

PROYECTO FINAL

QUITO - ECUADOR



INSTITUTO TECNOLÓGICO SUPERIOR
QUITO METROPOLITANO
FORMANDO PROFESIONALES DE ÉLITE

ITSQMÉT

EVIDENCIAR EL PROCESO DE APRENDIZAJE POR PARTE DE LOS ESTUDIANTES

WWW.ITSQMET.EDU.EC



**INSTITUTO TECNOLÓGICO SUPERIOR
QUITO METROPOLITANO**



INSTITUTO TECNOLÓGICO SUPERIOR
QUITO METROPOLITANO
FORMANDO PROFESIONALES DE ÉLITE
ITSQM^{ET}

**DESARROLLO DE UNA PÁGINA WEB PARA EL ALMACÉN
DE TECNOLOGÍA “COMPUNET”**

**PROGRAMACIÓN ORIENTADA A OBJETOS II
ADAPTACIONES WEB I
LIC. CRISTIAN PAUL NARANJO BARRIONUEVO**

INTEGRANTES:

LEONARDO JAVIER GUANOQUIZA YEPEZ

LEONARDO DAVID VASQUEZ ALEMAN

JOHANNA PAOLA QUINDE ABRIL

ABRIL 2022 – SEPTIEMBRE 2022

CONTENIDO

INDICACIONES GENERALES.....	5
Desarrollo de una Aplicación Web para el Almacén de Tecnología “Compunet”	6
INTRODUCCIÓN	6
1. CAPÍTULO I.....	6
1.1. PLANTEAMIENTO DEL PROBLEMA	6
1.2. OBJETIVO GENERAL.....	6
1.3. OBJETIVOS ESPECÍFICOS	7
1.4. JUSTIFICACIÓN	7
2. CAPÍTULO II.....	8
2.1. MARCO TEÓRICO.....	8
2.1.1. ¿Qué es NetBeans?	8
2.1.2. ¿Qué es Java?.....	8
2.1.1.2. Características de java.....	8
2.1.3. ¿Qué es JavaScript?	9
2.1.3.1. Funciones JavaScript.....	10
2.1.4. ¿Qué es AJAX?	11
2.1.4.1. Funcionamiento de AJAX.....	11
2.1.5. ¿Qué es JSP?.....	12
2.1.5.1. Motor JSP	12
2.1.6. ¿Qué es Bootstrap?	13
2.1.6.1. ¿Funcionamiento de Bootstrap?	14
2.1.7. ¿Qué es SQL?	14
2.1.8. ¿Qué es MySQL?.....	15
2.1.8.1. Funciones MySQL	15
2.1.9. ¿Qué es Maven?.....	16



2.1.10.	¿Qué es Tomcat?.....	18
2.1.11.	¿Qué es un servlet de Java?	19
2.1.12.	Java y Javax.....	20
2.1.13.	JDK.....	21
2.1.14.	¿Qué es HTML5?	21
2.1.15.	¿Qué es css?	22
3.	CAPÍTULO III	23
3.1.	DESARROLLO Y/O IMPLEMENTACIÓN	23
4.	CAPÍTULO IV	24
4.1.	ANÁLISIS DE RESULTADOS	24
4.1.1.	Anexos	26
BIBLIOGRAFÍA.....		27

Índice de Figuras

Ilustración 1	Funcionamiento de AJAX	11
Ilustración 2	HTML + CSS.....	22
Ilustración 3	Funcionamiento de la aplicación Web.....	23
Ilustración 4	Vista de la página de ingreso	24
Ilustración 5	Vista de la página de Inicio de sesión.....	25
Ilustración 6	Vista la tabla Clientes	25



INDICACIONES GENERALES

- Respetar el formato establecido por el docente.
- Las imágenes tienen que ir con descripciones y fuentes, con normas APA 7ma edición.
- El presente documento tiene por defecto configurado los diferentes recursos de normas APA, en la sección de estilos, usarlos para los diferentes, temas y subtemas desarrollados en clase, además de los márgenes y encabezados y pies de página.
- La fuente de letra es 12 y estilo Times News Román.



Desarrollo de una Aplicación Web para el Almacén de Tecnología “Compunet”

I NTRODUCCIÓN

1. CAPÍTULO I

1.1. PLANTEAMIENTO DEL PROBLEMA

En los inicios del almacén de tecnología “Compunet” se llevaba un registro de productos, clientes y proveedores de manera física, pero conforme el almacén a ido creciendo y con el avance de tecnología que nos dejó la pandemia, se vieron obligados a analizar la opción de automatizar el control su información a través de una aplicación web conectada a una base de datos cumpliendo con los requerimientos que fueron surgiendo en este proceso de crecimiento.

