00)

[https://api.cederdorff.com/wp-json/wp/v2/posts?  
orderby=date&order=asc](https://api.cederdorff.com/wp-json/wp/v2/posts?orderby=date&order=asc)

# EXAMPLES OF HEADLESS CMSS

<http://nanadorph.dk>

<http://www.sommerhusborkhavn.dk>

<https://headlesscms.cederdorff.com/>

<https://product-api.cederdorff.com/>

<https://api.cederdorff.com/>

<https://movie-api.cederdorff.com/>



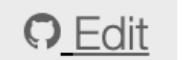
# REST API Handbook



## CHAPTERS

[REST API Handbook](#)[Key Concepts](#)[Frequently Asked Questions](#)[Using the REST API](#)[Extending the REST API](#)[Reference](#)[Changelog](#)Browse: [Home](#) / REST API Handbook

## REST API Handbook



The WordPress REST API provides an interface for applications to interact with your WordPress site by sending and receiving data as [JSON](#) (JavaScript Object Notation) objects. It is the foundation of the [WordPress Block Editor](#), and can likewise enable your theme, plugin or custom application to present new, powerful interfaces for managing and publishing your site content.

Using the WordPress REST API you can create a plugin to provide an entirely new admin experiences for WordPress, build a brand new interactive front-end experience, or bring your WordPress content into completely separate applications.

The REST API is a developer-oriented feature of WordPress. It provides data access to the content of your site, and implements the same authentication restrictions — content that is public on your site is generally publicly accessible via the REST API, while private content, password-protected content, internal users, custom post types, and metadata is only available with authentication or if you specifically set it to be so. If you are not a developer, the most important thing to understand about the API is that it enables the block editor and modern plugin interfaces without compromising the security or privacy of your site.

### What Is A REST API? #

An API is an Application Programming Interface. REST, standing for "REpresentational State

## TOPICS

[What Is A REST API?](#)[Using the WordPress REST API](#)[Next Steps](#)

# PROJECTS

headless-cms

headless-cms-acf-products

headless-cms-acf-teachers

headless-cms-template

spa-headless-cms-movies

spa-headless-cms-movies-template

spa-headless-cms-teachers-family-members

spa-headless-cms-template

# HEADLESS SPA EXERCISE

Template: spa-template

1. Create a SPA with the following views: All Persons, Family & Teachers.
2. Fetch data from the Headless CMS [headlesscms.cederdorff.com](https://headlesscms.cederdorff.com) and use the category id to fetch different post categories:
  1. All Persons: [https://headlesscms.cederdorff.com/wp-json/wp/v2/posts? embed](https://headlesscms.cederdorff.com/wp-json/wp/v2/posts?embed)
  2. Teachers: [https://headlesscms.cederdorff.com/wp-json/wp/v2/posts? embed&categories=2](https://headlesscms.cederdorff.com/wp-json/wp/v2/posts?embed&categories=2)
  3. [https://headlesscms.cederdorff.com/wp-json/wp/v2/posts? embed&categories=3](https://headlesscms.cederdorff.com/wp-json/wp/v2/posts?embed&categories=3)
3. Append the posts to the matching views in your SPA.
4. Implement a functional search field.
5. Order posts by name (title property).

