



PABLO MOLINA & VÍCTOR GALÁN

Advanced Fragments Workshop



PABLO MOLINA & VÍCTOR GALÁN

Advanced Fragments Workshop

Requirements

Liferay Portal CE installed and running (GA-77).

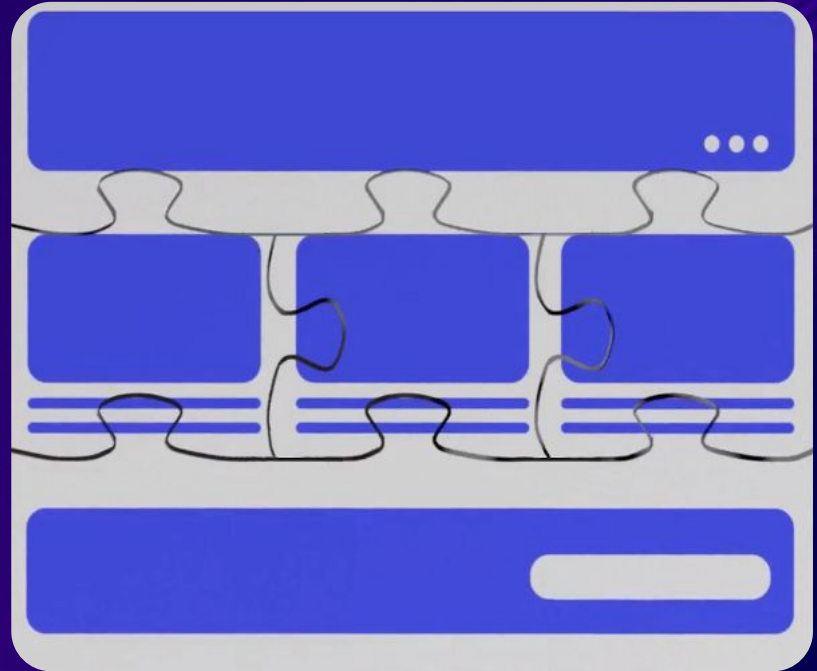
(Optional): Some terminal, code editor, and NodeJS v16.

Requirements

- Some recent Liferay Portal CE installed *and running*
 - We will be using GA-77.
- Optional, but desirable:
 - Some terminal.
 - Some code editor.
 - NodeJS v16.

What is a Fragment?

- Building blocks for Content Pages.
- CSS, HTML, and JavaScript.



Fragment development docs

Always look for documentation in [Liferay Learn](#).

Adding Configuration Options to Fragments

Available: Liferay DXP 7.2 SP1+

Configurable options help make your Fragments flexible, so you don't have to maintain many examples, instead of having one Fragment that has a heading with style A and another Fragment with style B, you can create one Fragment that has a configurable style for the heading with options. You'll learn how to add configuration options to a Fragment:

- Deploy a Configurable Fragment
- Examine the Configuration
- Modify the Configuration
- Propagate the Changes and Test
- Related Information

NOTE
For Liferay DXP 7.4+, Fragment Collections are called Fragment Sets in the Liferay UI.

Deploy a Configurable Fragment

Start a new Liferay instance by running

```
docker run -it -m 8g -p 8080:8080 liferay/portal:7.4.3.55-ga55
```

Sign in to Liferay at <http://localhost:8080>. Use the email address `test@liferay.com` and the password prompted, change the password to `learn`.

Then, follow these steps to deploy an example to see how Fragment configuration options work

Using the Fragments Editor

Liferay DXP includes a built-in editor for developing Content Page Fragments. To access the editor, click the **Fragment Editor** icon and go to *Design* → *Fragments*. From here, you can view, manage, and create Fragments and Fragment Sets.

- Creating a Fragment Set
- Creating a New Fragment

NOTE
For Liferay DXP 7.4+, Fragment Collections are called Fragment Sets in the Liferay UI.

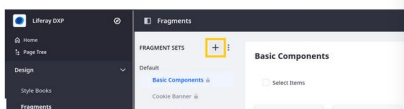
Creating a Fragment Set

Before creating new Fragments, you must first create a new Fragment Set to put them in:

- Open the *Site Menu*, and go to *Design* → *Fragments*

NOTE
In Liferay DXP 7.1 and 7.2, open the *Product Menu*, and go to *Site* → *Site Builder* → *Design* → *Fragments*.

- Click the *Plus* button (+) next to *Fragment Sets*.



