

Housing Solutions Portal

Rebuild (v10)

Specification Home

Table of Contents

Introduction	5
Version History & Stakeholders	6
Project Objectives	7
Common Terms and Definitions	8
Initial Sitemap	9
Umbraco CMS Features	10
Umbraco CMS Features - User Stories	13
Umbraco CMS Features (Custom)	14
Document Types	16
Homepage	17
Homepage - User Stories	18
Homepage (Membership)	19
Homepage (Membership) - User Stories	22
Dashboard	23
Dashboard - User Stories	24
Profile	25
Profile - User Stories	26
Standard Content	27
Standard Content - User Stories	28
Notifications	29
Site Settings	30
Site Settings - User Stories	31
Common Page Compositions	32
Common Page Compositions - User Stories	34
Rent Transactions	35
Payment Page	36
Repairs	37
Report a Repair	38
Schedule a Gas Service	39
Message Centre	40
Staff Login	41
Staff Tenant Search	43
Staff Tenant Impersonation	44
App Chat	45
App Form	46
Content Modules	47
Document Listings	48
Form	49
Grid/Icon Layout	51

Image Gallery	54
Listing	56
Rich text editor	58
Video	60
Content Modules - User Stories	62
Site Template Components	64
Sidebar Content	65
Sidebar Content - User Stories	66
Pagination	67
Pagination - User Stories	68
Site Header	69
Site Header - User Stories	70
Site Navigation - Top Level Only	71
Site Navigation - Top Level Only - User Stories	72
Site Navigation - Dropdown	73
Site Navigation - Dropdown - User Stories	74
Site Footer - User Stories	75
Third-Party Integration	76
App Engine API	77
Umbraco Forms	78
Umbraco Forms - User Stories	80
OneTrust Cookie Tool	81
Testing	82
Not in Scope	83
Sign Off	84

Housing Solutions Portal Rebuild (v10) Specification Home



Welcome to your new space

Use it to create something wonderful.

To start, you might want to:

- **Customise this overview** using the **edit icon** at the top right of this page.
- **Create a new page** by clicking the **+** in the space sidebar, then go ahead and fill it with plans, ideas, or anything else your heart desires.



Need inspiration?

- Get a quick intro into what spaces are, and how to best use them at [Confluence 101: organize your work in spaces](#).
- Check out our guide for ideas on how to [set up your space overview](#).
- If starting from a blank space is daunting, try using one of the [space templates](#) instead.

Introduction

This specification document serves as the main reference document on the project for all Housing Solutions stakeholders as well as Prodo Digital developers and designers.

This document represents a singular documented view of the project to be delivered, and is informed by, and supersedes client briefs and any subsequent discussions that have taken place. The specification document is an ongoing collaborative document and will be updated as and when new requirements within the scope of the project are clarified up to the point of specification sign-off. At this point, those become the agreed deliverables.

Any changes to the agreed project deliverables will be detailed as change requests in a separate document and are subject to analysis to determine whether there is any further impact on costs or delivery dates. Those would be subject to written agreement by both Prodo and Housing Solutions as per the terms of the Master Services Agreement between the parties.

The first version of the specification can be signed off with areas still 'To Be Confirmed' which will be added during subsequent stages of the project.

Assumptions

All areas marked **TBC1** or **TBC2** are to be confirmed with Housing Solutions.

- **TBC1**: These are areas that it is critical to get more information on before development commences.
- **TBC2**: These areas can be confirmed up to the commencement of development.

Version History & Stakeholders

Client: Housing Solutions

Project: Portal Rebuild

Authors: Jan Koren, Helen Pratt

Version: 0.1

Stakeholders

Version	Date	Author	Changes
0.1	28/02/23	Jan Koren, Helen Pratt	First draft

Stakeholders

Name	Role
Jenny Bradshaw	Head of Client Services - Prodo Digital
Laurence Earl	Head of Delivery & Experience - Prodo Digital
Jan Koren	Technical Architect - Prodo Digital
Helen Pratt	Project Manager - Prodo Digital
Richard Harvey	Housing Solutions - Head of Digital Transformation
Sam Denslow	Housing Solutions - Applications Manager

Project Objectives

Background

With Umbraco v7 coming to the end of life, a new version of the Housing Solutions portal is required to be built in v10.

Project Objectives

To rebuild the existing portal using the latest version of Umbraco (version 10), retaining all existing designs and functionality.


Common Terms and Definitions

Content within the website will be defined by document types. These describe the fields that are editable within the content but will not describe how they are displayed to the web site visitor. Below is a reference list of terms and their meanings that will be used throughout this document.

Content Management System (CMS)	A piece of software that is responsible for the management of the content and assets
Umbraco	A widely used opensource CMS
CMS User	A user that is responsible for curating content using the CMS
Website Visitor	An internet user who will access the website
Tenant	A registered tenant who has access to the Portal is the most common portal user type
Leasholder	A leaseholder may own one or more properties and is the second most common portal user type
Staff Member	A registered staff member who has access to the Staff Portal
Component	A component is part of the website which is not part of the modules list this are usual fixed items like the header, footer and banners
Block Editor	The block editor is the place that you select your modules from it's effectively a container for the bulk part of the content population
Module	A module is an individual piece of the site which is added to a page via the block editor.
Document Type	Content within the website will be defined by document types. These describe the fields that are editable within the content but will not describe how they are displayed to the web site visitor

Initial Sitemap

Due to the portal being built into a CMS which will ultimately be controlled by a CMS editor, there is scope for this to be updated both before go live and once the portal is delivered.

 The sitemap will mirror the current structure of the portal which can be altered by Housing Solutions at any stage of the project.

Umbraco CMS Features

The website will be built upon the Umbraco CMS platform enabling rapid and reliable development of the required custom features as well as providing a set of standard and common CMS features. The website will initially be based on the latest stable version of [Umbraco](#) at the time when development starts.

This section will detail the common features provided by Umbraco which can be used on your website.

User Management

Umbraco allows CMS Users to log in, using an email address and password, and manage CMS users with the same or lower user access. CMS Users can be granted various permissions allowing them to perform a variety of actions on the following Umbraco sections:

- Content
- Media
- Users
- Members
- Forms

Two levels of access will be provided for CMS Users of the site. Editors will have access to Content, Media and Forms sections, Admins will also have access to the Members and Users sections.

Prodo will set up the the initial CMS Users for the site. One of the provided users will be required to be an Admin which will allow you to add more users going forward without input from Prodo.

Member Management

Umbraco allows CMS Users to log in, using an email address and password, and manage CMS members. CMS Members can be added to various member groups allowing them to access different sections of the Portal.

Member Groups:

- Tenant
- Homeowner
- Leaseholder

Two levels of access can be provided for CMS Members of the site. Portal administrators will have access to create and manage both tenant and staff members under the member's section in Umbraco.

Member Types:








- Tenant Member
- Staff Member

Prodo will set up the the initial CMS Members for the site. One of the provided members will be set as a staff member and rest of them will be set as portal tenants. More members can be added in Umbraco by portal administrators.



Content Management





Users who have access to the Content section will be able to manage the content on the website. This section will cover the common content management tasks which are provided as default by Umbraco. The content features of each document type are [described later](#) in this document.

Umbraco allows a site structure to be built up using a navigation tree similar to those found on most content management systems. This allows a parent item to contain children and those children items in turn to also contain children. This allows a structured content tree to be set up reflecting the navigational needs of the website.

- ▼  Home
-  Example page
- ▼  News
 -  Category 1
 -  Category 2
-  Search
-  Sitemap

In some cases content items might be displayed in a list view as part of the parent node. This prevents the tree structure becoming visually overwhelming to the editor and improves responsiveness of the UI. Document types that utilise this view will be highlighted later in this specification.

Create News Article
 

<input checked="" type="checkbox"/>	Name	Status	Sort ▲	Last edited	Created by
	Test news article heading 1	Published	0	2020-07-02 13:37	Prodo Developers
	A slightly longer test news article heading t...	Published	1	2020-07-02 13:40	Prodo Developers
	Another interesting test news article heading	Published	2	2020-07-02 13:40	Prodo Developers
	Short news title	Published	3	2020-07-02 13:40	Prodo Developers

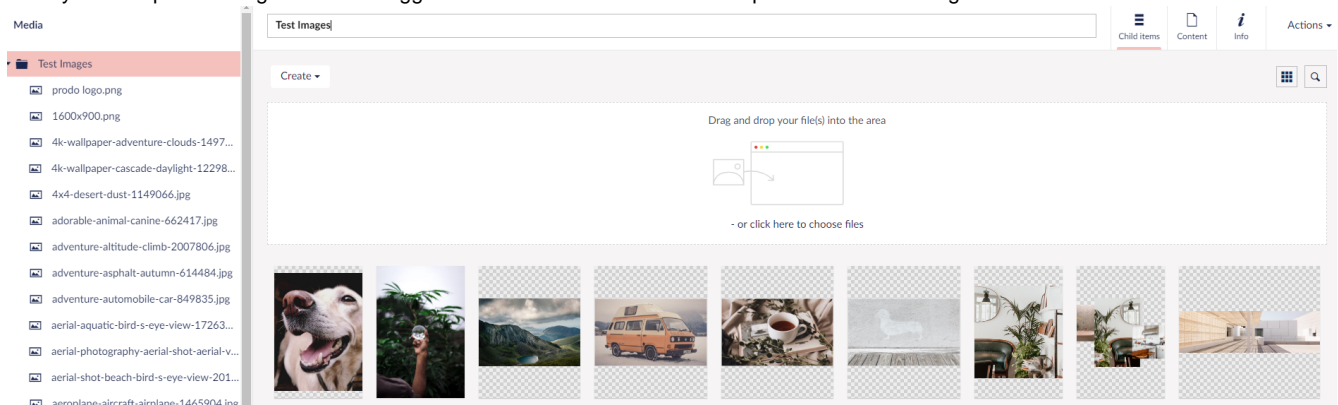
CMS users will be able to create and manage content pages and content elements, such as site settings, within the content section if they have access. Depending on the users access level they will be able to perform the following actions:

Action	Purpose
Create	Allows a CMS user to create a page or element.
Delete	Allows a CMS user to delete a page or element, removing it from the website and moving it to the Recycle Bin. Items can be restored from the Recycle bin or deleted completely meaning it can not be recovered.
Move	Allows a CMS to move a page or element to another valid location.
Copy	Allows a CMS to take a copy of the page or element.
Sort	Allows a CMS user to sort the child items.
Rollback	Allows a CMS user to rollback an item to a previous version.
Audit Trail	Allows a CMS user to view the audit history of the item highlighting when items have been published or updated.
Publish	Allows a CMS user to publish a page or item allowing website visitors to see the content.
Permissions	Allows a CMS user to set user permissions for the item. Such as not allowing an editor to publish a page.

Save	Allows a CMS user to save a version of the item without publishing it.
Preview	Allows a CMS user to see a preview of what the page will look like once published. Not applicable to elements.
Unpublish	Allows a CMS user to remove the page from the website without removing it from the CMS.
Scheduled Publish	Allows a CMS user to set a time and date at which the page should be published in its current saved state.
Scheduled Unpublish	Allows a CMS user to set a time and date at which the page should be unpublished from the website.

Media Management

Umbraco allows an editor to maintain a library of media assets which can be organised into folders and sub folders. Files can be uploaded directly via an upload dialog or can be dragged to the web browser in order to upload them to the targeted location.



Once uploaded, the files will be available on the website to users who know the URL and can be used by any appropriate media pickers. A list of places where the asset is used is shown within Umbraco to assist editor tracking asset use.

For images that have been uploaded to the CMS Umbraco will display an image cropper interface which allows an editor to set a series of image crops based on the original version of the image. This allows the website to display multiple versions of the image at the correct dimensions and crop for different areas on the site. The specific crops, and where they are used on the site, will be defined by the final design.

An editor is also able to delete files from the website via Umbraco. If an asset is deleted via the back office it is first placed into the recycle bin, whilst in this location the file will still be accessible on the website. To remove the asset completely from the site the editor will have to remove it from the recycle bin or use the Remove file(s) option found on the editor page.

Umbraco CMS Features - User Stories

The following user stories will be used to verify the Umbraco CMS Features are working correctly

	Module	User Story	Success Criteria
1	User Management	As a CMS User , I want the ability to log in.	The CMS user will be able to login Umbraco using their Email and Valid Password.
2	User Admin	As a CSM User , I want the ability to add new users if I am an admin	The CMS user will be able to access the User section and add a new user if they are in the admin user group
3	Content Management	As a CMS user , I want the ability to manage content and elements	<p>The CMS user will be able to perform the following actions on pages and elements in line with the content structure of the site:</p> <ul style="list-style-type: none"> • Create • Delete • Move • Copy • Sort • Rollback • View Audit Trail • Save • Preview
4	Content Management	As a CMS user , I want the ability to publish and unpublish page and elements on the website	<p>The CMS user will be able to perform the following actions on pages and elements:</p> <ul style="list-style-type: none"> • Publish • Unpublish • Schedule Publish • Schedule Unpublish

Umbraco CMS Features (Custom)

Scheduled Tasks

Umbraco will feature a dashboard that will list all the scheduled tasks. Each scheduled task will display a name, next execution, last execution, and date when it's been created. These scheduled tasks will run as a background job at the specified time.

