

REPUBLICA DOMINICANA
UNIVERSIDAD AUTONOMA DE SANTO DOMINGO
FACULTAD DE CIENCIAS
Escuela de Informática



TITULO DEL PROYECTO
DISEÑO Y DESARROLLO DE UN SISTEMA WEB EN PYTHON PARA
LA GESTIÓN DE PAGOS E INGRESOS DE CURSOS Y
DIPLOMADOS EN LA UNIDAD DE EDUCACIÓN CONTINUADA DE
LA FACULTAD DE CIENCIAS ECONÓMICAS Y SOCIALES (FCES)

Sustentantes

Deivi Alberto, Turbí Lachapel
Jose Antonio, Ventura Montas
Yonger Adonis, Suero Jean

Coordinador

Lic. Eddy Brito

Asesor Metodológico

Maestro José Manuel Amado Peralta

Santo Domingo, República Dominicana, Distrito Nacional
febrero del año 2025

REPUBLICA DOMINICANA
UNIVERSIDAD AUTONOMA DE SANTO DOMINGO
FACULTAD DE CIENCIAS
Escuela de Informática



TITULO DEL PROYECTO
DISEÑO Y DESARROLLO DE UN SISTEMA WEB EN PYTHON PARA
LA GESTIÓN DE PAGOS E INGRESOS DE CURSOS Y
DIPLOMADOS EN LA UNIDAD DE EDUCACIÓN CONTINUADA DE
LA FACULTAD DE CIENCIAS ECONÓMICAS Y SOCIALES (FCES)

Sustentantes

Deivi Alberto, Turbí Lachapel.....DE2214
Jose Antonio, Ventura Montas.....100346945
Yonger Adonis, Suero Jean.....100363860

Coordinador

Lic. Eddy Brito

Asesor Metodológico

Maestro José Manuel Amado Peralta

Santo Domingo, República Dominicana, Distrito Nacional
febrero del año 2025

TEMA

**DISEÑO Y DESARROLLO DE UN SISTEMA WEB EN
PYTHON PARA LA GESTIÓN DE PAGOS E INGRESOS DE
CURSOS Y DIPLOMADOS EN LA UNIDAD DE EDUCACIÓN
CONTINUADA DE LA FACULTAD DE CIENCIAS
ECONÓMICAS Y SOCIALES (FCES)**

FICHA GENERAL DEL PROYECTO

Información principal y autorización del proyecto	
Fecha: 09/08/2024	Nombre del Proyecto: DISEÑO Y DESARROLLO DE UN SISTEMA WEB EN PYTHON PARA LA GESTIÓN DE PAGOS E INGRESOS DE CURSOS Y DIPLOMADOS EN LA UNIDAD DE EDUCACIÓN CONTINUADA DE LA FACULTAD DE CIENCIAS ECONÓMICAS Y SOCIALES (FCES)
Coordinador del proyecto:	Mtro. Eddy Brito
Asesores del proyecto:	Metodológico: Mtro. José Manuel Amado Contenido: Iamel Matos Feliz
Miembros del proyecto: <div style="text-align: center;"> Deivi Alberto, Turbí Lachapel José Antonio, Ventura Montas Yonger Adonis, Suero Jean </div>	
Fecha de inicio del proyecto: 09/10/2024	Fecha de finalización: 02/26/2025
Necesidad del proyecto: La creciente demanda de optimización y modernización en el DISEÑO Y DESARROLLO DE UN SISTEMA WEB EN PYTHON PARA LA GESTIÓN DE PAGOS E INGRESOS DE CURSOS Y DIPLOMADOS EN LA UNIDAD DE EDUCACIÓN CONTINUADA DE LA FACULTAD DE CIENCIAS ECONÓMICAS Y SOCIALES (FCES). Actualmente, la administración trabaja manual, esto hace que el flujo financiero sea ineficiente y propenso a errores, dificulta la toma de decisiones y la transparencia. La implementación de un sistema web automatizado centralizará la información, mejorará la trazabilidad, comunicación y permitirá a la institución enfrentar los desafíos actuales con mayor competitividad.	

Tabla 1 Ficha General

AGRADECIMIENTOS

Muchas gracias a dios a Dios por haberme brindado la fortaleza y la claridad mental para concluir este proyecto con éxito. Sin su orientación, este camino habría sido mucho más difícil.

Agradezco a la Universidad Autónoma de Santo Domingo (UASD), a la Facultad de Ciencias y la Escuela de Informática, por proporcionarme las instalaciones y el apoyo académico necesarios para desarrollar esta investigación.

Quiero destacar el invaluable respaldo del Lic. Eddy Brito, coordinador del proyecto, quien con su experiencia y generosidad fue una pieza clave para la culminación de este trabajo. Su orientación constante fue fundamental para superar los desafíos.

Un agradecimiento especial al maestro José Manuel Amado Peralta, cuyo asesoramiento metodológico fue indispensable para darle forma a mis ideas y dar estructura a este trabajo. Y a mi asesor de contenido, Iamel Matos Feliz, por su apoyo y aporte a la mejora de mi proyecto.

A mi familia, por su paciencia, cariño y por brindarme el apoyo emocional necesario en los momentos de incertidumbre. Su amor me impulsó a seguir adelante. En especial, agradezco a mi madre Denia M. Lachapel y a mis hermanos Lic. Jairo Lachapel, Ing. Jorge Lachapel y Lic. Yenny Lachapel por su gran apoyo como familia.

Finalmente, agradezco profundamente a mis compañeros de equipo, José Antonio Ventura Montas y Yonger Adonis Suero Jean, por su esfuerzo y colaboración constante. Este trabajo es fruto del compromiso colectivo y refleja nuestro esfuerzo conjunto.

Deivi A. Turbi Lachapel

AGRADECIMIENTOS

Al finalizar este trabajo, no puedo dejar de expresar mi más profundo agradecimiento a todas aquellas personas que, de una u otra manera, hicieron posible la culminación de esta etapa.

En primer lugar, agradezco a mi madre, quien ha sido mi mayor inspiración y apoyo incondicional. Su amor, paciencia y sacrificio han sido fundamentales en cada paso de mi formación académica. A mi hermana, por su compañía y palabras de aliento en los momentos más desafiantes, por ser un pilar en mi vida y recordarme siempre la importancia de la perseverancia.

A mis abuelos, quienes con su sabiduría, cariño y consejos han sido una fuente constante de motivación. Su ejemplo de esfuerzo y dedicación me ha impulsado a seguir adelante con determinación y valentía.

A mis profesores, por su orientación y compromiso con la enseñanza, por compartir su conocimiento y guiarnos con paciencia a lo largo de este camino académico. Su dedicación ha sido clave para ampliar mis horizontes y fortalecer mis habilidades.

A mis amigos y compañeros de estudio, con quienes compartí largas horas de aprendizaje, discusiones enriquecedoras y momentos de desvelo. Su apoyo y compañía hicieron de este proceso una experiencia más llevadera y significativa.

Un agradecimiento especial a mis compañeros de grupo en este monográfico, con quienes trabajé codo a codo para hacer realidad este proyecto. Su esfuerzo, compromiso y colaboración fueron fundamentales para alcanzar este logro. Juntos superamos desafíos, aprendimos mutuamente y demostramos que el trabajo en equipo es la clave para alcanzar grandes metas.

Finalmente, a todas aquellas personas que, de manera directa o indirecta, han contribuido a la realización de este trabajo. Cada palabra de aliento, cada gesto de apoyo y cada consejo han dejado una huella en mi formación personal y académica.

A todos ustedes, gracias.

Jose Antonio Ventura Montas

AGRADECIMIENTOS

En primer lugar, quiero agradecer a Dios por darme la fortaleza y la sabiduría necesarias para completar este importante paso en mi vida.

A mis padres, quienes con su amor, apoyo incondicional y sacrificio me han permitido llegar hasta aquí. Ustedes son mi inspiración y mi motivación constante.

A mis hermanos, por ser mi apoyo y mis confidentes. Gracias por estar siempre a mi lado y por creer en mí.

A mi esposa, por su comprensión, paciencia y amor durante todo este proceso. Sin tu apoyo, esto no habría sido posible.

A mi hija, por ser mi luz y mi razón de ser. Gracias por darme la energía y la determinación para seguir adelante.

A mis profesores, por su guía, conocimientos y dedicación. Cada uno de ustedes ha contribuido de manera significativa a mi formación académica y personal.

A mis compañeros de tesis, y compañeros de grupo por su colaboración, esfuerzo compartido y espíritu de equipo. Ha sido un honor trabajar junto a ustedes.

Finalmente, quiero expresar mi gratitud a todos aquellos que, de una u otra manera, han sido parte de este viaje académico y personal. Su apoyo y aliento han sido fundamentales para alcanzar este logro.

