

# CMS

Everything you need to know



enonic

*The ultimate guide to the content management system (CMS), providing the marketer with a useful overview of functionality, best practices, CMS alternatives, and much, much more!*

# Contents

<b>What is a content management system (CMS)?</b> .....	4
Definition of CMS .....	6
The purpose of a CMS .....	9
Difference between CMS and digital experience platform .....	11
Different types of CMSs .....	13
On premises vs. hosted .....	14
Traditional CMS .....	16
Headless CMS .....	18
Decoupled CMS .....	20
When to choose what CMS .....	21
<b>Producing content with the CMS</b> .....	23
Text editing .....	24
Advantages and disadvantages of different text editor types .....	26
Media .....	27
Other media: video, pdf, etc. ....	30
<b>Distribution and presentation with the CMS</b> .....	32
Metadata and SEO .....	32
Marketing automation tools .....	35
Design .....	37
Back-end affects front-end .....	38
Templating support for changing designs .....	40
General site hierarchy .....	42
URL logic .....	44
Vanity URL .....	45
<b>Editorial management in the CMS</b> .....	46
Publish dates .....	46
On and off time .....	48
User roles .....	49
Revision and version control .....	51
Version control: is it published or not? .....	52
Accessibility and usability .....	53
<b>The future of CMS</b> .....	56
Brief history of CMS .....	56
What comes next? .....	58
Cloud .....	58
Headless .....	58
Beyond CMS .....	60
Progressive web apps .....	61
Future proofing your CMS .....	64
<b>CMS alternatives</b> .....	66
WordPress versus the world .....	68
Essential features for B2B, SME, etc. ....	71
CMS alternatives for B2B and SME .....	73
Closing remarks .....	75

## What is a content management system (CMS)?

Let us start at the beginning: Why should marketers care about what tools they work with? What do you care if you write in Word, Docs, or Pages, perform calculations in Excel, Sheets, or Numbers, or make presentations in PowerPoint, Slides, or Keynote? They are just tools, performing certain tasks. Or is that so?

Do you think carpenters care about what tools they use, like hammers, saws, and screwdrivers? It is unlikely that they are indifferent to the quality and function of their tools – a good hammer can knock and pull nails with precision, whereas a bad hammer knocks off balance, pulls poorly, or may even break during work.

The same principle applies to all other professions, like truck drivers, physicians, and programmers. All these professionals do care about their vehicles, scalpels, and computers, respectively. And so should you as a digital marketer. You ought to care about your tools, especially the content management system (CMS) of your organisation.



Almost every company, association, institution, you name it has a website. Websites are no longer the one-page business card from the early 90s, with simple HTML and design. The websites of today are often complex ecosystems of pages, images, links, system integrations, self-service, shopping, and other functions, and all these elements need to come together and be organised by a content management system.

Should you care about the tools that deliver your content on the web? Why, of course you should! Now let us continue with a definition of a CMS, as we need to know exactly what we're talking about.

## Definition of CMS

A content management system, or CMS for short, is a computer software system that manages and organises the creation, modification, and publication of digital content like documents and images, often in a collaborative environment, for presenting to a website. It may also be known by other terms, such as “web content management” (WCM) and “digital experience platform” (DXP).

If you have been working with websites of any kind of way in the last 20 years, you have most likely encountered a CMS or the like in one form or the other. The editorial staff of online newspapers use it, the digital marketing team of corporations use it, and departments at governmental agencies use it. Everybody uses CMS. Even private bloggers and small enterprises as well – they don’t edit single HTML files and upload them to the web manually, they use a certain system for handling this and many related tasks in a dynamic environment.

Of course, bloggers dealing with fashion, cooking, or just their own daily lives don’t need to think about the technical or organisational aspects of running a website, because the CMS does a lot of the heavy lifting for them, either it’s on their own hosted site or for instance on a blogging platform. The CMS automates and sets up templates for a lot of technical IT stuff, so e.g. bloggers and marketers don’t have to bother.



Practically everyone with a website presence has a CMS, and the CMS performs a lot of tasks, both seen and unseen. Let's have a look at some of the most common features the content management system provides. A CMS manages:

- Text editing
- Metadata
- Design
- General site hierarchy
- URL logic
- Media
- Publish dates
- User roles
- Organisation of the different posts and pages within the hierarchy
- History editing and version control
- Indexing
- Search
- Retrieval
- And so on

Some of these features any user can change, add, or remove, even though they might have limited technical knowledge. Other features are usually restricted to the webmaster or developers. Examples of the first group are article copy (“copy” is another word for “text”), images, URL, publish date, placement in the hierarchy, and some metadata fields. Examples of the latter group include templating, design, site hierarchy, user roles, and search.

All these features are handled by the CMS, which is a great advantage and timesaver for all editorial and marketing teams. Some other advantages with CMS include:

- Reduced need to code from scratch
- Easy to create a unified look and feel
- Version control
- Edit permission management

Disadvantages may include:

- Limited or no ability to create functionality not envisioned in the original CMS, like layouts, web apps, etc.
- Increased need for special expertise and training for content authors

## The purpose of a CMS

Now we know what a content management system is: it is software for managing content for being displayed through a digital channel. Although the purpose of a CMS may be more or less directly mentioned in our definition, for reasons of clarity and a more complete understanding of the concept we will in this section discuss the intent of the system.

As we have seen, with a CMS you can put any content on a website, either if it's simple text, images, and videos, or more complex content in the form of searchable databases, interactive tools, and advanced personalisation. What content should be placed where, what should be visible to whom, when should the content expire, and so on are questions that are solved by applying the answer of CMS.



Thus, the purpose of a CMS is first and foremost content management: You and your organisation get to decide what, when, and where to distribute your content, as well as how in the form of presentation (the “why” is beyond the scope of this article). The purpose is to effectively deliver your information to your audience, as well as opening for receiving information back from the visitors, like user preferences, behaviour, filled out forms, etc.

This leads us to the total website experience. The content is one thing, but the services, self service functions, URLs, design, templating, and so on are a part of the package too. A website is an integration of content and presentation, and the CMS is responsible for both aspects (albeit in different degrees depending on the type of CMS, as we will see below).

The CMS controls directly or indirectly the entire experience for both you and your visitors in regard to the digital meeting place that a website or a related service is, and the purpose is to effectively handle this “meeting” and ensure that necessary communication is delivered both ways: You might want to help your customers or target audience, and your customers or prospects would like to solve tasks – that your organisation and your digital offering can assist with.

