

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SOFTWARE

**DISEÑO DE EXPERIENCIA DE USUARIO DE UN EDITOR
GRÁFICO WEB PARA LA GESTIÓN Y PREPARACIÓN DE
INFRAESTRUCTURA**

**ESTUDIO COMPARATIVO DE FRAMEWORKS DE DESARROLLO
FRONT-END Y BACK-END PARA LA IMPLEMENTACIÓN DE UN
EDITOR VISUAL GRÁFICO WEB PARA EL APROVISIONAMIENTO
DE INFRAESTRUCTURA EN LA NUBE**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE
SOFTWARE**

ELVIS DAVID MONTALUISATOAPANTA

DIRECTOR: CARLOS EFRAÍN IÑIGUEZ JARRÍN Ph.D.

DMQ, agosto 2022

CERTIFICACIONES

Yo, MONTALUISA TOAPANTA ELVIS DAVID declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

MONTALUISA TOAPANTA ELVIS DAVID

Certifico que el presente trabajo de integración curricular fue desarrollado por MONTALUISA TOAPANTA ELVIS DAVID, bajo mi supervisión.

CARLOS EFRAÍN IÑIGUEZ JARRÍN
DIRECTOR

Certificamos que revisamos el presente trabajo de integración curricular.

NOMBRE_REVISOR1
REVISOR1 DEL TRABAJO DE
INTEGRACIÓN CURRICULAR

NOMBRE_REVISOR2
REVISOR2 DEL TRABAJO DE
INTEGRACIÓN CURRICULAR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

MONTALUISA TOAPANTA ELVIS DAVID

CARLOS EFRAÍN IÑIGUEZ JARRÍN

DEDICATORIA

El presente proyecto está dedicado a las personas que me apoyaron durante toda mi etapa académica principalmente mis padres quienes son el motor que me impulsaron a superarme como futuro profesional, a mis amigos quienes me ayudaron en el difícil camino universitario y siempre estuvieron en los buenos y malos momentos.

AGRADECIMIENTO

Agradezco a mi madre por haber estado siempre alentándome a conseguir mi meta profesional, ya que, siempre me llevo por el camino del bien y me enseñó a tener los mejores valores que me ayudaron a no darme por vencido en este duro camino académico.

A mi padre, quien estuvo pendiente de mí y sembró las bases de la responsabilidad que me permitió enfocarme en conseguir mis objetivos.

A mi director de integración curricular Carlos Iñiguez Ph.D. por su paciencia, sabiduría y constante supervisión, sin usted no hubiera sido posible alcanzar fácilmente los objetivos de este proyecto.

ÍNDICE DE CONTENIDO

RESUMEN	X
ABSTRACT	XI
1 INTRODUCCIÓN	1
1.2. Objetivo general	2
1.3. Objetivos específicos	2
1.4. Alcance	2
1.5. Marco teórico	2
1.5.1. Desarrollo de aplicaciones Web	2
1.5.2. Aplicaciones Web.....	3
1.5.3. Tipos de aplicaciones web.....	5
1.5.4. Aprovisionamiento en la computación en la nube.....	6
1.5.5. Tecnologías web actuales	6
1.5.6. Framework para el desarrollo de aplicaciones Web.....	7
1.5.7. Norma ISO/IEC 25010.....	10
1.5.8. Benchmark de aplicaciones.....	11
1.5.9. Biblioteca MxGraph.....	13
2. METODOLOGÍA	16
2.1. Planeación	16
2.1.1. Frameworks a evaluar.....	16
2.1.2. Características y sub-características a evaluar.....	17
2.1.3. Métricas	18
2.1.4. Consideración en la medición de características.....	20
2.2. Elaboración de benchmark.....	20
2.3. Medición de parámetros y generación de resultados	24
2.4. Análisis de resultados	25
2.4.1. Comportamiento temporal	25
2.4.2. Utilización de recursos	26
2.4.3. Pertinencia funcional	27
2.4.4. Completitud funcional	27

3. CONCLUSIONES Y RECOMENDACIONES.....	28
3.1. Conclusiones.....	28
3.2. Recomendaciones.....	29
4. REFERENCIAS BIBLIOGRÁFICAS	30
5. ANEXOS.....	32
ANEXO I. Aplicación Web desarrollada en React.....	32
ANEXO II. Aplicación Web desarrollada en Angular.....	34
ANEXO III. Aplicación Web desarrollada en JQuery.....	35

ÍNDICE DE FIGURAS

Figura 1: Esquema de funcionamiento de la web. Fuente: [11].	4
Figura 2: Frameworks FrontEnd. Adaptado de [16].	9
Figura 3: Frameworks BackEnd. fuente: [19].	9
Figura 4: Características de la norma ISO/IEC 25010. Fuente:[20].	10
Figura 5: Etapas de bechmarking.	13
Figura 6: Funcionamiento de MxGraph. Fuente: [26].	15
Figura 7: Frameworks a evaluar.	16
Figura 8: Diagrama de navegación.	20
Figura 9: Boceto crear cliente.	21
Figura 10: Boceto actualizar cliente.	21
Figura 11: Boceto eliminar cliente.	22
Figura 12: Esquema de componentes de la aplicación web.	23
Figura 13: Editor de código Visual Studio Code.	23
Figura 14: Plataforma Firebase.	24
Figura 15: Gráfico de promedio de tiempo (milisegundos) del comportamiento temporal.	26
Figura 16: Gráfico de la utilización de recursos.	27
Figura 17: Ejemplo FronEnd React.	32
Figura 18: Formulario para crear un cliente.	33
Figura 19: Editar cliente.	33
Figura 20: Ejemplo de FrondEnd framework Angular.	34
Figura 21: Interfaz para agregar un cliente.	34
Figura 22: Interfaz editar cliente.	35
Figura 23: Ejemplo FrondEnd jQuery.	35
Figura 24: Interfaz crear cliente.	36
Figura 25: Interfaz editar cliente.	36

ÍNDICE DE TABLAS

Tabla 1: Tecnologías FrondEnd. Fuente: [17].	7
Tabla 2: Tecnologías BackEnd. Fuente: [15].	7
Tabla 3: Paquetes de funcionalidad de MxGraph. Fuente: [27]	15
Tabla 4: Característica eficiencia de desempeño.	18
Tabla 5: Característica adecuación funcional.	19
Tabla 6: Resultados de medición: característica eficiencia de desempeño.	24
Tabla 7: Medición característica adecuación funcional.	25
Tabla 8: Promedio de tiempo (milisegundos) del comportamiento temporal por cada framework.	25
Tabla 9: Análisis característica pertinencia funcional.	27
Tabla 10: Análisis característica completitud funcional.	28

RESUMEN

El desarrollo de aplicaciones web se ha convertido en un requerimiento demandado por empresas y personas quienes tienen la necesidad de publicitar sus productos o servicios al mercado de una forma diferente e innovadora. En este sentido, agilizar el desarrollo de aplicaciones web se ha convertido en un tema de gran interés para la comunidad de desarrolladores web. Actualmente, existe un amplio abanico de frameworks que soportan el desarrollo front-end y back-end de aplicaciones web. Un framework es un conjunto de herramientas que simplifica y agiliza el proceso de desarrollo, asegurando buenas prácticas, evitando la repetición de código y manteniendo consistente el código. Seleccionar los mejores frameworks, previo a la implementación del proyecto, se convierte en una tarea clave que asegura el éxito de un proyecto.

El presente trabajo, enmarcado en el proyecto macro de implementación de un editor gráfico web que permita a los operadores de TI gestionar la infraestructura en la nube de forma interactiva, tiene como objetivo definir qué frameworks son técnicamente adecuados para la implementación del editor. Para esto, este trabajo presenta un estudio de Benchmark, donde se analiza y compara diferentes frameworks e identifica aquellos que mejor se adaptan a las necesidades técnicas involucradas en la implementación del editor gráfico. El resultado del Benchmark permitirá a los desarrolladores conocer el framework más idóneo para implementar el editor gráfico.

PALABRAS CLAVE: desarrollo web, infraestructura, análisis, editor web.

ABSTRACT

The development of web applications has become a requirement demanded by companies and people who need to advertise their products or services to the market in a different and innovative way. In this sense, speeding up the web applications development has become a topic of great interest for the web developer community. Currently, there are a wide range of frameworks that support the front-end and back-end web applications development. A framework is a set of tools that simplifies and speeds up the development process, ensuring good practices, avoiding repetition of code, and keeping the code consistent. Selecting the best frameworks, prior to the implementation of the project, becomes a key task that ensures the success of a project.

The present work, framed in the macro project for implementing a graphical web editor that allows TI operators to manage the infrastructure in the cloud interactively, aims to define which frameworks are technically adequate for the implementation of the editor. This work presents a benchmark study, where different frameworks are analyzed and compared. After that, those that best adapt to the technical needs involved in the implementation of the graphical editor are identified. The result of the benchmark will allow developers to know the most suitable framework to implement the graphical editor.

KEY WORDS: web development, infrastructure, analysis, web editor.

1 INTRODUCCIÓN

Una fase previa al desarrollo de cualquier aplicación software es la identificación de potenciales herramientas para su implementación. El estudio de herramientas, conocido como Bechmarking, permite a un equipo de desarrollo definir la o las herramientas que brindan mayor eficiencia en el trabajo, eficacia para conseguir los objetivos de desarrollo y que aportan a la mejor organización del equipo de desarrollo. La carencia de este estudio previo puede significar el éxito o fracaso en la implementación de la aplicación software.

Con el advenimiento de la computación en la nube o “cloud”, el aprovisionamiento o preparación de infraestructura se ha convertido en una tarea frecuente para los operadores de TI tanto en la industria como en la academia [1]. Aprovisionar infraestructura consiste en definir/installar infraestructura de TI (servidor, red, servicios, etc.) [2]. Para soportar un proyecto tecnológico, solo que en lugar de instalar la infraestructura de forma física como se lo hacía en épocas pasadas, hoy los operadores de TI gestionan/installan la infraestructura de manera virtual y automática usando portales web o consolas de líneas de comandos (CLI) proporcionadas por proveedores de infraestructura [3], como por ejemplo Amazon WS o Azure [4]. A pesar de la reducción de esfuerzo y espacio físico que supone aprovisionar infraestructura hoy en día, esta tarea no deja de tener complejidades. El operador de TI requiere conocer los comandos de CLI precisos para definir la infraestructura, además de lidiar con las características particulares de la infraestructura de cada proveedor.

En este contexto, la creación de un Editor Gráfico intuitivo accesible por la Web, que permita a los operadores de TI una interacción más fluida y menos compleja que la experimentada con el uso del CLI [4], se ha convertido en una necesidad. Sin embargo, previa a su implementación, es necesario seleccionar las herramientas tecnológicas a utilizar para su creación. La selección de una herramienta adecuada facilitará la implementación del Editor Visual Gráfico Web para el aprovisionamiento de infraestructura en la nube, permitiendo una adecuada interacción con el operador TI [5], de lado del FrontEnd (interfaz de usuario), así como el adecuado rendimiento y procesamiento, de lado del BackEnd (lado del servidor) [6]. En consecuencia, un estudio de Bechmarking permitirá evaluar y analizar las herramientas de desarrollo Web existentes en el mercado permitiendo definir aquellas herramientas que más se ajustan a las particularidades de implementación del editor [7].

