



Learning Paths - Level 01

Site Building

Session Breakdown

05 min	01	Introduction + Case Study Recap
05 min	02	Creating Sites + Template Options
05 min	03	Understanding the Different Types of Pages
15 min	04	Creating Pages + Template Options
10 min	05	Building Menus and Navigation
10 min	06	Getting Ready for Design + Client Extensions
10 min	07	What's Next? + Q&A



Part 01

Introduction and Session Scope



What do we mean by *Site Building* and why is it important?

Introduction to Site Building

- Term used both generically and broadly to describe “web sites”
- Focuses on enabling non-technical to manage create web sites and web based applications without deployment
- Deals with the structure and framework that acts as the foundation for web based solutions
- Leverages important concepts like Information Architecture and User Experience to provide best-in-class results
- Critical to be done right to ensure a solid foundation for the evolution of the solution used to support the business



Sites

Case study

Clarity Vision Solutions

- Global leader in optical technology
- Over 500 employees and 300+ million in revenues
- Traditional retail and online commerce
- Modern, stylish and comfortable products
- International Distribution via Partners

Elevate Your Brand with Premium Eyewear

Discover premium craftsmanship, innovative lenses, and designs that help your brand stand out.

[Start Your Journey](#)



See Further with Clarity Vision Solutions

Take advantage of exclusive offers on the latest products to deliver exceptional value for your business and your customers.

[Learn More](#)



Case study

Clarity Vision Solutions Challenges

- Overnight success has led to growing pains
- Complex and fragmented infrastructure
- Expensive maintenance and licensing costs
- Increased operational costs by requiring costly resources (experts and consultants)
- Highly technical tools limit pool of contributors and increase the time to market required for new initiatives

?



Clarity's Critical Success Factors

- ✓ 1. Non-technical participation
- ✓ 2. Simplified maintenance (and evolution)
- ✓ 3. Future proofed and flexible
4. Increase engagement and user experience through personalization
5. Risk mitigation with better governance
- ✓ 6. Lower investment / total cost of ownership



Part 02

Creating Sites + Template Options



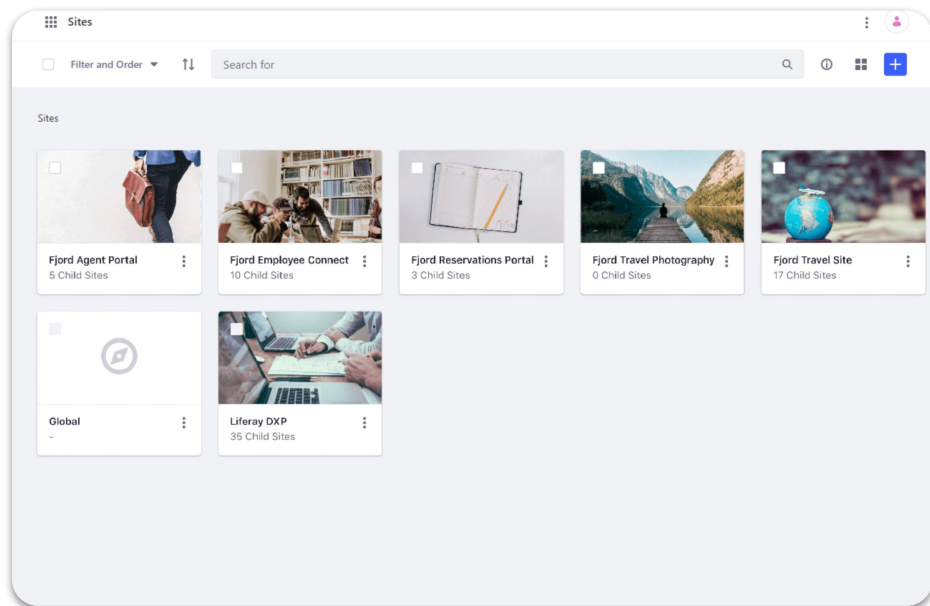
Clarity Business Requirements

- Reduce development time and effort by offering pre-built templates with common website structures and functionalities
- Minimize design costs compared to custom development by leveraging pre-designed templates
- Define common layout elements like headers, footers, navigation menus, and sidebars in a master page, ensuring consistency across all website pages
- Simplify maintenance by managing common elements in a single location, reducing the need to update code across multiple pages
- Utilize templates designed with accessibility best practices, ensuring your website is usable by everyone
- Maintain a consistent brand identity and user experience across the website



Create multiple Sites in a single DXP instance

- Users can create and manage sites and their respective audiences.
- Each site gets its own set of pages, content, permissions, etc.
- Site has its own look and feel via Style Books and Master Pages.
- If required, content can be shared across sites, although Asset Libraries are recommended for this purpose.
- Site Templates allow predefined sets of pages to be used as a starting point for new sites.
- See also Virtual Instances which provide additional data separation.





**Site
Templates**

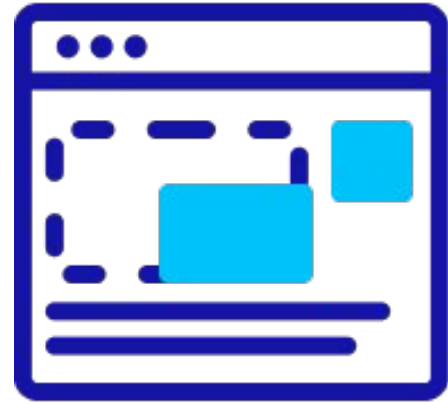
VS



**Site
Initializers**

Site Templates

- Site templates provide pre-built layouts and functionalities
- Ensure that all pages on your website adhere to your brand guidelines, creating a consistent and professional user experience
- By using pre-designed templates, you can avoid the need to pay for custom development work
- You don't need to rely on developers to make changes to your website
- Often designed with accessibility best practices in mind, ensuring that your website is usable by everyone
- Changes to the template itself, which will automatically update all of the pages that use that template



Site Initializers

- Define different Site Initializers for various website types or sections
- Reduce the time and effort required to create new sites by automatically generating core content
- Integrate with CI/CD pipelines for automated site deployments, streamlining your development and release process
- Manage entire sites with a single configuration file, simplifies content updates and maintenance compared to individual element management
- Store sensitive content, like API keys or database connection details, securely within the initializer code, improving security and access control
- Leverage Git version control for your Site Initializers, enabling tracking changes and collaborative development





**Site
Templates**

VS



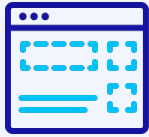
**Site
Initializers**

Which one should we use?