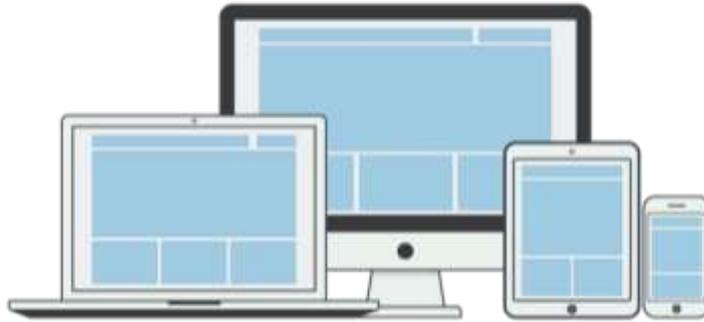
For a quick recap, you can say that a CMS is used for the creation and editing of digital content, usually in the form of websites, and as a part of your organisation’s digital experience toward your target audience. Which leads us to the question: What is the difference between CMS and digital experience?

## Difference between CMS and digital experience platform

You might have heard the term “digital experience” in relation to CMS. Forrester defines a digital experience platform as “software to manage, deliver, and optimize experiences consistently across every digital touchpoint”. Whether a given customer or lead interacts with your brand digitally via the good old computer, smartphone, tablet, wearables, beacons, or facial recognition tech, their experience should be holistic, predictable, and delightful.

At the same time all this information from across different channels ought to be analysed to provide your team with a more complete understanding of the customer’s identity, intentions, and sentiments as they interact with your brand. This is where digital experience software comes into the picture.

A digital experience platform (DXP) delivers a unified experience across different digital channels, while at the same time using analytics to understand the users and continuously improve. A DXP could aid your organisation in aligning strategies, processes, teams, and technologies toward an even more customer-centric approach than you already might have. Adobe is one of the large players talking about digital experience, and the concept has gained traction in the past few years.



The main difference between a CMS and a DXP is the scope of the service(s). A CMS can for instance provide a website with third-party marketing tools for SEO, analytics, and marketing automation. A CMS can also deliver a highly specialised functionality on the website, like a well-behaved content repository, advanced search, shopping features, user profiles with file archives, and much more. A DXP, on the other hand, may offer an integration between several different websites and digital touchpoints which all need to connect and communicate.

Whether your organisation needs a DXP or not is a different question. Is your organisation a member of a global network with strict brand and risk guidelines, a DXP might be the right choice for you for controlling and delivering a wide array of websites and functionality around the globe to a likewise wide group of users. Are you handling somewhat simpler, but still advanced projects, a flexible CMS with the possibility of integrating different tools seamlessly is likely the best option.

Remember, a CMS is at the core of any DXP, but it all depends on the number of complex operations and needs of your organisation or member firm network.

## Different types of CMSs

So far, we have discussed what a CMS is, the purpose of a CMS, and what separates it from a digital experience platform, or DXP for short. What different types of CMSs are there? In how many ways can you manage and organise the creation, modification, and publication of digital content for loading to a website? As it turns out, the answer is “many”. Here we will go through the most common and distinct forms of content management systems out there, and this will not be a detailed account of every slightly different software in existence.

*Note: There are several ways of classifying content management systems. We have chosen to classify them by key technical features dictating the relationship between content and presentation.*

## On premises vs. hosted

Let's say your organisation has decided to expand its online presence, first and foremost in having a great and forward-looking website that fulfils the needs of your customers. The first thing you have to decide is where you will host your digital offering. Will it be on premises or be hosted by a third-party company?

For private citizens this basic question usually surmounts to the alternative "managed vs. hosted CMS." A managed CMS is provided as a service, and all things technical are handled by the provider, like Shopify, Weebly, or WordPress.com. The advantages of a managed system are active support, continuous development, and software that is always up-to-date. The disadvantages include lack of control over certain design elements, features, content ownership, and several other factors.



A hosted CMS, on the other hand, is a “standalone” piece of software that may be applied to your own server or a server from a third-party provider. The main advantage of a hosted CMS is control: You (or your developers) can make everything on your website work exactly as you wish, and with integrations you can also add to the functionality of the digital offering. The disadvantage with respect to managed CMSs is that you are on your own in regard to updating, security, and design from the start – but this is rarely a problem for professional organisations.

Most organisations and businesses will face the alternative of “on premises vs. hosted,” rather than “managed vs. hosted”. “On premises” means your own organisation will take care of the hosting of the website and related products, with physical servers on a location that you control. The advantages we saw from hosting persist, with the added bonus of increased security and control over the data, and a more predictable cost plan. Disadvantages include a larger portion of the organisation’s resources allocated to hosting, and a larger share of risk in hosting the data yourselves.

Now that you have had time to assess the alternatives in how and where to host your content management system, you can start to look into the different types and which suits your organisation the best.

## Traditional CMS

A traditional CMS is something most of us have heard of, or have no problems understanding at a glance. In this kind of content management system, also known as web content management (WCM), there is a tight connection between features for the editors and features for the developers.



The editors usually have access to full preview, landing page editing, URL handling, detailed access control, media repository, and more. At the same time the developers can code, test, and deploy both editorial and end user functionality bundled together, made possible by a tight coupling between the delivery layer and the CMS. This is why such CMSs sometimes are called “coupled CMS” as well.

The feature set of such CMSs has grown over time and can be regarded as full-scale marketing suites. This has resulted in what we discussed above: software to manage, deliver, and optimise experiences consistently across all digital touchpoints – also known as digital experience platforms, or DXPs.

There are some limitations or disadvantages with the traditional CMS. For instance, the aforementioned tight connection between the presentation and the content makes multichannel presence a real chore. This is due to that a standard desktop web browser, a mobile app, and an Internet of Things (IoT) device cannot all read and present the content when the presentation layer is made for just one of them – the web.

If all your organisation needs is a basic website, the traditional or coupled kind of content management system is probably the right option for you. If you are looking for cross-platform integrations and / or more complex solutions, this is where the alternatives of headless and decoupled CMS come into the picture.

## Headless CMS

The rise of mobile apps, social media, smart watches, digital signage, self-service machines, robots, and other IoT devices is changing the way we consume digital content. All these platforms have unique properties and different methods of showing and managing content, so how do you make sure that your content is presented correctly? Making one version for each required platform is time consuming, expensive, and ineffective. Enter the so-called “headless” solution.



The main idea behind a headless CMS is to separate presentation and content. “Headless” refers to the presentation layer as the head, and the content as the body. With headless CMSs the content stays the same, while each client, either it is an app, a device, or a browser, is responsible for how the content is presented.

Developers usually do not code in the headless CMS itself, but use an application programming interface (API) instead. An API is a tool that makes it possible for two software services to communicate and share data. By doing this, the headless system always defines the content model and editors may start adding content, while developers consume the content through a defined API – e.g. presenting the content to an ATM machine or to an app on your smartphone.