Using Custom Fields in Page Fragments

If you've added **custom fields** to Liferay's users or pages, you can access them in fragments. To get the value of a page custom field, use


```
layout.getExpandoBridge().getAttribute("ATTRIBUTE_NAME")
```

To use a page custom field in a fragment, first **add the custom field**, then add or edit a page fragment via the fragments editor, from the *Site Menu* → *Fragments*.

This example HTML uses the custom field `showFooter` to show or hide the placeholder `footer` element on a page:

```
[#assign showFooter = layout.getExpandoBridge().getAttribute("showFooter")]
[#if showFooter]
<footer class="bg-dark pb-8 pt-6 section-footer text-white">
  Placeholder footer content...
</footer>
[/#if]
```

Toggling a page's `showFooter` value to true or false shows or hides the footer, respectively:



Workshop content

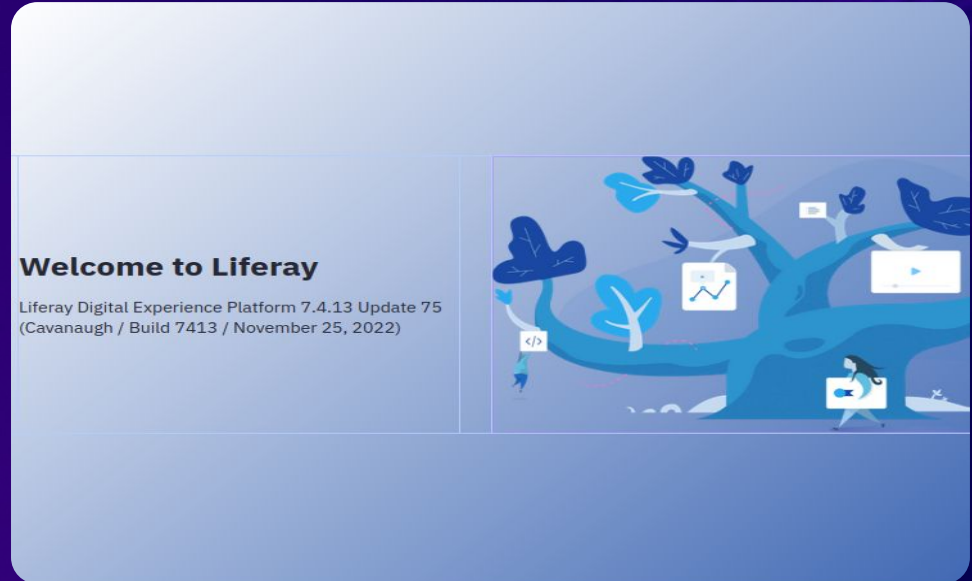
- Create a "Background Gradient" fragment using drop-zones and configuration.
- Create a "Select" form fragment.
- Create a "Chart" React fragment using Fragment Toolkit.
- Recap, questions & comments.



Or let's have fun
discussing and solving
your real use cases.

Background Gradient fragment

- Drop Zones.
- Configuration.




Select Form Fragment

- Form fragments.
- They *need* an input with the correct name attribute.
- Each input will be sent to the server when submitting the form.

