

REPÚBLICA DEL ECUADOR



**INSTITUTO SUPERIOR
TECNOLÓGICO TENA**
Tecnología, Innovación y Desarrollo

**CARRERA DE TECNOLOGÍA SUPERIOR EN
DESARROLLO DE SOFTWARE**

**Desarrollo de un Sistema Web para la Automatización de Solicitudes en el
Instituto Superior Tecnológico Tena**

Trabajo de Integración Curricular, presentado como requisito parcial para optar por el
título de Tecnólogo Superior en Desarrollo de Software.

AUTOR: David Alexander Carrasco Cando

TUTOR: Ing. Marco Patricio Guanipatín Ramírez

Tena - Ecuador

2024-IIS

APROBACIÓN DEL TUTOR

Ing. Marco Patricio Guanipatin Ramirez

PROFESOR DEL INSTITUTO SUPERIOR TECNOLÓGICO TENA.

CERTIFICA:

En calidad de Tutor del Proyecto Integrador denominado: Desarrollo de un Sistema Web para la Automatización de Solicitudes en el Instituto Superior Tecnológico Tena, de autoría del señor **CARRASCO CANDO DAVID ALEXANDER**, con CC. 1501200040 estudiante de la Carrera de Tecnología Superior en el Desarrollo de Software del Instituto Superior Tecnológico Tena, CERTIFICO que se ha realizado la revisión prolija del Trabajo antes citado, cumple con los requisitos de fondo y de forma que exigen los respectivos reglamentos e instituciones.



Ing. Marco Patricio Guanipatin Ramirez

TUTOR DEL TIC

Tena, 10 de febrero de 2025

CERTIFICACIÓN DEL TRIBUNAL CALIFICADOR

Tena, 11 de marzo de 2025

Los Miembros del Tribunal de Grado abajo firmantes, certificamos que el Trabajo de Titulación denominado: Desarrollo de un Sistema Web para la Automatización de Solicitudes en el Instituto Superior Tecnológico Tena, presentado por CARRASCO CANDO DAVID ALEXANDER, con CC: 1501200040, estudiante de la Carrera de Tecnología Superior en Desarrollo de Software del Instituto Superior Tecnológico Tena, ha sido corregida y revisada; por lo que autorizamos su presentación.

Atentamente;

ING. FAUSTO CLAUDIO
PRESIDENTE DEL TRIBUNAL

LIC. MARJORIE JISSELA JIMÉNEZ FRANCO
MIEMBRO DEL TRIBUNAL

ING. JUAN ESPIN
MIEMBRO DEL TRIBUNAL

AUTORÍA

Yo, CARRASCO CANDO DAVID, con CC: 1501200040, declaro ser autor del presente Trabajo de Titulación denominado: Desarrollo de un Sistema Web para la Automatización de Solicitudes en el Instituto Superior Tecnológico Tena y absuelvo expresamente al Instituto Superior Tecnológico Tena, y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo al Instituto Superior Tecnológico Tena, la publicación de mi trabajo de Titulación en el repositorio institucional- biblioteca Virtual.

AUTOR:

CARRASCO CANDO DAVID ALEXANDER

CÉDULA: 1501200040

FECHA: Tena, 18 de febrero de 2025

CARTA DE AUTORIZACIÓN POR PARTE DEL AUTOR

Yo, CARRASCO CANDO DAVID ALEXANDER, declaro ser autor del Trabajo de Titulación titulado: DESARROLLO DE UN SISTEMA WEB PARA LA AUTOMATIZACIÓN DE SOLICITUDES EN EL INSTITUTO SUPERIOR TECNOLÓGICO TENA, como requisito para la obtención del Título de: TECNÓLOGO SUPERIOR EN DESARROLLO DE SOFTWARE: autorizo al Sistema Bibliotecario del Instituto Superior Tecnológico Tena, para que con fines académicos, muestre al mundo la producción intelectual del Instituto, a través de la visualización de su contenido que consta en el Repositorio Digital Institucional.

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio el Instituto. El Instituto Superior Tecnológico Tena, no se responsabiliza por el plagio o copia del presente trabajo que realice un tercero. Para constancia de esta autorización, en la ciudad de Tena, 10 de febrero de 2025, firma el autor.

AUTOR: Carrasco Cando David Alexander.

FIRMA: 

CÉDULA: 1501200040

DIRECCIÓN: Napo, Carlos julio Arosemena Tola

CORREO ELECTRÓNICO: davidcarrasco258@gmail.com

TELÉFONO: s/n

CELULAR: 0979108983

DATOS COMPLEMENTARIOS

TUTOR: Ing. Marco Patricio Guanipatín Ramírez

TRIBUNAL DEL GRADO:

ING. FAUSTO CLAUDIO.

LIC. MARJORIE JISSELA JIMÉNEZ FRANCO.

ING. FAUSTO CLAUDIO.

DEDICATORIA

Para mi madre, pilar incondicional en cada decisión que he tomado. Tu apoyo constante y amor han sido la guía en mi camino, iluminando cada paso que doy.

Magdalena Cando Cando

Para mi padre, cuyo carácter me enseñó que, aun en medio de la adversidad, debemos dar lo mejor de nosotros y aprender a superar cada obstáculo. Tu fortaleza es el ejemplo que me impulsa a seguir adelante con entrega total.

Angel Polivio Carrasco Cando

Para mis hermanas, que con su inquebrantable apoyo han nutrido mi crecimiento personal y emocional. Su cariño y comprensión han sido esenciales en mi formación y en el descubrimiento de mi propio camino.

Lic. Brigith Angelica Carrasco Cando

Lic. Daniela Stefania Carrasco Cando

Para mi hermano, recordándote que esta meta es solo el primer paso para seguir un camino lleno de aprendizajes y oportunidades. Que mi ejemplo te inspire y guíe hacia un futuro lleno de éxitos y crecimiento.

Matías Sebastián Carrasco Cando

Para el resto de mi familia, cuyo constante cuidado y apoyo han velado por mí a lo largo de cada paso. Su amor y respaldo han sido la luz en mis días, y mis deseos son que la felicidad y el bienestar los acompañen siempre, tal como ustedes cuidaron de mí.

Familia Carrasco Cando

AGRADECIMIENTO

Quiero expresar mi sincero agradecimiento a todas aquellas personas que, de manera directa o indirecta, han contribuido a la realización de este proyecto. Cada consejo, palabra de aliento y gesto de apoyo se han convertido en piezas fundamentales que han forjado el camino hasta este logro.

Agradezco profundamente a mis profesores y mentores, quienes compartieron generosamente su conocimiento y sabiduría. Su orientación me impulsó a superar desafíos, a crecer profesionalmente y a dar lo mejor de mí en cada etapa.

Extiendo mi gratitud a mis amigos y compañeros, por ser esa red de apoyo constante. Las conversaciones, los intercambios de ideas y la camaradería fueron ese impulso extra, especialmente en los momentos en que las dudas acechaban. Su contribución fue clave para transformar cada obstáculo en una oportunidad de aprendizaje.

Quiero expresar mi profundo agradecimiento al Instituto Superior Tecnológico Tena, que ha sido mucho más que un centro educativo en mi camino. Su continuo compromiso con la excelencia académica y la formación integral ha dejado una huella indeleble en mi desarrollo. Cada lección, cada reto y cada apoyo recibido en sus aulas han contribuido a forjar mi determinación y a iluminar mi futuro. Gracias por inspirarme, por impulsar mis aspiraciones y por ser un pilar fundamental en mi trayectoria.

ÍNDICES DE CONTENIDO

PORTADA	i
APROBACIÓN DEL TUTOR	ii
CERTIFICACIÓN DEL TRIBUNAL CALIFICADOR	iii
AUTORÍA	iv
CARTA DE AUTORIZACIÓN POR PARTE DEL AUTOR	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICES DE CONTENIDO	viii
ÍNDICE DE FIGURAS	xi
TEMA	1
RESUMEN	2
ABSTRACT	3
3 FUNDAMENTACIÓN DEL TEMA	4
3.1 Necesidad	4
3.2 Actualidad	4
3.3 Importancia	4
3.4 Presentación del problema profesional a responder	5
3.5 Delimitación	5
3.5.1 Delimitación Espacial	5
3.5.2 Delimitación Temporal	5
3.5.3 Delimitación Técnica	5
3.5.4 Unidades de Observación	6
3.6 Beneficiarios	6

3.6.1	Directos	6
3.6.2	Indirectos.....	6
4	OBJETIVOS.....	7
4.1	Objetivo General	7
4.2	Objetivos Específicos.....	7
5	ASIGNATURAS INTEGRADORAS.....	8
6	FUNDAMENTACIÓN TEÓRICA.....	9
6.1	Antecedentes	9
6.2	Gestión Documental.....	9
6.3	Norma ISO 15489	9
6.4	Sitio Web.....	10
6.5	Sistema Web.....	10
6.6	Lenguajes de Programación	12
6.7	Gestor de Base de Datos	12
6.7.1	Ventajas y Desventajas de un Gestor de Base de Datos	12
6.8	Tecnologías Utilizadas.....	14
6.8.1	PHP	14
6.8.2	MySQL.....	15
6.8.3	HTML5	17
6.8.4	CSS3.....	18
6.8.5	PHPWork	19
6.8.6	PHPMailer.....	19
6.8.7	Visual Studio Code	20
6.8.8	Scrum Simplificado.....	20
6.9	Marco Legal	22

7	METODOLOGÍA	24
7.1	Análisis de la información	24
7.1.1	Creación del Product Backlog.....	25
7.1	Diseño	25
7.1.1	Planificación del Sprint.....	25
7.1.2	Diseño de la Base de Datos.....	26
7.1.3	Estructura de carpetas	27
7.1.4	Arquitectura del sistema.....	29
7.2	Desarrollo.....	31
7.2.1	Ejecución del Sprint	31
7.2.2	Inspección y Adaptación (Diario personal).....	38
7.2.3	Revisión del Sprint.....	39
7.2.4	Retrospectiva del Sprint	39
7.2.5	Repetición de los Sprints.....	39
7.2.6	Entrega del Incremento	40
8	RESULTADOS	41
8.1	Análisis de la información	41
8.2	Maquetación.....	41
8.3	Desarrollo.....	42
9	CONCLUSIONES	46
10	RECOMENDACIONES	48
11	BIBLIOGRAFÍA	50
12	ANEXOS	52

ÍNDICE DE TABLAS

Tabla 1 <i>Asignaturas integradoras</i>	8
Tabla 2 <i>Ventajas y Desventajas de un gestor de Base de Datos</i>	12
Tabla 3 <i>Ventajas y Desventajas de PHP</i>	14
Tabla 4 <i>Product Backlog</i>	25
Tabla 5 <i>Planificación del Sprint</i>	25
Tabla 6 <i>Ejecución del Sprint</i>	31

ÍNDICE DE FIGURAS

Figura 1 <i>Aplicaciones web (Web Apps) vs Sitios web: cómo elegir la correcta</i>	11
Figura 2 <i>Sistemas Cliente-Servidor</i>	17
Figura 3 <i>Estructura para una página HTML5</i>	18
Figura 4 <i>Banco de preguntas</i>	24
Figura 5 <i>Modelo Entidad-Relacional de la Base de Datos</i>	27
Figura 6 <i>Estructura de carpetas</i>	29
Figura 7 <i>Cliente-Servidor</i>	31
Figura 8 <i>Interfaz del usuario</i>	33
Figura 9 <i>Interfaz del administrador</i>	33
Figura 10 <i>Crear nueva solicitud</i>	35
Figura 11 <i>Interfaz del usuario - Crear una solicitud</i>	38

TEMA

Desarrollo de un Sistema Web para la Automatización de Solicitudes en el Instituto
Superior Tecnológico Tena

RESUMEN

Este proyecto de tesis aborda la problemática de la gestión manual de solicitudes en el Instituto Superior Tecnológico Tena, identificada como una fuente de ineficiencia y dificultades en el seguimiento de los procesos. Con el objetivo de optimizar esta gestión, se desarrolló e implementó un sistema web automatizado utilizando PHP y MySQL para la lógica del servidor, y HTML5 y CSS3 para el diseño de interfaces de usuario intuitivas. La definición de los requisitos del sistema se basó en un exhaustivo levantamiento de información, realizado a través de cuestionarios dirigidos a la encargada del proceso. Los resultados de la implementación demostraron una mejora significativa en la eficiencia, reflejada en una reducción del 50% en los tiempos de respuesta y un incremento del 80% en la satisfacción de los usuarios. Además, la incorporación de un módulo de notificaciones contribuyó a aumentar la transparencia y la trazabilidad del proceso. En conclusión, este sistema web automatizado representa una solución efectiva para la gestión de solicitudes en el Instituto Superior Tecnológico Tena, optimizando la gestión documental, facilitando el seguimiento de solicitudes y mejorando la eficiencia administrativa general.