Yonger Adonis Suero Jean

RESUMEN EJECUTIVO

Este proyecto tiene como objetivo desarrollar un sistema web que automatice la gestión de pagos e ingresos de los cursos y diplomados de la Unidad de Educación Continuada de la Facultad de Ciencias Económicas y Sociales (FCES) de la UASD. Actualmente, estos procesos se realizan de manera manual, lo que ocasiona ineficiencias y errores. Con la implementación de la solución propuesta, se centralizarán los procesos administrativos y financieros, optimizando la gestión de los pagos y mejorando la experiencia de los usuarios.

El sistema está diseñado para automatizar tareas como el registro de pagos y la generación de reportes financieros detallados. Personal administrativo se beneficiarán al acceder a una plataforma digital accesible desde cualquier dispositivo con conexión a internet. Con la digitalización de los procesos, problemas como la falta de transparencia, trazabilidad y los errores administrativos serán minimizados, logrando una mayor eficiencia en la gestión financiera de la unidad.

El desarrollo del sistema se llevará a cabo utilizando Python y Django para el backend, mientras que MySQL será la base de datos. La interfaz de usuario será implementada con Wix. Además, el sistema incluirá módulos esenciales como la autenticación de usuarios, la gestión de pagos, la generación de reportes y notificaciones automáticas. El proceso de desarrollo abarcará diversas fases, desde el análisis inicial hasta la puesta en marcha, con un plan de mantenimiento continuo para asegurar el funcionamiento a largo plazo.

Se espera que el sistema mejore la eficiencia operativa en al menos un 30%, reduciendo tanto la carga administrativa como los errores humanos. El proyecto requiere una inversión inicial para cubrir el desarrollo, adquisición de hardware, software, y la infraestructura necesaria, estimando un presupuesto que también contemple el mantenimiento a largo plazo. A largo plazo, esta solución representará una modernización clave en la gestión financiera de la unidad, asegurando transparencia y seguridad en las transacciones.

ÍNDICE

PRESENTACIÓN.....	i
TEMA.....	ii
FICHA GENERAL DEL PROYECTO.....	iii
AGRADECIMIENTOS.....	iv
RESUMEN EJECUTIVO	vii
1. Introducción.....	1
2. Descripción General Del Proyecto.....	2
2.1 Descripción Del Proyecto.	2
2.2 Objetivo General.	3
2.3 Objetivos Específicos.	3
2.4 Necesidad del Proyecto.....	3
2.5 Antecedentes Del Proyecto.....	4
2.6 Alcance Del Proyecto.	5
2.7 Descripción De Los Entregables Del Proyecto.	7
3. Equipo De Trabajo.....	10
3.1 Miembros del proyecto.....	10
3.2 Organigrama del proyecto.	11
3.3 Descripción De Las Funciones Y Responsabilidades.	11
4. Descripción De Los Aspectos Técnicos.....	14
4.1 Requerimientos de hardware.	14
4.2 Requerimientos de software.	16
4.3 Arquitectura técnica del proyecto.	19
4.3.1. Arquitectura General	19
4.3.2. Componentes de la Arquitectura.....	20
4.3.3. Flujo de Datos	21
4.3.4. Seguridad y Protección	21
4.3.5. Mantenimiento y Escalabilidad	22
5. Presupuesto.....	22
5.1. Costos de Desarrollo (Valores expresados en RD\$)	22
6. Lista De Actividades.	24

7. DESCRIPCIÓN DE ACTIVIDADES. (EXPRESADOS EN DÍAS)	33
8. Matriz De Precedencia	43
9. Matriz De Tiempo. (tiempo en horas).....	47
10. Matriz De Información. (tiempo en horas)	51
11. Matriz De Costos.	54
12. Matriz De Riesgo.....	57
13. Descripción De Las Limitaciones De Recursos.	59
14. Calendario De Ejecución De Proyectos.	61
14.1 Fechas relevantes	61
15. Diagrama de Gantt	62
16. Herramientas De Seguimiento Y Control	65
17. Conclusiones Y Recomendaciones.....	67
16. Bibliografía Y Referencias Electrónicas	69
19. Anexos.....	70
A1 (Guía de entrevistas - Preguntas formuladas al cliente)	70
A2.0 (Factura cotización de equipos Mínimo)	76
A2.1 (Factura cotización de equipos Media)	77
A2.2 (Factura cotización de equipos Máximo).....	78
A2.3 (Factura cotización de equipos Recomendada).....	79
A2.4 (Factura cotización de equipos Recomendado).....	80
A2.5 (Factura cotización de equipos Mínimo)	81
A3.0 (-Base de datos conectada)	82
A4.0 (-Diccionario de Datos).....	89
A5.0 (-Cotización).....	93
A6.0 Informe Final Análisis Software Antiplagio	96

1.INTRODUCCIÓN

En el presente proyecto se desarrolla un sistema web en Python destinado a la gestión de pagos e ingresos de los cursos y diplomados ofrecidos por la Unidad de Educación Continuada de la Facultad de Ciencias Económicas y Sociales (FCES) de la Universidad Autónoma de Santo Domingo (UASD).

La necesidad de esta herramienta surge debido a la ineficiencia de los métodos manuales utilizados actualmente, que presentan dificultades en la trazabilidad de las transacciones, la transparencia y la toma de decisiones financieras. El objetivo principal es automatizar los procesos relacionados con los pagos y generar informes financieros detallados, mejorando así la experiencia del personal administrativo.

Para ello, se implementará una plataforma accesible desde cualquier computador con conexión a internet, con altos estándares de seguridad y una interfaz intuitiva que centralizará toda la información financiera y administrativa.

El sistema se desarrollará utilizando Python (Django) como tecnología backend, que proporcionará la estructura y lógica necesarias para gestionar los datos y procesos del sistema. En el frontend, se utilizará Wix, facilitando una interfaz amigable para los usuarios finales. Además, se emplearán bases de datos MySQL para almacenar y organizar la información generada por el sistema. El monitoreo y la estabilidad del sistema serán asegurados mediante el uso de UptimeRobot, una herramienta de monitoreo que garantizará la disponibilidad del servicio y la pronta respuesta ante posibles incidentes.

El documento detalla los aspectos clave del desarrollo del sistema, incluyendo la planificación, los costos, los riesgos, las herramientas utilizadas y el seguimiento del progreso, les invitamos a conocer más de este proyecto a continuación.

2. DESCRIPCIÓN GENERAL DEL PROYECTO.

2.1 Descripción Del Proyecto.

Nuestra investigación se enfoca en el desarrollo de un sistema web innovador para la gestión eficiente de los procesos financieros en la Unidad de Educación Continuada. Este sistema tiene como objetivo mejorar la eficiencia y la transparencia en la gestión de pagos e ingresos relacionados con los cursos y diplomados ofrecidos, abarcando desde el registro de pagos hasta la generación de informes financieros detallados.

Como equipo de desarrollo, hemos diseñado un sistema web que será implementado utilizando tecnologías modernas de desarrollo, garantizando su accesibilidad y facilidad de uso. Los usuarios administrativos, podrán acceder al sistema desde cualquier computador con conexión a internet, lo que facilitará la administración de pagos, consulta del estado financiero y el seguimiento de las transacciones en tiempo real. Además, el sistema contará con módulos especializados para cubrir áreas clave como la gestión de pagos, seguimiento de ingresos, control de cuentas por cobrar y la generación automática de reportes financieros.

Entre las características clave del sistema se incluyen:

- a) **Control de Usuarios:** Control de acceso según roles, permitiendo que el personal administrativo y contable tengan acceso a las funcionalidades que les corresponden.
- b) **Automatización de Procesos:** Reducción de tareas manuales mediante la automatización de procesos como la conciliación de pagos y generación de comprobante.
- c) **Centralización de Información:** Consolidación de toda la información financiera en una plataforma única para mejorar la precisión y accesibilidad de los datos.
- d) **Interfaz Amigable:** Diseño intuitivo que facilita la navegación y el uso del sistema, tanto para usuarios internos como externos.
- e) **Seguridad y Privacidad:** Implementación de altos estándares de seguridad para proteger la información financiera y personal de los usuarios administrativos.

Estas características permitirán a la Unidad Continuada mejorar su eficiencia operativa, reducir errores en la gestión de pagos e ingresos, y optimizar la experiencia tanto de los participantes como del personal administrativo. El sistema no solo contribuirá a una gestión administrativa y financiera más efectiva, sino que también fortalecerá la transparencia y la confianza en los procesos administrativos y financieros de la Unidad.

2.2 Objetivo General.

Diseñar y desarrollar un sistema web en python para la gestión de pagos e ingresos de cursos y diplomados en la unidad de educación continuada de la Facultad De Ciencias Económicas Y Sociales (FCES)

2.3 Objetivos Específicos.

Nuestro proyecto tiene como objetivo alcanzar los siguientes objetivos específicos:

- ✓ Diseñar la arquitectura del sistema web para la gestión de registros en los pagos e ingresos.
- ✓ Desarrollar un módulo de registro y seguimiento de pagos.
- ✓ Implementar un sistema de generación de reportes financieros.
- ✓ Integrar notificaciones para participantes y administradores.
- ✓ Garantizar la accesibilidad del sistema web para todos los participantes.
- ✓ Realizar pruebas de rendimiento y seguridad del sistema.

