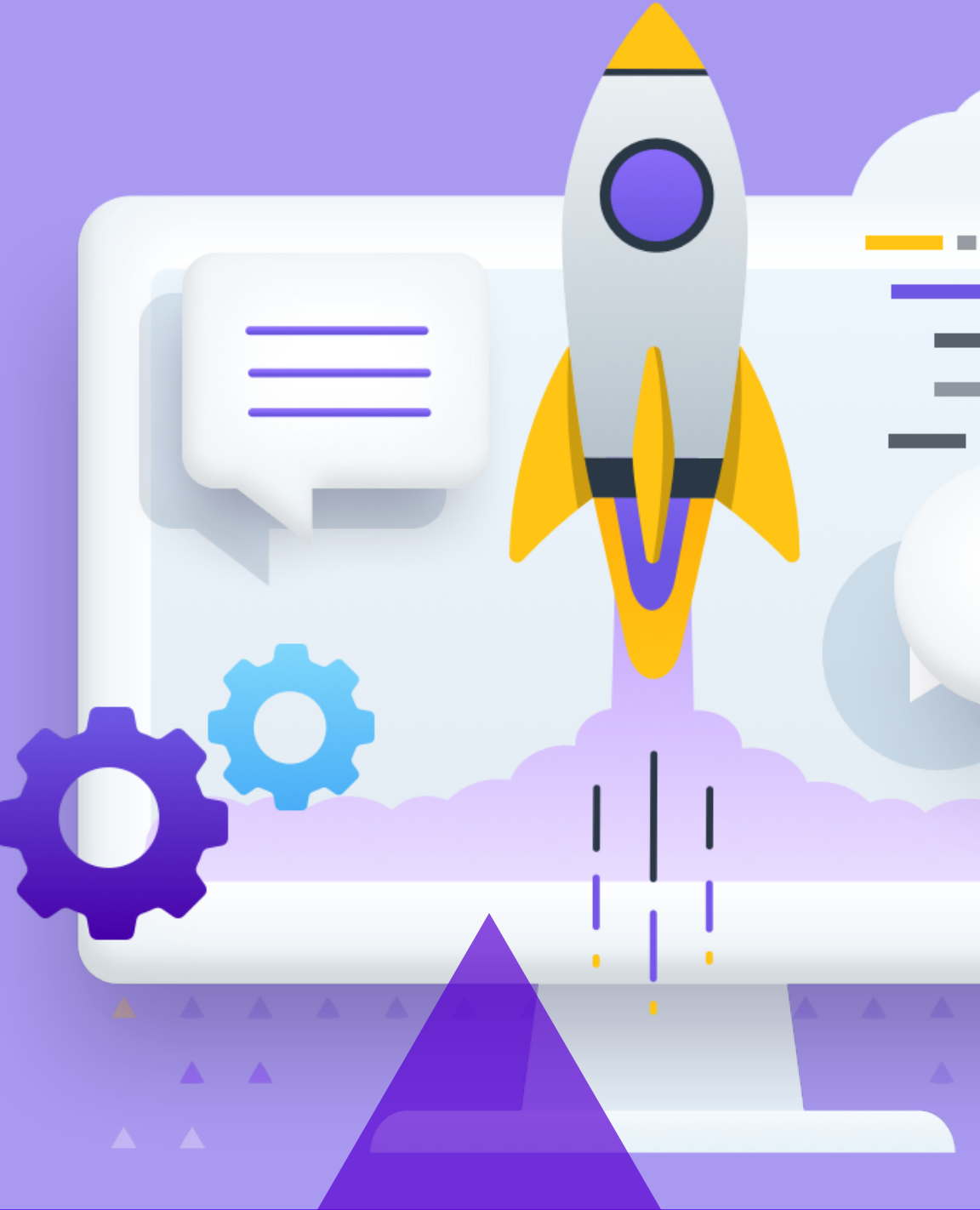


# Why Agility CMS has Page Management?

agility

Guide

The main benefit of headless CMS is that it provides a much more flexible and efficient way to maintain a sophisticated content architecture across multiple digital properties.

The "compose once, publish everywhere" paradigm is a major factor in why folks choose headless CMS. Send content to mobile apps, websites, social networks, marketing automation platforms, and more... the list is as long as your digital strategy. Most headless platforms eschew the concept of "page management."

They claim that the separation of content from its presentation disallows the management of "pages," as they are only intended for a single output - your website. While this is true, it also ignores the fact that the main anchor for online content is your website, and it needs a special case.