1.2. OBJETIVO GENERAL

Realizar una aplicación web que tenga conexión con una base de datos con el objetivo mejorar el manejo de información para el almacén de tecnología “COMPUNET” aplicando los conocimientos adquiridos en la asignatura.



1.3. OBJETIVOS ESPECÍFICOS

1. Implementar una aplicación web para una correcta administración de información del almacén de tecnología “Compunet”.
2. Por medio de consultas sql conectadas a nuestra aplicación web poder controlar la inserción, actualización y eliminación de clientes, proveedores y productos de nuestro almacén de tecnología “Compunet”.
3. Mediante jsp realizar las vistas correspondientes para nuestra aplicación web, que le permitan al usuario visualizar la información de una manera más organizada y concisa.

1.4. JUSTIFICACIÓN

La idea de hacer una aplicación web para el almacén de tecnología “COMPUNET” nace de la problemática que ha existido en estos últimos años, con el avance de tecnología que nos dejó la pandemia y el crecimiento que ha tenido nuestro almacén, con la implementación de la aplicación web se desea dejar de lado el ingresar datos de manera física, y automatizar todo este proceso para un mejor y más fácil manejo de información de nuestra empresa.



2. CAPÍTULO II

2.1. MARCO TEÓRICO

2.1.1. ¿Qué es NetBeans?

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE1 es un producto libre y gratuito sin restricciones de uso.

2.1.2. ¿Qué es Java?

Java es un tipo de lenguaje de programación y una plataforma informática, creada y comercializada por Sun Microsystems en el año 1995. Se constituye como un lenguaje orientado a objetos. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una sola vez y lo ejecuten en cualquier dispositivo.

2.1.1.2. Características de java

- Java es un lenguaje potente, derivado de C y C++, pero sin las características menos usadas y más confusas de estos, haciéndolo más sencillo para los usuarios.
- El enfoque orientado a objetos (OO) es uno de los estilos de programación más populares. Permite diseñar el software de forma que los distintos tipos de datos que se usen estén unidos a sus operaciones.
- Java proporciona una gran biblioteca estándar y herramientas para que los programas puedan ser distribuidos.



- Java es portable esto significa que cualquier programa escrito en el lenguaje Java pueden ejecutarse en cualquier tipo de hardware.
- Proporcionando una plataforma segura para desarrollar y ejecutar aplicaciones que, administra automáticamente la memoria, provee canales de comunicación segura protegiendo la privacidad de los datos y, al tener una sintaxis rigurosa evita que se quiebre el código, es decir, no permite la corrupción del mismo.
- Java logra llevar a cabo varias tareas simultáneamente dentro del mismo programa. Esto permite mejorar el rendimiento y la velocidad de ejecución dando como resultado su funcionamiento multihilo.

2.1.3. ¿Qué es JavaScript?

JavaScript es el lenguaje de programación encargado de dotar de mayor interactividad y dinamismo a las páginas web. Cuando JavaScript se ejecuta en el navegador, no necesita de un compilador. El navegador lee directamente el código, sin necesidad de terceros. Por tanto, se le reconoce como uno de los tres lenguajes nativos de la web junto a HTML (contenido y su estructura) y a CSS (diseño del contenido y su estructura).

No conviene confundir JavaScript con Java, que es un lenguaje de programación muy diferente. La confusión proviene del nombre, registrado por la misma empresa creadora de Java (Sun Microsystems). JavaScript (JS) se creó posteriormente, y la empresa norteamericana lo que hizo simplemente fue cambiar el nombre que le habían puesto sus creadores al comprar el proyecto (LiveScript). El lenguaje de programación Java está orientado a muchas más cosas que la web desde sus inicios.



2.1.3.1. Funciones JavaScript

Con este lenguaje de programación del lado del cliente podemos crear efectos y animaciones sin ninguna interacción, o respondiendo a eventos causados por el propio usuario tales como botones pulsados y modificaciones del DOM (document object model). Por tanto, nada tiene que ver con el lenguaje de programación Java, ya que su principal función es ayudar a crear páginas webs dinámicas.

El código de programación de JavaScript se ejecuta en los navegadores, ya sean de escritorio o móviles, ya sean Android o Iphone. Sirve para exactamente lo mismo, da igual en el tipo de dispositivo que se ejecute el navegador.

JavaScript es capaz de detectar errores en formularios, de crear bonitos sliders que se adapten a cualquier pantalla, de hacer cálculos matemáticos de forma eficiente, de modificar elementos de una página web de forma sencilla. Pero también JS es el encargado de que existan herramientas como Google Analytics, Google Tag Manager, Facebook Pixel y tantas otras, que son claros ejemplos de JavaScript.

