

PROYECTO FINAL

QUITO - ECUADOR



INSTITUTO TECNOLÓGICO SUPERIOR
QUITO METROPOLITANO
FORMANDO PROFESIONALES DE ÉLITE

ITSQMET ITSQMET

EVIDENCIAR EL PROCESO DE APRENDIZAJE POR PARTE DE LOS ESTUDIANTES

WWW.ITSQMET.EDU.EC



**INSTITUTO TECNOLÓGICO SUPERIOR
QUITO METROPOLITANO**



DESARROLLO DE LA APLICACIÓN TO-DO APP

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

INTEGRANTES:

- **BRAVO RAMIREZ LUIS DAVID**
- **GREFA SANCHEZ ANDERSON JULIO**
- **GUERRERO JARA MARÍA PAULA**

ABRIL 2022 – SEPTIEMBRE 2022

CONTENIDO

INDICACIONES GENERALES	5
DESARROLLO DE LA APLICACIÓN WEB TO-DO APP.....	6
INTRODUCCIÓN	6
1. CAPÍTULO I	6
1.1. PLANTEAMIENTO DEL PROBLEMA	6
1.2. OBJETIVO GENERAL	7
1.3. OBJETIVOS ESPECÍFICOS.....	7
1.4. JUSTIFICACIÓN	7
2. CAPÍTULO II.....	8
2.1. MARCO TEÓRICO.....	8
2.1.1. Mysql - 5.7.....	8
2.1.2. Excalidraw	9
2.1.3. Tomcat - 10.0.8.....	9
2.1.4. Jakarta – 6.0.0.....	9
2.1.5. Maven - 4.0.0.....	10
2.1.6. GSON – 2.9.1	10
2.1.7. Java – 17	10
2.1.8. Bootstrap – 5.2.....	11
2.1.9. Bootstrap-table – 1.21.0.....	12
2.1.10. Html5	12
2.1.11. CSS3	13
2.1.12. JavaScript.....	14
2.1.13. Fetch.....	14
2.1.14. MVC	15
“Modelos	16



Vistas	16
Controladores	17
Arquitectura de aplicaciones MVC	17
3. CAPÍTULO III	20
3.1. DESARROLLO Y/O IMPLEMENTACIÓN	20
LOGIN	20
NOTAS.....	20
TABLAS	21
4. CAPÍTULO IV	22
4.1. ANÁLISIS DE RESULTADOS	22
BIBLIOGRAFÍA.....	24
ANEXOS	26

Índice de Figuras

Ilustración 1:Modelo MVC recuperado de: https://desarrolloweb.com/articulos/que-es-mvc.html	18
---	----

Índice de Tablas

No se encuentran elementos de tabla de ilustraciones.



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 LA APLICACIÓN WEB TO-DO APP

INTRODUCCIÓN

La materia de Programación Orientada a Objetos II y Adaptaciones Web I de acuerdo con el PEA de la asignatura indican que buscan “implementar programación orientada a objetos en soluciones de software orientados a la industria, empresas o software a medida, para la cual, el estudiante utilizará lo aprendido en conjunto con herramientas de desarrollo de software, tales como: Netbeans, Java y MySql” y también busca “analizar, diseñar e implementar soluciones en la web utilizando HTML y los fundamentos de javascript para problemática de presentación de contenido de hipermedia. Las herramientas que los estudiantes aplicarán para la implementación de soluciones web son: Netbeans y Apache” (David, 2022).

Como se detalla en el párrafo anterior el objetivo de las asignaturas es dar soluciones de software utilizando herramientas tecnológicas. Partiendo de esta premisa se propone la elaboración de una aplicación web que consiste en la creación de un check-list para el control de actividades.

1. CAPÍTULO I

1.1. PLANTEAMIENTO DEL PROBLEMA

El constante desarrollo tecnológico y cambio acelerado en la vida de la persona ha hecho que se su tiempo sea limitado, por lo que papelerías, imprentas han producido miles de agendas, organizadores y cuadernos para apuntar tareas pendientes, también se cuenta con aplicaciones móviles como el blog de notas, o aplicaciones de organización, pero estas tienen limitantes ya que se ofrece una versión demo a fin de comprar suscripciones que habiliten funciones. Por tal motivo y frente a la necesidad como estudiantes de tener una aplicación que esté disponible en



todo momento y permita categorizar las tareas, y dentro de cada categoría anotar pendientes con fechas límite. Se propone la creación de la aplicación to-do app misma que tiene un boceto de post-it para las tablas y de lista para las notas a fin de tener mayor orden y control sobre las actividades.

1.2.OBJETIVO GENERAL

Diseñar y programar una aplicación web para generar check-list de tareas.

