



**UNIVERSIDAD TÉCNICA DE BABAHOYO**

**FACULTAD DE ADMINISTRACIÓN, FINANZAS E INFORMÁTICA.**

**PROCESO DE TITULACIÓN**

**DICIEMBRE 2021 – ABRIL 2022**

**EXAMEN COMPLEXIVO DE GRADO O DE FIN DE CARRERA**

**PRUEBA PRÁCTICA**

**PREVIO A LA OBTENCIÓN DEL TÍTULO DE:**

**INGENIERO EN SISTEMAS DE INFORMACIÓN**

**TEMA:**

**ANÁLISIS COMPARATIVO DE LOS FRAMEWORKS MÓVILES ENTRE VUE Y**

**ANGULAR AÑO 2022**

**ESTUDIANTE:**

**TITO JOHNNIER BARBOTÓ ÁLVAREZ**

**TUTOR:**

**ING. NELLY KARINA ESPARZA CRUZ**

**AÑO 2022**

## **Planteamiento del Problema**

El significativo aumento en los avances tecnológicos y la indiscutible dependencia de la sociedad para estar interconectados ha permitido que cada vez el concepto de movilidad se vaya anclando en la vida de las personas, es por ello que el desarrollo de aplicaciones con entorno web y móvil se han vuelto hoy en día una de las actividades con mayor demanda, y con ello la introducción al mercado de herramientas de desarrollo es cada vez mayor haciendo que la programación de aplicaciones sea más óptima, legible y reutilizable permitiendo a los desarrolladores optimizar el tiempo de entrega.

## **Justificación**

Con la gran cantidad de frameworks para el desarrollo de aplicaciones basados en la facilidad de movilidad, los desarrolladores tienen un gran dilema al momento de seleccionar entre uno y otro framework, es por ello que este proyecto de investigación nos permitirá conocer de mejor manera los frameworks VUE.JS y ANGULAR por medio del análisis comparativo entre ellos tomando en cuenta sus principales características, ayudando de esta manera a que el desarrollador de aplicaciones web y móviles tome la mejor opción al momento de seleccionar el framework apropiado de acuerdo a la magnitud de su proyecto.

## **Objetivos**

### **Objetivo General**

- Realizar un análisis comparativo de los frameworks móviles entre VUE y ANGULAR.

### **Objetivos Específicos**

- Identificar las características principales de cada uno de los frameworks que forman parte de este estudio
- Identificar las bondades entre los frameworks VUE.JS y ANGULAR
- Realizar una tabla comparativa entre los frameworks VUE.JS y ANGULAR

## **Línea de Investigación**

Este proyecto de investigación está basado en la sublínea de investigación de la carrera de sistemas de información, la cual habla sobre las redes y tecnologías inteligentes de software y hardware, y en dirección a la línea de investigación de los sistemas de información, comunicación emprendimiento e innovación.

## Marco Conceptual

Al hablar sobre el desarrollo de aplicaciones web y móviles, se hace necesario mencionar los frameworks ya que los estándares de las aplicaciones web van aumentando a la par que incrementa la complejidad tecnológica.

Los frameworks contiene bibliotecas con código pre-escrito y características adicionales, que permiten a los desarrolladores crear aplicaciones web funcionales e interactivas, es importante mencionar que la mayoría de estos framework se encuentran desarrollados bajo el lenguaje C y JAVA haciéndolos adaptables y reutilizables.

Stack Overflow es un sitio web muy popular dentro del mundo de los desarrolladores donde se presentan encuestas con análisis de tecnologías populares, frameworks web, bibliotecas, herramientas, bases de datos, etc. (Stackoverflow, 2021)

**Tabla 1**

*Breve descripción de los frameworks más utilizados.*

FRAMEWORK	DESCRIPCIÓN
<b>DJANGO</b>	Es un framework que funciona alrededor de la calidad de las aplicaciones web. Django se usa generalmente para el rápido desarrollo de API y aplicaciones web. Sin embargo, funciona mucho mejor para aplicaciones web con una sólida administración de back-end.
<b>SPRING</b>	Es un framework de Modelo-Vista-Controlador para aplicaciones basadas en Java. Spring es realmente útil cuando administra transacciones confiables, almacenamiento en caché, seguridad o registro.
<b>LARAVEL</b>	Es el mejor framework de desarrollo web de PHP que sigue el patrón Modelo-vista-controlador y un marco de fondo completo. Los desarrolladores generalmente trabajan con Laravel para crear funcionalidades sólidas, como autenticación, integración de API, ejecución de vulnerabilidades habituales, habilitación de pruebas web de automatización y almacenamiento en caché.
<b>REACT</b>	React no es un framework, pero es una biblioteca frontend de JavaScript creada por desarrolladores de Facebook. Aun así, muchos desarrolladores