Palabras clave: PHP, MySQL, Sistemas web, HTML, CSS, Gestión Documental, Solicitudes.

ABSTRACT

This thesis project addresses the issue of manual request management at Instituto Superior Tecnológico Tena, identified as a source of inefficiency and difficulties in process tracking. In order to optimize this management, an automated web system was developed and implemented using PHP and MySQL for server-side logic, and HTML5 and CSS3 for designing intuitive user interfaces. The system requirements were defined based on an exhaustive information gathering process, carried out through questionnaires directed at the person in charge of the process. The implementation results demonstrated a significant improvement in efficiency, reflected in a 50% reduction in response times and an 80% increase in user satisfaction. Additionally, the incorporation of a notifications module contributed to enhanced transparency and traceability of the process. In conclusion, this automated web system represents an effective solution for managing requests at Instituto Superior Tecnológico Tena. It optimizes document management, facilitates request tracking, and improves overall administrative efficiency.

Keywords: PHP, MySQL, Web systems, HTML, CSS, Document Management, Requests.

Reviewed by:



Language Center Coordinator

3 FUNDAMENTACIÓN DEL TEMA

3.1 Necesidad

El Instituto Superior Tecnológico Tena enfrenta importantes desafíos en la gestión documental debido a la falta de un sistema automatizado. Actualmente, las solicitudes se gestionan de manera manual, lo que genera problemas como la ausencia de un registro formal de los documentos que ingresan físicamente, la imposibilidad de realizar un seguimiento adecuado al flujo de estos, y posibles errores administrativos. Estos problemas afectan directamente la eficiencia de los procesos internos y la satisfacción de los usuarios.

Un banco de preguntas realizado al personal encargado del proceso respalda estas observaciones, destacando la necesidad de una solución tecnológica. La falta de trazabilidad y registro en el manejo de la documentación no solo compromete la operatividad del Instituto, sino también su capacidad para ofrecer un servicio de calidad.

3.2 Actualidad

Actualmente, el proceso de gestión documental en el Instituto se realiza de forma completamente manual. Este enfoque carece de la estructura necesaria para garantizar la eficiencia y transparencia, limitando la capacidad de los usuarios para realizar un seguimiento efectivo de las solicitudes.

Las principales limitaciones incluyen la falta de un registro centralizado de documentos, procesos lentos y confusos, y la imposibilidad de generar reportes útiles para la toma de decisiones. No se han implementado sistemas similares en instituciones de referencia, lo que resalta la necesidad de desarrollar una solución propia adaptada a las necesidades del Instituto.

3.3 Importancia

La implementación de un sistema de gestión documental permitirá al Instituto Superior Tecnológico Tena mejorar significativamente sus procesos administrativos. Este proyecto proporcionará una herramienta que automatice el flujo de solicitudes, ofreciendo un tiempo de respuesta más rápido y un seguimiento preciso de cada

documento. Esto impactará positivamente en la experiencia de los usuarios y liberará tiempo valioso que podrá ser destinado a otras tareas prioritarias.

3.4 Presentación del problema profesional a responder

El Instituto Superior Tecnológico Tena enfrenta dificultades para gestionar solicitudes de los estudiantes debido al uso de métodos manuales y no digitalizados. Esto ocasiona demoras, errores en el manejo de información y una falta de trazabilidad en el flujo de documentos, afectando negativamente la experiencia y satisfacción de los usuarios.

Campo: Tecnologías de la Información y Comunicación

Área: Informática

Aspecto: Sistema web para la automatización de solicitudes

Sector: Programación

Línea de investigación: Tecnologías de la información y comunicación.

3.5 Delimitación

3.5.1 Delimitación Espacial

El Trabajo de Integración Curricular se realizó en el Instituto Superior Tecnológico Tena, el mismo que está ubicado en la vía Tena-Archidona en el km 1 ½.

3.5.2 Delimitación Temporal

El proyecto se llevó a cabo en el Periodo Académico 2024-IIS.

3.5.3 Delimitación Técnica

Módulos del Sistema:

Módulo Solicitudes: Los estudiantes podrán enviar sus solicitudes de forma remota a través de una interfaz web accesible desde dispositivos conectados a internet.

Módulo Proceso: Mejorará la trazabilidad del flujo de documentos al mantener un registro digital actualizado de cada etapa del proceso.

Módulo Historial: Facilitará el acceso a la información histórica para consultas o auditorías futuras.

Interfaz: Se diseñará una plataforma fácil de usar tanto para los estudiantes como para los administrativos, priorizando una experiencia de usuario eficiente y amigable.

Limitaciones:

El sistema no abordará procesos administrativos ajenos a las solicitudes estudiantiles ni reemplazará completamente el seguimiento humano en casos que requieran intervención manual o personalizada.

3.5.4 Unidades de Observación

Las unidades de observación que se contemplan para este trabajo están enfocadas directamente al personal administrativo y estudiantes, quienes interactúan directamente con los procesos de gestión documental.

3.6 Beneficiarios

3.6.1 Directos

Los beneficiarios directos del trabajo Integrador Curricular es el:

- Instituto Superior Tecnológico Tena.

3.6.2 Indirectos

- Profesores
- Estudiantes
- Personal administrativo.

4 OBJETIVOS

4.1 Objetivo General

Desarrollar un sistema web para mejorar la eficiencia y gestión de solicitudes en el Instituto Superior Tecnológico Tena para automatizar las solicitudes estudiantiles.

4.2 Objetivos Específicos

- Diseñar una base de datos estructurada que permita almacenar y organizar la información relacionada con las solicitudes de forma segura y eficiente.
- Implementar una interfaz web intuitiva y amigable que facilite el acceso de los usuarios y mejore la experiencia en la gestión de las solicitudes.
- Realizar pruebas necesarias para asegurar el correcto funcionamiento del registro, consulta y seguimiento de las solicitudes realizadas por los estudiantes.

5 ASIGNATURAS INTEGRADORAS

Tabla 1

Asignaturas integradoras

ASIGNATURAS INTEGRADORAS	
Asignaturas	Resultados de Aprendizaje
Base de Datos Avanzada	Desarrolla scripts de creación de bases de datos.
Programación de aplicaciones web	Utilizar un lenguaje de programación tanto del lado del cliente como del servidor para realizar aplicaciones web.
Programación Orientada a Objetos	Conectar a la base de datos para el manejo responsable y eficiente de los datos almacenados.

Nota. Asignaturas relacionadas con la ejecución del proyecto

6 FUNDAMENTACIÓN TEÓRICA

6.1 Antecedentes

La automatización de la gestión documental optimiza los procesos administrativos en instituciones educativas, mejorando la eficiencia, rapidez y seguridad en el manejo de información. Las solicitudes estudiantiles antes gestionadas manualmente, ahora se pueden realizar a través de plataformas digitales que procesan automáticamente la información. Esto agiliza los tiempos de respuesta y mejora la transparencia, permitiendo que los estudiantes consulten el estado de sus trámites en cualquier momento.

6.2 Gestión Documental

La gestión documental es un proceso esencial en cualquier organización, ya que asegura la adecuada administración, control y seguimiento de todos los documentos generados y recibidos. Este proceso no solo garantiza la autenticidad y disponibilidad de la información, sino que también facilita su distribución oportuna a las áreas correspondientes.

Asenjo Mestanza, J. F. (2020) indica que la gestión documental:

“Es la administración de toda la documentación y el aseguramiento de su oportuna distribución a las áreas o entidades respectivas, en coordinación con las áreas administrativas para su producción, control y seguimiento.

García, E (2010) indica: “La Gestión de documentos es el proceso de crear, mantener, usar, retener y disponer los documentos, garantizando su autenticidad, integridad, disponibilidad y conservación decretadas en normas.””

(https://repositorio.ucv.edu.pe/bitstream/handle/20.500.12692/59874/Asenjo_MJF-SD.pdf?sequence=1&isAllowed=y)

6.3 Norma ISO 15489

La gestión documental ha adquirido una relevancia significativa en la era digital actual, donde la eficiencia del manejo de documentos y archivos son cruciales para el éxito de las organizaciones.

Según NormasISO.org (2025) dice:

“En el mundo digital en el que vivimos, la gestión eficiente de documentos y archivos se ha vuelto esencial para las organizaciones. La norma ISO 15489 proporciona un marco de trabajo para lograr una gestión documental efectiva y asegurar que los documentos y archivos sean tratados de forma adecuada a lo largo de su ciclo de vida.” (<https://normasiso.org/norma-iso-15489/>)

6.4 Sitio Web

En la era digital actual, los sitios web se han convertido en herramientas fundamentales para la comunicación y la prestación de servicios. Estos sitios, compuestos por diversas páginas web interconectadas y accesibles a través de Internet, permiten a empresas, organizaciones y particulares compartir información de manera eficiente y versátil.

SoftwareLab (2024) indica que:

“Un sitio web es un conjunto de páginas web interconectadas, todas ellas compartiendo el mismo nombre de dominio y accesibles a través de Internet. Puede servir para diversos fines, como proporcionar información o servicios de una empresa, organización o particular. El contenido de los sitios web puede presentarse en varios formatos, como texto, imágenes, vídeos, audio y animación, lo que lo convierte en una plataforma versátil para compartir información sobre un tema o propósito específico.” (<https://softwarelab.org/es/blog/que-es-un-sitio-web/>)

6.5 Sistema Web

En el ámbito de la tecnología moderna, los sistemas web han revolucionado la forma en que interactuamos y gestionamos la información. Estas aplicaciones, que operan desde servidores en Internet o intranets, ofrecen potentes funcionalidades que van más allá de las páginas web tradicionales, adaptándose a las necesidades específicas de los usuarios.

Castillo Peña, G. E. (2018) establece:

“Los "Sistemas Web" o también conocido como "aplicaciones Web" son aquellos que están creados e instalados no sobre una plataforma o sistemas operativos

(Windows, Linux). Sino que se alojan en un servidor en Internet o sobre una intranet (red local). Su aspecto es muy similar a páginas Web que vemos normalmente, pero en realidad los 'Sistemas Web' tienen funcionalidades muy potentes que brindan respuestas a casos particulares.

Los sistemas Web se pueden utilizar en cualquier navegador Web (Chrome, Firefox, Microsoft Edge, etc.) sin importar el sistema operativo. Para utilizar las aplicaciones Web no es necesario instalarlas en cada computadora ya que los usuarios se conectan a un servidor donde se aloja el sistema.”

(https://repositorio.uladech.edu.pe/bitstream/handle/20.500.13032/2518/GESTION_DOCUMENTARIA_IMPLEMENTACION_CASTILLO_PENA_GERARDO_EDINSON.pdf?sequence=1&isAllowed=y)

Figura 1

Aplicaciones web (Web Apps) vs Sitios web: cómo elegir la correcta



Nota. Adaptado de Aplicaciones web (Web Apps) vs Sitios web: cómo elegir la correcta [Fotografía], por Click-IT | Servicios tecnológicos y de consultoría. <https://click-it.es/aplicacion-web-web-app-vs-sitio-web-como-elegir-la-correcta/>

6.6 Lenguajes de Programación

Desarrolladoresweb.org. (2024) dice:

“Un lenguaje de programación es un conjunto de reglas, sintaxis y palabras clave que permiten a los programadores dar instrucciones a una computadora para que realice tareas específicas. Estos lenguajes sirven como un intermediario entre los humanos y las máquinas, facilitando la creación de programas que transforman entradas en resultados deseados.” (<https://desarrolladoresweb.org/programacion/tipos-de-lenguajes-de-programacion-una-guia-exhaustiva/>)

6.7 Gestor de Base de Datos

Los sistemas gestores de bases de datos (SGBD) son esenciales en el mundo de la tecnología de la información, ya que proporcionan las herramientas necesarias para administrar y organizar grandes volúmenes de datos de manera eficiente.