1.3.OBJETIVOS ESPECÍFICOS

- Diagramar la funcionalidad de la aplicación y de su arquitectura
- Utilizar javascript para realizar tareas en el lado del frontend
- Utilizar html, css y bootstrap para estilizar la aplicación web
- Usar java servlets en el lado del backend aplicando el modelo MVC
- Usar mysql para la persistencia de datos

1.4.JUSTIFICACIÓN

La creación de la aplicación web para generar un check-list de tareas de suscripción gratuita tiene como finalidad llevar un mayor control de tareas y actividades del usuario. La interfaz de la aplicación es sencilla y de uso intuitivo debido a que trabaja con cuatro vistas que se detallan a continuación:

- Log in
- Tablas
- Notas
- Acerca de

En la vista login el usuario tendrá la opción de registrarse e iniciar sesión.



En la vista tablas podrá ver, crear, editar y eliminar tablas las cuales se caracterizan por tener título, descripción y un color.

En la vista de notas se visualiza el nombre de la table, descripción y se puede agregar, editar, eliminar y visualizar las notas en forma de lista. Además, se considera clasificar en la pestaña asuntos las tareas para tener una mejor organización.

En la vista acerca de se encontrará información sobre los objetivos de creación de to-do app y los nombres de los desarrolladores.

To-do app pretender ayudar a los usuarios a manejar su tiempo y llevar un control de actividades por categorías en tablas a fin de evitar listas manuales o notas electrónicas que no priorizan importancia y tiempos de finalización. Finalmente, el desarrollo de to-do app ha permitido al equipo desarrollador aplicar los conocimientos obtenidos en el tercer nivel de la Tecnología en Desarrollo de Software.

2. CAPÍTULO II

2.1.MARCO TEÓRICO

2.1.1. Mysql - 5.7

De acuerdo con (W3schools, s.f)

“MySQL es un sistema de gestión de bases de datos relacionales es de código abierto y gratuito. Además, es gratis es ideal para aplicaciones pequeñas y grandes.

Es un sistema rápido, confiable, escalable y fácil de usar. Es multiplataforma y cumple con el estándar ANSI SQL, se distribuye por Oracle Corporation”.



2.1.2. Excalidraw

De acuerdo con (Cruz, 2021)

“Excalidraw es una forma absolutamente magnífica de crear Diagramas Técnicos, tipo Sketch, y con sensación de dibujo a mano. Es una herramienta online, absolutamente gratuita, que te ayuda a conseguir todos esos puntos mencionados anteriormente, pero lo más importante es que el resultado se ve tan limpio, que he notado una gran respuesta por parte de las personas a las que les envío estos dibujos, con buenas palabras siempre a la velocidad de envío, el nivel de detalle, o simplemente el aspecto y la sensación.”

2.1.3. Tomcat - 10.0.8

De acuerdo con (Blanch, y otros, 2017) explican que:

“Apache Tomcat (o, sencillamente, Tomcat) es un contenedor de servlets que se puede usar para compilar y ejecutar aplicaciones web realizadas en Java. Implementa y da soporte tanto a servlets como a páginas JSP (Java Server Pages) o Java Sockets. Además, Tomcat es compatible con otras tecnologías como Java Expression Language y Java WebSocket, del ecosistema Java.

Tomcat puede funcionar de manera autónoma como motor de aplicaciones web desarrolladas con Java, aunque habitualmente se usa en combinación con otros productos como el servidor web Apache, para dar un mayor soporte a tecnologías y aumentar sus características”.

2.1.4. Jakarta – 6.0.0

(BMI, 2021) Menciona que:



“Jakarta crea y mantiene software de código abierto para la plataforma Java. Opera como un proyecto paraguas bajo el auspicio de la *Apache Software Foundation*, y todos los productos producidos por Jakarta son liberados bajo la Licencia Apache.

Jakarta EE ofrece un modelo de código abierto dirigido por la comunidad con releases más frecuentes que Java EE, lo que le permite evolucionar más rápidamente para satisfacer las necesidades de las aplicaciones modernas”

2.1.5. Maven - 4.0.0

“Maven es una herramienta open-source, que se creó en 2001 con el objetivo de simplificar los procesos de build (compilar y generar ejecutables a partir del código fuente)” (Garzas, 2014)

“Maven utiliza un Project Object Model (POM) para describir el proyecto de software a construir, sus dependencias de otros módulos y componentes externos, y el orden de construcción de los elementos. Viene con objetivos predefinidos para realizar ciertas tareas claramente definidas, como la compilación del código y su empaquetado”.

