**CODIFICACIÓN DE MÓDULOS DEL SOFTWARE STAND ALONE, WEB Y MÓVIL DE ACUERDO AL PROYECTO A DESARROLLAR**

**GA7220501096-AA3-EV01**

**FASE- EJECUCIÓN**

**EVIDENCIA #001**

**APRENDIZ:**

**TATIANA SÁNCHEZ CARRERO**

**SERVICIO NACIONAL DE APRENDIZAJE SENA**

**REGIONAL CUNDINAMARCA**

**CENTRO DE LA TECNOLOGÍA Y DISEÑO DE LA PRODUCTIVIDAD EMPRESARIAL**

**MODALIDAD VIRTUAL**

**TECNÓLOGO EN ANÁLISIS Y DESARROLLO DE SOFTWARE**

**FICHA 2758348**

**GIRARDOT**

**2024**

**CODIFICACIÓN DE MÓDULOS DEL SOFTWARE STAND ALONE, WEB Y MÓVIL DE ACUERDO AL PROYECTO A DESARROLLAR**

**GA7220501096-AA3-EV01**

**FASE- EJECUCIÓN**

**EVIDENCIA #001**

**APRENDIZ:**

**TATIANA SÁNCHEZ CARRERO**

**INSTRUCTOR:**

**MILTON IVAN BARBOSA GAONA**

**SERVICIO NACIONAL DE APRENDIZAJE SENA**

**REGIONAL CUNDINAMARCA**

**CENTRO DE LA TECNOLOGÍA Y DISEÑO DE LA PRODUCTIVIDAD EMPRESARIAL**

**MODALIDAD VIRTUAL**

**TECNÓLOGO EN ANÁLISIS Y DESARROLLO DE SOFTWARE**

**FICHA 2758347**

**GIRARDOT**

**2024**

**TABLA DE CONTENIDO**

Pág.

1.[INTRODUCCIÓN 4](#_Toc87213197)

[2. OBJETIVOS 5](#_Toc87213198)

[3. CODIFICACION DEL MODULO………………………………………………………6](#_Toc87213199)

[4. CONCLUSIONES 8](#_Toc87213200)

[BIBLIOGRAFÍA 9](#_Toc87213201)

# **INTRODUCCIÓN**

En este informe de dará a conocer la codificación de módulos de software stand alone de el proyecto de software de la floristería tejulia.

Se enfoca en los objetos que los programadores necesitan manipular, en lugar de centrarse en la lógica necesaria para esa manipulación.

1. **OBJETIVO GENERAL**

OBJETIVOS ESPECÍFICOS

* Implementar el módulo utilizando Spring Boot, asegurando que el código esté bien comentado y siga los estándares de codificación.
* Integración con Base de Datos: Utilizar MySQL Workbench para gestionar la base de datos y asegurar una correcta conexión e interacción desde la aplicación.
* Crear y mantener el proyecto utilizando un sistema de control de versiones (como Git), facilitando la colaboración y seguimiento de cambio

**3.** **CODIFICACIÓN DE MÓDULOS DEL SOFTWARE STAND ALONE, WEB PROYECTO**

React ha sido diseñado desde su inicio para ser adoptado gradualmente, así puedes usar tan poco o mucho de React como necesites. Si quieres una prueba de React, agregar interactividad a una simple página HTML o empezar una aplicación compleja con React, los enlaces de esta sección te ayudarán a empezar. Cientos de grandes empresas de todo el mundo, como Netflix, Airbnb y American Express, utilizan React para crear sus aplicaciones web.

Es una de las librerías más populares de JavaScript para el desarrollo de aplicaciones móviles y web. Creada por Facebook, React contiene una colección de fragmentos de código JavaScript reutilizables utilizados para crear interfaces de usuario (UI) llamadas componentes. Una de las mayores ventajas de usar React es que puedes infundir código HTML conJavaScript. Los usuarios pueden crear una representación de un nodo DOM declarando lafunción Element en React.

Caracteristicas

-Componentes: React es declarativo y basado en componentes. Las páginas web sedividen en pequeños componentes para crear interfaces de usuario. la función decomponentes resulta muy útil cuando se trata de mantener el código mientras se trabajacon proyectos a gran escala.

-JSX: es una extensión JavaScript que indica que el script necesita ser procesado yconvertido a Javascript real. JSX utiliza sintaxis más cortas y sencillas para simplificar lacodificación en comparación con Javascript.

-Vinculación de datos: Vinculación de datos unidireccional junto con una infraestructurade aplicación llamada controles Flux. El flujo de datos unidireccional facilita elrazonamiento sobre una aplicación, y Flux es un patrón que mantiene los datos unidireccionales.

-Una página basada en React.JS consta de un DOM virtual. Para cada objeto DOM,existe una representación (copia) de ese objeto. Objeto DOM

-Puede usar React con otros frameworks como Angular.js, Backbone.js con bastantefacilidad.

-El mantenimiento de React es sencillo gracias a su arquitectura basada encomponentes y a la reutilización de los componentes definidos.