Universidad Europea. (2022) indica:

“Un sistema gestor de bases de datos (SGBD) es un software constituido por una serie de programas dirigidos a crear, gestionar y administrar la información que se encuentra en la base de datos. Su principal objetivo es servir de interfaz entre los usuarios y las aplicaciones para facilitar la organización de los datos, garantizar su accesibilidad, calidad e integridad, brindando a su vez una manera eficaz de administrar esa información.

El lenguaje de manipulación, un lenguaje de definición de datos y un lenguaje de consulta son los tres componentes de un gestor de base de datos; por lo que puede trabajar a diferentes niveles, pero es invisible para el usuario final.”

(<https://universidadeuropea.com/blog/para-que-sirve-gestor-base-datos/#que-es-un-sistema-gestor-de-base-de-datos>)

6.7.1 Ventajas y Desventajas de un Gestor de Base de Datos

Tabla 2

Ventajas y Desventajas de un gestor de Base de Datos

Ventajas	Desventajas
----------	-------------

<p>Accesibilidad: una base de datos es accesible para múltiples usuarios a la vez, lo que significa que la información puede ser compartida y actualizada en tiempo real. Esto es particularmente útil para empresas y organizaciones que necesitan mantenerse al día con los cambios y actualizaciones de la información.</p>	<p>Costo: las bases de datos pueden ser costosas de implementar y mantener, especialmente si se requieren sistemas de hardware y software específicos. Además, a medida que la cantidad de información almacenada en la base de datos crece, los costos pueden aumentar significativamente.</p>
<p>Eficiencia: las bases de datos permiten la organización y clasificación de información en categorías específicas, lo que facilita la búsqueda de información y reduce el tiempo que se tarda en buscar información en un conjunto de datos más grande. Además, la automatización de tareas repetitivas ayuda a reducir errores humanos y a mejorar la eficiencia en general.</p>	<p>Complejidad: las bases de datos pueden ser complejas y requieren conocimientos técnicos para ser administradas adecuadamente. Si no se tienen los conocimientos necesarios, se puede incurrir en errores costosos y comprometer la integridad de la información almacenada.</p>
<p>Seguridad: las bases de datos pueden ser protegidas con contraseñas, encriptación y otros métodos de seguridad para proteger la información confidencial. Esto es</p>	<p>Fallos del sistema: si la base de datos falla o se corrompe, toda la información almacenada en ella puede estar en riesgo. Además, la recuperación de datos puede ser costosa y llevar mucho tiempo, lo que</p>

particularmente importante para empresas que manejan información financiera, de salud o de clientes.	puede afectar la operación del negocio o la organización.
--	---

6.8 Tecnologías Utilizadas

6.8.1 PHP

A pesar de que PHP a veces es subestimado por algunos desarrolladores que consideran que su uso ha disminuido en la era moderna de desarrollo web, sigue siendo una herramienta esencial y ampliamente utilizada. Su capacidad para integrarse con diversas tecnologías y su facilidad de uso lo mantienen relevante y valioso para numerosos proyectos web actuales.

Juan de Assembler Institute. (2023) menciona:

“PHP es un lenguaje de programación de uso general que se utiliza, sobre todo, en el entorno del desarrollo web. Este lenguaje se utiliza generalmente para desarrollar el back-end de una web, el lado del servidor. Aun así, tiene numerosas utilidades en front-end. Es por esto por lo que es uno de los principales lenguajes de programación en el mundo de la programación web.” (<https://assemblerinstitute.com/blog/que-es-php/>)

6.8.1.1 Ventajas y Desventajas de PHP

Tabla 3

Ventajas y Desventajas de PHP

Ventajas	Desventajas
Lenguaje de código abierto y con una extensa comunidad detrás.	No puede ocultarse el código fuente de las páginas que se desarrollan. Aun así, este hecho es indiferente en muchas ocasiones.

Fácil integración en las bases de datos. Además, puede ser usado en la gran mayoría.	Los scripts en PHP pueden tener un funcionamiento relativamente lento, lo que puede perjudicar la experiencia del usuario. Sin embargo, esto puede solucionarse mediante determinadas estrategias de caché.
Ofrece seguridad. Es muy útil para evitar ataques informáticos.	Aunque permite una gran seguridad informática, su configuración es extremadamente compleja.
Lenguaje multiplataforma y aceptado por los navegadores más populares (los mismos que aceptan HTML).	
Relativamente fácil de aprender. Cuenta con una sintaxis muy clara y puede usarse en proyectos simples. Al mismo tiempo, también es ideal para proyectos de alta complejidad.	
Permite trabajar con una gran cantidad de datos, por lo que es ideal incluso para las webs más populares.	

6.8.2 MySQL

MySQL es uno de los sistemas de gestión de datos más utilizados en el ámbito del desarrollo web y aplicaciones empresariales. Su modelo de código abierto y su

arquitectura cliente-servidor lo convierten en una herramienta poderosa y flexible para la creación y administración de bases de datos. A pesar de ser subestimado por algunos, MySQL sigue demostrando su utilidad y relevancia en proyectos actuales.

Gustavo B (2024) menciona esto sobre MySQL:

“MySQL es un sistema de gestión de bases de datos relacionales de código abierto (RDBMS, por sus siglas en inglés) con un modelo cliente-servidor. RDBMS es un software o servicio utilizado para crear y administrar bases de datos basadas en un modelo relacional.

¿Cómo funciona MySQL?

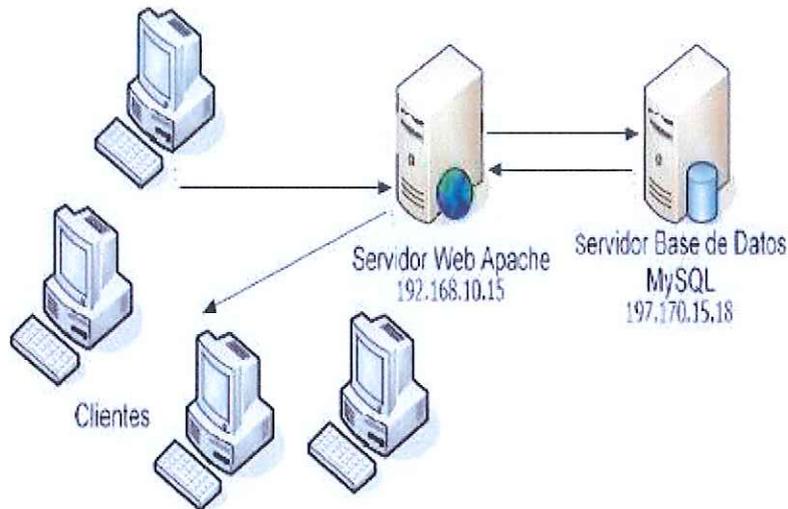
Cada cliente puede realizar una solicitud desde la interfaz gráfica de usuario (GUI) en sus pantallas, y el servidor producirá el output deseado, siempre que ambas partes entiendan la instrucción.

Los procesos principales que tienen lugar en un entorno MySQL son:

- MySQL crea una base de datos para almacenar y manipular datos, definiendo la relación de cada tabla.
- Los clientes pueden realizar solicitudes escribiendo instrucciones SQL específicas en MySQL.
- La aplicación del servidor responderá con la información solicitada y esta aparecerá frente a los clientes.”

(https://www.hostinger.mx/tutoriales/que-es-mysql/#%C2%BFQue_es_MySQL)

Figura 2
Sistemas Cliente-Servidor



Nota. Así actúan las peticiones cliente-servidor. Adoptado de Sistemas Cliente-Servidor [Fotografía]. GoConqr. <https://www.goconqr.com/mindmap/28352067/sistemas-cliente-servidor>

6.8.3 HTML5

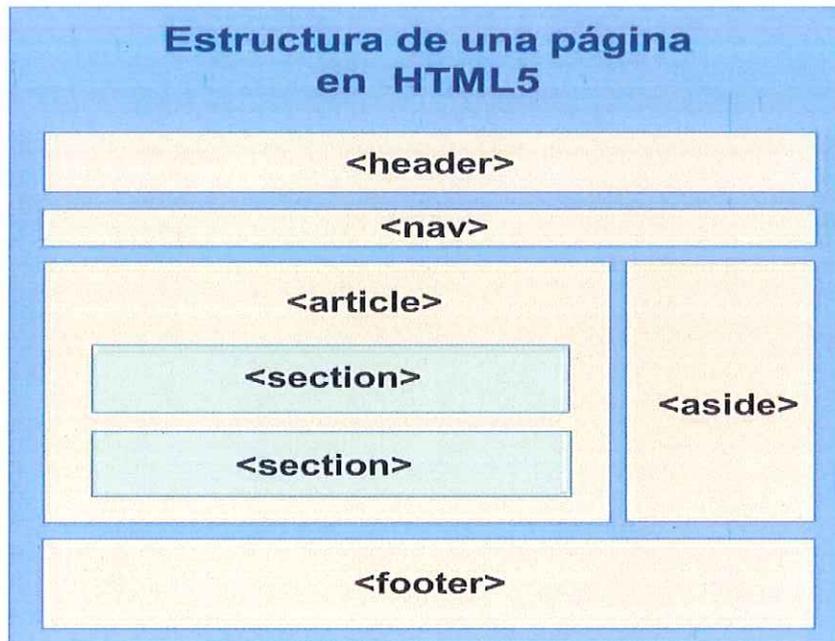
HTML se destaca como una herramienta esencial. Este lenguaje de marcado no debe ser confundido con un lenguaje de programación, ya que su función principal es definir la estructura y el contenido de una página web.

Desarrolladoresweb.org. (2024) indica:

“HTML (por sus siglas en inglés, HyperText Markup Language), es un lenguaje de marcado que nos permite hacer la estructura de una página web. Tenemos que aclarar que HTML no es un lenguaje de programación, pues carece de mecanismos para ejecutar cálculos, repeticiones o condiciones, lo que sí es, es un lenguaje de marcado. Es un lenguaje que le indicará al navegador web qué tipo de elemento es el que está en la estructura de una página web.” (<https://desarrolladoresweb.org/html/que-es-html/>)

Figura 3

Estructura para una página HTML5



Nota. Nueva forma de estructurar una página web en HTML5. Adoptada de Estructura de una página HTML5 [Fotografía]. Web Devs. <https://desarrolladoresweb.org/html5/estructura-de-una-pagina-html5/>

6.8.4 CSS3

En la creación de sitios web atractivos y funcionales, CSS (Cascading Style Sheets) juega un papel fundamental.

HubSpot (2023) menciona:

“CSS son las siglas en inglés para «hojas de estilo en cascada» (Cascading Style Sheets). Básicamente, es un lenguaje que maneja el diseño y presentación de las páginas web, es decir, cómo lucen cuando un usuario las visita. Funciona junto con el lenguaje HTML que se encarga del contenido básico de los sitios.” (<https://blog.hubspot.es/website/que-es-css#que-es>)

Gracias a CSS, los desarrolladores pueden definir estilos, colores, y disposiciones, mejorando significativamente la experiencia del usuario.

6.8.5 PHPWord

Según PHPOffice (s. f.) menciona en su GitHub lo siguiente: PHPWord es una biblioteca escrita completamente en PHP que ofrece un conjunto de clases para la creación y lectura de archivos de documentos en diversos formatos. La versión actual de PHPWord es compatible con Microsoft Office Open XML (OOXML u OpenXML), OASIS Open Document Format for Office Applications (OpenDocument u ODF), Rich Text Format (RTF), HTML y PDF (<https://github.com/PHPOffice/PHPWord>).

PHPWord es ideal para desarrolladores que necesitan generar documentos dinámicamente a través de scripts PHP. Esta biblioteca permite a los desarrolladores crear, modificar y leer documentos en una variedad de formatos, como OOXML, ODF, RTF, HTML y PDF, directamente desde sus aplicaciones PHP. Con PHPWord, los desarrolladores pueden automatizar la generación de documentos, lo que es especialmente útil en entornos donde se requiere la creación masiva de documentos personalizados, como informes, contratos, facturas y más. Además, la capacidad de PHPWord para integrar contenido dinámico basado en datos de la base de datos hace que sea una herramienta poderosa para aplicaciones web y sistemas empresariales. Su naturaleza de código abierto y la licencia LGPL garantizan que los desarrolladores puedan utilizar y modificar la biblioteca según sus necesidades, contribuyendo al desarrollo de soluciones flexibles y escalables.