There will be the following tasks available on the scheduled task dashboard:

- Send inactivity emails
- Sync member addresses

Member Import

Users who have access to the Members section will be able to upload and import the members into Umbraco. The import will only create the members who aren't registered with Umbraco. Member import will require a specific CSV file to be uploaded. CSV must contain the following headers:

- Firstname
- Surname
- TenancyRef
- Email
- DOB

Member Export

Users who have access to the Members section will be able to export all members. The exported data will be formatted into CSV file with the following headers included:

- First Name
- Last Name
- Tenancy Reference Number
- Person Reference Number
- Date of Birth
- Hear About
- Hear About Staff Name
- Approved
- Is Locked?
- Last Login Date
- Login
- Email
- Created

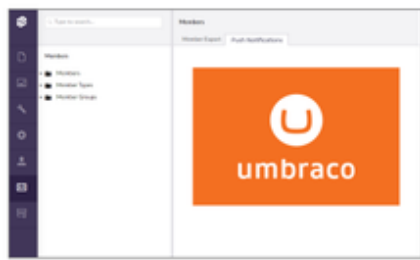
Push Notifications (App)

The Umbraco push notification tool is a custom built page in the members section of the CMS. The look and feel of it will remain consistent with Umbraco so no additional design time will be required. The interface essentially is a multi-step form which allow a CMS user to do the following:

- Select a distribution list, with a group or individual(s)
- Create a title (up to 64 characters)
- Create the message body (up to 240 character)

The following section will give a functional explanation of how the push notifications work. IOS and Android provide their own services to handle push notifications, APNs (Apple Push Notification service) and FCM (Firebase Cloud Messaging). React Native Expo does provide a tool that wraps the integration of both providers into one single API that we can target however we still need to write the code to integrate with the push notification API. There will be a custom Umbraco interface which allows a CMS User to target and send push notifications to either groups of tenants or individuals. Below is a diagram that illustrates the push notification workflow.

Umbraco Push Notification Tool



Brent's Expo Push Token
'ExponentPushToken[I22V7M...]'
message 'Hello!'

Expo backend



notification



if Android



notification



if iOS

Document Types

Content within the website will be defined by document types. These describe the fields that are editable within the content but will not describe how they are displayed to the web site visitor. Some fields on a document type might not directly impact the look of the page but could have further implications for area such as search.

Document types can contain common groups of fields called compositions while also containing more specific fields. These specific fields will be applicable to only the content the document type represents, for example a News Article may contain an Author field where as the Homepage would not contain such content.

Homepage

The homepage serves as the main entry point to the portal and all other pages and content items will be located under it. The following compositions will form part of the homepage document type:

- Heading
- Banner
- Summary
- Social
- SSL & Scripts
- SEO

The homepage will also contain a Block Editor field type allowing the following modules:

- Rich text editor

The Portal Home is a protected node and can not be deleted, the website relies on settings from this node to function properly.

Homepage - User Stories

The following user stories will be used to verify the Homepage is working correctly from the view of a CMS User.

Module	User Story	Success Criteria
Portal Homepage	As a CMS User , I want to be able to manage the portal homepage	The CMS user will be able to view and manage the portal homepage from the content section in Umbraco.
Portal Homepage	As a CMS User , I want to manage all Common Page Compositions of the portal homepage	The CMS user will be able to manage the Common Page Compositions in line with their User Stories

Homepage (Membership)

The portal homepage also allows members to login to the portal in order to access restricted pages. Members are created through a registration process or can be created by CMS users via Umbraco directly.

Login (Portal Home)

The portal homepage will include a login component that will serve as the main entry point for members attempting to access restricted content on the portal. The page will be available to all portal users allowing them to input their member credentials in the form of a username (email) and password.

The following compositions will form part of the homepage document type:

- Heading
- Banner
- Summary
- Social
- SSL & Scripts
- SEO

In addition to the default portal homepage compositions the following fields will be available under the login tab:

Field	Field Type	Notes
Introduction	Rich Text Area	Serves at an introduction to the login functionality providing links to other areas such as registration and password reset.

The login form shown on the page will contain the following fields:

Field	Field Type	Required	Notes
Username	Email	Yes	The members user name associated to their account
Password	Password	Yes	The members password, minimum of 8 characters long containing a number, lowercase letter and uppercase letter

Registration

The portal homepage will also include the registration component where new members are able to provide details in order to gain access to restricted content. The page is available to all portal users allowing them to fill out their details in order to register for an account.

As part of the registration process members will be prompted to validate their provided email address by way of a validation email that will be sent to the provided email address. The member will be required to click on a unique link that will be included in the email. Upon navigating to the link the members profile will be marked as approved and they will be able to log in.

In addition to the default portal homepage compositions the following fields will be available under the registration tab:

Field	Field Type	Notes
Introduction	Rich Text Area	Serves at an introduction to the register functionality providing links to other areas such as login and password reset.
Successful Registration Message	Rich Text Area	The message to show the user when they are successfully registered and can log in
Email Validation Message	Rich Text Area	The message to show the use when they are registered but need to validate their email address
Enable Email Validation	True/False	When set to true will enable the validate email process where a member needs to click on a link in an email before they can log in
Email Validation Content	Rich Text Area	The content to send to the member in the validate email message. Can use the magic strings of <code>{{dateTime}}</code> <code>{{memberName}}</code> & <code>{{tokenLink}}</code>

The registration form will contain the following fields:

Field	Field Type	Required	Notes
Email Address	Email	Yes	The members email, will be used as the username for the member
First Name	Textbox	Yes	The members first name, saved to their profile and used as the first part of the members name in Umbraco

Last Name	Textbox	Yes	The members first name, saved to their profile and used as the second part of the members name in Umbraco
Date of Birth	Date Picker	No	The members date of birth
Reference	Textbox	Yes	The members reference number stored in housing management system e.g.: tenancy or person reference
Password	Password	Yes	The members password, minimum of 8 characters long containing a number, lowercase letter and uppercase letter
Confirm Password	Password	Yes	Must match against the password provided

Forgotten Password

Request Password Reset

The forgotten password pages serves as the main entry point for a member who is unable to log into their account due to a forgotten password. The page is available to all websites users allowing them to fill out their details in order to generate an email that allows them to access the reset password page.

The page will validate if the members details are correct and if so generate a unique reset password link that is valid for one hour. Regardless of if the members details are found or not the same message will be shown on the website in order to reduce the chances of a malicious users gaining insight into the registered email addresses.

The following compositions will form part of the login document type:

- Heading
- Banner
- Summary
- Social
- SSL & Scripts
- SEO

In addition to these composition fields the following fields will be available on the page:

Field	Field Type	Notes
Introduction	Rich Text Area	Serves at an introduction to the forgotten password functionality providing links to other areas such as login or registration
Password Reset Message	Rich Text Area	The message to show once the member has submitted the form
Email Content	Rich Text Area	The content of the email that is sent to the member containing the password reset instructions. Can use the magic strings of <code>{{dateTime}}</code> <code>{{memberName}}</code> & <code>{{tokenLink}}</code>

The forgotten password form will contain the following fields:

Field	Field Type	Required	Notes
Email Address	Email	Yes	The members email that should be linked to the members account

Confirm Password Reset

The forgotten password page allows members to reset a forgotten password as long as the link to the page is valid. The page will check for the presence of a token and email address on the query string, if they do not exists or can not be confirmed then a error message will be shown to the member.

If the members details are confirmed then they will be shown a form allowing them to provide a new password.

In addition to these composition fields the following fields will be available on the page:

Field	Field Type	Notes
Introduction	Rich Text Area	Serves at an introduction to the reset password functionality
Password Reset Message	Rich Text Area	The message to show when the password has been reset

The reset password form will contain the following fields:

Field	Field Type	Required	Notes
-------	------------	----------	-------

Password	Password	Yes	The members password, minimum of 8 characters long containing a number, lowercase letter and uppercase letter
Confirm Password	Password	Yes	Must match against the password provided

Validate Email

The validate email page is the target page for the validate email process, it is where the email sent to the member to validate their email will direct to. The page will validate the token and email address passed in on the URL and if valid will set the members account to approved allowing them to log in.

The following compositions will form part of the login document type:

- Heading
- Banner
- Summary
- SSL & Scripts
- SEO

In addition to these composition fields the following fields will be available on the page:

Field	Field Type	Notes
Email Validated Message	Rich Text Area	The message to show if the email address could be validated
Email Not Validated Message	Rich Text Area	The message to show if the email address could not be validated

Homepage (Membership) - User Stories

The following user stories will be used to verify the Membership section of the site is working correctly.

Module	User Story	Success Criteria
Registration	As a Website Visitor , I want to be able to register on the website in order to access restricted content	<p>The website visitor will be able to access a page allowing them to provide details in order to register an account that is then saved in Umbraco allowing them to access pages that are protected by the Public Access rules defined in Umbraco.</p> <p>The website users must provide an email address, first name, last name and password in order for an account to be created.</p> <p>A website visitor can not create an account using an email address that is already registered.</p> <p>A CMS User should be able to define the messaging shown on the page and also enable the use of a email validation process. When active the user should receive an email with a link allowing them validate their email address.</p> <p>If this process if not active then a user should be marked as approved in Umbraco and allowed to log in immediately.</p>
Login	As a Portal Member , I want to be able to log in to the site in order to access restricted content	The website member should be able to provide their email and password which can then be validated. If valid the user should be able to access pages that are marked as restricted.
Forgotten Password	As a Portal Member , I want a process that allows me to reset my forgotten password	<p>The website member should be able to provide their email address and have a unique link emailed to them which allows them to reset their password .</p> <p>The code should be valid for an hour.</p>
Reset Password	As a Portal Member , I want to be able to reset my forgotten password	The website member should be able to follow the link emailed to them in order to reset their password.
Validate Email	As a CMS User , I want to make sure members emails are correct before they are allowed to access the website	The validate email process should email out a unique link which when clicked on validated the users email and marks them as approved allowing them to log in.

Dashboard

The dashboard section of the portal serves as a landing page after the portal user has successfully logged in. The purpose of the main dashboard is to access different areas of the portal quickly and effectively.

The following compositions will form part of the property search document type:

- Heading
- Summary
- SSL & Scripts
- SEO
- Sidebar Content

In addition to the composition fields the following compositions will be included as part of this document type

- Grid

Grid Composition

Field	Field Type	Notes
Grid Items	List	Grid items property will allow a CMS user to add multiple toolbox items. Each toolbox item will be configurable and will include the following fields: <ul style="list-style-type: none">• Icon• Title• Description• Primary Link• Secondary Link• Link Colour

Toolbox Items Examples

Usual toolbox items that would be accessible from this page are:

- Report a Repair
- Make a Payment
- Rent Transactions
- Repairs
- Profile

Dashboard - User Stories

The following user stories will be used to verify Dashboard section of the site is working correctly.

Module	User Story	Success Criteria
Customer Walkthrough	As a Portal User , I want to understand how to use the portal when I login on for the first time	<p>On first login, the customer will be presented with a step-by-step walkthrough of the key areas of the website so that they understand how the website functions and where to find their details.</p> <p>The tenant will be shown these instructions the first time they log in.</p> <p>The walk through will be available as an option for people who want to refresh their memory.</p>
Customer Walkthrough	As a CMS User , I want to provide tenants with a step-by-step guide to the portal when they first register	<p>The walkthrough steps will be content managed.</p> <p>Each step will have the ability to set:</p> <ul style="list-style-type: none"> • Text • Image/Video <p>Additional steps can be added.</p>
Toolbox Items	As a Portal User , I want to be able to quickly navigate to the key service areas of the portal so that I can easily manage my tenancy(s)	<p>The dashboard will contain a grid of toolbox call-to-actions (CTAs).</p> <p>Each toolbox CTA will include:</p> <ul style="list-style-type: none"> • Icon • Title • Description • Primary link • Optional secondary link • Link Colour <p>Links will drive tenants down actionable pathways e.g. the 'Rent' CTA could include a 'View rent' button and a 'Make a payment' button.</p> <p>There will be two service specific CTAs that can be setup which will pull through dynamic content:</p> <ul style="list-style-type: none"> • Rent: which will pull through the current rent balance, and next payment date • Notifications: this will display a numerical counter if there are any unread notifications.
Toolbox Items	As a CMS User , I can edit, update and remove CTA links in the toolbox so that I can make it easy for users to navigate to key services in the portal	<p>The toolbox CTA's will be fully CMS managed .</p> <p>There will be a dedicated section in the CMS to manage the toolbox CTA's .</p> <p>There are no limitations to the amount of CTAs added to the dashboard.</p> <p>Each CTA will contain content managed:</p> <ul style="list-style-type: none"> • Icon • Title • Description • Primary link • Optional secondary link • Link Colour

Profile

The profile section of the portal provides a summary of all the users details including tenancy details. The profile document type contains all the content that forms a profile page.