*We see websites as the anchor for your digital experiences, which is why we give websites special treatment.*

# Why Page Management?

One of the main benefits of a headless CMS is that it enables companies to build a modern content architecture across different digital platforms and devices. Headless CMSs allow content teams to compose once and then publish everywhere, sending content to every platform and device customers are using.

However, while headless CMSs give you robust functionalities, some vendors claim that the separation of content and presentation that characterizes the headless architecture makes the concept of page management obsolete. In most headless CMSs you often get a blank site that you can configure the way you like. While that's a great approach, we also provide out-of-the-box, and even go a bit beyond that.

Despite being headless, we think differently. We see websites as the anchor for your digital experiences, which is why we give websites special treatment.
Since part of our premise is to make things easier for everybody, we've added a robust page management system that's meant to simplify the lives of the content team, enabling them to create and manage actual pages on our sitemap.

One of the things that makes Agility CMS different from other CMSs is that it is content first and offloads the task of page management to the content marketing team. With Agility, content teams are enabled to update pages and sitemaps without the help of the IT team.



*The Agility team is knowledgeable and works diligently to ensure the success of its clients. The platform is flexible and powerful enough for all of our projects while remaining approachable for a range of non-technical staff.*

CRAIG CAMERON
*Director, Web and Digital Properties at Hockey Canada*

# Page Management

## Benefits of Page Management

Using Page Management, you can empower editors to create and manage pages for your digital solutions using re-usable building blocks (i.e. modules & page templates).

Editors can manage your site's page tree, page-level SEO properties, and determine what content and functionality will be on each page.

As a developer and architect, you still have full control over what page templates are available to the editor, where they can place modules within the page, and what the modules can do.

- Empowered editors who can do more without a developer
- Happier developers who can focus on new functionality and enhancements, and less time responding to new content requirements
- Increased productivity
- Fewer resources/expenses required

## Why do you need Page Management in Headless CMS?

We've curated a list of common scenarios showcasing situations editors and/or developers may have experienced:

I need to create a new landing page specifically for the press. It will have the exact same content layout as the home page, but just different content.

I need to be able to set the SEO properties for each page such as Meta Tags and Meta Descriptions.

On the home page, we want the YoutTube video to be BELOW the Main Rich Text, but on our press page, we want the YouTube video to remain ABOVE the Main Rich Text.

Our CEO doesn't like it, can we have the YouTube video ABOVE the Main Rich Text area on the homepage again?

In the scenarios above, you can see that the editor's needs are impossible to predict. It's not their fault, it's just the nature of the game. Managing content comes with a lot of changes.

Not taking advantage of Page Management for your digital solution means It's not flexible for editors, so developers spend most of their time taking orders, tinkering with existing code to support content requirements, and wishing they were doing something else! Content changes should not have to involve developers.
So what does this ultimately mean?

- Editors cannot create/manage pages on their own without a developer
- Editors cannot control which components are on each page
- Developers get bored or burnt out
- More development resources/expenses required
- Productivity on the website suffers

And, who's at fault in this? I'll give you a hint, it's not the editor, and it's not the developer... It's the architecture!

Chat with Expert >

## The Elements of Page Management In Agility CMS

With Agility CMS, you can define three things that make development and editing simple. Page templates, module definitions, and digital channels and sitemaps. Let's take a look at these three elements.

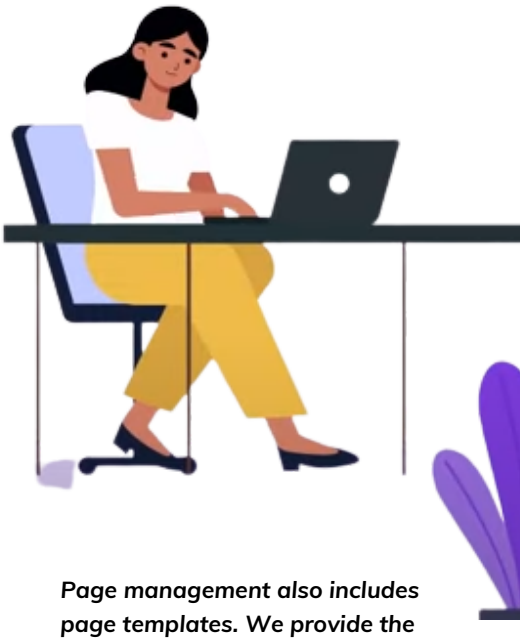
- **Page Templates:** Templates allow developers to determine a name and a set of modules where content editors can add content. Developers can also write code to render this template into an HTML document.
- **Modules:** Modules are reusable sets of fields that can go from a header to more complex UI elements such as a sliding image carousel. In short, modules are the building blocks that enable complex websites.
- **Digital Channels and Sitemaps:** A sitemap is a content structure that developers use to render the website. In Agility CMS, websites have one or more sitemaps defined, each in its own channel, enabling multi-site management and granular control over each digital property.