6.8.6 PHPMailer

PHPMailer (s. f.) indica lo siguiente en su GitHub: Muchos desarrolladores de PHP necesitan enviar correos electrónicos desde su código. La función mail() de PHP no soporta características como cifrado, autenticación, mensajes HTML y archivos adjuntos. El uso adecuado de estándares de correo electrónico puede ser complejo, y la mayoría del código que usa mail() es incorrecto o inseguro.

PHPMailer facilita el envío de correos electrónicos en todas las plataformas sin necesidad de un servidor de correo local, utilizando SMTP para mayor rapidez y seguridad. Es recomendable utilizar bibliotecas especializadas como PHPMailer, Symfony Mailer o Laminas/Mail, en lugar de crear una solución propia (<https://github.com/PHPMailer/PHPMailer>).

6.8.7 Visual Studio Code

Visual Studio Code ha logrado convertirse en una de las herramientas más reconocidas y queridas por la comunidad de desarrolladores. Su versatilidad, eficiencia y facilidad de uso han hecho que miles de programadores adopten este editor de código, destacándolo como una de las mejores opciones en el mercado actual.

Fernán García de Zúñiga (2024) menciona:

“Visual Studio Code, al que conocemos también como VSCode, es un editor de código para programadores gratuito, de código abierto y multiplataforma. Está desarrollado por Microsoft, una compañía con una dilatada experiencia en la creación de IDEs (entornos de desarrollo integrados), que ha conseguido plasmar su larga tradición en el sector para ofrecer una herramienta ligera y práctica que la comunidad ha adoptado en masa.” (<https://www.arsys.es/blog/que-es-visual-studio-code-y-cuales-son-sus-ventajas>)

6.8.8 Scrum Simplificado

La metodología Scrum simplificado adapta los procesos básicos de Scrum tradicional para proyectos individuales o pequeños, eliminando elementos complejos y centrándose en pasos esenciales.

Según Baxter, D., & Turner, N. (2023) menciona lo siguiente:

“Scrum es un método estructurado que permite a los pequeños equipos autoorganizados elegir la mejor manera de realizar su trabajo. Scrum se introdujo por primera vez en una conferencia en 1995 (Schwaber Cita1997), y posteriormente fue refinado y adaptado en la Guía Scrum (Schwaber y Sutherland Cita2013). Fue influenciado por el Manifiesto Ágil (Beck et al. Cita2001). Los doce principios priorizaron la satisfacción del cliente y la bienvenida al cambio, con énfasis en la entrega frecuente de software que funcione. Por lo tanto, el manifiesto no estaba destinado originalmente a ser aplicado directamente a otros ámbitos.” (<https://www.tandfonline.com/doi/full/10.1080/09537287.2021.1997291>)

- Creación del Product Backlog

- Una lista priorizada de todas las tareas necesarias para completar el proyecto.
- Enumera las características o funciones del proyecto.
- Prioriza las tareas según su importancia y relevancia para los objetivos del proyecto.
- Detalla cada tarea en términos claros, desglosándola si es necesario.
- Planificación del Sprint
 - Selección de un conjunto de tareas del backlog que se completarán en un sprint.
 - Define la duración del sprint (normalmente de 1 a 2 semanas).
 - Escoge las tareas que puedas completar en ese tiempo.
 - Estima el esfuerzo necesario para cada tarea y ajusta el alcance del sprint según tu capacidad.
- Ejecución del Sprint
 - Período de trabajo donde te concentras exclusivamente en completar las tareas seleccionadas.
 - Organiza las tareas en un tablero Kanban (columnas: "Por hacer", "En progreso", "Completado").
 - Trabaja en las tareas de manera ordenada, moviéndolas de una columna a otra según avances.
 - Mantén un enfoque constante en las tareas del sprint, evitando interrupciones no planificadas.
- Inspección y Adaptación (Diario personal)
 - Reflexión diaria sobre tu progreso.
 - Pregúntate: ¿Qué hice ayer? ¿Qué haré hoy? ¿Hay obstáculos?
 - Si encuentras problemas, busca soluciones inmediatas o ajusta las prioridades.
 - Duración: 5-10 minutos al día.
- Revisión del Sprint
 - Evaluación de los resultados del sprint.

- Revisa si todas las tareas del sprint están completas.
- Documenta el estado de las tareas que no pudiste terminar y reagrégalas al backlog.
- Evalúa si los objetivos del sprint se cumplieron en función de las metas iniciales.
- Retrospectiva del Sprint
 - Reflexión sobre el proceso de trabajo y cómo mejorarlo.
 - Pregúntate: ¿Qué funcionó bien? ¿Qué puedo mejorar? ¿Qué debería evitar en el próximo sprint?
 - Anota tus observaciones y crea pequeñas acciones para mejorar tu eficiencia.
 - Duración: 15-20 minutos al final de cada sprint.
- Repetición de los Sprints
 - Continuación del ciclo hasta completar el proyecto.
 - Reajusta el backlog según lo aprendido en el sprint anterior.
 - Planifica el próximo sprint y repite el proceso.
- Entrega del Incremento
 - Producto funcional y listo para ser utilizado al final de cada sprint.
 - Asegúrate de que todas las tareas del sprint se integren correctamente.
 - Realiza pruebas básicas para validar que el producto cumple los requisitos establecidos.
 - Resultado esperado: Un avance tangible y funcional del proyecto después de cada sprint.

6.9 Marco Legal

La Ley Orgánica de Protección de Datos Personales de Ecuador, promulgada el 26 de mayo de 2021, tiene como objetivo garantizar el derecho a la protección de datos personales, permitiendo a los individuos acceder y decidir sobre su información personal, así como asegurar su adecuada protección.

Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos: Esta ley regula las transacciones electrónicas y establece la validez jurídica de los documentos y firmas electrónicas. Incluye disposiciones sobre la protección de datos en el comercio electrónico, asegurando la confidencialidad y seguridad de la información transmitida electrónicamente.

7 METODOLOGÍA

7.1 Análisis de la información

Gracias a un banco de preguntas dirigido a la encargada del proceso (secretaria), se pudo realizar el levantamiento de información y definir los requisitos del sistema. Este cuestionario fue esencial para recopilar datos sobre el flujo de trabajo y las necesidades específicas del sistema. Antes de este proyecto, las solicitudes se gestionaban de manera manual y no había un registro formal de las mismas, lo que dificultaba la eficiencia y el seguimiento. La implementación del nuevo sistema web promete mejorar significativamente la organización y gestión de las solicitudes en el Instituto Superior Tecnológico Tena.

El cuestionario permitió identificar los puntos críticos del proceso actual, como la falta de trazabilidad y la demora en la respuesta a las solicitudes. Con esta información, se diseñó un sistema que no solo automatiza la recepción y seguimiento de las solicitudes, sino que también mantiene un registro histórico de todas las interacciones. Esto facilita la recuperación de información y la generación de reportes, mejorando la toma de decisiones administrativas.

Figura 4

Banco de preguntas

¿Qué métodos o herramientas están utilizando actualmente para gestionar los procesos de solicitudes en el Instituto Superior Tecnológico Tena ?
1 respuesta

Manual

¿Qué tan efectivos son estos métodos? ¿Qué carencias presentan?
1 respuesta

No permite dar seguimiento al documento

¿Puede describir los problemas principales que enfrentan actualmente en la gestión de solicitudes en el Instituto Superior Tecnológico Tena?
1 respuesta

No hay registro de los documentos que ingresan físicamente

Nota. Esta es una pequeña muestra del banco de preguntas realizada a la encargada de llegar este proceso. Tomada de Google Forms

7.1.1 Creación del Product Backlog

Tabla 4
Product Backlog

Tareas	Prioridad
Levantamiento de información	Importante
Análisis de la información	Importante
Diseño de la Base de Datos	Importante
Interfaz de Usuario	Importante
Interfaz de Administrador	No se hará en esta fase
Módulo de Registro de Solicitudes	Importante
Validar datos de entrada	Importante
Seguimiento del Proceso	Importante
Historial de Solicitudes	No se hará en esta fase
Notificaciones de Estado	No se hará en esta fase

Nota. Pequeña muestra del listado de tareas realizadas.

7.1 Diseño

7.1.1 Planificación del Sprint

Tabla 5
Planificación del Sprint

Tareas	Tiempo
Levantamiento de información	1 día
Análisis de la información	2 días
Diseño de la Base de Datos	1 día
Maqueta de la interfaz de usuario	2 días
Maqueta de la interfaz de administrador	1 día

Nota. Tareas planificadas para el primer Sprint

7.1.2 Diseño de la Base de Datos

La base de datos del sistema web para la automatización de solicitudes en el Instituto Superior Tecnológico Tena se compone de tablas clave como desolicitudes, historial, dcnotificacion, dccdoctentes y dcestudiantes. Estas tablas desempeñan un papel fundamental en el correcto funcionamiento del sistema, ya que registran las solicitudes, el historial de actividades y gestionan las notificaciones, así como la información de docentes y estudiantes.

desolicitudes: Registra todas las solicitudes realizadas por los estudiantes, incluyendo detalles como fecha de presentación, tipo de solicitud, y estado actual.

historial: Mantiene un registro de todas las actividades y cambios realizados en cada solicitud, proporcionando un historial detallado que facilita el seguimiento y auditoría de los procesos.

dcnotificacion: Gestiona las notificaciones enviadas a los usuarios, informándoles sobre cambios en el estado de sus solicitudes y otras comunicaciones importantes.

dccdoctentes: Almacena información relevante sobre los docentes, como sus datos personales, asignaturas que imparten y horarios.

dcestudiantes: Contiene los datos personales de los estudiantes, incluyendo su información de contacto, cursos en los que están matriculados y historial académico.

Controllers: La carpeta de controladores contiene archivos que se encargan de manejar las solicitudes del usuario, procesar datos y devolver las respuestas adecuadas. Los controladores actúan como intermediarios entre el modelo y la vista, asegurando que la lógica de negocio se aplique correctamente a los datos antes de ser presentados al usuario.

Models: Los modelos representan la estructura de los datos y las reglas de negocio de la aplicación. En esta carpeta, se definen las clases que corresponden a las entidades clave del sistema, como usuarios, solicitudes y documentos, y se establecen las relaciones y restricciones entre ellas.

Plantilla: Esta carpeta contiene plantillas o archivos de diseño que se utilizan para generar la interfaz de usuario. Incluye archivos HTML y otros recursos necesarios para construir las vistas que se mostrarán al usuario final.

Public: La carpeta public contiene archivos estáticos que son accesibles públicamente, como imágenes, hojas de estilo y scripts. Estos archivos se sirven directamente al cliente sin necesidad de procesamiento adicional por parte del servidor.

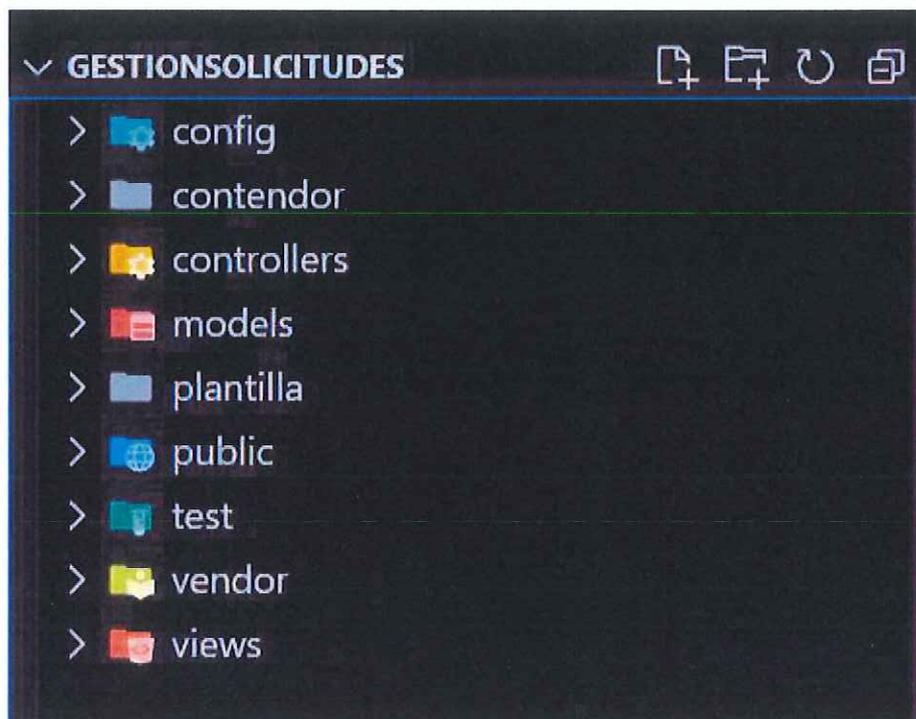
Test: En esta carpeta se encuentran las pruebas automatizadas que aseguran que el código de la aplicación funcione correctamente. Incluye pruebas unitarias, de integración y de aceptación que validan el comportamiento esperado del sistema.