Existe una tecnología llamada AJAX que permite intercambiar información con el servidor sin tener que recargar la página. Es decir, sólo cargamos de la página lo necesario. Esta tecnología desarrollada en JavaScript ha supuesto uno de los principales avances en el desarrollo web. Aunque no la sepamos reconocer, es la encargada de que podamos conseguir más mensajes, tweets, emails...sólo pulsando un botón, sin tener que recargar las paginaciones JavaScript

2.1.4. ¿Qué es AJAX?

AJAX significa JavaScript asíncrono y XML (Asynchronous JavaScript and XML). Es un conjunto de técnicas de desarrollo web que permiten que las aplicaciones web funcionen de forma asíncrona, procesando cualquier solicitud al servidor en segundo plano.

2.1.4.1. Funcionamiento de AJAX

- **HTML/XHTML** para el lenguaje principal y **CSS** para la presentación.
- **El Modelo de objetos del documento (DOM)** para datos de visualización dinámicos y su interacción.
- **XML** para el intercambio de datos y **XSLT** para su manipulación. Muchos desarrolladores han comenzado a reemplazarlo por **JSON** porque es más similar a JavaScript en su forma.
- El objeto **XMLHttpRequest** para la comunicación asíncrona.
- Finalmente, el lenguaje de programación **JavaScript** para unir todas estas tecnologías.

Modelo convencional	Modelo AJAX
1. Se envía una solicitud HTTP desde el navegador web al servidor.	1. El navegador crea una llamada de JavaScript que luego activará XMLHttpRequest.
2. El servidor recibe y, posteriormente, recupera los datos.	2. En segundo plano, el navegador web crea una solicitud HTTP al servidor.
3. El servidor envía los datos solicitados al navegador web.	3. El servidor recibe, recupera y envía los datos al navegador web.
4. El navegador web recibe los datos y vuelve a cargar la página para que aparezcan los datos.	4. El navegador web recibe los datos solicitados que aparecerán directamente en la página. No se necesita recargar.
Durante este proceso, los usuarios no tienen más remedio que esperar hasta que se complete todo el proceso. No solo consume mucho tiempo, sino que también supone una carga innecesaria en el servidor.	

Ilustración 1 Funcionamiento de AJAX



2.1.5. ¿Qué es JSP?

JSP es un acrónimo de Java Server Pages, (Páginas de Servidor Java). Es, pues, una tecnología orientada a crear páginas web con programación en Java.

Con JSP podemos crear aplicaciones web que se ejecuten en variados servidores web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java.

2.1.5.1. Motor JSP

El motor de las páginas JSP está basado en los servlets de Java -programas en Java destinados a ejecutarse en el servidor-, aunque el número de desarrolladores que pueden afrontar la programación de JSP es mucho mayor, dado que resulta mucho más sencillo aprender que los servlets.

En JSP creamos páginas de manera parecida a como se crean en ASP o PHP -otras dos tecnologías de servidor-. Generamos archivos con extensión .jsp que incluyen, dentro de la estructura de etiquetas HTML, las sentencias Java a ejecutar en el servidor. Antes de que sean funcionales los archivos, el motor JSP lleva a cabo una fase de traducción de esa página en un servlet, implementado en un archivo class (Byte codes de Java).



Esta fase de traducción se lleva a cabo habitualmente cuando se recibe la primera solicitud de la página .jsp, aunque existe la opción de precompilar en código para evitar ese tiempo de espera la primera vez que un cliente solicita la página

2.1.6. ¿Qué es Bootstrap?

Bootstrap es un framework CSS desarrollado por Twitter en 2010, para estandarizar las herramientas de la compañía.

El framework combina CSS y JavaScript para estilizar los elementos de una página HTML. Permite mucho más que, simplemente, cambiar el color de los botones y los enlaces.

Esta es una herramienta que proporciona interactividad en la página, por lo que ofrece una serie de componentes que facilitan la comunicación con el usuario, como menús de navegación, controles de página, barras de progreso y más.

Además de todas las características que ofrece el framework, su principal objetivo es permitir la construcción de sitios web responsive para dispositivos móviles.

Esto significa que las páginas están diseñadas para funcionar en desktop, tablets y smartphones, de una manera muy simple y organizada.



2.1.6.1. ¿Funcionamiento de Bootstrap?

Bootstrap está constituido por una serie de archivos CSS y JavaScript responsables de asignar características específicas a los elementos de la página.