2.1.6. GSON – 2.9.1

“GSON es un API en Java, desarrollada por Google, que se utiliza para convertir objetos Java a JSON (serialización) y JSON a objetos Java (deserialización)”

“**Gson** (también conocido como Google Gson) es una biblioteca de código abierto para el lenguaje de programación Java que permite la serialización y deserialización entre objetos Java y su representación en notación JSON”. (jarroba-, s.f)

2.1.7. Java – 17

De acuerdo con (Java, s.f):

“Java es una plataforma informática de lenguaje de programación creada por Sun Microsystems en 1995. Ha evolucionado desde sus humildes comienzos hasta impulsar una



gran parte del mundo digital actual, ya que es una plataforma fiable en la que se crean muchos servicios y aplicaciones. Los nuevos e innovadores productos y servicios digitales diseñados para el futuro también siguen basándose en Java”

Así mismo (enfasys, 2021) cometa que:

“Java 17 es la última versión de soporte a largo plazo (LTS, por sus siglas en inglés) dentro la cadencia de lanzamiento semestrales de Java y es el resultado de una amplia colaboración entre los ingenieros de Oracle y otros miembros de la comunidad global de desarrolladores de Java a través de la Comunidad Open JDK y el Proceso de la Comunidad Java (JCP, por sus siglas en inglés). Desde el lanzamiento del anterior JDK 11 LTS hace tres años, se han implementado más de 70 JEP”.

2.1.8. Bootstrap – 5.2

(Anónimo, rockcontent, 2020) menciona que:

“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”

2.1.9. Bootstrap-table – 1.21.0

“Las tablas HTML son elementos muy utilizados en las aplicaciones web. Básicamente, las tablas HTML se utilizan para representar datos en filas y columnas dentro de una página web. Lo más recomendable es hacer que las sean responsives para mejorar la experiencia del usuario” (Anónimo, Programación.net, s.f)

2.1.10. Html5

(Anónimo, arkaitzgarro.com, s.f) menciona que:

“**HTML5** (*HyperText Markup Language*, versión 5) es la quinta revisión del lenguaje HTML. Esta nueva versión (aún en desarrollo), y en conjunto con CSS3, define los nuevos estándares de desarrollo web, rediseñando el código para resolver problemas y actualizándolo así a nuevas necesidades. No se limita solo a crear nuevas etiquetas o atributos, sino que incorpora muchas características nuevas y proporciona una plataforma de desarrollo de complejas aplicaciones web (mediante los APIs).

HTML5 está destinado a sustituir no sólo HTML 4, sino también XHTML 1 y DOM Nivel 2. Esta versión nos permite una mayor interacción entre nuestras páginas web y el contenido



media (video, audio, entre otros) así como una mayor facilidad a la hora de codificar nuestro diseño básico”

2.1.11. CSS3

Para (Jiménez, 2019) “CSS es un lenguaje de diseño gráfico que permite definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Es muy usado para establecer el diseño visual de los documentos web e interfaces de usuario escritas en HTML”

“CSS3 es una tecnología que ha tenido una evolución en el tiempo, que actualmente se encuentra en su versión 3, como su propio nombre indica.

Sus siglas corresponden a “Cascading Style Sheets”, que tiene el siguiente significado:

- **Cascading**, que significa que los estilos que aplicamos a **los elementos de una página web se propagan a los elementos que contiene**, se propagan en cascada.
- **Style**, porque mediante CSS lo que hacemos es **aplicar estilos visuales a los distintos elementos** de nuestra página web.
- **Sheets**, que significa hojas, porque **los estilos de una página web se añaden en ficheros aparte**, en ficheros con la extensión .css de manera general.

CSS ha ido evolucionando desde hace 25 años a la versión actual, que es CSS3, pero su función es la misma, sirve para dar estilos visuales a las páginas web”



2.1.12. JavaScript

De acuerdo con (mdn Plus, s.f):

“**JavaScript**[®] (a menudo abreviado como **JS**) es un lenguaje ligero, interpretado y orientado a objetos con funciones de primera clase, y mejor conocido como el lenguaje de programación para las páginas Web, pero también se utiliza en muchos entornos que no son de navegador. Es un lenguaje de scripts que es dinámico, multiparadigma, basado en prototipos y admite estilos de programación orientados a objetos, imperativos y funcionales.

JavaScript se ejecuta en el lado del cliente de la web, y se puede utilizar para estilizar/programar cómo se comportan las páginas web cuando ocurre un evento. JavaScript es un potente lenguaje de scripts y fácil de aprender, ampliamente utilizado para controlar el comportamiento de las páginas web.

(...)

