



EDUCACIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

Instituto Tecnológico de Pabellón de Arteaga
Departamento de Ciencias Básicas

PROYECTO DE TITULACIÓN

*SISTEMA PARA LLEVAR EL CONTROL DE ACTIVIDADES ACADÉMICAS DE
ALUMNOS DE TESIS, RESIDENCIAS Y SERVICIO SOCIAL*

PARA OBTENER EL TÍTULO DE

*INGENIERA EN TECNOLOGÍAS DE LA INFORMACIÓN Y
COMUNICACIONES*

PRESENTA:

ESTEFANÍA GALLEGOS SILVA

ASESOR:

EDWIN CELESTINO GARCÍA ALCOCER

Mayo



ÍNDICE

| | |
|---|----|
| PRELIMINARES | 3 |
| 1. <i>Agradecimientos.</i> | 3 |
| 2. <i>Resumen</i> | 4 |
| 3. <i>Lista de Ilustraciones.</i> | 5 |
| 4. <i>Lista de tablas.</i> | 6 |
| CAPÍTULO 1: GENERALIDADES DEL PROYECTO | 7 |
| 5. <i>Introducción</i> | 7 |
| 6. <i>Descripción de la empresa u organización y del puesto o área del trabajo del residente.</i> | 8 |
| 7. <i>Problemas a resolver, priorizándolos.</i> | 11 |
| 8. <i>Justificación</i> | 12 |
| 9. <i>Objetivos General y Específicos</i> | 13 |
| 9.1. <i>Objetivo General.</i> | 13 |
| 9.2. <i>Objetivos Específicos.</i> | 13 |
| CAPÍTULO 3: MARCO TEÓRICO..... | 14 |
| 10. <i>Marco Teórico</i> | 14 |
| 10.1 <i>Herramientas para el desarrollo Web</i> | 14 |
| 10.1.1 <i>HTML (Hypertext Markup Language).</i> | 14 |
| 10.1.2 <i>CSS (Cascading Style Sheets).</i> | 14 |
| 10.1.3 <i>Javascript.</i> | 15 |
| 10.1.4 <i>React.</i> | 15 |
| 10.2 <i>Backend.</i> | 16 |
| 10.2.1 <i>PHP (Hypertext Preprocessor).</i> | 16 |
| 10.2.2 <i>Python.</i> | 16 |
| 10.2.3 <i>Node.</i> | 17 |
| 10.3 <i>Bases de datos.</i> | 17 |
| 10.3.1 <i>PostgreSQL.</i> | 18 |
| 10.3.2 <i>MySQL.</i> | 18 |
| 10.4 <i>Framework.</i> | 18 |
| 10.4.1 <i>Angular.</i> | 19 |
| 10.4.2 <i>ASP.Net.</i> | 19 |
| 10.4.3 <i>Laravel.</i> | 19 |
| 10.4.4 <i>Express.js.</i> | 20 |

| | |
|--|----|
| 10.4.5 <i>Peticiones http</i> | 20 |
| CAPÍTULO 4: DESARROLLO | 21 |
| 11. <i>Procedimiento y descripción de las actividades realizadas</i> | 21 |
| CAPÍTULO 5: RESULTADOS | 29 |
| 12. <i>Resultados</i> | 29 |
| CAPÍTULO 6: CONCLUSIONES..... | 33 |
| 13. <i>Conclusiones del Proyecto</i> | 33 |
| CAPÍTULO 7: COMPETENCIAS DESARROLLADAS..... | 34 |
| 14. <i>Competencias desarrolladas y/o aplicadas</i> | 34 |
| CAPÍTULO 8: FUENTES DE INFORMACIÓN..... | 35 |
| 15. <i>Fuentes de información</i> | 35 |
| CAPÍTULO 9. ANEXOS | 36 |

PRELIMINARES

1.Agradecimientos.

El proceso para llegar hasta este punto ha sido largo y nada sencillo, pero hoy estoy donde tanto anhele estar, gracias a todas las personas que han estado junto a mí en este largo camino, hoy quiero agradecer principalmente a mis padres por el esfuerzo que han hecho diariamente para que nada me falte, para que pueda lograr todos cada uno de mis objetivos, gracias por la paciencia, la ayuda, pero sobre todo la fe y las palabras de aliento y motivación, esté logro va por ustedes.

Gracias a mis amigos de la carrera, que más que amigos se han convertido en una familia, gracias por el apoyo, hoy puedo decir con orgullo que la carrera me dejó muchas enseñanzas y la amistad de grandes ingenieros.

2.Resumen

Sistema para llevar el control de actividades académicas de alumnos de tesis, residencias y servicio social, nace de la necesidad de implementar una mejor organización en instituciones públicas, de tener un mejor control de las documentaciones de relevancia para ciertas áreas educativas, este sistema se desarrollará en base a tecnologías reconocidas y sobre todo tecnologías que se acoplan a las necesidades que presentan.

El sistema requiere de 2 clientes (usuario y administrador) que son quienes podrán tener un acceso controlado y los cuales pueden contar con ciertos privilegios que los diferencia uno de otro, cada perfil tiene apartados distintos dentro de los cuales se pueden llevar a cabo actividades que reflejaran el objetivo funcional de dicho sistema.

3.Lista de Ilustraciones.

| | |
|---|----|
| Ilustración 1.Ubicación ITPA | 8 |
| Ilustración 2.Organigrama | 11 |
| Ilustración 3.Modelo cliente-servidor | 18 |
| Ilustración 4.Peticiones HTTP | 20 |
| Ilustración 5.Verisón de React | 21 |
| Ilustración 6.Versión de Laravel | 21 |
| Ilustración 7.Version de Php MyAdmin..... | 21 |
| Ilustración 8.Ciclo de vida de un prototipo..... | 22 |
| Ilustración 9.Diagramas de Casos de uso | 24 |
| Ilustración 10.Vista Login | 24 |
| Ilustración 11.Vista registro | 24 |
| Ilustración 12.Vista cargar documento y estatus | 25 |
| Ilustración 13.Vista del archivo y el avance del usuario..... | 25 |
| Ilustración 14.Estructura de carpetas | 25 |
| Ilustración 15.Componentes..... | 26 |
| Ilustración 16.Modelo físico de base de datos..... | 26 |
| Ilustración 17.Seeds | 27 |
| Ilustración 18.Pruebas de la BD | 27 |
| Ilustración 19.BD..... | 28 |
| Ilustración 20.Vista página inicial | 29 |
| Ilustración 21.Vista Login | 29 |
| Ilustración 22.Vista subir archivo..... | 30 |
| Ilustración 23.Vista selección de carreras | 30 |
| Ilustración 24.Vista de registro | 30 |
| Ilustración 25.BD..... | 31 |
| Ilustración 26.Representacion del funcionamiento de semillas..... | 31 |
| Ilustración 27.estatus de los documentos Ilustración 28.Vista de estatus | 32 |
| Ilustración 29.Vista de grafica de avances | 32 |

4. Lista de tablas

Tabla:1 Requerimientos23

CAPÍTULO 1: GENERALIDADES DEL PROYECTO

Cada proyecto está conformado por aspectos que lo diferencian de otros, características que lo describen de una manera general y que una vez plasmados permiten despertar el interés en el usuario por conocer acerca del proyecto. En este capítulo describe la organización y estructura que lo conforman, los alcances y los resultados a los que se estima se llegará en dicho proyecto.

5.Introducción

La implementación de la creación de un sistema que monitoree el avance de las actividades escolares dentro de las instituciones, nace de la necesidad de tener una mayor organización dentro de los avances de las actividades que se desarrollan en las distintas áreas de las instituciones.

Han existido sistemas con características similares al sistema que desea generarse, entre ellas sistemas como:

- Jira Software
- Trello
- Planner

Estas aplicaciones permiten administrar planes de trabajo, de forma que facilita la asignación de actividades de manera organizada, además de que es posible asignar una fecha de vencimiento y el responsable que lleva acabo cada actividad.

Estas herramientas permiten gestionar el flujo de trabajo y la supervisión de las tareas asignadas, permite añadir archivos, comentarios y añadir automatizaciones.

De tales herramientas nace la idea de un sistema que pueda realizar tareas similares a las que ya existen y tenga un plus que dé lugar a la adaptación de a las necesidades de un mercado diferente.

Para que el sistema cumpla con las necesidades y funcionalidades establecidas por el usuario es necesario que cuente con la autenticación para alumnos y responsables de

los proyectos, que permitirá tener el control de los usuarios que tienen acceso a la información.