Hay un archivo principal llamado bootstrap.css, que contiene una definición para todos los estilos utilizados. Básicamente, la estructura del framework se compone de dos directorios:

css: contiene los archivos necesarios para la estilización de los elementos y una alternativa al tema original.

js: contiene la parte posterior del archivo bootstrap.js (original y modificado), responsable de la ejecución de aplicaciones de estilo que requieren manipulación interactiva.

Para asignarle una característica a un elemento, simplemente debemos informar la clase correspondiente en la propiedad “class” del elemento que será estilizado

2.1.7. ¿Qué es SQL?

SQL es un acrónimo en inglés para Structured Query Language. Un Lenguaje de Consulta Estructurado. Un tipo de lenguaje de programación que te permite manipular y descargar datos de una base de datos. Tiene capacidad de hacer cálculos avanzados y álgebra. Es utilizado en la mayoría de empresas que almacenan datos en una base de datos. Ha sido y sigue siendo el lenguaje de programación más usado para bases de datos relacionales.



2.1.8. ¿Qué es MySQL?

MySQL es un sistema de gestión de bases de datos relacionales de código abierto (RDBMS, por sus siglas en inglés) con un modelo cliente-servidor. RDBMS es un software o servicio utilizado para crear y administrar bases de datos basadas en un modelo relacional.

2.1.8.1. Funciones MySQL

1. **Arquitectura Cliente y Servidor:** MySQL basa su funcionamiento en un modelo cliente y servidor. Es decir, clientes y servidores se comunican entre sí de manera diferenciada para un mejor rendimiento. Cada cliente puede hacer consultas a través del sistema de registro para obtener datos, modificarlos, guardar estos cambios o establecer nuevas tablas de registros.
2. **Compatibilidad con SQL:** SQL es un lenguaje generalizado dentro de la industria. Al ser un estándar MySQL ofrece plena compatibilidad por lo que si has trabajado en otro motor de bases de datos no tendrás problemas en migrar a MySQL.
3. **Vistas:** Desde la versión 5.0 de MySQL se ofrece compatibilidad para poder configurar vistas personalizadas del mismo modo que podemos hacerlo en otras bases de datos SQL. En bases de datos de gran tamaño las vistas se hacen un recurso imprescindible.
4. **Procedimientos almacenados.** MySQL posee la característica de no procesar las tablas directamente, sino que a través de procedimientos almacenados es posible incrementar la eficacia de nuestra implementación.
5. **Desencadenantes.** MySQL permite además poder automatizar ciertas tareas dentro de nuestra base de datos. En el momento que se produce un evento otro es lanzado para actualizar registros o optimizar su funcionalidad.



6. **Transacciones.** Una transacción representa la actuación de diversas operaciones en la base de datos como un dispositivo. El sistema de base de registros avala que todos los procedimientos se establezcan correctamente o ninguna de ellas. En caso por ejemplo de una falla de energía, cuando el monitor falla u ocurre algún otro inconveniente, el sistema opta por preservar la integridad de la base de datos resguardando la información.

2.1.9. ¿Qué es Maven?

Maven simplifica mucho el proceso de build del código, permitiéndonos compilar cualquier tipo de proyecto de la misma manera, librándonos de todas las dificultades que hay por detrás.

Pero lo cierto es que Maven es mucho más que una herramienta que hace builds del código.

Podríamos decir, que Maven es una herramienta capaz de gestionar un proyecto software completo, desde la etapa en la que se comprueba que el código es correcto, hasta que se despliega la aplicación, pasando por la ejecución de pruebas y generación de informes y documentación.



Para ello, en Maven se definen tres ciclos de build del software con una serie de etapas diferenciadas. Por ejemplo, el ciclo por defecto tiene las etapas de:

- **Validación** (validate): Validar que el proyecto es correcto.
- **Compilación** (compile).
- **Test** (test): Probar el código fuente usando un framework de pruebas unitarias.
- **Empaquetar** (package): Empaquetar el código compilado y transformarlo en algún formato tipo .jar o .war.
- **Pruebas de integración** (integration-test): Procesar y desplegar el código en algún entorno donde se puedan ejecutar las pruebas de integración.
- **Verificar** que el código empaquetado es válido y cumple los criterios de calidad (verify).
- **Instalar** el código empaquetado en el repositorio local de Maven, para usarlo como dependencia de otros proyectos (install).
- **Desplegar** el código a un entorno (deploy).

Para poder llevar a cabo alguna de estas fases en nuestro código, tan solo tendremos que ejecutar mvn y el nombre de la fase (la palabra que puse entre paréntesis). Además, van en cadena, es decir, si empaquetamos el código (package), Maven ejecutará desde la fase de validación (validate) a empaquetación (package). Así de simple.