The following compositions will form part of the profile document type:

- Heading
- Summary
- Sidebar Content
- SSL & Scripts
- SEO

Update Contact Details

Portal users can update their contact details using the update contact details form. The form will accept the following parameters:

- Email Address
- Home Telephone Number
- Mobile Telephone Number

Update Password

The profile page will also allow the portal user to update their password. The following parameters will be required:

- Current Password
- Change Password
- Confirm New Password

Profile - User Stories

The following user stories will be used to verify the Site Search is working correctly.

Module	User Story	Success Criteria
Profile	As a Portal User , I want to be able to view all of my personal information in one area of the portal so that it is easy for me to reference	<p>A user will be able to navigate to the User Details section from either the main portal navigation or the homepage toolbox control.</p> <p>Two large CTAs will be visible on the top of the profile page containing an anchor link to the form on the same page. An area containing the core tenant account information will be visible below two CTAs. The following fields of information will be displayed to the tenant.</p> <ul style="list-style-type: none"> • Title • Forenames • Surname • Date of Birth • National Insurance Number
Profile	As a Portal User, I want to be able to update my contact details	A user will be able to update their contact details including:

Standard Content

The standard content page serves as a generic page able to hold a range of content. It is built around the block editor field type which allows a multitude of modules to be placed onto the page. As well as the block editor the document type also consists of the [Common Page Compositions](#)

The modules available in the block editor are documented in the Content Modules section of this specification.

The following compositions will form part of the standard content document type:

- Heading
- Summary
- Side Content
- SSL & Scripts
- SEO

Standard Content - User Stories

The following user stories will be used to verify the functionality of the Standard Content page.

Module	User Story	Success Criteria
Standard Content	As a CMS User , I want to be able to create Standard Content pages	The CMS user will be able to create Standard Content pages below the following pages: <ul style="list-style-type: none">• Homepage• Standard Content
Standard Content	As a CMS User , I want to be able to manage the content on a Standard Content page	The CMS User will be able to manage the Common Page Compositions in line with their User Stories The CMS User will be able to populate the modules documented later in this specification.

Notifications

The news section of the website serves as an areas for press releases and updates to be published for a website visitor to read. It is made up of the following document types. TODO

Notifications Hub

TODO

Portal Notification

TODO

Notifications

TODO

Example:

News Article

The news article document type contains all the content that forms a article page. It will be displayed in line with the design.

The following compositions will form part of the news article document type:

- Heading
- Summary
- Social
- SSL & Scripts
- SEO

The news article will also contain a Block Editor field type allowing the following modules:

- Accordion
- Basic Map
- Document Listing
- Form
- Gallery
- Grid Layout
- Lightbox Gallery
- Listing
- Map
- Quote
- Raw HTML
- Rich text editor
- Tabbed Content
- Text Snippet
- Video

In addition to the composition fields the following fields will be included as part of this document type

Field	Field Type	Notes
Article Date	Date time	The date to use as the articles published date, used when ordering articles by date. Will default to create date.
Display Date	Text string	A text override that will be shown to the website visitor. Will not impact sorting date.
Tags	Tags	A selection of tags associated to the article.

Site Settings

The site setting element defines content and settings that have a site wide impact. This ensures page elements such as company name and defined footer links are managed centrally. Site settings is an element type and as such has no visible presence on the website.

Site settings is a protected node it isn't something that can be deleted, the website relies on settings from this node to function properly.

The following fields will be available on site settings:

Company Info

Field Name	Field Type	Required	Notes
Company Logo	Image Picker	Yes	The logo to be used on the website. Shown on all page templates inline with the design
Company Name	Text string	Yes	The Company Name to use on the website.

Footer

Field Name	Field Type	Required	Notes
Social Links Heading	Text string	No	The heading to place above the social links
Footer Links Heading	Text string	No	The heading to place above the footer links
Footer Links	Multiple Link Picker	No	The links to show in the footer. Allows you to pick an internal page or provide an external link along with a name

Scripts

Field Name	Field Type	Required	Notes
Top Head Scripts	Text area	No	Allows an editor to set global top head scripts to be placed on all pages
Head Scripts	Text area	No	Allows an editor to set global head scripts to be placed on all pages
Body Open Scripts	Text area	No	Allows an editor to set global Body Open Scripts on all pages
Body Close Scripts	Text area	No	Allows an editor to set global Body Close Scripts on all pages

Social

Field Name	Field Type	Required	Notes
Default OGP Image	Image Picker	No	The default image to use when sharing a page to a social platform
Social Links	Social Links Picker	No	Allows an editor to provide a set of social links containing a icon, link and title

SEO

Field Name	Field Type	Required	Notes
Website Name	Text string	No	The name used in the page title
Title Format	Text string	No	The format to used for the page title. Allows a user to pull in the page name and website name into the title format

Site Settings - User Stories

The following user stories will be used to verify the Site Settings are working correctly.

	Module	User Story	Success Criteria
1	Company Info	As a CMS User , I want to be able to change the websites main logo and name shown to a Website Visitor .	<p>The CMS User will be able to upload an image which is then shown as the site logo in line with the designs.</p> <p>The CMS user will be able to provide a company name which is used where the company name is shown in the designs.</p>
2	Footer	As a CMS User , I want to be able to control what is shown in the footer.	<p>The CMS User will be able to provide a footer links heading which is shown in the footer.</p> <p>The CMS User will be able to provide a number of footer links which are shown in the footer. A Website Visitor will be able to follow these links.</p> <p>The CMS User will be able to provide a header for the social links shown in the footer.</p>
3	Scripts	As a CMS User , I want to be able to disable the SSL redirect and provide custom scripts.	<p>The CMS User will be able to provide scripts which are output in the following location in the HTML of all pages:</p> <ul style="list-style-type: none"> • Top Head • Head • Body Open • Body Close
4	Social	As a CMS User , I want to be able to define how a page is shown when it is shared on a social platform and populate which social links are shown in the footer to a Website Visitor .	<p>The CMS User will be able to upload an image which will be the default OGP Image used in the appropriate metatag.</p> <p>The CMS User will be able to define a list of social links. They will be able to provide an icon, text and external link for each entry.</p>
5	SEO	As a CMS User , I want to be able define how the page title is shown in the Website Visitors browser and within search engine results.	<p>The CMS User will be able to provide a Website Name which can be used in the page name shown in the browser.</p> <p>The CMS User will be able to define how the title is constructed from the page name and the website name</p>

Common Page Compositions

The following field compositions will be found on all page document types found in Umbraco. They serve the purpose of providing basic functionality to enable a page to be controlled in areas such as SEO or navigation. These elements are provided in addition to the standard meta data stored for a page by Umbraco such as publish date or URL.

Page Name

This fields in this composition allow a page heading to be different from the page name which defines the URL.

Field	Type	Notes
Page Name	Text string	

Page Visibility

Fields contained in this composition are linked to the general visibility options of the page such as in the navigation or search.

Field	Type	Notes
Hide from Sitemap	Checkbox	Hides the site from the sitemap and XML sitemap
Hide From Navigation	Checkbox	Will hide the page from the site navigation
Hide From Search Results	Checkbox	Will hide the site from the internal site search
Hide Child Pages From Navigation	Checkbox	Turning this on will hide all the child pages from the navigation

SEO

Fields contained in this composition are linked to the general SEO of the page including local display options such as visibility in the site map.

Field	Type	Notes
Robots Meta Options	Checkbox List	Options are noindex and nofollow
Meta Title	Text string	
Meta Description	Text area	Soft limit of 155 characters
Cononical Link	Text string	
URL Alias	Text string	
Search Term Boosters	Repeatable Text string	Allows a page to be boosted in the search results for the given terms

Scripts

Fields in this composition are for allowing the insertion of custom scripts into the page.

Field	Type	Notes
Head Scripts	Text area	Allows scripts to be placed into the head of the page
Body Open Scripts	Text area	Allows scripts to be placed into the body open of the page
Body Close Scripts	Text area	Allows scripts to be placed into the body close of the page

Social Sharing

Fields in this composition allow a CMS user to control how the page is viewed when shared on various social media platforms.

Field	Type	Notes
OGP Title	Text string	Defaults to page name if not supplied
OGP Description	Text area	Defaults to page summary if not supplied

OGP Image	Single image picker	Defaults to the image set in site settings if not supplied
-----------	---------------------	--

Further to the compositions that are found on all pages the following compositions are will be applied to a sub set of the pages.

Banner

Fields in this composition allow a CMS user to define items to be shown at the top of the page in a banner inline with the designs.

Field	Type	Notes
Banner Items	Banner Items	Allows a CMS editor to set one or more banner items with the following fields. Image, Heading Subheading, Text, Link, Link Text
Disable Image Crops	Checkbox	When checked the default image crops will be removed and images will be displayed at their original dimensions

Summary

Fields in this composition allow a CMS user to define the summary of a page which can then be used on listings modules or other linked CTAs.

Field	Type	Notes
Summary Image	Single Image Picker	If populated this image will be used as the image for the page when shown on a listing
Summary Text	Text area	If populated this image will be used as the text for the page when shown on a listing

Common Page Compositions - User Stories

The following user stories will be used to verify the Common Page Fields are working correctly

	Module	User Story	Success Criteria
1	Heading	As a CMS User , I want the ability to provide a alternative heading for all pages to be shown to a Website Visitor .	The CMS User will be able to populate a heading field within the CMS for each page which is then visible on the page.
2	SEO	As a CMS User , I want the ability to set SEO meta data.	The CMS User will be able to populate the following fields which are then shown in the relevant tags in the HTML for the page. <ul style="list-style-type: none"> • Robots Meta • Meta Title • Meta Description • Canonical Link
3	Hide a web page	As a CMS User , I want the ability to restrict the visibility of a web page to Website Visitor .	The CMS User will have the ability to hide a webpage from: <ul style="list-style-type: none"> • Sitemap • XML Sitemap • Site search results • Main navigation
4	Page URL	As a CMS User , I want the ability to provide an alternative URL for a page.	The CMS User will have the ability to provide a URL in a text field which then allows the page to be accessed by the provided URL.
5	Social Sharing	As a CMS User , I want to define how a page is shown when shared on social platforms.	The CMS User will be able to populate the following fields which are then shown in the relevant tags in the HTML for the page. <ul style="list-style-type: none"> • OGP Title - defaults to page heading or name • OGP Description - defaults to page summary • OGP Image - defaults to image set in site settings

Rent Transactions

The rent transactions section of the website serves as an area for any credit or debit charges made to the tenancy account. Tenants can also filter by from and to date.

For each transaction on the rent transactions listing the page will show a date, description, debit or credit transaction and running balance. Listings will be paged with 10 transactions being shown on each page. Rent transactions shown on the page are arranged by payment date with the latest date shown first.

The following compositions will form part of the news listing document type:

- Heading
- Sidebar Content
- Summary
- Social
- SSL & Scripts
- SEO

In the Content section, the user can specify a payment link so it will show on the rent transactions page.

The rent transactions page will also contain a Block Editor field type allowing the following modules:

- Rich text editor

Payment Page

Property Selector

The portal user should have the ability to select the property/account they want to make a payment to

The property drop-down will pull all the accounts associated with the logged-in Portal User

- Clicking on the property drop-down will reveal the list of all the accounts
- Selecting an option from a drop-down will update the account summary section located below the drop-down

Payment Amount

There will be a payment amount field below the account summary details. This will allow the Portal user to pay a custom amount to their account.

- Clicking on the submit button will redirect the page to the payment provider, while it pre-populates the basic information such as the amount and reference number

Upon successful payment completion, Portal User will be redirected back to the portal and will be presented with a confirmation and thank you for message.


If the payment fails to process, Portal User will redirect back to the portal. Portal User will be presented with the confirmation message that the payment has not been processed e.g.: Oops, looks like something went wrong, please try again later.

Repairs

The repairs section of the website serves as an area to show all the historical repairs.

Repairs Listing

The repairs listing page is the main entry point to the repairs section. It lists all the repairs that have been reported by the tenant or housing team and allows a portal user to filter those repairs by the property, from date and to date.

 Repairs will only show if the tenant has properties associated with their tenancy account

The following compositions will form part of the repairs document type:

- Heading
- Side Content
- Summary
- SSL & Scripts
- SEO

Repairs Listing

For each repair listed on the repairs listings, the page will show a status, repair name, repair reference, date raised, the date scheduled, cancel and edit button. Listings will be paged with 10 items being shown on each page. Repairs shown on the page are arranged by date raised with the latest repair shown first.

If the repair has been cancelled, the cancel and edit buttons will be hidden.