El sistema podrá llevar a cabo su función de una manera satisfactoria una vez que se tenga el registro de los proyectos y datos del protocolo a los que se desea darles seguimiento, además debe permitir subir archivos que completen el proceso administrativo y permitir el registro de actividades individuales por proyecto, para que de esa forma la visualización de avance general en conjunto con las fechas de inicio y fin den lugar a la generación de los reportes de actividades en tiempo y forma.

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

Las residencias profesionales fueron realizadas en el Laboratorio de Iluminación Artificial (LIA) dentro del Tecnológico Nacional de México campus Pabellón de Arteaga fundado el 01 de septiembre de 2008, su ubicación es Carretera a la Estación de Rincón km 1, C.P. 20670, Pabellón de Arteaga, del estado de Aguascalientes.



Ilustración 1.Ubicación ITPA Fuente:(Google maps, 2022)

LIA es un centro capaz de impulsar, desarrollar y fortalecer iniciativas que incentiven la productividad y rentabilidad agroalimentaria, generando en el sector agroindustrial

nuevas filosofías de competitividad y calidad. Las herramientas tecnológicas encontradas en LIA permiten que el laboratorio tenga impacto a nivel internacional. Se cuentan con cuatro proyectos de investigación y desarrollo tecnológico enfocados a la línea de investigación de Biosistemas Mecatrónicas y Ambientes Artificiales. Su objetivo general se centra en el profundizar y clarificar los fenómenos no explicados hasta el momento sobre los efectos de la luz (continua y pulsada) en plantas y alimentos como su caracterización biológica con herramientas tecnológicas actuales LEDs, siendo un centro capaz de poseer el nivel de competitividad que otros laboratorios existentes en el mundo.

Como misión se ha propuesto 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 con productos agroalimentarios con valor agregado.

Principales actividades de la empresa u organización: Las herramientas tecnológicas encontradas en LIA permiten que el laboratorio tenga impacto a nivel internacional. Se cuentan con cuatro proyectos de investigación y desarrollo tecnológico enfocados a la línea de investigación de Biosistemas Mecatrónicas y Ambientes Artificiales.

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).

Desarrollo de Sistemas de desinfección de alimentos con radiación ultravioleta LED tipo A, B y C aplicados a productos agroindustriales y en fresco.

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.

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.

El equipamiento e infraestructura con la que cuenta LIA se despliega a continuación:

- a) 3 cámaras de crecimiento (descripción Temperatura 12C a 30C.
- b) Iluminación del área de trabajo. Humedad: Ambiente +5 % a 90 %).
- c) Programación día/noche de temperatura.
- d) Racks y accesorios para producción de cultivos comerciales, sistema de caracterización de la luz (reflexión, absorción y transmisión) que incluye dos espectrofotómetros Ocean, esfera de medición de esparcimiento de luz e integradora (0.5 a 1 m) con accesorios para la manipulación de objetos en el interior.
- e) Un espectro radiómetro ILT950, fluorómetro WALZ, SPAD 502, sensores de radiación fotosintéticamente activa, sensores climatológicos (humedad, temperatura, CO₂ y radiación), Controladores Lógicos Programables, Sistema de radiación artificial tipo LED RGBW (rojo, azul, verde y blanco) de 0 a 1000 μ mol de intensidad, con canal programable independiente para el color.
- f) Frecuencia de pulsado de 1 a 500 kHz. Con software y control independiente. Sistemas de radiación artificial equipados con UV-LED e Infrarrojo.

El equipo de LIA está conformado por 2 coordinadores, un grupo de investigadores, un equipo de doctorado y maestría, tesis y auxiliares presentes en el organigrama de la empresa (Ilustración 2), que hacen posible llevar a cabo cada uno de los proyectos y actividades que se tienen.

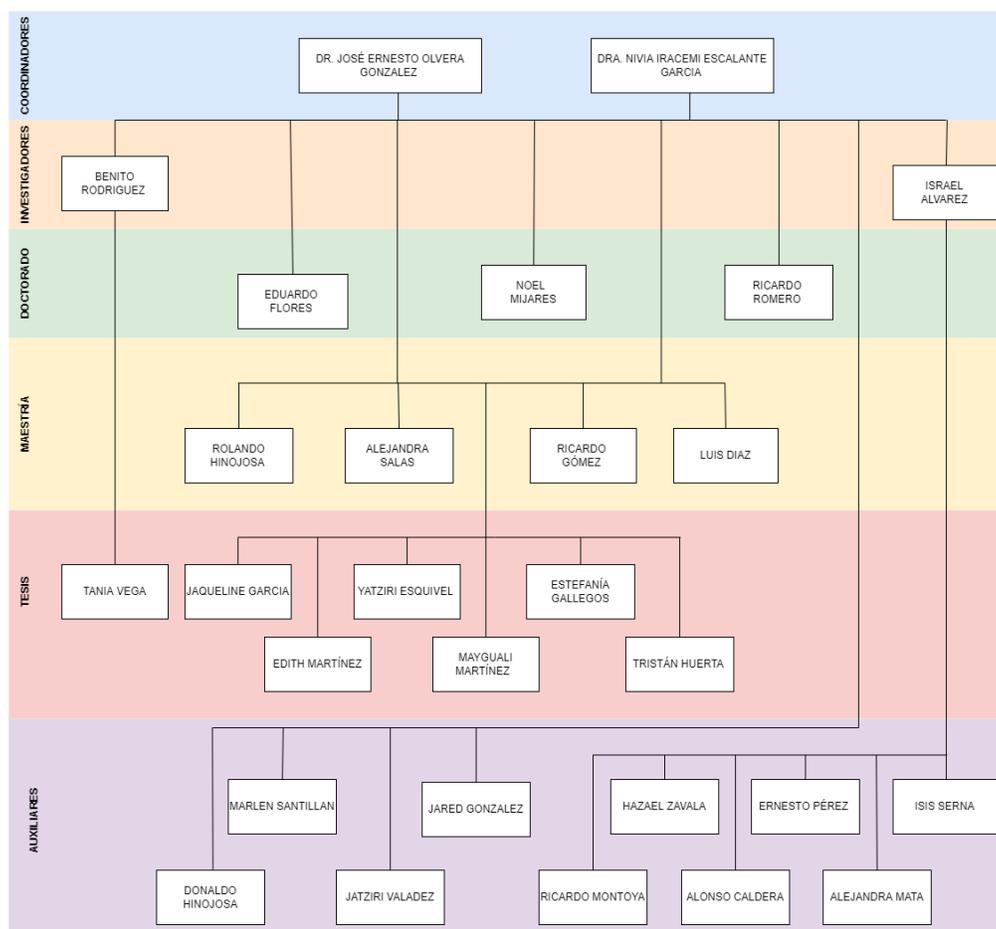


Ilustración 2. Organigrama Fuente: LIA

7. Problemas a resolver, priorizándolos.

Actualmente las instituciones Educativas se enfrentan al manejo de una gran cantidad de procesos y solicitudes educativos en las distintas áreas del plan educativo. Todos son procesos completamente diferentes, y cada uno de ellos requiere de un manejo distinto, es por ello que se requiere de herramientas informáticas que agilicen la resolución de las revisiones de los documentos y avance automatizado de cada proceso.

Este sistema permitirá agilizar la revisión de avances con respecto a proyectos y actividades relevantes dentro de áreas académicas, como servicio social y residencias, su implementación permitirá que cada actividad sea monitoreada supervisada y efectuada de modo que conlleve a un resultado satisfactorio. Su impacto en instituciones académicas, facilitará el trabajo de los asesores de actividades académicas de tal forma que podrán darle un mejor seguimiento, revisión y retroalimentación de forma individual a los distintos proyectos que se tienen registrados. Permitirá alertar acerca de los problemas y dificultades que se presenten en cada proyecto y que afecten al avance del mismo, además de que el supervisor podrá conocer cada una de las actividades que se han asignado dentro de un tiempo establecido, de tal forma que podrá saber detalladamente el avance se tiene del proyecto en general.

8. Justificación

Determinar los distintos recursos de las diversas actividades que integran un proyecto y contar con una herramienta o proceso que permita el monitoreo supervisado y ejecutado para la obtención de resultados satisfactorios; agilizando la revisión de contribuciones planeadas o programadas a cada proyecto y/o actividades urgentes o relevantes dentro de áreas académicas. Su implementación contribuye a facilitar el trabajo incluso de docentes para emitir un mejor seguimiento, revisión y retroalimentación de forma individual y precisa en el manejo de los diferentes proyectos en los que se encuentra colaborando. Además, concede la manera de puntuar el problema y el tiempo que impiden el avance de cada proyecto. Al imponer este tipo de sistemas se integran los conocimientos y habilidades adquiridos en la carrera de TICs (Tecnologías de la información y comunicaciones) dentro de las cuales se destaca la programación con calidad, la administración de proyectos, diseño de base de datos y pruebas de software, las cuales colaboran en conjunto para conseguir el objetivo planteado.