Por otra parte, con Maven la gestión de dependencias entre módulos y distintas versiones de librerías se hace muy sencilla. En este caso, solo tenemos que indicar los módulos que componen el proyecto, o qué librerías utiliza el software que estamos desarrollando en un fichero de configuración de Maven del proyecto llamado POM.

Además, en el caso de las librerías, no tienes ni tan siquiera que descargarlas a mano. Maven posee un repositorio remoto (Maven central) donde se encuentran la mayoría de librerías que se utilizan en los desarrollos de software, y que la propia herramienta se descarga cuando sea necesario.

Incluso, establece una estructura común de directorios para todos los proyectos. Por ejemplo el código estará en `${raíz del proyecto}/src/main/java`, los recursos en `${ raíz del proyecto }/src/main/resources`. Los tests están en `${ raíz del proyecto }/src/test` etc.

2.1.10. ¿Qué es Tomcat?

Estrictamente hablando, Tomcat no es un servidor web como Apache HTTPS Server o NGINX. Comenzado en 1999 y desarrollado como un proyecto de código abierto por la Apache Software Foundation (ASF), Apache Tomcat es un contenedor Java Servlet, o contenedor web, que proporciona la funcionalidad extendida para interactuar con Java Servlets, al tiempo que implementa varias especificaciones técnicas de la plataforma Java: JavaServer Pages (JSP), Java Expression Language (Java EL) y WebSocket.



2.1.11. ¿Qué es un servlet de Java?

Este es un software que permite que un servidor web maneje contenido web dinámico basado en Java utilizando el protocolo HTTP. JSP es una tecnología similar que permite a los desarrolladores crear contenido dinámico utilizando documentos HTML o XML. En términos de su capacidad para habilitar contenido dinámico, los Servlets de Java y JSP son ampliamente comparables a PHP o ASP.NET, solo basados en el lenguaje de programación Java.

Al reunir todas estas tecnologías basadas en Java, Tomcat ofrece un entorno de servidor web «Java puro» para ejecutar aplicaciones creadas en el lenguaje de programación Java.

Apache Tomcat vs Servidor HTTP Apache

Las comparaciones entre Apache Tomcat y Apache HTTPS Server pueden ser confusas, ya que ambas tienen Apache en su nombre, y ambas son desarrolladas por la ASF. Pero la verdad es que son dos paquetes de software muy diferentes.

Si bien Apache es un servidor web HTTPS tradicional, optimizado para manejar contenido web estático y dinámico (muy a menudo basado en PHP), carece de la capacidad de administrar Servlets Java y JSP. Tomcat, por otro lado, está casi totalmente orientado al contenido basado en Java. De hecho, Tomcat se desarrolló originalmente como un medio para proporcionar la funcionalidad JSP que Apache carecía.



Incluso con esto en mente, una comparación entre Tomcat Server y Apache no se reduce a una competencia directa. Esto se debe a que es completamente viable ejecutarlos uno al lado del otro. Entonces, en proyectos que involucran contenido basado en Java y PHP, por ejemplo, tiene sentido que Apache maneje la mayor parte del contenido estático y dinámico, mientras que Tomcat se encarga del JSP.

Sin embargo, por sí solo, Tomcat no es particularmente eficiente como un servidor HTTP tradicional, por lo que Apache es una opción mucho mejor para sitios web dinámicos creados únicamente con un lenguaje como PHP. Pero, por supuesto, cuando se trata de sitios basados completamente en JSP, Tomcat es el líder indiscutible de la pandilla.

2.1.12. Java y Javax

Java y Javax son los paquetes que se utilizan dentro del lenguaje de programación Java. Influenciado por el lenguaje C, Java deriva la mayor parte de sus sintaxis de C y C ++. Sin embargo, las facilidades que ofrece son superiores a los otros lenguajes de programación. Ha sido diseñado para fines generales y apenas depende de su implementación. En comparación con los otros lenguajes, está más orientado a objetos, así como concurrente y basado en clases. Inicialmente fue desarrollado en Sun Microsystems por James Gosling. Java se lanzó por primera vez en 1995 como un componente central de la plataforma Java de la empresa.

Tanto Java como Javax son paquetes. El comando que se usa para abrir un paquete es “importar java” mientras que el que se usa para abrir el paquete de extensión de java es “importar javax”.



Este lenguaje de programación está diseñado de tal manera que permite que el mismo código se ejecute en varias plataformas. Por lo tanto, los desarrolladores pueden escribir el código una vez y reutilizarlo en varias plataformas. De este modo, normalmente estas aplicaciones se compilan con código de bytes.