2.4 Necesidad del Proyecto.

La Unidad de Educación Continuada de la Facultad de Ciencias Económicas y Sociales (FCES) actualmente carece de un sistema automatizado para la gestión de los pagos e ingresos de los cursos y diplomados. Todos los procesos se realizan manualmente, lo que implica un gran esfuerzo por parte del personal administrativo, incrementa la probabilidad de errores humanos, y dificulta el seguimiento preciso de los ingresos. Esta situación provoca retrasos en la generación de reportes financieros, falta de transparencia en los procesos y posibles pérdidas de información crítica.

Ante este panorama, se hace indispensable el diseño y desarrollo de un sistema web en Python que automatice y centralice la gestión de pagos e ingresos. Esto permitirá optimizar los procesos, mejorar la eficiencia operativa, reducir errores y brindar un servicio más eficiente y accesible tanto para los participantes como para el equipo administrativo. A continuación, destacamos las principales razones que justifican la implementación de un **sistema web para la gestión de pagos e ingresos**:

- ✓ **Falta de automatización:** Los procesos de gestión de pagos e ingresos se realizan de forma manual, lo que aumenta la carga administrativa.
- ✓ **Propensión a errores humanos:** El manejo manual de los datos incrementa las posibilidades de errores en los registros y cálculos.
- ✓ **Ineficiencia operativa:** La administración manual ralentiza los procesos, lo que ocasiona demoras en la generación de reportes financieros.
- ✓ **Falta de transparencia:** No hay un seguimiento claro y preciso de los pagos e ingresos, lo que dificulta la supervisión financiera.
- ✓ **Riesgo de pérdida de información:** La falta de un sistema centralizado puede resultar en pérdidas o inconsistencias en los datos financieros.
- ✓ **Dificultad en la generación de reportes:** Crear reportes financieros precisos de manera manual es un proceso tedioso y poco eficiente.

2.5 Antecedentes Del Proyecto.

El desarrollo de sistemas web para la gestión administrativa en instituciones educativas ha sido objeto de interés en los últimos años, especialmente debido al aumento en la oferta de cursos y diplomados, y la creciente demanda por soluciones digitales que permitan automatizar procesos complejos. Numerosos centros de formación, tanto a nivel nacional como internacional, han implementado plataformas web que facilitan la gestión de sus programas académicos y administrativos, mejorando la eficiencia operativa y la satisfacción de los usuarios.

En este contexto, la **Unidad de Educación Continuada** ha operado tradicionalmente con sistemas de gestión manuales o herramientas no integradas, lo que ha generado problemas recurrentes, como la duplicación de esfuerzos, la desorganización de la información financiera y la dificultad para mantener una comunicación fluida entre los distintos actores involucrados (estudiantes, personal administrativo y docentes).

Existen varios ejemplos de instituciones que han modernizado sus procesos administrativos mediante **sistemas web integrales**. Por ejemplo, algunas universidades han implementado plataformas que abarcan desde la inscripción y pago de cursos hasta el seguimiento del rendimiento académico y la administración de recursos financieros. Estas soluciones no solo han optimizado los procesos operativos, sino que también han contribuido a mejorar la calidad del servicio al proporcionar una plataforma accesible para el personal administrativo, los estudiantes y los docentes.

Con los avances en las tecnologías de la información, hoy es posible desarrollar sistemas personalizados que se adapten a las necesidades específicas de cada institución educativa. En este sentido, nuestro proyecto se nutre de estos antecedentes y se adapta a las particularidades de la **Unidad de Educación Continuada**, que requiere una gestión eficaz de los pagos e ingresos de sus cursos y diplomados. La implementación de un sistema web se inspira en los aprendizajes obtenidos de otras instituciones, incorporando innovaciones que mejorarán la accesibilidad y eficiencia en la administración de los procesos internos.

A nivel local, algunas instituciones han comenzado a adoptar plataformas de gestión en línea, aunque estos esfuerzos aún no se han consolidado en áreas de educación continuada, que suelen tener requisitos más complejos debido a la naturaleza diversa de sus programas. Nuestro proyecto tiene como objetivo implementar un sistema adaptado específicamente a las necesidades de la Unidad de Educación Continuada, aplicando buenas prácticas en el desarrollo de software y aprendiendo tanto de los éxitos como de los desafíos enfrentados por otras instituciones educativas.

2.6 Alcance Del Proyecto.

El proyecto de desarrollo del **Sistema Web para la Gestión de Pagos e Ingresos en la Unidad de Educación Continuada** tiene como objetivo principal el diseño e implementación de una plataforma web integral que abarque todos los aspectos clave relacionados con la gestión de pagos e ingresos de los cursos y diplomados. Esta plataforma estará enfocada en optimizar los procesos administrativos de la unidad, mejorando la experiencia de los usuarios y facilitando una gestión más eficiente.

Aspectos claves del proyecto

Diseño e implementación del sistema web: Se desarrollará una plataforma accesible a través de un navegador web, que permitirá a los usuarios realizar gestiones relacionadas con la inscripción, el pago de cursos, y la administración de ingresos de manera automatizada. El sistema estará adaptado a las necesidades específicas de la **Unidad de Educación Continuada**, garantizando una experiencia optimizada y simplificada para el personal administrativo.

Módulos principales

- **Control de usuarios:** Se implementará un sistema de autenticación y autorización que permitirá al personal administrativo el ingreso a la plataforma con roles y permisos diferenciados, asegurando la privacidad y seguridad de los datos financieros.
- **Inscripción y gestión de registros en los pagos:** Los administradores con acceso a modificar podrán inscribir en cursos, gestionar pagos y consultar el estado de las transacciones. También podrán supervisar y controlar el flujo de ingresos, así como realizar conciliaciones de los pagos recibidos.
- **Notificaciones y comunicación interna:** El sistema enviará notificaciones automáticas cuando se realicen pagos de inscripciones. Esto mejorará la comunicación y facilitará la gestión de los trámites entre los usuarios.
- **Generación de reportes financieros:** El sistema permitirá generar reportes detallados sobre los ingresos generados por los cursos y diplomados, facilitando la toma de decisiones y el análisis de indicadores clave para la gestión financiera de la unidad.

Seguridad y privacidad de datos

El sistema estará diseñado con las más altas medidas de seguridad para garantizar la confidencialidad y protección de los datos financieros y personales de los usuarios. Esto incluirá autenticación segura, encriptación de datos sensibles y control de acceso basado en roles, lo cual es crucial para el manejo seguro de transacciones.

Interfaz amigable y accesible

El sistema contará con una interfaz de usuario intuitiva y accesible, diseñada para garantizar una experiencia de usuario fluida y sencilla para todos los perfiles administrativos.

Implementación y pruebas

Se realizarán pruebas exhaustivas del sistema para asegurar su correcto funcionamiento en diferentes dispositivos y navegadores web, garantizando su estabilidad y rendimiento. Además, se ofrecerá capacitación al personal administrativo sobre el uso de la plataforma, asegurando su correcta adopción.

Limitaciones geográficas y de acceso

Aunque el sistema está diseñado para la **Unidad de Educación Continuada**, al ser una plataforma en línea, los usuarios podrán acceder desde cualquier lugar con conexión a internet, lo que amplía considerablemente su alcance.

Mantenimiento y actualizaciones

El proyecto incluirá una fase de mantenimiento continuo, en la cual se corregirán errores y se realizarán mejoras necesarias para garantizar el buen funcionamiento del sistema a largo plazo. Asimismo, se implementará un plan de actualizaciones regulares que permitirá incorporar nuevas funcionalidades según las necesidades emergentes de la unidad.