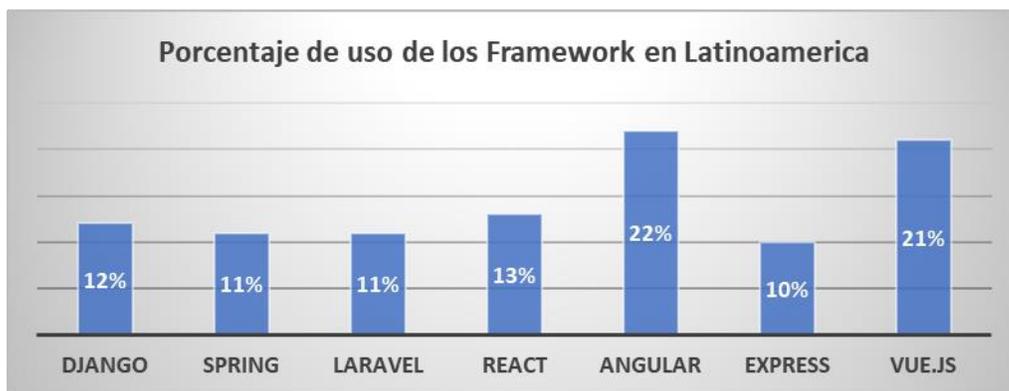
	consideran React un framework, porque tiene una arquitectura con componentes excelentes.
<b>ANGULAR</b>	Este framework de frontend fue desarrollado por Google, que actualmente posee una alta demanda. Se ejecuta en JavaScript, que es realmente útil para crear sitios web de una sola página. Angular usa HTML para construir interfaces de usuario y rompe la funcionalidad en los módulos. Además, este framework es compatible con la última versión de Chrome, Firefox, Edge, iOS y Android.
<b>EXPRESS</b>	Este framework es el número uno de Node.js, el cual adopta un enfoque minimalista y rápido para el desarrollo web. Empresas como Accenture, IBM y Uber utilizan este tipo de framework debido a su flexibilidad, soporte de aplicaciones completas y API REST.
<b>VUE.JS</b>	Vue.js es el más reciente framework para el desarrollo web, siendo muy popular entre los desarrolladores debido a sus increíbles características. Es un framework progresivo, esto significa que tu actual proyecto se puede adaptar a Vue sin ningún problema. De hecho, Vue tiene un generador de proyectos CLI con un asistente intuitivo que hace posible esta adaptación. Esto sucede debido a su funcionalidad básica simple que se puede mejorar con bibliotecas, complementos y API.

Nota: Esta tabla muestra los diferentes Frameworks y su descripción

Una vez que se han detallado los principales frameworks que interactúan a nivel web, móvil, entre otros. En el siguiente cuadro se muestra estadísticamente el porcentaje de uso en el continente latinoamericano.

### Gráfico 1

*Estadística de uso de los frameworks*



Nota: Este gráfico muestra los Frameworks más usados por medio de porcentajes.

En este proyecto de investigación realizaremos un análisis comparativo entre los frameworks Vue.js y Angular, para lo cual hablaremos de cada uno de ellos para luego realizar una tabla comparativa, la cual nos mostrará características propias y comunes de ambos frameworks.

En primer lugar, hablaremos de vue.js que es un framework de JavaScript de código abierto para la construcción de interfaces de usuario, cuenta con una arquitectura de adaptación gradual que se centra en la representación declarativa y la composición de componentes. La biblioteca central se enfoca sólo en la capa de vista. Las características avanzadas necesarias para aplicaciones complejas como el enrutamiento, la gestión de estados y las herramientas de construcción se ofrecen a través de librerías y paquetes de apoyo mantenidos oficialmente con Nuxt.js como una de las soluciones más populares.

Vue.js permite extender el HTML con atributos HTML llamados directivas. Las directivas ofrecen funcionalidad a las aplicaciones HTML, y vienen como directivas incorporadas o definidas por el usuario.

