

Titulo

Jose Juan Montiel

Version 1.0, 2016-03-22

Table of Contents

Primer fichero	1
Pongamos CSS, JS e IMG	1
Y ahora copy/paste	1
Slide Two's Title will not be displayed	1
Empezemos con Thymeleaf	1
Para que coja los cambios en caliente (empecemos con el HTML)	2
Hasta ahora solo es html	2
Si los transformamos en templates	2
Ahora que son templates, porque no usar toda la potencia	2
Porque no ir mas alla, formularios	2
Thymeleaf 3	2
JBake	3
Servicios rest - JSON.....	3
SASS / LESS - Gradle	3
Grooscript.....	3
Como usar Git Presenter.....	3
Como generar la presentacion con asciidoctor.....	3

Primer fichero

Primera version de un html

```
<!doctype html>
<html>
<head></head>
<body></body>
</html>
```

Pongamos CSS, JS e IMG

Segunda version del html usando css, js e imagenes.

```
<!doctype html>
<html>
  <head>
    <link type="text/css" rel="stylesheet" href="css/main.css" media="all">
    <script type="text/javascript" src="js/main.js"></script>
  </head>
  <body>
    
    <button onclick="ejemplo()">LOGICA</button>
  </body>
</html>
```

Y ahora copy/paste

Una vez que tenemos la estructura nos ponemos a hacer paginas como churros.

Slide Two's Title will not be displayed

By Neil T - CC BY-SA 2.0

Empezemos con Thymeleaf

Thymeleaf es un sistema de templating Java, que se suele utilizar dentro del ecosistema Spring.

Vamos a crear un proyecto spring-boot usando esta web: <http://start.spring.io/>

Y lo arrancamos con: `~/workspace/thymeleaf_talk$./gradlew bootRun`

Para que coja los cambios en caliente (empecemos con el HTML)

Añadimos al build.gradle

```
bootRun {  
    addResources = true  
}
```

Hasta ahora solo es html

Notar que hasta ahora estabamos mostrando estaticos, pero thymeleaf usa atributos distintos de los estandar con lo que podriamos seguir maquetando en "plano", notar los imports.

Si los transformamos en templates

Añadimos un controlador (para gobernarlos a todos) y ponemos los tags de thymeleaf (dejando los estandar) Hacer uso de un css distinto, y de un js distinto..

Ahora que son templates, porque no usar toda la potencia

Layouts Includes

Porque no ir mas alla, formularios

Binding, y la potencia de groovy

Thymeleaf 3

<http://www.thymeleaf.org/doc/articles/thymeleaf3migration.html>

Docoupled logic

JBake

http://jbake.org/docs/2.4.0/#project_structure

Servicios rest - JSON

<https://github.com/olivergierke/spring-restbucks> <https://github.com/ilopmar/contest>

SASS / LESS - Gradle

<https://github.com/robletcher/gradle-compass-bootstrap-with-gulp-2/>

<http://broonix-rants.ghost.io/spring-boot-building->

Grooscript

<http://grooscript.org/doc.html>

Como usar Git Presenter

Se han seguido los pasos de [esta documentacion](#).

Como generar la presentacion con asciidoctor

Se han seguido los pasos de [esta documentacion](#).

Para generar (ejecutar dentro de docs):

- HTML
 - `asciidoctor -T asciidoctor-deck.js/templates/haml manual.adoc`
- PDF
 - `asciidoctor -r asciidoctor-pdf -b pdf manual.adoc`

Usando gradle (en la raiz)

- `gradle asciidoctor`