Part 03

Understanding Page Types





**Widget
Pages**

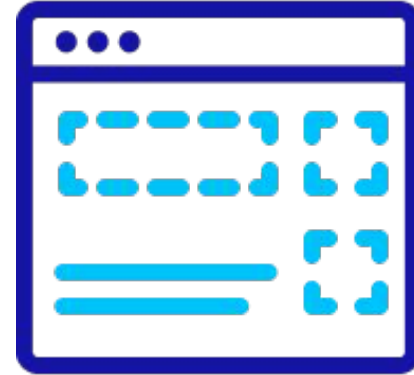
VS



**Content
Pages**

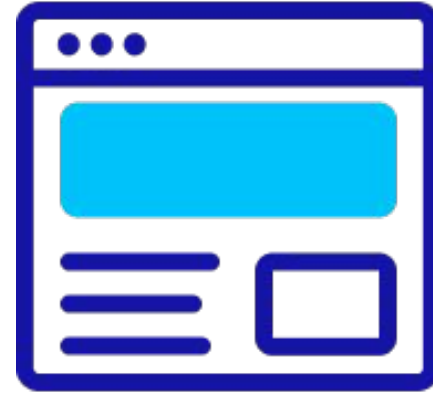
Widget Pages

- Best used for when primary focus is on functionality rather than content presentation
- Ideal for building dashboards, landing pages, or simple layouts with frequently changing content
- Good fit for solutions that require integrating multiple functionalities or services
- You need unique functionalities not available in existing Liferay features or widgets
- Allow you to easily add, remove, and rearrange widgets on the fly, providing high flexibility for creating dynamic and interactive content layouts
- Allow for sections to be marked as “editable by users” allowing users to creating personalized layouts



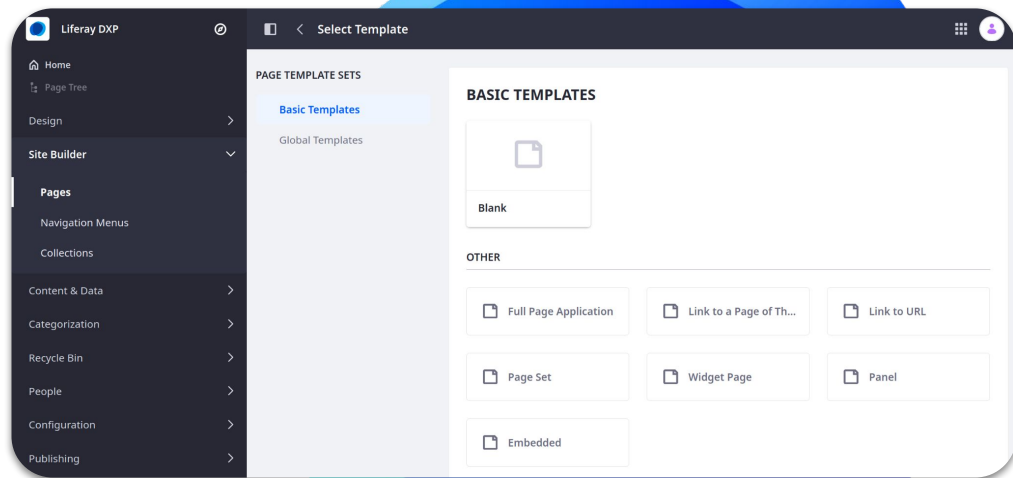
Content Pages

- Used to create content-rich, visually engaging pages
- Rich in-context visual editorial process that allows for page composition through the use of fragments and widgets
- Ideal for landing pages, marketing campaigns and other scenarios where presentation and engagement are key
- Create personalized page experiences and manage localized translations to engage your target audiences
- Optimize user experience with the responsive layout editor
- Finally, with A/B Testing, you can create alternative versions of a content page to track visitor data (e.g., bounce rate, clicks, etc.)



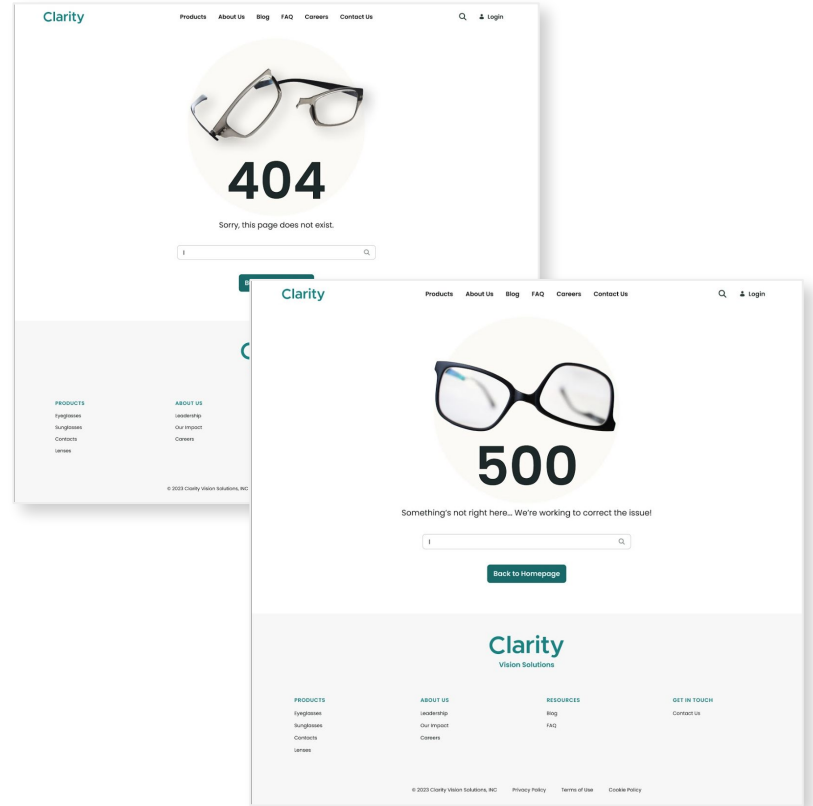
Additional Page Types

- Collection Pages
- Full Page Application
- Link to a Page of This Site
- Link to URL
- Page Set
- Panel
- Embedded



