



HITO INDIVIDUAL PROGRAMACIÓN

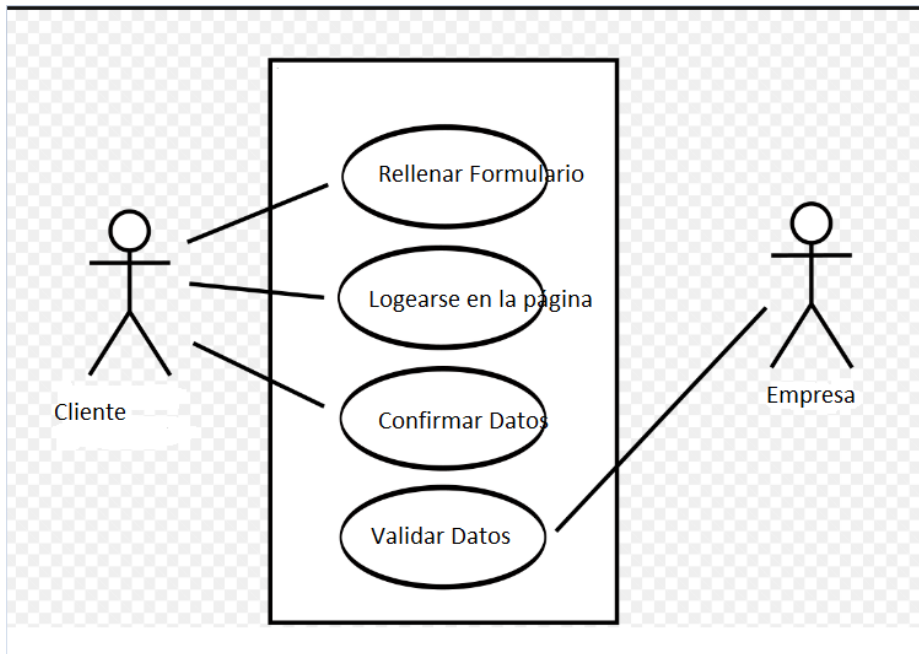
Adriel Sadia Mora



CAMPUS FP

Fase 1.

Lo primero sería entender al cliente para que pueda navegar fácil y en un entorno seguro. Crear la página web y conectarla a Apache desde la app XAMPP. Una vez hecho esto, almacenar la ip y la cookie del cliente en nuestra página y una vez lo tenga que haga el informe, lo envíe y cuando esté todo correcto, se guarden en nuestra base de datos los datos introducidos por nuestro cliente.



Explica correctamente cómo pasar el algoritmo a la implementación de la aplicación tanto en la parte visual como en la parte programática.

Se conecta mediante hilos para implementar la base de datos al html y css que esa es la parte visual de la página web. Mediante el código puesto escribes lo necesario para que funcione la página web o la aplicación.

Fase 2.

En esta fase voy a explicar un poco los archivos de mi hito.

El index.php es la página de inicio para el cliente hecho con html y vinculado a un css para que se vea agradable. También está vinculado a formulario.php y datos.php para que puedan acceder al formulario y poner todos los datos y a datos para ver todos los datos registrados en la base de datos.

En la página de inicio se pide que hablemos sobre los tipos de lenguajes de programación, y eso hice.