While the advantage of a headless CMS may be seen as a “one size fits all” approach, there are limitations also here. For each new client and device, the developer must manage link handling, formatting issues and templating, caching, permissions, and more.

Also, for marketers and editors, preview of the content is no longer handled in the CMS itself, but on each related API project. Without getting lost in technical discussions, other disadvantages include missing tree structure / hierarchy, URL handling, landing page building, and more.

## Decoupled CMS

Headless CMS has experienced a growth in popularity the last few years, and as a reaction to this competition several traditional CMS vendors have added web-based content APIs. This has introduced the so-called decoupled CMS. A decoupled CMS arrives with a presentation layer (head), but using it is optional. This means that you can have a standard website, while at the same time control the presentational output in other platforms as well by your own choice.

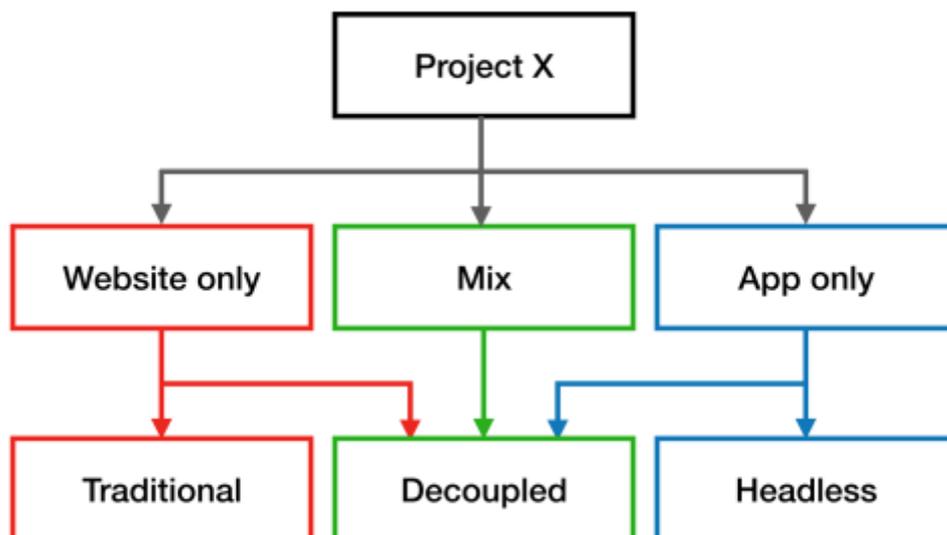
Offerings from vendors vary, and having an API is not sufficient on its own to properly qualify as a decoupled CMS. A high quality decoupled CMS should prioritise the management of structured content over building pages, support rich media handling, and include APIs with support for a wide range of programming languages.

## When to choose what CMS

Choosing the right CMS matters. What, then, should you choose when faced with the trichotomy of traditional, headless, and decoupled CMS?

- Choose a traditional CMS if you only need a basic website
- Choose a headless CMS if your project is focused on an app or IoT with limited editorial requirements
- Choose a decoupled CMS if your project involves rich web content, URL handling, extensive editorial requirements, special hosting needs, and reuse of content across projects

Still confused? Maybe this simplified flowchart can help you navigate the unfamiliar terrain:



You are now in possession of basic knowledge about the concept of a content management system (CMS). A CMS manages the creation, modification, and publication of digital content like documents, images, videos, and other functionalities for displaying on a website. The purpose of a CMS is content management and controlling the total website experience for the end user, and differs from a digital experience platform (DXP) in the scope of the offering, with the latter being focused on experiences across multiple digital touchpoints. Three main types of CMSs exist, with the traditional being for simple websites, headless for apps, and decoupled for a mixture.

Now, what exactly can you do with a CMS? Let's delve into the details!

## Producing content with the CMS

If you blog, work with some form of marketing, or have experience with online journalism you have probably encountered a CMS. Try to think about what you really do. If you're responsible for some content, most likely you write copy, upload a picture, fill in some metadata, and either publish the article or send it to approval. Maybe you also rearrange elements on the website, like the order of articles or what to feature on the front page. Or perhaps you maintain the employee profiles, adding new pictures, deleting old ones, and update the biographical information. If you are a site lead, you may work with an overview of articles, where they are placed on the website, what permissions users in the system have, and what third-party integrations are allowed. If you're a blogger, you may just make content and hit "publish", and that's it.

Either way, you expect the CMS to do your bidding, you expect it to be flexible and reliable, you expect it to simply work. No matter what task you have in mind – content organisation, media repository, URL structures, SEO tactics, design elements, you name it – you rely on certain features to work not just by themselves, but also to work in harmony and in unity with the whole system.

In our definition we listed several features you usually find in a content management system. Now we will treat these properties in a more structured and detailed way. We'll start with the one feature most people are familiar with: the text editor.

## Text editing

At the heart of any CMS lies the text editor tool. Like its pure counterparts in the operating system environment, the CMS text editor has grown to have a lot of strong features. In the days of yore most text editors required you to write in code, like `<bold>this</bold>` and `<p>that</p>`. However, with the rise of the WYSIWYG editors,

**What**

**You**

**See**

**Is**

**What**

**You**

**Get**

Instead of endless amounts of unreadable codes, you get the “finished” product in the form of formatted text, images, videos, embeds, headers, paragraphs, quotes, superscript, underlines, links, etc. This is both user friendly and effective, as it removes obstacles from what really matters: creating quality content.

Of course, this has been the industry standard for years and might seem like a banality to mention, but you shouldn't forget history or what really lies beneath the surface. Many CMSs let you switch seamlessly between a WYSIWYG (pronounced approximately like "we see wig") editor and an HTML editor, but a few has locked the editing to the visual level, for reasons of e.g. keeping to strict brand guidelines defined by the webmaster or the developers. Some CMSs even present you the text editor in close semblance to the actual finished product, so it feels like you are editing directly on the web page. Others may require basic HTML skills and only allow an HTML editor.



## Advantages and disadvantages of different text editor types

There are advantages and disadvantages with both types of editors. A WYSIWYG editor makes it a dream to produce content and may be the best option for corporations that want to limit the inclusion of certain risk-posing elements like unwarranted code, images, or the like, but this may limit your work, and the toolbar may lack options for simple features like tables, superscript, quote formatting etc. An HTML editor gives you full control over the output, but takes it for granted that you know some basic HTML, like how to format properly and how to write the code for links and images.

What you should look for in a CMS is a text editor that suits the needs of your organisation. If you are a small or medium-sized company, a hybrid text editor may fit you perfectly. If you are a giant, international corporation, a strict WYSIWYG editor with limited choices for a broad range of users might better comply with brand and security standards.