JavaScript puede funcionar como un lenguaje orientado a objetos (en-US) y procedimental. Los objetos se crean mediante programación en JavaScript, adjuntando métodos y propiedades a objetos que de otro modo **en tiempo de ejecución** estarían vacíos, a diferencia de las definiciones de clases sintácticas comunes en lenguajes compilados como C++ y Java. Una vez que se ha construido un objeto, se puede utilizar como plano (o prototipo) para crear objetos similares”

2.1.13. Fetch

Es un método de JavaScript se utiliza para hacer peticiones al servidor como lo menciona (mdn, s.f):



“La API Fetch proporciona una interfaz JavaScript para acceder y manipular partes del canal HTTP, tales como peticiones y respuestas. También provee un método global `fetch()` (en-US) que proporciona una forma fácil y lógica de obtener recursos de forma asíncrona por la red.

Este tipo de funcionalidad se conseguía previamente haciendo uso de `XMLHttpRequest`. Fetch proporciona una alternativa mejor que puede ser empleada fácilmente por otras tecnologías como `Service Workers` (en-US). Fetch también aporta un único lugar lógico en el que definir otros conceptos relacionados con HTTP como CORS y extensiones para HTTP” (Manz, s.f)

“**Fetch** es el nombre de una nueva API para Javascript con la cuál podemos realizar peticiones HTTP asíncronas utilizando promesas y de forma que el código sea un poco más sencillo y menos verbose. La forma de realizar una petición es muy sencilla, básicamente se trata de llamar a `fetch` y pasarle por parámetro la URL de la petición a realizar”

2.1.14. MVC

(desarrollador web, 2020) comenta que:

“En líneas generales, MVC es una propuesta de arquitectura del software utilizada para separar el código por sus distintas responsabilidades, manteniendo distintas capas que se encargan de hacer una tarea muy concreta, lo que ofrece beneficios diversos.

MVC se usa inicialmente en sistemas donde se requiere el uso de interfaces de usuario, aunque en la práctica el mismo patrón de arquitectura se puede utilizar para distintos tipos de aplicaciones. Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.



Su fundamento es la separación del código en tres capas diferentes, acotadas por su responsabilidad, en lo que se llaman Modelos, Vistas y Controladores, o lo que es lo mismo, *Model, Views & Controllers*, si lo prefieres en inglés”

“Modelos

Es la capa donde se trabaja con los datos, por tanto, contendrá mecanismos para acceder a la información y también para actualizar su estado. Los datos los tendremos habitualmente en una base de datos, por lo que en los modelos tendremos todas las funciones que accederán a las tablas y harán los correspondientes *selects, updates, inserts, etc.*

No obstante, cabe mencionar que cuando se trabaja con MCV lo habitual también es utilizar otras librerías como PDO o algún ORM como Doctrine, que nos permiten trabajar con abstracción de bases de datos y persistencia en objetos. Por ello, en vez de usar directamente sentencias SQL, que suelen depender del motor de base de datos con el que se esté trabajando, se utiliza un dialecto de acceso a datos basado en clases y objetos.

Vistas

Las vistas, como su nombre nos hacen entender, contienen el código de nuestra aplicación que va a producir la visualización de las interfaces de usuario, o sea, el código que nos permitirá renderizar los estados de nuestra aplicación en HTML. En las vistas nada más tenemos los códigos HTML y PHP que nos permite mostrar la salida.



En la vista generalmente trabajamos con los datos, sin embargo, no se realiza un acceso directo a éstos. Las vistas requerirán los datos a los modelos y ellas se generará la salida, tal como nuestra aplicación requiera.

Controladores

Contiene el código necesario para responder a las acciones que se solicitan en la aplicación, como visualizar un elemento, realizar una compra, una búsqueda de información, etc.

En realidad, es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestra aplicación. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo.

Arquitectura de aplicaciones MVC

A continuación, encontrarás un diagrama que te servirá para entender un poco mejor cómo colaboran las distintas capas que componen la arquitectura de desarrollo de software en el patrón MVC.