2.7 Descripción De Los Entregables Del Proyecto.

El proyecto de desarrollo del **Sistema Web para la Gestión de Pagos e Ingresos de la Unidad de Educación Continuada** contempla la entrega de los siguientes productos y servicios que garantizarán la implementación exitosa y el buen funcionamiento de la plataforma:

Entregables

- **E01 Documentación del proyecto (backend):** Se entregará un informe completo que detallará todos los aspectos del sistema, incluyendo la descripción funcional y no funcional, la arquitectura del sistema, los módulos y funcionalidades desarrolladas, así como el plan de implementación y las pruebas realizadas. [\(Link\)](#)¹
- **E02 Diseño de la interfaz de usuario (HTML):** Se entregará un diseño detallado de la interfaz de usuario que incluirá la estructura de las páginas, la navegación, la disposición de la información y la interacción con los usuarios administrativo. [\(Link\)](#)²
- **E03 Código fuente del sistema:** Se entregará el código fuente completo, incluyendo todos los archivos necesarios para la implementación y el funcionamiento del sistema. Este código será accesible para futuras actualizaciones o modificaciones. [\(Link\)](#)²
- **E04 Implementación del sistema:** El sistema se implementará en un servidor web, configurado para estar disponible para los usuarios de la Unidad de Educación Continuada, garantizando su acceso desde cualquier computador con conexión a internet. [\(Link\)](#)³
- **E05 Pruebas y validación:** Se realizarán pruebas exhaustivas para verificar el correcto funcionamiento del sistema y asegurar que cumpla con los requisitos planteados. Estas pruebas incluirán validaciones funcionales, de seguridad y de rendimiento. [\(Link\)](#)²
- **E06 Manual de usuario:** Se entregará un manual detallado que incluirá instrucciones paso a paso sobre cómo utilizar el sistema, con capturas de pantalla y explicaciones claras de cada funcionalidad. [\(Link\)](#)²

¹ <https://finalapi.up.railway.app/api/docs/>

² <https://github.com/deivilachapel/entregables>

³ https://wilberthvers03-wixsite-com.filesusr.com/html/4eb7f1_bbef188c7ac615ed25dadac44aaa81e5.html

Cronograma de entregables (expresado en semanas)

Nombres Entregables	Cantidad
Documentación del proyecto (backend)	1
Diseño de la interfaz de usuario (HTML)	1
Código fuente del sistema	2
Implementación del sistema	3
Pruebas y validación	3
Manual de usuario	4

Responsables de los entregables

- El **equipo de desarrollo** será responsable de la entrega de la documentación del proyecto, el diseño de la interfaz de usuario, el código fuente del sistema, la implementación, las pruebas y validación, así como del plan de mantenimiento y actualizaciones.
- El **personal administrativo** será responsable de participar en la capacitación y de utilizar el sistema de acuerdo con las indicaciones proporcionadas.
- Tanto el **equipo de desarrollo** como el **personal administrativo** colaborarán en la creación del manual de usuario, asegurando que sea claro y accesible para todos los usuarios.

No entregables

N.º	Descripción de la actividad	Breve descripción	Motivo
1	Entrenamientos A Los Participantes Del proyecto	Se realizan reuniones y se explican tanto la metodología como las tecnologías que se utilizaran en el proyecto.	Se haría en una segunda entrega por falta de tiempo
2	Carga de información Por QR Automático	Se lee la información de la cedula de identidad del usuario desde el QR	Se haría en la segunda entrega por falta de tiempo
3	Análisis Y Monitoreo Del Software	Se verifica el debido funcionamiento del sistema y su mantenimiento necesario.	Se haría en la segunda entrega por falta de tiempo

Tabla 2 No entregables

3. EQUIPO DE TRABAJO.

3.1 Miembros del proyecto.

El equipo encargado de la realización del proyecto está conformado por los siguientes miembros:

- ✓ **Project Manager:** Deivi Alberto Turbí Lachapel
- ✓ **Asesor:** José Amado
- ✓ **Encargado de Base de Datos:** Deivi Alberto Turbí Lachapel
- ✓ **Desarrollador Backend (Django):** Deivi Alberto Turbí Lachapel y José Antonio Ventura Montás
- ✓ **Desarrollador Frontend (Wix):** José Antonio Ventura Montás
- ✓ **Diseñador UI/UX:** José Antonio Ventura Montás
- ✓ **Tester/QA:** Yonger Adonis Suero Jean

3.2 Organigrama del proyecto.

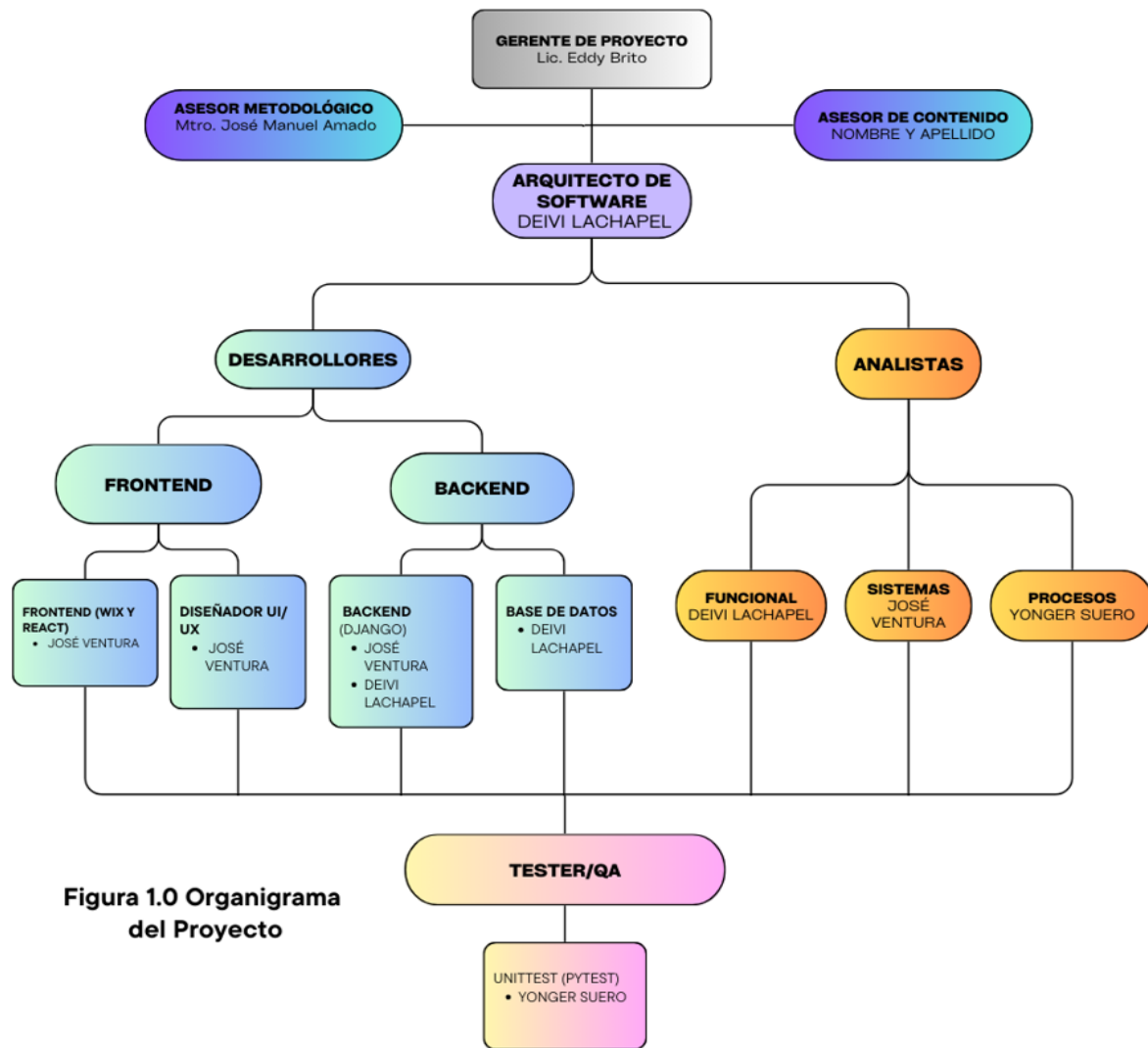


Figura 1.0 Organigrama del Proyecto

Responsable	Puesto	Descripción
Lic. Eddy Brito	Gerente de Proyectos	Responsable de la planificación, supervisión y control del proyecto, asegurando que el equipo cumpla con los plazos y los objetivos establecidos. Coordina la comunicación entre los miembros del equipo y gestiona los recursos necesarios para completar el proyecto de manera efectiva.
Maestro José Manuel Amado Peralta	Asesor de Metodológico	Brinda apoyo y orientación estratégica al equipo de desarrollo, asegurando que las decisiones tomadas estén alineadas con los objetivos del proyecto y las mejores prácticas de la industria.

Responsable	Puesto	Descripción
Iamel Matos Feliz	Asesor de Contenido	Colabora estrechamente con el equipo de desarrollo, brindando orientación sobre la correcta formulación de los documentos, la adecuación del lenguaje, y la integración de fuentes y referencias relevantes. Además, proporciona retroalimentación crítica para mejorar la claridad y precisión de los informes y garantizar que estos reflejen adecuadamente los avances y resultados obtenidos.
Deivi Alberto Turbí Lachapel	Arquitecto de Software	Diseña la estructura técnica de un sistema, tomando decisiones sobre las tecnologías, herramientas y metodologías que se utilizarán en el desarrollo. Es el responsable de asegurar que el sistema sea escalable, eficiente y fácil de mantener.
Deivi Alberto Turbí Lachapel	Encargado de Base de Datos	Responsable del diseño, implementación y mantenimiento de la base de datos. Garantiza la integridad, seguridad y eficiencia en el almacenamiento y acceso a la información, gestionando los backups y optimizando las consultas.
Deivi Alberto Turbí Lachapel José Antonio Ventura Montás	Desarrollad or Backend (Django)	Encargado del desarrollo del servidor, la lógica de negocio y la creación de APIs para la interacción con la base de datos. Se asegura de que el sistema backend sea eficiente, escalable y seguro.
José Antonio Ventura Montás	Desarrollad or Frontend (Wix)	Responsable de la implementación de la interfaz de usuario en el frontend, utilizando Wix. Asegura la correcta visualización y usabilidad de la plataforma en todos los dispositivos, optimizando el rendimiento y la interacción del usuario.
José Antonio Ventura Montás	Diseñador UI/UX	Diseña la experiencia de usuario y la interfaz gráfica, garantizando que sea intuitiva, atractiva y fácil de usar. Se enfoca en crear una navegación fluida y una estética visual coherente con la marca y el objetivo del proyecto.