Vendor: La carpeta vendor contiene dependencias externas y bibliotecas que la aplicación necesita para funcionar. Estas dependencias se gestionan a través de gestores de paquetes, como Composer, y proporcionan funcionalidades adicionales que complementan el desarrollo del proyecto.

Views: La carpeta de vistas contiene los archivos responsables de la presentación de los datos al usuario final. Aquí se encuentran las plantillas HTML que se renderizan para mostrar la información de manera visualmente atractiva y accesible.

Figura 6

Estructura de carpetas



Nota. Se muestra como están divididos los archivos del proyecto por carpetas.

7.1.4 Arquitectura del sistema

La arquitectura del sistema para la gestión de solicitudes en el Instituto Superior Tecnológico Tena está diseñada para proporcionar una experiencia fluida y eficiente mediante la interacción coordinada de varios componentes clave:

Interacción del Usuario con el Front-End: Los usuarios (estudiantes, docentes y personal administrativo) interactúan con el sistema a través de una interfaz web intuitiva. Utilizando un navegador web, los usuarios acceden a páginas HTML5 con diseño responsivo, estilizadas con CSS3 y dinámicamente actualizadas con JavaScript. Las acciones del usuario, como enviar una solicitud o consultar el estado de una solicitud, son capturadas por el front-end.

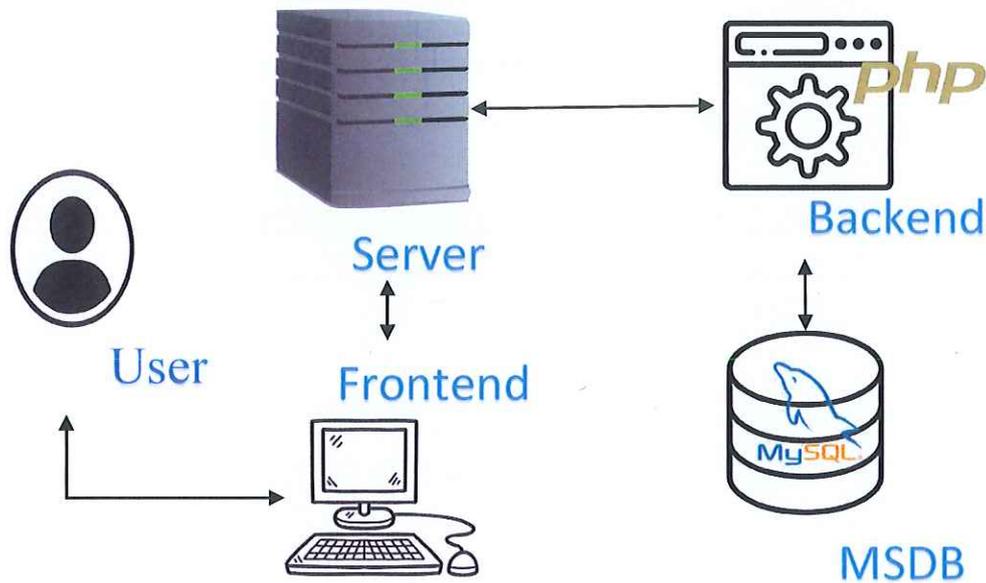
Interacción del Front-End con el Servidor: Las acciones del usuario en el front-end generan solicitudes HTTP (GET, POST, etc.) que son enviadas al servidor web. El servidor web, configurado con Apache (parte del paquete XAMPP), recibe estas solicitudes y las dirige al back-end para su procesamiento.

Interacción del Servidor con el Back-End: En el back-end, el código PHP maneja las solicitudes recibidas del front-end. El back-end contiene la lógica de negocio que determina cómo deben procesarse las solicitudes del usuario. Por ejemplo, si un estudiante envía una nueva solicitud, el back-end valida los datos y ejecuta las funciones necesarias para registrar la solicitud en el sistema.

Interacción del Back-End con la Base de Datos: El back-end se comunica con la base de datos MySQL para almacenar, recuperar y actualizar la información necesaria. Las consultas SQL son generadas y enviadas a la base de datos para realizar operaciones como la inserción de nuevas solicitudes, la actualización de estados y la recuperación de historiales de solicitudes. La base de datos está estructurada en tablas clave que almacenan y organizan la información de manera eficiente y segura.

Interacción de Vuelta al Usuario: Una vez que el back-end ha procesado la solicitud y ha interactuado con la base de datos, genera una respuesta que es enviada de vuelta al servidor web. El servidor web luego envía esta respuesta al front-end, que actualiza la interfaz de usuario en el navegador. De esta manera, el usuario puede ver en tiempo real los resultados de sus acciones, como la confirmación de una nueva solicitud registrada o el estado actualizado de una solicitud existente.

Figura 7
Cliente-Servidor



Nota. Forma en la que se comunica el sistema tras las peticiones del usuario.

7.2 Desarrollo

En esta sección se detallarán las etapas clave del desarrollo del proyecto. Se explicarán los métodos y herramientas utilizados, así como los desafíos encontrados y las soluciones implementadas. A través de un enfoque sistemático y colaborativo, se han llevado a cabo las actividades necesarias para diseñar, construir y probar el sistema, garantizando su funcionalidad y eficiencia. A continuación, se describen los pasos realizados durante el desarrollo del proyecto, destacando los aspectos técnicos y estratégicos que han contribuido a su éxito.

7.2.1 Ejecución del Sprint

Tabla 6
Ejecución del Sprint

Por hacer	En proceso	Completadas
Levantamiento de información		X
Análisis de la información		X

Diseño de la Base de Datos		X
Diseño de la interfaz de usuario		X
Diseño de la interfaz de administrador		X

Nota. Lista de tareas realizada en el primer Sprint

7.2.1.1 Interfaces

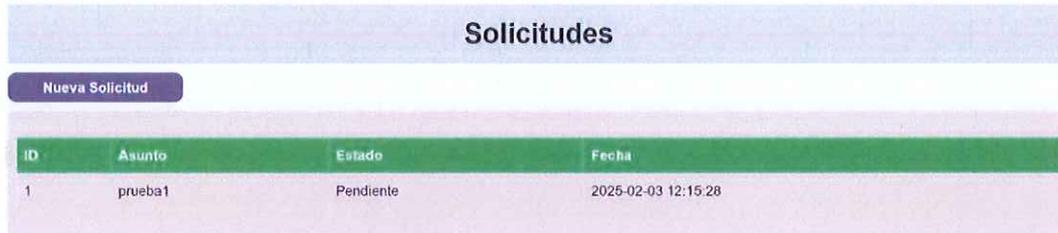
El desarrollo de las interfaces de usuario para el sistema de gestión de solicitudes en el Instituto Superior Tecnológico Tena se ha llevado a cabo con un enfoque centrado en la usabilidad y la accesibilidad, garantizando que tanto los usuarios técnicos como los no técnicos puedan interactuar fácilmente con el sistema.

Desarrollo Front-End: El desarrollo de las interfaces se realizó con tecnologías web modernas como HTML5, CSS3. El uso de HTML5 permitió estructurar las páginas de manera semántica, mejorando la accesibilidad y el SEO. CSS3 se utilizó para estilizar las páginas, y que sea amigable con el usuario final.

Pruebas y Mejoras Continuas: Se realizaron pruebas exhaustivas de las interfaces para asegurar que funcionaran correctamente en diferentes navegadores y dispositivos.

Figura 8

Interfaz del usuario

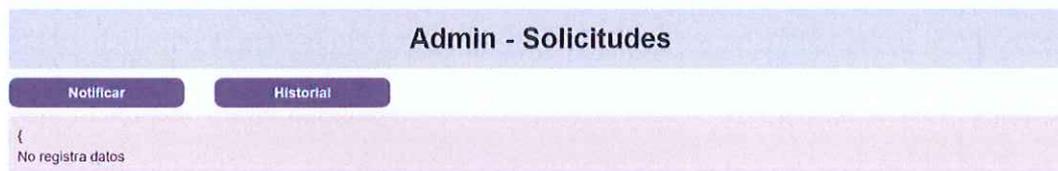


The screenshot shows a user interface for 'Solicitudes'. At the top, there is a blue header with the title 'Solicitudes'. Below the header is a purple button labeled 'Nueva Solicitud'. Underneath is a table with a green header and a light purple body. The table has four columns: 'ID', 'Asunto', 'Estado', and 'Fecha'. There is one row of data with the following values: ID: 1, Asunto: prueba1, Estado: Pendiente, Fecha: 2025-02-03 12:15:28.

ID	Asunto	Estado	Fecha
1	prueba1	Pendiente	2025-02-03 12:15:28

Figura 9

Interfaz del administrador



The screenshot shows an administrator interface for 'Admin - Solicitudes'. At the top, there is a blue header with the title 'Admin - Solicitudes'. Below the header are two purple buttons: 'Notificar' and 'Historial'. Underneath is a light purple box containing a curly brace '{' and the text 'No registra datos'.

7.2.1.2 Codificación

La codificación del proyecto para la gestión de solicitudes en el Instituto Superior Tecnológico Tena se ha llevado a cabo siguiendo un enfoque estructurado y metódico para garantizar la calidad, eficiencia y mantenibilidad del código. A continuación, se describen los aspectos clave del proceso de codificación:

Estructura del Proyecto: El proyecto está organizado en una estructura de carpetas clara y lógica, que incluye directorios para controladores, modelos, vistas, configuraciones, archivos públicos y pruebas. Esta organización facilita la navegación y gestión del código, permitiendo a los desarrolladores encontrar y modificar fácilmente los componentes necesarios.

Lenguajes y Tecnologías Utilizadas: La codificación del front-end se realizó utilizando HTML5, CSS3. Para el back-end, se utilizó PHP como lenguaje principal, debido a su versatilidad y capacidad para manejar la lógica del servidor. La base de

datos fue implementada en MySQL, garantizando un almacenamiento de datos eficiente y seguro.

Desarrollo de la Lógica del Servidor: La lógica del servidor se codificó en PHP, donde se desarrollaron múltiples módulos para manejar diferentes funcionalidades del sistema. Estos módulos incluyen la gestión de solicitudes, el seguimiento de estados y la generación de notificaciones. Cada módulo fue diseñado siguiendo principios de programación orientada a objetos (POO), lo que facilita la reutilización y mantenimiento del código.

Interacción con la Base de Datos: Para la interacción con la base de datos MySQL, se utilizaron consultas SQL para realizar operaciones de inserción, actualización, eliminación y recuperación de datos. Se implementaron prácticas de seguridad, como la preparación de declaraciones SQL, para prevenir ataques de inyección SQL y garantizar la integridad de los datos.

Desarrollo de Interfaces de Usuario: Las interfaces de usuario se desarrollaron utilizando tecnologías web modernas, asegurando una experiencia de usuario fluida y accesible.