Diagrama MVC

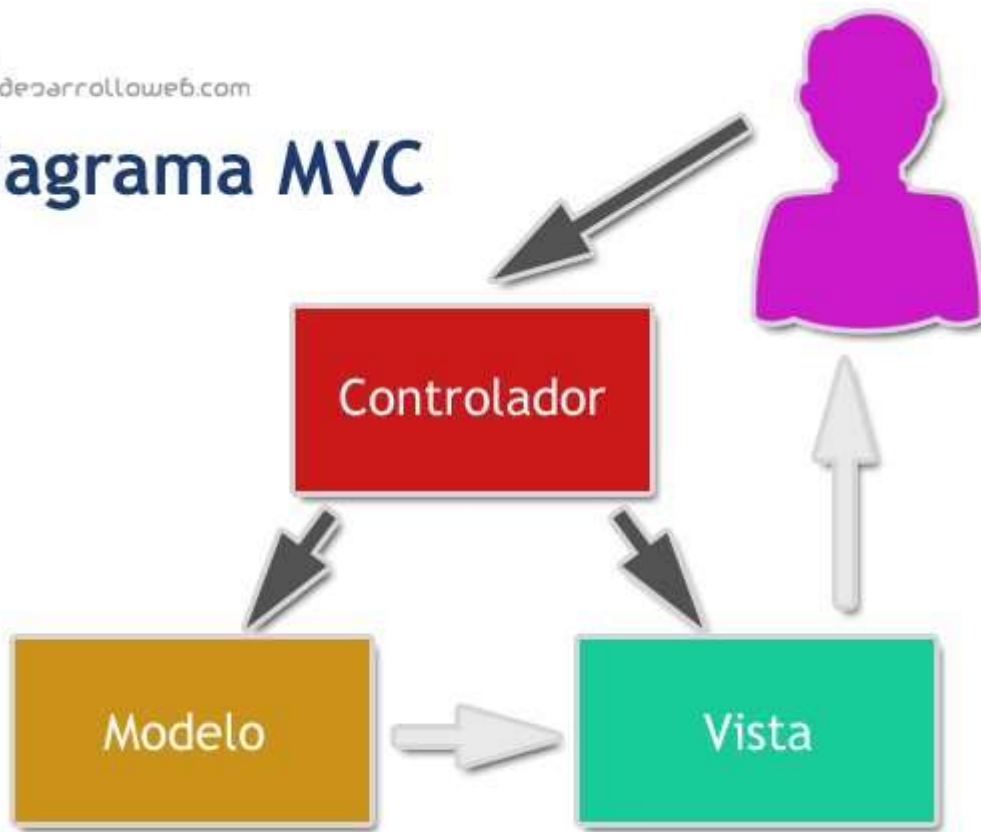


Ilustración 1: Modelo MVC recuperado de: <https://desarrolloweb.com/articulos/que-es-mvc.html>

En esta imagen hemos representado con flechas los modos de colaboración entre los distintos elementos que formarían una aplicación MVC, junto con el usuario. Como se puede ver, los controladores, con su lógica de negocio, hacen de puente entre los modelos y las vistas. Pero además en algunos casos los modelos pueden enviar datos a las vistas. Veamos paso a paso cómo sería el flujo de trabajo característico en un esquema MVC.

1. El usuario realiza una solicitud a nuestro sitio web. Generalmente estará desencadenada por acceder a una página de nuestro sitio. Esa solicitud le llega al controlador.
2. El controlador comunica tanto con modelos como con vistas. A los modelos les solicita datos o les manda realizar actualizaciones de los datos. A las vistas les solicita la salida



correspondiente, una vez se hayan realizado las operaciones pertinentes según la lógica del negocio.

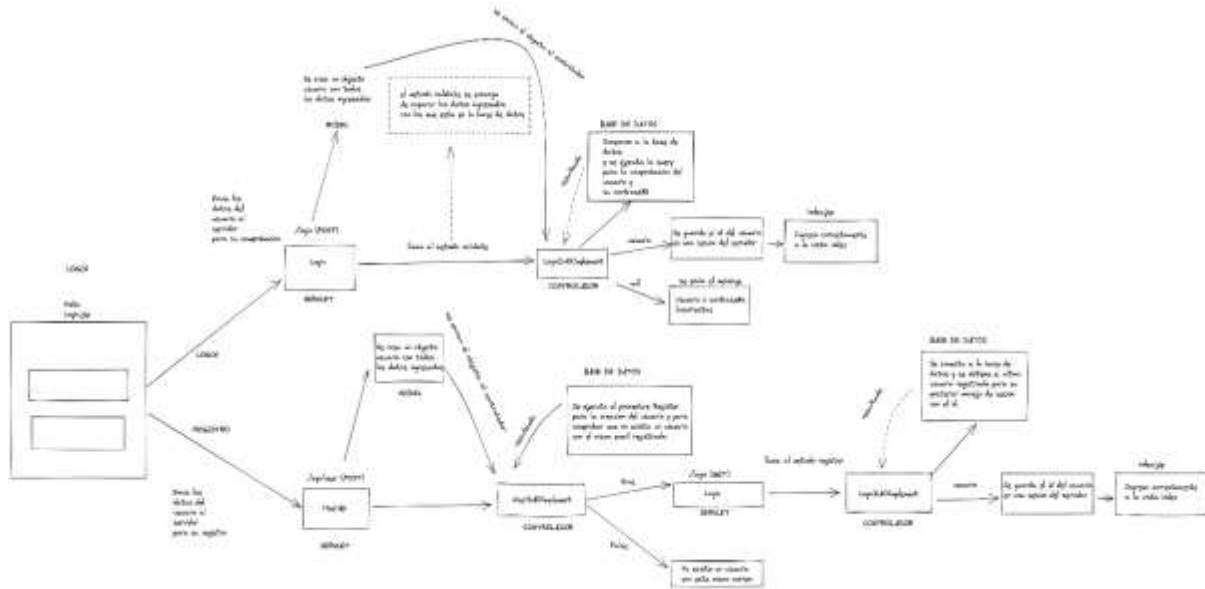
3. Para producir la salida, en ocasiones las vistas pueden solicitar más información a los modelos. En ocasiones, el controlador será el responsable de solicitar todos los datos a los modelos y de enviarlos a las vistas, haciendo de puente entre unos y otros. Sería corriente tanto una cosa como la otra, todo depende de nuestra implementación; por eso esa flecha la hemos coloreado de otro color.
4. Las vistas envían al usuario la salida. Aunque en ocasiones esa salida puede ir de vuelta al controlador y sería éste el que hace el envío al cliente, por eso he puesto la flecha en otro color”