Una de las características principales de Vue es que sus componentes extienden los elementos básicos de HTML para encapsular el código reutilizable. En un nivel alto, los componentes son elementos personalizados a los que el compilador de Vue adjunta comportamiento.

En Vue, un componente es esencialmente una instancia de Vue con opciones predefinidas. A continuación, se presenta el siguiente fragmento de código que contiene un ejemplo de un componente Vue. Dicho componente muestra un botón e imprime el número de veces que se hace clic en el mismo.

## Tabla 2

### *Ejemplo de un componente Vue*

```
<template>
  <div id="tuto">
    <button-clicked v-bind:initial-count="0"></button-clicked>
  </div>
</template>
<script>
Vue.component('button-clicked', {
  props: ['initialCount'],
  data: () => ({
    count: 0,
  }),
  template: '<button v-on:click="onClick">Clicked {{ count }} times</button>',
  computed: {
    countTimesTwo() {
      return this.count * 2;
    }
  },
  watch: {
    count(newValue, oldValue) {
      console.log(`The value of count is changed from ${oldValue} to ${newValue}.`);
    }
  },
  methods: {
    onClick() {
      this.count += 1;
    }
  },
  mounted() {
```

```
    this.count = this.initialCount;
  }
});
new Vue({
  el: '#tuto',
});
</script>
```

Nota: Esta tabla muestra un fragmento de código contiene de un componente Vue.

Vue utiliza una sintaxis de plantilla basada en HTML que permite vincular el DOM renderizado a los datos de la instancia subyacente de Vue. Todas las plantillas de Vue son HTML válido que puede ser analizado por navegadores que cumplan con las especificaciones y por analizadores HTML.

Adicionalmente Vue compila las plantillas en funciones de representación de DOM virtual. Un Modelo de Objeto de Documento virtual (o "DOM") permite a Vue renderizar componentes en su memoria antes de actualizar el navegador. Combinado con el sistema de reactividad, Vue es capaz de calcular el número mínimo de componentes a ser renderizados y aplicar la cantidad mínima de manipulaciones DOM cuando el estado de la aplicación cambia.

Los usuarios de Vue pueden usar la sintaxis de la plantilla o elegir escribir directamente las funciones de representación usando JSX. Las funciones de representación permiten que la aplicación se construya a partir de componentes de software.

Vue cuenta con un sistema de reactividad que utiliza objetos de JavaScript simples y una retransmisión optimizada. Cada componente mantiene un registro de sus dependencias reactivas durante su renderización, de modo que el sistema sabe con precisión cuándo volver a renderizar, y qué componentes volver a renderizar.

Vue proporciona una variedad de formas de aplicar efectos de transición cuando se insertan, actualizan o eliminan elementos del DOM.

Esto incluye herramientas para:

- Aplicar automáticamente clases para transiciones y animaciones CSS
- Integrar bibliotecas de animación CSS de terceros, como Animate.css
- Usar JavaScript para manipular directamente el DOM durante los hooks de transición.
- Integrar bibliotecas de animación JavaScript de terceros, como Velocity.js

Cuando un elemento envuelto en un componente de transición es insertado o removido, esto es lo que sucede:

1. Vue analizará automáticamente si el elemento objetivo tiene transiciones o animaciones CSS aplicadas. Si es así, las clases de transición CSS se añadirán/eliminarán en los momentos apropiados
2. Si el componente de transición proporcionó ganchos de JavaScript, estos ganchos serán llamados en el momento apropiado.
3. Si no se detectan transiciones/animaciones CSS y no se proporcionan ganchos de JavaScript, las operaciones DOM de inserción y/o extracción se ejecutarán inmediatamente en el siguiente cuadro

Por otro lado, Angular es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

La biblioteca lee el HTML que contiene atributos de las etiquetas personalizadas adicionales, entonces obedece a las directivas de los atributos personalizados y une las piezas de entrada o salida de la página a un modelo representado por las variables estándar de JavaScript.

Angular se basa en clases tipo "Componentes", cuyas propiedades son las usadas para hacer el binding de los datos. En dichas clases tenemos propiedades (variables) y métodos (funciones a llamar).