En este sentido, el desarrollo de este componente se enfocará en realizar un Benchmarking de herramientas existentes con el fin de definir aquellas que son más adecuadas para la implementación del editor Web [7].

1.2. Objetivo general

- Realizar un Benchmarking de herramientas de desarrollo web para la implementación del editor gráfico para la gestión y preparación de infraestructura.

1.3. Objetivos específicos

- Analizar las herramientas de desarrollo web existentes en el mercado.
- Evaluar las herramientas de acuerdo con las necesidades de los usuarios.
- Tomar decisiones en función de las herramientas más adecuadas para el desarrollo del editor web.

1.4. Alcance

El objetivo del proyecto involucra analizar y evaluar herramientas de desarrollo web adecuadas para la implementación de un editor visual web para el aprovisionamiento de infraestructura. Para esto, se seguirá la metodología de Benchmarking Robert C. Camp, este consiste en cinco etapas: planeación, elaboración de benchmark, medición de parámetro y generación de resultados, análisis de resultados y conclusiones y recomendaciones.

1.5. Marco teórico

1.5.1. Desarrollo de aplicaciones Web

El desarrollo web se ha convertido actualmente en una profesión de grandes oportunidades laborales. El desarrollo web es la profesión que más ha experimentado cambios (evolución) debido a la demanda incremental de aplicaciones web por parte de la industria. En este sentido, crear páginas web con grandes diseños y programar su funcionamiento se convierte en un gran mundo por descubrir y apetecible por muchos desarrolladores quienes

están motivados por descubrir cómo funciona el desarrollo web, para qué sirve y cuáles son los tipos de desarrollo web existentes [8]. En consecuencia, prepararse profesionalmente como desarrollador web brinda grandes oportunidades laborales debido a su alta demanda y a su constante crecimiento.

La web, un desafío de conocimiento para el desarrollador a nivel de creación y mantenimiento de sitios web supone que los desarrolladores conozcan una amplia gama de acciones como la codificación de los sitios, su diseño y la gestión de los contenidos de la web, realizados mediante la interacción constante entre el servidor web y el cliente web [9]. En la práctica, los desarrolladores web deben tener un amplio conocimiento de tecnologías estándar de la plataforma Web (HTML, CSS, JavaScript), así como lenguajes de programación orientados a la web, como por ejemplo JAVA, PHP, etc. Usando las tecnologías y lenguajes de programación, el desarrollador está desafiado a lograr un sistema práctico, ágil y dinámico, funcional y con diseño sea llamativo a la vez que sea fácil de usar. Es así, que los desarrolladores se convierten en piezas vitales para el desarrollo web [9].

Entonces, un desarrollador web debe una persona que se adapte a cualquier entorno de trabajo, debe acoplándose a los cambios y debe estar en constante formación en nuevos conocimientos, ya que, esta es la manera más adecuada de estar actualizado en las nuevas tecnologías que van surgiendo.

El conocimiento sobre desarrollo web hace posible que este pueda ser utilizado para implementar aplicaciones web, en donde dichas aplicaciones son desarrolladas como solución a las necesidades que presentan los clientes, no obstante, el tema de aplicaciones web será desarrollado en la siguiente sección, en donde se conocerá más a profundidad sobre su funcionamiento, aplicación y ventajas.

1.5.2. Aplicaciones Web.

Se considera como aplicaciones web a los programas capaces de funcionar a través de Internet, es decir, la interacción que se genera entre el usuario y la web es procesada y almacenada en servidores capaces de alojar a la aplicación web.

Una de las ventajas de utilizar aplicaciones web es que no necesitan ser descargadas en la computadora para funcionar, basta con tener un navegador para poder utilizarla [10]. Es por esto por lo que generalmente, el término “aplicación web” está relacionado al almacenamiento en la nube, es decir, al realizar una consulta sobre una determinada

información requerida por el cliente, esta petición es buscada, a través de internet, en los servidores alojados en la nube. Dichos servidores, que almacenan la información de forma permanente, envían la información solicitada como una respuesta a la petición formulada por el cliente, quien es capaz de visualizar dicha información en su navegador web.

La Figura 1 ilustra el funcionamiento, desde que el cliente web realiza la petición hasta cuando el cliente recibe la información por parte del servidor web. Utilizando la dirección única o URL de la página buscada, el cliente web envía (a través de Internet) la petición hacia el servidor. El servidor realiza la búsqueda dentro de su almacenamiento de información en búsqueda de la página web solicitada y, una vez encontrada, esta es enviada a través del mismo medio (Internet) hacia el navegador web, mismo que muestra la página solicitada [11].

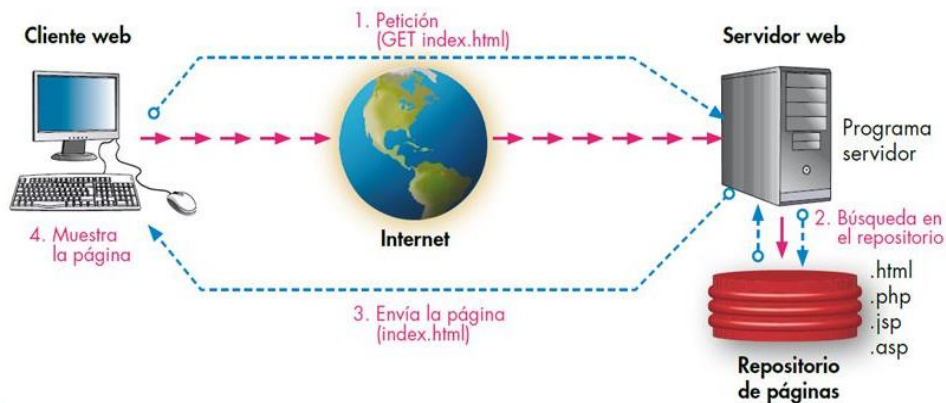


Figura 1: Esquema de funcionamiento de la web. Fuente: [11].

Las aplicaciones web en la nube son aplicaciones que funcionan a través de Internet mediante un servidor, estos servidores se encargan de facilitar gran parte de su funcionamiento, es decir, que los recursos que consume el cliente son recursos (hardware) de los servidores, en lugar de consumir los recursos locales de la computadora de los clientes.

En cuanto a su desarrollo, las aplicaciones web generalmente se dividen en dos partes: FrontEnd y BackEnd. El FrontEnd es la parte visual con la que los usuarios interactúan con la aplicación web haciendo uso del navegador web. El BackEnd es la lógica programable que interactúa con el servidor y la base de datos.

Entonces, se puede concluir que el desarrollo de aplicaciones web conlleva aplicar sus partes (FrontEnd y BackEnd) como un todo para que el sistema integre toda la funcionalidad completa, a fin de que, el sistema cumpla con las necesidades de los clientes.

1.5.3. Tipos de aplicaciones web

Internet es un medio que contiene información, entretenimiento, formación educativa, contenido multimedia, etc. Las aplicaciones web son el medio para acceder a todo este contenido que puede ser estático o dinámico. Las aplicaciones web presentan la información de forma visual y ordenada a fin de que el usuario logre comprender dicha información [12]. En función de como se muestra el contenido a los usuarios y como es la gestión de los contenidos con la aplicación, se considera los siguientes tipos de aplicaciones web:

- **Aplicación web estática:** Este tipo de aplicación web se utiliza cuando el contenido a presentar es mínimo y no se contempla realizar demasiados cambios en el contenido. Generalmente, suelen estar diseñadas en HTML para estructurar el contenido y CSS para agregar estilo visual al contenido.
- **Aplicación web dinámica:** Desarrollar este tipo de aplicación web se considera técnicamente más complejo que desarrollar una aplicación web estática. La información que se presenta al usuario es cargada desde una base de datos, permitiendo que los contenidos se actualicen cada vez que el usuario realiza una petición de acceso. Su desarrollo implica conocimiento de lenguajes de programación como JavaScript, Python, Java, PHP, etc.
- **E-commerce:** Es un tipo de aplicación web que virtualiza una tienda o comercio electrónico. Su desarrollo es más complejo que los dos tipos de aplicaciones web anteriores ya que es una aplicación web que incorpora una pasarela de pagos para gestionar las compras de la tienda, un módulo de logística de inventarios, así como de entregas.
- **Portal web:** Este tipo permite la creación de aplicaciones web destinadas a foros, chats, correo electrónico, entre otros.

1.5.4. Aprovisionamiento en la computación en la nube

La nube o computación en la nube (Cloud Computing), es un término que se popularizó al final de la década de los 2000, tomando fuerza a partir del año 2010. El organismo National Institute of Standards and Technology (NIST) define al “cloud computing” como “un conjunto de hardware y software, almacenamiento, servicios e interfaces que facilitan la entrada de información como un servicio” [13]. Así, la nube se convierte en un servicio de consumo con grandes ventajas, como, optimización de recursos de hardware e implementación de los servicios (Software) en los servidores, logrando proveer estos servicios mediante la red. Como herramienta, la nube ayuda a los técnicos de TI a reducir el gasto en recursos de hardware y software, ya que, toda la infraestructura es proporcionada por los proveedores de servicios en la nube (Amazon Web Services o AWS, Microsoft Azure, Google Platform, IBM, Oracle, etc), permitiendo tener mayores ventajas como alta escalabilidad, disponibilidad y rendimiento. Los proveedores proporcionan o aprovisionan servicios, de ahí que, en términos técnicos, *aprovisionamiento en la nube* significa asignar a un cliente los recursos y servicios que tiene el proveedor de servicios[14]. Es decir, el aprovisionamiento en la nube es un modelo funcional y característico de la computación en la nube, utilizado por los clientes que contratan los servicios y recursos de la nube.

En la nube, los clientes pueden requerir servicios de equipos hardware, software, etc. En este sentido, los servicios disponibles en la nube cubren estas necesidades y se clasifican en tres grupos principales: Infraestructura como servicio (IaaS), Software como servicio (SaaS), Plataforma como servicio (PaaS).

1.5.5. Tecnologías web actuales

Las tecnologías se actualizan constantemente, siendo necesario estar al día con los conocimientos adecuados a media que pasa el tiempo. Trabajar en diversos proyectos creando interfaces, conectando con bases de datos y programando la aplicación web, demanda gran entendimiento y dominio de estas tecnologías [15]. En esta sección se describe las tecnologías web actuales teniendo en cuenta sus características y ventajas, dependiendo del tipo de aplicación web que se desee implementar.

La Tabla 1 muestra las tecnologías actuales y más utilizadas según el sitio web StackOverflow para el año 2022 [16], en donde se publica estadísticas sobre temas de programación, la encuesta se aplicadas a un gran número de personas (Estudiantes,

profesores y desarrolladores profesionales) a fin de conocer que tecnologías se utilizan más, como tecnologías del lado del cliente (FronEnd):

Tabla 1: Tecnologías FrondEnd. Fuente: [17].