Además de los conocimientos aplicados en TIC's se aplican conocimientos de la gestión de la información, así como del uso de la gestión tecnológica ERP, esta gestión permite controlar, planificar, organizar todas las tareas y agilizar la toma de decisiones.

Esta innovación permitirá agilizar la revisión de avances con respecto a proyectos y actividades relevantes dentro de determinadas áreas. Dará lugar a una mejora dentro de la calidad de los procesos de modo que conlleve a un resultado productivo.

Permitirá resaltar los problemas y dificultades que se presenten en cada proyecto y que afecten al avance del mismo, además de conocer cada una de las actividades que se han asignado dentro de un tiempo establecido.

El proyecto dará lugar a la implementación de herramientas totalmente nuevas para el alumno. Se desarrollará el sistema antes mencionado con las características que han sido establecidas por el usuario.

El residente se desempeñará dentro del área de tecnologías, área dentro de la cual desarrollará principalmente sistemas, programación, administración de proyectos, diseño de base de datos, pruebas de software que ayuden a agilizar las tareas relevantes que se llevan a cabo dentro del laboratorio.

9. Objetivos General y Específicos

9.1. Objetivo General.

Desarrollar una herramienta web que colabore al seguimiento en las actividades académicas (tareas, integración de recursos humanos, servicio social, residencias, tesis) planteadas y/o realizadas a través de forma gráfica como cronogramas o listado para su posterior evaluación.

9.2. Objetivos Específicos.

- Observar el proceso actual de seguimiento en actividades académicas iniciando en el Laboratorio de Iluminación Artificial.
- Determinar los requerimientos generales (usuario, departamento, estudiante, entre otros.) para el diseño de la aplicación.
- Generar diagramas de diseño con diferentes arquitecturas de software para validar los procedimientos involucrados.
- Disponer de lenguajes de programación y tecnologías como Node JS para Backend, React para Frontend y MySQL para la primera versión.
- Evaluar y retroalimenta los errores que se presenten tanto en Backend y Frontend.

CAPÍTULO 3: MARCO TEÓRICO

10.Marco Teórico

La construcción de un sistema de información está conformada con la parte visual e interactiva que concede al usuario manejar los diferentes apartados o acciones para ejecutar sus procedimientos, con el objetivo de que el sistema sea amigable y fácil de utilizar.

Los aspectos mencionados llevan a fundamentar diversos conceptos que permiten la generación de la pantalla al usuario, así como el desarrollo de la parte interna del sistema. Considerando estas características, a continuación, se despliegan los fundamentos teóricos que necesarios para la creación del sistema de información.

10.1 Herramientas para el desarrollo Web

10.1.1 HTML (Hypertext Markup Language).

Es uno de los lenguajes más utilizados, facilita el uso y la edición de vídeo, imágenes y contenido de audio en páginas web, esté lenguajes es compatible y óptimo con los navegadores más utilizados (IONOS, 2022).

10.1.2 CSS (Cascading Style Sheets).

Utilizado principalmente por sus términos de estilo y diseño, el código puede realizar cambios de forma global (mdn, 2022a).

Esto permite que la página sea llamativa para el usuario y que le permita entender de manera sencilla su uso y su manejo.

Detrás del Sistema para llevar el control de actividades académicas existen herramientas de Frontend que se adecuan a los requerimientos y que permiten mostrarle al usuario una pantalla en la que sea sencillo llevar a cabo cada una de las funciones existentes para el usuario, existen diversos lenguajes dentro de los cuales se destacan, HTML (Hypertext Markup Language) el cual es uno de los lenguajes más utilizados, facilita el uso y la edición de vídeo, imágenes y contenido de audio en páginas web, esté lenguajes es compatible y óptimo con los navegadores más utilizados.

A pesar de ser uno de los lenguajes más utilizados, no significa que sea el mejor lenguaje que se pueda implementar en un Frontend, ya que para que el contenido desarrollado en este lenguaje se pueda mostrar de forma más atractiva es necesario confiar con otros lenguajes como lo son CSS (Cascading Style Sheets), este lenguaje es utilizado principalmente por sus términos de estilo y diseño, el código puede realizar cambios de forma global (Murcia, 2020).

Dentro de las características de contenido dinámico en el frontend, es necesario la implementación de lenguajes como JavaScript o PHP.

10.1.3 Javascript.

Es utilizado en páginas web para hacer interferencia a la eficacia de las acciones de la página, permite tener acciones dinámicas y gráficos animados lo que da lugar a la amplia funcionalidad de la página web, la interfaz y la experiencia con el usuario. Pero todas estas características no lo hacen el único lenguaje capaz de cumplir con los requerimientos necesarios.

Dentro de la lista de los lenguajes más aplicados y más adecuados para la implementación de un Frontend, destaca el lenguaje React, es identificado como una librería de JavaScript utilizada para el desarrollo de interfaces de usuario que necesita de otras librerías para completar las acciones y los requisitos dentro del desarrollo de la aplicación web, su funcionamiento se divide en 3 conceptos básicos, HTML, que hace referencia a la estructura e información de la página web; el esqueleto, CSS quien se encarga de la apariencia de la página web y Javascript quien determina básicamente la funcionalidad de la página (MDN Plus, 2022).

10.1.4 React.

Es de código integrado por componentes y abierto, por lo que permite ser reutilizado y anidado, dando lugar a la reducción del tiempo de desarrollo de aplicaciones web complejas.

Los componentes utilizados en React son piezas de User Interface, es decir cada pieza que se muestra en la pantalla y hace posible la interacción del usuario, estos deben de ser capaces de compartir, reutilizar y combinarse para formar mayores que darán lugar

a la creación de un sistema completo, son objetos de lógica y estructura contenidos en un objeto Javascript (Aplyca Tecnologías, 2022).

10.2 Backend.

Detrás del diseño y arquitectura de una página web se encuentra la parte lógica, o bien arquitectura interna donde se establece el funcionamiento de los elementos, componentes, funcionalidad, seguridad, acciones internas y optimización de los recursos necesarios de la página.

Para el correcto manejo de la programación de un Backend existen diversos lenguajes, cada uno con características que permiten diferenciar uno de otro, entre ellos.

10.2.1 PHP (Hypertext Preprocessor).

Se caracteriza principalmente por estar diseñado para ajustarse en HTML, al ejecutar un archivo PHP este llega en forma de HTML.

El lenguaje sirve para conectar ciertos elementos de una página, dando paso al cambio de contenido entre las peticiones de los usuarios. Tiene facilidad en la integración de la base de datos y ofrece una máxima seguridad, permite cargar los elementos de una página antes de que se muestre al usuario, lee los comandos y activa todos los elementos funcionales y la interfaz visual. Además de los elementos atractivos, la funcionalidad que estas llevan, PHP se caracteriza por tener un alto nivel de compatibilidad con las bases de datos entre ellas, Oracle, MySQL, Interbase y SQLite(Hostinger, 2022).

10.2.2 Python.

Un lenguaje multiplataforma como lo es Python, su objetivo es la optimización de procesos que dará lugar a la simplicidad de las tareas. Cuenta con una gran biblioteca que permite la reutilización de códigos, es compatible con otros lenguajes de programación (Java, C y C++) y es un lenguaje fácil de trasladar a diversos sistemas operativos (Windows, macOS, Linux y Unix). Dentro de sus características destaca el hecho de que es un lenguaje sencillo de leer y escribir, debido a su similitud con el lenguaje humano (crehana, 2022).

10.2.3 Node.

Gracias a un entorno multiplataforma como Node.js puede ejecutarse Java Script en cualquier plataforma móvil, escritorio o web. Es decir, pasar de ejecutarse solamente en un navegador, a algo que puede ejecutarse en los ordenadores como si se tratara de una aplicación independiente.

Node.js es utilizado para crear sitios dinámicos, simplifica la comunicación, su velocidad en la ejecución de código, es asíncrono y controlado, es decir las APIs de la biblioteca de Node.js son asíncronas. Su uso es principalmente para aplicaciones de transmisión de datos y aplicaciones intensivas de datos en tiempo real. Utiliza un modelo de entrada y salida ligero, eficiente y sin bloqueo, es capaz de manejar una gran cantidad de conexiones simultáneas con un alto nivel de rendimiento, opera un solo subproceso, lo que le permite soportar decenas de conexiones al mismo tiempo, se desarrolla en un entorno de tiempo de ejecución de fuente libre, manejo simultáneo de peticiones (DONGEE, 2022).