Javax es una dependencia externa que puede utilizarse con el lenguaje de programación Java.

2.1.13. JDK

Java™ Development Kit (JDK) es un software para los desarrolladores de Java. Incluye el intérprete Java, clases Java y herramientas de desarrollo Java (JDT): compilador, depurador, desensamblador, visor de applets, generador de archivos de apéndice y generador de documentación.

El JDK le permite escribir aplicaciones que se desarrollan una sola vez y se ejecutan en cualquier lugar de cualquier máquina virtual Java. Las aplicaciones Java desarrolladas con el JDK en un sistema se pueden usar en otro sistema sin tener que cambiar ni recompilar el código. Los archivos de clase Java son portables a cualquier máquina virtual Java estándar.

2.1.14. ¿Qué es HTML5?

HTML5 (HyperText Markup Language, versión 5) es la quinta revisión del lenguaje HTML y permite definir los nuevos estándares de desarrollo web, modificando el código existente para solucionar problemas y actualizándolo a las nuevas necesidades de hoy en día.

HTML5 nos permite crear la estructura de una página web, por así decirlo, su esqueleto, donde vamos a incorporar posteriormente, imágenes, textos, vídeos y todo tipo de material multimedia de forma que se pueda visualizar correctamente.

2.1.15. ¿Qué es css?

Las siglas CSS (*Cascading Style Sheets*) significan «Hojas de estilo en cascada» y parten de un concepto simple pero muy potente: aplicar estilos (colores, formas, márgenes, etc...) a uno o varios documentos (*generalmente documentos HTML, páginas webs*) de forma masiva.

Se le denomina estilos en cascada porque se aplican de arriba a abajo (*siguiendo un patrón denominado herencia que trataremos más adelante*) y en el caso de existir ambigüedad, se siguen una serie de normas para resolverla.

La idea de CSS es la de utilizar el concepto de separación de presentación y contenido, intentando que los documentos HTML incluyan sólo información y datos, relativos al significado de la información a transmitir (*el contenido*), y todos los aspectos relacionados con el estilo (diseño, colores, formas, etc...) se encuentren en un documento CSS independiente (*la presentación*):