-React puede utilizarse tanto en el servidor como en el cliente, lo que permitedistribuir lacarga de renderizado del servidor al cliente si es necesario

Codigo

<%--

Document : inicio

Created on : 7/09/2024, 10:46:19 p. m.

Author : tatia

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%> **<%--es un recurso que se utiliza en la cabecera HTTP (header) y que le indica al cliente o navegador qué clase de archivo o medio le está enviando el servidor. --%>**

**<%-- responsable de se muestre el contenido del texto correctamente decodificado, sin errores ni caracteres extraños.--%>**

<!DOCTYPE html> **<%--es una instrucción especial que se usa en HTML para poder indicar al navegador que estemos usando, qué tipo de versión de HTML estamos usando.--%>**

<html lang="en"> **<!-- especifica el idioma natural del contenido de una página web -->**

<head> **<%-- es la primera de las dos partes en que se estructura un documento HTML --%>**

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0"> **<%--controla el ancho de la página dependiendo del dispositivo en el que se encuentre el usuario. --%>**

<title>Inicio floristeria Tejulia</title>

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css"> **<%-- define una hoja de estilos preferida o alternativa. --%>**

<style>

body {

margin: 0;

padding: 0;

font-family: Arial, sans-serif;

background-image: url(imagenes/fondo.jpg);

background-size: cover;

background-position: center;

background-attachment: fixed;

color: white;

} **<%--aqui encontramos el cuerpo inicial de la pantalla (fondo) donde se configura el color, la imagen, el tipo de letra y demas. --%>**

#container1 {

width: 100%;

height: 110px;

background-color:rgba(205, 205, 205, 1) ;

display: flex;

justify-content: center;

align-items: center;

} **<%--aqui esta la configuracion de el primer container que es donde va el titulo, el tamaño, el color de la letra.. --%>**

#container1 img {

height: 110px;

margin-left: 40px;

} **<%--este contendor es el encargado del tamaño y margen de la imagen del body--%>**

#container1 h1 {

color: black;

font-weight: bold;

font-size: 20px;

text-align: center;

width: 100%;

}

.spacer {

height: 20px;

}

.container {

width: 50%;

height: 75px;

background-color: rgba(62, 191, 239, 1);

display: flex;

justify-content: center;

align-items: center;

margin: 0 auto;

}

.container button {

background-color: rgba(208, 228, 239, 1);

color: black;

font-weight: bold;

border: none;

width: 300px;

height: 50px;

margin: 0 10px;

border-radius: 5px;

font-size: 18px;

cursor: pointer;

transition: background-color 0.3s ease;

} **<%--este container es el encargado de la interaccion de la pagina, conectandolo con otras --%>**

.container .icon {

font-size: 50px;

margin-right: 10px;

color: black;

}

.container button:hover {

background-color: rgba(214, 169, 229, 1 );

}

</style>

<script>

function redirectToInicioSesion() { **<%--habilita el acceso a las variables de sesión --%>**

window.location.href = "inicio\_sesion.jsp";

}

</script>

</head>

<body>

<div class="container" id="container1"> **<%-- es un contenedor que sirve principalmente para contener a otros elementos HTML. --%>**

<img src="imagenes/logoP.jpg" alt="" />

<h1>Inicio-Floristeria Tejulia</h1>

</div>

<div class="spacer"></div>

<div class="container" id="container2">

<i class="fa fa-home fa-lg icon" aria-hidden="true"></i> **<%--iconos usados en la pag --%>**

<button type="button" onclick="redirectToInicioSesion()">Inicio</button> **<%--es una propiedad del objeto route que puede utilizar para especificar dónde redirigir una ruta. --%>**

</div>

<div class="spacer"></div>

<div class="container" id="container3">

<i class="fa fa-gift fa-lg icon" aria-hidden="true"></i>

<button type="button" onclick="window.location.href='productos.jsp'">Productos</button>

</div>

<div class="spacer"></div>

<div class="container" id="container4">

<i class="fa fa-wrench fa-lg icon" aria-hidden="true"></i>

<button type="button" onclick="window.location.href='servicios.jsp'">Servicios</button>

</div>

<div class="spacer"></div>

<div class="container" id="container5">

<i class="fa fa-phone fa-lg icon" aria-hidden="true"></i>

<button type="button" onclick="window.location.href='contacto.jsp'">Contacto</button>

</div>

<div class="spacer"></div>

<div class="container" id="container6">

<i class="fa fa-user-plus fa-lg icon" aria-hidden="true"></i>

<button type="button" onclick="window.location.href='nosotros.jsp'">Nosotros</button>

</div>

</body>

</html>

1. **CONCLUSIONES**

De acuerdo con lo expuesto, se puede afirmar que React es excelente cuando se trata demanejar funciones pequeñas y sin estado que reciben entradas y devuelven elementos como salida. Cada proyecto en React tiene una arquitectura diferente y una orientación limitada, y por lo tanto, es fácil equivocarse.

**BIBLIOGRAFÍA**

Tomado de:

Material Sena: “Frameworks para construcción de aplicaciones con JAVA.”.