

Proyecto Integrador

Anteproyecto

Alejandro López Corral e Iván Barchín

Grado Superior 2º ASIR - **Módulo** IAW

Índice

Índice.....	1
Introducción.....	2
Temática.....	2
Objetivos.....	2
Requisitos De La Página.....	3
Funciones Vitales.....	3
Funciones Adicionales.....	4
Plataformas o Herramientas a Utilizar.....	4
Visual Studio Code.....	4
Git / GitHub.....	4
XAMPP.....	5
Apache.....	5
MySQL.....	5
VirtualBox.....	5
Php.....	5
HTML5, CSS3 y JavaScript.....	6
División del trabajo.....	6
Integrantes del grupo.....	6
Alejandro López Corral.....	6
Iván Barchín Barrasa.....	6
Conclusión.....	6
Glosario.....	7

Introducción

Para el proyecto final de este trimestre se nos ha propuesto avanzar y aumentar las capacidades y funcionalidades del proyecto anterior, el cual fue uno más de aclimatación al desarrollo web con herramientas como **PHP** y **XAMPP**. En este proyecto se implantará el uso de accesos a Bases de Datos como **MariaDB**, a continuación se indagará en detalle sobre la organización del equipo de desarrollo y las herramientas utilizadas.

Temática

La temática de la página web de este proyecto estará centrada en un portal de compra online simulado de zapatillas y "**sneakers**". Las ventajas de crear un portal de compra incluyen:



- Integración fácil y eficaz de una base de datos.
- Gran potencial para la demostración de herramientas de desarrollo web como **PHP**, **MySQL**, **Apache**, **XAMPP**.
- Amplia Disponibilidad de recursos de aprendizaje
- Interés en el equipo por la temática en sí, incrementando la motivación por el proyecto

Objetivos

Para el éxito de este proyecto se han destacado varios aspectos que son esenciales para el funcionamiento eficaz del equipo y la fluidez en el desarrollo.

1. Coordinación del trabajo

Algo que ayudó mucho en el proyecto fue la división rápida del trabajo entre las funcionalidades de la página. Para el proyecto integrador, el cual tiene una escala mucho más grande comparado al anterior, esto es un aspecto vital para acelerar el desarrollo del proyecto.

2. Proficiencia con las herramientas

Para que el portal sea funcional, se deben de tener las habilidades necesarias para poder desarrollar una web, esto implica generar un código y eficiente que no afecte al rendimiento de la página gravemente.

3. Entendimiento entre los integrantes del grupo

Este objetivo engloba los 2 apartados anteriores, y principalmente se refiere a la documentación y el uso de comentarios en el código dentro del repositorio. Se debe buscar una manera de que los integrantes puedan entender rápidamente el funcionamiento del código formado.

4. Pies en el suelo ideando funcionalidades

Nuestro último objetivo está enfocado a los cimientos e ideas iniciales que conforman el proyecto. Esto se debe a que sin tener una idea clara desde el principio de cómo va a ser el producto final, es fácil perderse y pensar en nuevas funcionalidades sin primero cuando tiempo pueden llevar desarrollarlas. Por lo que se tiene que tener una idea firme y clara de los requisitos que se tienen que cumplir.

Requisitos De La Página

Continuando sobre el punto anterior, aquí presentamos un grupo de funcionalidades y características que debe tener la página para satisfacer el criterio del proyecto. Lo hemos dividido en 2 partes, unas que deben cumplirse estrictamente y otras que complementarán al proyecto.

Funciones Vitales

- ❖ Integración de base de datos (**MariaDB/MySQL**) con **PHP**
 - Se debe satisfacer y demostrar el uso de las operaciones **CRUD**
 - También se debe hacer de las funcionalidades de sesiones y cookies
- ❖ Opción multi idiomas
- ❖ Estilo eficaz y bien organizado
- ❖ División de usuarios comunes y administrativos.
 - Las herramientas de administración del servicio deben tener la seguridad suficiente y estar bloqueadas.

Funciones Adicionales

- ❖ Estilo de página con movimiento y transformaciones de páginas
- ❖ Servicio de host en un Servidor HTTPS con las herramientas de Apache

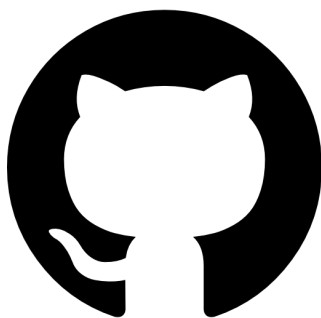
Plataformas o Herramientas a Utilizar

Visual Studio Code

Para el desarrollo y la formación de código hemos escogido el **IDE** de Visual Studio Code, ya que es uno de los más cómodos para poder configurar y encontrar errores. Además, el potencial que tiene mediante la instalación de extensiones hacen que sus funcionalidades incrementen. Otra función por la cual hemos escogido este **IDE** es la muy cómoda implementación con **Git** de la que dispone, facilitando la resolución de conflictos en **merge** y rápido paso entre las fases de trabajo dentro del repositorio.



Git / GitHub



Como hemos mencionado antes, escogimos Visual Studio Code para poder llevar tener una implementación de Git cómoda y rápida. En el proyecto anterior ya inicializamos y subimos nuestro repositorio en **GitHub** para tener un acceso muy fácil y tener complicación para estar al tanto sobre quién de los integrantes tiene la versión más avanzada.

XAMPP

Al tener que hacer uso de lenguajes de Back-End como **PHP** y herramientas de BBDD como **MySQL**. Necesitamos una herramienta para funcionar de servidor web al cual podremos conectarnos y llevar a cabo comprobaciones de funcionamiento y depuramiento de errores.