Nombre	Característica
JavaScript	Es un lenguaje de programación multiplataforma. Se puede emplear para la creación de aplicaciones web dinámicas e interactivas.
HTML	Es un lenguaje de marcado empleado para estructurar el contenido en las páginas web.
CSS	Es un lenguaje de marcado dirigido a la apariencia de la página web.

Por el lado del servidor, la Tabla 2 muestra las tecnologías más utilizadas según el sitio web StackOverflow para el año 2022 [16]:

Tabla 2: Tecnologías BackEnd. Fuente: [15].

Nombre	Característica
Python	Es un lenguaje sencillo, pero potente. Una de sus características es que usa menos líneas de código en comparación a otros lenguajes de programación.
PHP	Es un lenguaje interpretado de fácil uso y utilizado generalmente en aplicaciones del tipo Content Management System.
C#	Es un lenguaje sencillo, con una sintaxis fácil de aprender, con seguridad de tipos y orientado a objetos.

1.5.6. Framework para el desarrollo de aplicaciones Web

Un framework, es una herramienta preconfigurada que es utilizada para construir aplicaciones software, partiendo de una configuración incompleta a la cual se puede añadir las demás piezas que completan la aplicación. Usar un framwork de desarrollo implica usar una estructura de software que se basa en componentes que pueden ser personalizados e intercambiados [18].

Los frameworks son utilizados en el desarrollo de aplicaciones software de todo tipo (médicas, desarrollo de video juegos, etc). Algunos de los objetivos principales que se alcanzan al utilizar los frameworks son:

- **Acelerar el proceso de desarrollo:** Inicialmente es una aplicación incompleta que se puede modelar a las necesidades de la aplicación que se está desarrollando.
- **Reutilizar código:** Adaptar un código que ya existe a la aplicación, reduce el tiempo de desarrollo.
- **Promover buenas prácticas de desarrollo:** Este objetivo se refiere a tener políticas y reglas de programación, además de utilizar patrones de diseño.

En el campo del desarrollo Web, los frameworks permiten el desarrollo ágil de aplicaciones, debido a que, por lo general consisten en un conjunto de módulos, que permiten, al desarrollador centrarse solo en el problema que desea resolver en lugar de gastar tiempo en implementar la lógica de programación que ya pudo haber sido implementada en otra aplicación y con lo cual se puede reutilizar el código.

La Figura 2 muestra el diagrama de barras de los frameworks para el desarrollo Web más utilizados, según con el sitio web StackOverflow. Para el año 2022 los frameworks FrontEnd más utilizados son: React.js, jQuery y Angular [16]. Estos frameworks mencionados serán tomados en cuenta para el desarrollo del proyecto debido a su popularidad en uso y porque ocupan un gran porcentaje del mercado para desarrollo web del lado del cliente.

En la Figura 2 se puede observar que el framework React está en primer lugar con un porcentaje de uso del 42,62%, seguido de jQuery con un porcentaje de 28,57% y, por último, está el framework Angular con un porcentaje de 20,39%. Estos porcentajes indican claramente que los tres frameworks están presentes en la mayoría de los proyectos de desarrollo de aplicaciones.

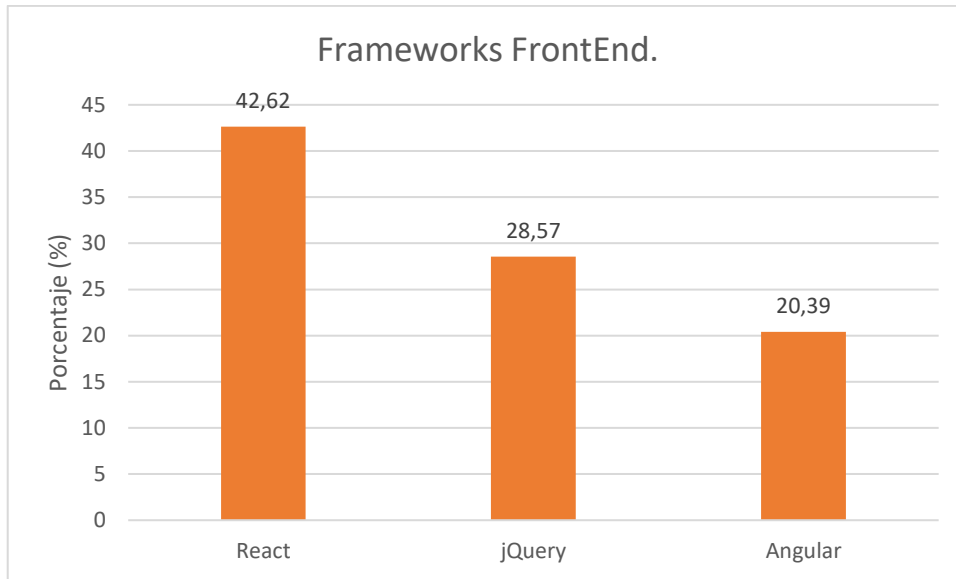


Figura 2: Frameworks FrontEnd. Adaptado de [16].

De acuerdo con los datos estadísticos del sitio web Statistics&Data, mostrado en la Figura 3, para el año 2022 los frameworks BackEnd más utilizados son: Laravel, Django y Flask [19]. Estos tres frameworks serán tomados en cuenta para el desarrollo del proyecto, ya que según el sitio web Statistics&Data se observa que ocupan un gran porcentaje del mercado en desarrollo web del lado del servidor.

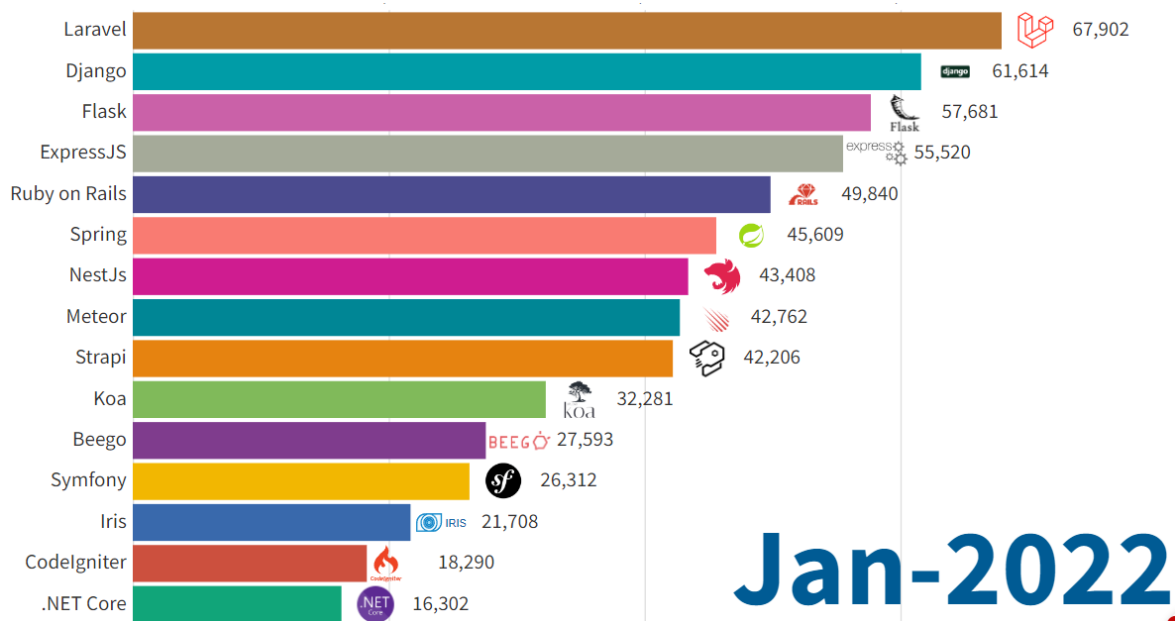


Figura 3: Frameworks BackEnd. fuente: [19].

1.5.7. Norma ISO/IEC 25010

La ISO/IEC 25010 es el modelo de calidad que se considera como la piedra angular dentro del modelo de calidad, puesto que, en esta norma se establece como se evalúa la calidad de un producto.

Para evaluar la calidad del producto software existe una serie de características que la norma contempla a la hora de evaluar las propiedades de un producto software, por lo tanto, mediante estas características se puede interpretar a la calidad de un producto como el grado en que este producto satisface las necesidades de un usuario, en la Figura 4 podemos observar que existen ocho características propuestas por la ISO/IEC 25010 para evaluar la calidad de un producto [20].



Figura 4: Características de la norma ISO/IEC 25010. Fuente:[20]

Para el desarrollo de este proyecto se usarán las dos primeras características que son: adecuación funcional y eficiencia de desempeño [20], con el objetivo de, usar estas características para poder definir criterios y evaluar un producto software, a continuación, se explica en que consiste cada una de estas dos características.

Adecuación Funcional

Esta característica evalúa la capacidad que tiene un producto software en satisfacer las necesidades de los usuarios siempre y cuando el producto sea utilizado bajo ciertas condiciones. Cada característica se divide en sub-características mencionadas a continuación.

- **Completitud funcional:** Es el grado en el que las funcionalidades de un software abarcan las tareas y objetivos especificadas por los usuarios.

- **Corrección funcional:** Es la capacidad de un producto software para proveer resultados de forma correcta con un nivel preciso requerido.
- **Pertinencia funcional:** Es la capacidad de un producto software para proporcionar funciones para realizar tareas, en consecuencia, cumplir con los objetivos del usuario.

Eficiencia de desempeño

Esta característica evalúa la cantidad de recursos utilizados por el software bajo ciertas condiciones, esta característica se divide en las siguientes sub-características.

- **Comportamiento temporal:** Es la cantidad de tiempo empleado en el procesamiento y respuesta del software, es decir, evalúa el tiempo que demora estas funciones en ejecutarse.
- **Utilización de recursos:** Es la cantidad de recursos que utiliza el software cuando ejecuta sus funciones.
- **Capacidad:** Es el grado en que los límites máximos de un software cumplen con los requisitos especificados.

1.5.8. Benchmark de aplicaciones

Benchmark es una palabra utilizada para realizar comparativas de rendimiento, a fin de conocer qué sistemas, políticas, empresas o iniciativas tienen un mejor comportamiento [21].

Para realizar esta comparativa de rendimiento, este debe complementarse con el proceso conocido como benchmarking, el cual, establece un proceso sistemático de forma continua, a fin de, evaluar productos, servicios o procesos de trabajo.

Un ejemplo determinante que logró la excelencia al aplicar el concepto de benchmarking es la empresa Xerox, ya que, esta empresa al estar atravesando una grave crisis a finales de la década de 1970 logró superar esta crisis, gracias a que, los ejecutivos de la época adoptaron el proceso de benchmarking, para realizar un estudio comparativo que identifican los procesos necesarios que la empresa debía adoptar respecto a sus competidores directos. Por ejemplo, este estudio identificó que la producción debía crecer un 18% cada año durante los 5 primeros para conseguir alcanzar a sus rivales [22].

Este ejemplo de análisis comparativo permite conocer cómo el benchmarking resultó extremadamente útil para la empresa Xerox, puesto que, esto le permitió reaccionar a tiempo y trazar nuevos objetivos, por consiguiente, la empresa llegó a ganar el Premio Nacional de Calidad Malcom Baldrige [23].