Zelda favourite game

Tears of the kingdom ▾

Submit



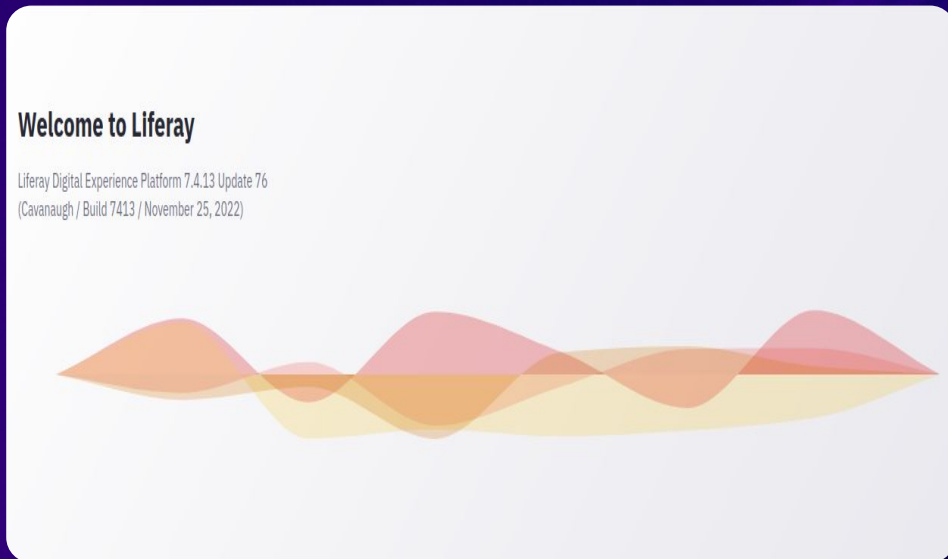
Zelda favourite game

✓ Tears of the kingdom
Minish cap
Breath of the wild

Submit

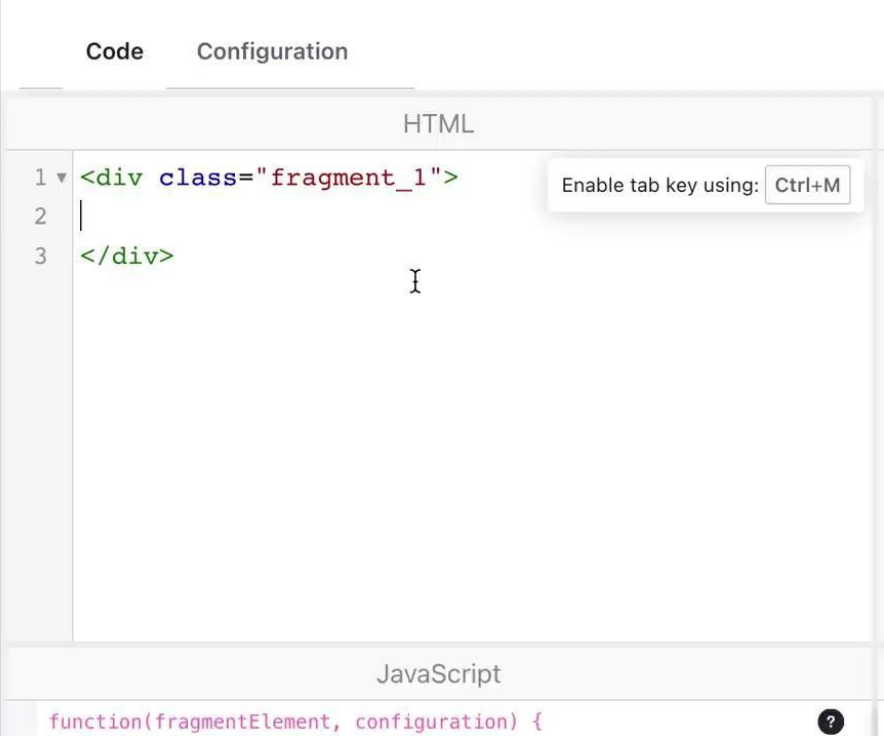
Chart React Fragment

- NodeJS v16.
- Fragment Toolkit.
- Chartist.



Fragment development extra

- Freemarker documentation (version 2.3.29 is used in the latest version of liferay).
- There are plenty context variables to be used within the fragment's freemarker:



The screenshot shows a code editor interface with two tabs: 'Code' and 'Configuration'. The 'Code' tab is active and displays HTML code. The code is as follows:

```
1 <div class="fragment_1">
2 |
3 </div>
```

Line numbers 1, 2, and 3 are visible on the left side of the editor. A cursor is positioned at the end of line 3. A tooltip on the right side of the editor indicates 'Enable tab key using: Ctrl+M'. Below the HTML code, the 'JavaScript' tab is visible, showing the start of a function definition:

```
function(fragmentElement, configuration) {
```

A question mark icon is located at the bottom right of the JavaScript code area.

Fragment development extra

TemplateContextContributor

Blade template sample

```
1 @Component(  
2     immediate = true,  
3     property = {"type=" + TemplateContextContributor.TYPE_GLOBAL},  
4     service = TemplateContextContributor.class  
5 )  
6 public class YourCustomTemplateContextContributor  
7     implements TemplateContextContributor {  
8  
9     @Override  
10    public void prepare(  
11        Map<String, Object> contextObjects, HttpServletRequest request) {  
12  
13        contextObjects.put("yourNewBeanToUseFromTemplate", new  
14 YourBeanThatYouWantToUse());  
15 }  
16
```

Links

- Slides: bit.ly/advanced-fragments-slides
- Repo: bit.ly/advanced-fragments-repo





How was this session?

Please share your rating in
the event app. Thank you!