Report a Repair

Call-to-action

The 'Repairs' template should display a call-to-action button to 'Report a Repair' which when clicked will take the user to the first stage of the report a repair process

Select Category and Repair Type

The portal user should have the ability to select the repair category and repair type

The repair schedule tool should display a list of available categories and repairs types to the Portal User

- Clicking on a category drop-down will reveal the list of repair types associated with it.
- Clicking on a repair type will allow the Portal user to proceed to the next section
- Portal user should be able to upload an image and enter a description of the repair
- Clicking on a confirm button will take the user to the next step (Schedule a Repair Appointment)

Schedule Repair Appointment

The portal user should have the ability to schedule a date and time for when their repair job

The repair schedule tool should display a list of available appointment days to the Portal User

- Clicking on an appointment day will display a list of available time slots
- Clicking on a listed time slot will display a confirmation button to the Portal User
 - Clicking on the confirmation button will schedule the appointment using the Civica APIs

A confirmation and thank you message will be displayed to the customer once they have scheduled their repair job appointment

Re-schedule Repair Appointment

'Repair job' appointments will appear in the list of historical repairs

The 'Reschedule' button will only appear on appointments where the scheduled date is more than 24 hours from the current date/time

Clicking the 'Reschedule' button will take the appointment selection screen

- Once a new appointment has been selected, the original appointment will be cancelled, and the new one will be scheduled in Civica via their APIs

Cancel Repair Appointment

'Repair' appointments will appear in the list of historical repairs

Current 'Repair Job' entries in the repairs list will feature a 'Cancel' button

The 'Cancel' button will only appear on 'Repair Job' appointments where the scheduled date is more than 24 hours from the current date /time

Clicking the 'Cancel' button will reveal a popup modal window asking the tenant to confirm they'd like the appointment cancelled.

- Clicking the 'Confirm Cancellation' button will cancel the appointment in Civica via their APIs

Schedule a Gas Service

Alert

The portal user should receive a clear alert notifying them to make a 'Gas Service' appointment, if they do not currently have one scheduled

The gas servicing due date should be retrieved via an API call to Civica - and if the date is within the next calendar month, the notification will be shown to the portal user

The CMS User should have the ability to content manage the alert notification text

The portal user should have the ability to dismiss the alert for that session - however it will appear again until they make a booking

The alert notification should have a clear CTA button to arrange an appointment, which when clicked will take them to the appointment scheduling page

Call-to-action

The 'My Repairs' template should display the 'Next Gas Servicing' due date

The 'My Repairs' template should display a call-to-action button to 'Schedule Gas Servicing' which when clicked will take the user to the appointment selection screen

Schedule Gas Servicing Appointment

The portal user should have the ability to schedule a date and time for when their gas servicing will occur

The gas servicing schedule tool should display a list of available appointment days to the Portal User

- Clicking on an appointment day will display a list of available time slots
- Clicking on a listed time slot will display a confirmation button to the Portal User
 - Clicking on the confirmation button will schedule the appointment using the Civica APIs

A confirmation and thank you message will be displayed to the customer once they have scheduled their gas servicing appointment

Re-schedule Gas Servicing Appointment

Gas Servicing' appointments will appear in the list of historical repairs

Current 'Gas Servicing' entries in the repairs list will feature a 'Reschedule' button

The 'Reschedule' button will only appear on appointments where the scheduled date is more than 24 hours from the current date/time

Clicking the 'Reschedule' button will take the appointment selection screen

- Once a new appointment has been selected, the original appointment will be cancelled, and the new one will be scheduled in Civica via their APIs

Cancel Gas Servicing Appointment

Gas Servicing' appointments will appear in the list of historical repairs

Current 'Gas Servicing' entries in the repairs list will feature a 'Cancel' button

The 'Cancel' button will only appear on 'Gas Servicing' appointments where the scheduled date is more than 24 hours from the current date /time

Clicking the 'Cancel' button will reveal a popup modal window asking the tenant to confirm they'd like the appointment cancelled.

- Clicking the 'Confirm Cancellation' button will cancel the appointment in Civica via their APIs

Message Centre

Messages List

A staff member should have access to the 'Message Centre' template

The 'Message Centre' template should contain a paginated list of messages which the portal user has submit via the message centre

The list of messages should show up to 10 messages per page, and offer pagination to navigate through the pages

The list of messages and their relevant contents will be retrieved from the Civica API

Each listed message should contain a:

- Subject line
 - Submission date
 - Status
 - Read more button which when clicked will expand and reveal the:
 - Original message submitted by the portal user
 - Any message updates posted by Housing Solutions via the Civica CRM

Reply Message

The portal user should have the ability to reply to already raised incident via Message Centre.

The reply to incident form fields should capture:

- Message
 - Submitting the reply will automatically update an incident in Civica CRM.

On a page reload, the incident should have all the historical messages attached to it.

Staff Login

The staff login allows staff members to login to the portal in order to access restricted pages. Members are created manually in Umbraco by a registered users.

Login (Staff)

The portal homepage will include a login component that will serve as the main entry point for staff members attempting to access restricted content on the portal. The url to this page will be publicly available however won't be visible to the website users. Page will allow the staff member to input their member credentials in the form of a username (email) and password.

The following compositions will form part of the homepage document type:

- Heading
- Summary
- Social
- SSL & Scripts
- SEO

In addition to the default staff login compositions the following fields will be available under the login tab:

Field	Field Type	Notes
Introduction	Rich Text Area	Serves at an introduction to the login functionality providing links to other areas such as password reset.

The login form shown on the page will contain the following fields:

Field	Field Type	Required	Notes
Username	Email	Yes	The members user name associated to their account
Password	Password	Yes	The members password, minimum of 8 characters long containing a number, lowercase letter and uppercase letter

Registration

This will be strictly handled by admin CMS users only. Only registered users will have access to the member section where they can create new staff members.

Forgotten Password

Request Password Reset

The forgotten password pages serves as the main entry point for a member who is unable to log into their account due to a forgotten password. The page is available to all websites users allowing them to fill out their details in order to generate an email that allows them to access the reset password page.

The page will validate if the members details are correct and if so generate a unique reset password link that is valid for one hour. Regardless of if the members details are found or not the same message will be shown on the website in order to reduce the chances of a malicious users gaining insight into the registered email addresses.

The following compositions will form part of the login document type:

- Heading
- Summary
- Social
- SSL & Scripts
- SEO

In addition to these composition fields the following fields will be available on the page:

Field	Field Type	Notes
Introduction	Rich Text Area	Serves at an introduction to the forgotten password functionality providing links to other areas such as login or registration
Password Reset Message	Rich Text Area	The message to show once the member has submitted the form

Email Content	Rich Text Area	The content of the email that is sent to the member containing the password reset instructions. Can use the magic strings of {{dateTime}} {{memberName}} & {{tokenLink}}
---------------	----------------	--

The forgotten password form will contain the following fields:

Field	Field Type	Required	Notes
Email Address	Email	Yes	The members email that should be linked to the members account

Confirm Password Reset

The forgotten password page allows members to reset a forgotten password as long as the link to the page is valid. The page will check for the presence of a token and email address on the query string, if they do not exist or can not be confirmed then a error message will be shown to the member.

If the members details are confirmed then they will be shown a form allowing them to provide a new password.

In addition to these composition fields the following fields will be available on the page:

Field	Field Type	Notes
Introduction	Rich Text Area	Serves at an introduction to the reset password functionality
Password Reset Message	Rich Text Area	The message to show when the password has been reset

The reset password form will contain the following fields:

Field	Field Type	Required	Notes
Password	Password	Yes	The members password, minimum of 8 characters long containing a number, lowercase letter and uppercase letter
Confirm Password	Password	Yes	Must match against the password provided

Staff Tenant Search

The Tenant Search will feature a search form that will allow a Staff Member to search for a tenant with a registered portal account by:

- First name
- Last name
- Tenancy reference number
- Email address
- Address
- Once a search has been made, any results will display as a list below the search form, and will feature the tenants:
 - Full name
 - Tenancy reference number
 - Open account button - which will take the staff member to the 'Report a Repair' screen
- If no search results are found, a text message will be displayed suggesting "Unfortunately there are no registered tenants with those details found."

Staff Tenant Impersonation

A staff member or scheme manager can impersonate the member and raise a repair on their behalf.

Raise a Repair

The staff member should have access to the 'Report a Repair' page

The page should display a list of any 'Current' repairs reported by the customer

The page should allow the staff member to select a tenancy to raise the repair against

The page should allow the staff member to edit following items for the reported repair:

- Customer contact name
- Customer contact number
- Customer contact email

The page should allow the staff member to select a 'Repair Category'

Clicking a 'Repair Category' should reveal a list of 'Repair Job' options that can be selected

The page should offer the staff member the option to add a 'Repair Description' which will be a long text field

The page should allow the staff member to report the repair, which will raise the repair on behalf of the registered tenant in Civica via their APIs

Schedule a Repair

The staff member should have the ability to schedule a date and time for when a reported repair will be attended

The repair schedule tool should display a list of available appointment days to the staff member

- Clicking on an appointment day will display a list of available time slots
- Clicking on a listed time slot will display a confirmation button to the staff member
- Clicking on the confirmation button will schedule the appointment on behalf of the registered tenant in Civica via their APIs

A confirmation and thank you message will be displayed to the staff member once they have scheduled the repair

App Chat

The app chat page serves as an app generic page able to hold a range of content. It is built around the block editor field type which allows a multitude of modules to be placed onto the page. As well as the block editor the document type also consists of the [Common Page Compositions](#)

The modules available in the block editor are documented in the Content Modules section of this specification.

The following compositions will form part of the standard content document type:

- Heading
- Summary
- Side Content
- SSL & Scripts
- SEO

App Form

The app form page serves as an app generic page able to hold a range of content. It is built around the block editor field type which allows a multitude of modules to be placed onto the page. As well as the block editor the document type also consists of the [Common Page Compositions](#)

The modules available in the block editor are documented in the Content Modules section of this specification.

The following compositions will form part of the standard content document type:

- Heading
- Summary
- Side Content
- SSL & Scripts
- SEO

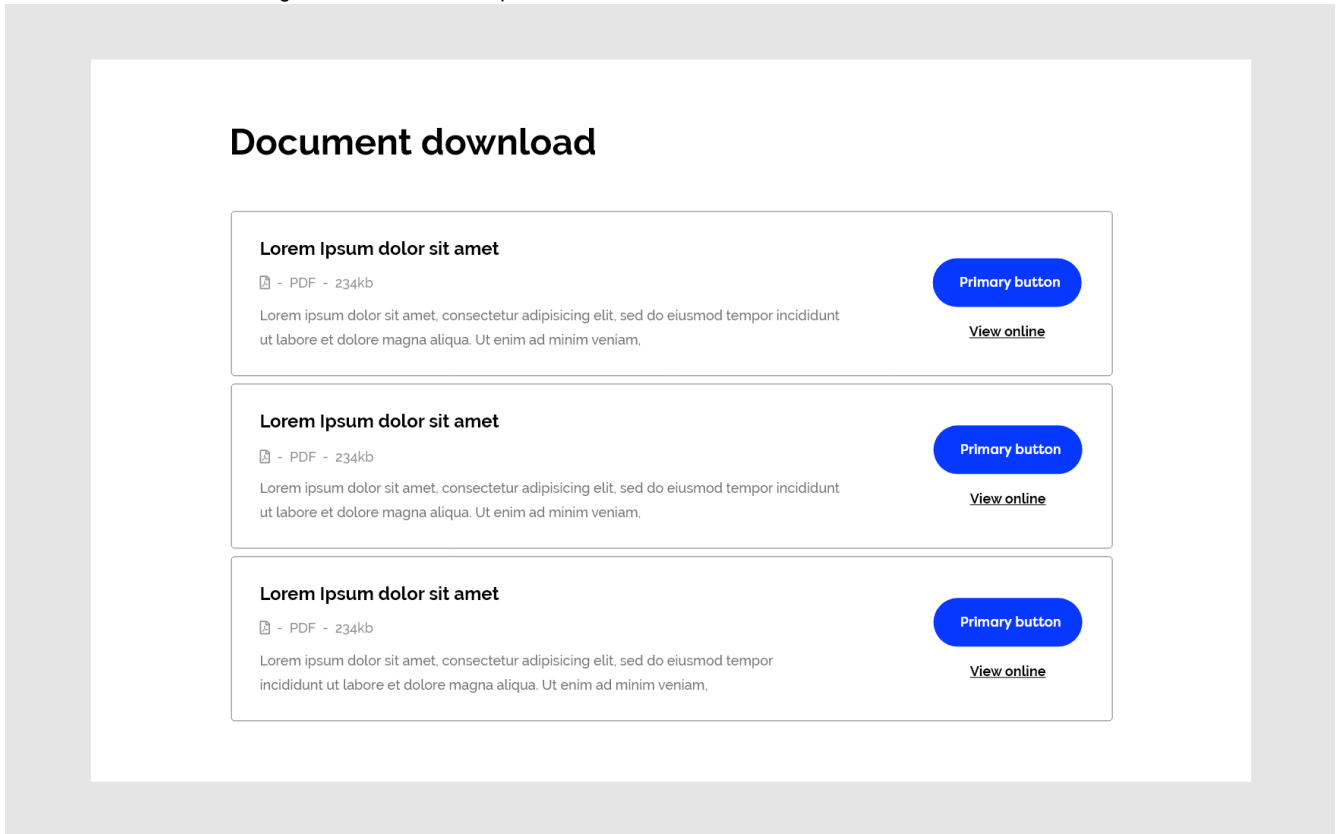
Content Modules

Content modules are the building blocks for content-focused pages on the website. Modules can be added to any page where a block editor is included. When published to the site, the look and feel of each module will be determined by the design. This section will specify the fields that make up each module.