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="UTF-8">
5 <title>Tipos de lenguajes de programación</title>
6 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
7 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
8 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/2.9.3/umd/popper.min.js"></script>
9 <script src="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/5.0.2/js/bootstrap.min.js"></script>
10 <link rel="stylesheet" href="style.css">
11
12 </head>
13 <body>
14 <div id="header-nav">
15 <nav class="navbar navbar-expand-lg navbar-dark">
16 <a class="navbar-brand" href="#">Tipos de lenguajes de programación</a>
17 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavDropdown" aria-controls="navbarNavDropdown"
18 <span class="navbar-toggler-icon"></span>
19 </button>
20 <div class="collapse navbar-collapse justify-content-end" id="navbarNavDropdown">
21 <ul class="navbar-nav">
22 <li class="nav-item">
23 <a class="nav-link" href="formulario.php">Formulario</a>
24 </li>
25 <li class="nav-item">
26 <a class="nav-link" href="datos.php">Todos los datos</a>
27 </li>
28 </ul>
29 </div>
30 </nav>
31 </div>
32 </header>
33 <main>
34 <section id="1">
35 <h2>Programación Orientada a Objetos POO</h2>
36 <p>La programación orientada a objetos se centra en los objetos que los programadores quieren manipular en lugar de enfocarse en la
37 <p>Algunos ejemplos de lenguajes de programación POO son Python y Java. Con POO se pueden hacer herencias, polimorfismos y encapsulación.
38 </p>
39 </section>
40 <section id="2">
41 <h2>Lenguajes procedimentales</h2>
42 <p>Los lenguajes de programación procedimentales son aquellos que se centran en la secuencia de instrucciones que un programa debe seguir.
43 <p>En un lenguaje procedimental, se definen los procedimientos que deben realizarse para llevar a cabo una tarea. Estos procedimientos se definen en
44 <p>Algunos ejemplos de lenguajes de programación procedimentales incluyen C, Pascal, Fortran y Basic.</p>
45 </section>
46 <section id="3">
47 <h2>Lenguaje orientado a eventos</h2>
48 <p>Los programas en lenguajes orientados a eventos están compuestos por diferentes componentes que interactúan entre sí enviándose eventos.
49 <p>Uno de los beneficios clave de los lenguajes orientados a eventos es que permiten la programación asíncrona, lo que significa que se puede hacer
50 </p>
51 </section>
52 </main>
53 </body>
54 </html>
```

Así quedaría:

Programación Orientada a Objetos POO

La programación orientada a objetos se centra en los objetos que los programadores quieren manipular en lugar de enfocarse en la lógica necesaria para manipularlos. Este enfoque de programación está hecho para programas que son grandes, difíciles y se actualizan diariamente. La organización de un programa orientado a objetos también hace que el método sea beneficioso para el desarrollo colaborativo, donde los proyectos se dividen en grupos.

Algunos ejemplos de lenguajes de programación POO son Python y Java. Con POO se pueden hacer herencias, polimorfismos y encapsulaciones

Lenguajes procedimentales

Los lenguajes de programación procedimentales son aquellos que se centran en la secuencia de instrucciones que un programa debe seguir para resolver un problema específico. Estos lenguajes se basan en la idea de que el programa es una secuencia de pasos o procedimientos que se ejecutan en un orden determinado.

En un lenguaje procedimental, se definen los procedimientos que deben realizarse para llevar a cabo una tarea. Estos procedimientos pueden tomar argumentos y devolver valores, y pueden ser llamados desde otros procedimientos en el programa. Los procedimientos se definen utilizando estructuras de control de flujo, como bucles y condicionales.

Algunos ejemplos de lenguajes de programación procedimentales incluyen C, Pascal, Fortran y Basic.

Lenguaje orientado a eventos

Los programas en lenguajes orientados a eventos están compuestos por diferentes componentes que interactúan entre sí enviándose eventos. Estos componentes pueden ser objetos, funciones o módulos que se comunican a través de eventos que son enviados y recibidos. Cada componente se suscribe a los eventos que le interesan, y cuando ocurren, el componente responde en consecuencia.

Uno de los beneficios clave de los lenguajes orientados a eventos es que permiten la programación asíncrona, lo que significa que las tareas pueden ejecutarse en segundo plano mientras se espera la respuesta de un dispositivo principal. Esto

En el formulario.php he creado un formulario en el que el cliente introduce su correo electrónico, título, contenido, fecha de publicación y una imagen. Les he puesto a todos required para que cuando envíe el cliente el formulario aparezcan todos los datos. Conectado a la base de datos con PDO y para que se guarden los datos a la base de datos está conectado con un INSERT INTO. Después de enviar los datos, este te lleva a autentificacion.php para el login del cliente.

```

HITO INDIVIDUAL
  hito
  autentificacion.php
  datos.php
  formulario.php
  index.php
  # style.css
  # style2.css
  # style3.css

hito > formulario.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Formulario</title>
5 <link rel="stylesheet" type="text/css" href="style2.css">
6 </head>
7 <body>
8 <h1>Formulario</h1>
9 <form action="formulario.php" method="post" enctype="multipart/form-data">
10 <label for="author_email">Email del cliente:</label>
11 <input type="email" name="correo" required>
12
13 <label for="title">Título:</label>
14 <input type="text" name="titulo" required>
15
16 <label for="content">Contenido:</label>
17 <textarea name="contenido" required></textarea>
18
19 <label for="publish_date">Fecha de publicación:</label>
20 <input type="date" name="fechapubli" required>
21
22 <label for="image">Imagen:</label>
23 <input type="file" name="imagen" required>
24
25 <input type="submit" value="Enviar">
26 <?php
27 if(isset($_POST['correo']) && isset($_POST['titulo']) && isset($_POST['contenido']))){
28 // $ID_FORMULARIO=$_POST['ID_FORMULARIO'];
29 $email=$_POST['correo'];
30 $titulo=$_POST['titulo'];
31 $fechapublicacion=$_POST['fechapubli'];
32 $imagen=$_POST['imagen'];
33 $contenido=$_POST['contenido'];
34
35 $conn = new PDO('mysql:host=localhost;dbname=test', 'root', '');
36 $insertar= "INSERT INTO 'hitoprogramacion' ('correo', 'titulo', 'fechapubli', 'contenido', 'imagen') VALUES ('$email', '$titulo', '$fechapublicacion', '$contenido', '$imagen');";
37 $registros = $conn->exec($insertar);
38 header("location:autentificacion.php");
39 }
40 >
41 </form>
42 </body>
43 </html>

```

Así sería el resultado:

Formulario

Email del cliente:

Título:

Contenido:

Fecha de publicación:

dd/mm/aaaa

Imagen:

Seleccionar archivo

Ninguno archivo selec.

Enviar

En el autentificacion.php te lleva a un login para que pongas tu correo electrónico y tu contraseña. Una vez puesto, saldrá un mensaje que has iniciado sesión con su ID:
XXXXXXXXXX