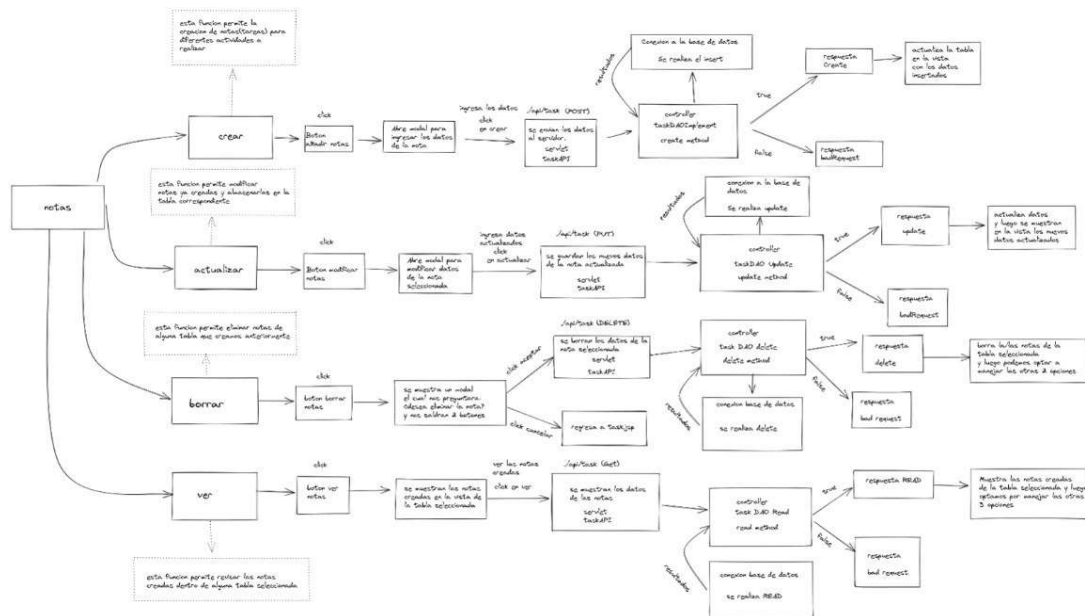
3. CAPÍTULO III

3.1. DESARROLLO Y/O IMPLEMENTACIÓN

LOGIN

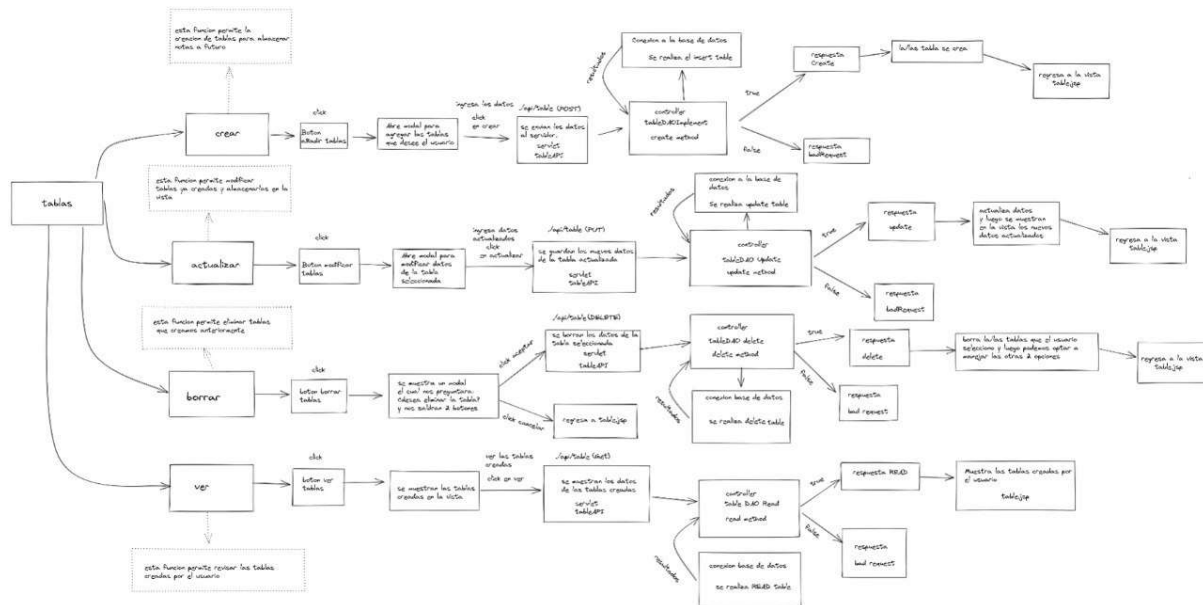


NOTAS





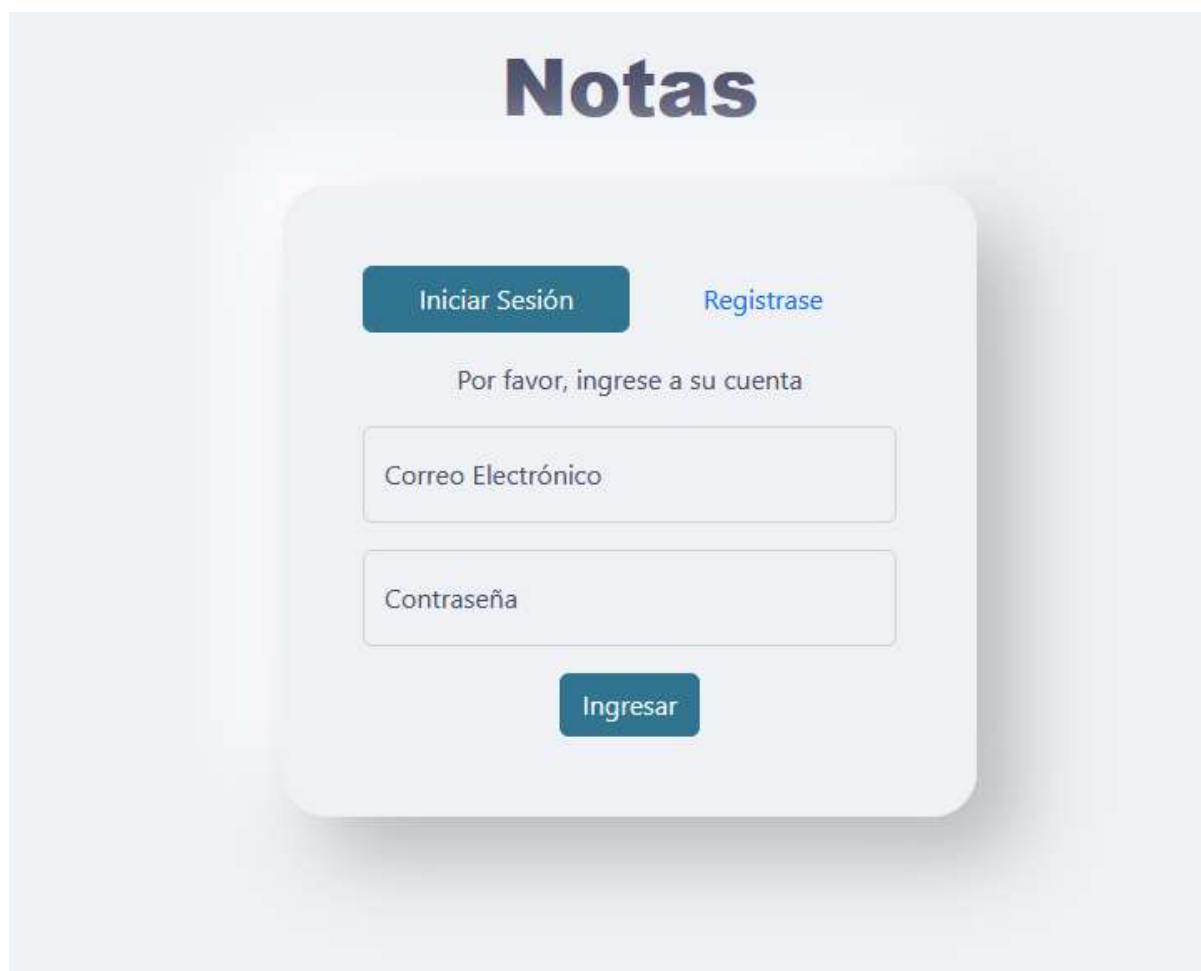
TABLAS





4. CAPÍTULO IV

4.1. ANÁLISIS DE RESULTADOS





Inicio | Perfil | Acciones

Inicio | Tareas

Tareas

Inicio

[+ Agregar Tarea](#) [Actualizar](#)

