

TÉCNICAS Y HERRAMIENTAS **MODERNAS I**

Módulo 1: Producción de textos e hipertextos **Mark Up Languages**

Nicolás Mondaca, Analía Quiroga, Matilde Berther, Juan Pegorin, Franco Carricart

`gabrielmondacanicolas@gmail.com`

`juanmapegorin@gmail.com`

`aquirogasottile@gmail.com`

`carricartfranco9@gmail.com`

`mathilde.berthet@insa-strasbourg.fr`

Keywords. Tecnología emergente, lenguajes marcados, navegación segura.

1 Introducción

En este primer módulo de la materia nos centramos en aprender herramientas y diversas técnicas con el objetivo de lograr una mayor visibilidad cuando, en un futuro, nos insertemos en el mercado laboral, además de mejorar nuestra reputación virtual. Este objetivo lo logramos mediante plantillas especiales tales como Látex o Word, que permiten el uso de diversos títulos y formatos de textos, así como la presencia de un rastro digital asociado a nuestra identidad. Todas estas herramientas nos dan la posibilidad de crear un curriculum vitae muy completo y profesional que obtendrá una mayor visibilidad entre las empresas.

También aprendimos acerca de cómo navegar los archivos de nuestra computadora de manera segura, evitando así la propagación de un posible virus existente en la misma. Del mismo modo aprendimos acerca de páginas útiles tales como GitHub, en la cual aprendimos a subir nuestra primera página web mediante el uso de HTML.

2 El lenguaje HTML

2.1 Sitio W3ols.html

Para aprender cuales son las etiquetas HTML que debemos emplear en nuestro código para crear una página web, debemos dirigirnos al sitio

`https://www.w3.org/TR/html4/index/elements.html`

En este se muestra una lista, con sus respectivos usos, de las etiquetas. En nuestro código elegimos emplear las etiquetas:

- html, para iniciar nuestra página web;
- head, para indicar la sección en la cual van a estar los datos de nuestra página. En este caso hay que tener en cuenta que esta información no se va a ver cuándo abramos el archivo con el navegador, a excepción del título (con etiqueta title) que se va a ver en la pestaña de la página;
- body, en la cual vamos a escribir el contenido de nuestra página.
- h1, nos indica que el texto de la página es un título. Si queremos que nuestra página presente diferentes tamaños de títulos para organizar el contenido, debemos usar las etiquetas h2, h3, h4, h5 y h6;
- hr, crea una línea horizontal en la página;
- br, nos da un salto desde una oración a otra;
- p, nos indica el comienzo de un párrafo;
- b, cambia el contenido a negrita.

Hay que tener en cuenta como se inician cada una de estas etiquetas. Para iniciarlas siempre se tiene que colocar el nombre de la etiqueta entre los signos “<” y para cerrar la etiqueta se tiene que colocar entre los signos “>”.

2.2 Nuestro primer ejemplo de una página HTML

En el primer paso para crear nuestra primera página HTML, hay que crear un documento de texto con extensión .txt. En este documento crearemos la estructura de nuestra página, que es la que se muestra a continuación.

```
<html>

<head>
  <title>Página de Prueba</title>
</head>

<body>

  <h1>Titulo H1</h1>
  <hr>
  <h2>Titulo H2</h2>
  <br>
  <p><b>Esta es nuestra primera página en HTML</b></p>

</body>
</html>
```

Este archivo no lo tenemos que guardar en el escritorio de nuestra computadora. Hay que crear una carpeta en el disco local C en el cual no tengamos ningún otro elemento, y luego guardar el archivo en esta carpeta. Una observación importante a la hora de guardar el archivo es la extensión de este. Nuestro archivo original tenía una extensión de tipo .txt, para poder abrirlo con nuestro navegador como una página web, tenemos que cambiar esta extensión a .html.