Ilustración 2 HTML + CSS

3. CAPÍTULO III

3.1. DESARROLLO Y/O IMPLEMENTACIÓN



Ilustración 3 Funcionamiento de la aplicación Web



4. CAPÍTULO IV

4.1. ANÁLISIS DE RESULTADOS

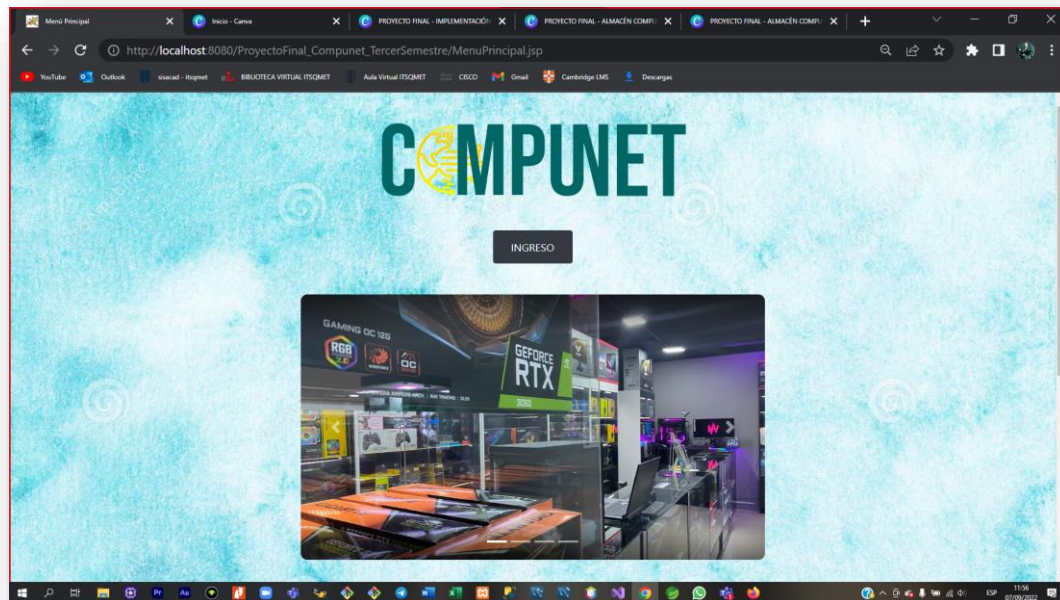


Ilustración 4 Vista de la página de ingreso

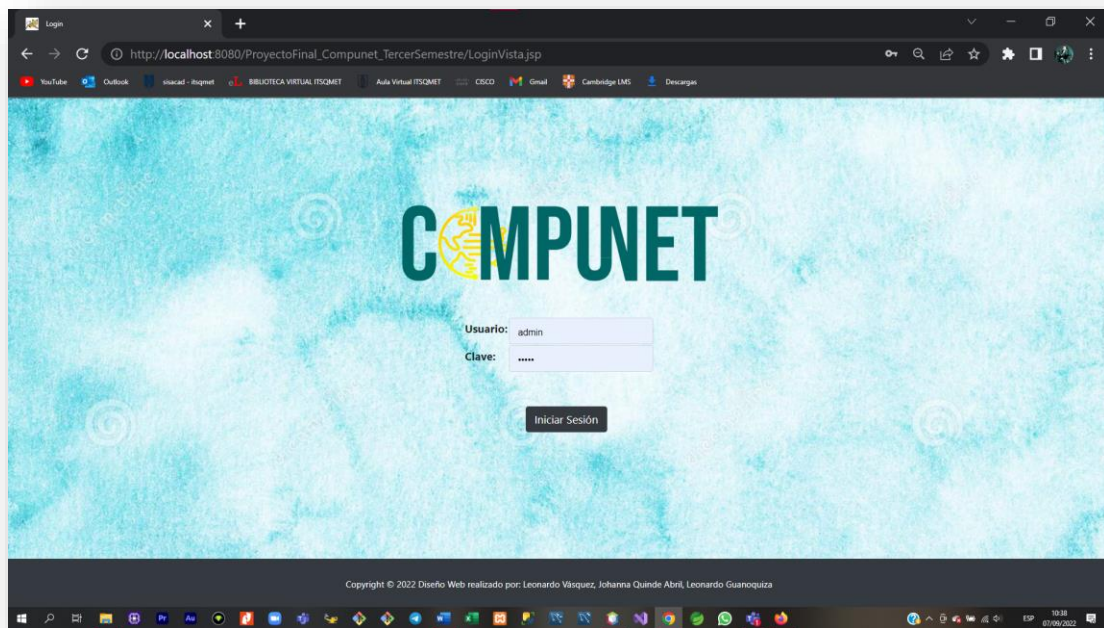


Ilustración 5 Vista de la página de inicio de sesión

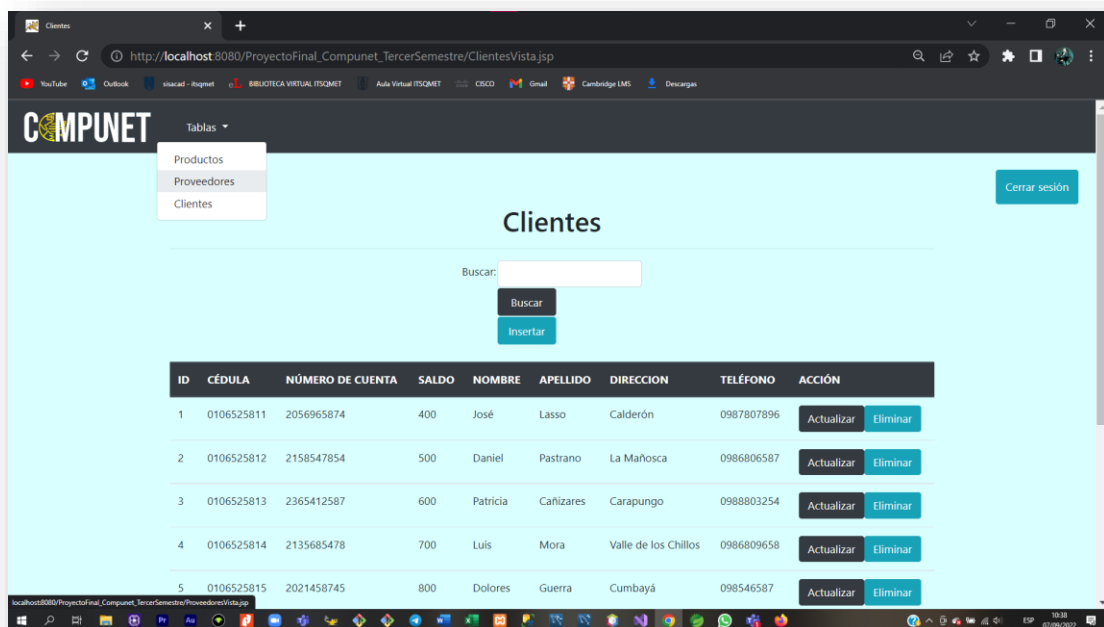


Ilustración 6 Vista la tabla Clientes



4.1.1. Anexos

<https://www.canva.com/design/DAFLj->

[Xx9sE/EqQaiUib12VT8QSur1byNQ/view?utm_content=DAFLj-](https://www.canva.com/design/DAFLj-Xx9sE/EqQaiUib12VT8QSur1byNQ/view?utm_content=DAFLj-)