Utility Pages

- Utility pages serve specific operational tasks or actions
- They are essential for users to interact with a website or application effectively
- Support for 400 and 500 error pages
- Provides out of the box pre-built pages that can be used or modified
- Allows for administrators to define their own custom error pages



Part 04

Creating Pages + Template Options

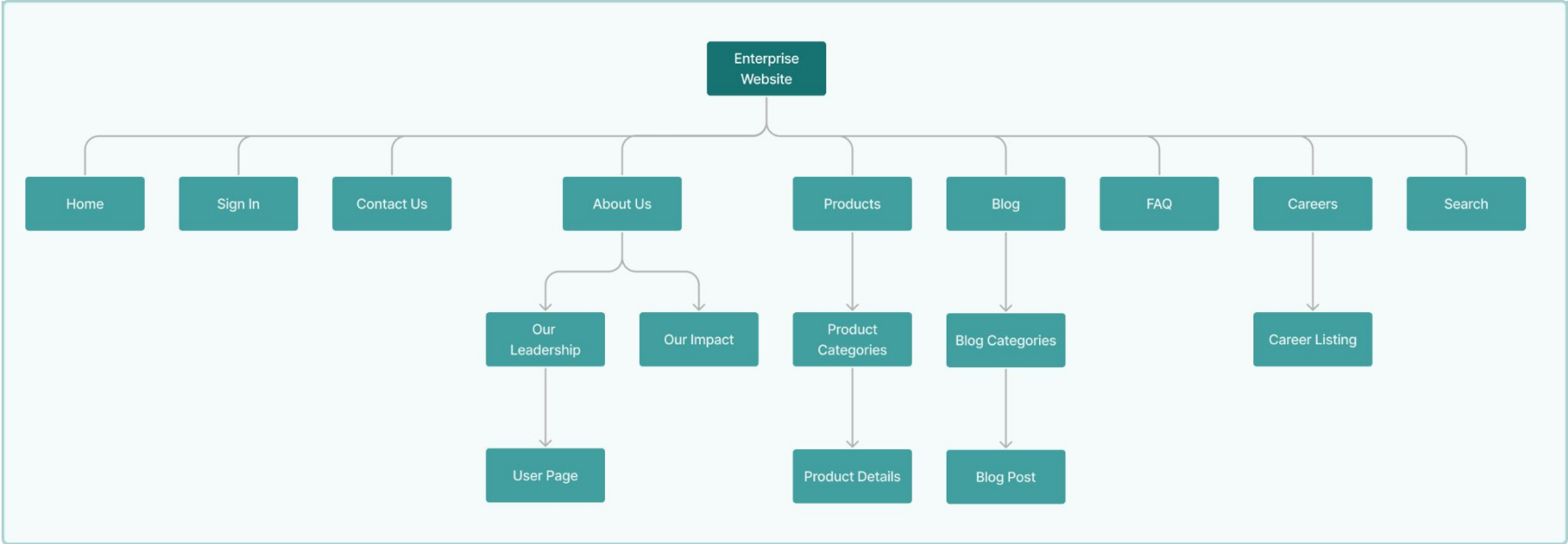


Clarity Business Requirements

- Ensure a consistent brand identity and user experience across different website sections
- Reduce development time and effort for creating new website pages or sections
- Simplify page creation and management for non-technical users
- Minimize design and development costs associated with website creation
- Maintain a consistent website structure and design while accommodating future growth and content expansion
- Utilize common layout elements and functionalities across different sections of the website



