Hito Individual 2º Trimestre Programación

Carlos Hernández

Fase 1

Esta primera fase de diseño y análisis nos debe permitir abordar la implementación de la aplicación web con garantías. Para ello, proponemos explicar con un algoritmo la tarea de publicar una entrada.

Tras definir el algoritmo, puedes utilizar un caso de uso para dejar más clara la funcionalidad.

Para publicar una entrada en un sitio web, se requieren varios pasos que pueden variar dependiendo de la plataforma.

Sin embargo, aquí explico en una serie de pasos cómo puedo utilizar un algoritmo para una guía para publicar una entrada:

1. El usuario inicia sesión en la aplicación.

2. Navega hasta la sección de "Crear nueva entrada".

3. La aplicación presenta un formulario de entrada con campos para el título, contenido y cualquier otra información relevante.

4. El usuario completa el formulario de entrada y hace clic en el botón de "Publicar".

5. La aplicación valida el formulario para comprobar que se cumplen con todos los requisitos.

6. Si la validación es exitosa, la aplicación guarda la entrada en la base de datos.

7. La aplicación redirige al usuario a la página de la entrada recién publicada.

8. Pero si la validación falla, la aplicación mostrará un mensaje de error al usuario indicando qué campos deben modificar para validarlo correctamente.

9. El usuario corrige los campos necesarios y repite el proceso de publicación ,para poder validarlo esta vez de una manera correcta.

Ahora voy a explicar un caso de uso que ejemplifica este algoritmo.

El cliente quiere publicar una nueva entrada en su blog personal. Para hacerlo, inicia sesión en su cuenta en la aplicación web y navega hasta la sección de "Crear nueva entrada". Completa el formulario de entrada con el título, contenido y etiquetas relevantes y hace clic en el botón de "Publicar".

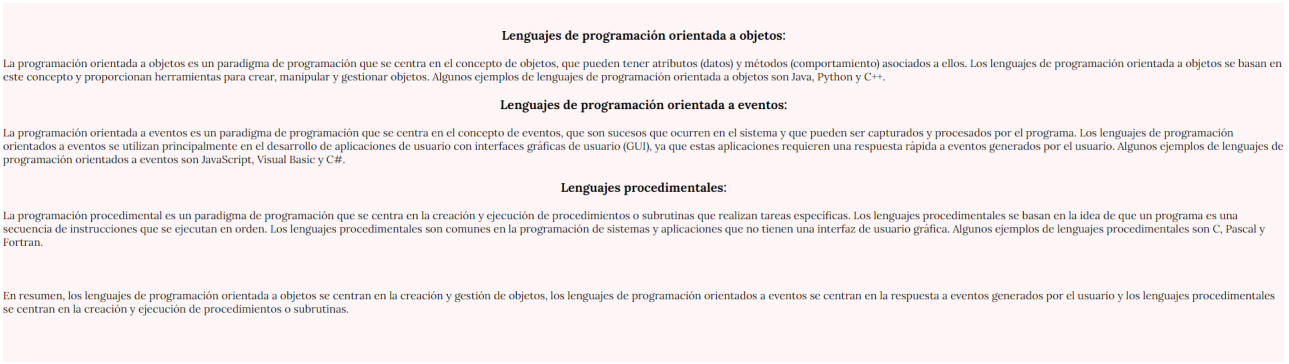
La aplicación valida los campos ingresados por el cliente, por lo que guarda la entrada en la base de datos y le redirige a la página de la entrada recién publicada. El cliente podrá ver su entrada en su blog y compartirla con sus amigos.

Fase 2

En esta segunda fase se realiza la implementación de la aplicación y la construcción de todos los elementos web necesarios para su correcto funcionamiento. Debemos poner especial cuidado en cumplir los estándares de accesibilidad y usabilidad establecidos por W3C, por ejemplo, en lo referente a validación de formularios, confirmación de registro, comunicación y recepción de datos…

Esta ha sido el index que he creado para la web, y en el que he publicado las diferencias entre los diferentes lenguajes de programación:

Cómo vemos en la parte central de la página están publicadas las diferencias entre estos lenguajes (item2):