## Media

After basic text editing the handling of images and other rich media files are probably the most common task performed in a CMS. Who doesn't want gorgeous pictures brightening up their content, making their website more colourful and breaking up the monotonous walls of text?

Beginning with basic images, again think about what tasks you usually solve with pictures in a CMS. You don't want your website to look dull and boring, so adding a little colour and breathers is most likely your and your organisation's basic motivation for including images in the first place, as well as illustrating the related topics or even difficult concepts touched upon in the copy.



An image is usually used in two ways in a CMS:

1. The image is placed directly in the given page, either through the text editor or as a fixed page element / module etc.
2. The image is linked to the page through metadata outside of the content in the article

These two methods are not mutually exclusive, in fact most CMSs uses both ways simultaneously. The first method is fairly easy to understand: You want an image here, after this paragraph, so you just insert it. The second method is for a more indirect use of the image, and most commonly refers to so-called “featured” images – the image you want to connect to an article through search results on the website, and as a preamble / preview in modules fetching content on e.g. topic pages or the front page. This second method may also dictate what image you see on top in the template of the article itself, as opposed to images you manually put in via the text editor or the like.

Another aspect to consider about images is the management and organisation of your growing collection of them. Once more, this touches upon a two-fold approach to image handling:

1. You upload the image directly from your computer, the Internet etc.
2. You fetch the image from your archive

In either method, you end up stuffing your CMS archive full of images. As this will always be the case, you should enforce a certain naming convention, as well as ensure your CMS has a database or folder structure to divide your images into several places (it is not a good idea to gather all the images in one single folder, as this increases both vulnerability and load on the server).

## Other media: video, pdf, etc.

Although images play an important part of any website and therefore any CMS, they are by far the only media type available on the world wide web. In fact, videos are an increasingly popular medium, due to the availability of platforms like YouTube and the enthralling nature of the moving pictures themselves.

Videos can be a great way to educate and entertain your prospects and user groups, but they can be more time consuming and often require more planning and logistics than simply taking a photograph (or finding an image on the web through stock photo services or free image services like Pixabay). Your organisation has to find out whether or not videos fit your target audience or buyer persona, and then plan and execute accordingly.



Videos can be uploaded locally or embedded through platforms like YouTube and Vimeo. Embedding videos from a third-party is the easiest way to integrate videos with your CMS, but as you have no control over the service, you are not guaranteed that the video will always be available when you prefer. In this case you should consider using a tool like Qbrick, which allows you to upload videos to the cloud and still decide the look and feel of the video player.

Other kinds of documents, like PDFs, presentations or spreadsheets, are usually uploaded to the CMS via a media uploader wizard. Then you link to the given document in the text editor via an ID or URL function, where you browse your media depository just like with images. Embedding is also possible here, for instance via services like SlideShare.

Naming conventions and file size best practices also apply to these kinds of media, as much as they do to images. Be sure to have a logical system and keep those file sizes down!

# Distribution and presentation with the CMS

## Metadata and SEO

After you have had fun painting your website with invigorating copy and beautiful images, it is time to move on to the more technical, and some would say “boring” tasks of daily CMS routines. Yes, you have a thoughtful article, yes you have strong visuals, but you haven’t yet made your content fully visible to the search engines – the software that enables real human beings to find your content in the first place.

This is where metadata come into the picture. Metadata are fields and values that are used to tag, categorise, and describe your content. If your CMS is the library and your content the books, metadata can be seen as the filing system and corresponding labels. Metadata helps with the following:

- Search – adding keywords to the metadata enables users to find the given article, image, or PDF faster and easier
- Distribution – some values indicate when and where a given piece of content will be shared, determining the behaviour of certain applications that can distribute the content
- Access – ever tried to access a certain file, but were denied? This is due to restrictions set by the metadata, telling the system to bar everyone not fulfilling a specific requirement



Common metadata fields in relation to CMS include:

- Title
- Preamble / description
- Publication date
- Author
- URL
- Content type

Some advanced CMSs may include a ton more fields to fill out, like vanity URLs, redirect rules, on and off times, and much more. Whether this myriad of fields is optional or not depends on what your developers have decided and what is possible in the CMS.

For sharing in social media there is a standard called Open Graph protocol, which “enables any web page to become a rich object in a social graph.” In plain English this basically means that you tell social media platforms exactly what title, description, and image you want displayed on e.g. Facebook or LinkedIn.

The point of all this tagging and categorising is to optimise your website for search engines, also known as search engine optimisation, or SEO. SEO is a complex field where many experts will have their say, and to make matters even more complicated, SEO is ever-evolving. Google is the most famous search engine and always modifies its search algorithms to make searches better, more precise, and more relevant for the end users.

## Marketing automation tools

The ability to create web pages with web pages in the Internet browser does probably not impress neither you nor your organisation as it might have done in 1998. While web content management is a must, users today expect something more of a CMS; they expect it to also provide a solution for marketing automation.

Marketing automation is a common designation for tools that all contribute to deliver the right message to the right audience at the right time – all by an automated process based on user behaviour and data analytics. You, the marketer, produce the content and define some key parameters, before unleashing the automated machine on website visitors, email readers, and social media users.



Today some CMSs provide integrated marketing automation tools, while others offer integrations through plugins or apps. Some CMSs don't feature marketing automation tools or the possibility for them at all and need to be customised by developers to make the glove fit the hand.

If marketing automation is your cup of tea, and if you're working with marketing it probably is, you have to make sure that any CMS you consider have the ability to integrate with these kinds of tools in one way or the other.

Marketing automation tools give you the opportunity to do the following functions, among others:

- Effective delivery of messages and follow-up of leads
- A/B testing
- Personalisation
- Analytics
- Content marketing

## Design

The design or presentation is closely linked to your CMS, but as we saw under “headless CMS” in the section about different types of CMSs, it is possible to separate presentation and content. The goal is to send editorial content to other platforms and devices where traditional desktop or mobile web browser visuals do not apply.

However, for our purposes here we will talk about the design and the content as one side each of the same coin that is the website – wherein the coin-flipping hand is the CMS. The CMS, especially if it is a traditional or a decoupled CMS where you can choose when and where to make it headless, executes both the form and the style, just like you decide the spin and trajectory of the coin when tossing it.

What does this mean? Isn't the design of the website controlled by CSS (cascading style sheets) and HTML (hypertext markup language) anyway? Well, yes, but one thing is to ornament your skyscraper with nice fronts, touches and details, but what decides the blueprint and the support beams for the building itself? The architecture. And the architecture in our digital marketing world is the database and CMS itself; it lays out the guidelines and the basics, whereupon HTML (the structure) and CSS (the layout) add some finishing touches, albeit nice and necessary ones at that. The same principle applies to responsive and adaptive design.