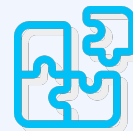
Clarity Solution Site Map





**Page
Templates**

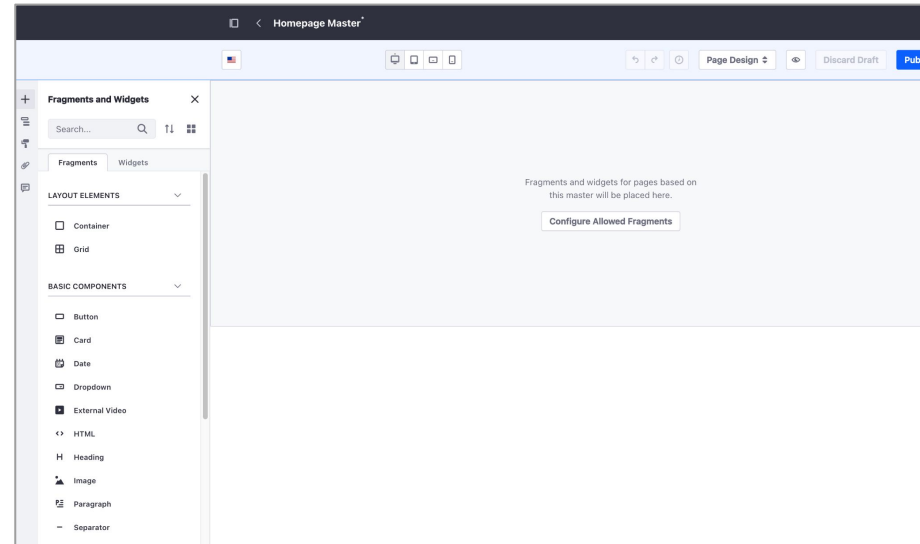
VS



**Master
Pages**

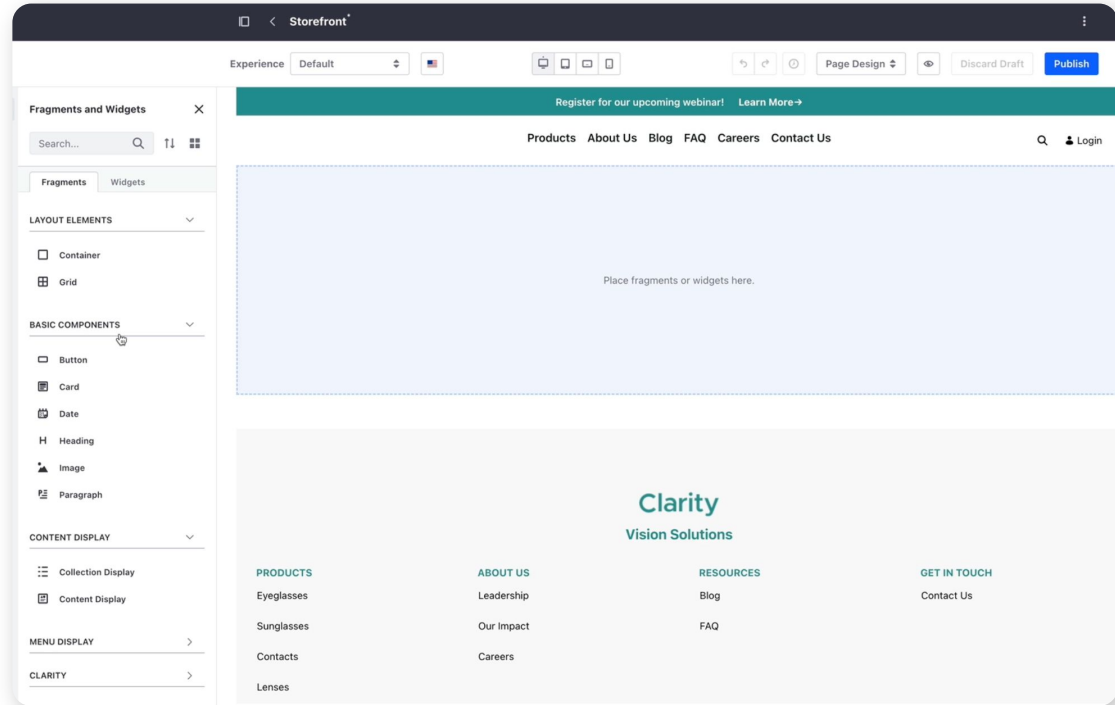
Master Pages

- Gives you control over the entire “canvas” that represents the page
- Leverages the same (familiar) editing tools and functionality as working with *Content Pages*
- Provides the means to lock down certain parts of the page by restricting which areas can be edited
- Provides the ability to control which fragments (design elements) are available for use
- Can be used as a standalone baseline when creating a new page or as part of a (Content Page) page template



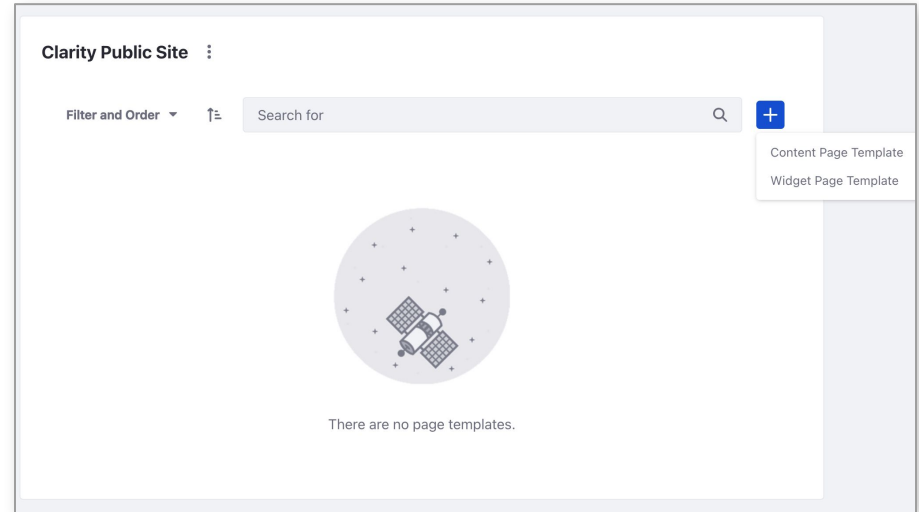
Master Pages in Action

1. Go to Page Templates
2. Create a new Master Page
3. Add three containers
4. Move the Editable area into the middle container
5. Add the notification and menu to the top container
6. Configure the primary navigation
7. Add the footer to the bottom container
8. Configure the footer menus
9. Configure the “allowable fragments”
10. Publish the Master
11. Create a new page from master and notice the layout and restrictions in the fragment palette