- [Content Modules - User Stories](#)
- [Document Listings](#)
- [Form](#)
- [Grid/Icon Layout](#)
- [Image Gallery](#)
- [Listing](#)
- [Rich text editor](#)
- [Video](#)

Document Listings

A document listing allows a CMS User to show documents to the Website Visitor. A website visitor will be able to see the name type and size of the file as well as being able to download or open it.



Folder Picker

This variant will display the child items of a selected media folder

Field	Type	Required	Notes
Folder to List	Media Folder Picker	Yes	Allows any folder in the media section to be selected

Picker

This variant allows an editor to explicitly pick which document to list

Field	Type	Required	Notes
Document To List	Media Picker (Multiple)	Yes	Allows a editor to pick files from the media section

Model Settings

The following setting will be available on the module:

Field	Type	Required	Notes
Top Module Spacing	Numeric	No	Sets the top spacing on the module
Bottom Module Spacing	Numeric	No	Sets the bottom spacing on the module

Form


Used to collect information from a website visitor using a number of different predetermined fields.

Form field title

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Short text
Lorem ipsum dolor sit amet

Short text error
Lorem ipsum dolor sit amet

 Please enter a valid email address

Long text
Lorem ipsum dolor sit amet

Dropdown

- Select -

Document upload

[Choose file](#) | No file chosen

Checkbox list

Lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet


Lorem ipsum dolor sit amet

Radio

Lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet

I'm not a robot  [Privacy - Terms](#)

Allows a CMS User to insert a Umbraco Form into the page allowing the Website Visitor to provide feedback or complete information which is then captured.

Field	Type	Required	Notes
Form	Form Picker	Yes	Allows a CMS User to pick a form they have access to from the Umbraco Forms section

Form Content	Basic Rich Text Editor	No	Optional content to show with the form
--------------	------------------------	----	--

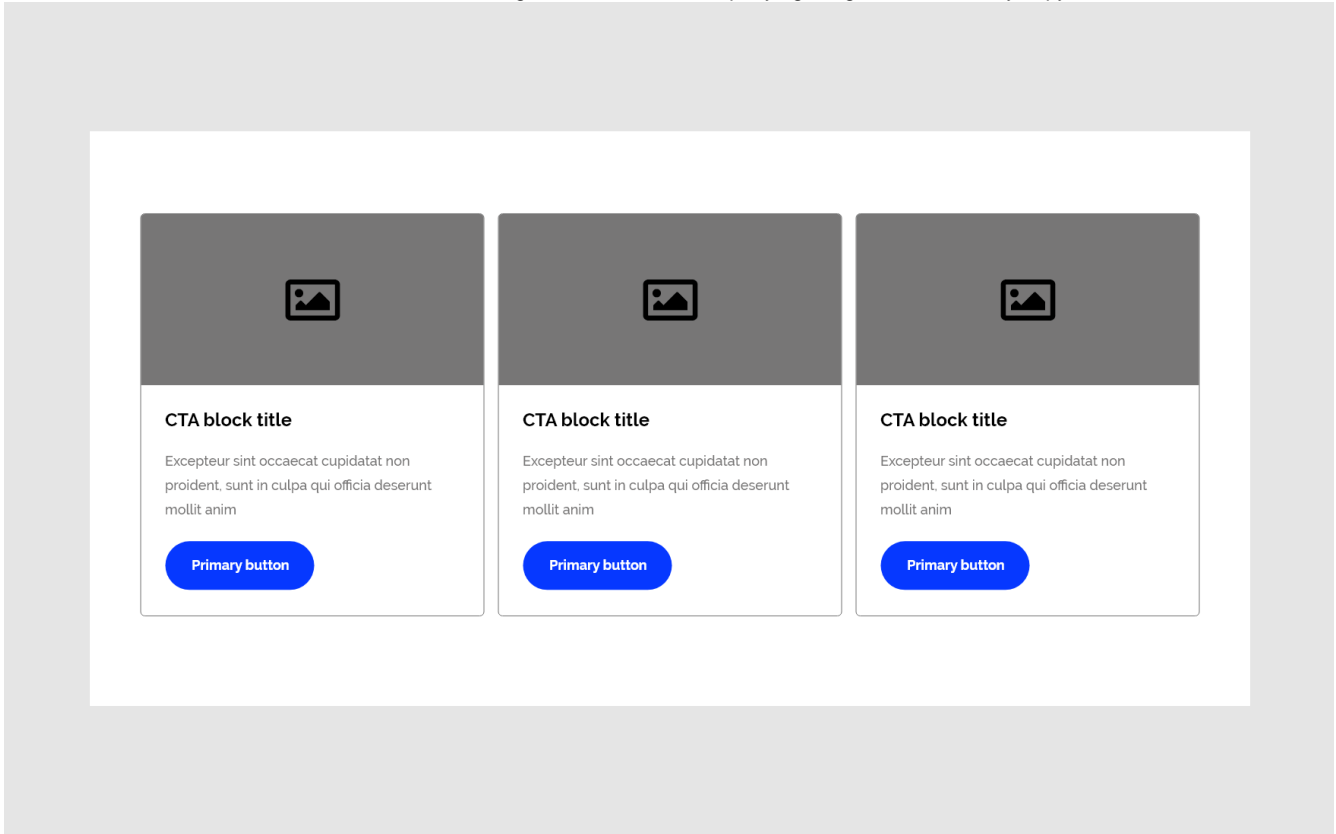
Model Settings

The following settings will be available on the module:

Field	Type	Required	Notes
Top Module Spacing	Numeric	No	Sets the top spacing on the module
Bottom Module Spacing	Numeric	No	Sets the bottom spacing on the module

Grid/Icon Layout

This module is used to showcase a series of links in a grid format with accompanying images, title and body copy.



Allows a CMS User to build a module containing CTA Blocks which consist of an Image, Title, Summary, and Link and Link Text. These are shown to the Website Visitor in accordance with the design.

There are three different approaches available to populating the grid module. Each are described below.

Child Pages

This variant will display a list of child pages to a website visitor that are published below a chosen parent page. The child pages Name, Summary, and Summary image will be displayed in the listing along with a link to the page. The following options will be available to the CMS User.

Field	Type	Required	Notes
Parent Page	Link picker	Yes	Allows a CMS user to set which page should act as the parent page for the listing.

Picker

This variant will display a list of pages to a website visitor which have been chosen by the CMS User. The selected pages Name, Summary, and Summary image will be displayed in the listing along with a link to the page. The following options will be available to the CMS User.

Field	Type	Required	Notes
Pages to Display	Multiple Content Picker	Yes	Allows a CMS user to set which pages should be shown in the listing

Manual

This variant allows a CMS editor to build up the Grid without them being based on Umbraco pages. The CMS User will be able to manually enter details for each listing. The following options will be available to the CMS User.

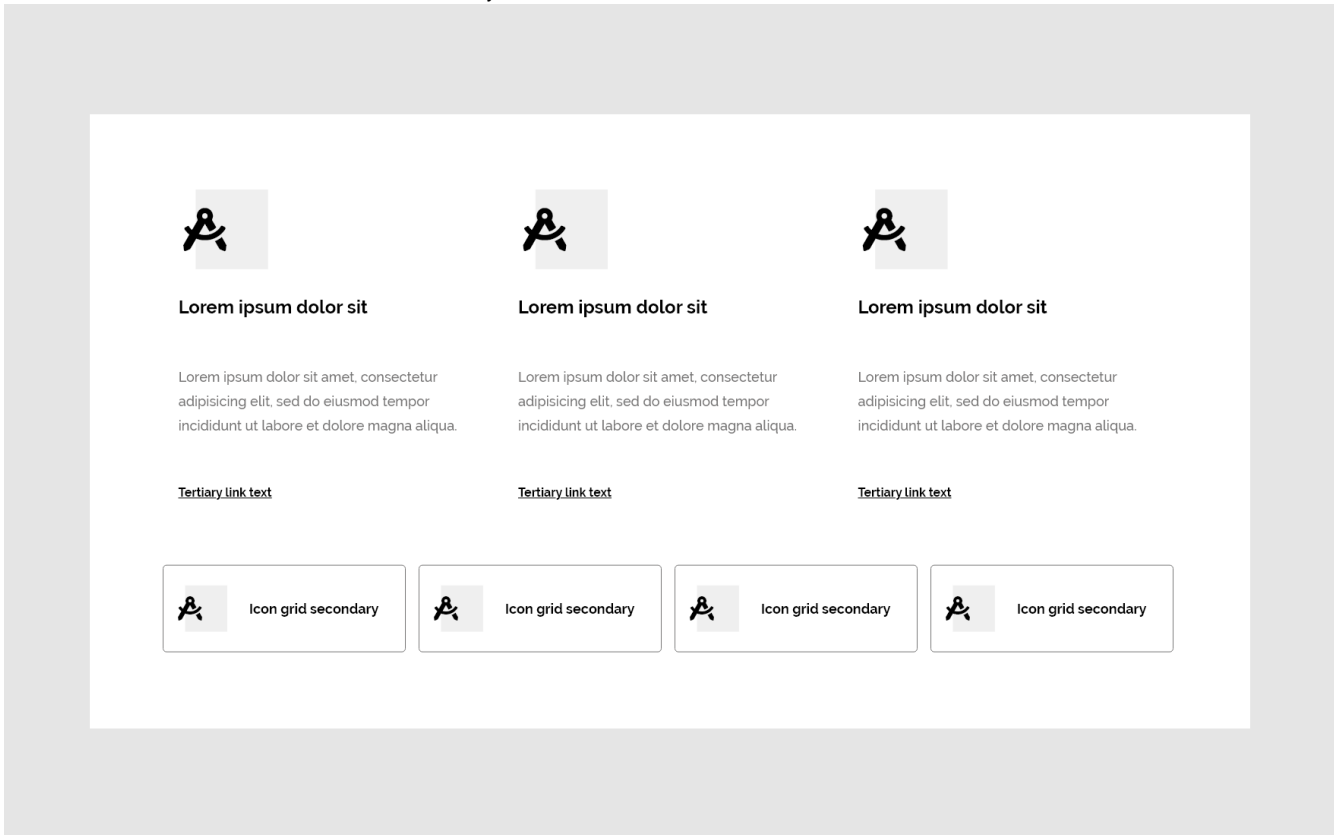
Field	Type	Required	Notes
Grid Items	List Item Content	Yes	Allows a CMS User to define multiple Grid items which consist of a Heading, Image, Copy, Link and Link Text

Icon

This module is used for displaying features, benefits or services where imagery might not suit the application. Row 1 consists of primary features, including They consist of an icon, heading, summary, and button. Row 2 consists of secondary features, including icon, heading and link.

NOTE

It is recommended that overall text length between grid items is similar to prevent spacing issues. The module utilizes Font Awesome which is a library of over 2000 icons.



The user has the option to add primary and secondary CTA's to emphasize the priority of some calls to action. The primary CTA's appear before the secondary items.