Entre las principales características de angular tenemos:

- **Generación de código:** Angular convierte tus plantillas en código altamente optimizado para las máquinas virtuales de JavaScript de hoy en día, ofreciéndote todas las ventajas del código escrito a mano con la productividad de un framework.
- **Universal:** Ejecuta la primera vista de tu aplicación en node.js, .NET, PHP, y otros servidores para renderizado de forma casi instantánea obteniendo solo HTML y CSS. También abre posibilidades para la optimización del SEO del sitio, incluyendo configuración.
- **División del código:** Las aplicaciones de Angular se cargan rápidamente gracias al nuevo enrutador de componentes. Este ofrece una división automática de códigos para que los usuarios solo carguen el código necesario para procesar la vista que solicitan.
- **Plantillas:** Permite crear rápidamente vistas de interfaz de usuario con una sintaxis de plantilla simple y potente.
- **Angular CLI:** Las herramientas de línea de comandos permiten empezar a desarrollar rápidamente, añadir componentes y realizar test, así como pre visualizar de forma instantánea la aplicación.

- **IDEs:** Obtén sugerencias de código inteligente, detección de errores y otros comentarios en la mayoría de los editores populares e IDEs.

## **Marco Metodológico**

El método utilizado en la realización de este estudio de caso consistió en consultar información obtenida de contenido de artículos, revistas y libros teniendo como conclusión que la metodología utilizada es bibliográfica.

Esta se basa en la selección y recopilación de información por medio de documentos o materiales utilizados como fuentes de referencias para el estudio a realizarse teniendo como objetivo ejecutar un análisis basado en las características y puntos claves de los frameworks Vue.js y Angular usados por desarrolladores expertos o principiantes en la creación de páginas web.

## Resultados

Angular ha estado evolucionando rápidamente con actualizaciones constantes, lanzando funcionalidades adicionales para acelerar el desarrollo y ofrecer un entorno de desarrollo sin esfuerzo y una experiencia de usuario fluida.

Especialmente después del lanzamiento de Angular 2, se ha convertido en un importante marco de interfaz para crear interfaces de usuario y manejar la interfaz de forma independiente. Angular ya está a punto de brindar una experiencia de desarrollo espléndida al poner el compilador Ivy en su forma completa.

Aunque el marco parece un poco antiguo en comparación con los marcos desarrollados recientemente como Vue, no deja nada sin remover para mantenerse al día con los avances recientes en la experiencia del usuario y las necesidades de desarrollo. Angular es una opción obvia para crear aplicaciones empresariales debido a sus amplias funcionalidades integradas y soporte comunitario.

Vue es un marco técnicamente sólido cuando se trata de crear interfaces de usuario y resolver problemas complejos. Aunque Vue tiene una excelente columna vertebral, no es tan popular como Angular.

Pero esto no define el futuro de Vue porque el marco tiene un ecosistema robusto con una sintaxis bastante buena. Los desarrolladores han mostrado interés en adoptar el marco a pesar de que tienen marcos con todas las funciones como Angular. Es probable que Vue traiga actualizaciones como Angular en sus próximas versiones.

## Discusión de Resultados

Vue.js es un marco front-end progresivo para construir interfaces de usuario usando JavaScript y HTML. Está diseñado de tal manera desde abajo para ser adoptable de forma incremental. Vue.js es ampliamente utilizado para el desarrollo web. Angular es un marco estructural JavaScript del lado del cliente con arquitectura de controlador de vista de modelo (MVC) y modelo de vista de modelo (MVVM) para aplicaciones web dinámicas.

Finalmente, podemos decir que Vue.js tiene más ventajas que Angular.js y Vue.js funciona a la perfección para integrarse con otras aplicaciones, mientras que Angular.js tiene restricciones. Entonces, según los requisitos del proyecto y el marco de tiempo, decida el marco que mejor se adapte al proyecto.

En la siguiente tabla podemos ver diferentes aspectos en los cuales se apalanca el resultado de esta comparación.

**Tabla 3**

*Cuadro comparativo entre angular y vue.*

<b>PUNTOS DE DISCUSIÓN</b>	<b>ANGULAR</b>	<b>VUE</b>
<b>COMPLEJIDAD</b>	Angular.js no es tan simple como Vue.js en términos de diseño y API como se crea vue.js después del lanzamiento de los principales Frameworks. Angular.js tardará más tiempo en desarrollar aplicaciones no triviales en comparación con Vue.js.	Vue.js es muy simple en términos de diseño y API, con algunos conocimientos para construir aplicaciones no triviales, tomará menos de un día con Vue.js
<b>INTEGRACIÓN FÁCIL</b>	Angular.js también es fácil de integrar, pero con componentes de terceros y JavaScript es popular, muchos desarrolladores desarrollaron componentes que se pueden	Vue.js es fácil de integrar con otras bibliotecas front-end, lo que lo convierte en un marco versátil y fácil de usar. Si falta alguna característica o necesita alguna mejora, podemos integrarla fácilmente con la biblioteca.

	integrar en cualquier aplicación angular.js	
<b>ENLACE DE DATOS</b>	En Angular.js, utiliza el enlace bidireccional entre ámbitos, pero si queremos conectarnos con servicios asíncronos necesitamos tener componentes desarrollados por terceros para integrarlos con la aplicación.	Vue.js utiliza el flujo de datos en el camino entre los componentes que facilita el flujo de datos y lleva a desarrollar aplicaciones no triviales en menos tiempo. Cuando nos conectamos con servicios asíncronos, podemos usar bibliotecas si es necesario.
<b>FLEXIBILIDAD</b>	Angular.js es más flexible ya que ofrece soporte oficial para una variedad de sistemas sin restricciones en la estructura o aplicación del proyecto, ya que a muchos desarrolladores les encanta no tener reglas, por supuesto. Estas características hacen que angular.js sea muy flexible	Vue.js no es tan flexible como Angular.js, ya que es menos obstinado ya que tiene algunas reglas para seguir la estructura del proyecto al desarrollar aplicaciones, pero es más flexible para soluciones modulares que lo hacen adaptable a una amplia variedad de proyectos. Entonces Vue.js también es flexible, no como Angular.js
<b>CURVA DE APRENDIZAJE</b>	La curva de aprendizaje de Angular.js es más pronunciada ya que la documentación de la API es muy grande y los usuarios deben familiarizarse con algunos conceptos antes y su complejidad se debe en gran medida a su diseño. Es difícil para las personas con menos experiencia recoger y desarrollar una aplicación.	La curva de aprendizaje de Vue.js no es más pronunciada, ya que si el usuario conoce HTML, CSS y JavaScript, puede desarrollar aplicaciones de una sola página y desarrollo de aplicaciones no triviales en menos de un día. Entonces la curva de aprendizaje es fácil para Vue.js
<b>MECANOGRAFIADO</b>	Angular.js utiliza el mecanografiado, ya que se requiere esencialmente ya que todas las fuentes de aprendizaje y la documentación se basa en mecanografiado, lo que ofrece la comprobación de tipos estáticos para aplicaciones a gran escala. Es un impulso para los desarrolladores con Java como fondo, pero tiene sobrecarga	Vue.js no tiene mecanografiado, pero tiene algunos tipos de decoradores oficiales y están interactuando con el equipo mecanografiado de Microsoft y Vasco de team para apoyar con Vue.js.

*Elaborado por: Tito Johnnier Barboto Álvarez*

## Conclusiones

De acuerdo con el desarrollo del caso de estudio, ambos frameworks ofrecen al usuario la facilidad para resolver y simplificar el código al momento de desarrollar aplicaciones web y móviles. No obstante, Vue.js, siendo un framework progresivo hace un gran trabajo al no depender necesariamente de una codificación desde cero, de manera que Vue.js se puede adaptar sencillamente al código ya precodificado si utilizamos en los inicios del proyecto Angular, React.js o cualquier otro framework frontend, hecho que no ocurre cuando hacemos uso de Angular debido a su limitada cooperación en un proyecto que no se encuentra codificado desde el inicio con dicho framework.

A diferencia de Angular que requiere un previo nivel de aprendizaje en cuanto a programación JavaScript modelado con TypeScript se refiere, demostrando su usabilidad en la demanda para el desarrollo avanzado de aplicaciones dinámicas multiplataforma a gran escala, Vue.js resalta por su fácil e intuitiva codificación al depender en menor medida de un conocimiento básico de JavaScript. Además Vue.js incorpora lo mejor de la librería React.js como lo es el virtual DOM que proporciona un mejor renderizado en poco tiempo y, adicionando de Angular las directivas que sirven para el modelado en la estructura del código.

