

WEAPON KKEY

Manual Técnico

Proyecto: Sitio web de compra de skin

MATERIA

Modelos y Sistemas

E.E.S.T N°4 De Berazategui

7mo 3ra

Integrantes

Basualdo Alan, González Marcos, Guerra Luciano,
Gutiérrez Tobías, Alejo García, Carranza Braian

WeaponKkey es un sitio web que se encarga de vender skin de juegos, lo que tiene de diferente nuestra página es la especificación de la necesidad del usuario, es decir, no hará falta que compre cajas como en cualquier otra página de venta de skin, acá encontraras lo que necesites a un buen precio.

La página principal esta diseñada en varias secciones, cómo, por ejemplo:

- Header: es el menú que esta distribuida por links donde nos mandará a las secciones de la página y al Login que estará dirigida a otra página. La mayoría de los links son para ubicarlos más debajo de la página, así que en el “href” pondremos un “#” con el nombre de la sección, como se muestra en la imagen:

```
<!-- Header -->
<section id="header">
  <div class="header_container">
    <div class="nav-bar">
      <div class="brand">
        <a href="#hero">
          <h1><span>W</span>eapon <span>K</span>key</h1>
        </a>
      </div>
      <div class="nav-list">
        <div class="hamburger">
          <div class="bar"></div>
        </div>
        <ul>
          <li><a href="#hero" data-after="Hogar">Hogar</a></li>
          <li><a href="#services" data-after="Servicios">Servicios</a></li>
          <li><a href="#projects" data-after="Juegos">Juegos</a></li>
          <li><a href="#about" data-after="About">About</a></li>
          <li><a href="#contact" data-after="Contacto">Contacto</a></li>
          <li><a href="/php-login/login.php" data-after="Login">Iniciar sesion</a></li>
        </ul>
      </div>
    </div>
  </div>
</section>
<!-- End Header -->
```

Aparecen muchos “div” con “clases” en el cual mas adelante con el CSS le daremos una forma para que el menú cuando se achique la pantalla se muestre de otra manera.

- Service action: en esta sección de la página hicimos fue agregarle imágenes mediante una URL que sacamos de una página, ósea que no tuvimos que descargar la imagen, solamente copiar la URL de la misma. Luego le seguimos agregando textos para complementar esta parte de la página.
- Projects Section: en esta parte lo que hicimos es agregar las etiquetas del “h1” para ponerle el titulo a esta sección, “h2” para los subtítulos especificando para que tecnología se tenía que utilizar y la etiqueta “p” para agregar textos complementarios y rellenar la idea que se intenta decir. Luego en este caso descargamos una imagen y le pusimos en el codigo la URL de que carpeta la tenemos guardada con su tipo de formato, que puede ser: .JPG, .PNG, etc.
- Contact Section: en esta sección de la pagina hicimos que los títulos (etiqueta “h1”) las principales formas de comunicación hacia nosotros luego con la etiqueta “h2” pusimos la forma de contactarnos. En este caso utilizamos iconos donde tuvimos que sacarlos de Google, fuimos a una pagina de iconos y copiamos la “clase” de la misma y la copiamos en el “div” de nuestro codigo, para que pueda aparecer como se muestra en la imagen:

```

<div class="contact-item">
  <div class="icon"></div>
  <div class="contact-info">
    <h1>Email</h1>
    <h2>WeaponKkey@gmail.com</h2>
  </div>
</div>

```

WeaponKkey no solamente tiene la función de verse de una manera, ya que lo que hicimos fue que nuestra página sea responsiva, para que esto suceda lo agregamos desde el CSS. Con la función de “keyframes” y “media” logramos que la página cuando se reduzca su tamaño que sus partes cambian de posición o que el menú se transforme de otra manera. Para el tamaño de algunos botones o títulos utilizamos el valor “rem” ya que nos hace escribir y acordarnos de una manera más fácil el tamaño de ciertas secciones. También es por la parte de que utilizamos valores bastante grandes y que queremos hacer que los cambios se hagan más rápidos y sencillos, si utilizamos el valor de “px” (píxeles) Tendremos que cambiar constantemente sus valores e ir probando varias veces.

Algunos de los ejemplos de nuestro código:

```

/* Media Query For Desktop */
@media only screen and (min-width: 1200px) {
  /* header */
  #header .hamburger {
    display: none;
  }
  #header .nav-list ul {
    position: initial;
    display: block;
    height: auto;
    width: fit-content;
    background-color: transparent;
  }
  #header .nav-list ul li {
    display: inline-block;
  }
  #header .nav-list ul li a {
    font-size: 1.8rem;
  }
  #header .nav-list ul a:after {
    display: none;
  }
}

```

Base de datos:

Hicimos una base de datos llamada “php_login_database” donde también creamos una tabla “users” que contendrá nueve campos, todo esto creado por phpmyadmin. El campo de “id” hicimos que se autoincrementa a medida que los usuarios se agreguen, ya que así podremos identificar que número de usuario es.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/> 1	id	int(11)			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/> 2	nombre	varchar(200)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 3	apellido	varchar(200)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 4	nacionalidad	varchar(200)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 5	dni	varchar(200)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 6	email	varchar(250)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 7	usuario	varchar(250)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 8	imagen	varchar(250)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 9	password	varchar(250)	utf8mb4_general_ci		No	Ninguna			Cambiar Eliminar Más

Como se puede ver en la imagen en los valores utilizamos la opción de “varchar” en todos, ya que así se nos facilitaría la manera de ingresar los datos a la base de datos. En el único caso que no usamos el varchar fue en el “id” ya que necesitábamos un número fijo y que se incrementa.