Admin: <https://headlesscms.cederdorff.com/wp-login>

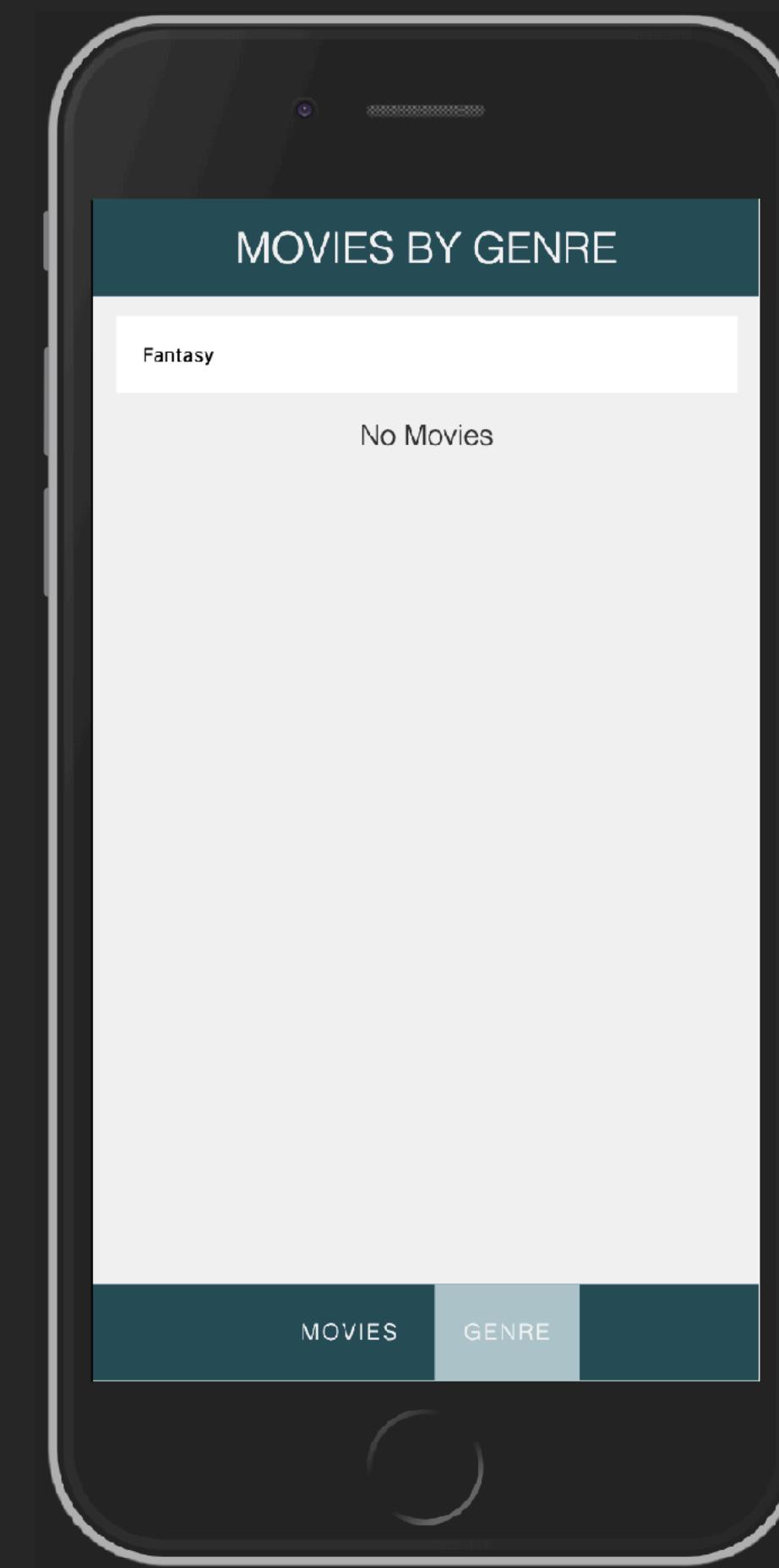
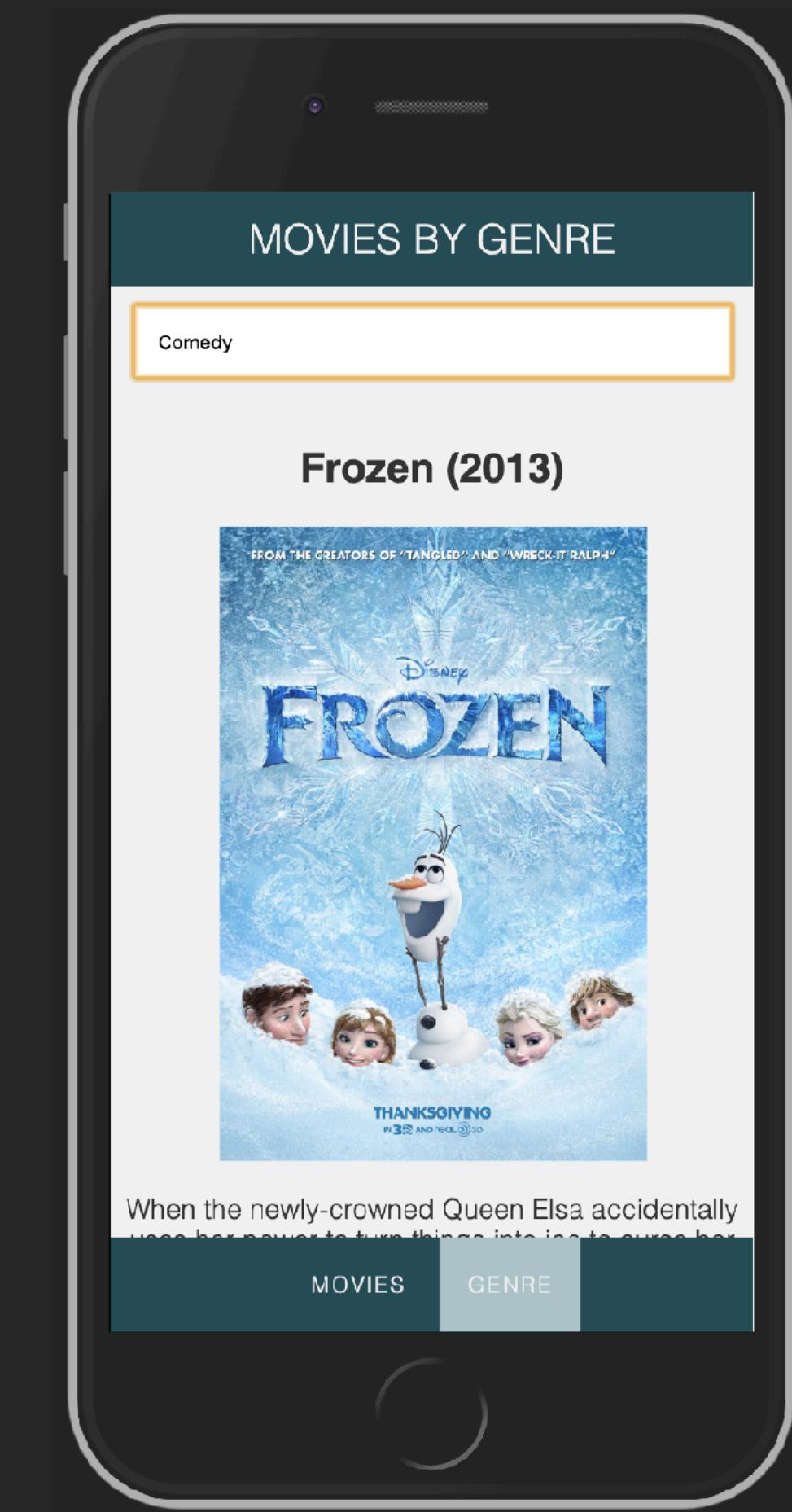
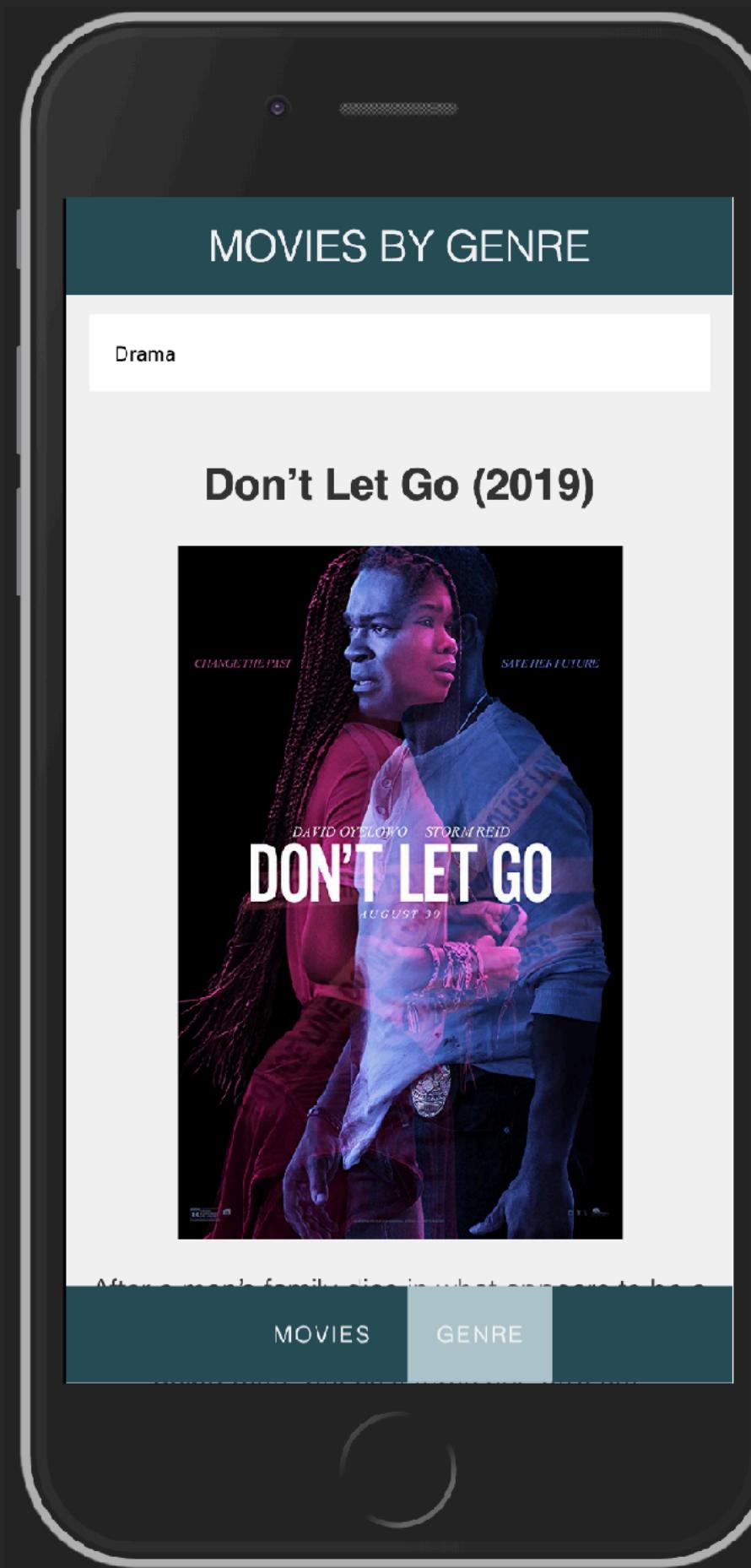
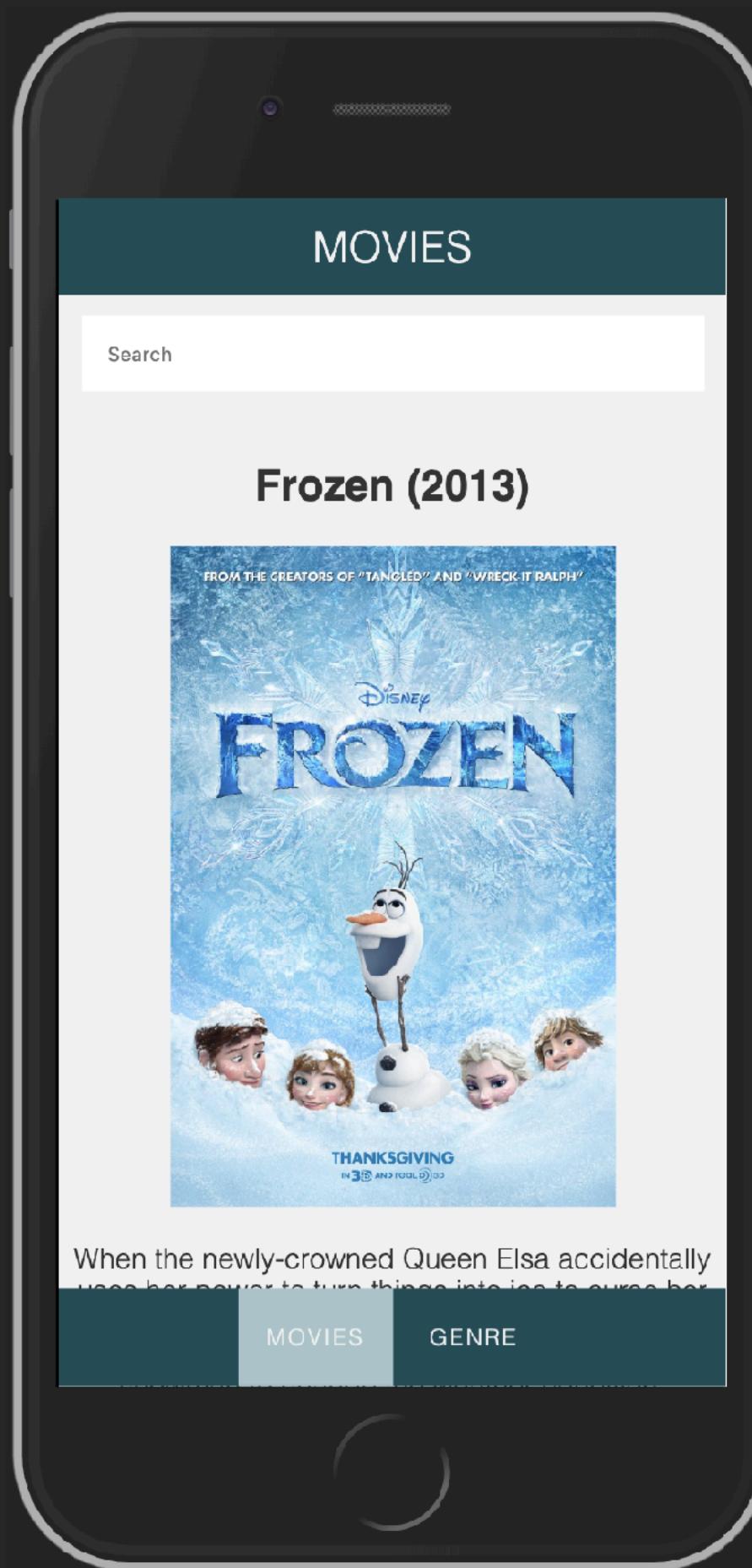
Username: mdu-front