Responsable	Puesto	Descripción
Deivi Alberto Turbí Lachapel	Analista Funcional	Actúa como un intermediario entre los usuarios finales y el equipo técnico, asegurando que las funcionalidades del sistema cumplan con los objetivos del negocio.
José Antonio Ventura Montás	Analista de Sistemas	Es responsable de investigar, evaluar y diseñar soluciones tecnológicas que optimicen el rendimiento de los sistemas informáticos. Su función principal es analizar los requisitos del cliente, traducirlos en especificaciones técnicas y colaborar en el diseño de soluciones eficientes.
Yonger Adonis Suero Jean	Analista de Procesos	Se encarga de estudiar y mejorar los procesos de negocio de una organización. Su objetivo es identificar inefficiencias y proponer mejoras para optimizar el flujo de trabajo y aumentar la productividad.
Yonger Adonis Suero Jean	Tester/QA	Encargado de realizar pruebas de calidad en el sistema para identificar errores, asegurando que todas las funcionalidades del sistema funcionen correctamente. Se enfoca en la detección de fallos y el aseguramiento de la calidad antes de la implementación final. Proporciona soporte técnico a los usuarios finales del.

Tabla 3 Funciones Y Responsabilidades

4. DESCRIPCIÓN DE LOS ASPECTOS TÉCNICOS.

4.1 Requerimientos de hardware.

Para garantizar el correcto funcionamiento del **Sistema Web para la Gestión de Pagos e Ingresos de la Unidad de Educación Continuada**, es necesario disponer de una infraestructura de hardware adecuada tanto para el desarrollo como para la implementación y uso del sistema. A continuación, se describen los requerimientos de hardware necesarios en cada fase del proyecto:

Servidor de desarrollo y pruebas:

- ✓ **Procesador:** Intel Core i5 o equivalente, con al menos 4 núcleos y 3.0 GHz de velocidad de reloj.
- ✓ **Memoria RAM:** 8 GB o superior, para permitir la ejecución de múltiples procesos simultáneamente durante el desarrollo y pruebas.
- ✓ **Almacenamiento:** 256 GB de disco duro SSD o superior, para asegurar la velocidad en la carga y procesamiento de archivos del proyecto.
- ✓ **Conectividad a Internet:** Conexión de banda ancha estable con al menos 10 Mbps de velocidad de descarga y carga.
- ✓ **Sistema operativo:** Windows 10 o superior / Ubuntu 20.04 LTS o superior.

Servidor de producción:

- ✓ **Procesador:** Intel Xeon E5 o superior, o AMD EPYC, con al menos 8 núcleos para manejar las solicitudes de varios usuarios de manera eficiente.
- ✓ **Memoria RAM:** 16 GB o superior, para garantizar que el sistema pueda gestionar múltiples usuarios de manera fluida, así como transacciones y consultas a la base de datos.
- ✓ **Almacenamiento:** 512 GB de disco duro SSD o superior, asegurando suficiente espacio para almacenar los datos de usuarios, registros de pagos y otros documentos administrativos.
- ✓ **Conectividad a Internet:** Conexión dedicada con al menos 50 Mbps de velocidad de carga y descarga, para soportar las operaciones en línea de varios usuarios simultáneamente.
- ✓ **Sistema operativo:** Linux Ubuntu Server 20.04 LTS o superior / CentOS 8, optimizados para servidores web y base de datos.
- ✓ **Soporte para backups automáticos:** Un sistema de almacenamiento secundario de al menos 1 TB para realizar copias de seguridad periódicas.

Equipos de los usuarios finales (personal administrativo):

- ✓ **Procesador:** Intel Core i3 o equivalente, con al menos 2 núcleos.
- ✓ **Memoria RAM:** 4 GB o superior.
- ✓ **Almacenamiento:** 128 GB HDD o SSD.
- ✓ **Conectividad a Internet:** Conexión estable con al menos 5 Mbps de velocidad de descarga para acceder a la plataforma sin interrupciones.
- ✓ **Navegador web:** Últimas versiones de Google Chrome, Mozilla Firefox, o Microsoft Edge.

Otros requerimientos de hardware:

- **Dispositivos de respaldo y seguridad:** Para proteger la infraestructura del servidor, se recomienda contar con un **sistema de alimentación ininterrumpida (UPS)** y un servidor de respaldo para casos de fallo crítico o desastres naturales.
- **Sistema de protección contra incendios y control climático:** En el centro de datos, se requiere contar con un ambiente controlado, tanto en términos de temperatura como de seguridad, para evitar sobrecalentamiento y otros daños a los equipos.

4.2 Requerimientos de software.

Para el desarrollo e implementación del **Sistema Web para la Gestión de Pagos e Ingresos de la Unidad de Educación Continuada**, que utiliza **Django** como backend, **Wix** como frontend, y **MySQL** para la base de datos, se deben considerar los siguientes requerimientos de software:

Software para el desarrollo

- **Sistema operativo:**

Windows 10 o superior, o **Ubuntu 20.04 LTS** o superior, dependiendo del entorno de desarrollo preferido.

- **Entorno de desarrollo integrado (IDE):**

Visual Studio Code, para facilitar la programación y gestión del código en Python (Django).

➤ **Frameworks y herramientas:**

- ✓ **Django 4.x:** Framework de Python para el desarrollo del backend, gestión de la lógica del negocio y la interacción con la base de datos.
- ✓ **Wix:** Plataforma de desarrollo frontend utilizada para la creación de la interfaz de usuario. Aunque Wix es una plataforma no tradicional para desarrollo frontend, se debe integrar con el backend de Django a través de APIs o herramientas disponibles.

➤ **Base de datos:**

MySQL 8.x: Sistema de gestión de bases de datos relacional, desplegado para almacenar y gestionar los datos del sistema, incluyendo usuarios, pagos y registros.

➤ **Lenguajes de programación:**

- ✓ **Python 3.9** o superior para el desarrollo del backend con Django.
- ✓ **JavaScript (ES6+)** para el frontend, principalmente en la integración con Wix si es necesario.
- ✓ **HTML5 y CSS3:** Para diseño y estilos de la interfaz de usuario, especialmente si Wix permite modificaciones de código.

➤ **Control de versiones:**

Git: Herramienta para la gestión del código fuente y colaboración en el proyecto. Plataformas como **GitHub** o **GitLab** se utilizarán para el almacenamiento del código.

➤ **Servidor web local:**

XAMPP: Aunque XAMPP es generalmente utilizado para entornos de desarrollo PHP, se puede utilizar para pruebas locales de MySQL y conexiones de backend.

Software para la implementación en producción

➤ **Sistema operativo del servidor:**

Ubuntu Server 20.04 LTS o **CentOS 8:** Recomendado para la implementación del backend Django y el servicio de base de datos MySQL.

➤ **Servidor web:**

Apache 2.4 o **Nginx**: Servidores web compatibles con Django y la integración con la base de datos MySQL.

➤ **Base de datos en producción:**

MySQL 8.x: Configurado para manejo eficiente de datos en un entorno de producción.

➤ **Lenguajes y frameworks de servidor:**

- ✓ **Python 3.9+** con **Django 4.x** para el backend.
- ✓ **WIX** para cualquier integración frontend.

➤ **Sistema de gestión de backups:**

rsync y **cron jobs**: Para realizar copias de seguridad periódicas de la base de datos y configuraciones del servidor.

➤ **Certificados de seguridad:**

SSL/TLS: Para cifrar la comunicación entre los usuarios y el servidor, protegiendo los datos sensibles.

Software de seguridad y monitoreo

➤ **Firewall y seguridad:**

UFW (Uncomplicated Firewall) en Ubuntu o **Firewalld** en CentOS para proteger el servidor contra accesos no autorizados.

➤ **Monitoreo del sistema:**

Uptimebot: Para la monitorización del rendimiento y la estabilidad del sistema.

Software para los usuarios finales (personal administrativo)

➤ **Navegador web compatible:**

Últimas versiones de **Google Chrome**, **Mozilla Firefox**, o **Microsoft Edge** para una experiencia óptima al acceder al sistema web.