Registro:

Antes de poder hacer el registro de los usuarios hacia la base de datos, hicimos una conexión con la base de datos, creamos un archivo llamado “database.php” en donde especificamos el servidor, usuario, contraseña y la base de datos que vamos a utilizar, que en este caso es “php_login_database”.

Luego renombramos ese archivo en el archivo de “signuo.php” para hacer los registros. Luego hicimos es escribir todos los campos que queremos que llene el usuario (nombre, apellido, nacionalidad, etc.) mas adelante lo que hicimos es que tocando el logo de WeaponKkey es que te vuelva al inicio, lo que hicimos fue un “a href=“Dirección del archivo”>” y adentro colocarle el “h1” con el título de WeaponKkey.

Después de todo esto hicimos un formulario que cuando se termine de registrar el usuario nos lleve a la misma página. Casi todos tienen tipo “text” ya que así se podrá mostrar lo que escribe el usuario, en cambio en los campos de contraseña y confirmar contraseña no lo tendrán, ya que esos mismos se mostraran en puntos, ósea que no son visible para el usuario. Para que los campos del formulario muestren que tienen que introducir el usuario hicimos un “placeholder” que es para escribir adentro del campo que datos tiene que ingresar. Imagen del código:

```
<form action="signup.php" method="POST">
  <input name="nombre" type="text" placeholder="Introduce tu nombre">
  <input name="apellido" type="text" placeholder="Introduce tu apellido">
  <input name="nacionalidad" type="text" placeholder="Introduce de que país eres">
  <input name="dni" type="text" placeholder="Introduce tu dni">
  <input name="email" type="text" placeholder="Introduce tu email">
  <input name="usuario" type="text" placeholder="Introduce tu nombre de usuario">
  <input name="password" type="password" placeholder="Introduce tu contraseña">
  <input name="confirm_password" type="password" placeholder="Confirme su contraseña">
  <input type="submit" value="Enviar">
</form>
```

Login:

Como en el registro tuvimos que adquirir el archivo de la base de datos que es “database.php” para poder acceder a la base de datos que queremos usar. Luego agregamos un “if” que dice que si la contraseña y el email son correctos los dejara acceder a la siguiente página y agregamos un “else” que diga que si no cumple esa condición nos mostrara un mensaje que diga que los datos no coinciden.

Volvimos hacer un formulario, pero en este caso solo agregamos dos campos, uno para el email, el otro para la contraseña y un botón donde se podrán enviar los datos. Para que el botón tenga un texto tuvimos que agregar dentro del “input” un “value=” para poder escribirlo y después del “=” podemos agregarle lo que queremos que muestre.

```
<form action="login.php" method="POST">
  <input name="email" type="text" placeholder="Introduce tu email">
  <input name="password" type="password" placeholder="Introduce tu contraseña">
  <input type="submit" value="Enviar">
</form>
```

Al momento que los datos coinciden con la base de datos se cumplirá la función, así que nos mandara al destino ya estipulado por la función.

Le damos la bienvenida al usuario usando una variable “users” y el campo usuario de la base de datos, para poder especificar dependiendo de que usuario ingrese. Luego hicimos varios links que nos mande a otra carpeta que tenemos creada específicamente para los usuarios y así poder organizar nuestros archivos y proyectos.

```

<?php if(!empty($user)): ?>
    <br> Bienvenido <?= $user['usuario'];?>

    <br>Has iniciado sesion correctamente

    <br>
    <br> Presione este boton si quiere acceder para comprar los productos
    <br> <a href="/Fil/index.php">Compras</a>

    <br>
    <br>
    <br> Presione este boton si quiere acceder a los detalles de su cuenta
    <br> <a href="/usuario/index.php">Cuenta</a>

    <br>
    <br>
    <br> Si quieres cerrar la cuenta presiones el enlace para cerrar se
    <br>
    <a href="logout.php">
        Cerrar sesion
    </a>

```

Compras: para esta parte del proyecto hicimos que agregar un menú lateral para los productos, es decir que la categorización este en el lado izquierdo de la página, por eso agregamos dentro la etiqueta “a” pondremos la clase “category” para que cuando estemos haciendo los productos agreguemos el tipo de producto, así tendremos una separación de las skins. Todo esto esta dentro de un “div class=“category_list””.