Password: headlessfront01

# HEADLESS CMS

## MOVIE SPA

# SPA + HEADLESS CMS



spa-headless-cms-movies

# WP HEADLESS SETUP

1. Install WordPress on your domain: Subdomain or folder
2. Change Permalink Setting: post name
3. Create some posts
4. Test WordPress REST API in your browser, ex: <https://movie-api.cederdorff.com/wp-json/wp/v2/posts>
5. Fetch posts, ex: <https://movie-api.cederdorff.com/wp-json/wp/v2/posts>
6. Install plugins: Advanced Custom Fields & ACF to REST API
7. Setup custom fields
8. Display posts with custom fields
9. Use categories to categorise your posts and fetch categories

# 1. INSTALL WORDPRESS

 [Forside](#)

---

racedev.dk

-  [Administration](#)
-  [Website](#)
-  [File Manager](#)
-  [1-Click](#)
-  [E-mail](#)
-  [DNS](#)
-  [MySQL](#)
-  [MS SQL](#)

---

Simply.com Services

-  [Driftoversigt](#)
-  [SSL certifikater](#)
-  [Backorder](#)

---

 [Log ud](#)

**WordPress 1-Click installation racedev.dk**

Installer WordPress på dit webhotel, med blot et enkelt klik

WordPress is a state-of-the-art semantic personal publishing platform with a focus on aesthetics, web standards, and usability. What a mouthful. WordPress is both free and priceless at the same time. More simply, WordPress is what you use when you want to work with your blogging software, not fight it. With WordPress, you can create any type of website you want: a personal blog or website, a photoblog, a business website, a professional portfolio, a government website, a magazine or news website, an online community, even a network of websites. You can make your website beautiful with themes, and extend it with plugins. You can even build your very own application.