Consecuentemente, el término tuvo mayor repercusión cuando fue aplicado dentro del área de fabricación de la empresa Xerox, publicando los resultados de las primeras experiencias en el libro de Camp de 1989 [24].

Para este proyecto se aplica una adaptación del benchmarking de Camp consistente en 5 etapas (ver Figura 5) ajustadas al estudio de herramientas en el área de software [25].

1. **Planeación:** Esta etapa está destinada a realizar la planificación de la investigación (sobre lo que se realizará), es decir esta planificación debe responder a las siguientes preguntas: qué queremos medir y cómo lo vamos a hacer.

Se establecen las respectivas características y sub-características que se evaluarán.

2. **Elaboración de benchmark:** En esta etapa se procederá con el desarrollo de una aplicación web para cada framework, a fin de evaluar las características y sub-características.
3. **Medición de parámetros y generación de resultados:** El objetivo de esta etapa es obtener los resultados al medir cada característica y sub-característica que fue propuesta en la etapa de elaboración.
4. **Análisis de resultados:** Interpretar los datos obtenidos y realizar el análisis de los resultados.
5. **Conclusiones y recomendaciones:** Se utilizan los datos analizados en la etapa anterior para emitir un resultado que determine la elección adecuada del framework web para implementar el editor gráfico.

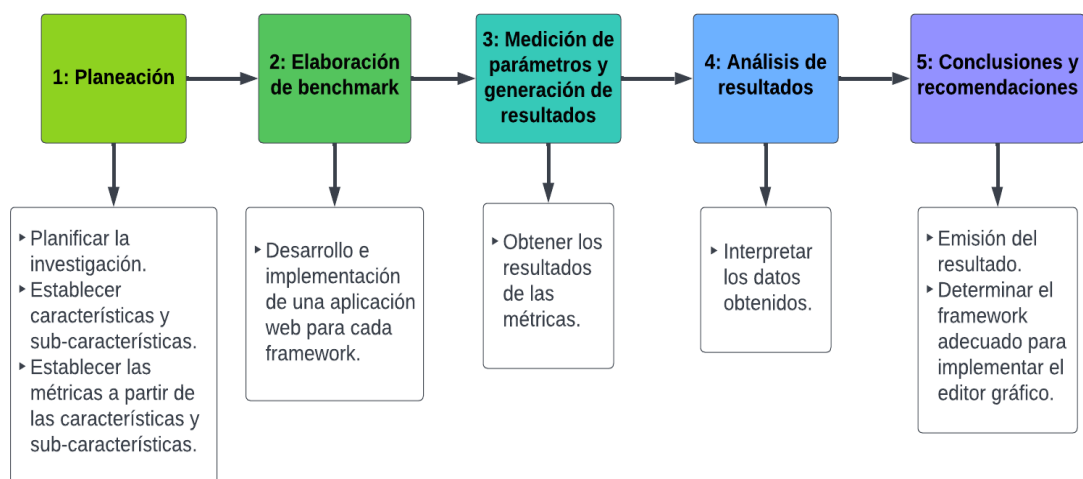


Figura 5: Etapas de bechmarking.

1.5.9. Biblioteca MxGraph

MxGraph es una biblioteca JavaScript que provee soporte para el despliegue de gráficos y diagramas en aplicaciones web. A través de esta biblioteca, es posible crear, diseñar o editar todo tipo de diagramas como diagramas de flujo, diagramas de red y gráficos de forma general en páginas web. La biblioteca está compuesta por una serie de programas de FrontEnd implementados en lenguaje JavaScript que permiten la creación de diagramas de forma sencilla. Además, provee múltiples ejemplos de integración para programas del lado del servidor (backEnd) [26].

Con esta biblioteca, se puede crear y personalizar gráficos o diagramas, ya que la biblioteca permite generar una interacción con el usuario el cual puede diseñar diversos diagramas o gráficos de forma rápida, además el usuario puede copiar, pegar, borrar, arrastrar, conectar, modificar su tamaño, la forma de los elementos y sus estilos gráficos. Entonces las ventajas que el usuario adquiere al utilizar la biblioteca son múltiples, ya que las interacciones hacen sencillo que cualquier usuario aprenda a crear todo tipo de diagramas.

Una de las ventajas de utilizar esta biblioteca es la vectorización y la compatibilidad con navegadores. Los gráficos se ejecutan de forma nativa en cualquier navegador [27]. Otra ventaja que tiene esta biblioteca es que no es necesario adquirir algún tipo de licencia, debido a que es gratuita y libre para cualquier usuario, de esta forma se puede integrar en cualquier proyecto de desarrollo web. Además, la biblioteca se caracteriza por otorgar

flexibilidad y robustez; ventaja que ha hecho que grandes empresas utilicen esta biblioteca, por ejemplo, el sitio web Draw.io [28].

Ejemplo de implementación

Draw.io es una aplicación implementada con la biblioteca Mxgraph, esta aplicación tiene características para la creación de diagramas ofreciendo un entorno de diseño gráfico intuitivo para los usuarios, ya que la aplicación ofrece las interacciones para arrastrar y soltar los objetos, iconos y diferentes gráficos geométricos, además cada gráfico puede ser personalizado en su forma y tamaño [29].

Funcionamiento de la biblioteca

MxGraph es un componente gráfico que proporciona una interfaz para páginas web, la Figura 6 muestra la arquitectura del funcionamiento de este componente, la cual consta de tres capas: web client, web server y backEnd. El funcionamiento de MxGraph radica en la integración de estas capas.

- Web client, esta capa es un componente gráfico provisto de un contenedor compatible con la interfaz gráfica de la aplicación web.
- Web server, esta capa se comunica directamente con el web client la cual necesita de un servidor que provee archivos (Html, Xml, Js, Css, etc.) necesarios para su funcionamiento. En e caso de no disponer de un servidor se puede utilizar el sistema de archivos local para proveer los archivos.
- BackEnd, esta última capa gestiona la lógica del funcionamiento para ser integrada en la aplicación web. El backEnd puede ser implementado en cualquiera de los tres lenguajes de programación soportados: Java, .Net y Php.

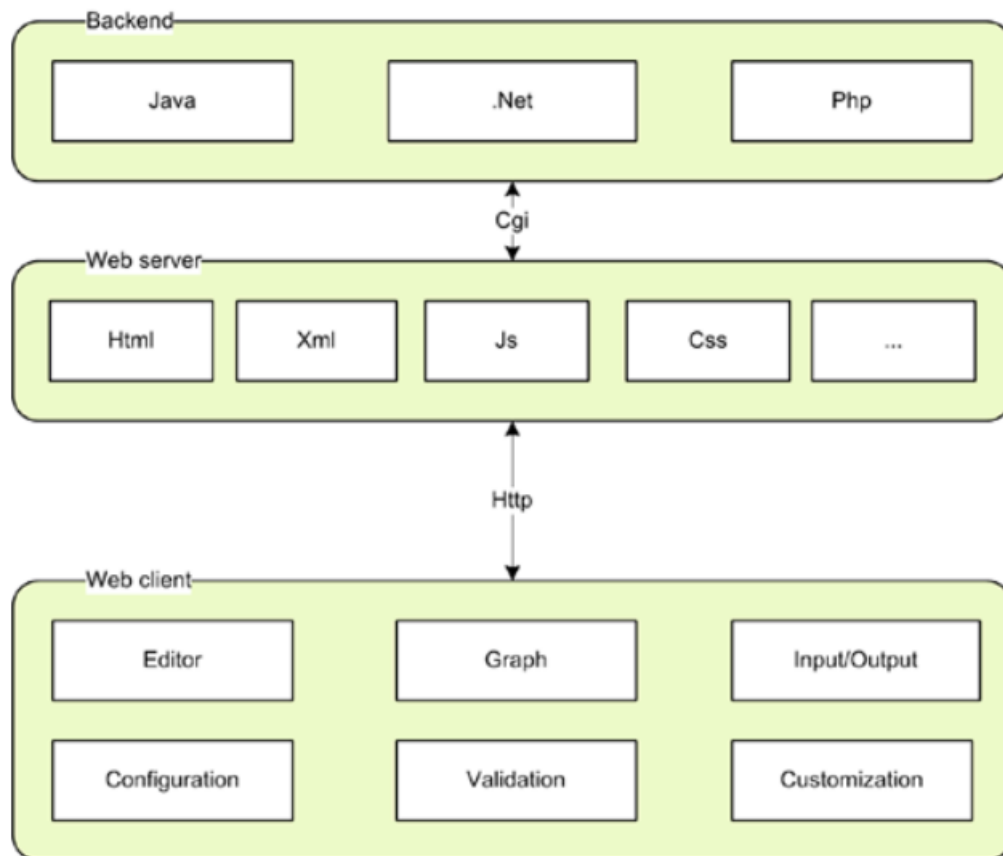


Figura 6: Funcionamiento de MxGraph. Fuente: [26]

MxGraph proporciona a los desarrolladores un Core completo (frontEnd y backEnd) para la creación de aplicaciones de diseño gráfico que funcionan en entornos web. La biblioteca destaca puesto que, únicamente utiliza JavaScript como lenguaje de programación, todo esto sin depender de librerías de terceros lo que libera a la biblioteca de la dependencia de proveedores de complementos adicionales, esto es considerado de vital importancia a la hora de elegir esta biblioteca para implementar proyectos de diseño gráfico de diagramas.

La estructura de la biblioteca consiste en ocho paquetes que definen su funcionalidad. Estos paquetes de funcionalidad están descritos en la siguiente Tabla 3.

Tabla 3: Paquetes de funcionalidad de MxGraph. Fuente: [27]

Paquete	Descripción
editor	Proporciona las clases necesarias para crear un editor de diagramas.
view y model	Proporcionan los elementos necesarios para crear y dibujar los elementos en el diagrama.

handler, layout y shape	Integran los manejadores de eventos y la forma en la que se distribuyen los elementos y las formas de las figuras.
util	Integra las utilidades necesarias para la interacción de arrastrar y soltar los elementos gráficos.
io	Proporciona las diferentes clases para codificar y decodificar los diagramas en XML.

2. METODOLOGÍA

En este capítulo se utilizará la metodología de benchmarking descrita Robert C. Camp mencionada en el marco teórico (sección 1.5.8). En este apartado se describe cada una de las etapas aplicadas al objetivo de este trabajo: analizar los frameworks de desarrollo para la implementación del editor gráfico.

2.1. Planeación

Esta fase se enfoca en definir qué frameworks se van a comparar, qué aspectos se van a medir (características) y cómo se van a medir dichos aspectos (métricas).

2.1.1. Frameworks a evaluar

Respecto a los frameworks, su selección se basó en los datos estadísticos del sitio Stack Overflow y Statistics&Data, descrito en la (sección 1.5.6). Es decir, los frameworks a evaluar son (ver Figura 7): React, jQuery, Angular.



FIGURA 7: FRAMEWORKS A EVALUAR.