## Headless Page Management In Agility CMS

Most developers and architects start building their content using simple content definitions, and while that's a good way to get started, it doesn't scale well.

That's why we offer users a built-in page management system.
Page management empowers editors to create and manage content using reusable building blocks, also called modules.

Using these building blocks, editors can manage their site's page tree, edit SEO, and determine content relationships and the functionalities of each page. Also, developers can choose which modules editors can add, where they go, and what they do.



*Page management also includes page templates. We provide the ability to create templates to narrow down the possibilities of what kind of content your website shows, giving the editors the tools to customize them. Each page template has content zones that give editors a location to drag and drop modules.*

## *Empowering Content Editors and Developers.*

*The main thing about Agility CMS is that instead of coding and routing each page individually, developers now have to code only a few page templates and modules.*

*These templates can be reused across the website, sharply reducing the amount of work that developers need to do from the beginning.*

*Now, with the help of page management, content editors can create and move pages around the sitemap in any way they see fit, adding modules to the websites and updating the properties of these modules as well.*

*If you're ready for more effective content management or you're looking to future-proof your company, start a conversation with our experts or contact us to request a demo.*

# How Agility CMS Handles Pages

Agility CMS allows you to define 3 things that make working with pages really easy for both content editors and developers to work with. Developers set up the templates and definitions, while content editors determine how and where they want to use those templates and definitions. This part of the content architecture is fundamental to the success of any CMS solution that includes a website.

## Modules

Modules are the building blocks that enable a content editor to create a complex website structure with tools the developer has set up for them to be successful.

This allows a developer to define a set of fields to render a reusable item inside a module zone of a template. This can be anything as simple as a header with a description that is rendered as an <H2> and a <P> element.

Or, it could be a complex user interface component, such as a slider carousel, contact form, or a listing of blog posts.

The developer has the choice of what properties the content editor will have to work with.

These can be simple text, number, or date fields; they can be image or file attachment fields, or they can link to other content that is defined elsewhere, such as shared content, or can be a contact us form for collecting leads and sending them to your CRM or Personalization extension.

## Digital Channels & Sitemaps

Each Agility CMS website can have one or more sitemaps defined, each in its own digital channel. Each digital channel can be used to control a separate digital property.

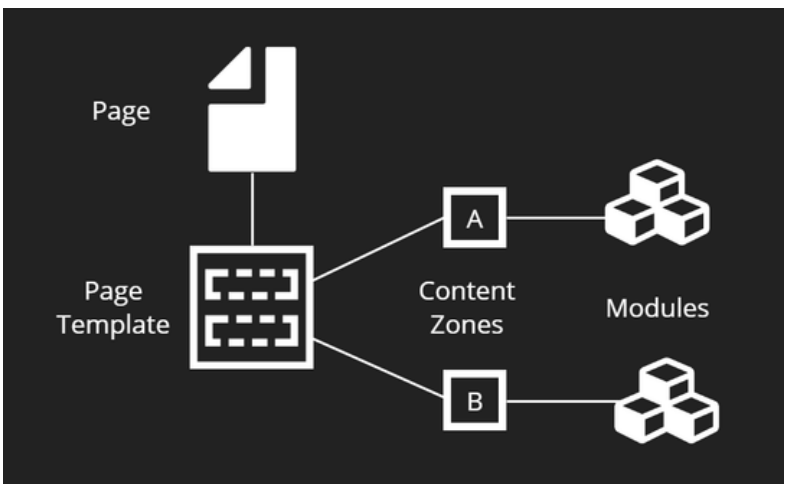
This could, for example, be used for a contest or a completely separate branded site.

Essentially a sitemap is a nested content structure which the developer will use to render on the website. Giving the content editor the control to manage multiple channels of page content is an extremely powerful and cost-effective way to manage content.

## Page Templates

This allows a developer to determine a template name and a set of module zones that content editors can drop content into using modules.

The developer can then write code in the website to render this template into
HTML.



## Shared Content is Paramount

We still believe in a "content first" approach to designing your content architecture, and that starts with the shared content section of Agility CMS.