Primary icon grid items

Field	Type	Required	Notes
Icon	Icon picker	Yes	A list of font awesome icons are available to choose from
Heading	Text string	No	A heading for the icon grid item
Summary	Text string	No	A summary for the icon grid item
Link URL	Link	No	A link for the icon grid item to link to
Link text	Text string	No	A string of text to be used as a click prompt. This may be hidden in favour of design and a hover effect

Secondary icon grid items

Field	Type	Required	Notes
Icon	Icon picker	Yes	A list of font awesome icons are available to choose from
Heading	Text string	No	A heading for the icon grid item
Link URL	Link	No	A link for the icon grid item to link to

Module Settings

The following setting will be available on the module:

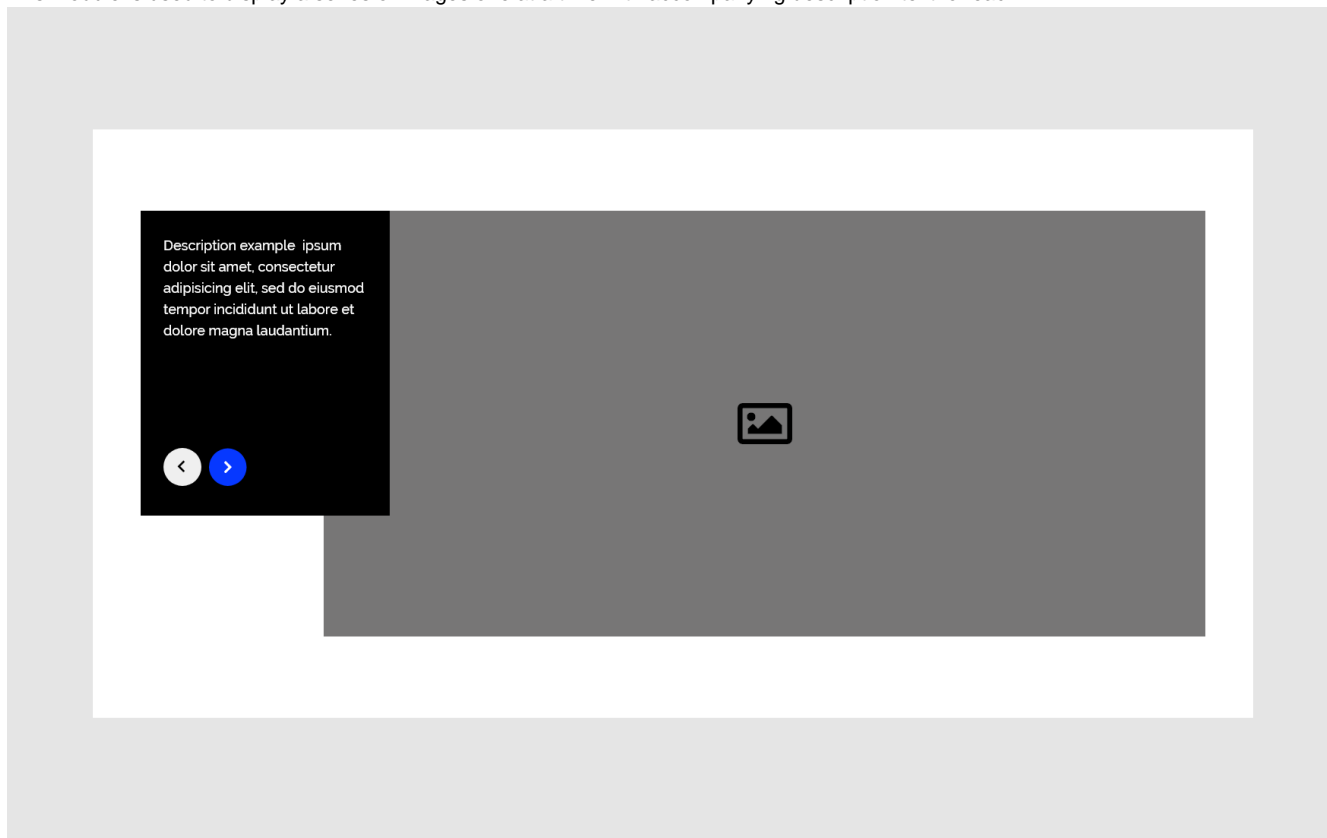
Paginate	Checkbox	No	Allows the CMS User to define if the CTA items should be paginated.
Items to Take	Number	No	Defines the number of items to show if the CMS user has set the grid to be paginated.
Summary Text Character Limit	Numeric	No	The number of characters to be shown in the summary text field before add an ellipsis
Disable image crops?	Checkbox	No	When checked the default image crops will be removed and images will be displayed at their original dimensions
Top Module Spacing	Numeric	No	Sets the top spacing on the module
Bottom Module Spacing	Numeric	No	Sets the bottom spacing on the module

Image Gallery

Allows a CMS User to select multiple images to be shown on the page to a Website Visitor. Can be used as a Thumbnail/Lightbox gallery allowing a website visitor to interact with the module.

Standard Mode

This module is used to display a series of images one at a time with accompanying description text for each.



Thumbnail Mode

This module is used to display a series of images all at the same time, which can also be viewed in full resolution in a lightbox.



Both modes have the following fields:

Field	Type	Required	Notes
Images	Multiple Image Picker	Yes	Allows a CMS User to select multiple images form the media section

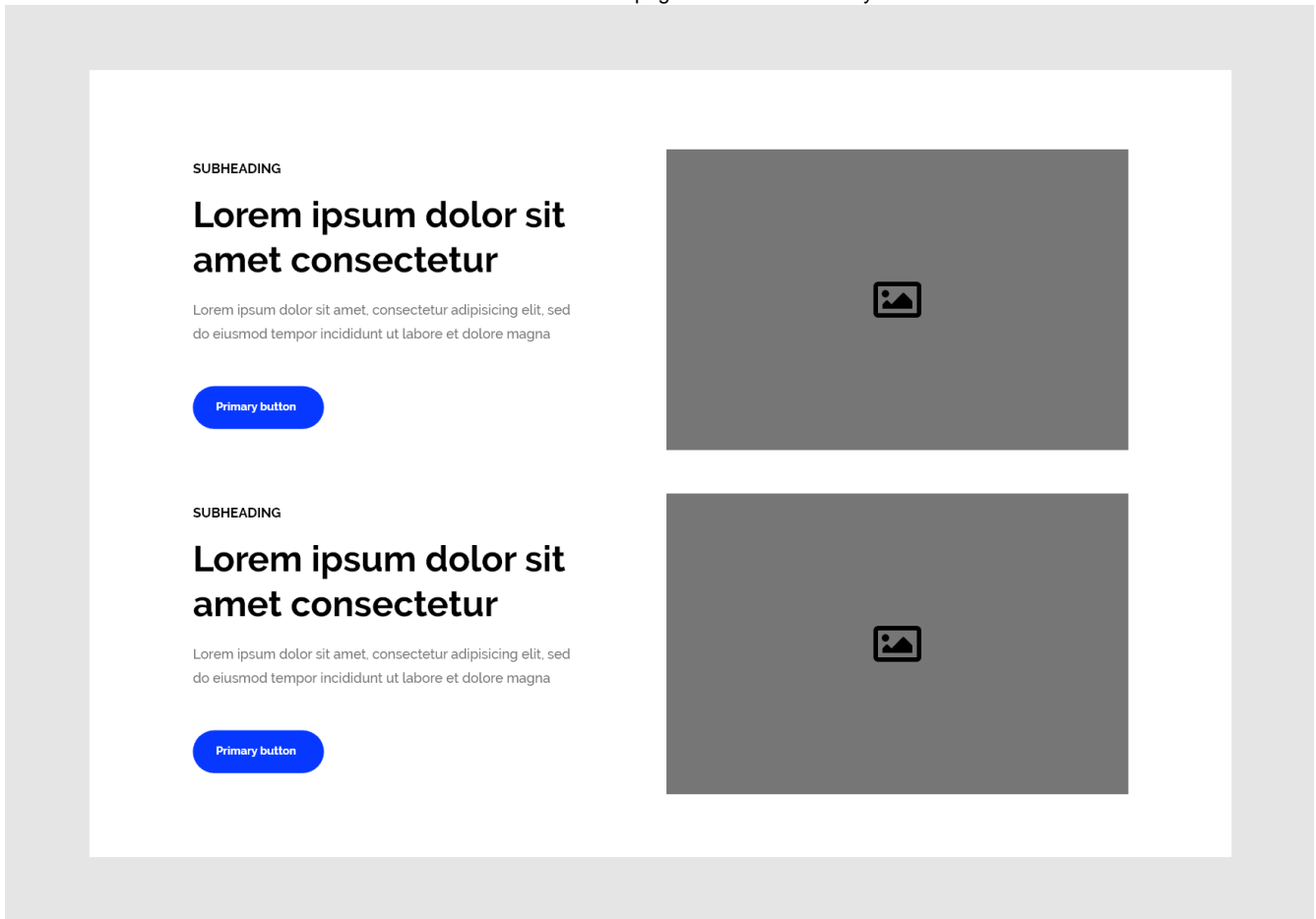
Model Settings

The following settings will be available on the module:

Field	Type	Required	Notes
Show as Thumbnails	Checkbox	No	Renders the module as a lightbox gallery
Disable image crops?	Checkbox	No	When checked the default image crops will be removed and images will be displayed at their original dimensions
Top Module Spacing	Numeric	No	Sets the top spacing on the module
Bottom Module Spacing	Numeric	No	Sets the bottom spacing on the module

Listing

A module that allows a CMS User to build out a module that can list pages in a structured way.



Three variations of this module will be available: A module that allows a CMS User to build out a module that can list pages in a structured way. Three variations of this module will be available:

Child Pages

This variant will display a list of child pages to a website visitor that are published below a chosen parent page. The child pages Name, Summary, and Summary image will be displayed in the listing along with a link to the page. The following options will be available to the CMS User.

Field	Type	Required	Notes
Parent Page	Single Content Picker	Yes	Allows a CMS user to set which page should act as the parent page for the listing.

Picker

This variant will display a list of pages to a website visitor which have been chosen by the CMS User. The selected pages Name, Summary, and Summary image will be displayed in the listing along with a link to the page. The following options will be available to the CMS User.

Field	Type	Required	Notes
Pages to Display	Multiple Content Picker	Yes	Allows a CMS user to set which pages should be shown in the listing

Manual

This variant allows a CMS editor to build up the listing without them being based on Umbraco pages. The CMS User will be able to manually enter details for each listing. The following options will be available to the CMS User.

Field	Type	Required	Notes
List Items	List Item Content	Yes	Allows a CMS User to define multiple list items which consist of the fields shown below

List items will consist of the following fields:

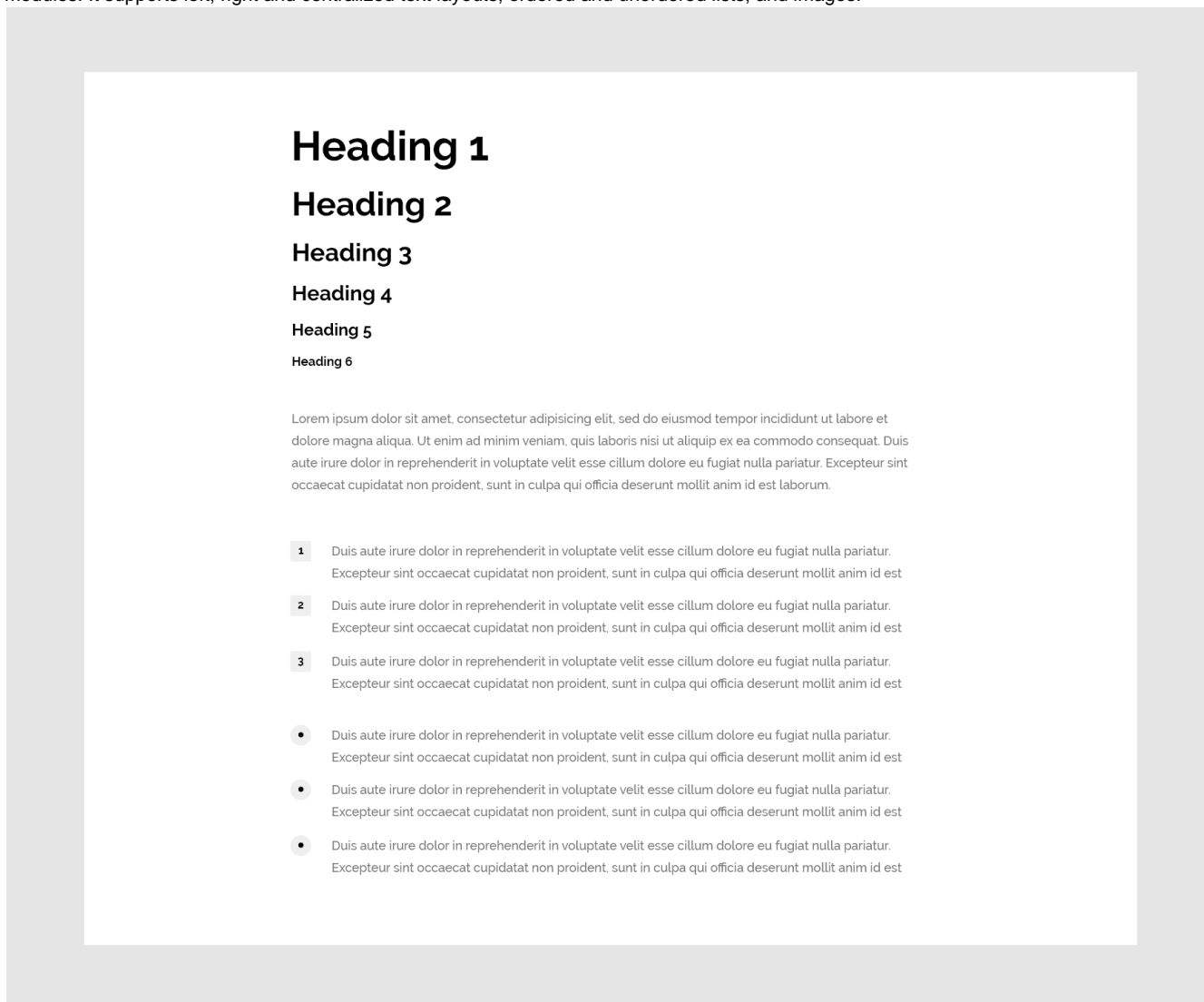
Field	Type	Required	Notes
Heading	Textstring	Yes	The heading of the list item
Subheading	Textstring	No	The subheading of the list item
Image	Single Image Picker	No	The image of the list item
Copy	Simple Rich Text Editor	No	The copy to show on the list item
Link URL	Link	No	The CTA link to show on the list item
Link Text	Textstring	No	The text to show on the CTA link

Model Settings

The following settings will be available on the module:

Field	Type	Required	Notes
Paginate	Checkbox	No	Allows the CMS User to define if the CTA items should be paginated.
Items to Take	Number	No	Defines the number of items to show if the CMS user has set the grid to be paginated.
Summary Text Character Limit	Number	No	The number of characters to be shown in the summary text field before add an ellipsis
Alternate Image and Text	Checkbox	No	Swaps the order of which the image and text gets rendered out
Disable image crops?	Checkbox	No	When checked the default image crops will be removed and images will be displayed at their original dimensions
Top Module Spacing	Numeric	No	Sets the top spacing on the module
Bottom Module Spacing	Numeric	No	Sets the bottom spacing on the module

This module is used for long form text sections such as news articles or case studies, but can also be used in conjunction with other modules. It supports left, right and centralized text layouts, ordered and unordered lists, and images.