Una acotación relevante en la selección de frameworks es que se consideró únicamente a frameworks dirigidos al FrontEnd, dejando de lado aquellos dirigidos al BackEnd. Esta marginación de frameworks está justificada por el hecho de que, durante el desarrollo del proyecto, los directores del proyecto decidieron mantener, en el BackEnd, la plataforma JAVA sobre la cual funciona un Motor de Transformación de modelos que fue construido en una investigación previa y que es parte esencial del proyecto macro de gestión y preparación de infraestructura y que contiene al proyecto de implementación del editor gráfico.

Con esta consideración, los frameworks de FrontEnd seleccionados deben soportar aspectos interactivos propios de la interfaz gráfica y ajustados a las necesidades de interacción del editor web. Por ejemplo, los frameworks deberán permitir implementar interacciones tales como arrastrar y soltar iconos sobre un diagrama, ingresar valores a los atributos de cada elemento en el diagrama y presentar el código script relacionado al diagrama visual.

2.1.2. Características y sub-características a evaluar

Respecto a las características, se eligió el modelo de calidad ISO/IEC 25010 (sección 1.5.7), porque las características propuestas por este modelo de calidad se adecuan a los objetivos de análisis y comparación de los frameworks web, debido a esto las dos características que se tomarán en cuenta para el desarrollo de este proyecto son: **adecuación funcional** y **eficiencia de desempeño** (ver Figura 4).

2.1.2.1. Eficiencia de desempeño

La **eficiencia de desempeño** permite evaluar la cantidad de recursos utilizados por el software evaluándolo bajo condiciones determinadas. A su vez, esta característica se divide en sub-características de las cuales se ha elegido a dos sub-características para realizar la evaluación respectiva de la característica:

- **Utilización de recursos**, permite medir la cantidad de recursos utilizados por el software bajo condiciones determinadas.
- **Comportamiento temporal**, evalúa los tiempos de respuesta y procesamiento de un software cuando lleva a cabo sus funciones.

2.1.2.2. Adecuación funcional

La característica **adecuación funcional** permite evaluar la capacidad del software para satisfacer las necesidades funcionales que el software debe ofrecer. Esta característica a

su vez se divide en sub-características, de las cuales se ha elegido a dos sub-características para realizar la evaluación respectiva de la característica:

- **Complejidad funcional**, evalúa si el conjunto de funcionalidades cubre con todas las tareas y objetivos que los usuarios especifican.
- **Pertinencia funcional**, evalúa si el software proporciona las funciones apropiadas para cumplir con las tareas y objetivos.

Estas características y sub-características son tomadas en cuenta, ya que son necesarias para evaluar la calidad del producto y permiten analizar qué framework es el adecuado para el desarrollo del editor gráfico. En efecto, se puede evaluar el rendimiento, tiempo de ejecución de las operaciones que realiza el software y las funcionalidades que este tiene.

2.1.3. Métricas

En esta sección se presentan las métricas, es decir, las formas de medir las subcaracterísticas de cada característica mencionada en la (sección 2.1.2).

2.1.3.1. Métricas para Eficiencia de desempeño

Esta característica mide el tiempo de ejecución de las operaciones Crear, actualizar y eliminar cuando son ejecutadas por el framework, además de la utilización de los recursos. La Tabla 4 describe la característica *eficiencia de desempeño con sus sub-características y la métrica para medir cada sub-característica*.

Tabla 4: Característica eficiencia de desempeño.

Característica	Sub característica	Métrica	Descripción de la métrica
Eficiencia de Desempeño.	Comportamiento temporal.	Tiempo de ejecución operación Crear.	Tiempo que tarda el framework en ejecutar la operación Crear.
		Tiempo de ejecución operación actualizar.	Tiempo que tarda el framework en ejecutar la operación actualizar.

		Tiempo de ejecución operación eliminar.	Tiempo que tarda el framework en ejecutar la operación eliminar.
	Utilización de recursos.	Utilización de la CPU.	Porcentaje de uso del procesador.
		Utilización de la memoria RAM.	Cantidad de memoria utilizada.

2.1.3.2. Métricas para Adecuación funcional

Esta característica mide la capacidad de los frameworks para proveer los servicios adecuados en el desarrollo del editor gráfico, a fin de, conseguir que estos cumplan con los requisitos funcionales en la implementación del editor, por ejemplo, poder arrastra y soltar iconos, además, de la compatibilidad de la biblioteca MxGraph en cada framework. La Tabla 5 resume la característica y su respectiva métrica que se utilizaran para obtener los resultados respecto a la adecuación funcional.

Para obtener los resultados de la adecuación funcional se utilizará la formula $X = \frac{A}{B}$ donde, A es el número de funcionalidades que están presentes en el framework y B es el número de funcionalidades totales tomadas en cuenta para el framework, además, $0 \leq X \leq 1$, entre mas cerca este el valor de 1 más completa será la funcionalidad del framework.

Tabla 5: Característica adecuación funcional.

Característica	Sub característica	Métrica	Descripción de la métrica
Adecuación funcional.	Pertinencia funcional.	Permite realizar tareas de manera eficiente.	Permite arrastrar, soltar los iconos en el entorno de diseño.
	Compleitud funcional.	Permite la utilización de la biblioteca MxGraph.	Permite la integración de la biblioteca gráfica con el framework.

2.1.4. Consideración en la medición de características

Respecto a la medición de las características, serán medidas en igualdad de condiciones, los frameworks serán ejecutados en entornos virtuales de iguales características. En efecto, la máquina virtual en la que se ejecutará cada framework tendrá el mismo sistema operativo (Windows 10) y mismas características de hardware (RAM y procesador). Esto permitirá obtener resultados numéricos bajo las mismas condiciones para luego comparar los resultados y elegir el framework con los mejores resultados, además se implementará una aplicación web para cada framework con el objetivo de, obtener los resultados respecto al tiempo respectivo a las operaciones de crear, actualizar y eliminar.

2.2. Elaboración de benchmark

Con el fin de medir la característica *eficiencia de desempeño*, se decidió diseñar una **aplicación web** que permita realizar las **operaciones de crear, actualizar y eliminar** un objeto. La idea es implementar esta aplicación en cada uno de los frameworks. Con esto garantizamos que los tres frameworks bajo estudio operen las mismas operaciones y los resultados de desempeño de cada framework sean comparables. En esta fase se describe el diseño, diagrama de navegación y esquema de componentes de la aplicación web.

La aplicación web constará de tres interfaces de usuario cuya navegación entre ellas se presenta en el diagrama de la Figura 8. La interfaz principal es “Listado Clientes” y permite visualizar el listado general de los clientes creados. La interfaz “crear cliente” permite crear un cliente y “Editar cliente”, permite actualizar un cliente. Por último, la operación eliminar cliente se visualizará en la interfaz principal como una acción en cada registro de un cliente.



Figura 8: Diagrama de navegación.

La aplicación web presentará una interfaz principal en la cual se visualizará la tabla general de los datos de clientes, las acciones para **actualizar** o **eliminar** un cliente y el botón **crear**.

El botón **crear** permitirá visualizar un formulario para la creación de clientes (ver Figura 9). Este formulario permitirá ingresar atributos para **crear** un cliente: nombre y cédula. El usuario deberá digitar estos datos en los campos correspondientes. Estos atributos estarán identificados en la interfaz con una etiqueta que hace referencia al nombre de atributo, por ejemplo, "Nombre" será la etiqueta que la interfaz de usuario mostrará para el atributo nombre. Esto evitará confusiones e inconvenientes al usuario al momento de llenar el formulario. Los datos serán guardados en la base de datos al dar clic sobre el botón guardar.

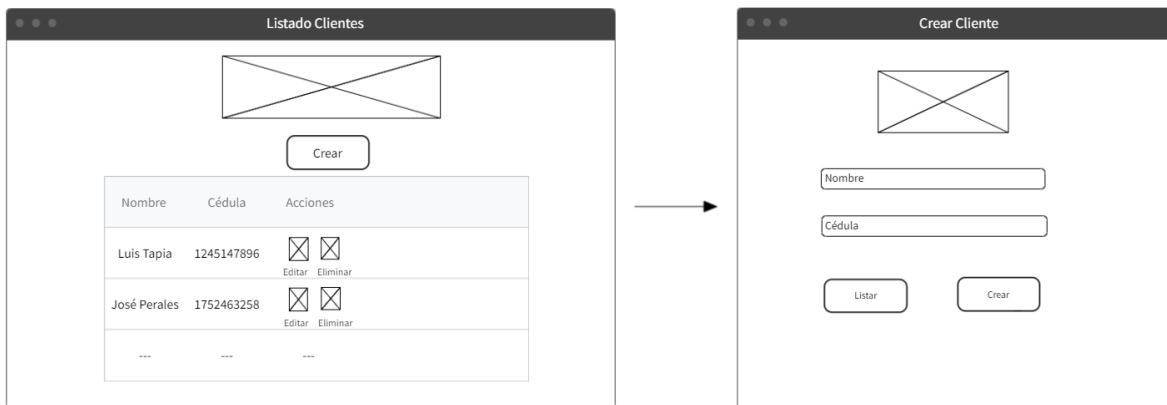


Figura 9: Boceto crear cliente.

La operación **editar** se visualizará en la columna de acciones de la interfaz principal (ver Figura 10). Esta acción permitirá recuperar el registro de un cliente de la base de datos. Para visualizar esta acción, el cliente debe haber sido previamente creado. La interfaz editar cliente presentará los datos del cliente en los campos respectivos, de esta forma se podrá actualizar cualquier atributo del cliente de forma rápida y sencilla.

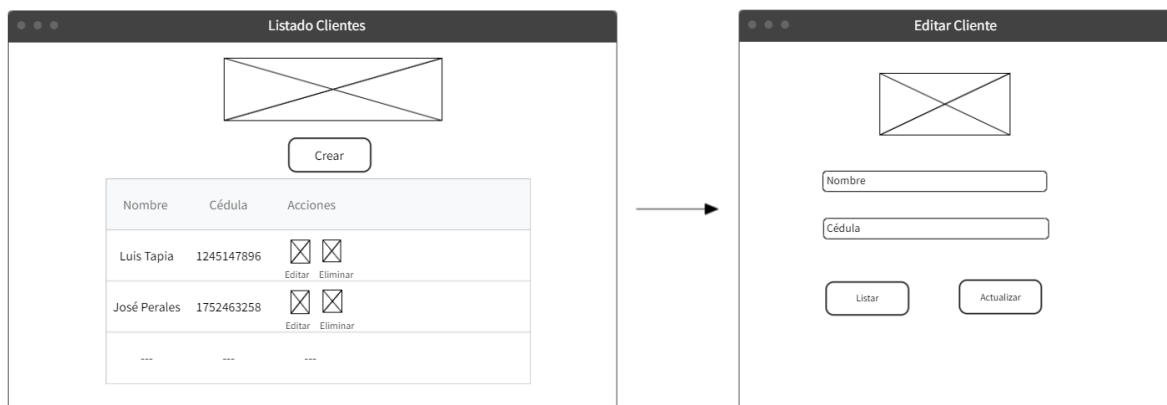


Figura 10: Boceto actualizar cliente.