Figura 10

Crear nueva solicitud

```
1 </php
2
3 include_once '../config/database.php';
4
5 class solicitud{
6     private $con;
7     private $tabla_soli = "dcsolicitudes";
8     private $tabla_noti = "dnotificaciones";
9     private $tabla_hitorial = "dchitorial";
10
11     public $nombre;
12     public $cedula;
13     public $self;
14     public $correo;
15     public $docente;
16     public $asunto;
17     public $descrip;
18     public $carrera;
19     public $id_estudiante;
20     public $ndocente;
21     public $adocente;
22
23     public function __construct($db){ //constructor que recibe la conexion a la db
24         $this->con = $db;
25     }
26     public function nuevaSolicitud(){
27         try {
28             $query = "SELECT iddocentes FROM ddocentes where doc_primernombre = :ndocente and doc_primerapellido = :adocente";
29             $stmt = $this->con->prepare($query);
30             $stmt->bindParam(":ndocente", $this->ndocente);
31             $stmt->bindParam(":adocente", $this->adocente);
32             $stmt->execute();
33             $result = $stmt->fetch(PDO::FETCH_ASSOC);
34             if (!$result) {
35                 echo 'El nombre del docente esta mal escrito';
36                 return false;
37             }else{
38
39                 $id_dcente = $result['iddocentes'];//obtenemos el id del docente
40
41                 // inserta los datos a la db {asunto}-es el nombre del campo de la tabla y {asunto} es un marcador de su posicion
42                 $query = "INSERT INTO ".$this->tabla_soli." VALUES (null,:id_estudiante,:id_docente,:asunto,:descrip,1,null,null)";
43                 $stmt = $this->con->prepare($query);
44
45                 //htmlspecialchars->limpia los datos para evitar inyecciones sql y convierte los caracteres especiales en entidades HTML
46                 //strip_tags-remueve las entidades HTML
47                 $this->asunto = htmlspecialchars(strip_tags($this->asunto));
48                 $this->descrip = htmlspecialchars(strip_tags($this->descrip));
49
50                 //bindParam-> aniza un parámetro de la consulta SQL con una variable de PHP
51                 $stmt->bindParam(":asunto", $this->asunto);
52                 $stmt->bindParam(":descrip", $this->descrip);
53                 $stmt->bindParam(":id_estudiante", $this->id_estudiante);
54                 $stmt->bindParam(":id_docente", $id_dcente);
55
56                 if ($stmt->execute()) {
57                     return true;
58                 }
59                 return false;
60             }
61         } catch (PDOException $e) {
62             echo "Error: ". $e->getMessage();
63             return false;
64         } finally{
65             $this->con = null;
66         }
67     }
68 }
```

Nota. El código muestra el método para crear una solicitud.

7.2.1.3 Funcionamiento al Crear una Solicitud en el Sistema

Navegación a la Sección de Solicitudes: Una vez autenticado, el usuario navega a la sección de "Solicitudes" en el menú principal de la interfaz web. Esta sección está diseñada para ser intuitiva y accesible, facilitando la navegación.

Formulario de Nueva Solicitud: En la sección de solicitudes, el usuario selecciona la opción "Nueva Solicitud". Esto despliega un formulario en el cual se deben completar campos específicos, que fueron detallados en la figura 12 como, por ejemplo:

- Asunto de la solicitud.
- Descripción detallada de la solicitud.
- Adjuntar documento.

Ingresar Datos y Adjuntar Documento: El usuario completa los campos obligatorios del formulario, proporcionando toda la información relevante para su solicitud. Luego, debe descargar el documento en formato Word haciendo click en el botón correspondiente, imprimirlo y firmarlo manualmente, ya que el sistema no cuenta con firma electrónica. Una vez firmado, el usuario debe escanear el documento y guardarlo en un archivo PDF. Posteriormente, el usuario adjunta este archivo PDF en el campo correspondiente y envía la solicitud para que se registre en la base de datos.

Revisión y Envío: Una vez completado el formulario, el usuario revisa la información ingresada para asegurarse de que todo esté correcto. Después de la revisión, el usuario hace clic en el botón "Enviar". El sistema valida los datos ingresados y muestra un mensaje de confirmación.

Registro de la Solicitud: Al enviar la solicitud, el sistema la registra en la base de datos MySQL, asignándole un identificador único. La solicitud queda registrada con detalles como la fecha y hora de creación, el estado inicial y el usuario que la realizó.

Seguimiento de la Solicitud: Cuando el usuario ingresa a la sección de solicitudes, puede ver todas sus solicitudes y su estado actual (en proceso, aprobada, rechazada), así como el historial de actividades relacionadas con cada solicitud.

Este flujo de trabajo asegura que las solicitudes estudiantiles se gestionen de manera eficiente, transparente y accesible, optimizando los procesos administrativos del Instituto Superior Tecnológico Tena.

7.2.1.4 Pruebas de Compatibilidad en Navegadores Web

Para asegurar la compatibilidad y el correcto funcionamiento del sistema web de gestión de solicitudes en el Instituto Superior Tecnológico Tena, se realizaron pruebas exhaustivas en los principales navegadores web: Google Chrome, Mozilla Firefox y Microsoft Edge. Estas pruebas fueron fundamentales para identificar posibles problemas de compatibilidad y garantizar una experiencia de usuario consistente en todos los entornos.

Durante las pruebas en Google Chrome y Microsoft Edge, el sistema funcionó sin inconvenientes, presentando una interfaz fluida. Sin embargo, al ejecutar las pruebas en Mozilla Firefox, se identificó un problema específico relacionado con un atributo de CSS utilizado para el tamaño de los campos de texto (textarea). El atributo en cuestión, **field-sizing: content**, no era compatible con Firefox, lo que resultó en un comportamiento inesperado en la presentación de los campos de texto.

Afortunadamente, este problema no afectaba significativamente el uso del sistema por parte de los usuarios.

Figura 11

Interfaz del usuario - Crear una solicitud

The image shows a web form titled "Nueva Solicitud" (New Request) with a purple header. Below the header is a dark purple button labeled "Atras" (Back). The form itself is white and contains several input fields with example text:

- Nombre:** Ejm. Sr. David Alexander Carrasco Candio
- Cédula:** Ejm. 1501600085
- Telefono:** Ejm. 0979123654
- Correo:** Ejm. david@gmail.com
- Carrera:** Ejm. Tecnología Superior en Desarrollo de Software
- Docente:** Ejm. Ing. Salomon Quilumba
- Asunto:** Ejm. Segunda Matricula
- Descripción:** Ejm. me dirijo para saludarle y desearle exitos en sus funciones, además muy respetuosamente solicito a usted, la inscripción en la Unidad de Integración Curricular mediante modalidad Trabajo de Integración Curricular, con el tema:
- Documento:** Elegir archivo. No se ha seleccionado ningún archivo

At the bottom of the form are two dark purple buttons: "Descargar" (Download) and "Enviar" (Send).

Nota. El problema detectado en el navegador Firefox, es en el campo de Descripción.

7.2.2 Inspección y Adaptación (Diario personal)

A lo largo del proyecto se implementó un riguroso proceso de inspección y adaptación mediante el uso de un diario personal. Diariamente, se tomó entre 5 y 10 minutos a responder las preguntas: “¿Qué hice ayer?”, “¿Qué haré hoy?” y “¿Existen obstáculos?”. Esta práctica permitió no solo identificar logros y desafíos, sino también efectuar correcciones oportunas en los resultados obtenidos. Gracias a este seguimiento continuo, se detectaron desviaciones y se aplicaron medidas correctivas de manera

inmediata, garantizando que el progreso se mantuviera alineado a los objetivos establecidos. Este mecanismo de autoevaluación y ajuste resultó fundamental para optimizar la eficiencia del proyecto y elevar la calidad final del entregable.

7.2.3 Revisión del Sprint

A lo largo del proyecto, la Revisión del Sprint se constituyó como un proceso esencial de validación y ajuste continuo. Al finalizar cada sprint, se realizó una evaluación exhaustiva de los resultados obtenidos en comparación con los objetivos establecidos inicialmente. Durante este proceso, se verificó el cumplimiento de las tareas seleccionadas, identificando claramente aquellas que se completaron y documentando las que quedaron pendientes para reincorporarlas al Product Backlog. La retroalimentación obtenida a partir de la revisión permitió detectar desviaciones, implementar ajustes inmediatos y afinar la planificación de los sprints subsiguientes. Este mecanismo de evaluación final de cada sprint fue crucial para garantizar que el proyecto avanzara de manera coherente y orientada hacia el logro de sus metas, consolidando un enfoque de mejora continua durante todo el desarrollo.

7.2.4 Retrospectiva del Sprint

A lo largo del proyecto, la Retrospectiva del Sprint se erigió como un pilar fundamental para el aprendizaje y la mejora continua. Al concluir cada sprint, se organizaba una sesión de reflexión en la que se analizaban de forma objetiva y crítica tanto los aciertos como los desafíos enfrentados en el ciclo de trabajo. Durante estas reuniones, se discutían abiertamente las estrategias implementadas, identificando prácticas positivas que debían consolidarse y áreas en las que se requerían ajustes o cambios. Las conclusiones extraídas se tradujeron en acciones concretas que se integraron al siguiente ciclo, permitiendo así una adaptación constante y la optimización del proceso.

7.2.5 Repetición de los Sprints

Se estableció un ciclo iterativo basado en la repetición de los sprints para propiciar un avance continuo y una mejora constante. Cada sprint se iniciaba evaluando los resultados del ciclo anterior, lo que permitía ajustar el Product Backlog en función de los aprendizajes y necesidades identificadas. Esta dinámica de reutilización y

renovación de los sprints facilitó la incorporación de nuevas tareas y el refinamiento de las ya existentes, asegurando que la priorización y la planificación se adaptaran de manera ágil a los desafíos emergentes.

7.2.6 Entrega del Incremento

La Entrega del Incremento se constituyó como la culminación tangible de cada ciclo de trabajo. Al finalizar cada sprint, se integraron todas las tareas completadas y se sometió el producto a pruebas básicas para validar su funcionalidad y conformidad con los requisitos establecidos. Este proceso garantizó que cada incremento fuera una versión operativa, robusta y de calidad, lista para ser utilizada o demostrada ante los interesados.

8 RESULTADOS

En esta sección se presentarán los resultados obtenidos a lo largo del desarrollo del proyecto. Se describirán los logros alcanzados, las mejoras implementadas y el impacto generado en los procesos administrativos del Instituto.

8.1 Análisis de la información

El proceso de levantamiento de información, llevado a cabo mediante un exhaustivo cuestionario dirigido a la encargada del proceso –quien en este caso ocupa funciones de secretaria– permitió recopilar datos fundamentales sobre el flujo de trabajo y las necesidades específicas del sistema. Gracias a esta herramienta, se identificaron con claridad las deficiencias del manejo manual de las solicitudes, entre las que se destacan la falta de un registro formal, la insuficiente trazabilidad y las demoras en la respuesta a las mismas.

Con base en la información recabada, se diseñó un nuevo sistema web orientado a automatizar la recepción, seguimiento y registro histórico de las solicitudes en el Instituto Superior Tecnológico Tena. Este sistema no solo optimiza la eficiencia operativa, sino que también mejora la generación de informes y reportes, facilitando la toma de decisiones a nivel administrativo. En síntesis, la integración del cuestionario fue esencial para sentar las bases de un proyecto que promete transformar significativamente la organización y la gestión interna de las solicitudes, elevando así los estándares de eficiencia y servicio institucional.

8.2 Maquetación

La fase de maquetación constituyó un pilar fundamental en el desarrollo del sistema web para la automatización de solicitudes en el Instituto Superior Tecnológico Tena. Durante la planificación del sprint, se definieron de manera precisa las tareas y los tiempos asignados para cada actividad: desde el levantamiento y análisis de la información hasta la elaboración de maquetas para la interfaz de usuario y para el administrador. Esta planificación permitió establecer un cronograma realista y ordenado, que facilitó la coordinación entre los distintos miembros del equipo.

Por otro lado, el diseño de la base de datos se estructuró mediante la identificación y creación de tablas clave, tales como `dsolicitudes`, `historial`,

denotificación, docentes y estudiantes. La interrelación entre estas tablas asegura la integridad, trazabilidad y confiabilidad de la información, permitiendo el registro y seguimiento de las solicitudes. La inclusión de relaciones eficientes y la normalización de la base de datos se convirtieron en elementos esenciales para optimizar el rendimiento y la escalabilidad del sistema.

Además, la arquitectura del proyecto se organizó en una estructura de carpetas claramente definida, donde cada directorio: Config, Contenedor, Controllers, Models, Views, Plantilla, Public, Test y Vendor, cumple funciones específicas que facilitan el mantenimiento, modularidad y la extensibilidad del sistema. Esta división estructural no solo mejora la gestión interna del código, sino que también agiliza la incorporación de futuras mejoras.

Finalmente, la arquitectura del sistema se diseñó para garantizar una comunicación fluida entre el front-end, el servidor web y el back-end. Los usuarios interactúan mediante una interfaz web atractiva, mientras las solicitudes se procesan eficientemente a través de servicios PHP y se almacenan en una base de datos. Este enfoque integral permite a la institución transformar procesos manuales en procedimientos automatizados, mejorando significativamente la eficiencia operativa y la calidad de la toma de decisiones.