## Back-end affects front-end

When considering a CMS, you should be aware of the structure and hierarchy of content in the back-end. For instance, in WordPress there is a strict distinction between what is known as “posts” and “pages”, and these two different content types decide certain behaviours, placements, and functionality on the finished website. In Adobe Experience Manager you can structure your entire site in a tree menu, decide various content types, like “topic page”, “service page”, “news article”, etc., and then enforce rules of what types of pages are allowed to be displayed where on the live site.



Keep in mind how the seemingly internal rules of a CMS in regard to the structuring of your various pages and files most likely have a direct impact on the possibilities of how you can set up your website and how it looks.

If you are on the lookout for a new CMS for your organisation, make sure it offers you flexibility in how you set up your entire site, with full control over the tree structure, category organisation, content type allocation, and so on. Not only does this solve several tasks for you (more on this further down under “site hierarchy” and “usability”); it also provides your marketing team with a sandbox of guidelines for what elements your design should prioritise in order to emphasise your most important content, offerings, etc.

## Templating support for changing designs

A template is a quick and easy way to make a professional looking resume in a text editor program or a gorgeous slideshow in a presentation program, but templating also applies to websites. Your chosen CMS should offer a way to template both entire pages and individual modules or elements within pages.

Templating makes it easy to switch the looks of your site, but also to implement new functionality fast and easy, as this only involves changing one parameter instead of copy-pasting long lines of HTML. Setting up and quality checking different templates is a way to make your content compliant with both brand guidelines and technical requirements.



Support for templating makes the running of a website not only flexible for the site as a whole, but also for all the modular parts, which you can use, re-use, and cross-use all over the site in a matter of seconds.

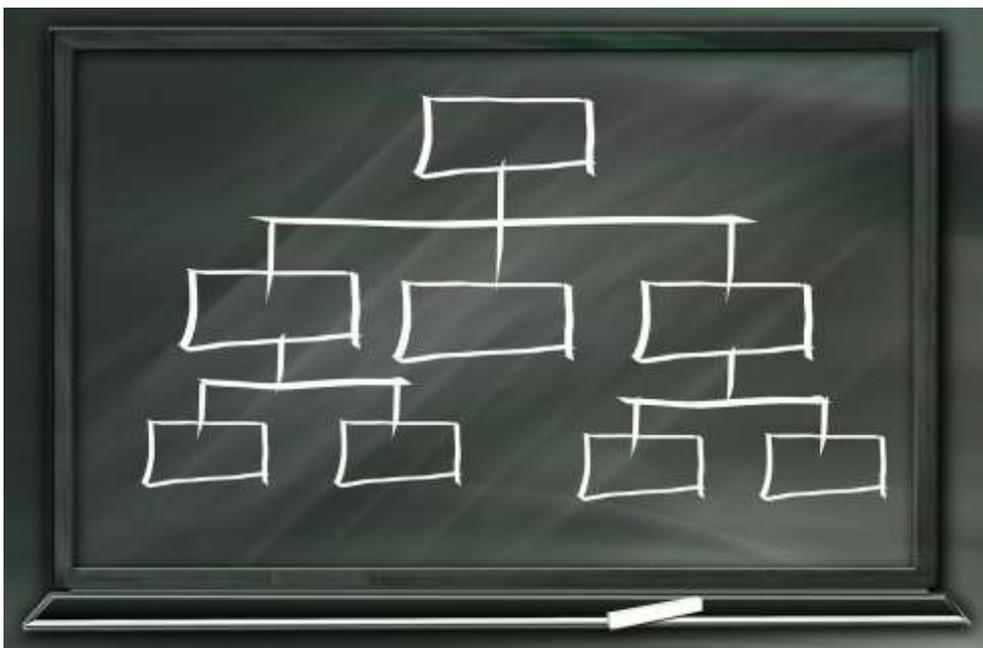
A quick checklist for design considerations should be in order. A design-friendly CMS should enable you to:

- Easily decouple the presentation layer (the design) from the content layer
- Set up a hierarchical site structure that can work as a blueprint or support beams for the live website
- Add flexibility for the developers, to let them innovate and add codes without breaking the presentation part of the website or app
- Support templating for easy changes and modular workings of the design

## General site hierarchy

We mentioned that the ordering of website pages in a CMS is instrumental to the looks and feeling of your finished product. Design is one thing, and another important aspect of a CMS' control over site hierarchy is for your own convenience: order.

Most people probably like to keep some semblance of order and tidiness in their homes, minds, and daily chores. Why should the tool for your online digital experiences be any different? Any great CMS keeps your content in an orderly fashion, with a clear-cut hierarchy of folders and files giving you a sense of context. The purpose of order, context, and hierarchy is to enable you and your co-workers to easily browse and retrieve published and unpublished content, as well as getting an overall picture of what task your site is supposed to fulfil.



How you organise your site is entirely up to you (or your organisation), but your chosen CMS should at least feature the following functionality regarding site hierarchy:

- A panel for a complete overview of the elements of your website (pages, images, fragments, etc.)
- Easy navigation of said overview
- Sorting options for a variety of preferences, e.g. sorting manually, alphabetically, and chronologically
- Closely connected to a toolbar, allowing you to easily search, create, delete, edit, duplicate, move, and preview content

Are there any general advice for best practice when it comes to organising your content within the site hierarchy? It all depends on what you need. If your CMS has a great search function, tight hierarchical organisation for browsing purposes might not be necessary. Some CMSs use a separate, silo-esque method of uploading, storing, and retrieving images in the same complex of folders and subfolders, but again, a good search function renders this unnecessary. If you only would like to get an overview of your website offering, maybe an XML sitemap will do the trick instead of rigid structuring in the CMS. It's all up to you and what your organisation needs, but make sure that the functionality of the CMS does a decent job in aiding you.

## URL logic

Another nifty function you can fully control with a CMS is the URL logic. This is closely related to the site hierarchy, as in most CMSs you need to set up pages and subpages in the same way they will appear in the address bar, also known as the URL.