10.3 Bases de datos.

Para delimitar las responsabilidades y funcionalidades dentro de un proyecto, es necesario definir siempre la estructura dentro de una base de datos, una base de datos es una colección organizada de información o datos estructurados, que normalmente se almacena de forma electrónica en un sistema informático, en donde se incluyen datos relevantes, y se agilizan las actividades del proyecto en el que se desea trabajar o implementar esta herramienta.

Esta colección de información que es almacenada electrónicamente, está diseñada para además tener acceso a una búsqueda y recuperación eficiente.

Existe la posibilidad de actualizar, modificar, eliminar y añadir datos que contengan relación con el sistema, existen diversos gestores de bases de datos que permiten cumplir con las tareas necesarias el funcionamiento y la correcta interacción de un sistema con el usuario, entre las cuales se encuentra, Postgre SQL, Mongo DB, MySQL (Oracle, 2022).

10.3.1 PostgreSQL.

Gestiona y consulta bases de datos relacionales de gran tamaño, puede procesar numerosas solicitudes de manera simultánea lo que significa que es capaz de acelerar el trabajo de los programadores al realizar numerosas tareas.

Otra alternativa como base de datos puede ser Mongo DB, este gestor de bases de datos está orientado a documentos, en donde, todos los datos registrados son guardados en documentos, esta base de datos no necesita de tablas, es muy rápido y todos los datos son almacenados en un mismo sitio. Otra alternativa de gestión de bases de datos es MySQL (OVH, 2022).

10.3.2 MySQL.

Es un sistema relacional de código abierto, es decir que es libre de usarlo y modificarlo según sus necesidades y a través de un modelo cliente-servidor, su función principal es crear y administrar bases de datos basadas en un modelo relacional.

EL modelo cliente-servidor, dentro de las computadoras permite ejecutar el programa mediante el cual se manipularán las acciones necesarias, el dispositivo cliente se conecta mediante una red específica a un servidor y a través de esta conexión es posible realizar una solicitud de la cual ambas partes entenderán, tal como se muestra en la Ilustración 3 (Kinsta, 2022a).

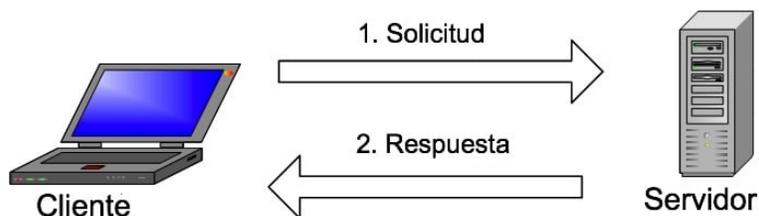


Ilustración 3. Modelo cliente-servidor Fuente: (MCS, 2022)

MySQL crea una base de datos que crea, manipula y almacena datos, dentro de esto los clientes pueden realizar solicitudes escribiendo sus instrucciones en MySQL. La información solicitada es procesada y aparecerá frente al cliente.

10.4 Framework.

Un conjunto de componentes que te ayudan a desarrollar sitios web más fácil y rápidamente son indispensables (rockocontent, 2022).

Para los programadores el framework se trata de una plantilla o estructura base que utilizan para elaborar o desarrollar una app, con objetivos específicos. Permiten acelerar el trabajo, reducir los errores y obtener resultados de mayor calidad, maneja la rapidez, la eficacia y un código limpio de la programación. Su implementación permite reutilizar el código cuantas veces sea necesario lo que hace que sea más sencillo encontrar errores, pero sobretodo evitar que estos se hagan presentes.

Dentro de los Frameworks más reconocidos y utilizados, se encuentran:

10.4.1 Angular.

Es un framework ideal para crear aplicaciones web a gran escala y de alto rendimiento, manteniéndolas fácilmente, Django facilita su trabajo al agrupar las diferentes funciones en una gran colección de módulos reutilizables, llamada marco de aplicación web. Los desarrolladores utilizan el marco web de Django para organizar y escribir su código de manera más eficiente y reducir significativamente el tiempo de desarrollo web (Hubspot, 2022).

10.4.2 ASP.Net.

Nos ayuda a construir aplicaciones potentes para PC, así como para dispositivos móviles. Es ligero y de alto rendimiento, ideal para crear aplicaciones web utilizando .Net. De gran potencia, productividad y velocidad (IEBS, 2022).

10.4.3 Laravel.

Es un framework PHP gratis y de código abierto que brinda un conjunto de herramientas y recursos para crear aplicaciones modernas, incluye herramientas que facilitan la construcción de aplicaciones web, haciendo de este proceso algo mucho más rápido y dando como resultado un código bien estructurado y fácil de mantener (talenty, 2022).

Proporciona la mejor contemplación de bases de datos, consulta y actualiza los datos sin esfuerzo, se combina perfectamente con:

- MySQL
- Postgres
- SQLite

- SQL Server.

Por otra parte, tenemos a Express.js.

10.4.4 Express.js.

Es un marco Node.js ligero, rápido y similar a Sinatra que proporciona funciones y herramientas potentes para crear aplicaciones del lado del servidor. Le proporciona un sistema fácil de navegar y funciones para expandir su infraestructura con potentes funciones y componentes que se adaptan a las condiciones de uso de su aplicación (mdn, 2022b).

10.5 Peticiones http.

Las peticiones HTTP ejemplifica la conexión a un servidor y solicita una aplicación específica o le envía datos. Es decir, un medio de comunicación entre servidores y clientes tal como se muestra en la Ilustración 4 (Kinsta, 2022b).

1. **GET.** Es el método de solicitud HTTP más utilizado. Una solicitud GET solicita información o recursos específicos de un servidor. Cuando visita un sitio web, su navegador envía muchas solicitudes GET para obtener la información necesaria para cargar la página.
2. **HEAD.** Con la solicitud HEAD, solo obtiene la información del encabezado de la página que desea cargar.
3. **POST.** Tu navegador utiliza el método de petición HTTP POST cuando necesita enviar datos al servidor.
4. **PUT.** Las peticiones PUT tienen una funcionalidad similar a la del método POST.

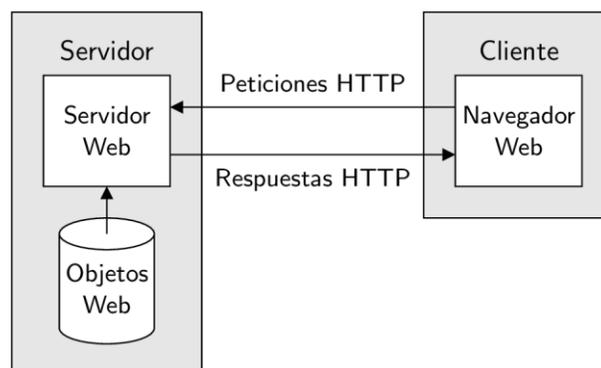


Ilustración 4. Peticiones HTTP Fuente: (UAL, 2022)

CAPÍTULO 4: DESARROLLO

En este capítulo abordará la descripción detallada de las actividades en las que se incluye información, procedimientos, técnicas y los tiempos que se han implementado para que el proyecto avance en tiempo y forma para un posterior resultado.

11.Procedimiento y descripción de las actividades realizadas.

