

UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

Facultad de Ingenierías y Arquitectura

Carrera de Ingeniería en Tecnologías de la Información

Tecnologías Web

Tarea 1

Periodo académico

Octubre 2022/ Febrero 2023



INFORME DE IMPLEMENTACION DE PAGINA WEB

1. TEMA DEL SITIO WEB

Tecnologías Backend

¿Qué es BackEnd?

El backend, conocido como CMS o back office, es la parte de la aplicación que el usuario final no puede ver. Su función es acceder a la información solicitada a través de la app, luego combinarla y devolverla al usuario final.

Características

- Acceder a la información solicitada a través de la app
- Combinar y transformar la información encontrada
- Devolver la información al usuario
- La sub capa, relacionada con el lenguaje específico
- La sub capa visual, relacionada con el diseño estético de la app

Flask

Definición

Flask es un micro framework escrito en Python. Flask no requiere validación, capa de abstracción de base de datos ni biblioteca preexistente.

Características

- Servidor web y depurador integrados
- Cookies seguras
- Documentación robusta
- Enriquecido con una variedad de clientes, herramientas y mecanismos
- Soporta Unicode

Ventajas

- Incluye un servidor web de desarrollo
- Tiene un depurador y soporte integrado para pruebas unitarias
- Admite de forma nativa el uso de cookies

Desventajas

- No contiene bibliotecas integradas
- Crea dificultades al realizar migraciones o pruebas
- La autenticación de usuario de Flask es muy simple

Node.JS

Definición

Node.JS es un marco de back-end de primer nivel para el desarrollo web de JavaScript. Además, Node.JS es un marco back-end multi plataforma de código abierto que se ejecuta en el motor V8. y los utiliza para desarrollar sitios web y API de back-end.

Características

- Muy rápido en la ejecución de código.
- Altamente escalable
- Transferencia de datos rápida desde un Sub proceso único
- Compatibilidad multiplataforma

Ventajas

- Lenguaje sencillo basado en JavaScript
- Permite la gestión concurrente de solicitudes
- Facilita la Creación de aplicaciones de Internet altamente competitivas.

Desventajas

- API Inestable
- Falta una biblioteca estándar
- Falta de bibliotecas en general

Golang

Definición

Golang, o simplemente Go, como lo llaman muchos desarrolladores, no es más que un lenguaje de programación de código abierto desarrollado por Robert Griesemer. Golang se puede utilizar siempre que los desarrolladores necesiten crear un sistema de aplicación modular y de alta velocidad.

Características

- Es un lenguaje concurrente. Esto significa que permite ejecutar múltiples tareas simultáneamente, aunque no utiliza hilos sino canales construidos a partir del mismo código fuente
- Admite miles de conexiones en el mismo programa.
- Permite la instalación en múltiples sistemas operativos como GNU/Linux, Windows, Mac OS X o FreeBSD.
- Orientado a objetos No hace falta enumerar las ventajas de la programación orientada a objetos, pero curiosamente GO no permite herencia de tipos
- Al igual que Java o .NET

Ventajas

- Documentación memorable y fácil de entender
- Excelente opción para el desarrollo de aplicaciones y sitios web
- Simple, confiable y eficiente

Desventajas

- La herencia no se puede implementar en este lenguaje de programación, pero una instancia de un tipo sí.
- Hay poca documentación, digamos manuales o libros de calidad.
- La comunidad detrás de Golang sigue creciendo, aunque la documentación es buena, el lenguaje no tiene una comunidad tan grande como JAVA, C# o PHP.

2. Tecnologías a usar

HTML

HTML es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Se trata de la sigla que corresponde a HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto, que podría ser traducido como Lenguaje de Formato de Documentos para Hipertexto.