For instance, say you work for a bank that has a topic page about investing, with subpages about stocks, bonds, and mutual funds. The topic page about investing has the address:

- “example-bank.com/investing”

While the other three have:

- “example-bank.com/investing/stocks”
- “example-bank.com/investing/bonds”
- “example-bank.com/investing/mutual-funds”

To achieve this logic under normal circumstances, you need to place all the three subpages under the topic page in the CMS site hierarchy:

- example-bank.com
  - investing
    - stocks
    - bonds
    - mutual-funds

## Vanity URL

When working with URLs in CMSs, you may also have heard about something called vanity URLs. What are those? In short, a vanity URL is a unique web address that is shortened to fit a marketing or usability purpose. An example may be as follows:

- Original URL: [example-bank.com/blog/2018-07-05/how-invest-your-money-financial-sector](http://example-bank.com/blog/2018-07-05/how-invest-your-money-financial-sector)
- Vanity URL: [example-bank.com/investing](http://example-bank.com/investing)

The purpose is to make URLs for important pages shorter and easier to remember, while keeping the original site structure in one piece. The vanity URL redirects to the original URL, but often you may choose to keep or discard the new and cosier URL in the address bar.

# Editorial management in the CMS

## Publish dates

If content is king, then timing is queen. Together, this royal couple can make or break a marketing strategy, and as you probably would have the king and queen make your strategy a success rather than a failure, you not only need to pay attention to content, but to timing as well – i.e. publish dates.

Back in the days of yore all kinds of organisations controlled when their ads, marketing material, and information brochures were distributed – when content were sent out with the postal service, when it was displayed in the newsstands, when invitations to events were distributed. This need of timely control has not evaporated with the emergence of digital technology. Today management of the content's timing is more crucial than ever, in a world where the Internet flows over with content. You need to reach the right audience with the right message at the right time.



Hence, most content management systems give you advanced control of drafting, publishing, unpublishing, republishing, on and off times, as well as detailed control over the publish date down to minutes and seconds.

Being able to plan and schedule content in the future enables you to have a steady flow of content, e.g. on a blog, making the experience predictable for the consumers. The ability to schedule content on a point of time when nobody in your team is available, but when your readers are most readily available to dive into it, is also a nice perk to have in mind.

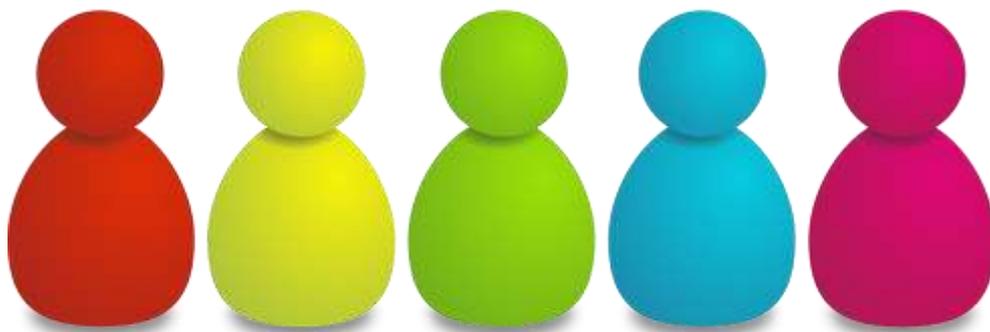
## On and off time

While most of the aspects of publishing might be self-explanatory, a feature like “on and of time” might not be. This function basically decides when a specific piece of content will be available and when it will not. Say, for instance, that your task is to organise dozens of events, and every event has its own page. Instead of manually returning to each event when it’s too late to sign up, you specify when the article will be unpublished, i.e. when it will be “turned off”.

Another example may be a seasonal event, say, a branded summer campaign. It makes no sense that this is available during the cold winter months, and through the CMS you can decide that the page will be “on” in May, June, July, and August, while it will be off in the remaining months.

## User roles

Most organisations have a hierarchy, i.e. a system where members are ranked above, below or at the same level as others according to relative status or authority. A CEO is higher in the hierarchy than a janitor, and the former has more executive power and insights into all the statements and records of the organisation than the latter.



The transferring of this principle to a CMS makes sense: Some co-workers have more responsibility than others, and yet others have a specific, technical function no one else in the organisation can replace. A person hired to write copy should not have access to change the structure of the website, an editor should not have access to change the codes behind the website, while a developer should have full access to all levels to code and quality check the test results in all facets, and so on.

Any CMS with respect for itself and the security of its customers have the possibility to manage users and roles. Roles vary depending on your needs, but common roles include:

- Administrator
- Editor
- Author
- Contributor

Your chosen CMS should provide you with an overview where you can fully control who gets to see and do what of tasks in the system. Who has the right to change code? Who has the right to rearrange the page hierarchy? Who has the right to only write articles? The user role control panel should help you decide.

## Revision and version control

You discover a most grievous mistake in one of your articles; a paragraph with a well written and thoroughly reasoned arguments have accidentally been deleted. And worse yet: the content was deleted several days ago! If this was the late 90s and your software was a text editing program, your carefully executed lines of text would now be gone forever.

Enter the new millennium, a time and place where you can undo these kinds of subversive mistakes with a beautiful thing called “revisions”. With the advent of services like Wikipedia and Google Docs, a version history is now always present. Every time you save changes in Wikipedia or do anything in a Google Doc, the system saves the previous version at the same time it both saves and publishes the new and current version.

Make sure that your CMS includes this feature, as you always can go back in history to find any given edition of your content, no matter how far back in time, and either bring it all back in its full retro glory or pick snippets to re-include in your current, live web page.

Revisions are an inbuilt security mechanism for your content and ensures that you never lose progress. It is also a form of audit, which you can use to quality check all published content over time and observe what actually did occur in the past. In some ways revisions are audit logs in a way similar to blockchain, used in e.g. bitcoin, where all transactions are automatically stored in nodes for everyone in the network to see and verify.



### Version control: is it published or not?

Publishing and bringing back lost content is one thing, but knowing and having full control over live versions and changes to it is another matter. When you have published content in some CMSs, all the following edits are made live on the fly. But what if you want change certain elements, but postpone the publishing to a later date?

There is a difference between saving and publishing / updating your content, but not all content management systems take this feature into account. If this is an important issue to you, make sure your chosen CMS includes the possibility to not only publish and saving changes to published content before updating live, but also to make this distinction crystal clear to you in the user interface.

## Accessibility and usability

Accessibility is the principle that sight or hearing-impaired people should be able to see and hear the contents of your website and navigate it easily. The more people who can consume your content, the merrier – and the same principle applies to your producers: the more who can use the CMS, the better. If the CMS fulfils accessibility standards, it also enables any person to contribute to your website.

One well-known accessibility standard is WAI-ARIA, short for “web accessibility initiative – accessible rich internet applications”. This is a technical specification by the World Wide Web Consortium (W3C), specifying how you can increase the accessibility of your web pages.



Certain functions of modern websites with dynamic content and advanced user interfaces are not available for people with disabilities. E.g., people who rely on screen readers or who cannot use a mouse might have a hard time navigating certain websites. This is where standards like WAI-ARIA come into the picture, to help you helping your audience.