```

HITO INDIVIDUAL
  hito
    autentificacion.php
    datos.php
    formulario.php
    index.php
    # style.css
    # style2.css
    # style3.css

hitos > autentificacion.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta http-equiv="X-UA-Compatible" content="IE=edge">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <link rel="stylesheet" href="style3.css">
8    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/5.3.2/css/bootstrap.min.css" integrity="sha384-TC2WJ8H8VQ343M0YU9t8A80y8X4rUwT8zF+v6Ug6eH2C8F0xwv8m8V7wU">
9    <title>Validación</title>
10 </head>
11 <body>
12   <h1>Validación de datos</h1>
13   <div class="container mt-5">
14     <form method="post">
15       <div class="mb-3">
16         <label for="email" class="form-label">Dirección de correo electrónico</label>
17         <input type="email" class="form-control" id="email" name="email">
18       </div>
19       <div class="mb-3">
20         <label for="contraseña" class="form-label">Contraseña</label>
21         <input type="password" class="form-control" id="contraseña" name="contraseña">
22       </div>
23       <button type="submit" class="btn btn-primary">Enviar</button>
24     </form>
25
26     <?php
27     if (isset($_POST['email']) && isset($_POST['contraseña'])) {
28       session_start();
29       $token = session_id();
30       echo '<div class="alert alert-success mt-3" role="alert">Ha iniciado sesión con su ID: ' . $token . '</div>';
31     }
32     >
33
34   </div>
35   <script src="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/5.3.2/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrKf7Ew8Y60zvdyktXWl5GEFLZywESLm4Ik518A4Tdy9Ih318414Y">
36 </body>
37 </html>

```

Este sería el resultado:

Validación de datos

Dirección de correo electrónico

Contraseña

Enviar

Dirección de correo electrónico

Contraseña

Enviar

Ha iniciado sesión con su ID: uhh86oq940dd0q5eqnavr4l0rb

Por último, en el archivo datos.php se pueden ver todos los datos introducidos por el cliente en nuestra base de datos ya sean todas las imágenes, correos, etc...













```

EXPLORER  ...  datos.php X  # style.css  # style2.css  # style3.css  formulario.php  autentificacion.php  index.php
HITO INDIVIDUAL
hito > datos.php
6 <meta name="viewport" content="width=device-width, initial-scale=1.0">
7 <title>Consultas</title>
8 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
9 </head>
10 <body>
11 <div class="container">
12 <h1>Datos</h1>
13 <table class="table table-striped">
14 <thead>
15 <tr>
16 <th>Email</th>
17 <th>Título</th>
18 <th>Fecha de publicación</th>
19 <th>Imagen</th>
20 <th>Contenido</th>
21 <th>Acciones</th>
22 </tr>
23 </thead>
24 <tbody>
25 <?php
26 $servername = "localhost";
27 $username = "root";
28 $password = "";
29 $dbname = "test";
30
31 $conn = mysqli_connect($servername, $username, $password, $dbname);
32
33 if (!$conn) {
34     die("Error al conectar a la base de datos: " . mysqli_connect_error());
35 }
36
37 $sql = "SELECT * FROM hitoprogramacion";
38 $result = mysqli_query($conn, $sql);
39 // var_dump($sql)
40
41 while ($row = mysqli_fetch_assoc($result)) {
42     echo "<tr>";
43     echo "<td>" . $row["correo"] . "</td>";
44     echo "<td>" . $row["titulo"] . "</td>";
45     echo "<td>" . $row["fechapubli"] . "</td>";
46     echo "<td><img src='uploads/' . $row[\"imagen\"] . \"' height='100'></td>";
47     echo "<td>" . $row["contenido"] . "</td>";
48     echo "<td>";
49     echo "<a class='btn btn-danger disabled' href='#'>Eliminar</a>";
50     echo "<a class='btn btn-primary disabled' href='#'>Actualizar</a>";
51     echo "</td>";
52     echo "</tr>";

```

Así quedaría:

Datos

Email	Título	Fecha de publicación	Imagen	Contenido	Acciones
adriel@gmail.com	ABCD	2023-04-12 09:42:17		EFGH	Eliminar Actualizar
adriel@campusfp.es	Adriel	2023-04-12 09:47:08		ABCD	Eliminar Actualizar
adriel@campusfp.es	Adriel	2023-04-12 09:49:06		ABCD	Eliminar Actualizar
adriel@campusfp.es	Adriel	2023-04-12 09:51:07		ABCD	Eliminar Actualizar
antonio@antonio.com	Hola	2023-04-05 10:25:59		235235	Eliminar Actualizar
pepe@gmail.com	0q830qgf	2023-04-08 00:00:00		werhwh	Eliminar Actualizar
antonio@antonio.com	w4wrhg	2023-03-31 00:00:00		wregw4h	Eliminar Actualizar
alvaro@alvaro.com	sas	2023-04-01 00:00:00		asas	Eliminar Actualizar
jorge@jorge.com	212	2023-04-10 00:00:00		1212	Eliminar Actualizar
carlos@carlos.com	11234	2023-04-01 00:00:00		qw123	Eliminar Actualizar
javiernieto@campusfp.es	asas	2023-03-31 00:00:00		asas	Eliminar Actualizar
adriel cadia@gmail.com	ae	2023-04-08 00:00:00		aca	Eliminar Actualizar

También añadí al código tres archivos de css para index.php, formulario.php y autenticación.php . Para poder cambiar el background la forma y el color del fondo de mis página web y la fuente de las letras.