  
WordPress 5.2.3

## Installer WordPress

Indtast herunder de oplysninger du ønsker din WordPress installation skal bruge.

Adresse

Brug midlertidig URL i installation

Database  Tabel præfiks 

### Oplysninger

Værdierne bestemmes alene af dig og opbevares ikke af os. Noter gerne de login-oplysninger du vælger, så du husker dem.

Administrator's login	admin
Password	
Administrator's email	rasmus@cederdorff.com
Site name	Movie API
Interface language	English

**Installer WordPress nu**

# 2. PERMALINK SETTINGS

The screenshot shows the WordPress dashboard with the 'Movie API' theme active. The left sidebar is visible with various menu items like Dashboard, Posts, Media, Pages, Comments, Appearance, Plugins (with 3 updates), Users, Tools, and Settings. The 'Settings' item is currently selected. The main content area is titled 'Permalink Settings'. A notice at the top says 'WordPress 5.3.2 is available! Please update now.' Below it, a section titled 'Common Settings' lists five permalink structures:

- Plain: `http://movie-api.cederdorff.com/?p=123`
- Day and name: `http://movie-api.cederdorff.com/2020/02/05/sample-post/`
- Month and name: `http://movie-api.cederdorff.com/2020/02/sample-post/`
- Numeric: `http://movie-api.cederdorff.com/archives/123`
- Post name: `http://movie-api.cederdorff.com/sample-post/`

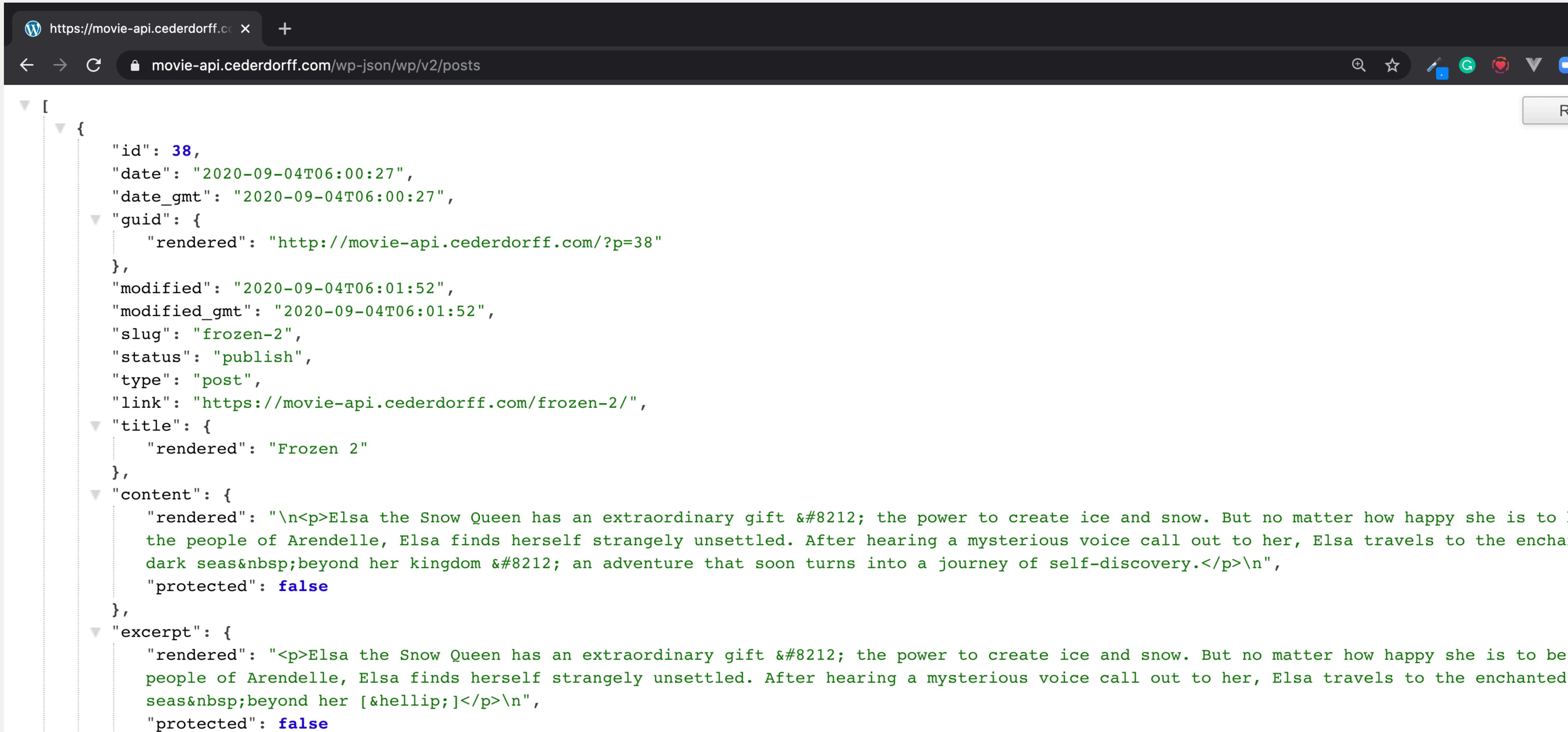
A box highlights the 'Post name' option. Below it, there's a 'Custom Structure' input field containing `http://movie-api.cederdorff.com/%postname%/` and a list of 'Available tags' including %year%, %monthnum%, %day%, %hour%, %minute%, %second%, %post\_id%, %postname%, %category%, and %author%.

# 3. CREATE SOME POSTS

The screenshot shows the WordPress dashboard for a site titled "Movie API". The left sidebar is dark-themed and includes links for Dashboard, Posts (which is selected and highlighted in blue), All Posts, Add New, Categories, Tags, Media, Pages, Comments, Appearance, Plugins (with a red notification badge showing '3'), Users, Tools, Settings, Custom Fields, and Collapse menu. The main content area has a header with the title "Posts" and a "Add New" button. A message at the top right says "WordPress 5.3.2 is available! Please update now." Below this, there's a table listing three posts: "Frozen" (Category: Animation, Comedy), "Don't Let Go" (Category: Drama, Horror), and "Lion King" (Category: Animation). Each post row includes a checkbox for bulk actions, the post title, the author (admin), and the category tags. At the bottom of the table are "Bulk Actions" and "Apply" buttons.