Preparación del entorno

```
sistemaavances > package.json > ...
6   "@testing-library/jest-dom": "^5.16.5",
7   "@testing-library/react": "^13.4.0",
8   "@testing-library/user-event": "^13.5.0",
9   "@types/jest": "^27.5.2",
10  "@types/node": "^16.11.66",
11  "@types/react": "^18.0.21",
12  "@types/react-dom": "^18.0.6",
13  "bootstrap": "^5.2.2",
14  "react": "^18.2.0",
15  "react-dom": "^18.2.0",
16  "react-router-dom": "^6.4.2",
17  "react-scripts": "5.0.1",
18  "typescript": "^4.8.4",
19  "web-vitals": "^2.1.4"
20 },
21 > Debug
22 "scripts": {
23   "start": "react-scripts start",
24   "build": "react-scripts build",
25   "test": "react-scripts test",
26   "eject": "react-scripts eject"
27 },
28 "eslintConfig": {
29   "extends": [
30     "react-app",
31     "react-app/jest"
32   ]
33 },
34 "browserslist": {
35   "production": [
36     ">0.2%",
37     "not dead",
38     "not op_mini all"
39   ]
40 }
41 }
```

Ilustración 5. Versión de React Fuente: Creación Propia

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Appsisistemacontroldeactividades> cd appi
PS C:\Appsisistemacontroldeactividades\appi> php artisan --version
Laravel Framework 9.35.1
PS C:\Appsisistemacontroldeactividades\appi> []
```

Ilustración 6. Versión de Laravel Fuente: Creación Propia

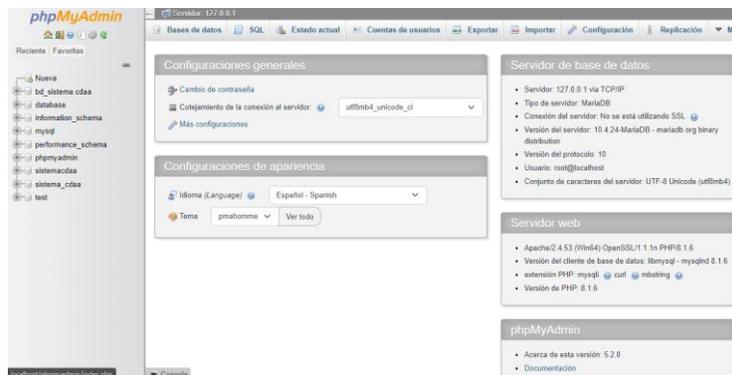


Ilustración 7. Versión de Php MyAdmin Fuente: Creación Propia

La Metodología Prototyping, ejemplificada en la Ilustración 8 fue elegido como un método ágil llamado creación de prototipos para desarrollar el sistema. Esta metodología fue seleccionada ya que se adecua a las necesidades y el enfoque del desarrollo del sistema, permite la retroalimentación etapa por etapa de las partes interesadas.

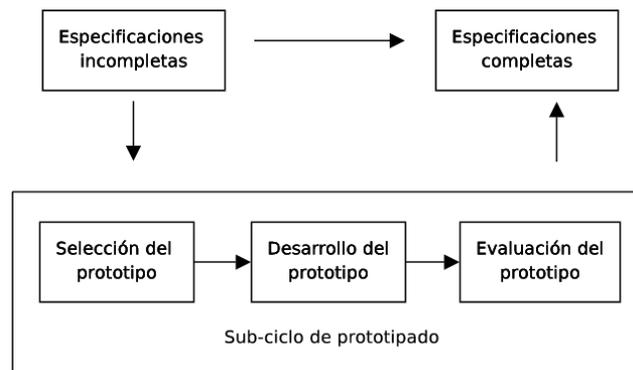


Ilustración 8. Ciclo de vida de un prototipo Fuente:(ResearchGate, 2022)

La creación de prototipos es una herramienta útil para casi todas las actividades de diseño y desarrollo de software. Para el problema táctico de este trabajo, el prototipo de diseño se basará en la especificación de requisitos con los siguientes supuestos:

A) Definición de los objetivos del prototipo.

A través de una serie de reuniones con el cliente, se dieron a conocer los objetivos específicos que conformarían al sistema.

- Analizar los requerimientos de usuario.
- Generar diagramas, esquemas y diseños del funcionamiento e interacción.
- Desarrollo del prototipo para el usuario que lo requiere
- Analizar el proceso actual de seguimiento de actividades académicas iniciando el área del Laboratorio de Iluminación Artificial (LIA).
- Determinar los requerimientos del usuario para el diseño de la aplicación.

- Establecer diagramas de diseño a través de diferentes arquitecturas de software que permitan la elaboración de diagramas, esquemas, simulaciones del funcionamiento.
- Usar lenguajes de programación y tecnologías como:
 - a) Node JS para Backend
 - b) React JS para Front end y
 - c) MySQL para el manejo de bases de datos, respectivamente para la obtención de la primera versión.
- Evaluar a través de pruebas de software (pruebas unitarias o Unit Testing)
- Retroalimentar el desarrollo de la aplicación con base a la funcionalidad solicitada por el usuario.