```
# style.css X
hito > # style.css > body
1  body {
2      font-family: 'Times New Roman', Times, serif, sans-serif;
3      margin: 0;
4      padding: 0;
5      box-sizing: border-box;
6  }
7
8  #header-nav {
9      background-color: #cca010;
10     color: #661919;
11     padding: 1em;
12 }
13
14 .navbar-brand {
15     font-size: 2em;
16 }
17
18 .nav-link {
19     color: #fff !important;
20     font-size: 1.2em;
21     margin-left: 1em;
22 }
```

```
# style2.css X
hito > # style2.css > body
1  body {
2      font-family: Arial, sans-serif;
3      background-color: #f2f2f2;
4      margin: 0;
5      padding: 0;
6  }
7
8  h1 {
9      text-align: center;
10 }
11
12 form {
13     background-color: rgba(202, 41, 20, 0.562);
14     border-radius: 10px;
15     box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
16     width: 80%;
17     max-width: 600px;
18     margin: 40px auto;
19     padding: 40px;
20 }
21
```



```
# style3.css x
hito > # style3.css > body
1  body {
2      font-family: Arial, sans-serif;
3      background-color: #f5f5f5;
4      color: #333;
5  }
6
7      .container {
8          max-width: 800px;
9          margin: auto;
10         padding: 20px;
11         background-color: #fff;
12         box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
13     }
14
15     h1 {
16         text-align: center;
17         font-size: 2rem;
18         margin-bottom: 30px;
19         color: #2e2828;
20     }
21
```

Fase 3.

Aquí he utilizado la aplicación XAMPP y Visual Studio. Lenguajes de programación que he utilizado: PHP y HTML. También he utilizado mysql para la base de datos desde localhost > PHPmyAdmin.

¿Para qué sirve XAMPP?

XAMPP permite a los desarrolladores web crear y probar aplicaciones web dinámicas utilizando tecnologías como PHP, MySQL y Apache en su propia computadora, sin tener que conectarse a un servidor web remoto. Además de estos componentes principales, XAMPP también incluye otros software útiles, como el servidor FTP FileZilla y el servidor de correo Mercury.

Es una herramienta para crear y probar aplicaciones web antes de implementarlas en un servidor remoto en línea.



Y para programar he utilizado VisualStudio Code. Es una herramienta multiplataforma que admite múltiples lenguajes de programación y frameworks, incluyendo JavaScript, TypeScript, Python, C#, Java, Ruby y muchos más.



Para mi es super bueno Visual Studio ya que puedo programar en muchos lenguajes de programación distintos fácil y rápido. Y el XAMPP lo utilizo para cuando tenga el código creado conectarlo a la base de datos en PHPmyAdmin.

He tenido algunos problemas pero no han sido cosa que no se pueda solucionar. Fallos como escribir mal el nombre de la base de datos y no se me conecte, tener que volver a poner el Insert Into porque no me cargaban los datos en la base de datos. Hay una cosa que no he podido solucionar y es con la foto, ya que se ve que mandas una foto pero se debería de ver el nombre de la foto. He estado bastante tiempo intentando ver el problema pero no lo he encontrado. También tuve un problema con el token osea la cookie pero encontré la solución por Internet para poder sacarla.

Para poder mejorar mi página lo que yo haría sería mejorar un poco más lo visual ya que como tal el código lo veo bien. Poner que se vea la imagen en sí, tener una navegación mejor al implementar más código, etc... Y mejorar en el apartado de Datos lo de eliminar datos de la base de datos para que funcione.