		Author	Categories	Tags
<input type="checkbox"/>	Title			
<input type="checkbox"/>	Frozen	admin	Animation, Comedy	—
<input type="checkbox"/>	Don't Let Go	admin	Drama, Horror	—
<input type="checkbox"/>	Lion King	admin	Animation	—
<input type="checkbox"/>	Title ▲			

# 4. TEST IN BROWSER



The screenshot shows a browser window with the URL `https://movie-api.cederdorff.com/wp-json/wp/v2/posts`. The page content is a JSON array representing a post. The first post in the array is displayed in detail:

```
[{"id": 38, "date": "2020-09-04T06:00:27", "date_gmt": "2020-09-04T06:00:27", "guid": {"rendered": "http://movie-api.cederdorff.com/?p=38"}, "modified": "2020-09-04T06:01:52", "modified_gmt": "2020-09-04T06:01:52", "slug": "frozen-2", "status": "publish", "type": "post", "link": "https://movie-api.cederdorff.com/frozen-2/", "title": {"rendered": "Frozen 2"}, "content": {"rendered": "\n<p>Elsa the Snow Queen has an extraordinary gift &#8212; the power to create ice and snow. But no matter how happy she is to be among the people of Arendelle, Elsa finds herself strangely unsettled. After hearing a mysterious voice call out to her, Elsa travels to the enchanted dark seas&nbsp;beyond her kingdom &#8212; an adventure that soon turns into a journey of self-discovery.</p>\n", "protected": false}, "excerpt": {"rendered": "<p>Elsa the Snow Queen has an extraordinary gift &#8212; the power to create ice and snow. But no matter how happy she is to be among the people of Arendelle, Elsa finds herself strangely unsettled. After hearing a mysterious voice call out to her, Elsa travels to the enchanted dark seas&nbsp;beyond her [&hellip;]</p>\n", "protected": false}}
```

# 5. FETCH POSTS

The screenshot shows the WordPress admin dashboard for a site titled "Movie API". The "Posts" menu item is selected, highlighted in blue. A notification bar at the top right says "WordPress 5.3.2 is available! Please update now." The main area is titled "Posts" with a "Add New" button. It displays a list of three published posts: "Frozen", "Don't Let Go", and "Lion King". Each post has a checkbox next to its title, followed by the author (admin), categories (e.g., Animation, Comedy for Frozen), and tags (e.g., — for Frozen). There are also "Bulk Actions" and "Apply" buttons at the bottom of the list.

	Title	Author	Categories	Tags
<input type="checkbox"/>	Frozen	admin	Animation, Comedy	—
<input type="checkbox"/>	Don't Let Go	admin	Drama, Horror	—
<input type="checkbox"/>	Lion King	admin	Animation	—

Global  
Variable

```
let _movies = [];

// fetch all movies from WP
async function getMovies() {
  let response = await fetch("https://movie-api.cederdorff.com/wp-json/wp/v2/posts");
  let data = await response.json();
  console.log(data);
  _movies = data;
  appendMovies(data);
  showLoader(false);
}

getMovies();
```

ARRAY  
movies

```
// append movies to the DOM
function appendMovies(movies) {
  let htmlTemplate = "";
  for (let movie of movies) {
    htmlTemplate += `
      <article>
        <h2>${movie.title.rendered} (${movie.acf.year})</h2>
        
        <p>${movie.acf.description}</p>
        <iframe src="${movie.acf.trailer}"></iframe>
      </article>
    `;
  }
  document.querySelector('#movies-container').innerHTML = htmlTemplate;
}
```

Inline variable

Store the current object  
from the array

function  
argument

variable inside  
the function

```
let _movies = [];

// fetch all movies from WP
async function getMovies() {
  let response = await fetch("https://movie-api.cederdorff.com/wp-json/wp/v2/posts");
  let data = await response.json();
  console.log(data);
  _movies = data;
  appendMovies(data);
  showLoader(false);
}

getMovies();
```

```
// append movies to the DOM
function appendMovies(movies) {
  let htmlTemplate = "";
  for (let movie of movies) {
    htmlTemplate += `
      <article>
        <h2>${movie.title.rendered} (${movie.acf.year})</h2>
        
        <p>${movie.excerpt.rendered}</p>
      </article>
    `;
  }
  document.querySelector("#content").innerHTML = htmlTemplate;
}
```

function argument

variable inside the function

Inline variable

Store the current object from the array

# 6. ADVANCED CUSTOM FIELDS

The screenshot shows the WordPress admin interface with the "Plugins" page selected. A banner at the top right informs the user that "WordPress 5.2.2 is available! Please update now." The main content area displays two active plugins: "ACF to REST API" and "Advanced Custom Fields".

Plugin	Description
<a href="#">ACF to REST API</a> <a href="#">Settings</a>   <a href="#">Deactivate</a>	Exposes Advanced Custom Fields Endpoints in the WordPress REST API Version 3.1.0   By Aires Gonçalves   <a href="#">View details</a>   <a href="#">Fork me on GitHub</a>   <a href="#">Make a donation</a>
<a href="#">Advanced Custom Fields</a> <a href="#">Deactivate</a>	Customize WordPress with powerful, professional and intuitive fields. Version 5.8.3   By Elliot Condon   <a href="#">View details</a>

On the left sidebar, the "Plugins" menu item is highlighted. Other menu items include Dashboard, Posts, Media, Pages, Comments, Appearance, and Settings. The "Custom Fields" menu item is also visible.

# 7. SETUP CUSTOM FIELDS

The screenshot shows the WordPress admin dashboard with the ACF plugin active. The left sidebar includes links for Dashboard, Posts, Media, Pages, Comments, Appearance, Plugins (with a red notification badge), Users, Tools, Settings, and Custom Fields (which is selected). Below these are sections for Field Groups, Add New, Tools, and a Collapse menu.

The main content area is titled "Edit Field Group" and shows a table for the "Movie" field group. The table has columns for Order, Label, Name, and Type.

Order	Label	Name	Type
1	Description	description	Wysiwyg Editor
2	Img	img	Url
3	Trailer	trailer	Url
4	Year	year	Text

Below the table, there is a section titled "Location" with a "Rules" box and a "Show this field group if" configuration. The configuration uses dropdown menus to set "Post Type" to "Post".

**Location**

**Rules**  
Create a set of rules to determine which edit screens will use these advanced custom fields

Show this field group if

Post Type is equal to Post

or

Add rule group

[Posts](#)

All Posts

Add New

Categories

Tags

[Media](#)[Pages](#)[Comments](#)[Appearance](#)[Plugins](#) 3[Users](#)[Tools](#)[Settings](#)[Custom Fields](#)[Collapse menu](#)

# Frozen

Start writing or type / to choose a block

**Movie****Description**[Add Media](#)[Visual](#) [Text](#)[Paragraph](#) [B](#) [I](#) [≡](#) [1/2](#) [“](#) [≡](#) [≡](#) [≡](#) [≡](#) [X](#) [≡](#)

When the newly-crowned Queen Elsa accidentally uses her power to turn things into ice to curse her home in infinite winter, her sister Anna teams up with a mountain man, his playful reindeer, and a snowman to change the weather condition.