B) Validación de los requerimientos a través de la funcionalidad del producto.

Tabla:1Requerimientos Fuente: Creación propia

| TABLA DE REQUERIMIENTOS | |
|---|------------------------|
| Requerimiento | Responsable |
| Registrarse | Usuario /Administrador |
| Hacer login | Usuario/Administrador |
| Subir archivos | Usuario |
| Visualizar archivos | Administrador |
| Definir el estatus de los archivos entregados | Administrador |
| Visualizar el estatus de los archivos | Usuario |

I. Diagramas. Los diagramas cumplieron con la finalidad de determinar cada una de las acciones a las que cada uno de los usuarios existentes (usuario y administrador) podría tener acceso (Ilustración9).

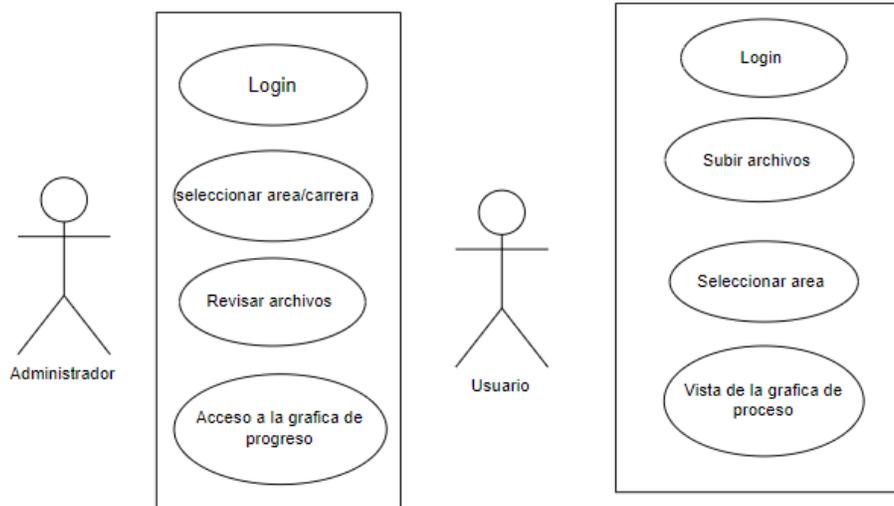


Ilustración 9. Diagramas de Casos de uso Fuente: Creación Propia

C) Diseñar y desarrollar el prototipo.

I. Mockups. El objetivo principal de los mockups es, tener una ligera idea acerca de cada uno de los elementos que deben conformar la página para que esta puedan ser manipuladas cada una de las acciones con las que cuenta. Dentro de la planificación del sistema, se generaron distintas ideas acerca de las vistas (Ilustración 10-13) que se adecuarían a las necesidades del cliente y a su interacción con la página.

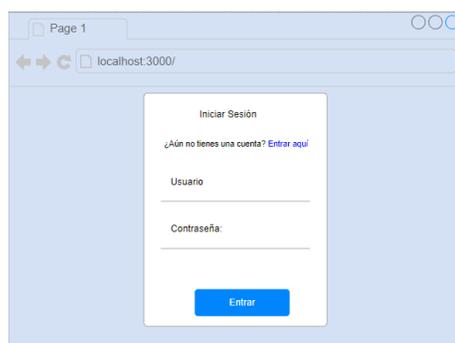


Ilustración 10. Vista Login Fuente: Creación Propia

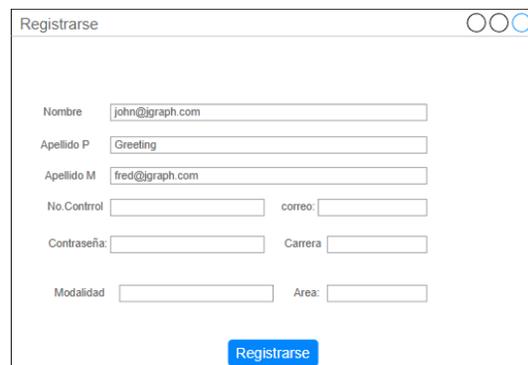


Ilustración 11. Vista registro Fuente: Creación Propia

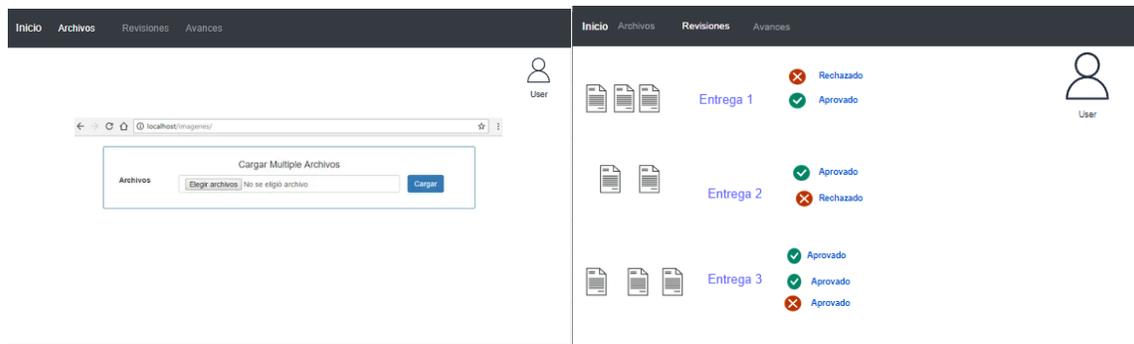


Ilustración 12. Vista cargar documento y estatus Fuente: Creación propia

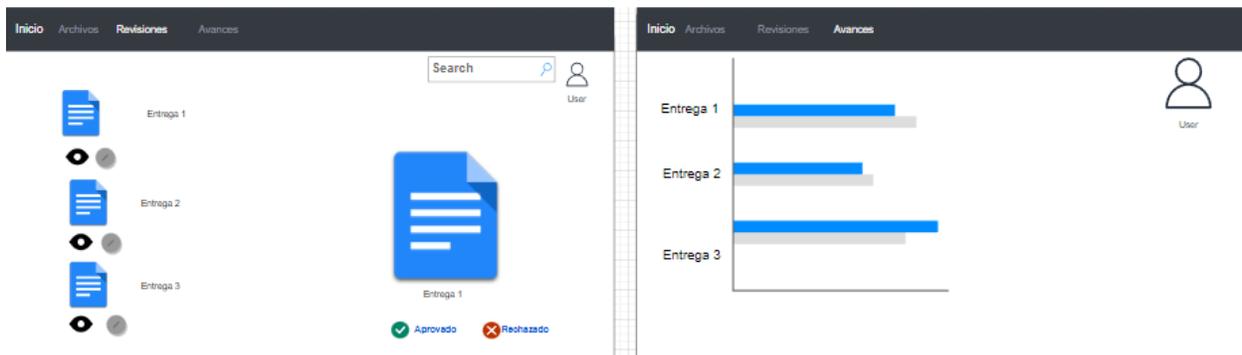


Ilustración 13. Vista del archivo y el avance del usuario. Fuente: Creación propia

II. Estructura de carpetas. La implementación de una estructura dentro de las carpetas (véase ilustración 14) nos da lugar a una mayor organización y una mejor accesibilidad dentro de los componentes que pueden ser reutilizados, eso permite tener una ruta específica para que estos puedan ser mandados llamar de una manera correcta.

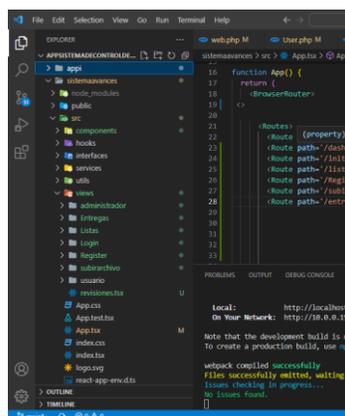


Ilustración 14. Estructura de carpetas Fuente: Creación propia

III. Componentes. El uso de los componentes nos permitió separar la interfaz de usuario en piezas independientes, reutilizables y pensar en cada pieza de forma aislada. La

existencia de los componentes permite la creación de las vistas que mantendrán la interacción con el cliente.

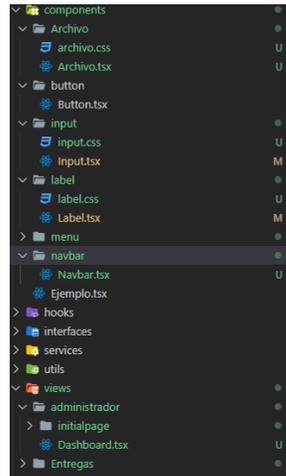


Ilustración 15. Componentes Fuente: Creación propia

IV. Migraciones. Se diseñó el modelo físico de la base de datos con el fin de conocer específicamente cada una de las tablas, los campos y las relaciones que existen entre cada una de las tablas y así dar lugar a las migraciones necesarias tal como se muestra en la imagen (Ilustración 16).

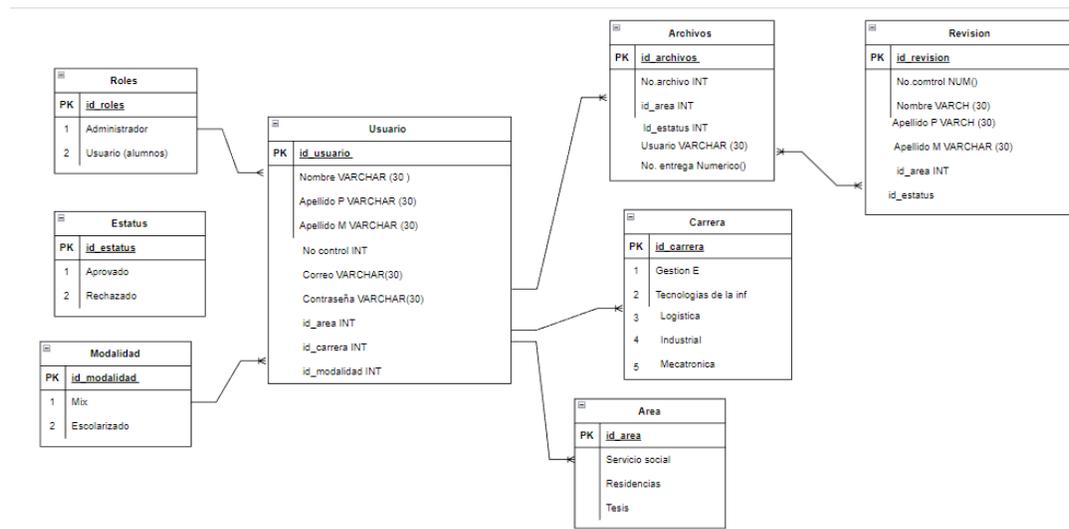
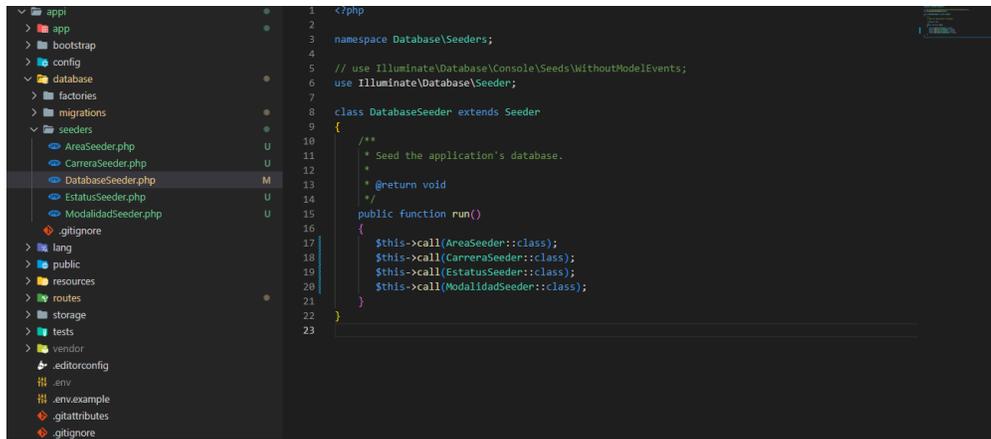


Ilustración 16. Modelo físico de base de datos Fuente: Creación propia

V. Semillas. Un Seed para insertar datos a nuestra tabla en la base de datos. Son componentes de Laravel que permiten comunicarse con la base de datos de una forma más eficiente y optimizada. (Ilustración 17).

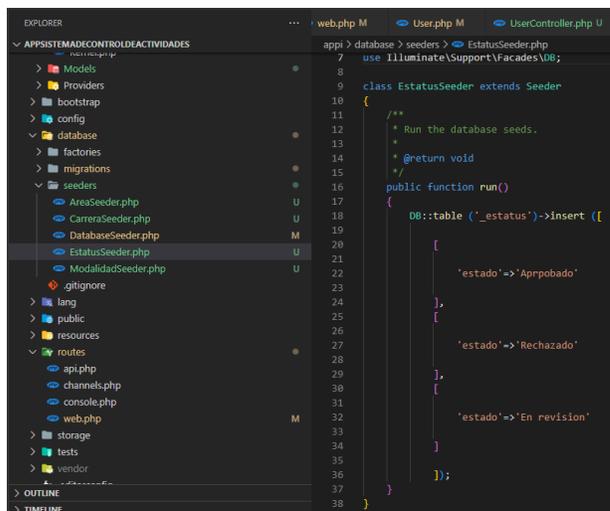