The CMS User will be able to format text in the following ways:

- Bold
- Italic
- Align Left
- Align Center
- Align Right
- Bullet List + variations
- Number List + variations
- Indented
- Linked

In addition to the basic formatting the CMS user will also be able to mark text as styled in the following ways in line with the design:

- Heading 1
- Heading 2
- Heading 3
- Heading 4
- Heading 5
- Heading 5
- Heading 6
- Primary Button Style
- Secondary Button Style

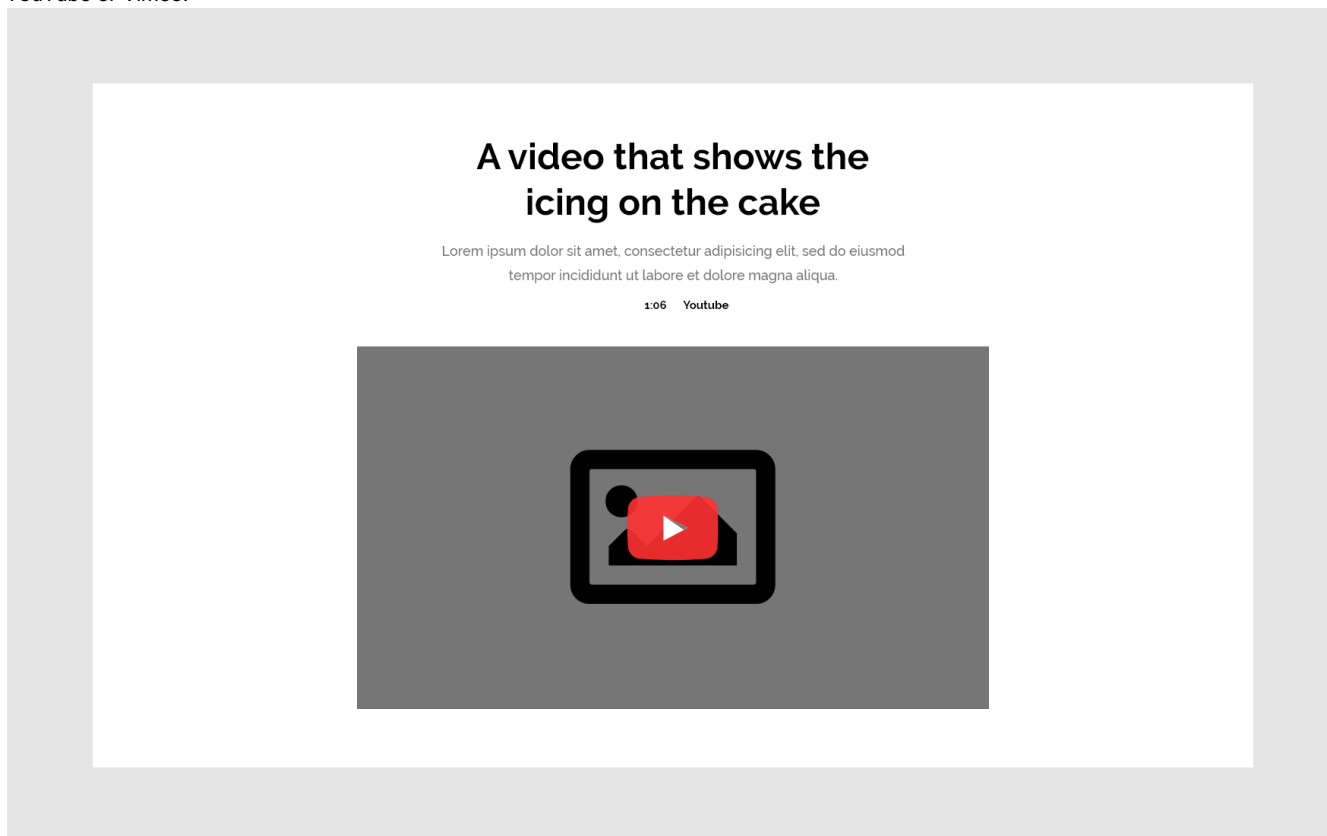
Model Settings

The following settings will be available on the module:

Field	Type	Required	Notes
Top Module Spacing	Numeric	No	Sets the top spacing on the module
Bottom Module Spacing	Numeric	No	Sets the bottom spacing on the module

Video

Allows a CMS user to embed an externally hosted video into the web page so the website visitor can view it. Videos can be embedded from YouTube or Vimeo.



The following options are available within the same module.

Embed mode

The user can choose to have the video embedded within the content and play when activated within the flow of content.

Lightbox mode

The user can choose to have the video open in a lightbox, with an image of the video used as a large background within the content of the page

Field	Type	Required	Notes
Video heading	Text string	Yes	Appears above the embedded video
Video summary	Text string	No	Appears near the heading
Video length	Text string	No	A field to inform the user the duration of the video. Appears near the summary
Video provider	Text string	No	A field to inform the user of where the video is hosted, Appears near the summary
Video image	Image picker	No	An image that appears as a background to the module when the lightbox option is ticked
URL	Text string	Yes	The URL of the video to embed. Once provided the CMS User will be able to retrieve the video which will then be previewed below the text string field.
Lightbox Mode	Checkbox	Yes	An option that changes the layout and behaviour of the module to open a video in the lightbox

Model Settings

The following settings will be available on the module:

Field	Type	Required	Notes
-------	------	----------	-------

Disable image crops?	Checkbox	No	When checked the default image crops will be removed and images will be displayed at their original dimensions
Top Module Spacing	Numeric	No	Sets the top spacing on the module

Content Modules - User Stories

	Module Name	User Story	Success Criteria
1	Document Listing	As a CMS User , I want to add documents to a webpage so that Website Visitors have a clear call-to-action to download a file/document	<p>The CMS user can upload a specific document/file or select a folder to add to a page If a folder is selected the document/files will automatically create a list based on the media items in it. The CMS user can add multiple files/documents</p> <p>The document list can include a: Title Description Download button view online button File size: e.g. 368kb</p>
2	Form	As a Website Visitor , I want to fill in an online form so that the organisation can help me with the request	<p>The website visitor can complete the required form that may include the following fields:</p> <p>Short text Long text</p> <p>Date Picker Dropdown options Checkbox list Radio button list File upload function</p> <p>The user can provide information based on different conditions If Recapture is required the user can submit the answer before submitting the form content (Google Recaptcha should be set up and the keys sent to Prodo) On submission the user is presented with a confirmation message Mandatory fields prompt the users to complete the information before submission</p>
3	Gallery - Large	As a CMS User , I want to add an image gallery to a page so that I can feature or promote key images to Website Visitors	<p>The image gallery module will allow the CMS User to: Select one or more images from the Umbraco media section to be displayed in a slider format</p>
4	Grid Module	As a CMS User , I want a clear method to promote key web pages within content, so that I can guide Website Visitors to the correct section of the website	<p>The grid module can be used as a basic call-to-action, or as a method to list out multiple pages to a visitor. Each grid item will feature an:</p> <p>Image thumbnail Title Description (optional) Button link (link text will be editable)</p> <p>The CMS USER will have the ability to either:</p> <p>Select a parent page from which the child pages will be displayed as grid items. With this option the grid item content will be automatically pulled through from the child page's summary content. This grid module option can be paginated This grid module option allows the CMS user to select individual pages to be listed as grid items</p> <p>OR</p> <p>Manually create individual grid items which can be used as call-to-actions. With this option, each individual grid item's content will be manually populated/controlled by the CMS USER. Again this does not need to include summary text, which is optional e.g. this would be used for call-to-actions which link through to individual pages</p>

5	List	As a CMS User , I want to create a list of web pages so that I can promote the content to Website Visitors	<p>The CMS User can create a list in three ways:</p> <ul style="list-style-type: none"> Manually Using a parent page Selecting individual web pages <p>Using the manual option the CMS User can create a list adding the following for each item:</p> <ul style="list-style-type: none"> Heading/Title Sub Heading (Manual List only) Image Content editor / box Link button and text which links through to the specific content. Text for button will be editable. Using the parent page option will list all of the associated child pages Using the picker option the CMS User can select pages multiple to appear in the list The CMS User can add pagination (optional) and the number of listings to display per page can be specified
6	Rich Text Editor	As a CMS User , I want to feature rich text content within a webpage, so that information can be presented to a Website Visitor	<p>The rich-text-area (RTE) module will allow the CMS USER to add the following text-based content styling:</p> <ul style="list-style-type: none"> Paragraph text Paragraph styles including bold, italic and underlined Heading styles including H1, H2, H3, H4, H5 Numbered list style and variations of numbered lists Bullet point list style and variations of bulleted lists
7	Video	As a CMS User , I would like to provide video on a webpage so that I can serve relevant content to Website Visitors	<p>The CMS USER can add the video module to standard content pages.</p> <p>The video module will allow the URL for a video hosted on YouTube or Vimeo to be inserted</p>
8	Icon Grid	As a CMS User , I want to promote a list of services or features using Font Awesome to Website Visitors	<p>The CMS USER can add the icon grid to standard content pages.</p> <p>The icon grid module will allow CMS users to set a main Heading, Summary, and Button to signpost users to a landing page of their choosing.</p> <p>The icon grid module will allow two types of list, the primary list can contain a icon, heading, summary and link. The secondary list can contain a icon, heading and link.</p>

Site Template Components

Site templates define the elements that will be shown to a website visitor and portal user when they visit the site. Templates can include common elements, such as navigation and header logos, side content, as well as elements that are specific to the document type.

This section will define what is contained in each common element and how they behave. The designs will define the appearance of each element.

Sidebar Content

The sidebar content component is used to highlight important information to the portal user. It also includes a link to the rent transactions page with the current balance.

The sidebar content will contain the following fields:

Field	Type	Required	Notes
Title	Text string	No	This appears above or below the main heading. This will be determined in the design.
Content	Text string	No	Content that appears below the title.
Hide Right Sidebar	Text string	No	A flag indicating if the sidebar is hidden for the current page.

Sidebar Content - User Stories

User Story:

User Story	Success Criteria
As a CMS User , I want to be able to manage the sidebar content section portal homepage	The CMS user will be able to view and manage the sidebar content section in Umbraco. The Sidebar content tab will include the following properties: <ul style="list-style-type: none">• Title• Content• Hide Right Sidebar
As a Portal User , I want an easy way to be notified if there is anything important happening at Client Name	Content will be visible on the sidebar if populated in Umbraco. Content can consist of the following elements: <ul style="list-style-type: none">• Text• Button
As a Portal User , I want to be able to see my balance when I log in	Current balance will be visible on the sidebar. The icon next to the balance value can either be: <ul style="list-style-type: none">• Red (in arrears)• Green (in credit)
As a Portal User , I want to be able to click on the link that will take me to the rent transactions page	There will be a clickable link on the sidebar. Clicking the link will take a portal user to the rent transactions page.

Pagination

This section of the website is used for navigation between multiple pages of content. Primarily, this is used to reduce page load times and chunk long lists of content on the following pages:

- Repairs
- Rent Transactions
- Staff Tenant Search
- Staff Report a Repair
- Staff Rent Transactions

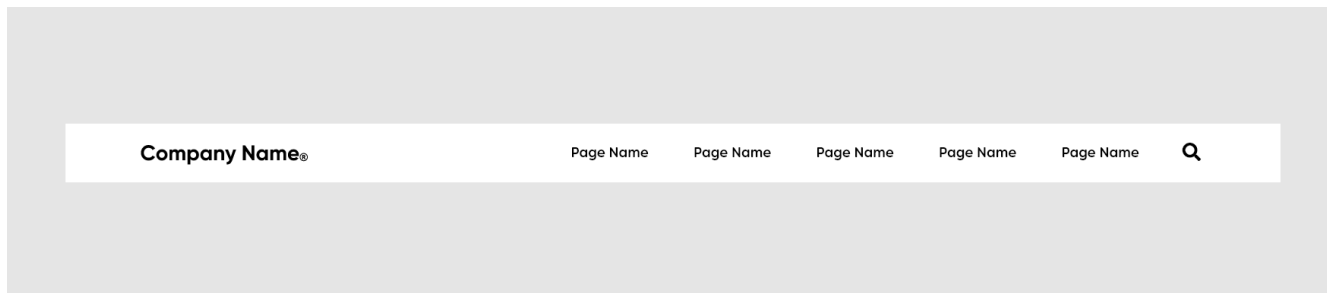
Pagination - User Stories

User Story	Success Criteria
As a Portal User , I am exploring the rent transactions page and would like to load the next page of results.	Clicking on any of the numbers rendered in the pagination will reload the page and the listing will be updated to display the page of results selected by the user.