Este ha sido el código empleado:

```html
<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Hito Individual Programación</title>    <!-- Título proyecto -->

    <!-- Enlazar los css -->
    <link rel="stylesheet" href="styles.css">
    <link href="https://fonts.googleapis.com/css?family=Lora:400,700&display=swap" rel="stylesheet">
    <link rel="shortcut icon" href="favicon.ico" type="image/x-icon">

</head>
<body>

    <header id="header">

    <!--  Menú de navegación -->
    <nav class="menu">
        <div class="logobox">
            <h1 class="logo"><a href="index.html">
               <i class="fas fa-play"></i>
               Hito Individual Programación
            </a></h1>

            <span class="btn-menu"><i class="fas fa-bars"></i></span>
        </div>

    <!--  Botones de inicio -->
        <div class="list-container">
            <ul class="lists">
                <li><a href="index.html" class="active">Inicio</a></li>

                <li><a href="formulario.php">Formulario (Item 3)</a></li>
                <li><a href="#">Tabla (Item 4)</a></li>
                <li><a href="#">Actualizar y eliminar(Item 5)</a></li>
            </ul>
        </div>
    </nav>

    <!-- Img Header -->
    <figure class="imagen11.jpg">
        <div class="welcome">
            <h4>"Encontrarás lo que necesites"</h4>

        </div>
    </figure>
```

```html
<main>


    <!--Texto que aparece en el centro de la página, sección about-us -->
    <section class="about-us">
        <div class="info">
            <h3>DIFERENCIAS ENTRE LENGUAJES DE PROGRAMACIÓN </h3>



        </div>
<br>
<h2> Lenguajes de programación orientada a objetos, a eventos y lenguajes procedimentales.</h2>
<br>
<br>
        <br> <h3>Lenguajes de programación orientada a objetos: </h3>

        <br> <p>La programación orientada a objetos es un paradigma de programación que se centra en el concepto de objetos, que pueden tener atributos (datos) y métodos (comportamiento) as
       <br>
        <br> <h3> Lenguajes de programación orientada a eventos:</h3>
        <br><p>La programación orientada a eventos es un paradigma de programación que se centra en el concepto de eventos, que son sucesos que ocurren en el sistema y que pueden ser captur

        </p>
        <br>
        <h3>Lenguajes procedimentales:</h3>
        <br> <p>
        La programación procedimental es un paradigma de programación que se centra en la creación y ejecución de procedimientos o subrutinas que realizan tareas específicas. Los lenguajes
        </p>

<br>
<br>
<br>
<p>En resumen, los lenguajes de programación orientada a objetos se centran en la creación y gestión de objetos, los lenguajes de programación orientados a eventos se centran en la respuest


    <!-- Footer de la página -->
    <footer>
      <p>&copy; Hito Individual 2ºTrimestre // Carlos Hernández Zabalgo DAM 1</p>
    </footer>

</main>

<!-- Scripts -->
<script src="https://kit.fontawesome.com/35db202371.js" crossorigin="anonymous"></script>


</body>
</html>
```

Y este el CSS:

```css
*{
    margin:0;
    padding:0;
    box-sizing: border-box;
}
body{
    background: hsla(0, 87%, 97%, 0.621);
    font-family: 'Lora', serif;
    overflow-x: hidden; /* color de fondo del index, fuente de letra*/
}

/** Navegation Menu **/
#header{
    position: relative;
    width: 100%; /* controlamos el tamaño de los botones del header*/
}


/** Carcterísticas de la banda negra que contiene todos los botones de enlace **/
.menu{
    position:absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 90px;
    background: #000;
    display: flex;
    justify-content: space-between;
    align-items: center;
    z-index:9999;
}
.btn-menu{display: none;} /*para el display de los botones del menú*/
.menu .logobox{margin-left: 30px;}
.menu .logobox h1 > a{ /*estilos al icono de la web*/
    text-decoration: none;
    color:#fff;
    font-size: 30px;
    font-weight: 400;
}
.menu .logobox h1 > a > i{color:#0091ca; margin-right: 10px;}
.menu .list-container{margin-right: 30px;}
.menu .list-container .lists{display: flex;}
.menu .list-container .lists li{list-style: none;}
.menu .list-container .lists li a{
    text-decoration: none;
    color:#fff;
    font-size: 19px;
    margin:0px 7px;
    padding: 5px;
```

```css
}
.menu .logobox h1 > a > i{color:#0091ca; margin-right: 10px;}
.menu .list-container{margin-right: 30px;}
.menu .list-container .lists{display: flex;}
.menu .list-container .lists li{list-style: none;}
.menu .list-container .lists li a{
    text-decoration: none;
    color:#fff;
    font-size: 19px;
    margin:0px 7px;
    padding: 5px;
    border-radius: 3px;
    transition: 300ms;
}
.menu .list-container .lists li a:hover{color:#10a1e4;} /* dándole un hover para que cuando pasemos el ratón por encima del botón salga en azul*/
.menu .list-container .lists li a.active{color:#10a1e4;}

/** Img header **/


/** Footer **/
footer{ /* editando el color del footer, su altura y anchura, también su margen*/
    width: 100%;
    height: 150px;
    font-size: 20px;
    margin-top: 100px;
    background: #000;

}
footer p{ /*editando el texto del footer, su color, el texto en el centro, etc*/
    text-align: center;
    line-height: 60px;
    color:#fff;
}

h1{ /* editando el texto de encima de cada imagen*/
    text-align: center;
    color: #000000;
    font-size: 50px;
}




h2{
    text-align: center;
}
```

```css
h1{ /* editando el texto de encima de cada imagen*/
    text-align: center;
    color: #000000;
    font-size: 50px;
}




h2{
    text-align: center;
}

h4{
    text-align: center;
}

h3{
    text-align: center;
}
```