```
1 <?php
2
3 namespace Database\Seeders;
4
5 // use Illuminate\Database\Console\Seeds\WithoutModelEvents;
6 use Illuminate\Database\Seeder;
7
8 class DatabaseSeeder extends Seeder
9 {
10     /**
11      * Seed the application's database.
12      *
13      * @return void
14      */
15     public function run()
16     {
17         $this->call(AreaSeeder::class);
18         $this->call(CarreraSeeder::class);
19         $this->call(EstatusSeeder::class);
20         $this->call(ModalidadSeeder::class);
21     }
22 }
23
```

Ilustración 17. Seeds Fuente: Creación Propia

D) Evaluar el funcionamiento del prototipo

El prototipo muestra como ya se tiene una conexión exitosa con la base de datos, en la base podemos apreciar las semillas que fueron definidas como parte de la programación. (Ilustración 18 y 19)



```
1 use Illuminate\Support\Facades\DB;
2
3 class EstatusSeeder extends Seeder
4 {
5     /**
6      * Run the database seeds.
7      *
8      * @return void
9      */
10     public function run()
11     {
12         DB::table ('_estatus')->insert ([
13             [
14                 'estado'=>'Aprprobado'
15             ],
16             [
17                 'estado'=>'Rechazado'
18             ],
19             [
20                 'estado'=>'En revision'
21             ]
22         ]);
23     }
24 }
25
26
27
28
```

Ilustración 18. Pruebas de la BD Fuente: Creación propia

The screenshot shows the phpMyAdmin interface for a MySQL database named 'sistema_cdaa'. The table '_estatus' is selected, and the following data is displayed:

| | id_estatus | estado | created_at | updated_at |
|--------------------------|------------|-------------|------------|------------|
| <input type="checkbox"/> | 1 | Aprobado | NULL | NULL |
| <input type="checkbox"/> | 2 | Rechazado | NULL | NULL |
| <input type="checkbox"/> | 3 | En revision | NULL | NULL |

The interface includes a left sidebar with a database tree, a top navigation bar with options like 'Examinar', 'Estructura', 'SQL', etc., and a main content area with a query editor showing 'SELECT * FROM `_estatus`', a table view with search and sorting options, and a bottom section for operations like 'Imprimir', 'Copiar al portapapeles', 'Exportar', 'Mostrar gráfico', and 'Crear vista'.

Ilustración 19.BD Fuente: Creación propia

CAPÍTULO 5: RESULTADOS

12. Resultados

Cada una de las vistas que conforman el Frontend fueron prediseñadas como parte del desarrollo del proyecto a través de Mockups con el fin de tener distintas versiones del diseño más realista que se le presentara al cliente. Cada una de las vistas tienen ligeros cambios estéticos (véase ilustraciones 20-24) que no afectan el funcionamiento del sistema en conjunto, generalmente fueron diseñadas con cada uno de los elementos necesarios para que el cliente pueda interactuar con el sistema de manera satisfactoria.



Inicia sesión

¿Aún no tienes una cuenta? [Entra aquí](#)

Usuario

Contraseña

Entrar

Ilustración 20. Vista Login Fuente: Creación propia



Ilustración 21. Vista página inicial Fuente: Creación propia

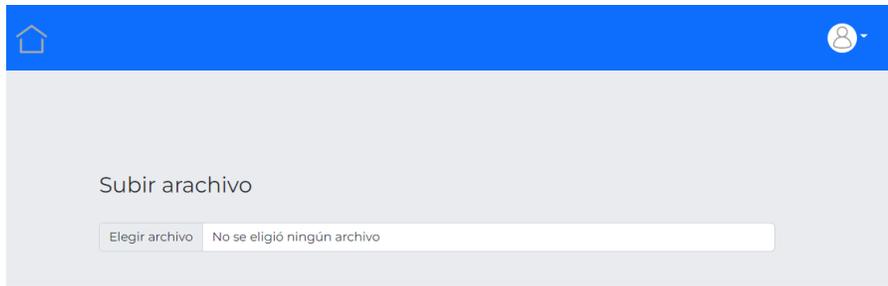


Ilustración 22. Vista subir archivo Fuente: Creación propia

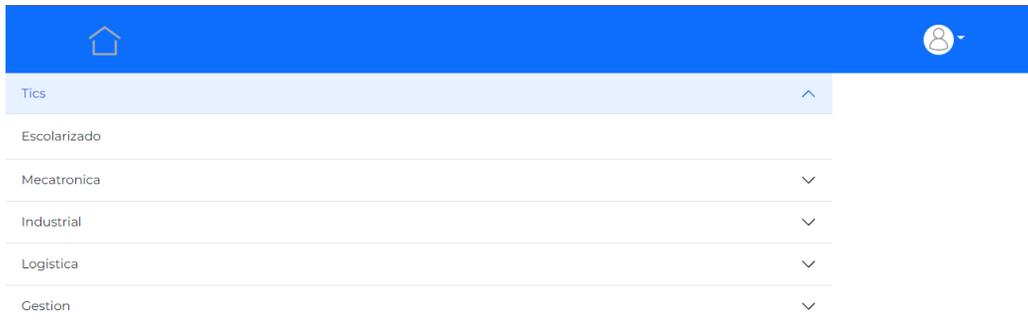


Ilustración 23. Vista selección de carreras Fuente: Creación propia

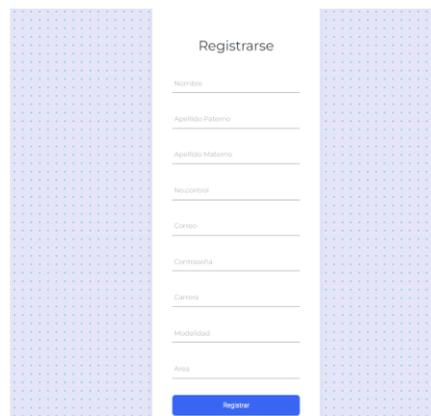


Ilustración 24. Vista de registro Fuente: Creación propia

El resultado de las integraciones de las migraciones, los controladores y las semillas se ven reflejados dentro de la base de datos (Ilustración 25 y 26) que ya había sido diseñada, y de la cual esperamos tener una exitosa conexión para el envío y actualización de la información necesaria.

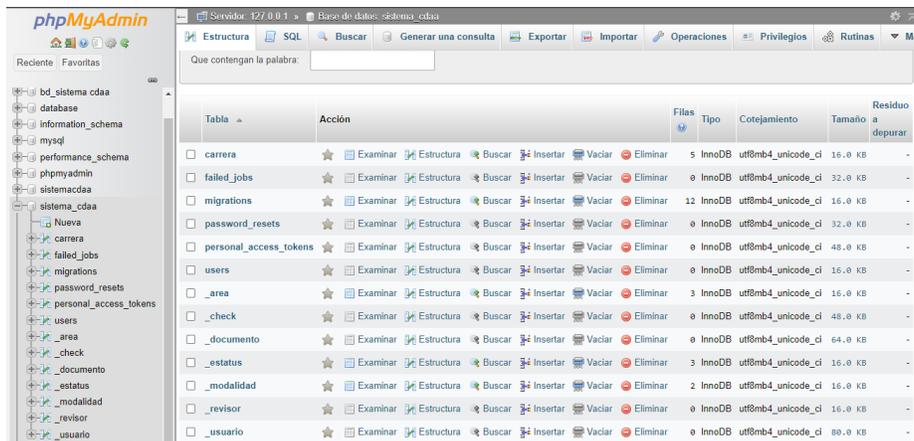


Ilustración 25. BD Fuente: Creación propia

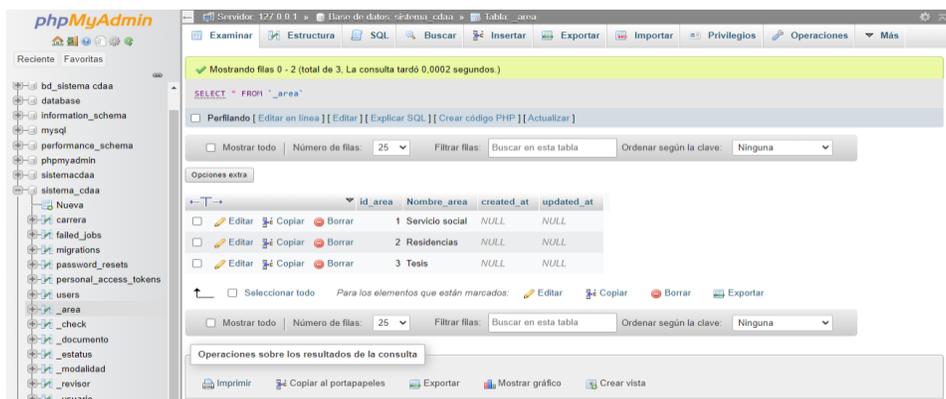


Ilustración 26. Representación del funcionamiento de semillas Fuente: Creación propia

Dentro del desarrollo del sistema se diseñaron vistas (Ilustración 27-29) de actividades principales para el administrador y el usuario, vistas en las que la interacción consiste en la carga de documentos, el estatus de cada uno de los usuarios que se encuentran registrados, y el avance de cada alumno reflejado gráficamente, la finalización de estas vistas no fue posible debido al tiempo de ejecución del proyecto y a algunas dificultades que se presentaron durante la programación y el intento de conexiones de Backend y Frontend las vistas quedan como actividades de seguimiento del proyecto.

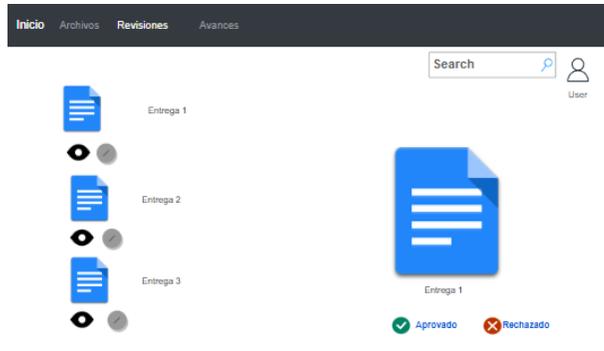


Ilustración 27. estatus de los documentos Fuente: Creación propia



Ilustración 28. Vista de estatus Fuente: Creación propia



Ilustración 29. Vista de grafica de avances Fuente: Creación propia

CAPÍTULO 6: CONCLUSIONES

13. Conclusiones del Proyecto

Para el desarrollo del prototipo de este sistema, se trabajó en conocer y aprender del manejo de nuevos lenguajes y herramientas con características relevantes, que permiten ser los ideales para este tipo de proyectos.