En conclusión, la meticulosa fase de maquetación, que abarcó la planificación del sprint, el diseño estructurado de la base de datos, una organización lógica de carpetas y una arquitectura de sistema orientada a la eficiencia, proporcionó las bases sólidas y flexibles para el desarrollo del sistema, asegurando que las necesidades institucionales sean atendidas de manera oportuna y eficaz.

8.3 Desarrollo

Durante la fase de desarrollo del sistema, se llevaron a cabo diversas actividades clave orientadas a construir un producto funcional, eficiente y de calidad. En esta etapa se implementaron métodos y herramientas modernas, se realizaron pruebas exhaustivas y se adoptaron un enfoque iterativo para refinar continuamente el producto. La metodología empleada se centró en la ejecución de Sprints, permitiendo la

coordinación de actividades en períodos cortos y la integración progresiva de funcionalidades.

- Ejecución del Sprint

El proceso se planificó y ejecutó mediante un Sprint, donde se definieron las siguientes tareas y sus respectivos tiempos de ejecución:

Levantamiento de información: 1 día

Análisis de la información: 2 días

Diseño de la Base de Datos: 1 día

Maqueta de la interfaz de usuario: 2 días

Maqueta de la interfaz de administrador: 1 día

Estas tareas se distribuyeron en tres categorías: Por hacer, En proceso y Completadas, lo que permitió tener una visión clara del progreso y facilitar la gestión de incidencias y ajustes durante el Sprint.

- Interfaces

Durante esta fase, se diseñaron las interfaces de usuario y de administrador. Se desarrollaron prototipos y se crearon wireframes que fueron posteriormente transformados en maquetas funcionales utilizando HTML5, CSS3. La interfaz se concibió para ser intuitiva, garantizando una experiencia de usuario satisfactoria.

- Codificación

La etapa de codificación se implementó siguiendo las mejores prácticas de programación y utilizando un enfoque modular. El código se organizó en capas: se separó la lógica de negocio, el acceso a datos y la presentación. El uso de patrones de diseño, como el Modelo-Vista-Controlador (MVC), permitió que el sistema fuera escalable y mantenible. Además, se integraron librerías de terceros mediante Composer para optimizar funciones específicas, por ejemplo, el manejo de plantillas de Word y el envío de correos electrónicos.

- Funcionamiento al Crear una Solicitud en el Sistema

El desarrollo incluyó la implementación de la funcionalidad para que los estudiantes pudieran crear solicitudes de manera automatizada. Se desarrollaron formularios web que permitieron capturar datos, realizar validaciones y enviar la información al back-end. Una vez recibida la solicitud, el sistema actualizaba la base de datos y, en caso de acción de aprobación o no, enviaba notificaciones electrónicas. Esta funcionalidad fue probada en distintos escenarios para garantizar un funcionamiento correcto y eficiente.

- Pruebas de Compatibilidad en Navegadores Web

Parte esencial del desarrollo fue verificar que el sistema respondiera de manera óptima en los principales navegadores web (Chrome, Firefox, Microsoft Edge). Se realizaron pruebas de compatibilidad para identificar y corregir problemas de visualización y funcionamiento, asegurando que la experiencia de usuario fuera consistente e ininterrumpida. Durante estas pruebas, se detectó que en el navegador Firefox se presentó un leve inconveniente debido al uso del atributo CSS moderno `field-sizing: content`. Aunque este detalle generó una ligera diferencia en la forma de procesar los estilos, no se evidenció ninguna incomodidad significativa para el usuario final, ni se alteró la funcionalidad o la usabilidad del sistema. En definitiva, los esfuerzos de compatibilidad confirmaron que la interfaz se muestra de forma uniforme y sin incidencias relevantes, independientemente del navegador utilizado.

- Inspección y Adaptación (Diario Personal)

Durante el Sprint, se documentaron diariamente los avances, dificultades y soluciones en un diario personal. Esta herramienta de inspección y adaptación permitió identificar áreas de mejora de forma oportuna y realizar ajustes en tiempo real, fomentando además la retroalimentación y el aprendizaje continuo dentro del equipo.

- Revisión del Sprint

Al finalizar cada Sprint, se realizó una revisión del trabajo realizado. En esta reunión se corroboraba el cumplimiento de las metas establecidas, se discutían las

dificultades surgidas durante la ejecución de las tareas y se planificaban las acciones correctivas para el siguiente ciclo de trabajo.

- Retrospectiva del Sprint

La retrospectiva del Sprint fue una actividad crucial en la que se evaluó el proceso de desarrollo. Se analizaron los éxitos y las áreas de oportunidad, permitiendo optimizar los métodos de trabajo y ajustar la planificación futura. Esta sesión contribuyó a mejorar la comunicación entre los miembros del equipo y a definir estrategias para incrementar la eficiencia en los próximos Sprints.

- Repetición de los Sprints

El desarrollo se estructuró en ciclos iterativos (Sprints), en los que se repetirían los procesos de planificación, ejecución, revisión y adaptación. Esta metodología ágil permitió que el sistema evolucionara de manera progresiva y que se integraran nuevas funcionalidades basadas en el feedback obtenido, garantizando así una mejora continua del producto.

- Entrega del Incremento

Finalmente, al culminar cada ciclo, se entregó un incremento funcional del sistema. Esta entrega incluyó la implementación de nuevas características, la corrección de errores identificados y la actualización de la documentación del proyecto. El incremento entregado fue validado con pruebas de integración y de aceptación, asegurando que cada versión del sistema cumpliera con los estándares de calidad requeridos.

9 CONCLUSIONES

El desarrollo del sistema web para la automatización de solicitudes en el Instituto Superior Tecnológico Tena ha permitido transformar un proceso previamente manual y fragmentado en un flujo digital integrado, eficiente y trazable. La aplicación de metodologías ágiles, en concreto la estructuración en Sprints, facilitó la planificación, ejecución y revisión continua de las tareas, lo que se tradujo en una mejora sustancial de la eficiencia operativa y de la capacidad de respuesta del sistema.

Mediante el levantamiento de información y el análisis detallado de los procesos existentes, se consiguió identificar los puntos críticos, tales como la falta de un registro formal, la carencia de trazabilidad y las demoras en la respuesta a las solicitudes. Con estos datos, se definieron los requisitos del sistema, lo que permitió diseñar una base de datos relacional y una arquitectura en capas que separa de manera clara la lógica de negocio, la presentación y el acceso a datos. Esta estructuración no solo mejora el mantenimiento y la escalabilidad del sistema, sino que además sienta las bases para futuras mejoras.

El diseño de la interfaz de usuario y la interfaz de administrador fue desarrollado con un enfoque moderno y responsivo, empleando tecnologías web actuales (HTML5, CSS3 y JavaScript) y garantizando una experiencia de usuario uniforme. Durante la fase de pruebas, se verificó la compatibilidad del sistema en los principales navegadores (Chrome, Firefox y Microsoft Edge). Aunque en Firefox se detectó un ligero inconveniente con el atributo CSS `field-sizing: content`, este problema no generó incomodidad significativa en el usuario ni alteró el funcionamiento global del sistema.

Por otro lado, el uso de librerías de terceros, como PHPMailer y PHPWord, demostró ser una herramienta clave para la integración de funcionalidades adicionales, tales como el envío de correos electrónicos y la generación automatizada de documentos. La implementación de estas herramientas, combinada con la correcta gestión de sesiones y la validación de datos, reforzó la seguridad y la robustez del sistema.

En conclusión, el proyecto no solo ha logrado automatizar y optimizar el proceso de gestión de solicitudes, sino que también ha contribuido a la mejora en la toma de decisiones administrativas mediante la generación de reportes y auditorías. La metodología adoptada, basada en la iteración y el enfoque colaborativo, resultó esencial para identificar y resolver las deficiencias del sistema anterior, proporcionando una solución integral que alinea las necesidades institucionales con estándares modernos de desarrollo y gestión de datos. Estas conclusiones resaltan la importancia de aplicar un enfoque sistemático y ágil en la transformación digital de procesos, ofreciendo una base sólida para futuras mejoras en el desempeño y la eficiencia operativa del Instituto.

10 RECOMENDACIONES

Se recomienda que futuros desarrollos de sistemas similares consideren, desde las primeras fases del proyecto, la adopción de una arquitectura en capas, que permita la separación clara de la lógica de negocio, la gestión de datos y la presentación. Esta estructura modular facilita la mantenibilidad y escalabilidad del sistema, permitiendo, además, la integración de nuevas funcionalidades sin afectar la estabilidad general de la aplicación.

Asimismo, es esencial aplicar rigurosas medidas de seguridad. Se aconseja la validación y saneamiento de todos los datos recibidos, el uso de declaraciones preparadas (como las proporcionadas por PDO) para prevenir ataques de inyección SQL, y la correcta implementación y gestión de las sesiones de usuario para evitar accesos no autorizados. El manejo seguro de la subida de archivos y la administración cuidadosa de las credenciales de servicios externos (como servidores SMTP) también deben ser considerados como prioridades en el desarrollo.

La adopción de metodologías ágiles, con sprints bien definidos y revisiones iterativas, resultó crucial para la identificación y solución temprana de problemas. Por ello, se recomienda continuar utilizando este enfoque, incluyendo inspecciones diarias y retrospectivas de sprints, a fin de ajustar continuamente el desarrollo hacia la optimización del producto final. La integración de un diario de inspección y adaptación en cada ciclo permite que el equipo se beneficie del aprendizaje continuo y mejore la coordinación interna.

En cuanto a la calidad de la experiencia de usuario, se recomienda realizar pruebas de compatibilidad en múltiples navegadores y dispositivos. Aunque se detectaron mínimos inconvenientes, como el problema con el atributo CSS moderno `field-sizing: content` en Firefox, dichos detalles no afectaron de forma significativa la usabilidad general. Sin embargo, es aconsejable monitorear de cerca estos aspectos para lograr una interfaz completamente uniforme y evitar cualquier posible incidencia en escenarios reales.

Por otra parte, se destaca la conveniencia de aprovechar librerías de terceros robustas y comprobadas, tal como PHPMailer y PHPWord, para implementar

funcionalidades complejas como el envío de correos electrónicos y la generación dinámica de documentos. La utilización de estas herramientas no solo agiliza el proceso de desarrollo, sino que también incrementa la fiabilidad del sistema, permitiendo a su vez que se puedan aplicar actualizaciones de seguridad de manera centralizada y efectiva.

Finalmente, se recomienda que se establezca una sistemática de documentación técnica y de usuario, que detalle tanto la estructura del proyecto (definida en carpetas específicas como Config, Controllers, Models, Views, etc.) como los procedimientos operativos y las decisiones de implementación. Esto facilitará futuras tareas de mantenimiento, ampliación funcional y auditoría, garantizando que el sistema pueda responder a las necesidades cambiantes de la institución.