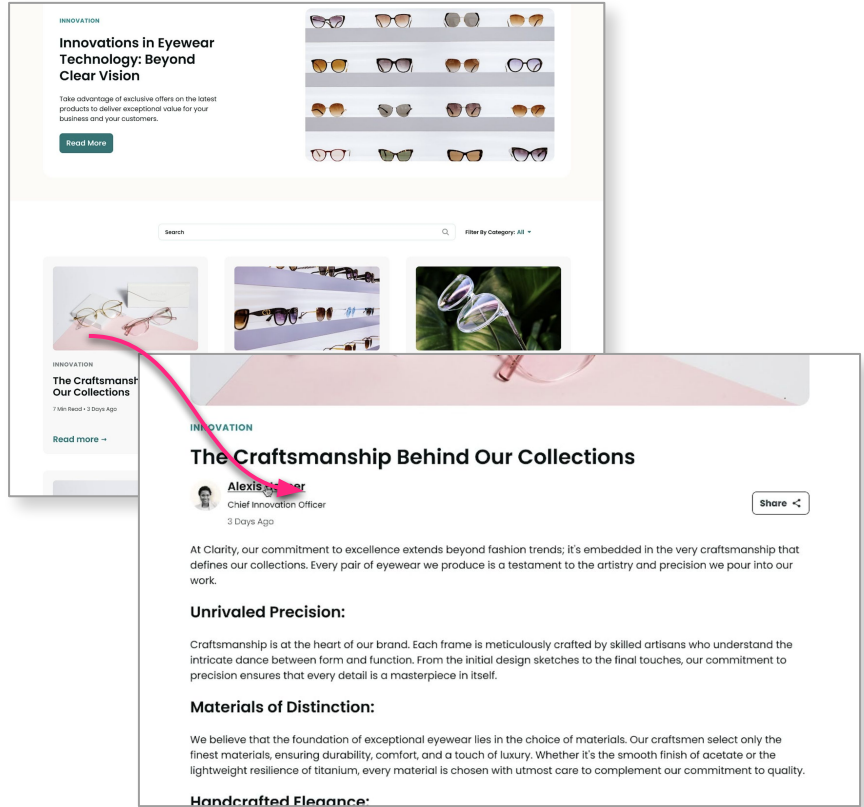
Page Templates

- Come in two flavours
 - Widget Page Templates
 - Content Page Templates
- Ideal use cases
 - Consistent website sections with similar layouts and functionalities
 - Landing pages with a specific design and call to action
 - Articles requiring a structured format
- Significantly reduce development time and costs
- Lower the bar for entry – less Liferay knowledge required to add new pages to site
- Provide boilerplate starters but can be modified
- Allow for propagation of changes



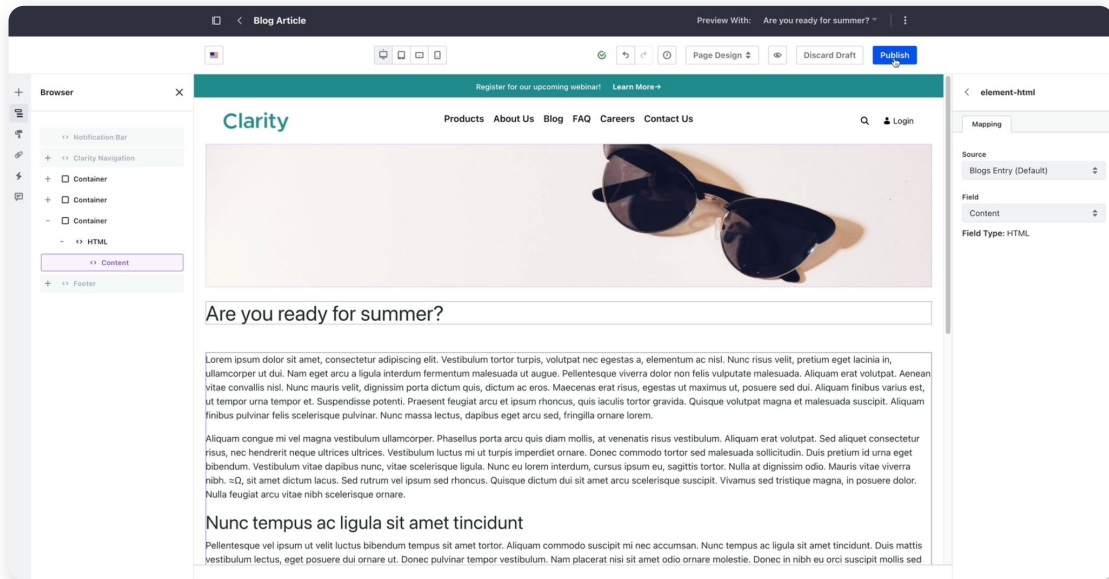
Display Page Templates

- Predefined layouts and functionalities for Liferay pages
- **Faster:** Reduce development time by using pre-built layouts and functionalities
- **Reduced Costs:** Minimize design and development expenses
- **Content Management Efficiency:** Non-technical users can easily manage content within sections of the template
- **Scalability and Reusability:** Create new pages that inherit the same design and structure
- **Modular Design:** Utilize common layout elements and functionalities across different sections of the website
- Streamline content management and website maintenance



Display Pages in Action

1. Go to Page Templates
2. Go to the Display Templates tab
3. Create a new template for *Blog Entry*
4. Set the entry to visualize results
5. Add the image fragment and map to cover image
6. Add the heading fragment and map to title
7. Add the HTML fragment and map to content
8. Publish template
9. Go back to Blog Listing
10. Click through to new template