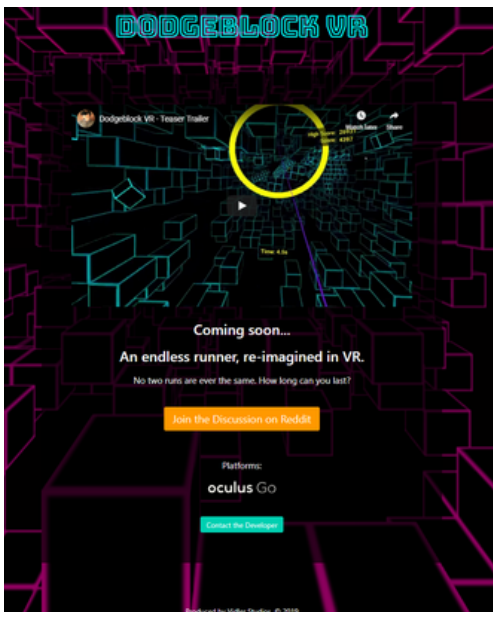
*This is where all your main lists and items will live, and this is where you'll create most of your content that is reused across multiple destinations or endpoints, such as your app, social networks, etc.*

*The pages section, however, is often the first place that users look to find the content they are working on, as the primary responsibility of your content teams is the website itself, and finding what page something on just makes sense to people.*

## *Using Content Models to Manage Page Content:*

*Let's go through some content architecture that is using content models for each type of page and see how it holds up. For simplicity, let's start with a single landing page for an upcoming video game.*