La operación **eliminar** será visible en la interfaz principal en la columna de acciones (ver Figura 11), esta acción permitirá eliminar el registro de un cliente de la base de datos. Cuando el registro de un cliente sea eliminado, se visualizará el mensaje “Cliente borrado con éxito”. Además, la opción “eliminar” solo será visible en la interfaz principal cuando exista un cliente que haya sido creado previamente.

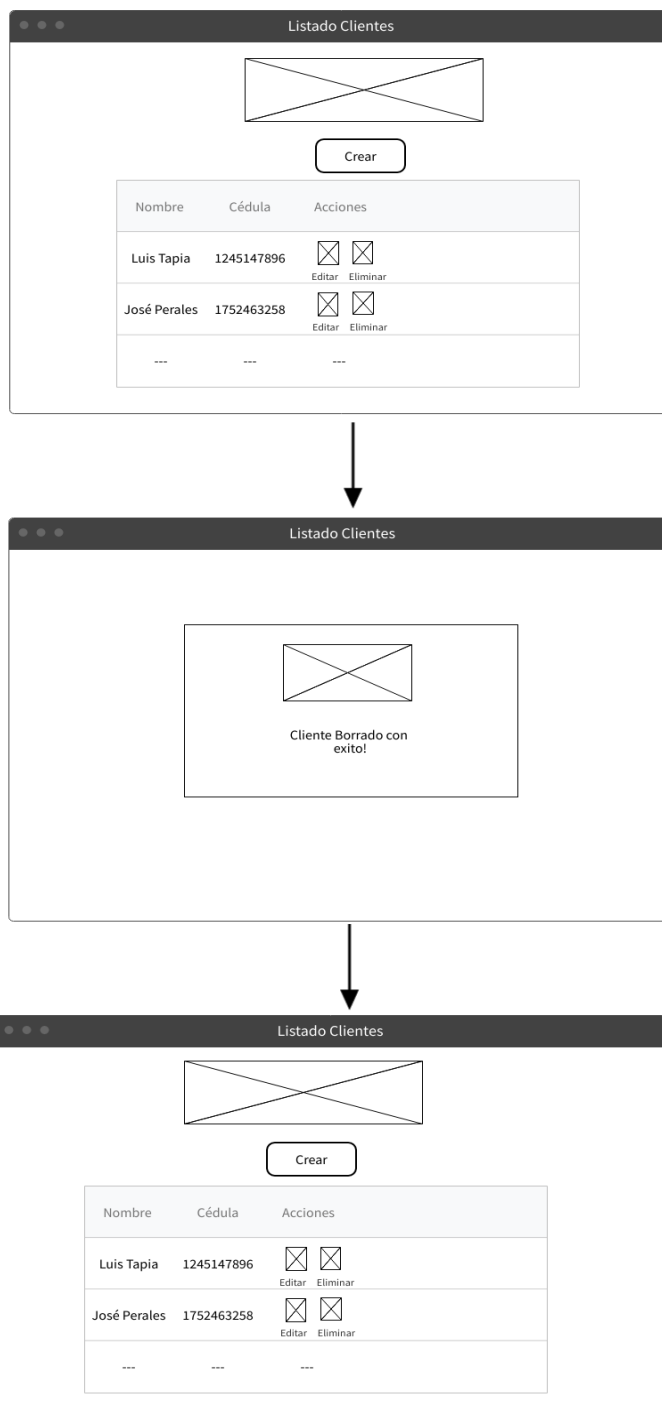


Figura 11: Boceto eliminar cliente.

En la Figura 12 se puede observar el esquema de componentes de la aplicación web que se implementará. Este esquema de componentes utilizará el BackEnd de Firebase proporcionado por Google, este BackEnd a su vez presenta varios servicios que se pueden integrar a la aplicación web. Para esta aplicación web se utilizará Cloud Firestore, este servicio cuenta con una base de datos no relacional que servirá para almacenar los registros de los clientes.

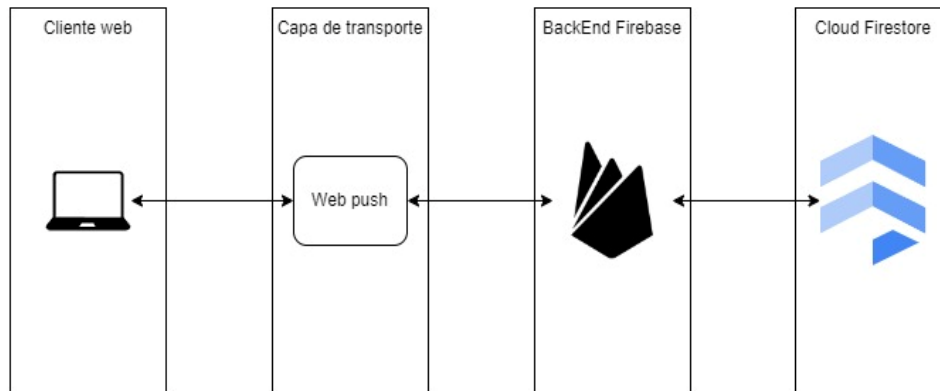


Figura 12: Esquema de componentes de la aplicación web.

Para implementar la aplicación web se utilizó el editor de código Visual Studio Code (ver Figura 13), ya que permite agregar extensiones para personalizar y agregar funcionalidades a la hora de programar en cualquier lenguaje, siendo de gran ayuda para implementar las aplicaciones web en los frameworks elegidos.

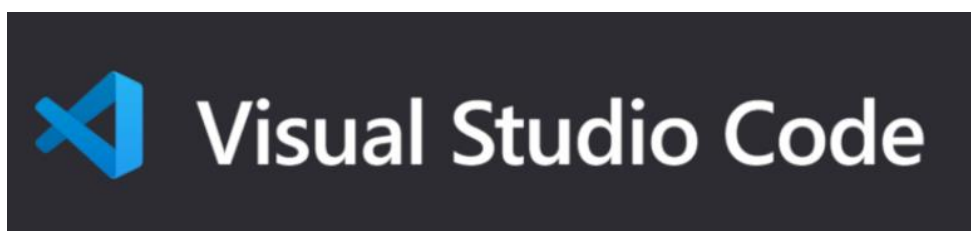


Figura 13: Editor de código Visual Studio Code.

Respecto a la base de datos, se utilizará la plataforma Firebase (ver Figura 14), esta plataforma permite el desarrollo y creación de aplicaciones para móviles como web, además ofrece diversas opciones para implementar el BackEnd; una de estas opciones es la base de datos Cloud Firestore utilizada para crear la base de datos de la aplicación web

a desarrollar, esta base de datos permite almacenar, consultar y eliminar los registros fácilmente.



Figura 14: Plataforma Firebase.

La aplicación web descrita será implementada en cada framework, a fin de, obtener los datos de rendimiento de cada framework y con esto poder comparar los frameworks y determinar cuál de ellos es el más adecuado para desarrollo del FrontEnd del editor visual. El desarrollo de la aplicación web para los frameworks React, Angular y Jquery se describen en los anexos I , II y III, respectivamente.

2.3. Medición de parámetros y generación de resultados

En esta fase se realizan las mediciones de las características propuestas en la fase de planeación y que fueron tomadas de la ISO/IEC 25010. Se aplican las métricas indicadas para medir cada característica.

La Tabla 6 muestra los resultados obtenidos para las sub-características *comportamiento temporal* y *utilización de recursos* respecto a los Frameworks estudiados. Para la sub-característica *comportamiento temporal*, los resultados hacen referencia al tiempo (en milisegundos) de ejecución de las operaciones crear, actualizar y eliminar. Para la sub-caracaterística *utilización de recursos*, los resultados hacen referencia al porcentaje de uso de la CPU y la cantidad de megabytes de uso de la memoria RAM.

Tabla 6: Resultados de medición: característica eficiencia de desempeño.

Sub característica	Métrica	React	Angular	jQuery
Comportamiento temporal.	Crear	217.50 ms	0.4147 ms	0.7059 ms
	Actualizar	253.36 ms	0.8450 ms	0.9824 ms
	Eliminar	229.47 ms	0.2160ms	0.2847 ms
Utilización de recursos.	Utilización de la CPU	0.1%	0.1%	0.1%

	Utilización de la RAM	1085 MB	936 MB	919 MB
--	-----------------------	---------	--------	--------

La Tabla 7 contiene los resultados de la característica *adecuación funcional*, la misma que consta de las sub características *pertinencia funcional* y *completitud funcional*. Los resultados afirmativos se marcan con una “X”, es decir, cuando el framework tiene la sub-característica correspondiente la casilla será marcada, caso contrario si no tiene la sub-característica la casilla quedara vacía. Por ejemplo, en el caso de la sub-carácterística *pertinencia funcional* las tres casillas están marcadas con la “X”, entonces los tres frameworks React, Angular y jQuery adoptan la sub-característica para poder arrastrar y soltar iconos cuando se está creando diagramas.

Tabla 7: Medición característica adecuación funcional.

Sub característica	Métrica	React	Angular	jQuery
Pertinencia funcional.	Arrastrar y soltar iconos.	X	X	X
Completitud funcional.	Compatibilidad con MxGraph.	X	X	X
Resultados		1	1	1

2.4. Análisis de resultados

En esta fase se analiza los datos obtenidos en la fase anterior.

2.4.1. Comportamiento temporal

En la sub-característica *comportamiento temporal* se midió la cantidad de tiempo (en milisegundos) que toma al framework en realizar las operaciones Create, Update y Delete. Para poder analizar mejor los resultados de los tiempos se promediará el tiempo de las tres operaciones. La Tabla 8 muestra el promedio de tiempo (en milisegundos) para cada framework.

Tabla 8: Promedio de tiempo (milisegundos) del comportamiento temporal por cada framework.

Framework	Tiempo promedio (milisegundos)

React	233.44
Angular	0.4919
Jquery	0.6576

Los resultados en cuanto al tiempo de ejecución de las operaciones están a favor del framework Angular como lo muestra la Figura 15 donde Angular ejecuta las operaciones en menor tiempo en comparación al framework jQuery que está en segundo lugar. React toma más tiempo en ejecutar las operaciones por lo que está en tercer lugar.