There is, however, a divide between what you see externally on your website and how your CMS looks and feels “internally”. As this is a page about CMS, we concentrate on the latter. How accessible is your CMS, and how easy is it to use?

Accessibility is often connected to usability and user friendliness / user experience (UX). UX is a large topic by itself. Some general UX principles to consider in your CMS include:

- Minimise the number of options available for users
- Incorporate auto-save functionality and robust error-handling
- Provide task-based interfaces
- Hide technical details from end-users
- Provide efficient user interfaces
- Provide help and instructions
- Minimise training required
- Support self-sufficiency
- Present the CMS in a logical organisation
- Have clear and easy to understand iconography
- Enforce logical and neat naming conventions
- Provide thorough documentation
- Be aware how third-party software affects your CMS organisation

Of course, a lot more could be said and written about UX. You should see this as a start, and not as a detailed investigation about how you can think about usability and accessibility in your chosen CMS.

# The future of CMS

We now know what a content management system is and what you could do and should do with it. But where is the CMS headed? What is the future for the technology? Let's have a brief look at the past before we delve into the future.

## Brief history of CMS

In the 1990s most organisations and persons with an online presence made their websites by hard-coding static HTML pages. By the late 1990s progress in programming languages enabled developers to make site owners and editorial teams to edit content on their own websites. This gave way for the birth of the content management systems.

A CMS allowed users to write, copy, upload images and publish to the web without being especially knowledgeable about technical stuff. Most people wrote their own, custom CMSs, but the era also witnessed the advent of the commercial CMS, wherein a company sold a supported CMS to other organisations.

In the early 2000s several open-source CMSs began to appear, increasing in popularity together with the commercial, proprietary CMSs all the way up to present day. In the late 2000s and early 2010s user needs changed from just being able to produce and publish on the web to include more rich marketing abilities and features, as well as addressing the issue of multichannel presence. This is where we are today.



## What comes next?

The development of content management systems has certainly not been standing still since the 1990s, but what can we expect for the next decade?

### Cloud

Historically, local deployments have dominated the CMS arena. With the rapidly rising popularity of cloud, CMS vendors have started to offer their software as a service (SaaS), effectively reducing the complexity of deployments and hosting. Vendors that have previously been pure software companies are now offering their solutions as a service. Be careful when choosing your solution, as proprietary cloud-only offerings will lock you in compared to software-based solutions and especially open source systems.

### Headless

The future of CMS will remain closely aligned with the web. However, the number of devices and interfaces we use to consume and interact with content has exploded: social media, apps, watches, speech interfaces, virtual and augmented reality, and digital signage are just the tip of the iceberg.

There are numerous ways to feed these sources with content, but for any organisation looking to provide consistent content across channels, the answer is summarised in three letters: API (application programming interface).

Content served through APIs will primarily be structured, simply meaning layout and actual presentation needs to be handled by each client and device. Clients must be able to retrieve content through well-defined APIs, and we can expect to see many different styles of APIs – but web-based APIs will dominate completely. The niche area within the world of CMS is known as headless CMS, and the number of players focusing on cloud-first headless CMS has exploded. Traditional CMS vendors are also providing web APIs for their content, and this is often referred to as a decoupled approach.



## Beyond CMS

The riveting technological development of the web, in both software and hardware, leaves a lot of uncertain factors about the immediate and the more distant future of CMS. But this seems to be certain: Several CMS vendors have moved beyond simple web content management, and many already offer a wide range of marketing tools or integrated marketing suites.

Players in this complex game include Sitecore and Adobe, and their offerings may be labelled as “digital experience platforms,” as we saw earlier in the difference between CMS and digital experience. These kinds of comprehensive solutions seek to manage, deliver and optimise the end-user’s digital experience across a variety of platforms and channels, like the mentioned social media apps, watches, speech interfaces and so on.

In these marketing package solutions, we can observe an increased focus on tools for personalisation and dynamic, contextual marketing – as well as marketing automation and continuous optimisation of the user and customer experience.

What the future essentially holds for CMS is therefore more than web content management, exemplified by integrated marketing suites and subsumed under the term “digital experience.” This means a lot of different tools and services, available through several different platforms and channels, and aimed at different segments of audiences – the right content at the right time to the right people. You can expect these solutions to be available primarily as cloud-based services.

## Progressive web apps

Ever since Steve Jobs introduced the original iPhone in 2007, the Internet has been divided in how it is presented: Usually this is manifested in a brand offering being presented as a regular website in a desktop browser and as a dedicated app on iOS, Android, etc. This two-fold (or x-fold) solution has proven both time consuming and resource intensive: developers basically have had to make the same product twice, thrice or even more.



The multi-platform approach is a waste of time and can be confusing and frustrating for the end-users. Why is functionality  $x$  present in the desktop version of the website, but not in the app? And vice versa. Wouldn't it be nice having just one solution working for several platforms and channels?

Enter progressive web apps. A progressive web application, or PWA for short, is a website able to function like a native mobile app. This means that the developers make one solution, the editors publish in one solution, and the consumers receive a consistent experience across devices.

A PWA is:

- Progressive – works for every user, regardless of browser
- Responsive – fits any form factor
- Connectivity independent – allows offline work or on low quality networks
- Faster / better performance – capitalises on the speed benefits of the web
- App-like – feels like an app with app-style interactions and navigations
- Leaner – much smaller size than a native app
- Fresh – always up-to-date due to service worker update process
- Safe – served via HTTPS
- Discoverable – identifiable as “applications” and allows search engines to find them
- Installable – can be added to the home screen without involving an app store (it is optional to be included)
- Linkable – easily shared via a URL and not requiring a complex installation

Examples of PWAs include Twitter, Tinder, and Debenhams, and the development of the technology is supported by Google, Microsoft and Apple. Several of the early test cases have reported about increased conversion rates, making PWAs a must for any future proof organisation.

A CMS for the future should support PWAs. When you are researching your next content management system, make sure it is able to work with progressive web app technology.

## Future proofing your CMS

No matter what CMS you finally go for, there are several ground rules to check and common technical aspects you should consider in order to make your CMS ready for the future. You have already read several tips throughout this guide, and here are some more:

- **Flexibility** – your CMS should be adaptable to any kind of need of your organisation, either in its core or through integrations
- **Support** – the CMS should support all designs and types of content you can throw at it
- **Image scaling** – handling images both in editing and in presentation is no luxury, it's a necessity in an increasingly mobile world

- **Language support** – as the web more or less has rendered national boundaries moot, any website or app should offer multilingual support, and the back-end is no exception
- **Integrations** – having a close relationship with flexibility, integrations are key to staying ahead, as your CMS will not be able to offer everything to everyone. Custom solutions always make their appearances, and a future proof CMS takes this into consideration
- **Scaling and performance** – whether you run a small association or are a part of a large company, the CMS must grow alongside your traffic and needs. A CMS should offer a robust foundation, on which you can build and scale elements in the system seamlessly, as well as making sure the overall performance of the system is kept smooth

## CMS alternatives

Now you know what a content management system is, the possibilities it offers, some best practice advice, and where the CMS development might be heading for the future. However, an important question arises: what CMS should your organisation choose? There are literally hundreds out there, so how do you orient yourself in this seemingly confusing maze?