Se trata de un formato abierto que surgió a partir de las etiquetas SGML (Standard Generalized Markup Language). Concepto traducido generalmente como «Estándar de Lenguaje de Marcado Generalizado» y que se entiende como un sistema que permite ordenar y etiquetar diversos documentos dentro de una lista. Este lenguaje es el que se utiliza para especificar los nombres de

las etiquetas que se utilizarán al ordenar, no existen reglas para dicha organización, por eso se dice que es un sistema de formato abierto.

Es un lenguaje muy simple y general que sirve para definir otros lenguajes que tienen que ver con el formato de los documentos. El texto en él se crea a partir de etiquetas, también llamadas tags, que permiten interconectar diversos conceptos y formatos.

Para la escritura de este lenguaje, pues, se crean las etiquetas que aparecen especificadas a través de corchetes o paréntesis angulares: < y >. Entre sus componentes, los elementos dan forma a la estructura esencial del lenguaje, ya que tienen dos propiedades (el contenido en sí mismo y sus atributos).

Por otra parte, cabe destacar que el HTML permite ciertos códigos que se conocen como scripts, los cuales brindan instrucciones específicas a los navegadores que se encargan de procesar el lenguaje. Entre los scripts que pueden agregarse, los más conocidos y utilizados son JavaScript y PHP.

VISUAL STUDIO CODE

Visual Studio Code (VS Code) es un editor de código fuente desarrollado por Microsoft. Es software libre y multiplataforma, está disponible para Windows, GNU/Linux y macOS. VS Code tiene una buena integración con Git, cuenta con soporte para depuración de código, y dispone de un sinnúmero de extensiones, que básicamente te da la posibilidad de escribir y ejecutar código en cualquier lenguaje de programación.

Características:

Multiplataforma: Es una característica importante en cualquier aplicación y más si trata de desarrollo. Visual Studio Code está disponible para Windows, GNU/Linux y macOS.

IntelliSense: Esta característica está relacionada con la edición de código, autocompletado y resaltado de sintaxis, lo que permite ser más ágil a la hora de escribir código. Como su nombre lo indica, proporciona sugerencias de código y terminaciones inteligentes en base a los tipos de variables, funciones, etc. Con la ayuda de extensiones se puede personalizar y conseguir un IntelliSense más completo para cualquier lenguaje.

Depuración: Visual Studio Code incluye la función de depuración que ayuda a detectar errores en el código. De esta manera, nos evitamos tener que revisar línea por línea a puro ojo humano para encontrar errores. VS Code también es capaz de detectar pequeños errores de forma automática antes de ejecutar el código o la depuración como tal.

Uso del control de versiones: Visual Studio Code tiene compatibilidad con Git, por lo que puedes revisar diferencias o lo que conocemos con git diff, organizar archivos, realizar commits desde el editor, y hacer push y pull desde cualquier servicio de gestión de código fuente (SMC). Los demás SMC están disponible por medio de extensiones.

Extensiones: Hasta ahora, he mencionado varias veces el término extensiones porque es uno de los puntos fuertes. Visual Studio Code es un editor potente y en gran parte por las extensiones. Las extensiones nos permiten personalizar y agregar funcionalidad adicional de forma modular y aislada. Por ejemplo, para programar en diferentes lenguajes, agregar nuevos temas al editor, y conectar con otros servicios. Realmente las extensiones nos permiten tener una mejor

experiencia, y lo más importante, no afectan en el rendimiento del editor, ya que se ejecutan en procesos independientes.

3. Bosquejo General y de las páginas adicionales

Bosquejo General

NOMBRE EMPESA						
LOGO EMPRESA						
MENU 1	MENU 2	MENU 3	MENU 4	MENU 5	MENU 6	MENU 7
<div>VIDEO</div> <div></div>						
PIE DE PAGINA						

Bosquejo Adicionales

NOMBRE EMPESA						
LOGO EMPRESA						
MENU 1	MENU 2	MENU 3	MENU 4	MENU 5	MENU 6	MENU 7
CONTENIDO				IMAGEN		

NOMBRE EMPESA						
LOGO EMPRESA						
MENU 1	MENU 2	MENU 3	MENU 4	MENU 5	MENU 6	MENU 7
REDES SOCIALES						
FORMULARIO						

4. GITHUN

Link GitHub

<https://github.com/SantiagoTapia25/TAREA-1>

5. Resultados del análisis con el validador de cada una de las páginas

Index.html

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show
☒ source
☐ outline
☐ image report
Options...