**Img**[https://m.media-amazon.com/images/M/MV5BMTQ1MjQwMTE5OF5BMl5BanBnXkFtZTgwNjk3MTcyMDE@.\\_V1\\_SY1000\\_CR0,0,674,1000\\_AL\\_.jpg](https://m.media-amazon.com/images/M/MV5BMTQ1MjQwMTE5OF5BMl5BanBnXkFtZTgwNjk3MTcyMDE@._V1_SY1000_CR0,0,674,1000_AL_.jpg)**Trailer**[https://www.youtube.com/embed/TbQm5doF\\_Uc](https://www.youtube.com/embed/TbQm5doF_Uc)**Year**

2013

[Document](#) [Block](#)

x

**Status & Visibility**

^

Visibility

[Public](#)

Publish

[Sep 2, 2019 8:29 am](#) Stick to the Front Page

Author

admin

[Move to trash](#)[3 Revisions](#)

Permalink

▼

Categories

▼

Tags

▼

Featured Image

▼

Excerpt

▼

Discussion

▼

# 8. DISPLAY CUSTOM FIELDS

```
// append movies to the DOM
function appendMovies(movies) {
  let htmlTemplate = ``;

  for (let movie of movies) {
    htmlTemplate += `
      <article>
        <h2>${movie.title.rendered} (${movie.acf.year})</h2>
        
        <p>${movie.acf.description}</p>
        <iframe src="${movie.acf.trailer}"></iframe>
      </article>
    `;
  }

  document.querySelector('#movies-container').innerHTML = htmlTemplate;
}
```

# 9. USE CATEGORIES

Categories < Movie API — Word X + movie-api.cederdorff.com/wp-admin/edit-tags.php?taxonomy=category

Movie API 0 + New Howdy, admin

Dashboard Posts All Posts Add New Categories Tags Media Pages Comments Appearance Plugins Users Tools Settings Custom Fields Collapse menu

## Categories

Add New Category

Name  The name is how it appears on your site.

Slug  The “slug” is the URL-friendly version of the name. It is usually all lowercase and contains only letters, numbers, and hyphens.

Parent Category

Description

The description is not prominent by default; however, some themes may show it.

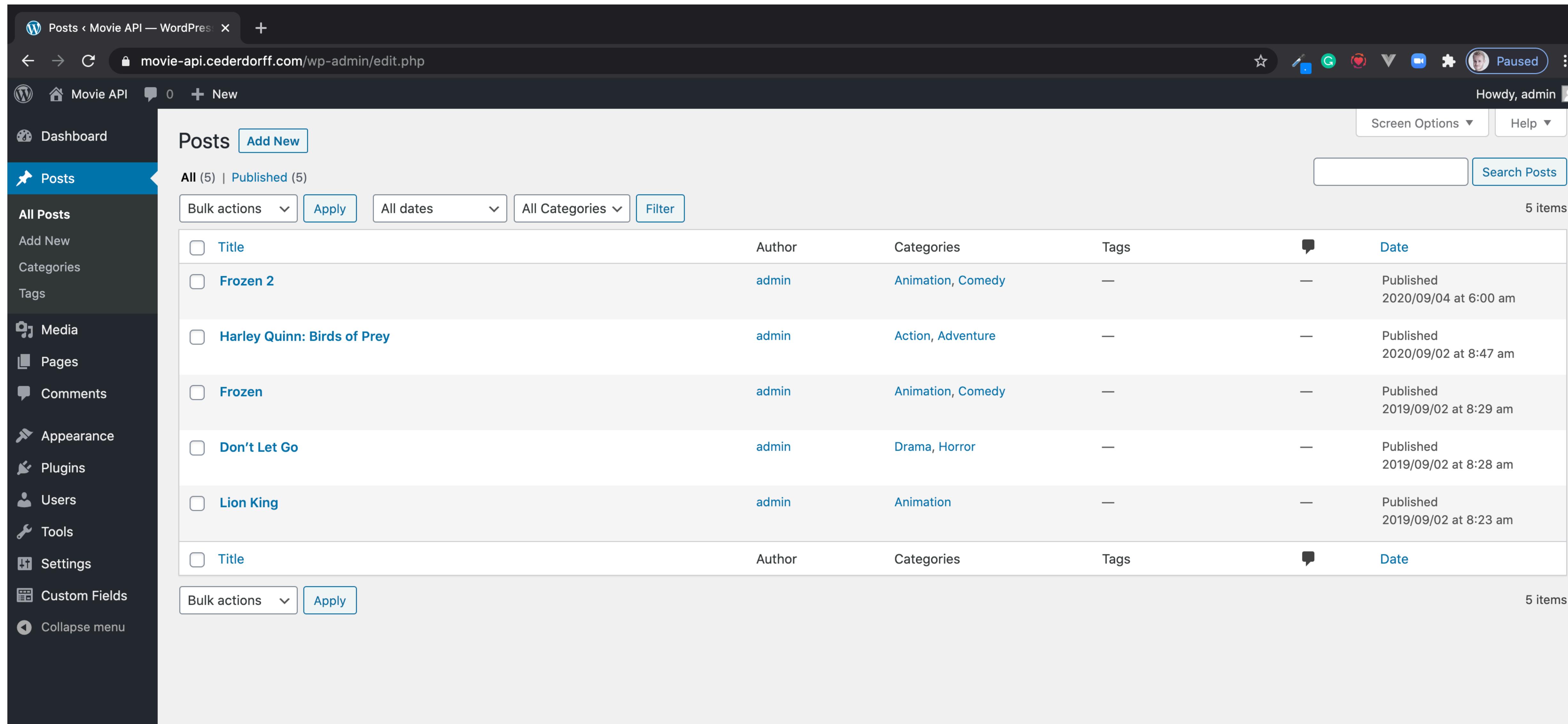
Add New Category

Bulk actions  7 items

<input type="checkbox"/> Name	Description	Slug	Count
<input type="checkbox"/> Action	—	action	1
<input type="checkbox"/> Adventure	—	adventure	1
<input type="checkbox"/> Animation	—	animation	3
<input type="checkbox"/> Comedy	—	comedy	2
<input type="checkbox"/> Drama	—	drama	1
Fantasy	—	fantasy	0
<input type="checkbox"/> Horror	—	horror	1
<input type="checkbox"/> Name	Description	Slug	Count

Bulk actions  7 items

# 9. USE CATEGORIES



The screenshot shows the WordPress admin interface for the 'Posts' screen. The left sidebar is dark-themed and includes links for Dashboard, Posts, All Posts, Add New, Categories, Tags, Media, Pages, Comments, Appearance, Plugins, Users, Tools, Settings, Custom Fields, and Collapse menu. The main content area has a light background and displays a list of five published posts. The columns in the list are: Title, Author, Categories, Tags, and Date. The posts are:

Title	Author	Categories	Tags	Date
Frozen 2	admin	Animation, Comedy	—	Published 2020/09/04 at 6:00 am
Harley Quinn: Birds of Prey	admin	Action, Adventure	—	Published 2020/09/02 at 8:47 am
Frozen	admin	Animation, Comedy	—	Published 2019/09/02 at 8:29 am
Don't Let Go	admin	Drama, Horror	—	Published 2019/09/02 at 8:28 am
Lion King	admin	Animation	—	Published 2019/09/02 at 8:23 am

At the bottom of the list, there is a header row with the same column titles: Title, Author, Categories, Tags, and Date. Below the list, there are buttons for Bulk actions (with dropdown and Apply) and a link for 5 items.

