

Laboratorio RIA 2022



Grupo 4

Alexis Peralta – Nicolás Guillén

Contenido:

- **Introducción y problemática planteada**
- **Implementación: Tecnologías Utilizadas**
- **Problemas encontrados durante el desarrollo**
- **Single Page Application vs Server Side Rendering**
- **Fuentes y Referencias**

Introducción y Problemática Planteada

La carrera “Tecnólogo en Informática” tiene una sección web dentro del sitio de FING/UdelaR, dicha sección es un sistema muy antiguo que actualmente está en desuso. Si bien la carrera tiene una sección dentro de la página de UTEC, es deseable mantener actualizada la información dentro de la sección del sitio de FING.

Para ello se propone desarrollar un sitio web del tipo SPA con el framework Angular y un framework CSS asignado por sorteo.

El sistema tendrá un único usuario administrador que será capaz de modificar el contenido dinámico del sitio.

Tecnologías Utilizadas

HTML

El Lenguaje de Marcado de Hipertexto (HTML) es el código que se utiliza para estructurar y desplegar una página web y sus contenidos. Por ejemplo, sus contenidos podrían ser párrafos, una lista con viñetas, o imágenes y tablas de datos.

HTML no es un lenguaje de programación; es un lenguaje de marcado que define la estructura de tu contenido. HTML consiste en una serie de elementos que se usan para encerrar diferentes partes del contenido para que se vean o comporten de una determinada manera



CSS

Hojas de Estilo en Cascada o CSS es el lenguaje de estilos utilizado para describir la presentación de documentos HTML o XML. CSS describe como debe ser renderizado el elemento estructurado en la pantalla, en papel, en el habla o en otros medios.



JavaScript

JavaScript (JS) es un lenguaje de programación ligero, interpretado, o compilado justo-a-tiempo con funciones de primera clase.

Si bien es más conocido como un lenguaje de scripting para páginas web, es usado en muchos entornos fuera del navegador, tal como Node.js.

JavaScript es un lenguaje de programación basada en prototipos, multi paradigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa.



Angular

Angular es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

Angular se basa en clases tipo "Componentes", cuyas propiedades son las usadas para hacer el binding de los datos. En dichas clases tenemos propiedades (variables) y métodos (funciones a llamar).



Ng-Bootstrap

Framework que consiste en una gran biblioteca de componentes de Angular basadas en el Bootstrap original.

Esta librería nos ofrece componentes con la hoja de estilo de Bootstrap pero sin la necesidad de utilizar las librerías JQuery, Popper y la librería Javascript propia del Bootstrap original.



Git

Git es un software de control de versiones pensado en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente.

Su propósito es llevar registro de los cambios en archivos de computadora incluyendo coordinar el trabajo que varias personas realizan sobre archivos compartidos en un repositorio de código.



Problemas encontrados durante el desarrollo

Consideramos que durante todo el proceso de desarrollo de la aplicación no nos encontramos con ningún gran problema que nos dificultara o retrasara en cuanto al avance progresivo del mismo.

Sin embargo, podemos mencionar aquellos aspectos que requirieron un poco más de investigación u ocuparon algo más del tiempo que lo esperado del total utilizado en el proyecto:

- **Realizar las peticiones a la API:** No teníamos mucha experiencia realizando peticiones de este tipo así que demandó un poco más de estudio e investigación
- **Utilizar los componentes del Ng-Bootstrap:** A diferencia del Bootstrap normal, estos componentes al estar diseñados para Angular, utilizan el bindeo entre el template HTML y la lógica en el archivo TypeScript, así que cambiaba un poco la manera de usarse.

Single Page Application (SPA) vs Server Side Rendering (SSR)

SPA:

Las SPA son páginas que se sirven siempre desde el mismo fichero, es decir, que se tienen que descargar antes de poder ejecutarse.

Esto tiene una mejora en cuanto a la experiencia de usuario, ya que el usuario tiene la sensación de estar navegando entre páginas de una manera muy rápida.

Una SPA no tiene páginas, tiene vistas, y aunque veamos que la URL va cambiando, realmente esto no es así, ya que se sirve todo desde la misma página.

Esto hace que, a día de hoy, los robots de indexación, como el de Google, tienen problemas para ejecutarlas, ya que no son capaces de hacerlo correctamente y mostrar todo el contenido HTML, por eso no son capaces de entenderlo y de indexarlas correctamente.

Este tipo de páginas se ejecutan siempre de lado del cliente, con sus cosas buenas y sus cosas negativas, ya que mejoran la experiencia de usuario, pero, en el caso de tener una conexión lenta, la descarga de esa página para acceder a ella puede ser un poco más tedioso.

SSR:

Las páginas de SSR, también llamadas páginas isomorfas, vienen a solucionar los problemas que tenían las páginas SPA.

Mejoran la indexación por parte de los robots, ya que el código HTML se sirve en una sola vez y los robots no tienen que ejecutar nada en JavaScript para poder obtenerlas.

Parte del renderizado se realiza por parte del servidor, lo que mejora mucho la performance de la aplicación, y hace que también que la indexación y el SEO (Search Engine Optimization) se mejore.

Fuentes y Referencias

- HTML: https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/HTML_basics
- CSS: <https://developer.mozilla.org/es/docs/Web/CSS>
- JavaScript: <https://developer.mozilla.org/es/docs/Web/JavaScript>
- Angular: <https://angular.io>
- Ng-Bootstrap: <https://ng-bootstrap.github.io/#/home>
- Git: <https://git-scm.com>
- SPA vs SSR: <https://openwebinars.net/blog/diferencias-entre-spa-y-ssr/>