Check by
☒ text input
☐ css

```

<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tecnologías Backend</title>
</head>
<body>
  <header>
    <h1>TECNOLOGIAS BACKEND</h1>
    
  </header>
  <header>

```

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

Document checking completed. No errors or warnings to show.

Source

Backend.html

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show ☒ source ☐ outline ☐ image report [Options...](#)

Check by ☒ text input ☐ css

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>BACKEND</title>
</head>
<body>
  <header>
    <h1>TECNOLOGIAS BACKEND</h1>
    
  </header>
  <header>
    <nav>
```

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

Document checking completed. No errors or warnings to show.

Flask.html

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show ☒ source ☐ outline ☐ image report [Options...](#)

Check by ☒ text input ☐ css

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>FLASK</title>
</head>
<body>
  <header>
    <h1>TECNOLOGIAS BACKEND</h1>
    
  </header>
  <header>
    <nav>
```

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

Document checking completed. No errors or warnings to show.

Node.html

display the markup source of the input document.

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show ☒ source ☐ outline ☐ image report [Options...](#)

Check by ☐ css

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>NODEJS</title>
</head>
<body>
  <header>
    <h1>TECNOLOGIAS BACKEND</h1>
    
  </header>
  <header>
    <nav>
```

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

[Message Filtering](#)

Document checking completed. No errors or warnings to show.

Golang.html

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show ☒ source ☐ outline ☐ image report [Options...](#)

Check by ☐ css

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>GOLANG</title>
</head>
<body>
  <header>
    <h1>TECNOLOGIAS BACKEND</h1>
    
  </header>
  <header>
    <nav>
```

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

[Message Filtering](#)

Document checking completed. No errors or warnings to show.

Cursos.html

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show ☒ source ☐ outline ☐ image report

Check by ☐ CSS

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>CURSOS</title>
</head>
<body>
  <header>
    <h1>TECNOLOGIAS BACKEND</h1>
    
  </header>
  <header>
    <nav>
```

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Document checking completed. No errors or warnings to show.

Contactos.html

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show ☒ source ☐ outline ☐ image report

Check by ☐ CSS

```
<datalist id="mydata">
  <option label="Sr" value="Señor">
  <option label="Sra" value="Señora">
  <option label="Srta" value="Señorita">
</datalist> <br> <br>
<form action="https://www.utpl.edu.ec/" method="post" target="_blank">
  <p>Nombre de usuario: <input type="text" name="usuario"></p><br>
  <p>Correo electrónico: <input type="email" name="email" required></p><br>
  <p>Asunto <input type="text" name="usuario"></p><br>
  <p>Tu mensaje: <input type="text" name="usuario"></p><br>
  <p><input type="submit" value="Enviar datos"></p>
</form>
</header>
</body>
</html>
```

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Document checking completed. No errors or warnings to show.

6. Bibliografía

Muñoz, J. (22 de Enero de 2018). *OpenWebinars*. Obtenido de <https://openwebinars.net/blog/crea-tu-primera-aplicacion-en-flask/>

DreamHost. (03 de Agosto de 2021). Obtenido de <https://help.dreamhost.com/hc/es/articles/360043547431-Ejemplos-de-scripts-de-Node-js>

Josue. (s.f.). *LovTechnology*. Obtenido de <https://lovtechnology.com/nodejs-y-golang-cual-elegir/>

Krypton Solid. (s.f.). Obtenido de <https://kryptonsolid.com/estos-ejemplos-de-codigo-de-go-muestran-sus-caracteristicas-principales/>