Después agregamos una etiqueta “section” donde pondremos todos los productos con el nombre, foto, etc.

```

<section class="products-list">
    <!--Francotiradores-->
    <div class="product-item" category="francotiradores">
        
        <a href="#">SCAR-20</a>
    </div>
    <div class="product-item" category="francotiradores">
        
        <a href="#">G3SG1</a>
    </div>
    <div class="product-item" category="francotiradores">
        
        <a href="#">SSG 08</a>
    </div>
    <div class="product-item" category="francotiradores">
        
        <a href="#">AWP Camouflage</a>
    </div>

```

Como se puede ver en la imagen tenemos la clase “category” que en este caso le asignamos “francotiradores”. Luego esta la etiqueta para agregar la imagen, que tenemos una carpeta con todas las fotos descargadas, así que solamente tendríamos que poner el nombre de la carpeta y el nombre del archivo con su extensión, que puede ser .JPG .PNG, etc. Luego mas abajo encontraremos el nombre del producto con un enlace que por el momento no nos lleva a ningún lado, que esta dentro de la etiqueta “a”.

Este es un ejemplo con la sección de los francotiradores, pero también se podrá encontrar lo mismo con los demás productos.

Cuenta: Usamos muchas cosas de la página de inicio, pero con la diferencia de que pusimos el nombre de la cuenta del usuario, con la variable “users[‘usuario’]” para poder llamar ese campo de la base de datos.

En este caso lo que hicimos fue la reutilización del código para el menú, para poder ahorrar espacio en las líneas de código y no estar copiando línea por línea cada vez que queramos usar el menú. Creamos una carpeta “partes” donde adentro tendremos un archivo “header.php” que contiene todo el menú ya echo. Para poder usar ese menú en los archivos tenemos que crear una sentencia <?php ?> y ahí adentro poner “include ‘partes/header.php’;”

```
<?php
| include 'partes/header.php';
?>
```

- Información del usuario: para poder mostrar todos los datos que el usuario tiene en la base de datos y que anteriormente había agregado en el registro lo que tuvimos que hacer es que usar la conexión de base de datos, con el archivo “database.php” y pedir todos los campos que queremos saber.

```
if (isset($_SESSION['user_id'])) {
    $records = $conn->prepare('SELECT id, nombre, apellido, nacionalidad, dni, email, usuario, imagen, password FROM users');
    $records->bindParam(':id', $_SESSION['user_id']);
    $records->execute();
    $results = $records->fetch(PDO::FETCH_ASSOC);
```

Después de eso usamos la variable “\$users” que en otro archivo usamos para poder llamar al campo del nombre de usuario, ahora haremos lo mismo, pero con la diferencia que llamaremos a todos los campos de la base de datos, como la nacionalidad, el id, el email, etc.

```
<?php if(!empty($user)):?>
    ID: <?= $user['id'];?>
    <br>
    Nombre: <?= $user['nombre'];?>
    <br>
    Apellido: <?= $user['apellido'];?>
    <br>
    Nacionalidad: <?= $user['nacionalidad'];?>
    <br>
    Dni: <?= $user['dni'];?>
    <br>
    EMAIL: <?= $user['email'];?>
    <br>
    NOMBRE DE USUARIO: <?= $user['usuario'];?>
```

- Foto de perfil: para poder subir una imagen a la página tuvimos que agregar un campo a la base de datos llamada “imagen” para que en el campo salga la URL de la imagen donde esta ubicada la imagen. Hicimos un formulario con dos botones, uno para subir la foto y otro botón para subir esa foto a la base de datos y a la página. Todo esto con la etiqueta “form” y dentro de ella con dos “input” como se muestra en la imagen:

```
<form method="POST" action="" enctype="multipart/form-data">
  <input type="file" name="imagen">
  <input type="submit" name="subir" value="subir">
</form>
```

Creamos una carpeta llamada “fotoperfil” donde estarán ubicadas las fotos de los usuarios, usamos la función “if” que si pulsan el botón “subir” que se mande a la carpeta que anteriormente mencionamos con el nombre del “id” del usuario y usamos una extensión determinada por nosotros, que será .JPG. Una vez que todo esto pase que se agregue a la base de datos con la carpeta “fotoperfil” el numero de id del usuario y la extensión

```
<?php
if (isset($_POST['subir'])) {
    $ruta = "fotosperfil/";
    $fichero = $ruta.basename($_FILES['imagen']['name']);
    $rut = $ruta.$_SESSION['user_id'].".jpg";
    if(move_uploaded_file($_FILES['imagen']['tmp_name'], $ruta.$_SESSION['user_id'].".jpg")){
        require("database.php");

        $insertar = $conn->query("UPDATE users SET imagen = '$rut' WHERE id = '".$_SESSION['user_id']."'");
        header("Location: cuenta.php?user_id=".$_SESSION['user_id']."");
    }
}
```

El archivo de “logout.php” es para cerrar sesión en cualquier momento de la página, para poder salirse el usuario tuvimos que agregar un “sesión_start();” para iniciar la sesión y luego un “session_destroy();” para destruir/o salirse. Luego un “header” que pondremos la ubicación de donde queremos que nos mande cuando nos salgamos, en este caso lo mandamos a la página de inicio de nuestro proyecto.

```
<?php
    session_start();

    session_unset();

    session_destroy();

    header('Location: /proyect/index.php');
?>
```