# 9. USE CATEGORIES

```
// fetch all genres / categories from WP
async function getCategories() {
  let response = await fetch("https://movie-api.cederdorff.com/wp-json/wp/v2/categories");
  let data = await response.json();
  console.log(data);
  appendCategories(data);
}

getCategories();

// append all genres as select options (dropdown)
function appendCategories(categories) {
  let htmlTemplate = "";
  for (let category of categories) {
    htmlTemplate += /*html*/
      `<option value="${category.id}">${category.name}</option>
      `;
  }
  document.querySelector('#select-category').innerHTML += htmlTemplate;
}
```

# 9. USE CATEGORIES

```
<section id="category" class="page">
  <header class="topbar">
    <h2>Movies By Category</h2>
  </header>
  <select id="select-category" onchange="categorySelected(this.value)">
    <option value="">Select Category</option>
  </select>
  <section id="movies-by-category-container" class="grid-container">
    <p>Please, select category</p>
  </section>
</section>
```

# 9. USE CATEGORIES

```
// category selected event - fetch movies by selected category
async function categorySelected(categoryId) {
  if (categoryId) {
    showLoader(true);
    let response = await fetch(`https://movie-api.cederdorff.com/wp-json/wp/v2/posts?_embed&categories=${categoryId}`)
    let data = await response.json();
    appendMoviesByCategory(data);
    showLoader(false);
  } else {
    document.querySelector('#movies-by-category-container').innerHTML = /*html*/
      `<p>Please, select category</p>
    `;
  }
}

// append movies by genre
function appendMoviesByCategory(moviesByCategory) {
  let htmlTemplate = "";
  for (let movie of moviesByCategory) {
    htmlTemplate += /*html*/
      `<article>
        <h2>${movie.title.rendered} (${movie.acf.year})</h2>
        
        <p>${movie.acf.description}</p>
        <iframe src="${movie.acf.trailer}"></iframe>
      </article>
    `;
  }
  // if no movies, display feedback to the user
  if (moviesByCategory.length === 0) {
    htmlTemplate = /*html*/
      `<p>No Movies</p>
    `;
  }
  document.querySelector('#movies-by-category-container').innerHTML = htmlTemplate;
}
```

# WORDPRESS API ROUTES

Meta data and routes: <https://movie-api.cederdorff.com/wp-json/>

Posts: [https://movie-api.cederdorff.com/wp-json/wp/v2/posts? embed](https://movie-api.cederdorff.com/wp-json/wp/v2/posts?embed)

Categories: <http://movie-api.cederdorff.com/wp-json/wp/v2/categories>

[http://movie-api.cederdorff.com/wp-json/wp/v2/posts?  
\\_embed&categories=4](http://movie-api.cederdorff.com/wp-json/wp/v2/posts?_embed&categories=4)

Admin: <https://movie-api.cederdorff.com/wp-admin>

Username: mdu-front

Password: headlessfront01

# PROJECTS

headless-cms

headless-cms-acf-products

headless-cms-acf-teachers

headless-cms-cederdorff-com

headless-cms-template

spa-headless-cms-movies

spa-headless-cms-movies-template

spa-headless-cms-teachers-family-members

spa-headless-cms-template

# DATA STRUCTURES

# MY OWN SIMPLE STRUCTURE

JS main.js X ...

```
17  /**
18   * Fetch movie from data source: JSON file
19   */
20  async function loadMovies() {
21    let response = await fetch("./json/movies.json");
22    let data = await response.json();
23    return data;
24  }
25
26 /**
27  * Appending movies to the DOM by the giving argument
28  */
29 function appendMovies(movies) {
30  let html = "";
31  for (const movie of movies) {
32    console.log(movie);
33    html += /*html*/
34      <article>
35        <h2>${movie.title} (${movie.year})</h2>
36        
37        <p>${movie.description}</p>
38        ${generateFavMovieButton(movie.id)}
39      </article>
40    `;
41  }
42  document.querySelector("#movie-container").innerHTML = html;
43 }
```

```
[  
  {  
    "id": 68,  
    "date": "2021-09-03T19:43:15",  
    "date_gmt": "2021-09-03T19:43:15",  
    "guid": {  
      "rendered": "https://movie-api.cederdorff.com/?p=68"  
    },  
    "modified": "2021-09-03T19:45:57",  
    "modified_gmt": "2021-09-03T19:45:57",  
    "slug": "suicide-squad",  
    "status": "publish",  
    "type": "post",  
    "link": "https://movie-api.cederdorff.com/suicide-squad/",  
    "title": {  
      "rendered": "Suicide Squad"  
    },  
    "content": {  
      "rendered": "\n<p>A secret government agency recruits some  
mission: save the world from the apocalypse.</p>\n",  
      "protected": false  
    },  
    "excerpt": {  
      "rendered": "<p>A secret government agency recruits some  
mission: save the world from the apocalypse.</p>\n",  
      "protected": false  
    },  
    "author": 1,  
    "featured_media": 71,  
    "comment_status": "open",  
    "ping_status": "open",  
    "sticky": false,  
    "template": "",  
    "format": "standard",  
    "meta": [],  
    "categories": [  
      7,  
      5,  
      1  
    ],  
    "tags": [],  
    "acf": {  
      "description": "<p>A secret government agency recruits some  
mission: save the world from the apocalypse.</p>\n",  
      "year": 2021  
    }  
  }]
```

```
// append movies to the DOM  
function appendMovies(movies) {  
  let htmlTemplate = "";  
  for (let movie of movies) {  
    htmlTemplate += /*html*/ `  
      <article>  
        <h2>${movie.title.rendered} (${movie.acf.year})</h2>  
          
        <p>${movie.excerpt.rendered}</p>  
      </article>  
  `;  
}  
document.querySelector('#movies-container').innerHTML = htmlTemplate;
```

# INSPECT STRUCTURE, ARRAYS, OBJECTS & PROPERTIES

SINGLE PAGE WEB APP TEMPLATE  
127.0.0.1:5501

MOVIES

Search

Suicide Squad (2016)



A secret government agency recruits some of the most dangerous incarcerated super-villains to form a defensive task force. Their first mission: save the world from the apocalypse.