One obvious starting point is the list of content management systems on Wikipedia. This list is quite technical and is primarily aimed at developers, so be sure to involve your more technical proficient colleagues if you want to read it closer.



Another tactic in your strategy to find a suitable CMS is to use Google and search for keywords you and your team values along with “CMS”, for instance “headless CMS” or “flexible fast easy CMS”. Use your time and be patient, do not pick the first nice looking content management system you happen to stumble upon. Note the properties of the different systems and weigh the pros and cons against each other, all according to what your team and organisation need and want.

## WordPress versus the world

If you go and check out the statistics of the most popular content management systems, one CMS in particular stands out. You have probably guessed it already: WordPress sits on the CMS throne with a whopping 51 percent of the market share.

What does WordPress offer that makes it so appealing to so many actors across the world wide web? Here are some WordPress features:

- Does not require specialised knowledge about code or web design
- Fairly easy to set up and maintain
- The user interface is easy to understand
- WordPress is both a blog host (.com) and a blogging software (.org), with the former introducing many potential users for the functions they can continue to use with ease and familiarity if they decide to go for the latter
- Plugins provide a solution to many different needs that WordPress does not offer in its core package
- A sea of different themes lets you handpick the design you want
- Active community keeps the development alive

There is no doubt that WordPress is a popular and user-friendly platform, but a whole host of e-commerce sites, businesses and larger organisations use other content management systems. Why is that? This might be due to WordPress having some shortcomings:

- **Security** – WordPress offers plugins to do just about anything you can think of, from changing database parameters and altering JavaScript codes to offering SEO tools and email forms. Most of these plugins are developed by third parties, both companies and private persons alike, and they pose a threat to the general security of your site in two ways: 1) all plugins have an extra set of codes that can disturb or ruin some function of your website, and 2) all the extra code basically builds new doors to your stone wall that is the core CMS, allowing malevolent actors new potentially weak entryways into your kingdom

- **Customisation** – Although WordPress offers themes, making your website look and feel as you wish, they do not alter the core of the system. WordPress offers little customisation in its core functionality, despite the outward appearances and features that you can change with themes, widgets, and plugins. As we have seen, third-party elements like themes and plugins pose an unnecessary threat, something businesses cannot tolerate. The lack of customisation therefore makes brands and larger organisations choose solutions they can tailor together with dedicated consultants from scratch, fitting their exact needs and in compliance with risk guidelines

Don't get us wrong. WordPress is an ideal tool for blogs, personal websites, and small-scale websites, but every tool must fit its purpose. Just as a carpenter would never use a wood screw in a concrete wall or vice versa, you should not select a solution that doesn't fulfil your organisation's requirements of scale, flexibility, security, performance, and so on in regard to your digital experiences.

## Essential features for B2B, SME, etc.

So, if you work in business to business (B2B) or a small or medium-sized enterprise (SME), what should you look for in an alternative to WordPress or your current CMS? Well, we have actually written this entire guide explaining just that. But, since we are the good guys, here are the most important essentials repeated. This is what you should look for in your next CMS:

- **Decoupled CMS** – Need a traditional or headless CMS? Why not choose both, when (and if!) it suits you, in the form of decoupled CMS?
- **Supported** – Is the CMS actively supported by the developers or an external agency? Do your research and ask around, they are probably eager to answer your questions.
- **Scalable and modular** – Try to understand how the CMS works at the core. Will it be able to add functionality and modules, as well as trim off unnecessary parts if needed?
- **Responsive** – No matter how or when you edit content, you should be able to do it on any device and on any bandwidth
- **User friendly** – Keep in mind that your CMS should be easily understood and easily handled by your users, with a focus on logical navigation, accessibility functions, and so on

- **Marketing friendly** – A modern CMS with respect for itself incorporates tools that makes engaging with customers and the target audience easier, namely marketing automation, SEO, personalisation, content marketing, and collaboration tools
- **Revision and version control** – Never lose a day's work (or a minute's, for that matter) with revisions and have a full overview of what you show externally on your digital experiences with version control
- **User management** – Who has access to what? Any CMS that takes risk precautions provides you with a clear overview of users and user privileges
- **Future proof** – Take an extra look on the developers behind the CMS: are they an open source community? What frequency is the CMS updated at? Does the developer / vendor / support seem stable and long-term? Together with all the aforementioned aspects, these factors will give you a clue if the CMS is future proof, and therefore worth investing in for you and your organisation



## CMS alternatives for B2B and SME

There are several giant CMSs / WCMs / DXPs out there, like Adobe Experience Manager and SiteCore. These solutions offer tons of features and functionality, and are primarily aimed toward large, multinational corporations that have other needs than small and medium-size enterprises, like a website hierarchy across the globe with content inheritances, large and dedicated support teams, integral advanced analytics, and several other technical considerations.

The segment between private citizens, associations and tiny organisations on the one hand and large, multinational corporations on the other is, however, a notable one. Small and medium-sized enterprises and organisations make up a sizable portion of the economy, and often have their own needs of complex websites and custom-made services.

Cross-check the essential features mentioned above with the CMS list on Wikipedia to find a serious, professional and long-term CMS provider, catering to your specific needs.

Enonic XP is a CMS and web application platform in one that fulfil all the essential requirements of a future proof system. The platform enables you to build progressive web apps, web-based APIs and / or complex websites. As the CMS is decoupled, you can create landing pages and traditional websites, as well as loading the editorial content onto any device or client.

Add to this a responsive and lean drag and drop editor, support for multiple sites and languages, structured content hierarchy, advanced image editing, permissions and user roles management, bulk publishing, a verified “app store” for marketing tools integrations, and much more, and you might have the CMS you are looking for.

## Closing remarks

So, there you have it! Every essential thing you need to know about content management systems. Use this guide as a reference guide whenever you're stuck or have forgotten some important point when researching or exploring your CMS of choice. Good luck in your quest to find the perfect CMS for you!



This guide is proudly delivered by Enonic and written by Vegard Hovland Ottervig.



Vegard Hovland Ottervig holds a master's degree in film studies and has worked with journalism and marketing since 2010. He loves cycling, philosophy, gaming, and writing.