Para la parte del **Item 3**, sobre el formulario, creé este formulario en php y lo vinculé con el botón del header Formulario(item3):

Este fue el código php empleado:

```html
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">

    <title>¡Crea tu Blog!</title>
    <link rel="stylesheet" type="text/css" href="estilo.css">
</head>
<body>
    <form action="db_hitoindividual2.php" method="post">
        <h1>¡Crea tu Blog!</h1>
        <p>Título del blog: </p>
        <input type="text" name="name" placeholder="Título del Blog">
        <p>Tu Email: </p>
        <input type="email" name="email" placeholder="Email">
        <p>Contenido del blog: </p>
        <input type="text" name="name" placeholder="Contenido del Blog">
        <p>Fecha de publicación: </p>
        <input type="date" name="name" placeholder="Fecha de publicación">
        <p>Imagen del blog: </p>
        <input type="text" name="name" placeholder="Imagen del Blog">
        <br>
        <br><input type="submit" name="register">
    </form>
        <?php
        include("db_hitoindividual2.php");
        ?>
</body>
</html>
```

No he sabido cómo implementarlo con la base de de datos que he creado, me han surgido bastantes problemas que no he podido resolver.

Esta es la base de datos que creé:

☐	1	ID 🔑	int(11)		No	*Ninguna*	AUTO_INCREMENT	Cambiar	Eliminar	Más
☐	2	título	varchar(100)	utf8mb4_general_ci	No	*Ninguna*		Cambiar	Eliminar	Más
☐	3	email	varchar(100)	utf8mb4_general_ci	No	*Ninguna*		Cambiar	Eliminar	Más
☐	4	contenido	varchar(100)	utf8mb4_general_ci	No	*Ninguna*		Cambiar	Eliminar	Más
☐	5	fecha	date		No	*Ninguna*		Cambiar	Eliminar	Más
☐	6	imagen	varchar(100)	utf8mb4_general_ci	No	*Ninguna*		Cambiar	Eliminar	Más

Para el **Item 4**, mostrar los datos en una tabla, tampoco lo pude enlazar de una manera correcta, pero intenté realizar este código:

```php
<?php

$título=$_POST['título'];
$email=$_POST['email'];
$contenido=$_POST['contenido'];
$fecha=$_POST['fecha']
$imagen=$_POST['imagen']
$consulta="INSERT INTO `formulario` (`ID`, `título`, `email`, `contenido`, `fecha`, `imagen`) VALUES (NULL, ?, ?, ?, NOW(), ?);"


$conn= new PDO('mysql:dbname=test;host=localhost','root','');

$insertar=$conn->prepare($consulta);
$resultado=$insertar->execute([$título, $email, $contenido, $fecha, $imagen ]);

echo("<p> Se ha realizado correctamente el blog");


?>
```

Para el **Item 5,** no pude realizar ningún código porque no podía mostrar los datos de ninguna manera, así que nunca podría actualizar o eliminar datos si no podía escribirlos.

Fase 3

Para finalizar, realizamos las pruebas pertinentes de la aplicación y control de excepciones.

Validación de campos y añadir seguridad en las comunicaciones. Debemos evaluar la implementación de la aplicación en base a la petición recibida inicialmente por el cliente. Es muy importante realizar un “feedback” de qué problemas nos han surgido a lo largo del desarrollo y cómo los hemos resuelto.

Supe realizar el index de la web e implementar los botones pedidos en el Header, tuve muchos problemas con php para enlazar correctamente el código con la Base de datos creada que los mejoraré en la recuperación.

Las diferencias fueron bastante sencillas, fui mirando en los apuntes de la 1era y 2ª evaluación y las diferencié correctamente.

Lo que debo mejorar es: enlazar bien el formulario PHP con la DB.

Carlos Hernández