

ÁRBOL DE PÁGINAS

- >

Configuración y enseñanza
- >

Desarrollo de luz

>

[Configuración] Entorno de desarrollo de luz

>

Base de desarrollo ligero

>

Plantillas: FreeMarker
- >

Desarrollo de sitios web

>

Redundancias: detectar y eliminar redundancias

>

Área: Herencia de contenido

>

Definición del sitio en JCR: general

>

Definición del sitio en YAML: decorar según Yaml

>

Definición del sitio en YAML: Plantillas-Prototipo

>

Definición del sitio en YAML: Disponibilidad de plantillas OTB

>

Temas: Descripción completa

>

(O) Tipos de plantilla y ayuda de tipo de plantilla

>

(O) Área: Área de escritura pura - Incluye fantasía

>

(O) Los recursos dinámicos incluyen: DefParameters y Themes

>

(O) Definición del sitio en YAML: Plantillas personalizadas-Disponib
- >

(O) Área: Esqueleto de página limpio y flexible

>

(O) [Ejercicio] Transferir 'pageHeader' a un área

>

(O) [Ejercicio] Transferir 'htmlHeader' a un área

>

(O) [Ejercicio] Transferir 'endPageJsScripts' a un área

>

(O) Definición de página: Anulaciones de 'Plantillas-Prototipo'
- >

Magia de back-end de JS
- >

(O) Material adicional LD

Herramientas de espacio



(O) [Ejercicio] Transferir 'endPageJsScripts' a un área

Este ejercicio depende del último ejercicio:
(O) [Ejercicio] Transferir 'htmlHeader' a un Área
Asegúrate de haberlo hecho antes de este ejercicio.

Objetivos

- Ser capaz de tener un esqueleto de página súper limpio en ' defaultPageScript.ftl '.
- ¡Uso de un script de página global **para cualquier situación en cualquier sitio y para cualquier página!**

Resultados

- La inclusión de recursos JS se ha **transferido** a un **área dedicada** : área ' endPageJsScripts '.
- ¡El ' defaultPageScript.ftl ' ahora está listo para su uso en **cualquier situación, en cualquier sitio y en cualquier página!**

Tareas / Procedimiento

Crear área ' htmlHeader '

1. Cree la definición de área ' endPageJsScripts ' .
Pasos detallados

- a. Abra la definición del prototipo ' prototipo-de-todas-las-plantillas-de-página.yaml ' .
Agregue una nueva definición de Área ' endPageJsScripts ' en el nodo ' áreas ' en la parte inferior.

```
areas:
  endPageJsScripts:
    type: noComponent
    createAreaNode: false
    templateScript: /training-templating-website/templates/pages/areas/endPageJsScriptsArea.ftl
```

2. Cree script de área para ' endPageJsScripts ': endPageJsScripts.ftl
Pasos detallados

- a. En la carpeta ' templates/pages/areas ' , cree ' endPageJsScripts.ftl ' y pegue el siguiente código:

```
[#-- Including JS manually:
<!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
<script src="${ctx.contextPath}/.resources/training-templating-website/webresources/jquery.min.js"></script>
<!-- Include all compiled plugins (below), or include individual files as needed -->
<script src="${ctx.contextPath}/.resources/training-templating-website/webresources/bootstrap.min.js"></script>
--]

[#-- Using the 'resfn' for a more dynamic approach. --]
${resfn.cachedJs(["/training-templating-website.*jquery.*js", "/training-templating-website.*bootstrap.*js"] ) }
```

3. Renderice el área ' endPageJsScripts ' .
Pasos detallados

- a. En ' defaultPageScript.ftl ' reemplace la inclusión JS a través de ' resfn ' con este código:

```
<!-- endPageJsScripts -->
[#-- For loading the JS resources you can link them manually. It is the inner path within the light module that always follows '${ctx.contextPath}/.r
[@cms.area name="endPageJsScripts"/]
<!-- End: endPageJsScripts -->
```

Esto activa la representación del área ' endPageJsScripts ' .

Tenga en cuenta: Básicamente, movimos el código de inclusión anterior de ' defaultPageScript.ftl ' a un script de área específica: ' endPageJsScripts.ftl ' .
Así que liberamos ' defaultPageScript.ftl ' de cualquier inclusión JS relevante de módulo de diseño/ligero, ahora es súper neutral !

- b. **Pruebe** su trabajo:
rendice su página de ' capacitación ' fuera de AdminCentral de Magnolia usando este botón en la esquina inferior derecha:

Preview in tab

Debería ver que se incluyen los mismos recursos que antes.

4. ¡Entiende los **beneficios** de tu trabajo!
Pasos detallados

- a. Script de página predeterminado **limpio y flexible** : defaultPageScript.ftl
Vea lo agradable y neutral que es ahora el script de página general:

```
<!DOCTYPE html>
<html xml:lang="${cmsfn.language()}" lang="${cmsfn.language()}">
<head>
  [@cms.page /]

  <!-- htmlHeader -->
  [@cms.area name="htmlHeader"/]
  <!-- End: htmlHeader -->
</head>

<body>

  <div class="container">
    <!-- Navigation -->
    [@cms.area name="navigation"/]
    <!-- End: Navigation -->

    <!-- Page Header -->
    [@cms.area name="pageHeader" /]
    <!-- End: Page Header -->

    <!-- Main Area -->
    [#-- 'main' Area containing Components: 'textImage' & 'twoColumns' --]
    [@cms.area name="main" /]
    <!-- End: Main Area-->

    <!-- Footer -->
    [#-- 'footer' Area containing Components: MTK's 'Text and image' component. --]
    [@cms.area name="footer"/]
    <!-- End: Footer -->
  </div>

  <!-- endPageJsScripts -->
  [@cms.area name="endPageJsScripts"/]
  <!-- End: endPageJsScripts -->

</body>
</html>
```

Acerca del área 'htmlHeader':
En el [ejercicio anterior](#) movimos el **antiguo código de etiqueta 'head'** de 'defaultPageScript.ftl' **a un script de área específico**: 'htmlHeaderArea.ftl'. Esto significa, como se explicó en el [ejercicio anterior](#), que cualquier inclusión relevante de diseño (CSS) o JS se administra mediante el área 'htmlHeader'. **Acerca del área 'pageHeader'**: En el segundo [ejercicio anterior](#) movimos la **estructura de contenido** de la página de 'defaultPageScript.ftl' **a un script de área específico**: 'pageHeaderArea.ftl'. Esto significa, como se explicó en el segundo [ejercicio anterior](#), que la estructura del contenido de la página ahora se puede definir libremente de forma general y por tipo de página. **Acerca del área 'endPageJsScripts'**: En este ejercicio movimos el código **de inclusión JS anterior** de 'defaultPageScript.ftl' **a un script de área específico**: 'endPageJsScripts.ftl'. Esto significa que cualquier inclusión relevante de diseño (CSS) o JS es manager por el área 'htmlHeader' y el área 'endPageJsScripts' recién agregada.



Relizar: ¡Ahora está todo terminado! ¡Tenemos el esqueleto de script de página perfecto que se puede utilizar en **cualquier situación, en cualquier sitio y en cualquier página!**