Site Header

This section of the website sits at the top of the viewport and houses your logo, navigation, navigation dropdown and any additional functionality outlined in this document.

Example:



This allows users to quickly navigate throughout your website. It behaves responsively to accommodate all browser sizes, whether on a desktop, tablet or mobile device.

Site Header - User Stories

User Story	Success Criteria
<p>As a Website Visitor, I want to see a header on every page that contains controls so I can easily access and navigate the website</p>	<p>The header will be consistently at the top of every web page</p> <p>The header must contain the following elements:</p> <ul style="list-style-type: none">• Logo• Page name• Dropdown (with email label) <p>Clicking the dropdown will reveal a list of options:</p> <ul style="list-style-type: none">• Link to edit my profile page• View start guide• Logout <p>Clicking on the logo will redirect a Website Visitor to the homepage</p>

Site Navigation - Top Level Only

Depending on the client requirements you should select only one Site Navigation page to include in the client specification

This section of the website is located within the left sidebar and mirrors content structure of your CMS. Each top level page located underneath the homepage is listed here.

Site Navigation - Top Level Only - User Stories

User Story:

User Story	Success Criteria
As a Portal User , I can click on a link in the main navigation so that I can easily browse between different areas of the website	The main navigation will be visible on the left sidebar of every web page Clicking on an item in the navigation will open the linked page in the same browser tab

Site Navigation - Dropdown

Depending on the client requirements you should select only one Site Navigation page to include in the client specification

This section of the website is located within the left sidebar and mirrors the content structure of your CMS. Each top level page located underneath the homepage is listed here.

When a user clicks on the '+' plus icon next to the menu item the child pages of the item will be shown in a dropdown. On mobile the user will be able to access the menu by tapping the burger menu icon in the bottom left corner.

Site Navigation - Dropdown - User Stories

User Story	Success Criteria
As a Portal User , I can expand the main navigation item and click on a child pages link so that I can easily browse between different areas of the website	The second level of navigation will be visible on the sidebar content of every web page. Clicking on an item in the navigation will open the linked page in the same browser tab.

Site Footer - User Stories

1	Footer	As a Website User , I want to see a footer on every page containing links which show me core business information so that I have consistent access to this information across the site.	<ul style="list-style-type: none"> • The footer will be consistently at the bottom of every web page • The footer can contain links to relevant social media accounts / sites • The footer can contain CMS user specified links
2	Footer Navigation	As a Website User , I can click on a link in the footer navigation so that I can easily browse between different areas of the website	<ul style="list-style-type: none"> • The footer navigation will be visible in the footer of every web page and will be content managed via Umbraco • Clicking on an item in the navigation will redirect the user to the associated web page • If the navigation contains an external link then it will open in a new browser tab • Internal links will open in the same browser tab
3	Footer Social Links	As a Website User , I can click on a social link in the footer so I can see the associated social feeds	<ul style="list-style-type: none"> • The footer social links will be visible in the footer of every web page and will be content managed via Umbraco • Clicking on an item will redirect the user to the associated social channel • Each link will show an icon linked to the social channel that is managed in Umbraco

Third-Party Integration

The Housing Solutions portal currently uses third-party integrations. There won't be any additions or changes to the existing integration within this scope. Prodo will utilise the existing endpoints and the business logic from the previous build.

Integrations included in the scope:

- Capita Open Housing
- Capita Payments
- Castleton Documotive
- Civica Servitor
- Microsoft Cognitive Services
- NAudio

App Engine API

Supported integrations:

- Umbraco Web API (Housing Solutions)
- Capita Open Housing
- Civica Servitor

The Prodo App Engine API will wrap around the above API's and expose access to all the required calls to the mobile device. We will establish a machine to machine connection between the app and the Prodo App Engine, this will be achieved through the use of a bearer token and will happen automatically when a user logs into the app.

The above APIs will then surface tenant, payment and repairs data where required, this will be explored in more detail on a screen by screen basis in section 4 of this document, screens & Features.

The Umbraco members API will also be utilised by the app, this will handle all of the authentication features that the app will require specifically the following processes:

- Logging into the app
- Registry and sign up if they're not already a portal user
- Password reset

Umbraco Forms

The creation of and display of forms to a website visitor is a key requirement of the website. Forms allow a user to submit data via the website in a easy and structured way.

The form functionality on the site will be provided by [Umbraco Forms](#), a paid for extension to Umbraco which is created and maintained by the Umbraco core team. It allows a CMS user to create a form based on a set number of standard field types which can then be embedded into content pages. A CMS user can then defines what happens to the data when a user submits it, this can include sending the submission to an email or storing it within the Umbraco database for retrieval at a later date.

This specification will highlight the key features of Umbraco Forms and how a CMS user will be able to use it within the context of the website build. The full list of features available in Umbraco Forms will be in line with the details provided by Umbraco in relation to the product.

Form Creation

CMS users who have been granted access to the Umbraco Forms section will be able to create and manage forms which can be placed onto the website.

A form will be made up of pages and fields, with a page containing a number of fields. The following field types (answer types) will be available for a CMS user to place onto a form page.

Field Type	Description
Short answer	Shown as a text string field allowing a small amount of plain text to be input by the website visitor.
Long Answer	Shown as a text area field allowing a larger amount of plain text to be input by the user.
Date	Shown as a date picker in line with designs or standard browser UI. Allows a website visitor to select a date from a calendar.
Checkbox	Shown as a single checkbox in line with the designs allowing a website visitor to check or uncheck the option related to the provided label
File upload	Shown as a file upload in line with the designs. Allows a website visitor to upload a file as part of the form submission.
Password	Shown as a password field where the text input is masked. Allows a website visitor to input a text string related to a password
Multiple Choice	Shown as a checkbox list allowing a website visitor to select one or more options that have been defined by the CMS user.
Data Consent	Shown as a checkbox with supporting text. Allows a website visitor to consent to having their data stored and processed in line with GDPR requirements.
Dropdown	Shown as a standard drop down inline with designs. Allows a website visitor to select one option from a list defined by the CMS user.
Single Choice	Shown as a radio button list inline with designs. Allows a website visitor to select one option from a list defined by the CMS user.
Title and description	Allows a CMS user to define a message that will be shown on the form in order to provide the website visitor with supporting information about the form or section of form. The appearance of the message will be defined by the design.
Hidden	A hidden field which will not be shown to the website visitor. Allows a a CMS user to gather information that has not been provided direct by the user such as from a session or query string value.
Recapture	A Google Recapture challenge field which can be placed on a form to reduce spam submissions. Appears inline with the Google design and requires an API key to be set up and provided to Prodo (TBC2) .

Each field type allows the field to be marked as required and for field types that support it additional validation rules related to formatting of the answer can be provided. These allow a CMS user to dictate that a text field should be provided in the form of an email address for example.

Custom field types can be developed to meet certain form requirements. The website scope currently does not require this.

Form Workflows

Form workflows determine how the submitted data is managed both at the submission and approval event. By default Umbraco Form's provides the following workflow types.

Workflow Type	Description
Change Record State	Allows a CMS user to set up rules based on what data has been submitted in order to delete or approve the submission. For example the workflow could be configured to delete submissions that contain a known spam phrase.
Post as XML	Allows a CMS user with developer input to send the submitted data in a XML structure to an API endpoint for further processing.
Save as XML file	Allows a CMS user with developer input to save the submitted data in a XML structure to a given location on the hosting storage.
Save as Umbraco content node	Allows a CMS user to create an Umbraco content node based on the submitted data. Each field can be mapped to a field contained on the target data type. This is used in cases where you would want to make the submission public such as a forum post.
Send email	Allows a CMS user to send the submission via a plain text email to a given email address.
Send email with template (Razor)	Allows a CMS user to send a styled email to a given email address. The email is styled based a template which needs to be produced by a developer.
Send form to URL	Allows a CMS user with input from a developer to send the submitted data to a URL endpoint. Mapping can be set up between the field on the form and the fields required by the endpoint.
Slack	Allows a CMS user to surface the data submitted into Slack.

Custom workflows can also be developed to integrate with other systems. The website scope currently does not require this.

Umbraco Forms - User Stories

The following user stories will be used to verify Umbraco Forms are working correctly.

Module	User Story	Success Criteria
Answer Types	As a CMS User , I want to be able to add questions to a form I am building.	<p>The CMS User will be able to add the following answer types:</p> <ul style="list-style-type: none"> • Short answer • Long answer • Date • Checkbox • Field upload • Password • Multiple choice • Data Consent • Dropdown • Single Choice • Title and Description • Hidden • Recapture <p>The fields will be shown to the website visitor inline with the designs.</p>
Form Conditions	As a CMS User , I want to be able to show and hide questions based on a previous answer.	The CMS user will be able to set conditions in line with the default Umbraco Forms functionality .
Form Workflows	As a CMS user , I want to be able to control what happens to data once a user submits a form.	<p>The CMS user will be able to assign the following workflows to a form:</p> <ul style="list-style-type: none"> • Change Record State • Post as XML • Save as XML file • Save as Umbraco content node • Send email • Send email with template (Razor) • Send form to URL • Slack
Display Forms	As a CMS user , I want to be able to embed an Umbraco Form into any web page that contains the form module.	The CMS user will be able to use the grid editor to add a form to a content page in order for the website visitor to see and interact with the form.

OneTrust Cookie Tool

A notification will pop up on the website for the user to accept or reject cookie preferences through the use of the third-party tool provided by OneTrust. Features of OneTrust include:

- Scan websites for first and third party tracking technologies including cookies, tags, pixels, web beacons, and more
- Automatically categorise the trackers discovered with the Cookiepedia database of over 20,000 pre-categorized cookies and trackers
- Automatically block trackers until consent is given with zero code required using OneTrust Auto-blocking
- Maintain a centralised audit trail to demonstrate compliance and dashboarding to monitor opt-in rates

OneTrust's script will be added to the website to enable the Cookie Control tool functionality across all pages. Preferences, scans, content and categorisation of cookies for the tool will be managed from within the OneTrust admin area by Housing Solutions.

Previously Prodo implemented the OneTrust cookie tool on the Housing Solutions website meaning the Prodo user developers@prodo.com is still active with the Housing Solutions account. With this, there are no actions for Housing Solutions prior to the implementation of OneTrust on the new version of the website.

Prodo will assist in the running of the first scan and subsequent categorisation of any cookies that were not automatically identified within OneTrust's Cookiepedia database.

Testing

We are constantly reviewing our browser support based on worldwide usage trends. Below are the details of browsers that we fully test along with browsers that are tested functionally.

All of the browsers below will be tested however we will not test or develop to browsers with compatibility mode enabled.

Complete Layout and Functional Testing

These are browsers that we test that the site will function as desired and match the appearance of designs in their entirety.

Web

- Microsoft Edge
- Chrome - Latest version at time of development
- Firefox - Latest version at time of development
- Safari - Latest version at time of development

Mobile (within HubSpot supported devices/software updates)

- Safari on iPhone and iPad
- Android Browser on Tablet and Mobile
- Android Browser on Android

Accessibility Testing

The site will be developed and tested to the WCAG 2 AA standard. These standards are used as a guideline and may not pass on small areas of the website with the clients permission. Anything that may not pass accessibility will be raised during the design phase.

Primary content sections of the website, for example Rich Text Areas, will always meet AA standards unless a specific requirement has been made.

Not in Scope

Browser Support, Integrations, Content and Testing

- We support all major browsers. However, support for functional requirements in older browsers and those not listed in the Testing section is not within the scope of this project. The site may have display differences in other browsers.
- 3rd Party Integrations outside of what has been defined within this specification document.
- Meeting any specific accessibility standards e.g. A, AA, or AAA standards, unless specifically defined in this document.
- The completion of any load or penetration testing
- Any fixes based on the outcome of penetration testing unless deemed to be a high-risk issue
- Domain Name Management.
- Content importation or migration beyond that defined in the specification.
- Any work that is marked as Out of Scope within or outside of this document.

Sign Off

By signing this approval document, I hereby confirm that the specification has been produced to my requirements and I authorise Prodo Digital to advance to development according to this specification.

Housing Solutions Sign Off

Contact Name: Richard Harvey

Signed:

Date:

Prodo Sign Off

Head of Client Services: Jenny Bradshaw

Signed:

Date: