

**REPORTE FINAL PARA ACREDITAR LA RESIDENCIA
PROFESIONAL DE LA CARRERA DE INGENIERÍA EN
TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES**

**AÑO Y PERIODO DE ELABORACIÓN:
JULIO-DICIEMBRE DEL 2022**

**PRESENTA:
JAQUELINE GARCIA LUEVANO**

**CARRERA:
INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIONES**

***APLICACIÓN WEB PARA LLEVAR EL CONTROL DE LAS REVISIONES DE
ARTICULOS CIENTÍFICOS***

Laboratorio de Iluminación Artificial (LIA)



Nivia Iracemi Escalante García.

Edwin Celestino García Alcocer.

Pabellón de Arteaga, Aguascalientes a 02 de diciembre de 2022

Índice

CAPÍTULO 1: PRELIMINARES	3
1. Agradecimientos.	3
2. Resumen.	4
3. Lista de Tablas	5
4. Lista de Figuras	5
CAPÍTULO 2: GENERALIDADES DEL PROYECTO	6
5.- Introducción	6
6. Descripción de la empresa u organización y del puesto o área del trabajo del residente.	7
7. Problemas a resolver, priorizándolos.	11
8. Justificación	12
9. Objetivos (General y Específicos).....	12
CAPÍTULO 3: MARCO TEÓRICO	14
10. Antecedentes	14
10.2 Aspectos generales	15
10.2.1 Revisión por pares.....	15
10.2.2 Tipos de RPP	15
10.2.3 Proceso de la revisión por pares.....	16
10.3 Aplicación Web	17
10.3.1 Características y Ventajas de las Aplicaciones Web.....	17
10.3.2 Usos Comunes de las Aplicaciones Web.....	18
10.4 Modelo-Vista-Controlador (MVC)	18
10.5 FrontEnd	19
10.6 React Js.....	20
10.6.1 Ventajas de React Js.....	21
10.6.2 JavaScript XML (JSX).....	21
10.7 Bootstrap	21
10.8 HTTP	22
10.9 Axios.....	23
10.10 Backend.....	23
10.11 API.....	23
10.12 PHP (acrónimo recursivo de Hypertext Preprocessor)	24

10.13 Laravel.....	25
10.14 Postman.....	25
10.15 Base de datos	26
10.16 Sistema Gestor de Base de Datos (SGBD)	26
10.17 MySQL.....	26
10.18 Servidor Web Apache.....	27
10.19 ORM (Object Relational Mapping).....	28
10.20 Eloquent.....	29
CAPÍTULO 4: DESARROLLO	30
11. Procedimiento y descripción de las actividades realizadas.	30
11.1 Definición de los objetivos del prototipo	31
11.2 Validación de los requerimientos a través de la funcionalidad del producto.....	32
11.3 Diseñar y desarrollar el prototipo	34
11.4 Evaluar el funcionamiento del prototipo.....	47
CAPÍTULO 5: RESULTADOS	49
CAPÍTULO 6: CONCLUSIONES.....	52
CAPÍTULO 7: COMPETENCIAS DESARROLLADAS.....	53
CAPÍTULO 8: FUENTES DE INFORMACIÓN	54
CAPÍTULO 9: ANEXOS	55

CAPÍTULO 1: PRELIMINARES

1. Agradecimientos.

A dios por brindarme salud y aptitudes para continuar con mi educación a lo largo de los años.

A mi madre Maricela por su paciencia, apoyo y cuidado en el ámbito físico y mental, por recordarme que soy una mujer capaz de lograr aquello que quiera, a mi padre Ernesto que sé que desde el cielo me cuida, está orgulloso de mí y de lo que he logrado.

A mi hermano Ernesto por ser uno de los pilares fundamentales en mi vida, por el cariño, enseñanzas, las risas compartidas, por las palabras de aliento, por apoyarme en mis decisiones y no dudar de mí, incluso cuando yo lo hacía.

A mi tía Sandra por su apoyo a lo largo de los años y por brindarme la oportunidad de seguir estudiando sin dudar de mí y mis capacidades.

A mi abuelo Jesús por enseñarme el valor del esfuerzo y trabajo, a mi abuela Ma. del Carmen por siempre incluirme en sus oraciones.

A aquellos amigos que me acompañaron durante el proceso de mi formación universitaria, por festejar conmigo mis logros y por estar presentes en aquellos momentos no tan buenos.

A los profesores que genuinamente se involucraron en mi formación profesional, los cuales me brindaron su enseñanza, apoyo, experiencia, consejos, retroalimentación y oportunidades de trabajo a lo largo de mi estancia universitaria.

2. Resumen.

El objetivo del presente proyecto es crear una aplicación web para llevar el control de las revisiones de artículos científicos que los investigadores tienen asignados en su proceso de revisión por pares. Considerando la gran importancia que tiene la inclusión de las Tecnologías de la Información y Comunicaciones (TICs) en la vida cotidiana y profesional, una herramienta como la mencionada es de gran utilidad para dicho propósito ya que agiliza el proceso y la calidad de las revisiones.

La aplicación se ha desarrollado bajo la metodología de desarrollo por prototipo. Con tecnologías de desarrollo web tales como React Js, PHP y MySQL. Se obtuvieron como resultado las interfaces gráficas de usuario desarrolladas con componentes, los modelos y controladores los cuales a su vez se conectan a la base de datos para realizar las manipulaciones de datos necesarias. Es decir, los módulos del FrontEnd y BackEnd.

3.Lista de Tablas

Tabla 1. RF1	32
Tabla 2. RF2	32
Tabla 3. RF3	32
Tabla 4. RF4	32
Tabla 5. RF5	33
Tabla 6. RF6	33
Tabla 7. RF7	33
Tabla 8. RF8	33
Tabla 9. RF9	34

4.Lista de Figuras

Figura 1. Vista satelital del Instituto Tecnológico de Pabellón de Arteaga, Aguascalientes (22°09'40"N 102°16' 08"W), Fuente: Google Earth.	7
Figura 2. Organigrama de la empresa, Fuente LIA.	10
Figura 3. Tipos de RPP, Fuente (Ladrón de Guevara Cervera et al., 2008).	16
Figura 4. Modelo Vista Controlador, Fuente: (Garcia Islas et al., 2021).....	19
Figura 5. Ciclo de metodología por prototipo, Fuede: Elaboración propia,2022.....	30
Figura 6. Arquitectura de la aplicación. Fuente: Elaboración propia	35
Figura 7. Diagrama de flujo de la arquitectura de la aplicación	36
Figura 8. Diagrama de casos de uso.....	37
Figura 9. Diagrama de flujo para Login.	38
Figura 10. Diagrama de flujo para registrar usuario.....	39
Figura 11. Diagrama de flujo para ejecutar una revisión.	40
Figura 12. Diagrama de flujo para agregar una revisión.....	41
Figura 13. Diagrama físico de la base de datos	42
Figura 14. Bocetos de la aplicación web	43
Figura 15. Versión instalada de Laravel.	44
Figura 16. Versión instalada de Reac Js	44
Figura 17. Sistema Gestor de Base de Datos	44
Figura 18. Permisos del usuario bdAdmin.....	45
Figura 19. Edición del archivo .env para conexión a base de datos.	46
Figura 20. Programación de la semilla de la tabla Tc_Categorias.....	46
Figura 21. Inserción de un usuario a la base de datos.	48
Figura 22. Revistas con las que cuenta un usuario.	48
Figura 23. Base de datos de la aplicación.....	49
Figura 24. Vistas de la aplicación.....	50

CAPÍTULO 2: GENERALIDADES DEL PROYECTO

El planteamiento del objeto a desarrollar es fundamental al inicio de cualquier proyecto ya que permite delimitar la meta que se pretende lograr, qué problemática se atiende y la importancia de resolverla. La estructura de la investigación concede una visión detallada de lo que se pretende alcanzar, la información presentada en este capítulo aborda los aspectos generales para la comprensión del proyecto, así como los detalles de la elaboración y aplicación.

5.- Introducción

La revisión por pares de artículos científicos es uno de los procesos con más relevancia dentro de la publicación de los mismos, ya que a través de dichas revisiones los expertos ayudan a mejorar los trabajos próximos a publicar dando así validez y confianza científica.

Actualmente, el proceso se lleva a cabo de manera manual ocasionando retrasos en las revisiones y falta de organización por lo cual se tiene como objetivo crear una aplicación web que facilite el proceso de las revisiones ofreciendo al usuario comodidad, eficacia y dinamismo en sus revisiones. La aplicación web destaca entre sus actividades principales el registro para obtener una cuenta, la carga de archivos, registro de las revistas y comentarios colocados en el archivo.

A lo largo del documento se abordará lo relacionado con la elaboración de dicha aplicación web tales como la justificación, los objetivos, fundamentos teóricos y las actividades realizadas para solucionar la problemática detectada.

6. Descripción de la empresa u organización y del puesto o área del trabajo del residente.

Las residencias tuvieron lugar en el Laboratorio de Iluminación Artificial (LIA) dentro del Tecnológico Nacional de México campus Pabellón de Arteaga (ITPA). El ITPA fue fundado el 01 de septiembre de 2018, ubicado sobre Carretera la Estación de Rincón Km 1, C.P. 20670 en Pabellón de Arteaga, Aguascalientes, México. La Figura 1 muestra la posición geográfica de la Institución.



Figura 1. Vista satelital del Instituto Tecnológico de Pabellón de Arteaga, Aguascalientes (22°09'40"N 102°16'08"W), Fuente: Google Earth.

Gracias al apoyo recibido por parte de CONACyT con la Convocatoria Convocatoria de Infraestructura Científica y Tecnológica del CONACyT (INFRA-2016-01, Project No. 270665) en el ITPA se construyó el Laboratorio de Iluminación Artificial (LIA). Parte del equipamiento se ha consolidado con apoyos federales y estatales como CB-2016-01, Project No.287828, IDSCEA, SADER, por mencionar algunos.

LIA establece como objetivo principal profundizar y clarificar fenómenos no explicados hasta el momento de los efectos de la luz (continua y pulsada) en plantas y alimentos como su caracterización biológica con herramientas tecnológicas actuales LEDs teniendo el nivel de competitividad que otros laboratorios en el mundo.

Misión

Proyectar e impulsar el desarrollo de nuevas actividades en el sector agroindustrial a través de procedimientos que ayuden a mejorar la calidad de vida alrededor del mundo, generando productos agroalimentarios con valor agregado.

Visión

Ser un centro capaz de impulsar, desarrollar y fortalecer iniciativas que conlleven a la productividad y rentabilidad agroalimentaria; estableciendo nuevas filosofías de competitividad y calidad en el sector agroindustrial.

En LIA, los proyectos de investigación básica y aplicada han sido planeados y desarrollados de acuerdo al enfoque de la SADER federal y de la Secretaría de Desarrollo Rural y Agroempresarial del Gobierno del Estado de Aguascalientes, impulsando las actividades del sector agroalimentario y agroindustrial, además de la Electrónica y las Tecnologías de la Información y Comunicaciones (TICS) que son estrategias de la agenda de Innovación del Estado creada por CONACyT, junto con los objetivos de la agenda de Seguridad Alimentaria del Plan Nacional de Desarrollo y algunos de los objetivos (2, 11, y 12) de la Organización de las Naciones Unidas.

Las acciones planeadas dentro de LIA son puntos críticos presentados por instituciones como la FAO (La Organización de las Naciones Unidas para la Alimentación y la Agricultura), OMS (Organización Mundial de la Salud), Programa Mundial de Alimentos (PMA), FIDA (Fondo Internacional de Desarrollo Agrícola) y son Agricultura Sustentable y Protegida, Seguridad Alimentaria, Enfermedades Transmitidas por los Alimentos, Sanidad Vegetal, entre otros.

Dentro de las principales actividades de la empresa se encuentran:

- a. Diseño, construcción e implementación de sistemas de producción multinivel en espacios cerrados con luz artificial LED que permiten potenciar el contenido

- nutrimental de cultivos como microgreens, lechuga, lenteja, espinaca, albahaca, alfalfa, entre otras, a través de recetas de luz (diferentes combinaciones de color).
- b. Desarrollo de Sistemas de Desinfección de Alimentos con Radiación Ultravioleta LED tipo A, B y C Aplicados a Productos Agroindustriales y en Fresco.
 - c. Implementar estrategias tecnológicas con luz artificial tipo LED para preservar, extender y/o acelerar el tiempo de vida en almacén de frutas y verduras, además de analizar el efecto sobre la biosíntesis de compuestos (licopeno, capsaicina, entre otros.) durante su estancia en el anaquel.
 - d. Investigación, integración y aplicación de la agricultura de precisión con el uso de tecnología aérea no tripulada (VANTs) para el monitoreo y detección de plagas en diferentes cultivos con el objetivo de evitar pérdidas en la producción.
 - e. Radiación Artificial Multiespectral Tipo LED para la Detección de Contaminantes en la Industria Alimentaria.
 - f. Medición de la cantidad de Absorción en Plantas de Lechuga con Diferentes Recetas de Luz.
 - g. Sistema Multinivel de Producción de Plantas para Consumo Humano.
 - h. Zonas de Radiación LED para Optimizar el Crecimiento de Plantas para Consumo Humano.

El equipo de LIA está conformado por Investigadores y Estudiantes de Doctorado, Maestría y Licenciatura tal como se muestra en la Figura 2:

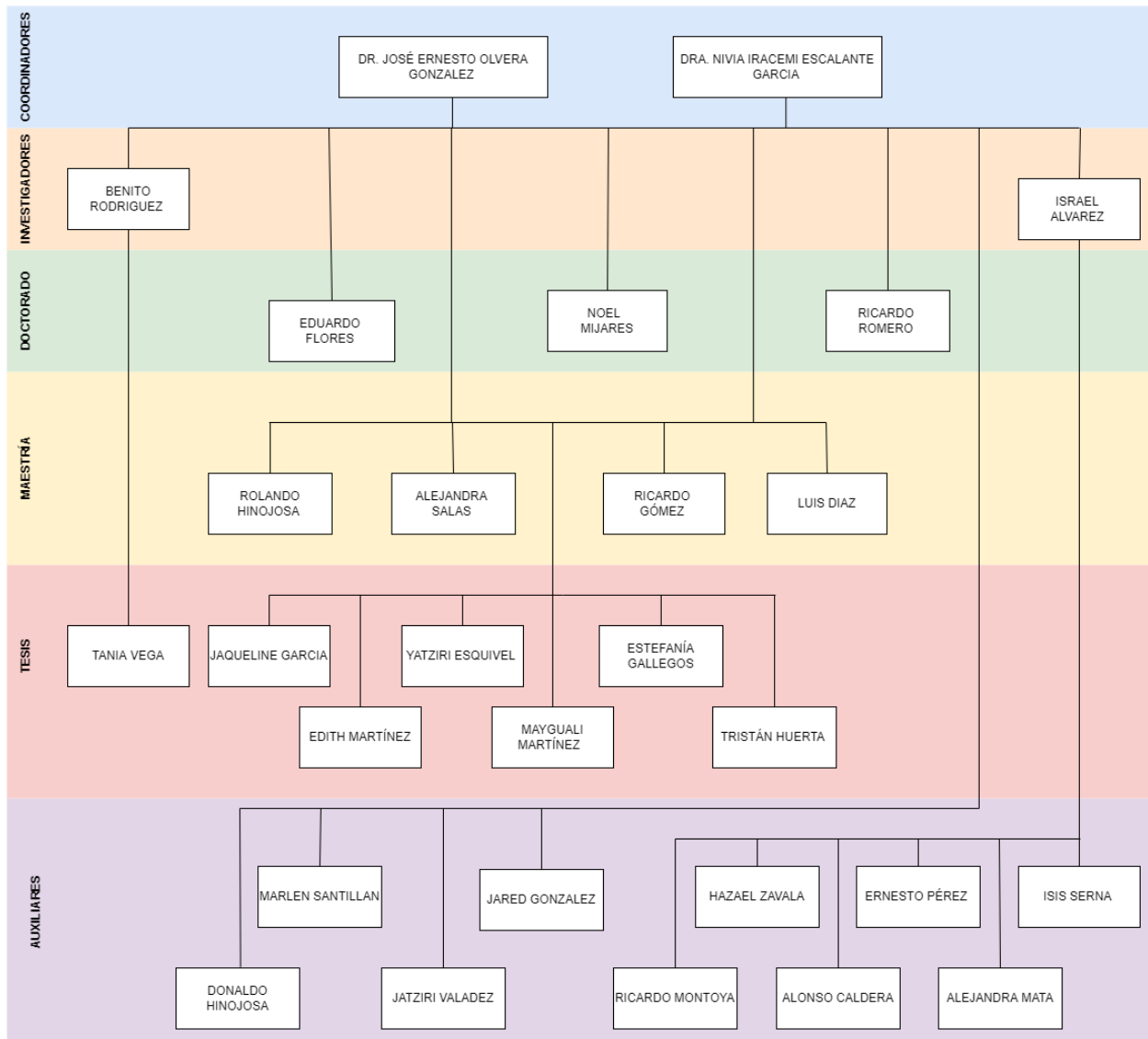


Figura 2. Organigrama de la empresa, Fuente LIA.

Relevancia

LIA es el único Laboratorio en el país especializado en la aplicación de la luz artificial LED para el desarrollo de proyectos e investigaciones de gran impacto en el sector agroindustrial, involucrando diversas áreas para lograr las metas y objetivos. Se incluyen las Tecnologías de la Información y Comunicación ofreciendo digitalización, procesamiento, almacenamiento y el óptimo acceso de la información para que las investigaciones puedan alcanzar mayor impacto.

El residente se desarrolló en el Laboratorio de Iluminación Artificial, desarrollando capacidades intelectuales y habilidades de razonamiento lógico a través de la programación de nuevos sistemas informáticos. De igual forma, fueron fortalecidos los valores de responsabilidad y compromiso. Dado que las habilidades de adquirir conocimientos por sí mismo, han sido trascendentales para la elaboración e implementación de la propuesta. Además, detectó, propuso y solucionó problemas planteados respecto al tema en cuestión.

7. Problemas a resolver, priorizándolos.

Actualmente la revisión por pares se lleva a cabo de la manera tradicional, es decir, se examina el documento abriendo una pestaña para visualizarlo en el formato en el que está y posteriormente alguna otra aplicación para hacer las correcciones y anotaciones, causando así que el usuario llene su pantalla de ventanas las cuales saturan la vista y complican la interacción que tiene con las aplicaciones dando como consecuencia un bajo rendimiento en su revisión. Dado que los potenciales usuarios son investigadores los cuales efectúan varias revisiones durante un tiempo determinado, necesitan que el proceso de la revisión sea lo más optimizado posible para enfocarse únicamente en la calidad de su revisión y no en el manejo de las aplicaciones.

La presentación de la información es una parte crucial para que el usuario se sienta cómodo con las acciones que ejecuta desde una interfaz, por lo cual, el desconocimiento sobre qué herramientas son las más convenientes de usar también conlleva una pérdida de tiempo.

La organización es otro punto destacable debido a que es propenso a fallar ya que al tener mucho trabajo por realizar los usuarios tienden a descargar y colocar los archivos a revisar en sitios donde posteriormente no recuerdan o no encuentran fácilmente, lo cual a su vez provoca las dobles descargas y saturación de la memoria de almacenamiento del equipo de cómputo. Otro problema en cuanto a la organización es el pobre manejo de aquellos artículos que se encuentran pendientes por revisar, cuáles

son los que cuentan con un avance y cuáles los faltantes, originando que el usuario no cumpla con la revisión en tiempo y forma.

8. Justificación

La revisión por pares es un instrumento utilizado para la valoración crítica de los artículos antes de su respectiva publicación en revistas o sitios web, con el propósito de valorar la estructura, calidad y contenido de las investigaciones. Dentro de dicho proceso revisores externos proporcionan los comentarios que consideren necesarios para mejorar, corregir o complementar los trabajos. Actualmente, la revisión se lleva a cabo de manera manual, es decir, se colocan los respectivos comentarios cada vez que sea necesario, originando retrasos y desorganización. El uso de una aplicación web por parte de los revisores brinda la oportunidad de gestionar los artículos pendientes por examinar, además de la posibilidad de usar comentarios anteriormente registrados, lo cual proporciona ventajas, tales como agilizar los tiempos de revisión, identificar errores comunes y realizar las observaciones correspondientes. Adicionalmente, se pueden obtener estadísticas de errores más frecuentes cometidos en los artículos, brindando información que ayuda a investigadores a detectar los detalles a cuidar en nuevos documentos propios.

9. Objetivos (General y Específicos)

General:

Desarrollar una aplicación web para llevar el control de las revisiones de los artículos científicos indexados, arbitrados y de congresos que un investigador tiene asignado como parte de su proceso de revisión por pares.

Específicos:

- a. Evaluar los requerimientos para el desarrollo de la aplicación web a través de instrumentos de recolección de información.
- b. Seleccionar los elementos adecuados para la conformación de la BD a través de diagramas UML de entidad relación, diagrama clases, casos de uso, diagrama de actividades, diagrama de flujo, por mencionar algunos.

- c. Generar los diseños de las interfaces gráficas de usuario para el manejo, organización y representación de la información, considerando la interacción humano-computadora, reglas de Murch, teoría del color, entre otros.
- d. Preparar el entorno de proyecto a través de la instalación de ReactJS, NodeJs y MySQL.
- e. Programar cada elemento que conforma la aplicación web.
- f. Integrar el front-end y back-end, para la evaluación de su funcionamiento.
- g. Analizar y obtener estadísticas de los datos de los comentarios de revisores.
- h. Evaluar y retroalimentar el funcionamiento general del sistema a través de la aplicación de pruebas de software.

CAPÍTULO 3: MARCO TEÓRICO

Dentro del presente apartado se abordan los fundamentos teóricos referentes a la elaboración y desarrollo del proyecto. Se incluyen diversos autores que han estudiado el tema en cuestión con el fin de fomentar la comprensión del proyecto y ofrecer un respaldo a la información presentada.

10. Antecedentes

La necesidad de sustentar una investigación local y establecer un espacio para su difusión motivó a diversas instituciones y sociedades científicas a contar su propia revista, la que, en la medida que crece y va obteniendo un prestigio, recibe o acepta artículos de otras instituciones locales y de distintas latitudes del planeta; ganando visibilidad global e impacto científico.

En este camino de continuo crecimiento y consolidación editorial, es de gran relevancia contar con el respaldo y compromiso de una institución que permita a los autores enviar sus escritos, además, de que sean evaluados por editores y especialmente de los revisores que son eslabón fundamental en el proceso editorial a través de su participación en la “revisión por pares (RPP)”. Entonces, las revistas científicas consideran el proceso que avala un nuevo conocimiento que se publica en ellas. El proceso es conocido como revisión por pares -peer reviews- o RPP, donde se genera una valoración crítica y subjetiva de los manuscritos enviados a una revista científica por parte de expertos que no forman parte del personal editorial.

La Sociedad Real de Edimburgo en su revista *Mediacal Essays and Observatios* fue la pionera en aplicar la RPP en 1731, en dicho proceso se enviaron los artículos a personas conocedoras del tema antes de su publicación con el fin de evaluar los escritos, con ello el RPP se ha convertido en un proceso aplicado en múltiples instituciones a través de los años ya que confían en la validez que ofrece para la posterior publicación de los manuscritos (Martínez, 2012).

10.2 Aspectos generales

10.2.1 Revisión por pares

Es un instrumento usado para la valoración crítica de los escritos enviados a publicar con el fin de evaluar su rigurosidad científica, tomando en cuenta aspectos como novedad, coherencia, metodología, credibilidad, estructura, calidad y pertinencia (Ladrón de Guevara Cervera et al., 2008; Baiget, 2020).

Los expertos designados para la revisión ofrecen retroalimentación destacando los puntos fuertes y débiles del documento, además, de las recomendaciones para su mejora.

Los escritos que se evalúan con el RPP son:

- a. Artículos de investigación original.
- b. Artículos de revisión.
- c. Comunicaciones breves.
- d. Análisis o estudios de caso.
- e. Ensayos.

Así mismo la RPP cuenta con distintos tipos que, dependiendo de las revistas o plataformas, aplican de acuerdo a las necesidades y acuerdos a los que lleguen los involucrados.

10.2.2 Tipos de RPP

Existen tres tipos de revisión por pares las cuales se describen en la Figura 3

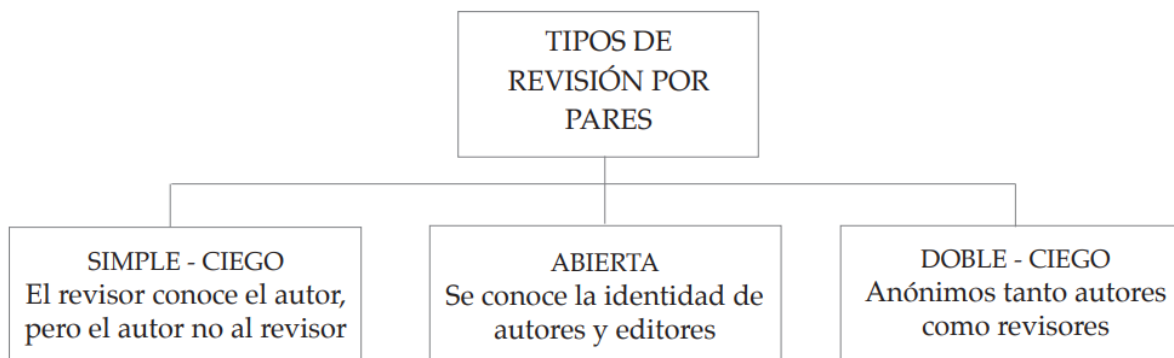


Figura 3. Tipos de RPP, Fuente (Ladrón de Guevara Cervera et al., 2008).

10.2.3 Proceso de la revisión por pares

El proceso de la revisión por pares se puede generalizar en los siguientes pasos:

- a) La revista recibe un manuscrito.
- b) Se comprueba que cumple con los requisitos formales que exige la revista.
- c) Se verifica que se haya realizado una revisión general del manuscrito por el editor de sección.
- d) Una vez que cumple con los pasos anteriores se envía a expertos en el tema del cual trata el manuscrito.
- e) Como parte inicial se procede a evaluar si es que cuenta con plagio a través de programas online.
- f) Posteriormente los editores deciden si se puede o no continuar con la revisión.
- g) Los pares revisores evalúan el manuscrito y determinan si se debe aceptar, rechazar o incluso pueden recomendar realizar modificaciones.
- h) Por último, los editores de las revistas determinan si el manuscrito se aprueba, rechaza o se modifica además de que se los comunica debidamente a los autores.
- i) Si es que existen modificaciones por parte de los autores se reingresa en el sistema de gestión de la revista y repite nuevamente el proceso (Robaina Castellanos & Semper González, 2019).

Actualmente, dentro del RPP se ven involucradas las Tecnologías de la Información y Comunicaciones (TICS) ya que, por lo regular, los escritos son enviados vía correo electrónico a los encargados de revisarlos. Sin embargo, las herramientas tecnológicas usadas para efectuar la revisión varían en cada persona, de ahí la importancia de crear alternativas en aplicaciones para que los usuarios puedan tener la libertad de elegir y adaptar su trabajo de revisión por pares a la era de la digitalización.

Para lograr dicho objetivo es necesario conocer los fundamentos tecnológicos detrás de la creación de la aplicación web que pretende dar una herramienta a los investigadores para efectuar la RPP (Robaina Castellanos & Semper González, 2019).

10.3 Aplicación Web

Según Lujan Mora (2002), “Una aplicación web (web-based application) es un tipo especial de aplicación cliente/servidor, donde tanto el cliente (el navegador, explorador o visualizador) como el servidor (el servidor web) y el protocolo mediante el que se comunican (HTTP) están estandarizados y no han de ser creados por el programador de aplicaciones”.

Las aplicaciones web poseen ventajas y cualidades propias que las permiten diferenciar sobre otras aplicaciones.

10.3.1 Características y Ventajas de las Aplicaciones Web.

Entre las características principales de las aplicaciones web se pueden encontrar:

- a. Herramienta a la que los usuarios pueden acceder a través de un servidor web.
- b. Se codifican en un lenguaje que sea soportado por los navegadores ya que de estos depende su ejecución.
- c. Se puede acceder a ella desde cualquier sitio y dispositivo siempre y cuando estos cuenten con un navegador y conexión a internet.

Por su parte en las ventajas destacan:

- a. La portabilidad.
- b. No ejecuta actualizaciones en cada cliente.
- c. No hay problemas de incompatibilidad.
- d. No son necesarias las instalaciones en cada cliente.
- e. Son aplicaciones más ligeras a comparación de las de escritorio.
- f. Consume pocos recursos del equipo donde se ejecuta.
- g. No se requieren características especiales para el equipo (Hernández Villena, 2018)

10.3.2 Usos Comunes de las Aplicaciones Web

El uso de una aplicación depende del ámbito y para el propósito para el cual fue desarrollada, sin embargo, en aspectos generales el uso que se le puede dar a estas son:

- a. Permite localizar información de manera rápida y sencilla.
- b. Dar de alta, actualizar o eliminar información u archivos.
- c. Ofrece organización y navegación a través de la aplicación web.
- d. Recoger, guardar y analizar los datos dados por los usuarios (Adobe, 2021)

Para que el flujo de información de la aplicación tenga un correcto funcionamiento es común usar patrones de diseño ya que de esta manera se controla lo que el usuario va a ver además de las acciones a ejecutar por parte de la aplicación y su relación con la base de datos.

10.4 Modelo-Vista-Controlador (MVC)

Es un patrón de diseño de software que divide las partes que conforman una aplicación web en las partes: modelo, vista y los controladores, con el objetivo de separar las funciones del código, reducir el esfuerzo en la programación, mejorar la percepción de

los errores, garantizar la actualización y el mantenimiento del software mediante la escalabilidad del modelo.

Este se clasifica en tres partes primordiales con el propósito de separar los datos, la interfaz de usuario y la lógica de control, cada una de las partes contemplan tareas específicas las cuales se describen a continuación:

- a. El modelo es el objeto que maneja, controla los datos y sus transformaciones, además de que es el encargado de la lógica para desplegar la información.
- b. Por su parte, la vista tiene la responsabilidad de visibilizar los datos del modelo y mostrarlos al usuario de manera gráfica y entendible.
- c. El controlador se encarga de interactuar con el modelo y la vista. Gestiona el flujo de información que provee el usuario mediante acciones y ejecuta cambios en el modelo (Orozco Aguirre, 2019).

La interacción de las partes del MVC se muestra en la Figura 4

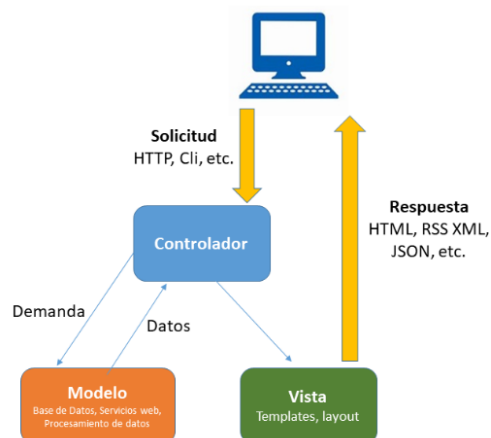


Figura 4. Modelo Vista Controlador, Fuente: (Garcia Islas et al., 2021).

10.5 FrontEnd

Un punto crucial en el ambiente de desarrollo es el FrontEnd conformado por diversos lenguajes como HTML, CSS y JavaScript que son ejecutadas por parte del usuario final

(cliente). El FrontEnd tiene como función presentar la información adecuada para que el cliente la pueda comprender e interactuar con ella.

El FrontEnd requiere de técnicas de experiencia de usuario para brindar una buena interacción humano-computadora, así como el manejo de las propias tecnologías (Pérez Ibarra et al., 2021).

Este tipo de desarrollo ha dado pauta a la creación de distintas tecnologías las cuales permiten agilizar la escritura, escalabilidad y mantenimiento de código, ofreciendo a los desarrolladores herramientas que les permiten seguir una estructura sin afectar la lógica que los mismos apliquen en la programación.

Algunas de las librerías y/o frameworks más populares para dicho propósito son Angular Js, Vue Js y React Js, de las cuales la última es la que se usará para el desarrollo de la presente aplicación web.

10.6 React Js

De acuerdo con su sitio oficial, React Js es una librería de JavaScript desarrollada y usada en producción por Facebook para construir interfaces de usuario.

Tiene la característica de ser declarativo ya que ayuda a crear interfaces de usuario interactivas de manera simple, también se encarga de actualizar y renderizar los componentes correctos cuando los datos cambian. Las vistas declarativas hacen que el código sea predecible y fácil de depurar.

React es basado en componentes, es decir, crea componentes encapsulados que manejan su propio estado y posteriormente los convierte en interfaces de usuario complejas. Un componente es como una función de JavaScript la cual acepta entradas arbitrarias y retorna elementos de React que describen lo que debe de aparecer en la pantalla (React Js Org, 2022).

Dicha librería destaca por sus cualidades y resulta una buena opción en el desarrollo FrontEnd gracias a las ventajas que ofrece.

10.6.1 Ventajas de React Js

- a. Sencillo de aprender dada su simplicidad en sintaxis.
- b. Flexibilidad y máxima responsividad.
- c. Modelo de Objetos del Documento o DOM (Document Object Model) virtual el cual permite convertir los distintos documentos tales como HTML, XHTML o XML en un árbol manejable para el navegador.
- d. Flujo de datos de “padre” a “hijo”, es decir, que los elementos tienen un lazo “hacia abajo”.
- e. Es de código abierto y con una gran comunidad detrás.
- f. Fácil migración entre versiones.
- g. Permite la elección de la arquitectura y herramientas a usar (Anchundia Medrano, 2022)

React Js toma el rol de desarrollar las vistas en el MVC y para poder mostrar cómo debe de lucir la interfaz de usuario se utiliza JavaScript XML o JSX por sus iniciales.

10.6.2 JavaScript XML (JSX)

Es una extensión de la sintaxis de JavaScript que une el maquetado y la lógica. Permite describir en JavaScript cómo se verán los componentes usando una estructura similar a HTML (React Js Org, 2022).

Así mismo la librería React Js se puede apoyar de marcos de trabajo (o frameworks en inglés), los cuales brindan componentes o plantillas predeterminadas que ayudan al desarrollador al momento de programar. Uno de los más populares es Bootstrap.

10.7 Bootstrap

Es un framework front-end utilizado para desarrollar sitios web responsivos compatibles con dispositivos móviles. Incluye plantillas de diseño basadas en HTML Y CSS tales como formularios, botones, tablas, barras de navegación, imágenes, carruseles, entre

otros. Adicionalmente involucra plug-ins de JavaScript para funciones y diseños responsivos (Shahu Gaikwad & Adkar, 2019).

Para usarlo en React Js dentro de un proyecto existen varias alternativas, sin embargo son dos las más populares. La primera es con Bootstrap CDN en la que no se necesitan descargas ni instalaciones ya que se vincula con el enlace en la aplicación. La segunda se involucra al manejador de paquetes que se esté usando en el proyecto ya que Bootstrap es instalado como dependencia para el proyecto. Siendo esta última la manera de aplicarla en el desarrollo de la aplicación web debido a que es el método más recomendado.

Una vez que el desarrollo FrontEnd se lleva a cabo con las tecnologías anteriormente mencionadas, es necesario establecer el contacto a través del protocolo de transferencia de hipertexto o HTTP (Hypertext Transfer Protocol) por sus siglas en inglés.

10.8 HTTP

Es el protocolo estandarizado de comunicación en base al modelo OSI (Open System Interconnection) que usan los navegadores y servidores para transferir información en formatos XML y HTML, dicha acción lo lleva a ser de los más utilizados dado a su trabajo en las páginas y aplicaciones web.

Está basado en la arquitectura cliente-servidor y diseñado bajo el paradigma de petición/respuesta lo cual permite que el intercambio de información entre ambas partes sea basado en solicitudes.

El protocolo HTTP define cuatro métodos principales con los cuales el cliente puede interactuar con los servicios ya sea para solicitar, crear, actualizar o borrar información, y estos son GET, POST, PUT y DELETE respectivamente (Carbonel Martínez, 2020).

Un ejemplo de su funcionamiento sería cuando un usuario solicita una página web, el navegador envía la solicitud de los objetos al servidor, este recibe la solicitud y envía una respuesta HTTP que contiene los objetos, posteriormente el navegador los recibe y los muestra al usuario final. Así mismo se ve involucrado el Protocolo de Control de

Transmisión o TCP (Transmission Control Protocol) por sus siglas en inglés ya que este evita la pérdida de información en la transmisión de información (Anturi Figueroa, 2020).

10.9 Axios

Según su sitio oficial “Axios es un cliente HTTP basado en promesasnode.js para el navegador. Es isomorfo (= puede ejecutarse en el navegador y nodejs con la misma base de código). En el lado del servidor usa el httpmódulo nativo node.js, mientras que en el cliente (navegador) usa XMLHttpRequests”.

Entre sus características principales se encuentra hacer XMLHttpRequests desde el navegador, hacer solicitudes http desde node.js, admite la API de Promise, transforma datos de solicitud de respuesta además de ofrecer transformaciones automáticas para datos JSON (Axios, 2022).

10.10 Backend

Dentro del desarrollo web se encuentra la parte del Backend la cual incluye la lógica de programación de la aplicación web por lo cual tiene la característica de que no es accesible para el usuario final. Se trabaja desde el lado del servidor y la persona encargada de esta parte debe de tener conocimiento de lenguajes del servidor tales como PHP, Java, C# y NodeJs, además de las distintas interacciones con las bases de datos (Pérez Ibarra et al., 2021).

Para el desarrollo del Backend se recurrió al desarrollo de una interfaz de programación de aplicaciones o API (application programming interface) por sus siglas en inglés.

10.11 API

Es la agrupación de definiciones y protocolos que son usados en el diseño e integración del software de las aplicaciones, es decir, determinan cómo es que estas últimas y los dispositivos se conectan y comunican entre sí (RedHat, 2020).

10.12 PHP (acrónimo recursivo de Hypertext Preprocessor)

Lenguaje de código abierto usado para el desarrollo web y que puede ser incrustado en HTML (PHP Net, 2022).

Fue creado por Rasmus Lerdorf en 1994 en el lenguaje de programación C con el fin de monitorear su currículum en línea e información personal por lo cual las siglas PHP hacían referencia al nombre de “Personal Home Page”. Con el paso de los años se reconstruyó el núcleo de PHP obtener lo que tenemos hoy en día (Flórez Fernández & Hernández Rodríguez, 2021).

Entre sus principales características destacan:

- a. Facilidad de aprendizaje.
- b. Aporta un análisis correcto de los requisitos.
- c. El modelo de base de datos lógica permite el rápido desarrollo.
- d. Soporte para distintas bases de datos.

Dentro de PHP el uso de un framework resulta esencial si se requiere que la aplicación web sea eficiente, confiable, con capacidad de mantenimiento y escalabilidad (Bautista Villegas, 2021). Existe una gran variedad frameworks por lo cual se debe de elegir aquel acorde a las necesidades del proyecto, algunos de ellos son:

- a. Laravel
- b. Sympony
- c. CodeIgniter
- d. Zend Framework
- e. CakePHP

Para el desarrollo de la presente se optó por Laravel el cual permite crear aplicaciones web estables y seguras.

10.13 Laravel

Es un framework PHP el cual fue desarrollado por Taylor Otwell, cuenta con una sintaxis expresiva y promete el desarrollo ágil de aplicaciones web ya que utiliza el patrón de diseño MVC (Ovando Ortega, 2019).

Gracias a que cuenta con componentes de seguridad, generación y validación de formularios, acceso a la base de datos y enrutamiento, permite un desarrollo rápido y con menos empleo de recursos lo cual es ideal para proyectos web a gran escala (Bautista Villegas, 2021).

Entre sus principales características destacan:

- a. Buena curva de aprendizaje.
- b. Documentación completa y simple.
- c. Tiene una amplia comunidad de desarrolladores.
- d. Cuenta con servicios y herramientas de infraestructura para facilitar su funcionamiento en distintos entornos.
- e. Ofrece versiones LTS (Long Term Support).
- f. Gracias a que utiliza el sistema de plantillas con un sistema caché permite mejorar el performance de los sitios (Cíceri Vazquez, 2018).

Para corroborar el funcionamiento del BackEnd antes de su integración con el FrontEnd es necesario utilizar herramientas como Postman

10.14 Postman

Es una herramienta que permite realizar pruebas a las API's ya que es un cliente HTTP que, por medio de una interfaz gráfica de usuario, se testean las peticiones HTTP y se obtiene respuesta de ello. Este ofrece los métodos de GET, POST,PUT, PATCH y DELETE para interactuar con los endpoints (Romero, 2021).

10.15 Base de datos

Una base de datos es un conjunto de información o datos estructurados los cuales son comúnmente almacenados de manera electrónica en sistemas informáticos comúnmente controladas por un Sistema Gestor de Base de Datos (Oracle, 2022).

10.16 Sistema Gestor de Base de Datos (SGBD)

Un SGBD es una colección de datos interrelacionados a los cuales se acceden a través de un conjunto de programas con el principal objetivo de proporcionar una forma de almacenar y recuperar información de una base de datos de manera práctica y eficiente.

Dichos sistemas son diseñados para gestionar grandes cantidades de información, la gestión implica la definición de estructuras para almacenar la información y mecanismos para manipularla (Silberschatz et al., 2002).

Además, los SGBD proporcionan abstracción de información, independencia, redundancia mínima, consistencia, seguridad, integridad, respaldo y recuperación, control de la concurrencia y tiempo de respuesta considerable (Universidad Tecnológica de Puebla, 2012).

Existen distintos SGBD en la actualidad y unos de ellos son MySQL, Oracle, Postgres, MySQL Server, entre otros. Sin embargo, el elegido para el propósito de abordar la base de datos en esta ocasión es MySQL.

10.17 MySQL

Según Casillas Santillán et al. (2014) es un sistema gestor de base de datos (SGBD) el cual es popular y muy usado por su simplicidad y notable rendimiento. Carece de características avanzadas a comparación de otros SGB, sin embargo, es una opción atractiva para aplicaciones comerciales y de entretenimiento debido a su facilidad de uso

y rápida puesta en marcha, adicionalmente cuenta con una libre distribución en Internet bajo la licencia GPL la cual garantiza un alto grado de estabilidad y rápido desarrollo.

MySQL tiene como principal objetivo ser una base de datos fiable y eficiente.

Otras de sus características son:

- a. Está disponible para múltiples plataformas.
- b. La herramienta mysql-client permite interactuar con un servidor MySQL (local o remoto) en modo texto.
- c. Está desarrollado en C/C++.
- d. Optimizado para equipos con múltiples procesadores.
- e. Se puede usar como cliente-servidor o incrustado en aplicaciones.
- f. Soporte para múltiples métodos de almacenamiento de tablas que cuentan con prestaciones y rendimiento diferentes.
- g. Administración basada en usuarios y privilegios.
- h. Altamente estable.

A las bases de datos se puede aplicar la técnica de mapeo objeto-relacional la cual consiste en convertir los datos entre el sistema de tipos con lenguaje de programación orientado a objetos y la propia utilización de la base de datos (Mozún Villamayor, 2018).

10.18 Servidor Web Apache

Según Márquez Díaz et al. (2002) es un “Servidor Web gratuito desarrollado por Apache Server Project con el objetivo de ofrecer un servidor web fiable, eficiente y extensible con un código fuente abierto gratuito”.

Es una máquina que ejecuta el programa llamado daemon http, httpd, este último recibe peticiones de un cliente web. Para las configuraciones de red, el servidor Apache debe

de iniciarse con permisos de root además de enlazarse al puerto 80 para escuchar peticiones y aceptar conexiones.

Entre sus ventajas destacan la licencia de código abierto, amplia comunidad de desarrolladores, arquitectura modular, es robusto y seguro además de que trabaja con versiones de Unix y Linux, Windows y Be Os (Márquez Díaz et al., 2002).

10.19 ORM (Object Relational Mapping)

Modelo de programación que permite mapear las estructuras de una base de datos relacional como SQL Server, Oracle, MySQL, entre otras, sobre una estructura lógica de entidades con el objetivo de simplificar y acelerar el desarrollo de aplicaciones.

Las estructuras de la base de datos relacional quedan vinculadas con las entidades lógicas o base de datos virtual definida en el ORM, de tal modo que las acciones CRUD (Create, Read, Update, Delete) a ejecutar sobre la base de datos física se realizan de forma indirecta por medio del ORM.

Los ORMs ayudan a los desarrolladores a omitir el uso de código SQL (Structured Query Language) para la realización de consultas y gestionar la persistencia de datos. Así, los objetos o entidades de la base de datos virtual creada en el ORM podrán ser manipulados por medio de algún lenguaje según el tipo de ORM utilizado (Deloitte, 2022).

Adicionalmente las estructuras ORM reducen la brecha entre la programación orientada a objetos y las bases de datos relacionales al convertir datos en objetos de programación apropiados y viceversa (Barsoti & Gibertoni, 2020).

Existen distintos ORM que se utilizan de acuerdo al lenguaje de programación o framework que se esté utilizando para la programación del BackEnd, algunos de ellos son Active Record para el lenguaje Ruby, Peewee para Python, Entity Framework para C# y Eloquent el cual será utilizado en el desarrollo de la presente aplicación ya que trabaja con PHP.

10.20 Eloquent

Es un ORM que facilita el contacto de la aplicación programada con Laravel y una base de datos relacional, para que dicha acción se lleve a cabo es necesario ejecutar un llamado al modelo de la entidad y posteriormente actuar sobre ella desde el controlador (A. B. Martínez, 2022), es decir, cada tabla tiene un modelo que permite que se interactúe con la misma ofreciendo acciones de inserción, actualización y eliminación de registros (García Lillo, 2022).

CAPÍTULO 4: DESARROLLO

A continuación, se muestra el procedimiento y descripción de las actividades que se vieron involucradas durante el desarrollo, siguiendo la metodología planteada y respetando los objetivos del proyecto.

Durante el desarrollo se identificó al tiempo como una limitante para terminar el proyecto en el framework Express ya que era poco conocido y el desarrollo de validaciones, modelos y controladores requerían de más tiempo para su ejecución, por tal motivo se optó por cambiar a Laravel ya que gracias a su extensa documentación permitió agilizar los procesos de desarrollo.

11. Procedimiento y descripción de las actividades realizadas.

La ejecución de la aplicación web se guió por una metodología ágil de desarrollo de software la cual lleva por nombre Prototyping o, en español, metodología por prototipo.

Tal como se puede apreciar en Figura 5, consta de diferentes fases como definición de los objetivos del prototipo, validación de los requerimientos a través de la funcionalidad del producto, diseño y desarrollo del prototipo y por último la evaluación del funcionamiento del mismo.



Figura 5. Ciclo de metodología por prototipo, Fuente: Elaboración propia,2022.

El ciclo de la Figura 5 se repite cuantas veces sea necesario hasta obtener el resultado deseado.

La elección de la presente se basó en la iteración, fácil detección de errores, detección de funcionalidades faltantes, fácil reacomodo de los requisitos e inclusión del cliente. Además de que aporta beneficios tales como la retroalimentación constante, desarrollo controlado, pruebas y reconstrucción del prototipo según las necesidades del cliente.

11.1 Definición de los objetivos del prototipo

La primera etapa consistió en definir los objetivos del proyecto de la mano del cliente. A través de entrevistas por medios virtuales tales como Google Meet, se estableció la problemática a atacar con el proyecto, el alcance del mismo, las tecnologías y herramientas a usar con el fin de formalizar las actividades de diseño y desarrollo que llevaran al correcto despliegue del proyecto.

La entrevista consistió en las preguntas que son mostradas a continuación:

- a. ¿Cuál es la problemática?
- b. ¿Cómo se resuelve actualmente?
- c. ¿Cuál se pretende que sea el resultado?
- d. ¿Cuáles son los requerimientos del servidor?
- e. ¿Tienen una idea acerca del acomodo de los componentes?
- f. ¿Cuántos tipos de usuario tendrá el sistema?
- g. ¿Tendrá un apartado de login?
- h. ¿Qué campos son necesarios para el registro?
- i. ¿Requiere de una barra de navegación?
- j. ¿Contendrá alertas?
- k. ¿Se puede utilizar un ORM (Object Relational Mapping o Mapeo Objeto-Relacional) para la interacción con los modelos y bases de datos?

11.2 Validación de los requerimientos a través de la funcionalidad del producto

Una vez conocidos los objetivos del proyecto se definieron los requerimientos funcionales (plasmados en las Tablas 1,2,3,4,5,6,7,8 y 9) y los no funcionales:

Funcionales:

Tabla 1. RF1

ID del requerimiento	RF1
Función	El usuario creará una cuenta en el sistema.
Descripción	El usuario proporcionará los respectivos datos para la creación de una cuenta de acceso al sistema.
Prioridad	Alta
Frecuencia	Alta

Tabla 2. RF2

ID del requerimiento	RF2
Función	El usuario deberá ingresar al sistema.
Descripción	Al acceder a la página principal el usuario podrá visualizar la opción de ingresar al sistema siempre y cuando se haya registrado previamente para obtener una cuenta. Tendrá que completar los campos con los respectivos datos y si estos son correctos podrá acceder al sistema.
Prioridad	Alta
Frecuencia	Muy alta

Tabla 3. RF3

ID del requerimiento	RF3
Función	Página de inicio con menú.
Descripción	Al acceder de manera correcta se le presentará al usuario una página de inicio la cual contendrá las revisiones de los artículos que este posee, así como un menú con las opciones de generar una nueva revisión, salir y la propia para volver a la página de inicio.
Prioridad	Alta
Frecuencia	Muy alta

Tabla 4. RF4

ID del requerimiento	RF4
Función	Opción de agregar una nueva revisión

Descripción	El usuario podrá agregar una nueva revisión a través de un formulario donde proporcionará los datos del artículo, tales como nombre, autores, nombre de la revista, fecha límite de revisión y la carga de los archivos necesarios.
Prioridad	Alta
Frecuencia	Alta

Tabla 5. RF5

ID del requerimiento	RF5
Función	Selección de archivos para revisión.
Descripción	Una vez cargados los archivos, el usuario podrá seleccionar el elemento PDF que desea revisar.
Prioridad	Muy alta
Frecuencia	Muy alta

Tabla 6. RF6

ID del requerimiento	RF6
Función	Revisión de artículo.
Descripción	Al seleccionar una revisión de un artículo se le mostrará al usuario una ventana en la cual se visualizará el PDF de dicho artículo además de la opción para agregar los comentarios y generar los reportes.
Prioridad	Muy alta
Frecuencia	Muy alta

Tabla 7. RF7

ID del requerimiento	RF7
Función	El usuario agregará los comentarios
Descripción	Se le mostrará al usuario un catálogo de comentarios predeterminados los cuales podrá elegir, sin embargo, se tendrá la opción de generar nuevos a través de un formulario.
Prioridad	Muy alta
Frecuencia	Muy alta

Tabla 8. RF8

ID del requerimiento	RF8
Función	El usuario generará un reporte con los comentarios recopilados.

Descripción	Una vez terminada la revisión el usuario podrá generar un archivo de texto o PDF con los comentarios colocados en el artículo.
Prioridad	Muy alta
Frecuencia	Alta

Tabla 9. RF9

ID del requerimiento	RF9
Función	El usuario tendrá la opción de salir del sistema.
Descripción	El usuario tendrá la posibilidad de cerrar sesión en el sistema.
Prioridad	Alta
Frecuencia	Alta

No funcionales:

- a. Se programará el BackEnd en el lenguaje de PHP en su versión 7.4 junto con el Framework de Laravel en su versión 8.83.
- b. Se utilizará la librería de React Js en su versión 18.2.0 para la programación FrontEnd.
- c. Se usará MySQL como sistema gestor de base de datos de la mano de Xampp.

Así mismo se plantearon los requerimientos y configuraciones del servidor donde se encontrará alojada la aplicación web, dicho servidor será Ubuntu 20 con 2 sockets, 2 cores, 8GB de memoria RAM y 50 GB de disco duro.

11.3 Diseñar y desarrollar el prototipo

Se comenzó con el diseño de la arquitectura con el fin de plantear la relación entre los componentes de la misma, tal como se muestra en la Figura 6 estos actúan como un todo desde sus respectivas funciones.

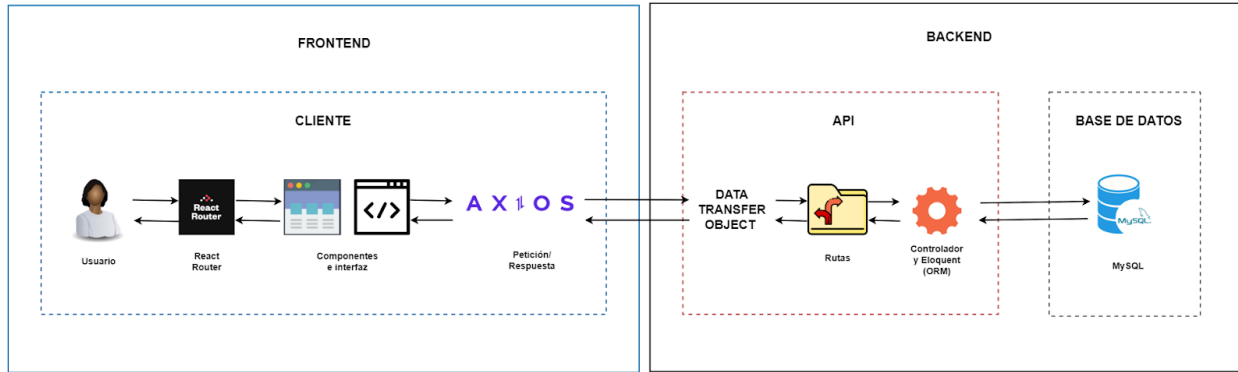


Figura 6. Arquitectura de la aplicación. Fuente: Elaboración propia

La Figura 7 representa de forma general el flujo entre los componentes de la arquitectura presentada en la Figura 6.

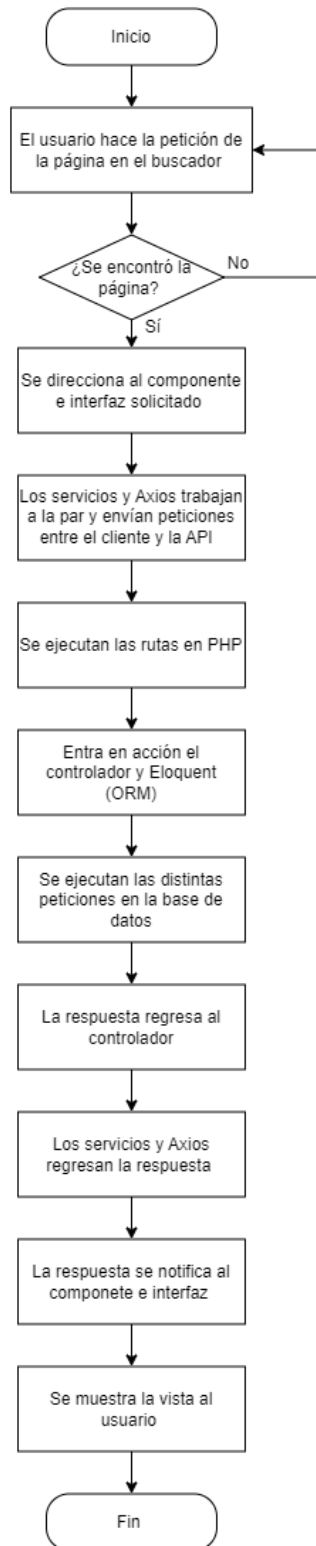


Figura 7. Diagrama de flujo de la arquitectura de la aplicación

La Figura 8 indica los casos de uso por parte del usuario, es decir, las acciones que podrá realizar el usuario dentro de la aplicación web.

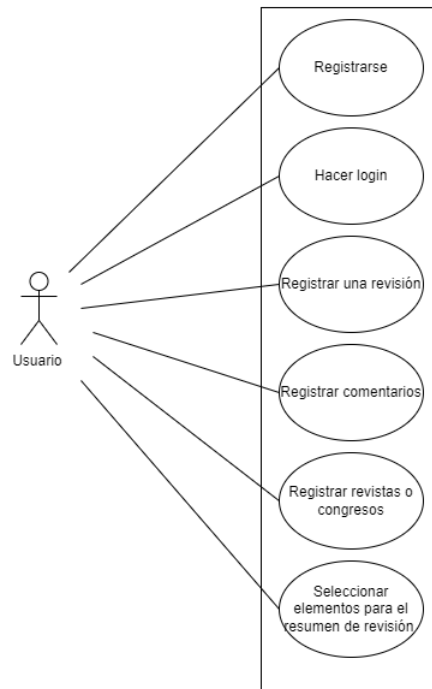


Figura 8. Diagrama de casos de uso

La Figura 9 representa el orden de las actividades a realizar por parte del usuario para acceder a la página web, las cuales a su vez se ligan con los diagramas de flujo de las Figuras 10, 11 y 12:

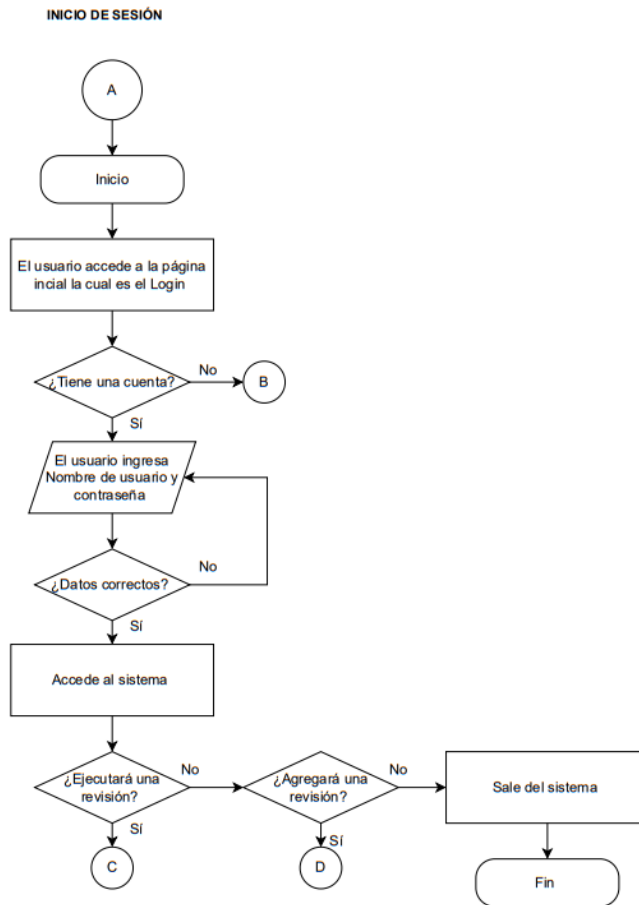


Figura 9. Diagrama de flujo para Login.

Por su parte en la Figura 10 se muestran los pasos para registrar un nuevo usuario en la aplicación web.

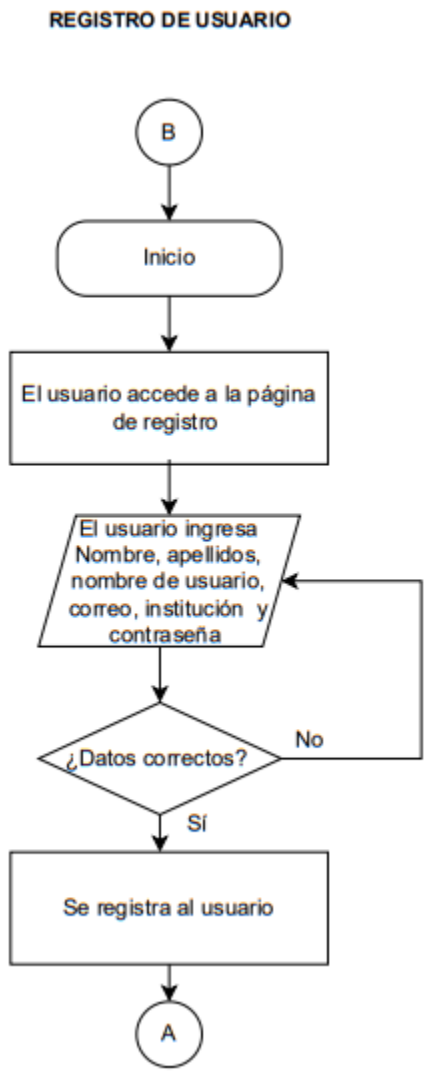


Figura 10. Diagrama de flujo para registrar usuario.

La secuencia de pasos para ejecutar una revisión por parte del usuario se puede apreciar en la Figura 11:

EJECUTAR UNA REVISIÓN

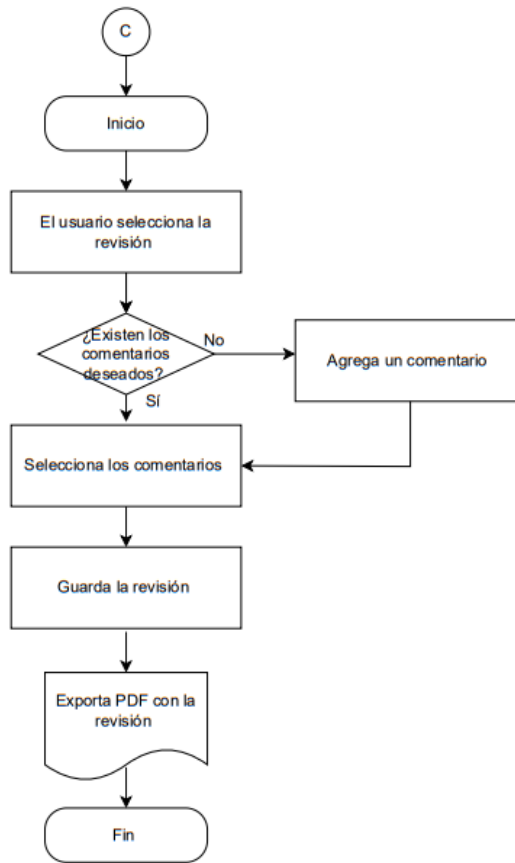


Figura 11. Diagrama de flujo para ejecutar una revisión.

Así mismo en la Figura 12 se plasmó el procedimiento para agregar una revisión la cual continua con el del flujo de actividades de la Figura 11:

AGREGAR UNA REVISIÓN

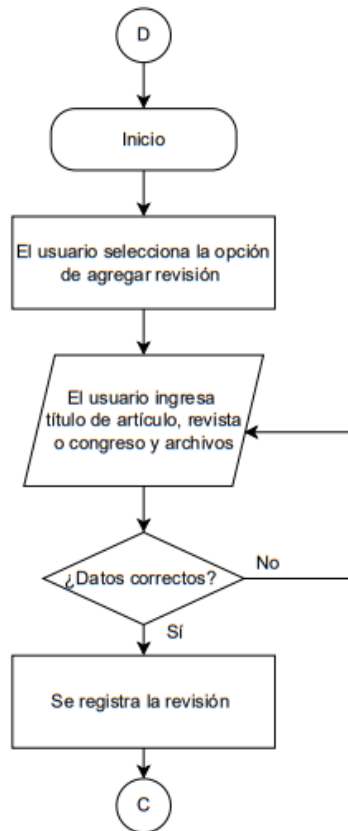


Figura 12. Diagrama de flujo para agregar una revisión.

También se identificaron los datos necesarios para la elaboración de la base de datos y se plasmaron en el diagrama físico (Figura 13) de manera que el planteamiento y programación resultara más simple. Dichos diseños se realizaron en el software de Diagrams.net el cual es un software de dibujo de gráficos multiplataforma gratuito y de código abierto desarrollado en HTML5 y Javascript.

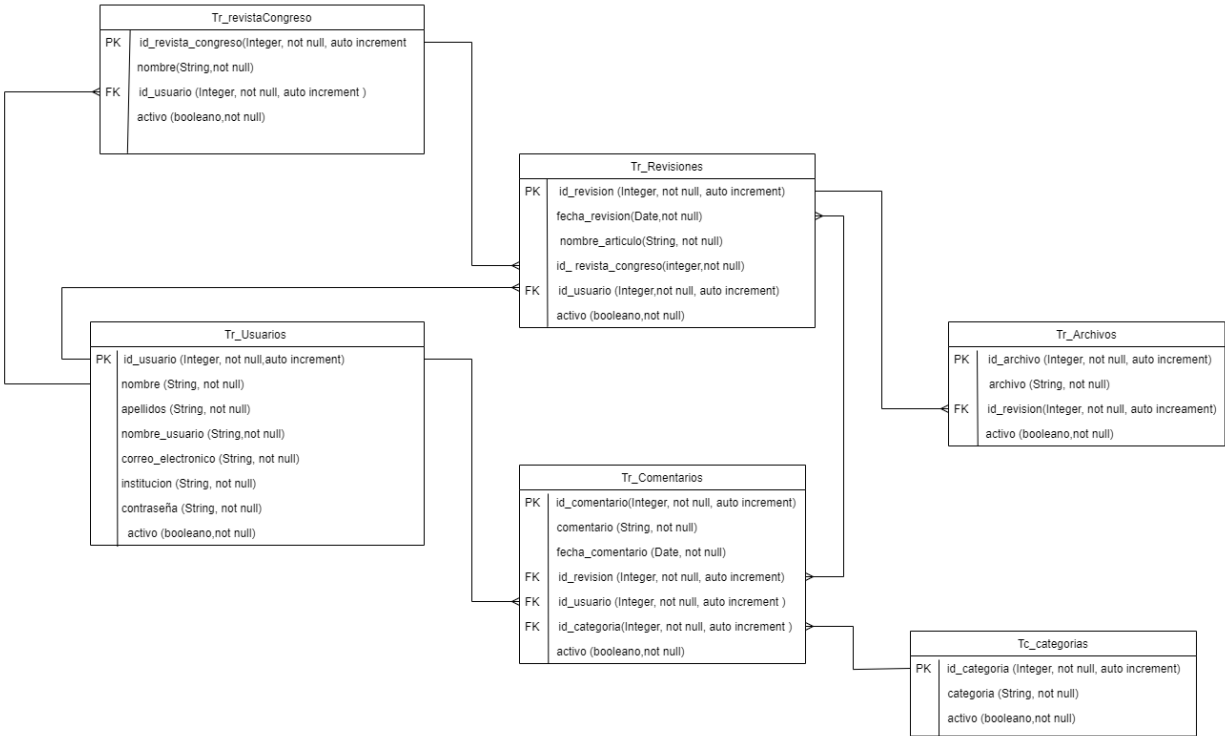


Figura 13. Diagrama físico de la base de datos

Se diseñaron los bocetos de las vistas para que fuera sencillo visualizar la posición de cada componente y acción de la aplicación web los cuales se muestran en la Figura 14:



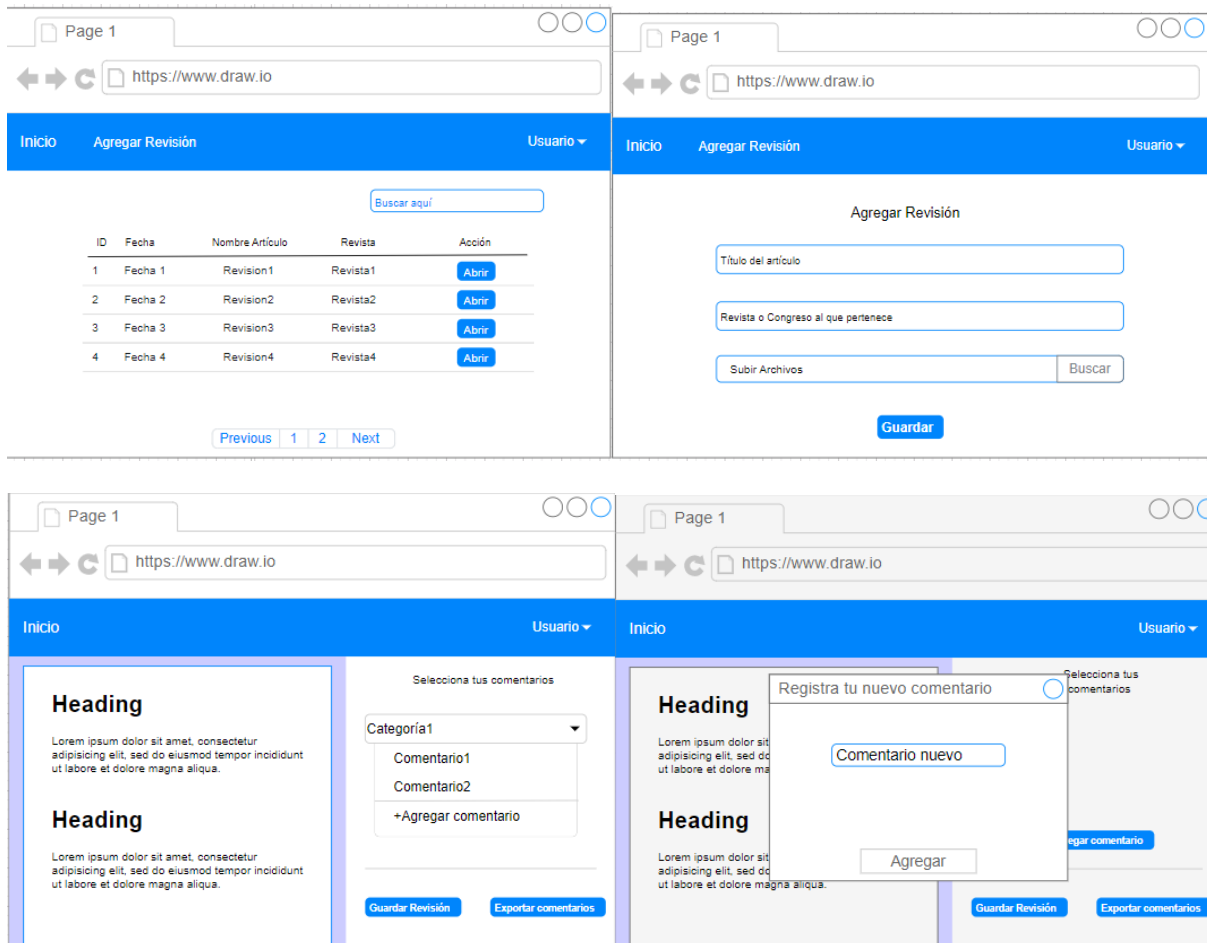


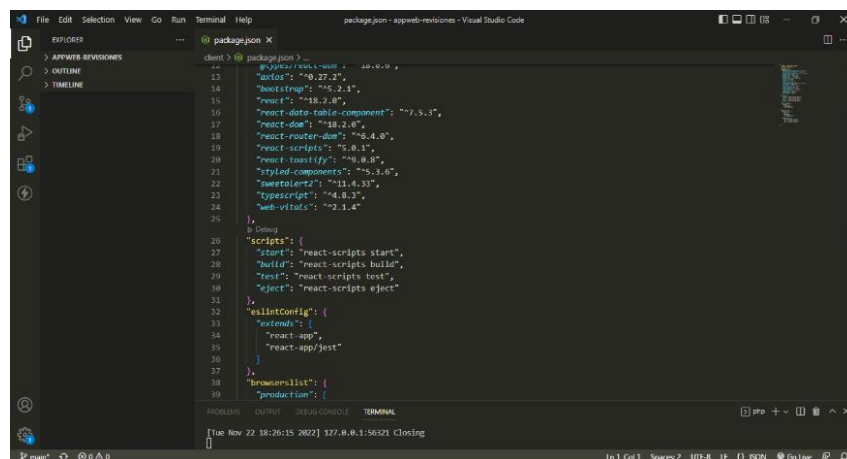
Figura 14. Bocetos de la aplicación web

Para el desarrollo de la aplicación se llevó a cabo la preparación del entorno, se instalaron las tecnologías y herramientas mencionadas en los requisitos no funcionales, tal como se puede apreciar en las Figuras 15,16 y 17.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

[Tue Nov 22 18:26:15 2022] 127.0.0.1:56321 Closing
PS C:\appweb-revisiones\api> php artisan --version
Laravel Framework 8.83.25
PS C:\appweb-revisiones\api> █
```

Figura 15. Versión instalada de Laravel.



```
package.json
13  "react": "18.2.0",
14  "react-dom": "18.2.0",
15  "react-router-dom": "6.4.0",
16  "react-scripts": "5.0.1",
17  "react-test-renderer": "18.2.0",
18  "styled-components": "5.3.0",
19  "sweetalert2": "11.4.33",
20  "typescript": "4.8.3",
21  "web-vitals": "2.1.4",
22  },
23  "scripts": {
24    "start": "react-scripts start",
25    "build": "react-scripts build",
26    "test": "react-scripts test",
27    "eject": "react-scripts eject"
28  },
29  "eslintConfig": {
30    "extends": [
31      "react-app",
32      "react-app/jest"
33    ]
34  },
35  "browserslist": {
36    "production": [
37      >>>
```

Figura 16. Versión instalada de React Js

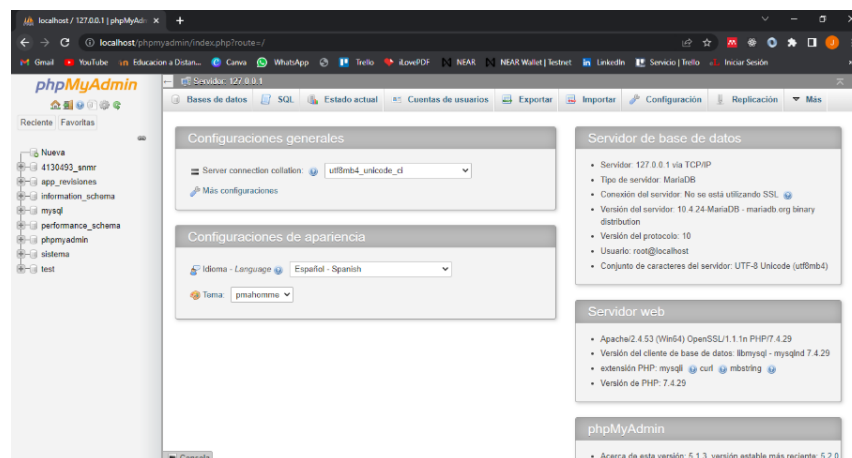


Figura 17. Sistema Gestor de Base de Datos

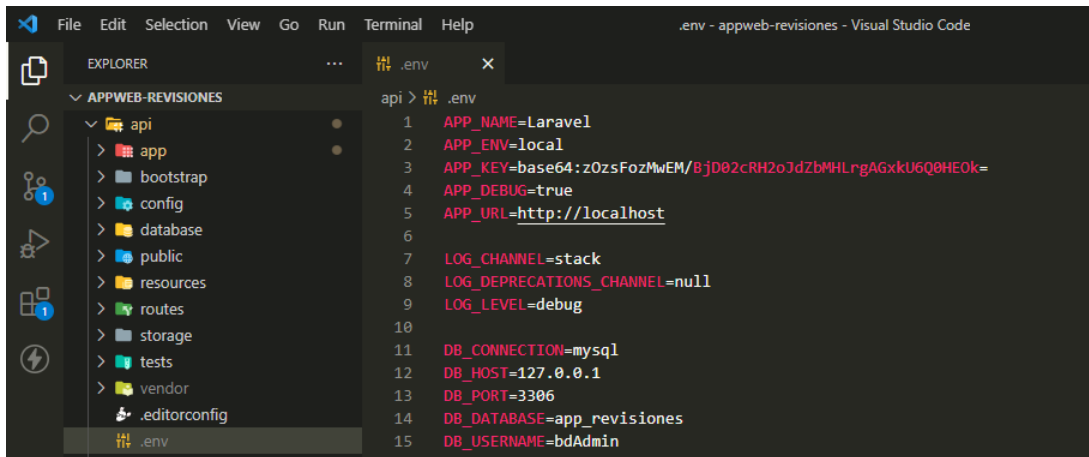


Figura 19. Edición del archivo .env para conexión a base de datos.

Para crear las tablas, campos de las mismas y sus relaciones en la base de datos se usaron migraciones de Laravel las cuales permiten gestionar la creación y cambios que se van realizando sobre las mismas.

Una vez creadas las tablas se programó la semilla para la tabla “Tc_Categorias”, tal como se muestra en la Figura 20, la cual permite inicializar la tabla con datos.

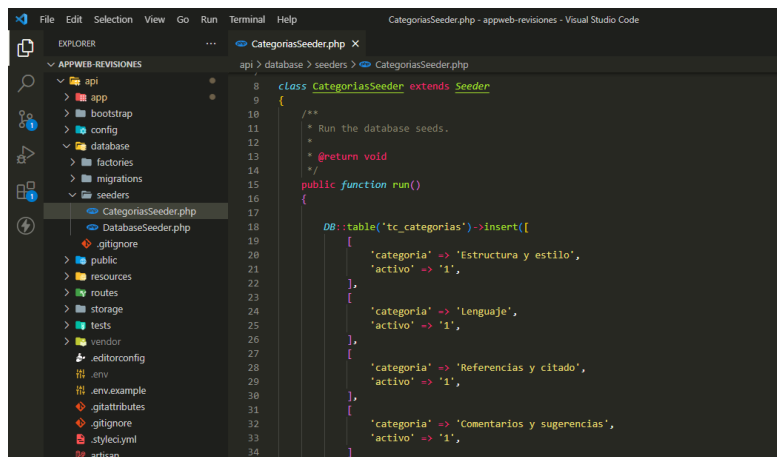


Figura 20. Programación de la semilla de la tabla Tc_Categorias.

Una vez configurado lo anterior se procedió a crear los modelos y los controladores con el fin de seguir el MVC. Los modelos incluyen las tablas a las que van conectados

permitiendo realizar acciones sobre las mismas a través de la creación de los controladores que es donde se incluye toda la lógica de programación de la aplicación.

Así mismo dentro de cada controlador se incluyeron las validaciones de tipos de datos, estructura y longitud permitidos para que se realizara la correcta manipulación de los mismos en el modelo por parte del usuario.

También se llevó a cabo la instalación de la librería FPDF ya que esta serviría para la estructuración del formato que incluye los comentarios y el resumen de la revisión.

11.4 Evaluar el funcionamiento del prototipo

Como parte de la evaluación del prototipo se pasó a probar el FrontEnd dando como resultado el correcto paso de parámetros a los componentes ya que estos son utilizados en distintas vistas cada uno con las propiedades que se le ingresaban.

Por su parte el BackEnd se probó con la aplicación de Postman en la que se realizaron las peticiones a la API y así corroborar su correcto funcionamiento. Las realizadas con POST abarcaron el login, registro de usuario, añadir comentario, registrar revisión y agregar una revista o congreso. Por su parte en las peticiones GET se encuentra el mostrar revisiones, comentarios y revistas o congresos por id de usuario.

En la Figura 21 se pueden apreciar un ejemplo de la petición POST para insertar un nuevo usuario en la base de datos, la cual a su vez resulta en una inserción exitosa.

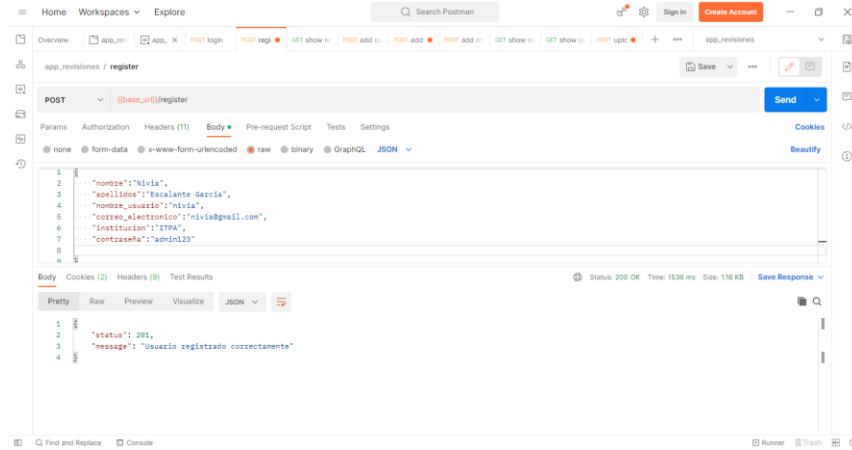


Figura 21. Inserción de un usuario a la base de datos.

Por su parte una petición GET trae todos los datos que sean solicitados, por ejemplo, en la Figura 22 se puede apreciar cómo se obtienen las revistas o congresos con los que cuenta un usuario.

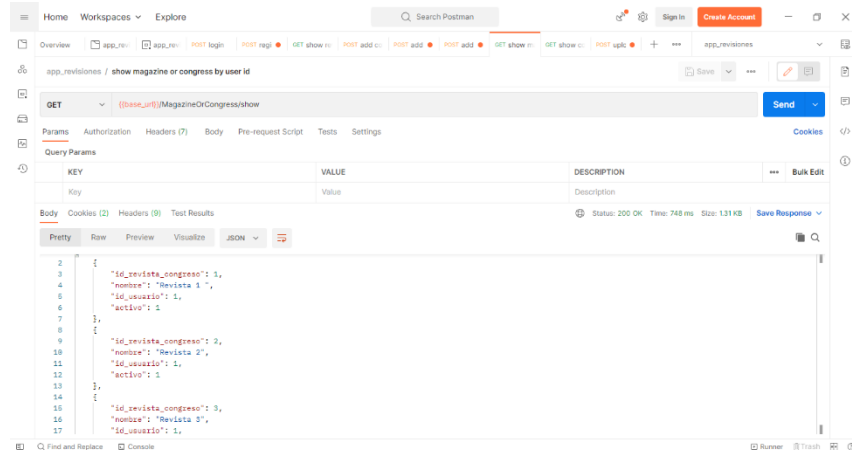


Figura 22. Revistas con las que cuenta un usuario.

CAPÍTULO 5: RESULTADOS

Los resultados son una parte fundamental en cada proyecto ya que reflejan las consecuencias de las actividades realizadas. Por tal motivo en el presente capítulo se plasman los hallazgos y efectos derivados del desarrollo.

Como primer resultado se obtuvo que, gracias a la creación de migraciones en laravel, se generó la base de datos (Figura 23) sobre la cual se pueden ejecutar acciones de control y manipulación de los datos.

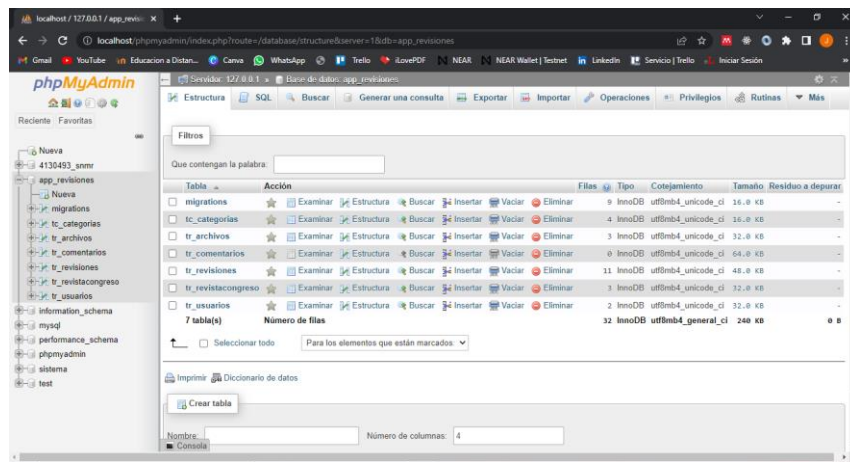
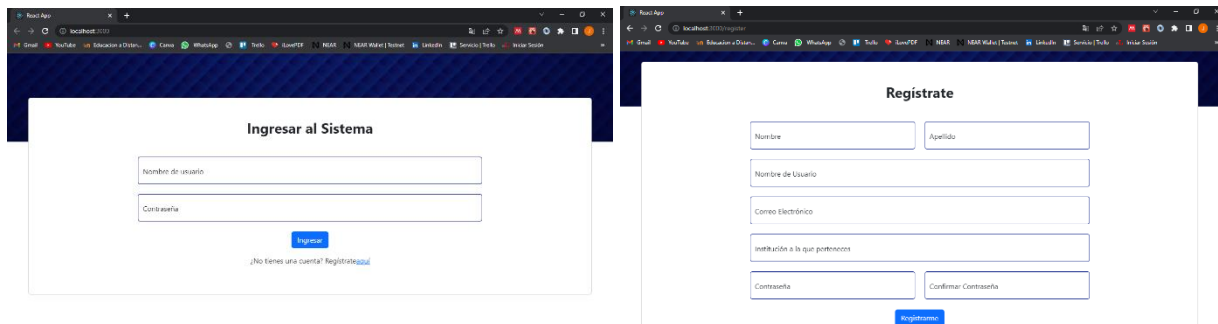


Figura 23. Base de datos de la aplicación

Tal como se puede visualizar en la Figura 24, se desarrolló la parte de FrontEnd siguiendo los principios de crear componentes reutilizables que crean vistas más elaboradas. También se basó en la responsividad de la página ya que esta se adapta según el dispositivo en el que se accede.



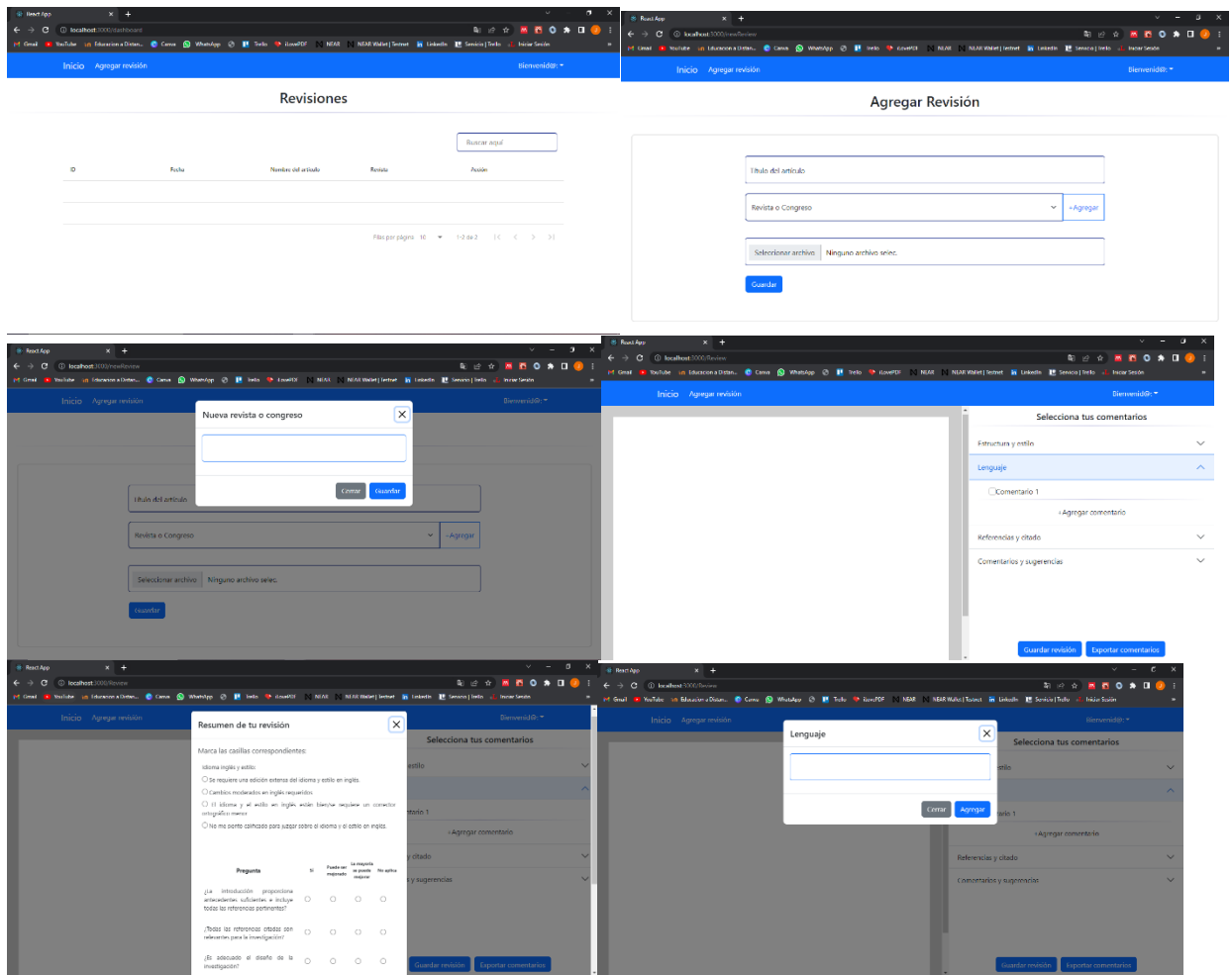


Figura 24. Vistas de la aplicación

Otro resultado fundamental fue la creación del BackEnd en el cual destacan las funciones de login, registrar un usuario, registrar y mostrar revisiones por usuario, añadir y mostrar comentarios además de registrar y visualizar las revistas o congresos por el usuario, dichos avances se pueden observar en el apartado “api” en el repositorio de GitHub: <https://github.com/jaqueline16/appweb-revisiones>

Tal como se mencionó en el capítulo 4, se realizó un cambio de tecnología para el desarrollo del BackEnd. Dicho cambio involucró más tiempo del esperado ya que la curva de aprendizaje del nuevo framework se llevó a un ritmo moderado, por lo cual no se logró completar la integración del FrontEnd con el BackEnd además de la realización del módulo de análisis y estadísticas de los datos de los comentarios de revisores. Sin embargo, lo desarrollado en el BackEnd cuenta con un patrón de diseño MVC que

permite separar las funciones, mejorar la percepción de los errores, garantizar la actualización y el mantenimiento del modelo, es decir, que la aplicación sea escalable. Así mismo las acciones cuentan con distintos filtros de seguridad en la validación de campos para garantizar la correcta inserción y manipulación de la información en la base de datos.

CAPÍTULO 6: CONCLUSIONES

En conclusión, la implementación de las aplicaciones web dentro del ámbito de investigación permite la inclusión de las tecnologías de la información y comunicaciones en procesos importantes tales como la revisión por pares de los artículos científicos, dando como beneficios la buena organización por parte de los revisores así como la optimización de tiempos en realizarlas.

Los módulos del FrontEnd y BackEnd generados permiten llevar el control de las revisiones de cada usuario, gestionando los comentarios con los que éste cuenta, las revistas o congresos que registra, así como seleccionar los elementos necesarios para el resumen de su revisión.

CAPÍTULO 7: COMPETENCIAS DESARROLLADAS

La elaboración de un proyecto conlleva asumir retos y adquirir formación sobre los tópicos relacionados para que la ejecución del mismo se lleve a cabo con éxito. Así mismo el proyecto permite medir el aporte que se tuvo sobre la persona encargada de efectuarlo. Es por eso que en el siguiente apartado se enlistan el conjunto de conocimientos y habilidades que se adquirieron durante la elaboración del proyecto:

- a. Apliqué habilidades en la toma de decisiones y el trabajo en equipo para cumplir con las actividades asignadas.
- b. Desarrollé habilidades para programar tiempos de entrega de actividades en un entorno empresarial con el fin de cumplir en tiempo y forma.
- c. Interpreté los requisitos funcionales y no funcionales en base a las necesidades del cliente para dar una solución apropiada de desarrollo.
- d. Apliqué el patrón de diseño de software MVC lo cual me permitió separar las funciones del código, reducir el esfuerzo en la programación, mejorar la percepción de los errores, garantizar la actualización y el mantenimiento del software mediante la escalabilidad del modelo.
- e. Aprendí a usar manejadores de paquetes para instalar las dependencias o librerías necesarias a lo largo del proyecto.
- f. Usé un framework de desarrollo de BackEnd con el fin de agilizar el desarrollo y fomentar la seguridad de la aplicación.
- g. Separé los componentes del FrontEnd dándoles un formato y estilo con el fin de reutilizar los mismos en distintas vistas, agilizando de esta manera el desarrollo.
- h. Apliqué interfaces en los componentes del FrontEnd para pasar parámetros y propiedades de programación.
- i. Utilicé la plataforma de Postman para testear las funcionalidades del BackEnd desarrolladas.

CAPÍTULO 8: FUENTES DE INFORMACIÓN

- Adobe. (2021). *¿Qué son las aplicaciones web y las páginas web dinámicas?*
<https://helpx.adobe.com/mx/dreamweaver/using/web-applications.html>
- Anchundia Medrano, L. A. (2022). *ANÁLISIS COMPARATIVO DE TECNOLOGÍAS FRONT
END ANGULAR JS VS REACT JS, EN EL MODELO DE PROCESOS PARA EL
DESARROLLO DE APLICACIONES WEB.*
- Anturi Figueroa, A. I. (2020). *USO PRÁCTICO DE ENCABEZADOS HTTP PARA MEJORAR
LA SEGURIDAD EN EL PROTOCOLO.*
<https://repositorio.uniandes.edu.co/bitstream/handle/1992/51490/23852.pdf?sequence=1&isAllowed=y>
- Axios. (2022). *Primeros pasos | Documentos de Axios.* <https://axios-http.com/docs/intro>
- Baiget, T. (2020). *Manual SCImago de revistas científicas. Creación, gestión y publicación.*
http://eprints.rclis.org/43532/1/Manual_definitivo.pdf
- Barsoti, N., & Gibertoni, D. (2020). *IMPACT THAT SEQUELIZEBRINGS TO THE
DEVELOPMENT OF ANAPIBUILT IN NODE.JSWITH EXPRESS.JS.*
<https://revista.fatectq.edu.br/interfacetecnologica/article/view/964/537>
- Bautista Villegas, E. (2021). *Metodologías ágiles XP y Scrum, empleadas para el desarrollo de
páginas web, bajo MVC, con lenguaje PHP y framework Laravel.*
<https://revistas.unamad.edu.pe/index.php/rad/article/view/168/275>
- Carbonel Martínez, A. (2020). *Diseño e implementación de un middleware CoAP-MQTT-HTTP
para la mejora de la interoperabilidad de los protocolos de aplicación en redes IoT.*
<https://zagan.unizar.es/record/96261/files/TAZ-TFM-2020-1104.pdf>
- Casillas Santillán, L. A., Gibert Ginesta, M., & Pérez Mora, Ó. (2014). *Bases de datos en
MySQL.* https://d1wqtxts1xzle7.cloudfront.net/54167148/MYSQL_MANUAL-with-cover-page-v2.pdf?Expires=1662929612&Signature=D99LBFiaQcgjmjjjHnMPsbvj~XMKsA5nFRGFwg dAo7ivZL31~hihu5UcMoOo1CdlfXqPM2JiyG03H19DBohXN55izCmkWNYVOPolgtMTMlv3 FilKrUw5Ai0RPfioXQ7fQTjfdjppg8xX-lbLHS7zuNKpOxstjocyfgKXQ4VWzHEESuFbMkQJpUi~DLPTuSsl4A8S6SbISrZXOCbDQ Rm-6dr10f2jGyJBwYpldqmUMRM~aKOrtT5fti2p5u~8rpSJYImVV-jU-Zeb8a485BiorNrYTG-yEN4wLJwrQxyzug2LMKbqNZhb9OSO-6oL7f7JbxFpV~IRHx2qWTOf3~k9ng__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA
- Cíceri Vazquez, M. J. N. (2018). *Introducción a Laravel: Aplicaciones robustas y a gran escala .*
<https://books.google.es/books?hl=es&lr=&id=sPyIDwAAQBAJ&oi=fnd&pg=PA4&dq=laravel &ots=mYwFtx9f9d&sig=GTpgYxWIM6MkRFIIZOY8A170rVw#v=onepage&q=laravel&f=false>
- Deloitte. (2022). *¿Qué es un ORM?*
<https://www2.deloitte.com/es/es/pages/technology/articles/que-es-orm.html>
- Flórez Fernández, H., & Hernández Rodríguez, J. (2021). *Aplicaciones web con PHP .*
<https://books.google.es/books?hl=es&lr=&id=Zb0xEAAAQBAJ&oi=fnd&pg=PR14&dq=que +es+el+lenguaje+php&ots=ctWNaaY6js&sig=8bxMATHSUG-wWDggX3ckHM4s99Y#v=onepage&q=que es el lenguaje php&f=false>
- García Islas, L. ., Franco Sánchez, K. ., Samperio Monroy, T. ., Pérez Tavera, I. ., & Ocampo López, A. (2021). *C-ToolP: Herramienta de apoyo para la enseñanza de desarrollo Web-SPA-MVC.* <https://repository.uaeh.edu.mx/revistas/index.php/icbi/article/view/7470/8282>
- García Lillo, J. (2022). *Aplicación web para comportar coche .*
https://rua.ua.es/dspace/bitstream/10045/125654/1/Aplicacion_web_para_compartir_coche _Garcia_Lillo_Javier.pdf

- Hernández Villena, A. (2018, December 12). *El uso de React Js para el desarrollo de sistemas web*. <http://repositorio.upsin.edu.mx/Fragmentos/tesinas/tesina48121.pdf>
- Ladrón de Guevara Cervera, M., Hincapié, J., Jackman, J., & Caballero Uribe, C. V. (2008). Revisión por pares: ¿Qué es y para qué sirve? . *Salud Uninorte*, 24(2). <http://www.redalyc.org/articulo.oa?id=81722411>
- Márquez Díaz, J., Sampedro, L., & Vargas, F. (2002). *Instalación y configuración de Apache, un servidor Web gratis*. <https://www.redalyc.org/pdf/852/85201202.pdf>
- Martínez, A. B. (2022). *¿Qué es un ORM y para qué sirve Eloquent? - Platzi*. <https://platzi.com/clases/1920-eloquent-laravel/28515-que-es-un-orm-y-para-que-sirve-eloquent/>
- Martínez, G. S. (2012). La revisión por pares y la selección de artículos para publicación. *Revista Colombiana de Psicología*, 21(1), 27–35.
- Mozún Villamayor, R. (2018). *Sistema de negocio de cambios de grupo*.
- Oracle. (2022). *Qué es una base de datos | Oracle México*. <https://www.oracle.com/mx/database/what-is-database/>
- Orozco Aguirre, H. R. (2019). *MODELO VISTA CONTROLADOR (MVC) Y WEB ARCHIVES (WARS)*. http://ri.uaemex.mx/bitstream/handle/20.500.11799/108712/secme-35486_1.pdf?sequence=1
- Ovando Ortega, D. J. (2019). *Bootstrap y Laravel, herramientas para el desarrollo de aplicaciones web*.
- Pérez Ibarra, S. G., Quispe, J. R., Mullicundo, F. F., & Lamas, D. A. (2021). *HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO WEB DESDE EL FRONTEND AL BACKEND*. <http://sedici.unlp.edu.ar/handle/10915/120476>
- PHP Net. (2022). *PHP: ¿Qué es PHP? - Manual*. <https://www.php.net/manual/es/intro-what-is.php>
- React Js Org. (2022). *React – Una biblioteca de JavaScript para construir interfaces de usuario*. <https://es.reactjs.org/>
- RedHat. (2020). *¿Qué es una API de REST?* <https://www.redhat.com/es/topics/api/what-is-a-rest-api>
- Robaina Castellanos, G. R., & Semper González, A. I. (2019). *Ética de la revisión por pares en publicaciones científicas*. <http://scielo.sld.cu/pdf/rme/v41n6/1684-1824-rme-41-06-1533.pdf>
- Romero, G. (2021). *Cómo realizar pruebas automatizadas con Postman*. <https://www.encora.com/es/blog/como-realizar-pruebas-automatizadas-con-postman>
- Shahu Gaikwad, S., & Adkar, P. (2019). *A Review Paper on Bootstrap Framework*. https://www.think247.com/vertical?s_pt=sou
- Silberschatz, A., Korth, F., H., & Sudarshan. (2002). *Fundamentos de bases de datos*.
- Universidad Tecnológica de Puebla. (2012). *Manual de Asignatura Basado en Competencias Profesionales*. https://d1wqtxts1xzle7.cloudfront.net/36735110/base_de_datos_1-with-cover-page-v2.pdf?Expires=1663000350&Signature=cev2okJJNLMMYfqLdoD-fZ6bSUiD47EqbOcm5Vud8sk~f0BvDvSftNysFI82BCZTuq4gwCvwzbtB9mf7W9lu-MemyDRiPOE2wNy41ES6M7swyhtPIp~klRmhtVvZ2F1JC3u2OxJqdN~RTyq0W1hllYcqAXryLS9Naa6dbA6Zfy959N3YhymRKGrcNBYd5nEAsjB-jgXWak1oMRRLVHJtbhVfpbruOyScGhBguRHvzost8TUzZJKCSiZrkJ6ihefklmr1WebdWvT0j~qEVjCS31K9VR9C5fPULopqbE0HJOciUEdPEZU7gr9TjiiB2WNvHZiLS1-0eued1zL5JvPw__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA#page=64

CAPÍTULO 9: ANEXOS



Instituto Tecnológico de Pabellón de Arteaga
Laboratorio de Iluminación Artificial

Pabellón de Arteaga, Ags., **02/agosto/2022**

Folio: LIA 045/2022

Asunto: Carta de aceptación

Dr. José Ernesto Olvera González
Director del Instituto Tecnológico
de Pabellón De Arteaga.

At'n: Dra. Julissa Elayne Cosme Castorena
Jefa Del Depto. Gestión Tecnológica Y Vinculación

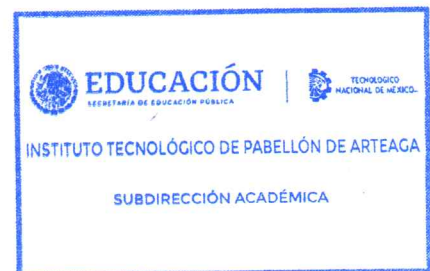
PRESENTE:

Por este conducto, me permito informarle que C. **Jaqueline Garcia Luevano** con número de control **181050155**, alumna de la carrera de: **Ing. En Tecnologías de la Información y Comunicaciones**, fue aceptada para realizar su Residencia profesional en el TecNM Campus Pabellón de Arteaga - Laboratorio de Iluminación Artificial (LIA), con el proyecto denominado "**Aplicación web para llevar el control de las revisiones de artículos científicos**" donde cubrirá un total de **500 horas, a partir del día 09 de agosto 2022.**

Sin otro particular por el momento, aprovecho la ocasión para enviarle un cordial saludo.

ATENTAMENTE

Dra. Nivia Iracemi Escalante García.
Coordinadora de LIA



Carretera a la Estación de Rincón Km 1, C.P. 20670
Pabellón de Arteaga, Aguascalientes
Tel. (465) 958-2482 y 958-2730, Ext. 119
e-mail: cyd_parteaga@tecnm.mx