Se estima que al proyecto en general se le pueda dar un seguimiento una vez que esté cumpla con los objetivos que se establecieron durante este lapso de tiempo. El proyecto está abierto a innovaciones y cambios que permitan la eficacia de sus funciones.

CAPÍTULO 7: COMPETENCIAS DESARROLLADAS

Durante este proyecto se desarrollaron habilidades y conocimientos totalmente nuevos y requeridos para que el proyecto pudiera ser implementado, todas y cada una de estas competencias llevaron su tiempo y su método de implementación para que estas pudieran llevarnos a los resultados esperados.

14. Competencias desarrolladas y/o aplicadas.

1. Apliqué habilidades directivas y de ingeniería en el diseño, para la estética y vista de los objetivos planteados.
2. Implemente el uso de nuevas tecnologías.
3. Gestiona posibles cambios que se adecuen a las necesidades que se presenten, sin afectar al funcionamiento y a los objetivos preestablecidos.
4. Aplica métodos de modelado de sistema a través de diagramas UML que permiten el análisis de cada una de las funciones posibles dentro del sistema.
5. Gestiona pruebas del sistema para gestionar la calidad y seguridad.
6. Gestiona sistemas integrales de calidad para la mejora de los procesos, ejerciendo un liderazgo estratégico y un compromiso ético.
7. Aplica metodologías de investigación para desarrollar e innovar modelos, sistemas y procesos de desarrollo.

CAPÍTULO 8: FUENTES DE INFORMACIÓN

15. Fuentes de información

Referencias de internet:

Aplyca Tecnologías. (2022). *React: framework multi-propósito - Aplyca*.

<https://www.aplyca.com/blog/react-biblioteca-javascript-multiproposito>

crehana. (2022). *¿Qué es Python y para qué sirve este sistema operativo?*

<https://www.crehana.com/blog/transformacion-digital/que-es-python/>

DONGEE. (2022). *¿Qué es Node JS? » Dongee*.

<https://www.dongee.com/tutoriales/que-es-node-js/>

Google maps. (2022). *Instituto Tecnológico de Pabellón de Arteaga - Google Maps*.

[https://www.google.com/maps/place/Instituto+Tecnológico+de+Pabellón+de+Arteaga/@22.161362,-](https://www.google.com/maps/place/Instituto+Tecnológico+de+Pabellón+de+Arteaga/@22.161362,-102.268966,15z/data=!4m5!3m4!1s0x0:0x34c1762f6b2e30a1!8m2!3d22.161362!4d-102.268966)

[102.268966,15z/data=!4m5!3m4!1s0x0:0x34c1762f6b2e30a1!8m2!3d22.161362!4d-102.268966](https://www.google.com/maps/place/Instituto+Tecnológico+de+Pabellón+de+Arteaga/@22.161362,-102.268966,15z/data=!4m5!3m4!1s0x0:0x34c1762f6b2e30a1!8m2!3d22.161362!4d-102.268966)

Hostinger. (2022). *¿Qué Es PHP? Una Introducción Para Principiantes*.

<https://www.hostinger.mx/tutoriales/que-es-php>

Hubspot. (2022). *¿Qué es Angular? Características y ventajas*.

<https://blog.hubspot.es/website/que-es-angular>

IEBS. (2022). *Framework: Qué es, para qué sirve y por qué deberías usarlo*.

<https://www.iebschool.com/blog/framework-que-es-agile-scrum/>

IONOS. (2022). *HTML*. [https://www.ionos.mx/digitalguide/paginas-web/desarrollo-](https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/lenguajes-de-programacion-web/)

[web/lenguajes-de-programacion-web/](https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/lenguajes-de-programacion-web/)

Kinsta. (2022a). *¿Qué es MySQL? Una Explicación para Principiantes*.

<https://kinsta.com/es/base-de-conocimiento/que-es-mysql/>

Kinsta. (2022b). *¿Qué es una Petición HTTP?* [https://kinsta.com/es/base-de-](https://kinsta.com/es/base-de-conocimiento/que-es-una-peticion-http/)

[conocimiento/que-es-una-peticion-http/](https://kinsta.com/es/base-de-conocimiento/que-es-una-peticion-http/)

MCS. (2022). *¿Qué Es MySQL? Explicación Detallada Para Principiantes*.

<https://www.hostinger.es/tutoriales/que-es-mysql>

mdn. (2022a). *¿Qué es el CSS? - Aprende sobre desarrollo web | MDN*.

https://developer.mozilla.org/es/docs/Learn/CSS/First_steps/What_is_CSS

- mdn. (2022b). *Introducción a Express/Node - Aprende sobre desarrollo web | MDN.*
https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction
- MDN Plus. (2022). *¿Qué es JavaScript? - Aprende sobre desarrollo web | MDN.*
https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/What_is_JavaScript
- Murcia. (2020). *Desarrollo de Aplicaciones WEB. Introducción a HTML y CSS. Página Rafael Barzanallana. Universidad de Murcia.*
<https://www.um.es/docencia/barzana/DAWEB/2017-18/daweb-tema-1-introduccion-html-css.html>
- Oracle. (2022). *Qué es una base de datos | Oracle México.*
<https://www.oracle.com/mx/database/what-is-database/>
- OVH. (2022). *¿Qué es PostgreSQL? Usos y soluciones asociadas | OVHcloud.*
<https://www.ovhcloud.com/es/lp/postgresql-definition/>
- ResearchGate. (2022). *4: Ciclo de vida prototipo | Descargar Diagrama Científico.*
https://www.researchgate.net/figure/Figura-24-Ciclo-de-vida-prototipado_fig1_39425588
- rockocontent. (2022). *Framework: ¿qué es y cuál es su función en Internet?*
<https://rockcontent.com/es/blog/framework/>
- talenty. (2022). *¿Qué es Laravel y para qué sirve? | Talenty Blog.*
<https://talenty.tech/blog/que-es-laravel/>
- UAL. (2022). *La Web.*
<https://w3.ual.es/~vruiz/Docencia/Apuntes/Networking/Protocols/Level-5/02-WWW/index.html>

CAPÍTULO 9. ANEXOS