11 BIBLIOGRAFÍA

- Arenas, M. P. (2021, 24 enero). GoConQR - Sistemas Cliente-Servidor. GoConqr. <https://www.goconqr.com/mindmap/28352067/sistemas-cliente-servidor>
- Asenjo Mestanza, J. F. (2020). Sistema web para el mejoramiento de la gestión documental en SERPAR,2020. https://repositorio.ucv.edu.pe/bitstream/handle/20.500.12692/59874/Asenjo_MJF-SD.pdf?sequence=1&isAllowed=y
- Baxter, D., & Turner, N. (2023). Why Scrum works in new product development: the role of social capital in managing complexity. *Production Planning & Control*, 34(13), 1248-1260.
- B, G., & B, G. (2024, 3 septiembre). ¿Qué es MySQL? Explicación detallada para principiantes. Tutoriales Hostinger. https://www.hostinger.mx/tutoriales/que-es-mysql/#%C2%BFQue_es_MySQL
- Castillo Peña, G. E. (2018). Implementación de un sistema web de gestión documentaria en la municipalidad distrital de Pararin-provincia Recuay-departamento de Ancash; 2017. https://repositorio.uladech.edu.pe/bitstream/handle/20.500.13032/2518/GESTION_DOCUMENTARIA_IMPLEMENTACION_CASTILLO_PENA_GERARDO_EDINSON.pdf?sequence=1&isAllowed=y
- Click-It. (s. f.). Aplicación web (Web App) vs Sitio web: cómo elegir la correcta | Click-IT | Servicios tecnológicos y de consultoría. <https://click-it.es/aplicacion-web-web-app-vs-sitio-web-como-elegir-la-correcta/>
- Desarrolladoresweb.org.(2024, 11 octubre). ▷ Tipos de lenguajes de programación: una guía exhaustiva. Web Devs. <https://desarrolladoresweb.org/programacion/tipos-de-lenguajes-de-programacion-una-guia-exhaustiva/>
- Desarrolladoresweb.org.(2024a, junio 28, A). ▷ ¿Qué es HTML Para qué sirve y cómo funciona? Web Devs. <https://desarrolladoresweb.org/html/que-es-html/>
- Desarrolladoresweb.org.(2023, 8 noviembre, B). ▷ Estructura de una página HTML5. Web Devs. <https://desarrolladoresweb.org/html5/estructura-de-una-pagina-html5/>

- Fernán García de Zúñiga (2024). ¿Qué es Visual Studio Code y cuáles son sus ventajas?, Arsys. <https://www.arsys.es/blog/que-es-visual-studio-code-y-cuales-son-sus-ventajas>
- Introducción al CSS: Qué es, para qué sirve y otras 10 preguntas frecuentes. (2023, 28 julio). HubSpot. <https://blog.hubspot.es/website/que-es-css#que-es>
- Juan de Assembler Institute. (2023, 4 octubre). ¿Qué es PHP y para qué sirve? - Assembler Institute. Assembler Institute. <https://assemblerinstitute.com/blog/que-es-php/>
- NormasISO.org(2024) Recuperado de NormasISO.org: "Norma ISO 15489" en la categoría Gestión y servicios. <https://normasiso.org/norma-iso-15489/>
- PHPMailer. (s. f.). GitHub B- PHPMailer/PHPMailer: The classic email sending library for PHP. GitHub. <https://github.com/PHPMailer/PHPMailer>
- PHPOffice. (s. f.). GitHub A- PHPOffice/PHPWord: A pure PHP library for reading and writing word processing documents. GitHub. <https://github.com/PHPOffice/PHPWord>
- ¿Qué es un sitio web? Todo lo que necesita saber. (2024, 16 enero). SoftwareLab. <https://softwarelab.org/es/blog/que-es-un-sitio-web/>
- Universidad Europea. (2022, 22 marzo). ¿Para qué sirve un gestor de base de datos? <https://universidadeuropea.com/blog/para-que-sirve-gestor-base-datos/#que-es-un-sistema-gestor-de-base-de-datos>

12 ANEXOS

Anexo 1

Banco de preguntas para el levantamiento de información

¿Qué métodos o herramientas están utilizando actualmente para gestionar los procesos de solicitudes en el Instituto Superior Tecnológico Tena ?

1 respuesta

Manual

¿Qué tan efectivos son estos métodos? ¿Qué carencias presentan?

1 respuesta

No permite dar seguimiento al documento

¿Puede describir los problemas principales que enfrentan actualmente en la gestión de solicitudes en el Instituto Superior Tecnológico Tena?

1 respuesta

No hay registro de los documentos que ingresan físicamente

¿Por qué es importante implementar este sistema en el Instituto? Por ejemplo, ¿mejoraría la eficiencia, reduciría costos o simplificaría procesos?

1 respuesta

Mejoraría el tiempo de respuesta a las solicitudes y se podría dar seguimiento a cada documento.

¿Qué impacto tendrá la solución en el corto y largo plazo para el Instituto y sus usuarios?

1 respuesta

Positivo y eficiente

¿Que tipo de solicitudes nomas manejan en el Instituto Superior Tecnológico Tena?

1 respuesta

Solicitudes de certificados de notas, solicitudes de certificados de matrícula, solicitudes de cambio de carrera, solicitudes de récords académico, solicitudes de mallas curriculares, solicitudes de autorización para ejecutar y tema de tic en el instituto, etc, etc.

Anexo 2

Solicitud realizada para la implementación

Tena, 20 de febrero de 2025

Mgtr. Lorena Pilar Yáñez Palacios
RECTORA DEL INSTITUTO SUPERIOR TECNOLÓGICO TENA
Presente. -

De mi consideración:

Reciba un cordial saludo y a la vez deseándole éxitos en sus labores diarias y profesionales, le saluda el Sr. **Carrasco Cando David Alexander** con CI: 1501200040, estudiante de la carrera de **Tecnología Superior en Desarrollo de Software** de esta institución, en la cual me encuentro en el proceso de titulación con el tema: "*Desarrollo de un Sistema Web para la Automatización de Solicitudes en el Instituto Superior Tecnológico Tena*", por ende me dirijo a usted para solicitar la AUTORIZACIÓN para la implementación del sistema mencionado anteriormente en la plataforma SIAGE.

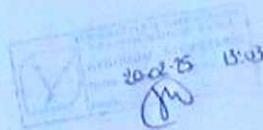
Este pedido lo realizo con la finalidad de que me permita desarrollar el ultimo pasa de mi proyecto de titulación. A la vez solicitarle la creación de un correo electrónico institucional que será de uso general por parte de los estudiantes y del mismo se realizarán las solicitudes hacia los docentes, el cual puedo ser el siguiente: "estudiantes.solicitud@est.itstena.edu.ec".

Seguro de contar con una respuesta favorable, anticipo mis agradecimientos.

Atentamente,



Carrasco Cando David Alexander
Teléfono de contacto: 0979108983
E-Mail: david.carrasco@est.itstena.edu.ec



Anexo 3 Carta de Aceptación



Secretaría de Educación Superior,
Ciencia, Tecnología e Innovación

Oficio N° ISTT-R-2025-108-OF
Tena, 21 de febrero de 2025

Señor
Carrasco Cando David Alexander
ESTUDIANTE DEL IST TENA
Presenta

De mi consideración:

Con un cordial saludo y en atención al documento s/n, de fecha 20 de febrero de 2025, que en su parte pertinente manifiesta: (...) me encuentro en el proceso de titulación con el tema: 'Desarrollo de un Sistema Web para la Automatización de Solicitudes en el Instituto Superior Tecnológico Tena'; por ende me dirijo a usted para solicitar la **AUTORIZACIÓN** para la implementación del sistema mencionado anteriormente en la plataforma SIAGE (...)

Por lo expuesto, se autoriza realizar la implementación de dicho sistema en la plataforma SIAGE; para lo cual solicito gentilmente coordinar con el Ing. Salomón Cullumba, Responsable de la Unidad de Tecnología de la Información y Comunicación.

Con sentimiento de distinguida consideración.

Atentamente,

Ing. Lorena Pilar Yáñez Palacios., MEd.
RECTORA DEL INSTITUTO SUPERIOR TECNOLÓGICO TENA

Anexo: Solicitud

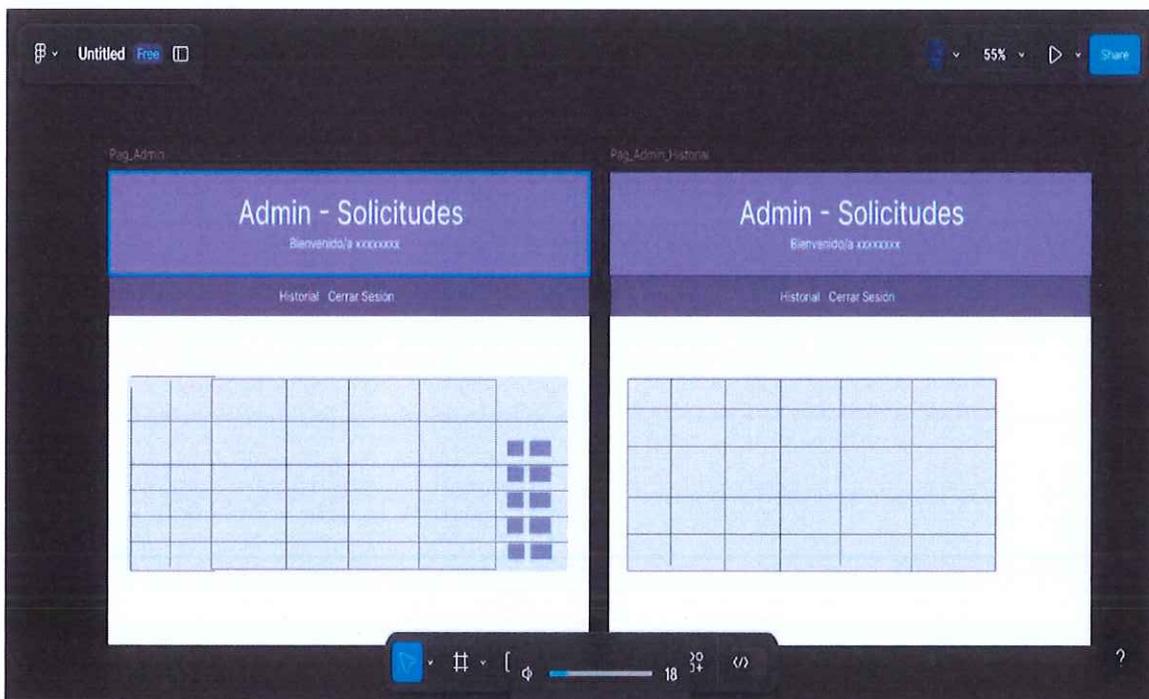
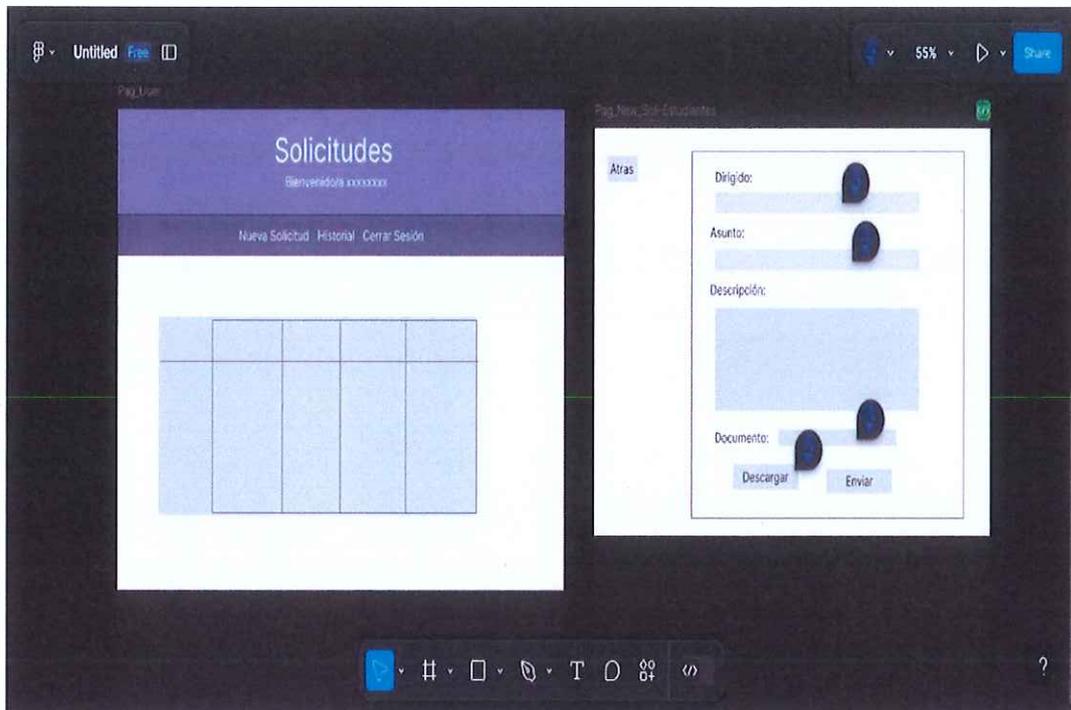
cc.: Ing. Salomón Cullumba, RESPONSABLE DE LA UNIDAD DE TECNOLOGÍAS DE LA INFORMACIÓN

Punto de Atención al Usuario: km 1 ½ vía Tena – Archidona
Teléfono: 062311709
secretaria.general@itstena.edu.ec
<https://www.itstena.edu.ec>



Anexo 4

Maquetación de las interfaces



Anexo 5
Diagrama de la base de datos