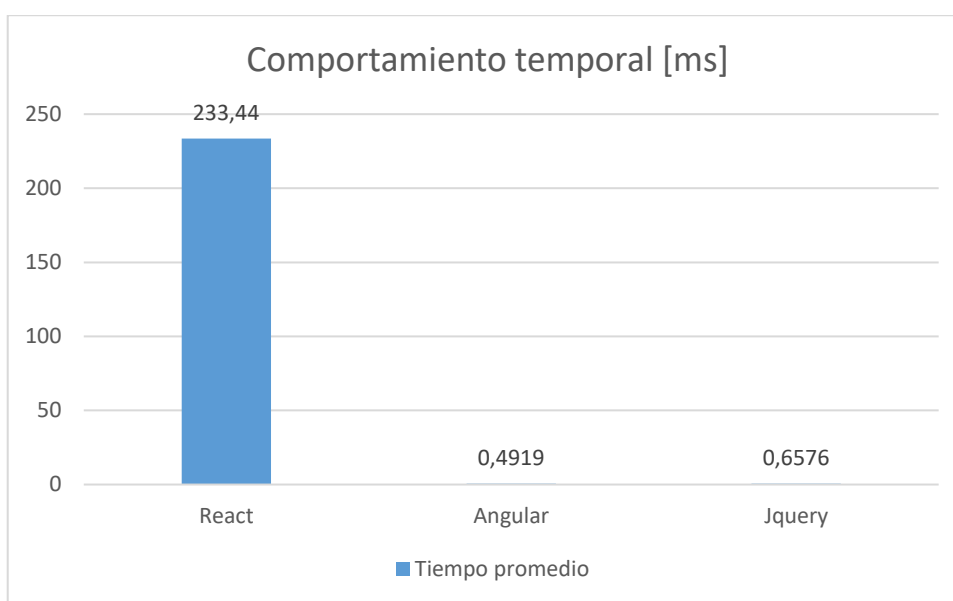


Figura 15: Gráfico de promedio de tiempo (milisegundos) del comportamiento temporal.

2.4.2. Utilización de recursos

Respecto a la *utilización de recursos*, la Figura 16 muestra que, en cuanto a la utilización de la CPU, tanto React, Angular y jQuery tienen un mismo comportamiento, ya que el porcentaje de uso de la CPU es de 0.1. En cuanto a la utilización de la RAM se puede observar que jQuery es el framework que menos cantidad de RAM utiliza con 919 MB.

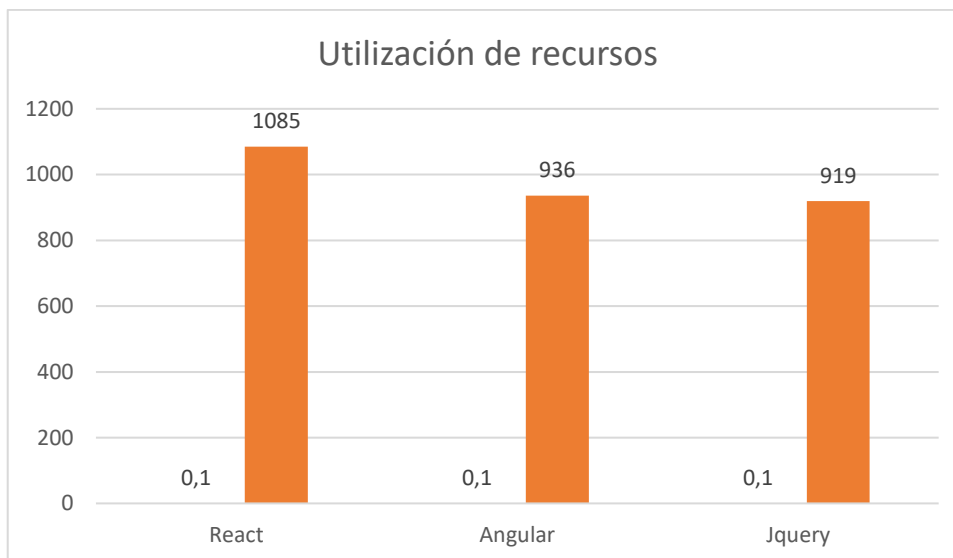


Figura 16: Gráfico de la utilización de recursos.

2.4.3. Pertinencia funcional

Respecto a la *pertinencia funcional*, se puede observar en la Tabla 9 que las casillas de los tres frameworks están marcadas con una “X”, es decir, que la interacción para arrastrar y soltar iconos en un entorno gráfico está disponible. Esta funcionalidad es importante, ya que permite la creación de diagramas de forma eficiente y rápida.

Tabla 9: Análisis característica pertinencia funcional.

Pertinencia funcional	
Framework	Arrastrar y soltar iconos
React	X
Angular	X
jQuery	X

2.4.4. Completitud funcional

Respecto a la *compatibilidad* con MxGraph, se observa en la Tabla 10 que las casillas de los tres frameworks están marcadas con una “X”, es decir, estos frameworks son compatibles con MxGraph, gracias a esto, la integración de esta biblioteca hace posible que cada framework pueda utilizar la biblioteca para implementar aplicaciones de diseño gráfico de diagramas.

Tabla 10: Análisis característica completitud funcional.

Completitud funcional.	
Framework	Compatibilidad con MxGraph.
React	X
Angular	X
jQuery	X

3. CONCLUSIONES Y RECOMENDACIONES

3.1. Conclusiones

En esta sección se describen las conclusiones en función de los objetivos planteados en este trabajo.

- Respecto al Objetivo 1: “Analizar las herramientas de desarrollo web existentes en el mercado”, se realizó un estudio previo con el fin de conocer los frameworks más utilizados y populares por los desarrolladores para implementar proyectos web. Para esto, se consultó dos páginas web que se dedican a obtener estadísticas sobre temas de programación. Para efectos del proyecto, se consideró los datos de la página Stack Overflow, donde los frameworks de FrontEnd más utilizados para 2022 son React, jQuery y Angular. Estos frameworks son los que se utilizaron para realizar la comparativa propuesta para este proyecto.
- Respecto al Objetivo 2: “Evaluar las herramientas de acuerdo con las necesidades de los usuarios”, se realizó un análisis de las funcionalidades interactivas, tales como arrastrar y soltar los iconos gráficos. Para esto, el estudio de la integración de los frameworks con la biblioteca MxGraph fue de vital importancia. Se encontró que MxGraph es compatible con los frameworks evaluados.
- Respecto al Objetivo 3: “Tomar decisiones en función de las herramientas más adecuadas para el desarrollo del editor web”, se utilizó las características, subcaracterísticas y métricas propuestas por la ISO 25010 para obtener los datos de las métricas, ya que, permitieron evaluar las características de cada framework. Como resultado, se obtuvo que el framework Angular es el más adecuado para el desarrollo del editor gráfico.

- Todas las actividades realizadas en los tres objetivos descritos anteriormente apoyan a la consecución del objetivo principal de este trabajo: “Realizar un Benchmarking de herramientas de desarrollo web para la implementación del editor gráfico para la gestión y preparación de infraestructura”. El análisis realizado de los tres frameworks y los datos obtenidos en cada característica medida, ponen en gran ventaja al framework Angular para el desarrollo del editor gráfico. Angular presenta características idóneas para el desarrollo de aplicaciones web tales como mejor tiempo de ejecución en operaciones de creación, eliminación y actualización. Además, Angular muestra ser compatible con la biblioteca MxGraph la misma que permite una gran interacción con los iconos: el usuario puede arrastrar y soltar los elementos gráficos en pantalla de forma sencilla. En consecuencia, Angular se convierte en el framework más adecuado para la implementación del editor gráfico.

3.2. Recomendaciones

La metodología benchmarking utilizada para el desarrollo del proyecto es adecuada para realizar la comparativa de los frameworks. Es recomendable que, en la tercera etapa del benchmarking (relacionada con la medición de parámetros y generación de resultados) se promedie varias medidas de tiempo. Esto debido a que existe una ligera variabilidad en cada medición, en consecuencia el valor promedio asegura que la medida de tiempo tenga mayor fiabilidad.

Para el desarrollo de la aplicación web, es recomendable elaborar una planificación para la implementación, realizando un diseño que considere aspectos de arquitectura e integración de una base de datos. De esta forma, la planificación reduce el tiempo de desarrollo de la aplicación web.

Es importante utilizar métricas adecuadas para la evaluación de los frameworks. La correcta elección de las métricas garantizará la comparación correcta de rendimiento, tiempos de respuesta o funcionalidades disponibles. En este proyecto las métricas elegidas fueron de gran ayuda para saber qué framework puede ser empleado para desarrollar el editor gráfico.

4. REFERENCIAS BIBLIOGRÁFICAS

- [1] "Aprovisionamiento en la nube – Techinfo." <https://techinfo.wiki/aprovisionamiento-en-la-nube/> (accessed Jun. 05, 2022).
- [2] "¿Qué es aprovisionamiento?" <https://www.redhat.com/es/topics/automation/what-is-provisioning> (accessed Jun. 05, 2022).
- [3] "Servicios de almacenamiento en la nube: comparativa - IONOS." <https://www.ionos.es/digitalguide/servidores/herramientas/servicios-de-almacenamiento-en-la-nube-comparativa/> (accessed Jun. 05, 2022).
- [4] "Infraestructura como código | Microsoft Docs." <https://docs.microsoft.com/es-es/dotnet/architecture/cloud-native/infrastructure-as-code> (accessed Jun. 05, 2022).
- [5] "¿Qué es un framework y para qué sirve? [2022]." <https://www.crehana.com/blog/desarrollo-web/que-es-un-framework/> (accessed Jun. 05, 2022).
- [6] "Backend y Frontend, ¿Qué es y cómo funcionan en la programación?" <https://www.servnet.mx/blog/backend-y-frontend-partes-fundamentales-de-la-programaci%C3%B3n-de-una-aplicaci%C3%B3n-web> (accessed Jun. 05, 2022).
- [7] "Benchmarking: ¿qué es y cómo hacer este análisis en tu empresa?" <https://rockcontent.com/es/blog/que-es-benchmarking/> (accessed Jun. 05, 2022).
- [8] Nicolás Seguro, "¿Qué es el desarrollo Web y por qué es importante?," Sep. 16, 2021. <https://latam.coderhouse.com/blog/que-es-el-desarrollo-web> (accessed Jun. 05, 2022).
- [9] "¿Qué es el desarrollo web? [Todo lo que necesitas saber]." <https://tekla.io/blog/que-es-desarrollo-web/> (accessed Jun. 05, 2022).
- [10] "Informática Básica: ¿Qué son las aplicaciones web?" <https://edu.gcfglobal.org/es/informatica-basica/que-son-las-aplicaciones-web/1/> (accessed Jul. 03, 2022).
- [11] J. Olivares García, A. Monroy, and W. Rainier Marquez, "Computación para Docentes." [Online]. Available: www.mhe.es/cf/informatica
- [12] YeePLY, "5 Tipos de desarrollo de aplicaciones web más relevantes," Jan. 2018. <https://www.yeeply.com/blog/6-tipos-desarrollo-de-aplicaciones-web/> (accessed Jun. 05, 2022).
- [13] L. Joyanes Aguilar, "Computación en la Nube e innovaciones tecnológicas El nuevo paradigma de la Sociedad del Conocimiento." [Online]. Available: www.forbes.com/forbes/2009/1228/technology-virtualization-vmware-wyse.html
- [14] Aprendiendo a Programar, "Aprovisionamiento en la computación en la nube: tipos, beneficios, herramientas, desafíos," Dec. 28, 2021. <https://aprendiendoaprogramar.es/blog/aprovisionamiento-en-la-computacion-en-la-nube-tipos-beneficios-herramientas->