Part 05

Building Menus and Navigation



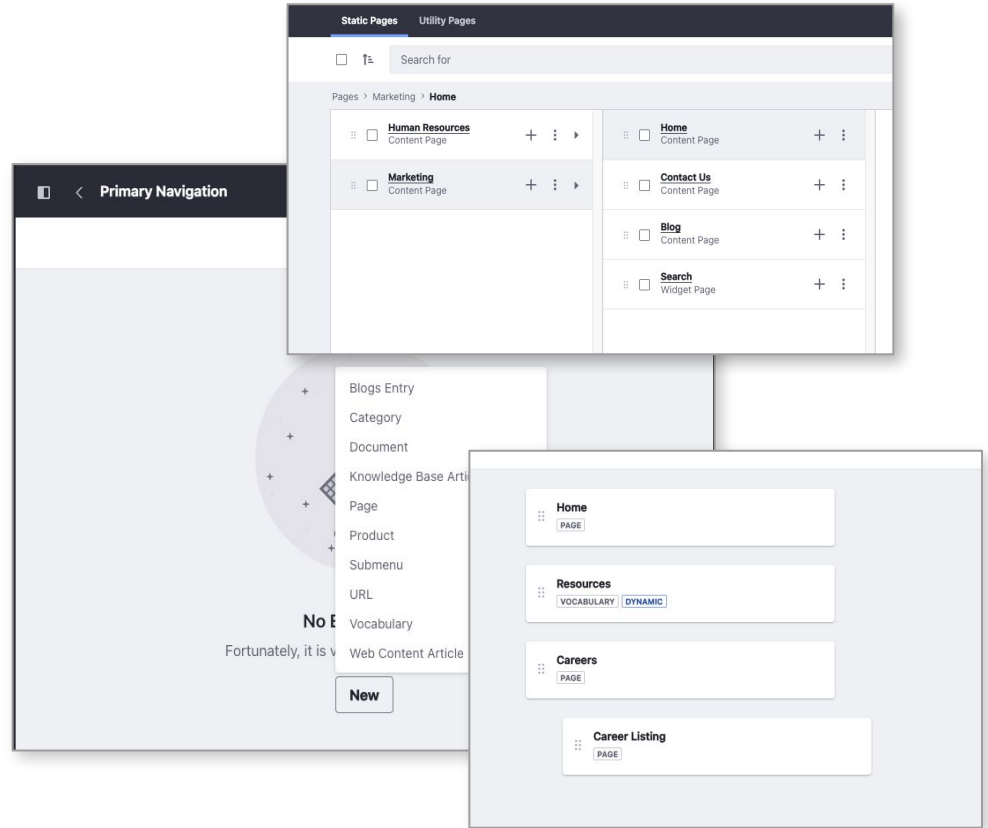
Clarity Business Requirements

- Provide users with a clear and intuitive navigation structure to easily find the information they need.
- Tailor navigation based on user roles, preferences, or location to deliver personalized experiences
- Ensure all website content is easily accessible and discoverable through navigation
- Support different website languages and display dynamic menus based on the selected language
- Control access to specific pages or sections of the website based on user roles or permissions
- Maintain consistent branding and visual identity across the website, including navigation menus



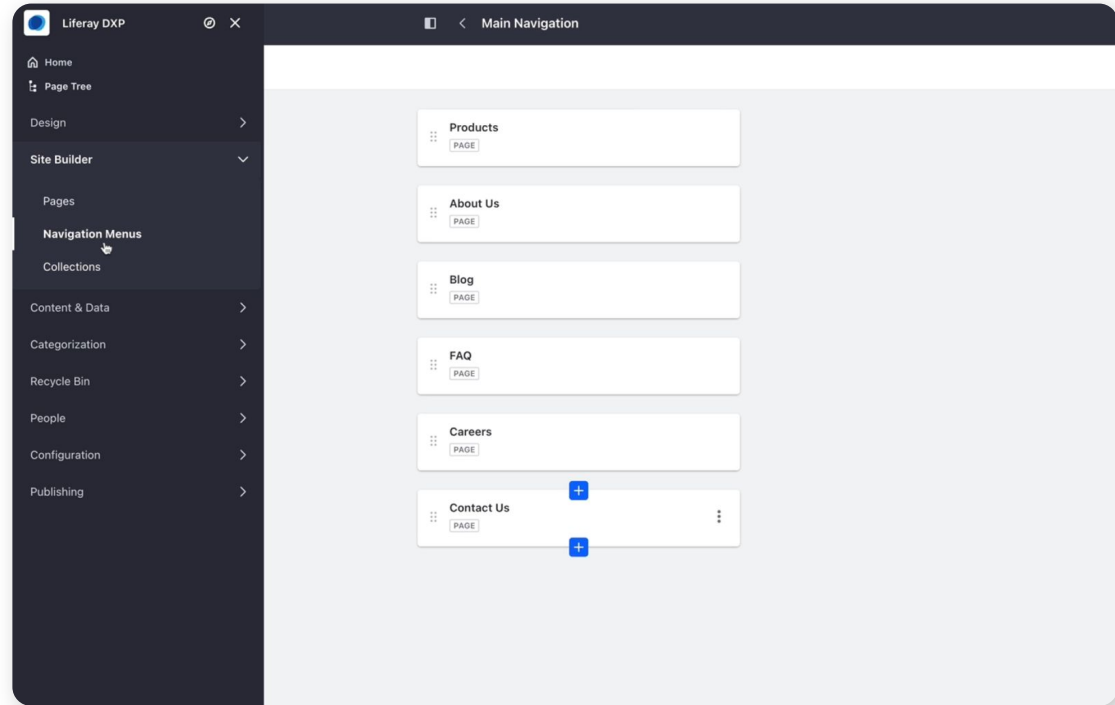
Building Menus

- Menus created independently of pages
- Liferay provides several menu types out of the box (eg. primary, secondary)
- Administrators are also free to define custom menu types
- Simple and intuitive interface for building menus
- Access controls by setting visibility for specific roles
- Supports dynamic content
 - Eg. Using a Vocabulary



Menu Building in Action

1. First we check the expected design
2. Edit a new page and add container and menu fragment
3. Next we go to *Navigation Menus*
4. Create a new menu – notice all the options
5. Add the pages as per the design
6. Reorder the items as per design
7. Go back to editing the new page
8. Configure the fragment with new menu
9. Update styles for container to center it
10. Publish
11. Check the results



Rendering Navigation

- Supported by *Content Pages* and *Widget Pages*
- Menu Display Fragment out of the box for standard horizontal and vertical menus
- Menu Display Widget for more advanced use cases
- Supports Widget Templates
- Templating based on Freemarker offers unlimited options for rendering

The screenshot displays the Liferay Admin Console interface. At the top, the 'Templates' section is active, showing 'Widget Templates'. A search bar is present, and a dropdown menu lists various templates, with 'Menu Display Template' selected. Below this, a table lists widget templates:

ID	Name	Description	Type
34337	Main Navigation		Menu Display Template

The main area shows the configuration for the 'Main Navigation' widget. The code editor displays the following Freemarker template:

```
1 <div class="clarity-navigation-container">
2   <#if entries?has_content>
3     <ul class="list-group list-group-horizontal">
4       <#list entries as navigationEntry>
5         <li class="list-group-item">
6           <a href="#">${navigationEntry.getName()}</a>
7         </li>
8       </#list>
9     </ul>
10  </#if>
11 </div>
```

The right sidebar shows the 'Elements' panel with a search bar and a list of fields: Header Type, Included Pages, Nested Children, Root Page Level, Root Page Type, and Navigation Entries*. The 'GENERAL VARIABLES' section shows the 'Current URI'.