## **Recomendaciones**

Vue.js al ser adaptable a todos los tipos de frameworks por su sistema progresivo es recomendable para la creación de proyectos pequeños donde no es necesario saber sobre desarrollo web avanzado, con el uso de este framework los proyectos pueden ir creciendo junto al conocimiento del desarrollador con su sistema de componentes básicos. Angular es recomendable para los desarrolladores que cuentan con conocimiento en el desarrollo web donde este usa herramientas más avanzadas que las de vue.js como el sistema de TypeScript una versión mejorada de JavaScript.

Vue.js es bueno para los estudiantes que desean aprender un framework de manera rápida y fácil, se recomienda por su curva de aprendizaje baja que cuenta con una codificación de JavaScript básica haciendo el desarrollo más dinámico haciendo posible terminar una aplicación en menos de un día permitiendo adaptar su codificación de manera fácil, Angular se recomienda para crear aplicaciones robustas, se puede aprender angular desde cero con su funcionamiento rápido pero tomará más tiempo que aprender Vue.js debido a su curva de aprendizaje alta donde la documentación de la API es extensa y los usuarios deben familiarizarse con algunos conceptos antes del desarrollo de los proyectos.

## Referencias

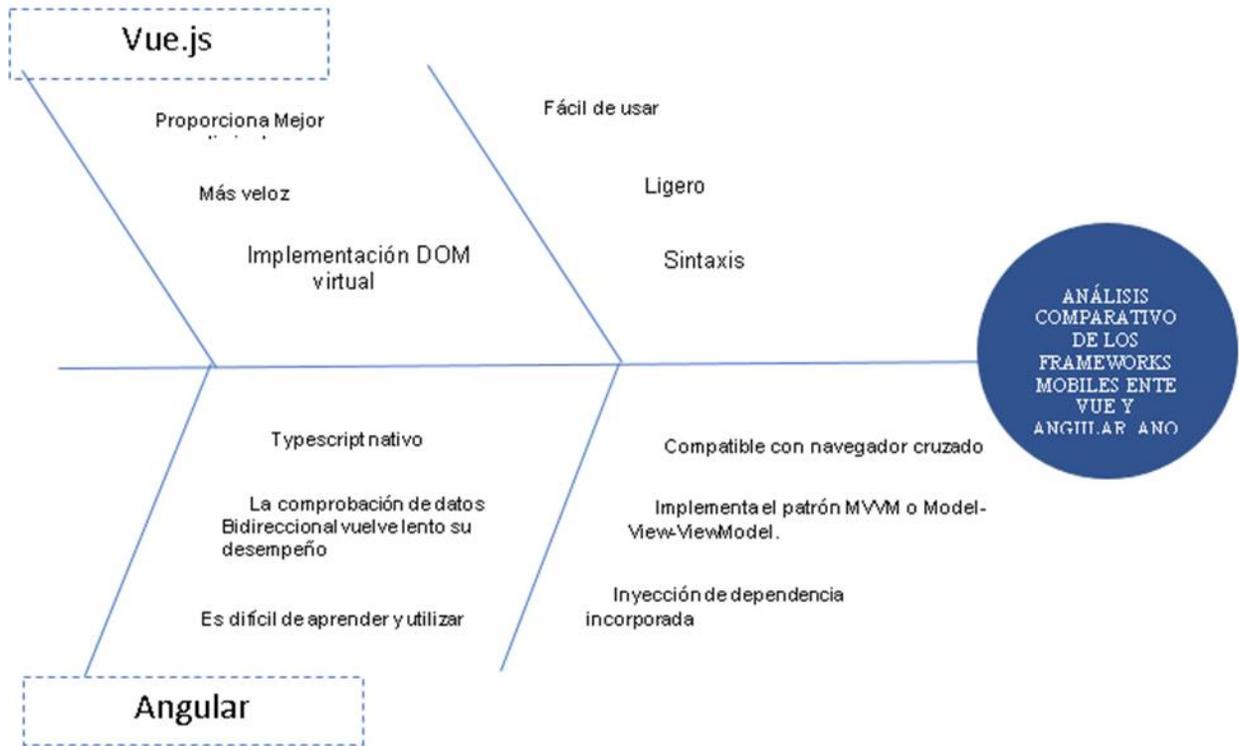
- Boada Oriols, M., & Gómez Gutiérrez, J. A. (2018). *El gran libro de Angular*. Barcelona: MARCOMBO.
- Alano, D. (2021, septiembre 5). *isbel*. Obtenido de isbel: <https://isbel.com/seguridad-de-la-informacion-vulnerabilidades-riesgos/>
- Alias Mendoza, R., & Santos del Carpio, G. (2013, Diciembre). Página web y aplicación en Android para el módulo de inscripción y acreditación del departamento de actividades extraescolares del Instituto Tecnológico de Tuxtla Gutiérrez. Chiapas, Tuxtla Gutiérrez, Mexico: INSTITUTO TECNOLÓGICO DE TUXTLA GUTIÉRREZ.
- Alves, C., & Bach, J. (2021). *TypeScript para principiantes*. Independently.
- BERTOLIN, J. A. (2008). *Seguridad de la información. Redes informática y sistemas de información*. . España: Paraninfo.
- Boada Oriols, M., & Gómez Gutiérrez, J. A. (2019). *El gran libro de Angular*. Mexico: Alfaomega.
- Burbano, V. (2021, 04 26). *VUE.JS es uno de los mejores frameworks para el frontend que te hace la vida más fácil*.
- Caffaratto Grandes, M. (2017, 04 27). *adictosaltrabajo*. Obtenido de adictosaltrabajo: <https://www.adictosaltrabajo.com/2017/04/27/introduccion-a-vue-js/>
- campusmvp. (2021, 07 07). *campusmvp*. Obtenido de campusmvp: <https://www.campusmvp.es/recursos/post/que-es-angular-para-que-sirve-y-por-que-aprenderlo.aspx>
- Cardona, O. D. (s.f.). *desenredando*. Obtenido de desenredando: <https://www.desenredando.org/public/libros/1993/ldnsn/html/cap3.htm>
- Chuburu, L. (2020). *Introducción al lenguaje HTML*.
- Cuéllar Martínez, H. G. (2021, 04 20). *VUE (CONSTRUCCIÓN DE INTERFACES)*.
- De Pablos Heredero, C., López Hermoso Agius, J. J., Martín-Romo Romero, S., & Medina Salgado, S. (2019). *Organización y transformación de los sistemas de información en la empresa*. Madrid: ESIC editorial. Obtenido de Dialnet.
- Desongles Corrales, J. (2006). *Técnicos de Informática Del Servicio Vasco de Salud-osakidetza. Temario Ebook*. España: Mad.
- Dimes, T. (2015). *JavaScript Una Guía de Aprendizaje para el Lenguaje de Programación JavaScript*.
- García de Jalón, J., Rodríguez, J. I., & Imaz, A. (1999, abril). *Aprende Servlets de Java como si estuviera en primero*. San Sebastián, Navarra, España.