Apache

Al formar parte del paquete **XAMPP**, Apache cargará con la función del protocolo **HTTP** y **HTTPS**, en el proyecto anterior ya se hizo uso de esta herramienta, pero una escala mucho más pequeña y se tenía para comprobar el correcto funcionamiento del código **PHP**



MariaDB

También forma parte del paquete **XAMPP**, y será la ampliación principal que se hará sobre el proyecto anterior. Aquí se podrán guardar los datos de los Productos, Usuarios y listados de compras dentro de nuestro portal de compra.

VirtualBox

Este gestor de máquinas virtuales nos será de gran utilidad en entornos como la clase de nuestro grado, en las que no se puede instalar en los usuarios locales las herramientas previamente mencionadas.

PHP

Una de los requisitos iniciales del proyecto y la principal herramienta que se usará en este proyecto como lenguaje de **Back-End** será **PHP**, principalmente por sus capacidades de conexión, **CRUD** y gestión con **MySQL**.

HTML5, CSS3 y JavaScript



Las 3 principales herramientas que se utilizan al empezar en el campo del desarrollo web. Ya tenemos experiencia años previos trabajando con estos lenguajes y son las herramientas más útiles para formar código de **Front-End**.

División del trabajo

Integrantes del grupo

Alejandro López Corral

- Se encargará principalmente del desarrollo en el Back-End.
- Se coordinará con Iván para entender el estilo de la página y montar las funcionalidades en torno a dicho estilo.
- Resolución de problemas con Git.

Iván Barchín Barrasa

- Se encargará principalmente del desarrollo en el Front-End
- Decidirá el estilo de la página y se coordinará con Alejandro para la implementación de Back-End con dicho estilo.
- La lista de productos, precios, y el **flujo de usuarios** será dirigida por él.

Casos de Uso

Usuario Básico

El usuario básico solo interactúa con la página mediante la interfaz web del Front-End. Podrá navegar el catálogo de productos pero no podrá hacer ninguna compra ya que no tiene una cuenta registrada dentro del sistema.

Usuario Registrado

El usuario registrado tendrá acceso a las funcionalidades no previamente disponibles. Cuando decida hacer una compra, la base de datos web

recibirá una orden de actualizar el stock y generará una nueva entrada en la tabla de pedidos. Podrá acceder a un menú donde podrá comprobar su información personal, junto a todos los pedidos que ha realizado.

Usuario Administrador

El usuario administrador tendrá la capacidad de alterar los valores de los productos, esto incluye el precio, stock, imagen y nombre. Además de eso tendrá acceso a un apartado restringido donde podrá ver todos los pedidos realizados.

Conclusión

Para dar por concluido este anteproyecto creemos que este va a ser un proyecto el cual va a requerir altas capacidades de organización, coordinación y resolución de problemas para poder presentar unos buenos resultados.

Por último queremos añadir que la experiencia que ganemos en este proyecto nos ayudará increíblemente en el TFG que tendremos al final de este curso.

Glosario

- **Sneakers**
 - Anglicismo procedente del inglés, se refiere a zapatillas deportivas que suelen ser utilizadas principalmente para la moda urbana.
- **PhP**
 - Lenguaje Interpretado dirigido al lado del servidor en desarrollo web.
 - [Sitio Web oficial](#)
- **MariaDB**
 - Se trata de un fork comercial de **MySQL**.
 - Concede un gran nivel de compatibilidad con MySQL y sus APIs
 - [Sitio Web oficial](#)
- **MySQL**
 - Sistemas de gestión de base de datos relacional.
 - [Sitio Web oficial](#)
- **Apache**
 - Servicio de servidor HTTP de código abierto.

- [Sitio web oficial](#)

➤ **CRUD**

- Set de operaciones básicas en una base de datos.
- Operaciones:
 - CREATE
 - READ
 - UPDATE
 - DELETE

➤ **IDE**

- Integrated Development Environment, o en español, entorno de desarrollo integrado.
- Se refiere al software que nos proporciona herramientas comprensivas y facilidades para el desarrollo de aplicaciones.
- Ejemplos de IDE:
 - Visual Studio
 - Visual Studio Code
 - Geany
 - NetBeans
 - Atom
 - Arduino IDE

➤ **Git**

- Sistema de control de versiones distribuido.
- Comprueba los cambios en el contenido dentro de un repositorio y el contenido de los archivos dentro del mismo.
- Permite una distribución rápida y potente entre proyectos colaborativos.

- [Sitio Web oficial](#)

➤ **GitHub**

- Servicio web para el almacenamiento de repositorios con el control del software Git.
- Usado principalmente para la distribución de aplicaciones de código abierto.

- [Sitio Web oficial](#)

➤ **Merge**

- La operación dentro de Git que desempeña la función de unir dos versiones diferentes dentro de un repositorio Git.
- [Documentación](#)

➤ **HTTP**

- Hyper-Text Transfer Protocol.
- Protocolo en la capa de aplicación, pilar esencial para la comunicación de páginas web.
- [Documentación](#)

➤ **HTTPS**

- Hyper-Text Transfer Protocol Secure
- Actualización al protocolo HTTP, crea una comunicación segura y encriptada mediante los protocolos TLS (Transport Layer Security) o SSL (Secure Sockets Layer).

➤ **Back-End**

- En una estructura Cliente-Servidor, esto se refiere al servidor
- Se puede entender como Back-End el código que reside en el servidor y el cliente no tiene capacidad de interactuar.

➤ **Front-End**

- En una estructura Cliente-Servidor, esto se refiere al cliente
- Se puede entender como Front-End el código que compone con lo que interactúa el usuario.

➤ **Flujo de Usuarios**

- Mapas que representan las navegaciones posibles a través de diferentes pantallas y funciones en una aplicación web.