➤ **Requisitos adicionales:**

- ✓ **Correo electrónico** para la recepción de notificaciones y comunicados del sistema.
- ✓ **PDF Reader** como **Adobe Acrobat Reader** para visualizar documentos generados por el sistema.

4.3 Arquitectura técnica del proyecto.

La **arquitectura técnica** del **Sistema Web para la Gestión de Pagos e Ingresos de la Unidad de Educación Continuada** se diseña para integrar eficientemente el backend desarrollado con Python (**Django**), el frontend creado en **Wix**, y la base de datos **MySQL**. La arquitectura sigue un enfoque de **arquitectura en capas** y **arquitectura de microservicios**, proporcionando una solución robusta, escalable y segura. A continuación, se describen los componentes y su interconexión:

4.3.1. Arquitectura General

Frontend (Wix):

Wix se utiliza para el diseño de la interfaz de usuario del sistema. Ofrece una plataforma para crear y gestionar la experiencia visual y la interacción del usuario. Wix se integra con el backend a través de **APIs** para realizar llamadas a servicios y obtener datos en tiempo real.

Backend (Django):

Django se encarga de la lógica del negocio, el procesamiento de datos y la gestión de usuarios. Django proporciona **APIs RESTful** que permiten la comunicación entre el frontend y el backend, facilitando la manipulación de datos y la ejecución de operaciones administrativas.

Base de Datos (MySQL):

MySQL es el sistema de gestión de bases de datos relacional que almacena toda la información del sistema, incluyendo datos de usuarios, transacciones, y registros académicos. La base de datos está conectada al backend Django mediante **conectores y ORM (Object-Relational Mapping)**.

4.3.2. Componentes de la Arquitectura

Frontend (Wix):

- ✓ **Diseño Visual:** Wix proporciona las herramientas para construir la interfaz de usuario, incluyendo la disposición de elementos, estilos y funcionalidades básicas.
- ✓ **Integración API:** Wix utiliza **JavaScript** para hacer solicitudes HTTP a las **APIs RESTful** proporcionadas por Django, permitiendo la interacción con el backend.

Backend (Django):

- ✓ **API RESTful:** Django expone endpoints de **REST API** que permiten al frontend consultar y actualizar datos en la base de datos. Estos endpoints están protegidos con **autenticación y autorización**.
- ✓ **Lógica del Negocio:** Django maneja la lógica de negocio, incluyendo la validación de datos, el procesamiento de pagos y la gestión de usuarios.
- ✓ **Autenticación:** Implementación de un sistema de **autenticación y autorización** para controlar el acceso a las diferentes funcionalidades del sistema.
- ✓ **Seguridad:** Utiliza **middleware** y **prácticas de seguridad** para proteger la aplicación contra vulnerabilidades comunes.

Base de Datos (MySQL):

- ✓ **Modelo de Datos:** Django utiliza **ORM** para interactuar con MySQL, facilitando la creación, lectura, actualización y eliminación de registros.
- ✓ **Estructura de Datos:** La base de datos está estructurada en tablas que reflejan los diferentes aspectos del sistema, como usuarios, pagos, cursos y diplomados.
- ✓ **Seguridad de Datos:** Implementación de medidas de seguridad para proteger los datos sensibles, incluyendo cifrado y control de acceso.

4.3.3. Flujo de Datos

1. **Solicitud del Usuario:** Un usuario realiza una acción en la interfaz de Wix, como registrar un curso, registrar una inscripción o un pago académico.
2. **Solicitud al Backend:** Wix envía una solicitud HTTP a uno de los endpoints RESTful expuestos por Django.
3. **Procesamiento en Django:** Django recibe la solicitud, realiza las operaciones necesarias en la base de datos MySQL y procesa la lógica del negocio.
4. **Respuesta al Frontend:** Django devuelve una respuesta al frontend con los datos solicitados o el resultado de la operación.
5. **Actualización de la Interfaz:** Wix actualiza la interfaz de usuario con la información proporcionada por Django.

4.3.4. Seguridad y Protección

- ✓ **Cifrado de Datos:** Utilización de **SSL/TLS** para cifrar la comunicación entre el frontend y el backend.
- ✓ **Autenticación y Autorización:** Implementación de mecanismos de autenticación y autorización en Django para proteger los endpoints de las APIs.
- ✓ **Protección contra Amenazas:** Implementación de medidas de seguridad en el backend para proteger contra ataques comunes como **inyección SQL** y **cross-site scripting (XSS)**.

4.3.5. Mantenimiento y Escalabilidad

- ✓ **Monitoreo:** Se utilizará UptimeRobot para supervisar el tiempo de actividad y el rendimiento del sistema. UptimeRobot permite configurar alertas para notificar a los administradores sobre cualquier interrupción o problema en el sistema, facilitando una respuesta rápida para resolver problemas y asegurar la disponibilidad Continuada del servicio.
- ✓ **Mantenimiento:** Implementación de un plan de mantenimiento regular para aplicar actualizaciones y correcciones.
- ✓ **Escalabilidad:** Diseño modular y basado en servicios para permitir la expansión y el manejo de un mayor número de usuarios y transacciones sin comprometer el rendimiento.

5. PRESUPUESTO.

El presupuesto para el **Sistema Web de Gestión de Pagos e Ingresos de la Unidad de Educación Continuada** abarca los costos asociados con el desarrollo, implementación y mantenimiento del sistema. A continuación, detallamos el presupuesto estimado para cada categoría clave:

5.1. Costos de Desarrollo (Valores expresados en RD\$)

5.1.1. Recursos Humanos:

- **Desarrolladores Backend (Django):**
 - ✓ Costo estimado: 93,350 por desarrollador.
 - ✓ Total (por 2 desarrolladores durante 4 semanas): 186,700
- **Desarrolladores Frontend (Wix):**
 - ✓ Costo estimado: 101,700 por desarrollador.
 - ✓ Total (por desarrollador durante 4 semanas): 101,700
- **Diseñador UI/UX:**
 - ✓ Costo estimado: 90,000
 - ✓ Total: 90,000

➤ **Tester/QA:**

- ✓ Costo estimado: 115,000 por tester.
- ✓ Total (por 1 tester durante 4 semanas): 115,000

5.1.2. Herramientas y Licencias:

➤ **Licencia de Software:**

Costos de licencias para herramientas de desarrollo (si es necesario): 12,000.

➤ **Plataforma Wix:**

Plan de Wix y posibles aplicaciones adicionales: 5,600

5.2. Costos de Implementación

5.2.1. Infraestructura y Servidores:

➤ **Servidor Web:**

Costo estimado del servidor (AWS o DigitalOcean): 232,000 al año.

➤ **Base de Datos:**

Costo de almacenamiento y gestión de base de datos en MySQL: 150,000 al año.

5.2.2. Dominio y Certificado SSL:

➤ **Nombre de Dominio:**

Costo anual: 5,600.

➤ **Certificado SSL:**

Costo anual para cifrado seguro: 4,500

5.3. Costos de Mantenimiento

5.3.1. Soporte Técnico y Mantenimiento:

➤ **Mantenimiento del Sistema:**

Costo estimado anual para correcciones y actualizaciones: 800,000.

➤ **Monitoreo (UptimeRobot):**

Plan de monitoreo y alertas: 6,000 al año.

5.3.2. Capacitación y Documentación:

➤ Capacitación del Personal Administrativo:

Costo de sesiones de capacitación: 70,000.

➤ Documentación y Manuales:

Costo de preparación y distribución de manuales: 85,000.

5.4. Costos Adicionales

5.4.1. Contingencia:

➤ Fondo para Imprevistos:

Porcentaje del presupuesto total para imprevistos: 10% del total estimado. 197,110

5.4.2. Otros Costos:

➤ Gastos Generales:

Costos asociados con reuniones y comunicación: 25,000.

Resumen del Presupuesto

- ✓ **Costos de Desarrollo:** 400,000
- ✓ **Costos de Implementación:** 392,100
- ✓ **Costos de Mantenimiento:** 961,000
- ✓ **Costos Adicionales:** 25,000

Total, Estimado del Presupuesto: 1,753,125

6. LISTA DE ACTIVIDADES.

6.1 Levantamiento de información.

N.º	Descripción de la actividad	Breve Descripción
1	Levantamiento De Requerimientos	Obtención de información necesaria para entender lo que el cliente y los usuarios finales necesitan del sistema. Se recopila la información de la problemática

Tabla 4 Levantamiento de información

6.2 Análisis del sistema.

N.º	Descripción de la actividad	Breve Descripción
2	Estudios de Factibilidad	Se realiza una evaluación preliminar que determina la viabilidad del proyecto desde tres enfoques: técnico (infraestructura y habilidades), económico (costos y beneficios), y operacional (impacto en las operaciones y usuarios). Concluye con un informe que resume resultados y ofrece recomendaciones para decidir sobre el proyecto.
3	Análisis del Sistema Propuesto	Se evalúa la viabilidad y estructura del nuevo sistema, describiendo sus objetivos generales y específicos, justificando su implementación y definiendo la metodología a seguir. También incluye diagramas de flujo de datos, narrativas de los procesos, y un diccionario de datos. Concluye con un informe detallado que valida la propuesta y su adecuación a los requerimientos del proyecto.
4	Requerimientos Funcionales	Son requisitos obligatorios para el funcionamiento del sistema para satisfacer los requisitos del cliente.
5	Casos De Usos	Documentar cómo los diferentes actores interactuarán con el sistema para cumplir con los objetivos planteados.