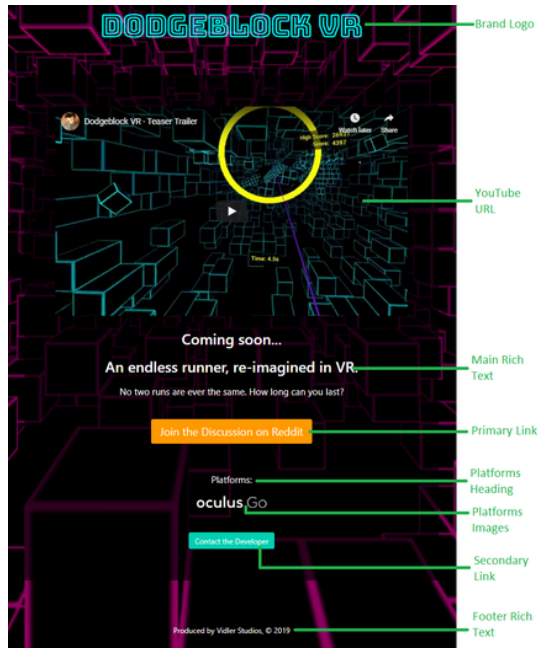
### *Page: Home*



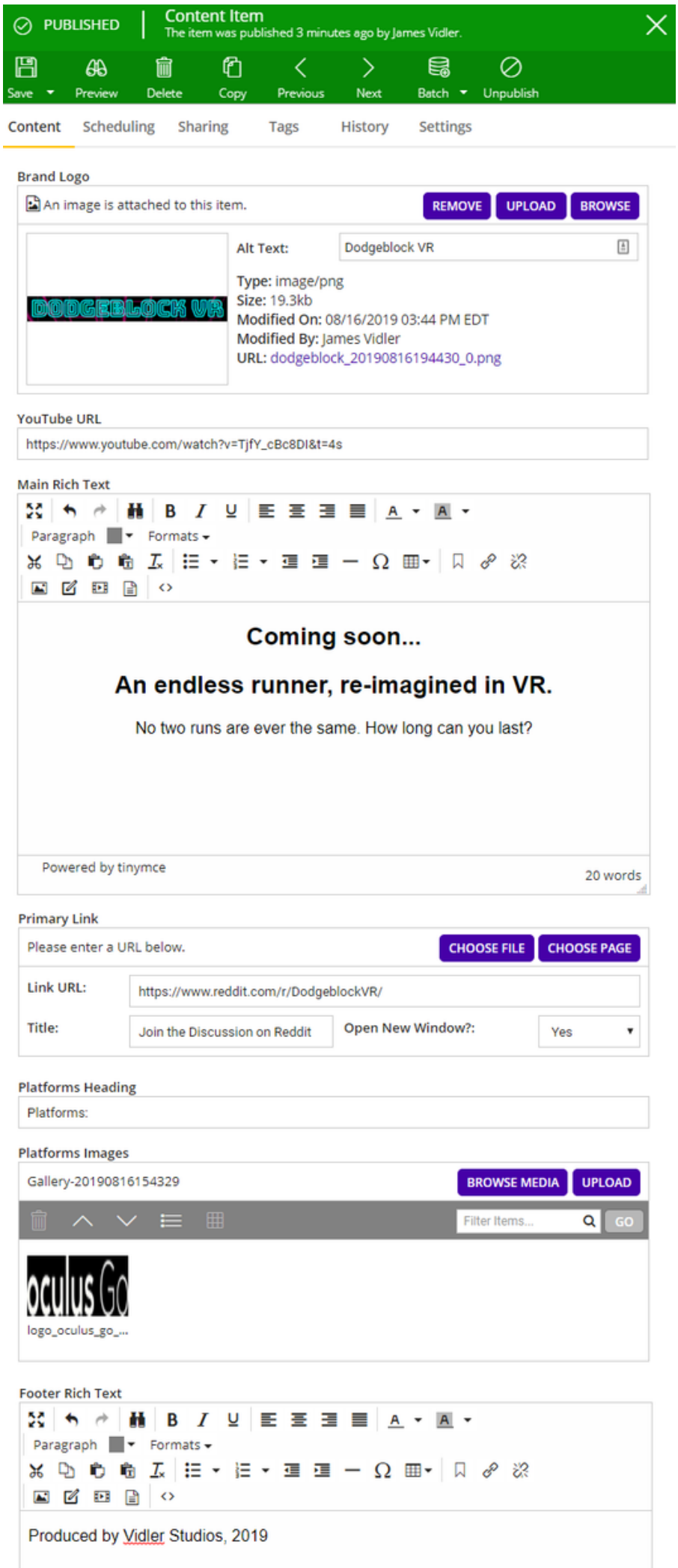

*How would you architect your content for this page?*

*Likely, you would create a Content Model that represents the content that is editable on this page.*

### *Content Model: Landing Page*



*And here's how it might look in the CMS:*



*And here's what you would get back from the Agility CMS Content Fetch API for this item:*

```
curl https://046a1a87-api.agilitycms.cloud/fetch/en-us/item/28
--header "APIKey:
defaultlive.2b7f3a91559d794bedb688358be5e13af2b1e3ae8cd39e8ed2433bbef5d8d6ac"
{
  "contentID": 28,
  "properties": {
    "state": 2,
    "modified": "2019-08-16T15:49:56.583",
    "versionID": 173,
    "referenceName": "dodgeblocklandingpage",
    "definitionName": "LandingPage",
    "itemOrder": 0
  },
  "fields": {
    "brandLogo": {
      "label": "Dodgeblock VR",
      "url": "https://046a1a87-
cdn.agilitycms.cloud/Attachments/NewItems/dodgeblock_20190816194430_0.png",
      "target": null,
      "filesize": 19767,
      "pixelHeight": "54",
      "pixelWidth": "440",
      "height": 54,
      "width": 440
    },
    "youTubeURL": "https://www.youtube.com/watch?v=TjfY_cBc8DI&t=4s",
    "mainRichText": "<h2 style=\"text-align: center;\">Coming soon...</h2>\r\n<h2 style=\"text-align:
center;\">An endless runner, re-imagined in VR.</h2>\r\n<p style=\"text-align: center;\">No two runs
are ever the same. How long can you last?</p>",
    "primaryLink": "<a href=\"https://www.reddit.com/r/DodgeblockVR/\" target=\"_blank\">Join the
Discussion on Reddit</a>",
    "platformsHeading": "Platforms:",
    "platformsImages": {
      "galleryid": 2
    },
    "footerRichText": "<p>Produced by Vidler Studios, 2019</p>"
  }
}
```

*In your code, you would then create a static page, consume this content item via the API and then use its properties to populate the content in the HTML you've already scaffolded out. Easy right?*

*Is there anything wrong with this approach? Well, no. It meets the basic requirements of allowing the editor to change the content and even create versions of this content for other languages.*

*So what's the big deal then? Well, this site and its content are likely to evolve and grow way beyond a single page. Keeping the current architecture in mind, let's see what would have to happen to handle these common requests in the below scenarios.*

# Visit Orlando

Visit Orlando, the official tourism association for Orlando, Florida - home to Disney World and one of the most visited tourist destinations in the US - wanted to increase their engagement, conversion rate metrics, and ticket sales.

To achieve this goal they wanted to manage their content across various channels and personalize marketing campaigns for a targeted audience.

