

Descripción de tareas del SPRINT 2

El SPRINT 2 consta de tres partes a desarrollar, cada una de las cuales se describe en detalle a continuación:

1. La primera de las partes consiste en incluir ya el contenido final que tendrá el sitio web a desarrollar de forma dinámica empleando para ello vanilla Javascript. Para hacerlo, este contenido final deberá incluirse en un fichero JSON que deberá estar estructurando adecuadamente como si de una base de datos se tratara, puesto que se supone que este fichero JSON debería ser generado exportando el contenido desde la base de datos. Este fichero JSON incluyendo los datos a cargar en el sitio web se encontrará (por ahora) almacenado localmente junto al resto de ficheros HTML, CSS y Javascript del sitio web pero debe tenerse en cuenta que, con posterioridad, podría ser obtenido remotamente desde un servidor. Una opción para simular esto podría ser utilizar un “fake server” como JSON-Server. En este fichero, además del contenido general del sitio web, podríamos incluir también los distintos usuarios que podrán ser validados y autenticados en el formulario de validación implementado para este propósito en el sitio web. Otra opción para generar este fichero JSON con el contenido del sitio web (incluyendo los usuarios) puede ser utilizar Strapi para generar el backend de una forma más cómoda y utilizar su API para acceder a los datos usando la API que incluye. En esta parte del Sprint es donde se verá el potencial de usar los templates para generar las distintas páginas que contendrá el sitio web cargando dinámicamente el contenido de los mismos desde el fichero JSON usando Javascript para hacerlo. El uso de Javascript también nos permitirá generar estas páginas del sitio web usando una composición de templates más básicos (por ejemplo, un header o un footer reusables) que vayan sumándose para generar el template final y rellenarlo entonces con el contenido adecuado para obtener la página web final.
2. La segunda de las partes consiste en incluir al menos un formulario en alguna de las paginas del sitio web y realizar su validación en el lado cliente usando el soporte para validación nativa que incorpora HTML5. Un caso de uso para formularios que se obliga a implementar es uno que permitiera registrar y autenticar usuarios de forma que podamos, por ejemplo, utilizar estos usuarios con posterioridad dentro del sitio web para mostrar un contenido diferente dependiendo de si el usuario se ha autenticado o no, o de haberse autenticado, diferenciar el contenido dependiendo del rol actual del usuario autenticado.
3. La tercera y última de las partes consiste en modificar los templates que se tengan actualmente aplicando RWD para adaptarlos a las características de los distintos dispositivos desde los que pueda ser accedido el sitio web a desarrollar. Para ello se deben realizar al menos tres variantes del diseño original de cada uno de los templates existentes, estos tres diseños mínimos requeridos deben ser adaptados para que su contenido se muestre adecuadamente en dispositivos de tamaño grande (por ejemplo, desktop y portátil), medio (por ejemplo, tablet en landscape) y pequeño (por ejemplo, móvil). Para esta adaptación a RWD de los actuales templates se pueden usar elementos flotantes, posicionamiento de elementos, layouts fluidos o flexibles, media queries y sistemas en grid. Para verificar que los diseños se adaptan correctamente deberemos ir disminuyendo o aumentando el tamaño de la ventana del browser hasta alcanzar los distintos breakpoints definidos y entonces comprobar que el diseño cambia adecuadamente y como se esperaba.