Tabla 5 Análisis del Sistema

6.3 Diseño Funcional

N.º	Descripción de la actividad	Breve Descripción
6	Diseño Funcional	Crear el diseño que especifica cómo debe funcionar el sistema a nivel de interacción, procesos y componentes, es decir se diseñan las diferentes funciones del proyecto.
7	Casos De Prueba	Definir los escenarios de prueba basados en los casos de uso y los requisitos funcionales, para validar que el sistema cumple con los requerimientos, es decir se prueban los casos previos para determinar si cumplen con las condiciones aceptables.

Tabla 6 Diseño Funcional

6.4 Diseño del Sistema

N.º	Descripción de la actividad	Breve Descripción
8	Arquitectura Preliminar	Diseño de la estructura general del sistema, incluyendo componentes clave como servidores, bases de datos y lógica de negocio, es decir se establece la arquitectura con la que se piensa realizar el proyecto.
9	Diseño Técnico	Definición técnica detallada del sistema, incluyendo la selección de tecnologías, plataformas y componentes técnicos específicos, es decir dependiendo del diseño funcional, este se encarga de darle los rasgos técnicos a este.
10	Creación De La Base De Datos (SQL)	Diseño e implementación de las tablas, relaciones y estructuras necesarias para la base de datos del sistema, es decir se crea la base de datos del sistema.
11	Diagramas UML Que Documentan El Proyecto	Representación visual de la estructura y comportamiento del sistema utilizando diagramas UML (como diagramas de clases, de secuencia, etc.). Son diagramas intuitivos que explican el funcionamiento del sistema de forma no tan técnica.

Tabla 7 Diseño Del Sistema

6.5 Desarrollo del Sistema

N.º	Descripción de la actividad	Breve Descripción
12	Cargar Proyecto a GitHub	Consiste en subir el código del sistema web a un repositorio en la plataforma GitHub. Esto implica inicializar un repositorio en local, conectar el proyecto con el repositorio remoto, realizar un commit de los archivos relevantes y finalmente ejecutar el comando git push para enviar los cambios al servidor de GitHub. Esta actividad asegura que el proyecto esté respaldado y accesible para el equipo de desarrollo o colaboradores.
13	Convertir Código a Python en Model.py	Adaptar o implementar el modelo de datos en el archivo model.py, utilizando el lenguaje de programación Python. Esta tarea implica definir las clases y atributos que representan las estructuras de datos del sistema, asegurando que sean compatibles con la lógica y la base de datos del proyecto.
14	Creación del archivo serializar.py	Consiste en diseñar y desarrollar un archivo dentro del proyecto web que define los serializadores. Estos se utilizan para convertir los datos complejos, como objetos de modelos, en formatos simples como JSON o XML, y viceversa.
15	Creación del archivo views.py	Esta actividad consiste en desarrollar el archivo donde se definen las vistas del sistema web. En un framework como Django, el archivo `views.py` contiene las funciones o clases responsables de procesar las solicitudes del usuario y devolver las respuestas correspondientes.

N.º	Descripción de la actividad	Breve Descripción
16	Creación del archivo url.py	Consiste en desarrollar el archivo encargado de definir las rutas o endpoints del sistema web. En un framework como Django, este archivo mapea las URLs que ingresan los usuarios en el navegador con las vistas correspondientes en el archivo views.py.
17	Despliegue del proyecto a GitHub	Consiste en subir la versión final o actualizada del proyecto al repositorio remoto en GitHub. Esta actividad implica verificar que todos los cambios estén confirmados localmente con git commit.
18	Re-subir proyecto a GitHub /GitHub and commit	realizar el envío de los cambios al repositorio remoto con git push, y asegurarse de que la estructura del proyecto esté bien organizada.
19	Despliegue de la base de datos en clever cloud	Consiste en configurar y publicar la base de datos del sistema web en la plataforma de alojamiento Clever Cloud. Esto incluye la creación de una nueva instancia de base de datos (por ejemplo, MySQL, PostgreSQL), la importación de los datos o esquemas necesarios, y la configuración de credenciales de acceso.
20	Despliegue del backend (aplicación de Django): Railway	Esta actividad consiste en configurar y publicar la aplicación web desarrollada con Django en la plataforma Railway.

N.º	Descripción de la actividad	Breve Descripción
21	Creación de las páginas en wix	Implica diseñar y desarrollar el sitio web utilizando las herramientas visuales de la plataforma. Primero, se planifica la estructura del sitio, definiendo qué páginas serán necesarias, como inicio, productos o contacto. Luego, se personaliza el diseño mediante arrastrar y soltar, ajustando elementos visuales como imágenes, colores y texto. Además, se integran datos dinámicos conectando la API del backend, permitiendo mostrar contenido actualizado en tiempo real.
22	Desarrollo De Controladores Y Programación	Implementación del código que gobierna el comportamiento del sistema, integrando las funcionalidades con los módulos técnicos, es decir se codifica el sistema en el lenguaje de programación elegido.
23	Desarrollo De La Interfaz De Usuario	Creación y diseño de la parte visual del sistema con la que interactuarán los usuarios, es decir se crea la interfaz del sistema (menú, módulos, gráficos).

Tabla 8 Desarrollo del Sistema

6.6 Pruebas y Validación

N.º	Descripción de la actividad	Breve Descripción
24	Ambiente De Pruebas	Preparación del entorno de pruebas donde se ejecutarán los test, es decir se inician las pruebas del sistema.
25	Pruebas Unitarias	Validar cada componente del sistema de manera aislada para asegurar que funcionan correctamente, es decir se prueban que todos los componentes cumplen con su función de manera correcta.
26	Pruebas De Errores Del Software	Ejecutar pruebas que identifiquen posibles fallos o errores en el software, es decir se determina los bug o errores que pueda tener el sistema en funcionamiento.
27	Pruebas De Conexión Con La Base De Datos	Validar que la conexión entre el sistema y la base de datos funcione sin problemas, es decir se verifica la conexión y el debido funcionamiento.
28	Pruebas De Aceptación De Usuario	Confirmar que el sistema cumple con las expectativas y necesidades de los usuarios finales mediante pruebas reales, es decir los usuarios usan la aplicación y dan sus opiniones acerca de esta.

Tabla 9 Prueba y Validación

6.7 Planificación y Documentación

N.º	Descripción de la actividad	Breve Descripción
29	Planificación	Definir el cronograma y los plazos para cada fase del proyecto, garantizando una ejecución ordenada, es decir se le da forma a la información recopilada para elaborar el plan y establecer el horizonte del proyecto.
30	Finalizar Documentación Del Software Preliminar	Completar la documentación inicial del sistema, cubriendo aspectos funcionales y técnicos, es decir es un documento preliminar que sirve de guía para comenzar con la parte técnica del proyecto.
31	Reportes De Pruebas	Documentar los resultados de las pruebas realizadas y los incidentes encontrados durante el proceso de validación, es decir son los posibles inconvenientes que pueda tener el sistema en producción.

Tabla 10 Planificación y Documentación

6. 8 Entrega y Cierre

N.º	Descripción de la actividad	Breve Descripción
32	Ambiente De Entrega	Preparar el entorno final donde se desplegará el sistema para su uso productivo, es decir es el Inicio del proceso de implementación del sistema en producción.
33	Entrega De Documentación Y Producto	Hacer la entrega formal del sistema terminado y su documentación al cliente, es decir se implementa el sistema con la debida documentación y capacitación del personal.