Spider-Man: No Way Home (2021)



For the first time in the cinematic history of Spider-Man, our friendly neighborhood hero is unmasked and no longer able to separate his normal life from the high-stakes of being a Super Hero. When he asks for help from Doctor Strange, the stakes become even more dangerous, forcing him to discover what it truly means [...]

MOVIES    CATEGORIES

Elements    Console    Sources    Network    4    Default levels

5 Issues: 4 1

Array(8) i

0:

- acf: {description: '<p>A secret government agency recruits some of the most dangerous incarcerated super-villains to form a defensive task force. Their first mission: save the world from the apocalypse.', author: 1}
- categories: (3) [7, 5, 1]
- comment\_status: "open"
- content:
  - protected: false
  - rendered: "\n<p>A secret government agency recruits some of the most dangerous incarcerated super-villains to form a defensive task force. Their first mission: save the world from the apocalypse."
- [[Prototype]]: Object
- date: "2021-09-03T19:43:15"
- date\_gmt: "2021-09-03T19:43:15"
- excerpt: {rendered: '<p>A secret government agency recruits some of the most dangerous incarcerated super-villains to form a defensive task force. Their first mission: save the world from the apocalypse.'}
- featured\_media: 71
- format: "standard"
- guid: {rendered: 'https://movie-api.cederdorff.com/?p=68'}
- id: 68
- link: "https://movie-api.cederdorff.com/suicide-squad/"
- meta: []
- modified: "2021-09-03T19:45:57"
- modified\_gmt: "2021-09-03T19:45:57"
- ping\_status: "open"
- slug: "suicide-squad"
- status: "publish"
- sticky: false

- tags: []
- template: ""
- title: {rendered: 'Suicide Squad'}
- type: "post"

# WORDPRESS API ROUTES

Meta data and routes: <https://movie-api.cederdorff.com/wp-json/>

Posts: [https://movie-api.cederdorff.com/wp-json/wp/v2/posts? embed](https://movie-api.cederdorff.com/wp-json/wp/v2/posts?embed)

Categories: <http://movie-api.cederdorff.com/wp-json/wp/v2/categories>

[http://movie-api.cederdorff.com/wp-json/wp/v2/posts?  
\\_embed&categories=4](http://movie-api.cederdorff.com/wp-json/wp/v2/posts?embed&categories=4)

Admin: <https://movie-api.cederdorff.com/wp-admin>

Username: mdu-front

Password: headlessfront01

# ADD CATEGORIES

The screenshot shows the WordPress Admin Categories page. On the left, a sidebar menu is visible with options like Posts, Categories (which is currently selected), and others like Media, Pages, and Comments. The main area has a title 'Categories' and a sub-section 'Add New Category'. It includes fields for 'Name' (with a placeholder 'The name is how it appears on your site.'), 'Slug' (with a placeholder 'The "slug" is the URL-friendly version of the name. It is usually all lowercase and contains only letters, numbers, and hyphens.'), 'Parent Category' (set to 'None'), and 'Description' (with a note that it's not prominent by default). To the right, a table lists existing categories:

Name	Description	Slug	Count
Action	—	action	3
Adventure	—	adventure	3
Animation	—	animation	3
Comedy	—	comedy	3
Drama	—	drama	1
Fantasy	—	fantasy	1
Horror	—	horror	1

# ADD CATEGORIES TO POSTS

The screenshot shows the WordPress admin dashboard with the "Posts" screen selected. The left sidebar includes links for Dashboard, Posts, All Posts, Add New, Categories, Tags, Media, Pages, Comments (6), Appearance, Plugins, Users, Tools, Settings, Custom Fields, and Collapse menu. The main area displays a list of posts with columns for Title, Author, Categories, Tags, and Date. The posts listed are:

Title	Author	Categories	Tags	Date
Suicide Squad	admin	Action, Adventure, Fantasy	—	Published 2021/09/03 at 7:43 pm
Spider-Man: No Way Home	admin	Action, Adventure	—	Published 2021/09/03 at 7:40 pm
Cruella	admin	Comedy	—	Published 2021/09/03 at 6:47 pm
Frozen 2	admin	Animation, Comedy	—	Published 2020/09/04 at 6:00 am
Birds of Prey	admin	Action, Adventure	—	Published 2020/09/02 at 8:47 am
Frozen	admin	Animation, Comedy	—	Published 2019/09/02 at

# DEFINE HTML

```
<!-- categories page -->
<section id="categories" class="page">
  <header class="topbar">
    <h2>Categories</h2>
  </header>
  <select id="select-category" onchange="categorySelected(this.value)">
    <option selected disabled>Select Category</option>
  </select>
  <section id="movies-by-category-container" class="grid-container">
    <article class="empty">
      <p>Please, select category</p>
    </article>
  </section>
</section>
```

# FETCH & APPEND CATEGORIES

```
// fetch all genres / categories from WP
async function getCategories() {
    let response = await fetch("https://movie-api.cederdorff.com/wp-json/wp/v2/categories");
    let data = await response.json();
    console.log(data);
    appendCategories(data);
}

getCategories();

// append all genres as select options (dropdown)
function appendCategories(categories) {
    let htmlTemplate = "";
    for (let category of categories) {
        htmlTemplate += /*html*/
        <option value="${category.id}">${category.name}</option>
    };
}
document.querySelector('#select-category').innerHTML += htmlTemplate;
```

# DEFINE SELECT ONCHANGE EVENT

```
// category selected event - fetch movies by selected category
async function categorySelected(categoryId) {
  if (categoryId) {
    showLoader(true);
    let response = await fetch(`https://movie-api.cederdorff.com/wp-json/wp/v2/posts?_embed&categories=${categoryId}`);
    let data = await response.json();
    appendMoviesByCategory(data);
    showLoader(false);
  }
}
```

# APPEND MOVIES BY CATEGORY

```
// append movies by genre
function appendMoviesByCategory(moviesByCategory) {
    let htmlTemplate = "";
    for (let movie of moviesByCategory) {
        htmlTemplate += /*html*/ `
            <article>
                <h2>${movie.title.rendered} (${movie.acf.year})</h2>
                
                <p>${movie.excerpt.rendered}</p>
            </article>
        `;
    }
    // if no movies, display feedback to the user
    if (moviesByCategory.length === 0) {
        htmlTemplate = /*html*/ `
            <p>No Movies</p>
        `;
    }
    document.querySelector('#movies-by-category-container').innerHTML = htmlTemplate;
}
```



Code  
Every  
Day