Part 06

Getting Ready for Design + Client Extensions



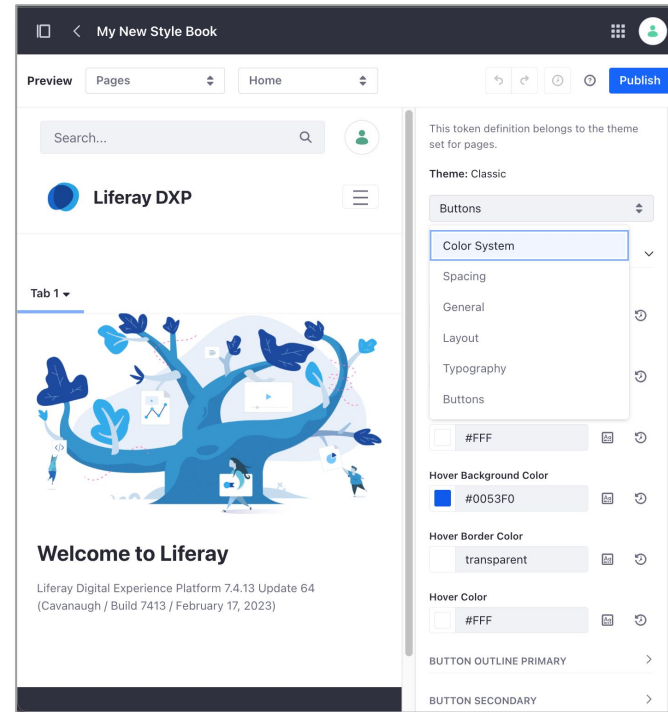
Clarity Business Requirements

- Ensure a consistent visual identity and user experience across all website elements, including fonts, colors, logos, and layouts
- Facilitate collaboration among designers, developers, and content creators to maintain design consistency
- Reduce design effort and time spent on repetitive tasks like defining and applying styles
- Integrate Liferay with external systems and applications to access or display data from external sources
- Leverage existing JavaScript libraries or frameworks to build custom functionalities within Liferay
- Reduce development time and simplify deployment of new features or functionalities



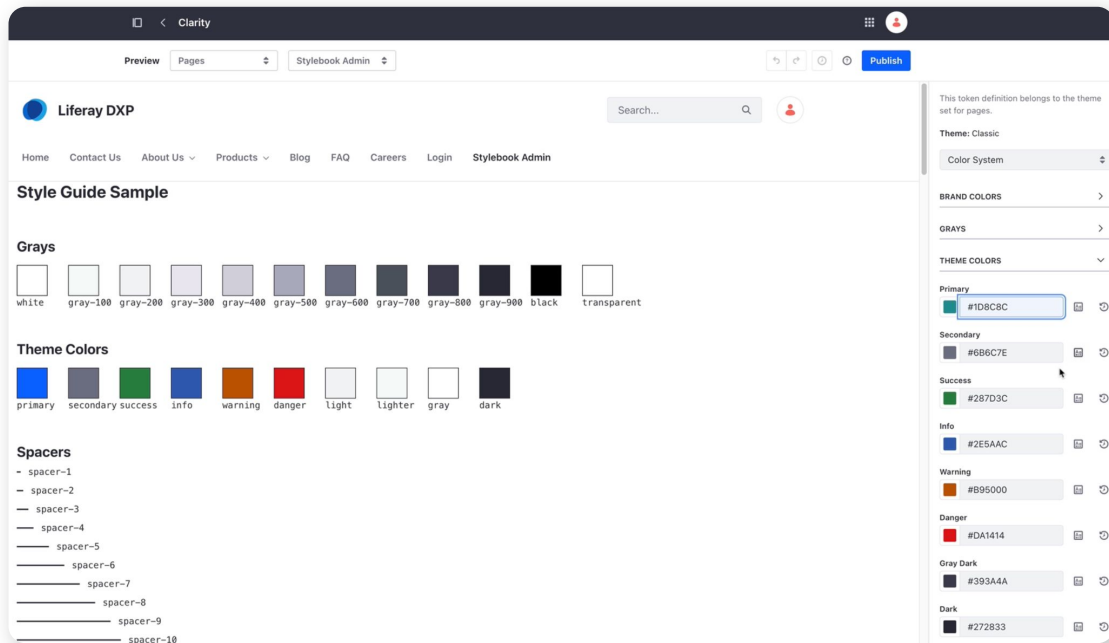
Live Edits with Style Books

- New functionality introduced in DXP 7.3
- Provide the ability to define which CSS (variables) can be adjusted at runtime
- Allows for Theme variations similar to the old “color schemes” capability
- Scope of customizability can be established inherently (omission of certain variables)
- Eliminates complex or lengthy change processes
- Allows designers and marketers to own design management function



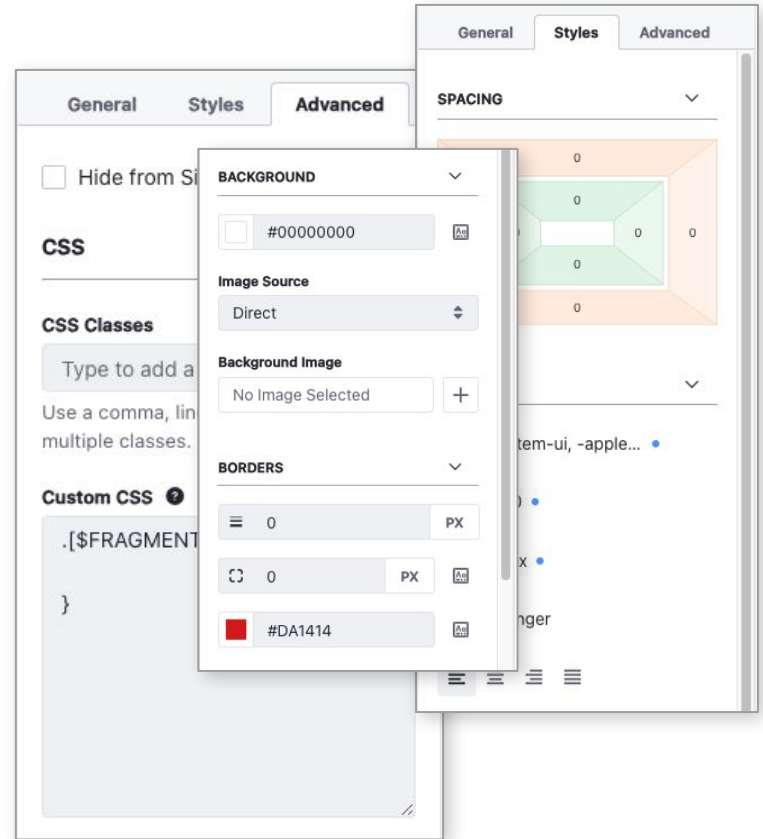
Style Books in Action

1. Create a test page with the “Classic Style Book” widget on it
2. Create a new Style Book
3. Modify the theme colors
4. Modify the base font size
5. Modify the heading using pixels as the measurement given by designers
6. Modify brand colors for primary and secondary button
7. Go back and EDIT test page again
8. Using the paint roller to see stylebooks
9. Choose the Style Book to apply
10. Publish the page