Visit Orlando was severely hampered by an inefficient ticket management system and process and needed to revamp its entire digital strategy and bring everything under one umbrella.

Through Agility's Headless CMS they now have a multi-site, multi-channel content management system and a new ticket sales and POS solution. Their team is now able to focus on adding new properties onto the platform, both as new website instances or as digital channels.

**"In December, my team had taken a 4-day weekend for Thanksgiving for the first time in many years, thanks to Agility CMS!"**

- Shawn Hart, Director of Web Development & Enterprise Applications Visit Orlando



Visit Orlando

# Visit Orlando: 20% increase in revenue with agile Headless Commerce

Agility helped Visit Orlando reduce manual processes and increase revenue by 20%.

**10%**
Conversion rate increased

**20%**
Online Revenue Increased

**25%**
Online engagement increased




agility

*Agility CMS is a Content Management System (CMS) with built-in Hybrid components for better Editor Experience.*

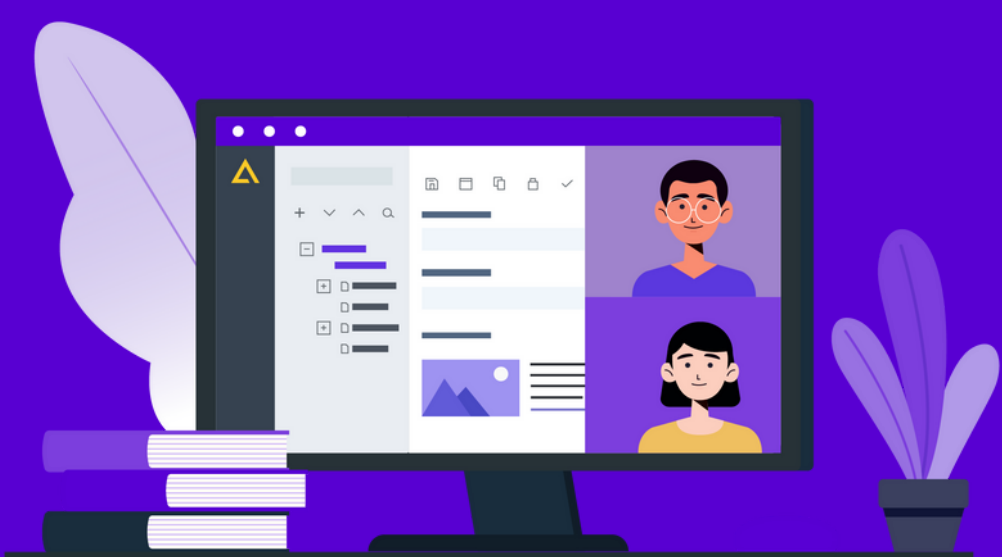
*The platform adopts all the aspect of Headless CMS that makes an omnichannel approach easy, with the understanding that Page Management is of utmost importance in any website project.*

*Agility is proprietary CMS with full Support and Security that gives creative freedom to both Developers and Marketers and eliminates friction between the teams.*

*Developers have full freedom to use any coding language and any framework and code and define custom Page Templates and Module Definitions.*

*These pre-defined custom Modules and Page Templates are available to the Marketing Teams to effortlessly create a variety of pages to meet their fast-changing needs and market requirements. Agility CMS believes in Content First Approach and heavily invests in helping clients to build future-proof Content Architecture.*

Book a Demo >



## Agility CMS Differentiators

- *It is a modern Enterprise-grade Composable DXP with flexible Headless approach in its heart.*
- *Unlike many other Headless CMS, Agility CMS also offers tools for Enterprise teams to effortlessly manage content delivery to websites and apps like Page Management, Sitemaps, Content Modelling.*
- *With Page Management built-in, developers don't have to build an entirely new way to manage pages and routing. At the same time, marketers can easily edit content across a variety of channels without any coding experience.*
- *Agility CMS provides enterprise-level security and support, it is SOC2 compliant.*
- *Agility CMS is Jamstack friendly and allows to start quickly with your favorite frameworks and deploy with a few clicks.*

# Details



***License Model:*** *Agility CMS is a multi-tenant SaaS platform, and hosted on MicrosoftAzure Cloud*



***Price:*** *Starts with FREE Developer Edition. Paid plans have extensive TRIAL. Learn more.*



***Key Components:*** *Page Management, Content Relationships, Content Modelling, POS, Search, User Generated Content, Personalization, Digital Asset Management. Unlimited content models, Locales, API calls,*



***Upgrades:*** *Automatic, monthly platform updates.*

agility