[Xx9sE&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton](https://www.canva.com/design/DAFLj-Xx9sE/EqQaiUib12VT8QSur1byNQ/view?utm_content=DAFLj-Xx9sE&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)



BIBLIOGRAFÍA

- @Manz. (S.f.). *lenguajecss.com*. Obtenido de ¿Qué es CSS?:
<https://lenguajecss.com/css/introduccion/que-es-css/>
- Anónimo, & Ortiz, A. (26 de marzo de 2020). *hostdime.com*. Obtenido de ¿Qué es Apache Tomcat?: <https://www.hostdime.com.ar/blog/que-es-apache-tomcat/>
- Confora, G. (2021). *miformaciongratis.com*. Obtenido de HTML5, ¿qué es y para qué sirve?:
<https://www.miformaciongratis.com/blog-post/html5-que-es-y-para-que-sirve/>
- Garzas, J. (2022). *javiergarzas.com*. Obtenido de Simple y rápido. Entiende qué es Maven en menos de 10 min.: <https://www.javiergarzas.com/2014/06/maven-en-10-min.html>
- IBM. (14 de abril de 2021). *IBM*. Obtenido de Java Development Kit:
<https://www.ibm.com/docs/es/i/7.3?topic=platform-java-development-kit>
- Miguel. (S.f.). *scribd.com*. Obtenido de Diferencia Entre Java y JavaX:
<https://es.scribd.com/document/480579642/Diferencia-entre-Java-y-JavaX>
- EL AUTOR. ¿Qué es AJAX y cómo funciona? (2022). Recuperado el 1 de septiembre del 2022 de: <https://www.hostinger.es/tutoriales/que-es-ajax>
- Qué es JSP. (2022). Recuperado el 1 de septiembre del 2022 de:
<https://desarrolloweb.com/articulos/831.php>
- Bootstrap: guía para principiantes de qué es, por qué y cómo usarlo. (2022). Recuperado el 1 de septiembre del 2022 de: <https://rockcontent.com/es/blog/bootstrap/>
- Datademia. ¿Qué es SQL?. (2022). Recuperado el 1 de septiembre del 2022 de:
<https://datademia.es/blog/que-es-sql>



Escrito por Angel Robledano el 24 de Septiembre de 2019 4 min de lectura Big Data. Qué es

MySQL: Características y ventajas | OpenWebinars. (2022). Recuperado el 1 de

septiembre del 2022 de: <https://openwebinars.net/blog/que-es-mysql/>

¿Qué es NetBeans? Ventajas y usos. (2022). Recuperado el 1 de septiembre del 2022 de:

<https://immune.institute/blog/que-es-netbeans/>

¿Qué es Java? Conoce las particularidades de este lenguaje de programación. Recuperado el 1

de septiembre del 2022 de: <https://rockcontent.com/es/blog/que-es-java/>

¿Qué es JavaScript y para qué sirve? (2022). Recuperado el 1 de septiembre del 2022 de:

<https://soyrafaramos.com/que-es-javascript-para-que-sirve/>

EL AUTOR. ¿Qué es AJAX y cómo funciona? (2022). Recuperado el 1 de septiembre del

2022 de: <https://www.hostinger.es/tutoriales/que-es-ajax>

Qué es JSP. (2022). Recuperado el 1 de septiembre del 2022 de:

<https://desarrolloweb.com/articulos/831.php>

Bootstrap: guía para principiantes de qué es, por qué y cómo usarlo. (2022). Recuperado el 1

de septiembre del 2022 de: <https://rockcontent.com/es/blog/bootstrap/>

Datademia. ¿Qué es SQL?. (2022). Recuperado el 1 de septiembre del 2022 de:

<https://datademia.es/blog/que-es-sql>

Escrito por Angel Robledano el 24 de Septiembre de 2019 4 min de lectura Big Data. Qué es

MySQL: Características y ventajas | OpenWebinars. (2022). Recuperado el 1 de

septiembre del 2022 de: <https://openwebinars.net/blog/que-es-mysql/>