Instance Based Styles

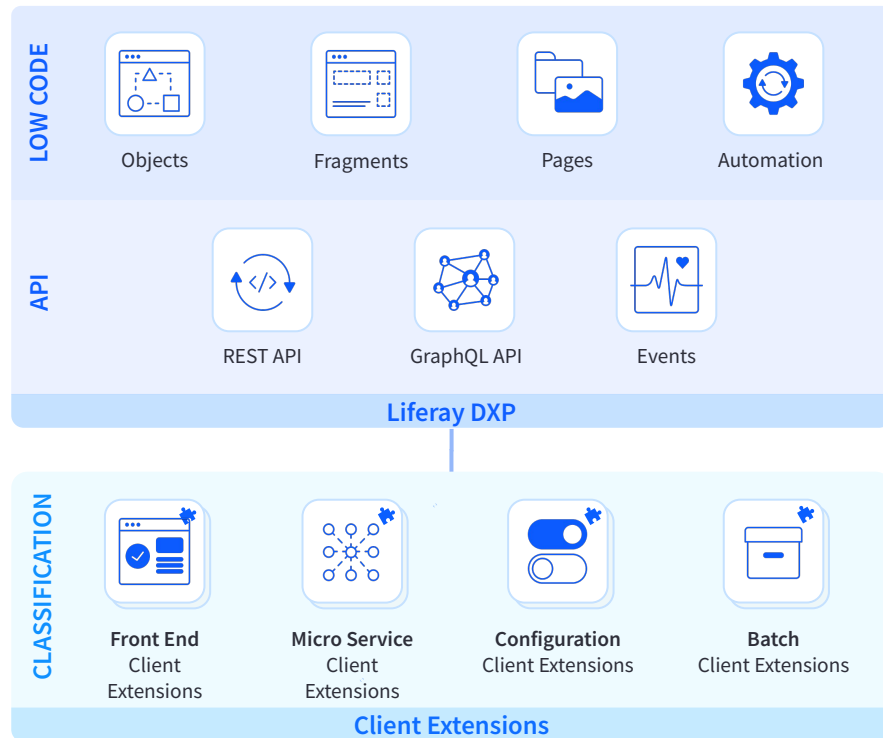
- Cases where you need something temporary or a one-off for a short period of time
- Equivalent to “inline” styling but can be done visually through point-and-click configuration
- Applicable to all fragments and widgets when working with *Content Pages*
- Does not require duplication of code
- Only applicable to fragment/widget being edited – even if same item is used more than once on same page



JS + CSS Client Extensions

- Alternative option to traditional plugins
- Client Extensions (CX) are “Externalized” plugins and extensions
- Frontend Client Extensions support JS, CSS, Spritemap, Favicon, Custom Elements and more
- Registered in DXP so that it is aware of the resources, but DXP does not control or manage them
- Supported by all three offerings
 - Liferay Self-Hosted
 - Liferay PaaS
 - Liferay SaaS
- Provide extension while simultaneously mitigating many challenges with updates

■ Can be used in place of traditional Theme plugins



Client Extensions Registered

- Can be added manually through configuration UI, or via deployment (zip containing metadata in YAML)
- Liferay administrative UI allows you to choose where and which extensions to be applied
- Supports granular approach (application at page level) or something more broad
- Can be used to augment existing resources or replace them (the theme) entirely
- Remember – these can be used in place of traditional Theme plugins

The image shows two overlapping screenshots from the Liferay administrative interface. The left screenshot, titled "Client Extensions", displays a table with three entries:

Name	Type
Animate CSS	CSS
SweetAlert2	JS
SweetAlert2	CSS

Below the table, it indicates "10 Items" and "Showing 1 to 3 of 3 entries." The right screenshot, titled "New Theme CSS", shows a configuration page for defining CSS resources. It includes a search bar, a "Name" field, and a language selector set to "en-US". A rich text editor is visible with a toolbar containing options like "Styles", "B", "I", "U", and "Source". Below the editor, there are sections for "CONTENT" with fields for "Main CSS URL" (with a note "This CSS replaces main.css.") and "Clay CSS URL" (with a note "This CSS replaces clay.css.>").

Client Extensions Configured

- Supported by both Widget Pages and Content Pages
- Accessible from the Page Settings, under the Design section
- Support multiple items meaning you can have more than one external resource listed (no need to manually merge)
- Control the order in which added entries are loaded into the page (when more than one configured)
- Javascript extensions can be loaded in either the head or bottom of the page

Customization

CSS JavaScript

THEME CSS CLIENT EXTENSION

Use this client extension to fully replace the default CSS contained in the theme. Be aware that all the declared design tokens and CSS variables will be overridden. Therefore, style books will not provide a way to style pages anymore.

Theme CSS

No theme CSS client extension was loaded.

CSS CLIENT EXTENSIONS

Extend this page CSS with client extensions. They will be loaded after the theme CSS and after master extensions.

No CSS Client extensions were loaded.

THEME SPRITEMAP CLIENT EXTENSION

Info To add or edit the existing spritemap, simply copy/paste and make changes as needed to your registered extension.

Use this client extension to fully replace the default spritemap contained in the theme.

Theme Spritemap Client Extension

Name



Craft your vision, effortlessly. Build
engaging websites with Liferay

Thank you