TAREA	DESCRIPCION	ASUNTO	IMPORTANCIA	FINALIZA	ACCIONES	
PROJ	Proyecto		4	2022-08-08	Modificar	Eliminar

Mostrando desde 1 hasta 1 - En total 1 resultados

Inicio | Perfil | Acciones

Inicio | Acerca de

Acerca de:

Este proyecto

OBJETIVO GENERAL
Diseñar y programar una aplicación web para generar check list de tareas.

OBJETIVOS ESPECÍFICOS

- Diseñar la funcionalidad de la aplicación y de su requerimiento
- Utilizar javascript para realizar tareas en el lado del frontend
- Utilizar html,css y bootstrap para estilizar la aplicación web
- Usar java servlet en el lado del backend aplicando el modelo MVC
- Usar mysql para la persistencia de datos

INTEGRANTES

- Bruni Ramirez Leticia David
- Guernero Luis María Roldán
- Oribe Sanchez Anderson Jairo





BIBLIOGRAFÍA

- Anónimo. (2020). *rockcontent*. Obtenido de Bootstrap: guía para principiantes de qué es, por qué y cómo usarlo: <https://rockcontent.com/es/blog/bootstrap/>
- Anónimo. (s.f). *arkaitzgarro.com*. Obtenido de Capítulo 1 ¿Qué es HTML5?: <https://www.arkaitzgarro.com/html5/capitulo-1.html>
- Anónimo. (s.f). *Programación.net*. Obtenido de Cómo crear tablas responsive en Bootstrap para listar datos: https://programacion.net/articulo/como_crear_tablas_responsive_en_bootstrap_para_listar_datos_1841
- Blanch, A., Fuentes, F., León, M., Lozano, M., García, M., Benito, M. Á., . . . García, F. (2017). *arsys*. Obtenido de ¿Por qué usar Tomcat sobre Servidores Cloud?: <https://www.arsys.es/blog/tomcat-servidores-cloud>
- BMI. (2021). *BMI*. Obtenido de Jakarta EE 8 y Java EE 8 en Liberty: <https://www.ibm.com/docs/es/was-liberty/base?topic=overview-jakarta-java-ee-8-in-liberty>
- Cruz, J. D. (2021). *jorgedelacruz.es*. Obtenido de Excalidraw: <https://www.jorgedelacruz.es/2021/09/20/excalidraw-una-forma-absolutamente-magnifica-de-crear-diagramas-tecnicos-tipo-sketch-y-con-sensacion-de-dibujo-a-mano/>
- David, G. (2022). *PEA*. Quito: ITSQMET.
- desarrollador web. (2020). *desarrollador web*. Obtenido de Qué es MVC: <https://desarrolloweb.com/articulos/que-es-mvc.html>



enfasys. (2021). *enfasys*. Obtenido de Oracle anuncia el lanzamiento de Java 17:

<https://www.enfasys.net/2021/10/05/oracle-anuncia-el-lanzamiento-de-java-17/>

Garzas, J. (2014). *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>

jarroba-. (s.f). *jarroba*. Obtenido de Gson (JSON) en Java, con ejemplos:

<https://jarroba.com/gson-json-java-ejemplos/>

Java. (s.f). *Java*. Obtenido de ¿Qué es la tecnología Java y por qué la necesito? :

https://www.java.com/es/download/help/whatis_java.html

Jiménez, J. D. (2019). *OpenWebinars*. Obtenido de Qué es CSS3 y sus fundamentos:

<https://openwebinars.net/blog/que-es-css3/>

Manz. (s.f). *Lenguaje JS*. Obtenido de <https://lenguajejs.com/javascript/peticiones-http/fetch/>

mdn Plus. (s.f). *mdn web docs*. Obtenido de Acerca de JavaScript:

https://developer.mozilla.org/es/docs/Web/JavaScript/About_JavaScript

mdn. (s.f). *mdn Plus*. Obtenido de

https://developer.mozilla.org/es/docs/Web/API/Fetch_API/Using_Fetch

W3schools. (s.f). *W3schools*. Obtenido de MySQL:

https://www.w3schools.com/mysql/mysql_intro.asp

ANEXOS