- García de Zúñiga, F. (2020, 05 27). *Vue.js: Qué es y por qué usarlo como framework de referencia*.
- Gomez, A. (2014). *Enciclopedia de la seguridad informatica*. España: RA-MA.
- Gonçalves, M. J. (2021, 11 24). *Angular vs React vs Vue*.
- implementandosgi. (s.f.). *implementandosgi*. Obtenido de implementandosgi:  
<https://www.implementandosgi.com/procesos/analisis-de-vulnerabilidad-plan-de-emergencias/>
- Lederkremer, M. (2019). *Redes informaticas*. Buenos aires: REusers.
- Romero, M. y. (2018). *introduccion a la seguridad informatica y el analisis de las vulnerabilidades*. Manabi: Editorial Area de innovacion y desarrollo 3ciencias.
- Ruiz Calle, J. (2018, junio 6). *Modelización de Web Components reutilizables para simplificar el proceso de paso de información en aplicaciones web*.
- San Martín Calera, M. A. (2007, Septiembre). *Curso AJAX Y DOM 2007*.
- Stackoverflow. (2021, Octubre 26). *Top Framework*. Obtenido de Top Framework:  
<https://stackoverflow.com/>
- Tarazona, C. (2007). Amenazas informaticas y seguridad de la informacion. *HeinOnline*, 137.  
Obtenido de HeinOnline:  
<https://heinonline.org/HOL/LandingPage?handle=hein.journals/dpencrim28&div=29&id=&page=>
- WROV. (2018, 01 08). *vfpavanzado.wordpress.com*. Obtenido de vfpavanzado.wordpress.com:  
<https://vfpavanzado.wordpress.com/2018/01/08/diferencias-entre-angular-y-angularjs/>

## Anexos

### Anexo 1

#### Diagrama de Ishikawa



Nota: Diagrama de Ishikawa mostrando la causa y el efecto de Angular y Vue.js