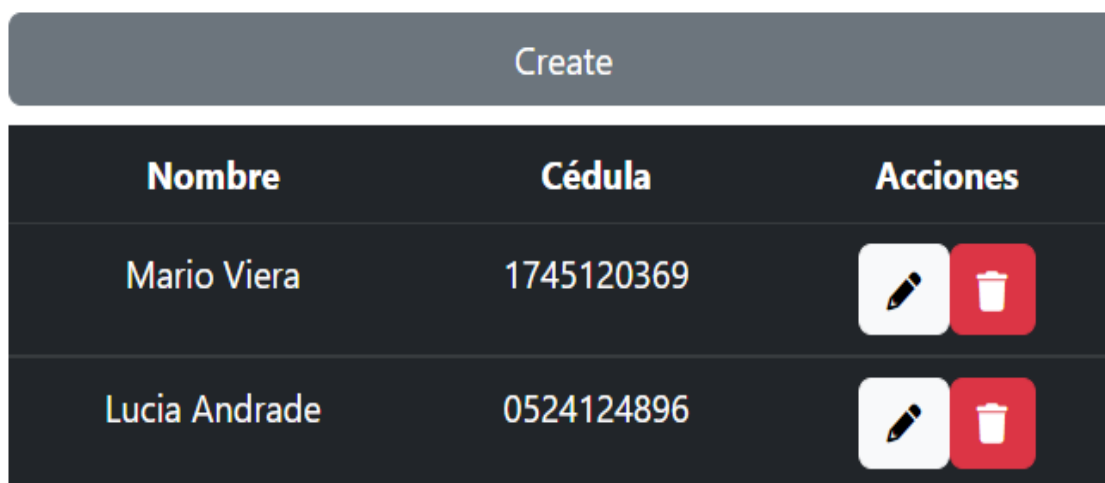
- desafios/#:~:text=El%20aprovisionamiento%20en%20la%20nube,de%20una%20organizaci%C3%B3n%20de%20TI. (accessed Jun. 05, 2022).
- [15] “Tecnologías para el desarrollo web más actuales | proun Madrid - Asturias.” <https://www.proun.es/blog/tecnologias-web-actuales/> (accessed Jun. 05, 2022).
- [16] “Encuesta para desarrolladores de Stack Overflow 2022.” <https://survey.stackoverflow.co/2022/> (accessed Jul. 17, 2022).
- [17] “7 Tecnologías para Desarrollo Web | KeepCoding Tech School.” <https://keepcoding.io/blog/7-tecnologias-para-desarrollo-web/> (accessed Jun. 05, 2022).
- [18] J. J. Gutiérrez, “¿Qué es un framework web?”
- [19] “Frameworks backend más populares - 2012/2022 - Estadísticas y datos.” <https://statisticsanddata.org/data/most-popular-backend-frameworks-2012-2022/> (accessed Jul. 17, 2022).
- [20] “ISO 25010.” <https://iso25000.com/index.php/normas-iso-25000/iso-25010> (accessed Aug. 13, 2022).
- [21] “Qué es un benchmark.” <https://www.tableau.com/es-es/learn/articles/what-is-a-benchmark> (accessed Jun. 05, 2022).
- [22] “Benchmarking: ¿qué es y cómo hacer este análisis en tu empresa?” <https://rockcontent.com/es/blog/que-es-benchmarking/> (accessed Aug. 21, 2022).
- [23] “Luchar por ser el mejor de los mejores.”
- [24] M. Gurutze, I. Clemente, C. Ochoa, L. Eva, and V. Balmaseda, “¿ES EL BENCHMARKING UNA HERRAMIENTA DE APRENDIZAJE ORGANIZACIONAL?”
- [25] W. Rodrigo, A. Moyano, D. Fernando, and I. López, “ESCUELA POLITÉCNICA NACIONAL FACULTAD DE INGENIERÍA DE SISTEMAS BENCHMARK PARA EL USO DE TECNOLOGÍAS RELACIONADAS A SERVIDOR DE APLICACIONES GLASSFISH Y JBOSS PROYECTO PREVIO A LA OBTECIÓN DEL TÍTULO DE INGENIERO ESPECIALISTA EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN.”
- [26] “Advanced mxGraph (1) mxGraph Tutorial-Guía de inicio de desarrollo - programador clic.” <https://programmerclick.com/article/9071483079/> (accessed Aug. 28, 2022).
- [27] “TFG_2017_Varela Bellot_Daniel”.
- [28] “Draw.io.” [Online]. Available: <https://www.youtube.com/watch?v=n6s4aP5LU04>
- [29] “ONLYOFFICE integra Draw.io: bocetos y diagramas en el editor.” <https://www.onlyoffice.com/blog/es/2022/03/onlyoffice-integra-draw-io/> (accessed Sep. 01, 2022).

5. ANEXOS

ANEXO I. Aplicación Web desarrollada en React

Se elaboró un diseño sencillo para la creación de clientes (Figura 17), en donde los atributos para la creación de los clientes son el nombre y la cédula.

La vista principal de la aplicación consta de la tabla que muestra los datos de los clientes y las respectivas acciones para editar y borrar un cliente (columna "Acciones").



The image shows a user interface for a web application. At the top, there is a grey button labeled "Create". Below it is a table with a dark background and white text. The table has three columns: "Nombre", "Cédula", and "Acciones". There are two rows of data. The first row shows "Mario Viera" with ID "1745120369". The second row shows "Lucia Andrade" with ID "0524124896". Each row in the "Acciones" column contains two icons: a pencil icon for editing and a trash can icon for deleting.





Nombre	Cédula	Acciones
Mario Viera	1745120369	 
Lucia Andrade	0524124896	 

Figura 17: Ejemplo FronEnd React.

La opción "Create", en la parte superior de la pantalla, despliega el formulario para registrar un cliente. El formulario permite ingresar el nombre y la cédula del cliente, como se muestra en la Figura 18.

Crear Cliente

Nombre

Cedula

Guardar

Figura 18: Formulario para crear un cliente.

La opción de Editar un cliente (icono de lápiz, en la Figura 17) permite actualizar los datos en el caso de que estos requieran ser cambiados. Esta opción editar, devuelve los registros desde la base de datos en los respectivos campos, de esta forma permite actualizar estos datos de forma sencilla. Ver Figura 19.

Editar Cliente

Nombre

Mario Viera

Cedula

1745120369

Actualizar


Figura 19: Editar cliente.

Por último, la opción de borrar un cliente (icono de bote de basura, Figura 17) permite eliminar el registro de un cliente que está almacenado en la base de datos,

ANEXO II. Aplicación Web desarrollada en Angular

Se implementó una aplicación web para crear un listado de clientes. Este listado consta de los siguientes atributos: nombre y cédula.

La interfaz principal muestra la información detallada de cada cliente, esta información es almacenada en la base de datos Firestore, además existe una columna con las acciones que permite editar y eliminar los registros de los clientes Figura 20.









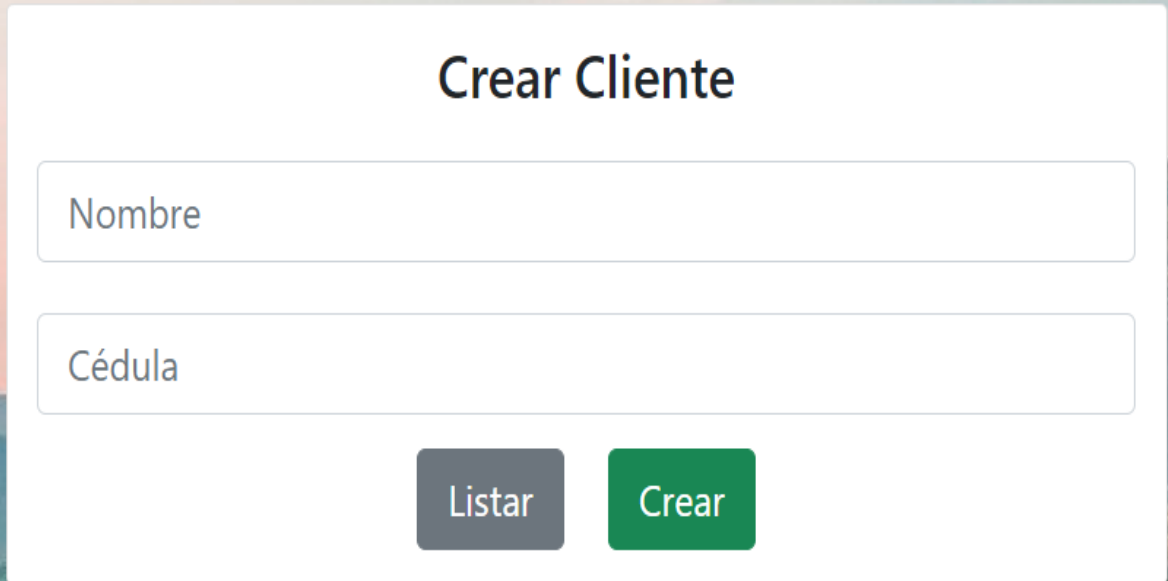
Nombre	Cédula	Acciones
Elvis Montaluisa	0503408080	 
Andrea Marin	1754896458	 
Luisa Mera	1245178954	 

Figura 20: Ejemplo de FrondEnd framework Angular.

La interfaz crear un cliente, permite agregar los datos de los clientes: nombre y cédula. Esta interfaz consta de dos botones, el botón para crear que registra los datos en Firestore y el botón listar (ver Figura 21), este botón permite visualizar el listado de todos los clientes ver Figura 20.



Crear Cliente

Nombre

Cédula

Listar Crear

Figura 21: Interfaz para agregar un cliente.

La opción editar cliente (icono de persona con un lápiz, Figura 20) presenta los datos que pertenecen a ese registro, de esta forma, pueden ser actualizados ver Figura 22.



Editar Cliente

Elvis Montaluisa

0503408080

Listar Actualizar

Figura 22: Interfaz editar cliente.

Por último, la opción para eliminar los registros de un cliente (icono bote de basura, Figura 20) está presente en la columna acciones y permite eliminar el registro de la base de datos.

ANEXO III. Aplicación Web desarrollada en JQuery

Se implementó una aplicación web para la crear un listado de clientes, este listado consta de los siguientes atributos mostrados en la interfaz principal: nombre y cédula.

La interfaz principal muestra el listado detallado de los clientes almacenada en la base de datos Firestore ver Figura 23.



NOMBRE	CÉDULA	ACCIONES
Elvis	0503408080	 
Luis	1785412459	 

Figura 23: Ejemplo FrondEnd jQuery.

La interfaz crear un cliente (icono de cuadrado con un símbolo más, Figura 23), permite agregar los datos de los clientes: nombre y cédula. Esta interfaz consta de dos botones, el botón para guardar que registra los datos en Firestore y el botón listar, este botón permite visualizar los datos del cliente ver Figura 24.

Crear Cliente

Nombre

Cédula

Listar Crear

Figura 24: Interfaz crear cliente.

La opción editar cliente (icono de lápiz, Figura 23) presenta los datos que pertenecen a ese registro, de esta forma, pueden ser actualizados ver Figura 25.

Editar Cliente

Nombre

Cédula

Listar Actualizar

Figura 25: Interfaz editar cliente.

Por último, la opción para eliminar (icono bote de basura, Figura 23) los registros de un cliente están presente en la columna acciones y permite borrar el registro de la base de datos.