3 Forma de navegar segura

Para navegar de forma segura por nuestro ordenador, sin que virus de dicho ordenador interfieran a la hora de ejecutar un archivo, nos vamos a dirigir al sitio:

```
file:///C:/
```

En este sitio, vamos a encontrar el índice de Windows. En el vamos a ver todas las carpetas que corresponden a dicho disco. De este modo, ingresando por el navegador de Google, evitamos ejecutar mal archivos por algún virus que se descargó o estaba en el ordenador.

4 Sitio de GitHub

El próximo paso es compartir el archivo HTML que habíamos creado con los profesores, y los futuros informes que debamos realizar, a través de GitHub.

GitHub es una plataforma de desarrollo colaborativo para alojar proyectos. Lo primero que debemos hacer es crear una cuenta con nuestros datos (nombre, mail, etc.) y verificar la cuenta a través de nuestro mail.

Una vez que la cuenta es creada, debemos ingresar y crear nuestro primer proyecto. Para esto debemos hacer click en el botón Start a project, como aparece en la Figura 1. Luego de esto va a aparecer un formulario, que se ve en la Figura 2, en el cual tenemos que colocar los datos del proyecto (el nombre del repositorio, la descripción y si es público o no). Tenemos que marcar la opción de crear un archivo README en el proyecto para luego hacer click en crear el repositorio.

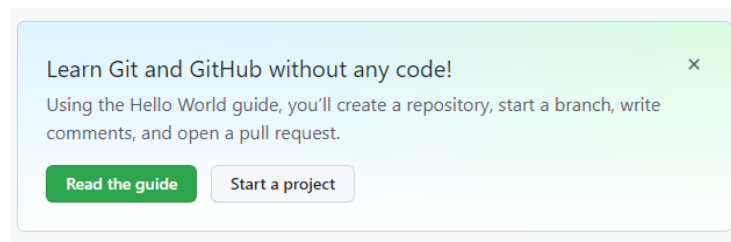



Fig. 1

Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


Owner * Repository name *

 AQuirogaS /

Great repository names are short and memorable. Need inspiration? How about `refactored-fiesta`?

Description (optional)

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☐ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

☐ **Add .gitignore**
Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**
A license tells others what they can and can't do with your code. [Learn more.](#)


 You are creating a public repository in your personal account.

Fig. 2.

Cuando ya se tiene creado nuestro proyecto, podemos editar el archivo README y utilizar las etiquetas aprendidas en HTML para poder darle una estructura a este elemento. En este caso, README debe contener los nombres de los integrantes del grupo y de los profesores.

El último paso en este proceso es subir la página web que habíamos creado, e invitar a los profesores a ver nuestro proyecto. Para esto, nos dirigimos a Add file, como se muestra en la Figura 3, y hacemos click en subir archivo para luego buscarlo y guardar los cambios que se han realizado.

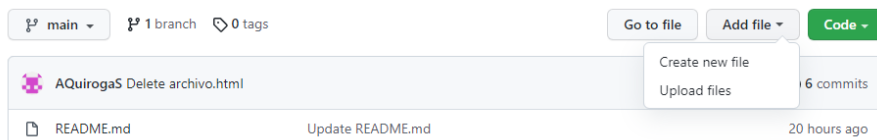


Fig. 3.

Para invitar a otras personas a que vean nuestro proyecto nos tenemos que colocar en colaboradores en la pestaña de configuración del proyecto. Buscamos el colaborador que queramos agregar a través del mail o el nombre con el que aparece en GitHub y lo agregamos al repositorio.

5 Plantillas de la cátedra de Word

En la página de la catedra se presentan dos opciones de plantillas para los documentos que vamos a utilizar, una en un formato de archivo de Word y otro en LaTeX.

En una primera instancia vamos a utilizar la primera plantilla que se presenta cuya extensión es .dot(x). Cuando descargamos este archivo y lo abrimos, vemos que en la pestaña de complementos aparecen varias opciones. A través de estos elementos (H1, H2, Title, Subtitle, Author, etc.) podemos crear un documento bien organizado y estructurado de una manera más sencilla. Además, a través de estas plantillas, los metadatos de los archivos que vayamos a crear no se van a perder.