Tabla 11 Entrega Y Cierre

7. DESCRIPCIÓN DE ACTIVIDADES. (EXPRESADOS EN DÍAS)

1. Levantamiento de Requerimientos

- ✓ **Descripción:** Recolectar información para identificar las necesidades del cliente y los usuarios finales.
 - ✓ **Duración:** 7.
 - ✓ **Responsable:** Analista de sistemas.
 - ✓ **Entregable:** Documento de requerimientos.
-

2. Estudios de Factibilidad

2.1. Factibilidad Técnica

- ✓ **Descripción:** Evaluar si la organización cuenta con la infraestructura tecnológica y habilidades necesarias para llevar a cabo el proyecto.
 - ✓ **Duración:** 3.
 - ✓ **Responsable:** Analista de sistemas.
 - ✓ **Entregable:** Informe de factibilidad técnica.
-

3. Análisis del Sistema Propuesto

3.1. Descripción General

- ✓ **Descripción:** Descripción del sistema propuesto y cómo resolverá las necesidades detectadas en los requerimientos.
 - ✓ **Duración:** 2.
 - ✓ **Responsable:** Analista de sistemas.
 - ✓ **Entregable:** Documento de análisis del sistema propuesto.
-

4. Requerimientos Funcionales

- ✓ **Descripción:** Definir y documentar las funcionalidades que el sistema debe cumplir.
 - ✓ **Duración:** 5.
 - ✓ **Responsable:** Analista funcional.
 - ✓ **Entregable:** Documento de especificación funcional.
-

5. Casos de Usos

- ✓ **Descripción:** Detallar cómo los usuarios interactúan con el sistema mediante casos de uso.
- ✓ **Duración:** 4.
- ✓ **Responsable:** Analista de sistemas.
- ✓ **Entregable:** Documento de casos de uso.

6. Diseño Funcional

- ✓ **Descripción:** Desarrollar el diseño funcional, incluyendo pantallas, flujo de datos y casos de uso.
 - ✓ **Duración:** 9.
 - ✓ **Responsable:** Diseñador UI/UX.
 - ✓ **Entregable:** Documento de diseño funcional.
-

7. Casos de Prueba

- ✓ **Descripción:** Definir casos de prueba para validar que el sistema cumple con los requisitos.
 - ✓ **Duración:** 4.
 - ✓ **Responsable:** Tester/QA.
-

Entregable: Matriz de casos de prueba.

8. Diseño del sistema

- ✓ **Descripción:** Planificar y estructurar la arquitectura del sistema web, incluyendo la definición de los componentes principales, su interacción y las tecnologías a utilizar. Esto abarca el diseño de la base de datos, los diagramas de clases, los flujos de información y la interfaz de usuario. El objetivo es garantizar un sistema funcional, escalable y fácil de mantener.
 - ✓ **Duración:** 7.
 - ✓ **Responsable:** Analista de sistemas y Diseñador UI/UX.
 - ✓ **Entregable:** Documentación del diseño del sistema, incluyendo diagramas y prototipos de la interfaz.
-

9. Arquitectura Preliminar

- ✓ **Descripción:** Crear una primera versión de la arquitectura del sistema, considerando componentes clave y su interacción.
 - ✓ **Duración:** 4.
 - ✓ **Responsable:** Arquitecto de software.
 - ✓ **Entregable:** Diagrama de arquitectura preliminar.
-

10. Diseño Técnico

- ✓ **Descripción:** Definir la estructura técnica del sistema, seleccionando tecnologías e infraestructura.
 - ✓ **Duración:** 5.
 - ✓ **Responsable:** Arquitecto de software.
 - ✓ **Entregable:** Documento de diseño técnico.
-

11. Creación de la Base de Datos (SQL.)

- ✓ **Descripción:** Diseñar e implementar la base de datos con todas las tablas y relaciones necesarias.
 - ✓ **Duración:** 5.
 - ✓ **Responsable:** Administrador de base de datos.
 - ✓ **Entregable:** Base de datos operativa.
-

12. Diagramas UML que Documentan el Proyecto

- ✓ **Descripción:** Crear diagramas UML para representar gráficamente la estructura del sistema.
 - ✓ **Duración:** 5.
 - ✓ **Responsable:** Arquitecto de software.
 - ✓ **Entregable:** Conjunto de diagramas UML.
-

13. Despliegue del proyecto a GitHub

- ✓ **Descripción:** Subir el proyecto completo al repositorio remoto en GitHub para garantizar su respaldo y facilitar la colaboración en equipo. Incluye la configuración inicial, realizar commits y ejecutar el push al repositorio.
 - ✓ **Duración:** 1.
 - ✓ **Responsable:** Desarrollador.
 - ✓ **Entregable:** Proyecto alojado y actualizado en un repositorio de GitHub.
-

14. Convertir código a Python en models.py

- ✓ **Descripción:** Crear las clases y estructuras del modelo de datos en Python, representando las entidades del sistema en el archivo models.py. Esto incluye definir atributos, relaciones y métodos según los requerimientos del sistema web.
 - ✓ **Duración:** 2.
 - ✓ **Responsable:** Desarrollador Backend.
 - ✓ **Entregable:** Archivo models.py con las clases y atributos correctamente definidos.
-

15. Creación del archivo serializers.py

- ✓ **Descripción:** Diseñar y desarrollar los serializadores que convierten datos complejos en formatos simples (como JSON) y viceversa, permitiendo la comunicación entre el backend y las aplicaciones cliente.
 - ✓ **Duración:** 1.
 - ✓ **Responsable:** Desarrollador Backend.
 - ✓ **Entregable:** Archivo serializers.py con los serializadores definidos y funcionales.
-

16. Creación del archivo views.py

- ✓ **Descripción:** Desarrollar las vistas responsables de procesar las solicitudes y devolver las respuestas correspondientes. Incluye implementar la lógica necesaria para interactuar con los modelos y serializadores, y gestionar las funcionalidades del sistema.
 - ✓ **Duración:** 2.
 - ✓ **Responsable:** Desarrollador Backend.
 - ✓ **Entregable:** Archivo views.py con las vistas definidas y funcionales.
-

17. Creación del archivo urls.py

- ✓ **Descripción:** Definir las rutas del sistema que mapean las solicitudes a las vistas correspondientes. Esto permite estructurar y organizar el acceso a las diferentes funcionalidades del sistema web.
 - ✓ **Duración:** 1.
 - ✓ **Responsable:** Desarrollador Backend.
 - ✓ **Entregable:** Archivo urls.py con las rutas configuradas y funcionales.
-

18. Re-subir proyecto a GitHub / GitHub and Commit

- ✓ **Descripción:** Actualizar el repositorio en GitHub con los cambios recientes del proyecto. Incluye realizar nuevos commits con una descripción adecuada de los cambios y ejecutar el comando push para sincronizar los archivos locales con el repositorio remoto.
 - ✓ **Duración:** 1.
 - ✓ **Responsable:** Desarrollador.
 - ✓ **Entregable:** Repositorio de GitHub actualizado con los últimos cambios del proyecto.
-

19. Despliegue de la base de datos en Clever Cloud

- ✓ **Descripción:** Configurar y alojar la base de datos del sistema en Clever Cloud para garantizar su disponibilidad en un entorno de producción. Incluye la creación de una instancia de base de datos, la configuración de credenciales y la importación de los datos necesarios.
 - ✓ **Duración:** 2.
 - ✓ **Responsable:** Administrador de base de datos (DBA).
 - ✓ **Entregable:** Base de datos funcional y accesible en Clever Cloud.
-

20. Despliegue del backend (aplicación de Django) en Railway

- ✓ **Descripción:** Configurar y publicar el backend del sistema web, desarrollado con Django, en la plataforma Railway. Incluye ajustar configuraciones para producción, establecer variables de entorno (como SECRET_KEY y base de datos), y realizar el despliegue para que el sistema esté disponible en línea.
 - ✓ **Duración:** 3.
 - ✓ **Responsable:** Desarrollador Backend.
 - ✓ **Entregable:** Aplicación de Django desplegada y accesible en Railway, con la URL pública para su uso.
-

21. Creación de las páginas en Wix

- ✓ **Descripción:** Diseñar y desarrollar las páginas del sitio web utilizando la plataforma Wix. Esto incluye definir la estructura del sitio, personalizar el diseño visual (colores, tipografía, imágenes), y agregar funcionalidades interactivas mediante Wix Code. Además, se integran datos dinámicos a través de APIs para mostrar contenido actualizado en las páginas.
 - ✓ **Duración:** 4.
 - ✓ **Responsable:** Desarrollador Frontend.
 - ✓ **Entregable:** Páginas completas y funcionales en Wix, con diseño personalizado e integración de datos.
-

22. Desarrollo de Controladores y Programación

- ✓ **Descripción:** Programar la lógica de negocio del sistema, asegurando la correcta integración con la base de datos y la interfaz de usuario.
 - ✓ **Duración:** 5.
 - ✓ **Responsable:** Desarrollador backend.
 - ✓ **Entregable:** Código fuente de los controladores.
-

23. Desarrollo de la Interfaz de Usuario

- ✓ **Descripción:** Diseñar y desarrollar las interfaces gráficas con las que interactuarán los usuarios.
 - ✓ **Duración:** 5.
 - ✓ **Responsable:** Desarrollador frontend.
 - ✓ **Entregable:** Interfaz de usuario funcional.
-

24. Ambiente de Pruebas

- ✓ **Descripción:** Configurar el entorno de pruebas donde se evaluará el rendimiento y la seguridad del sistema.
 - ✓ **Duración:** 2.
 - ✓ **Responsable:** Tester/QA.
 - ✓ **Entregable:** Entorno de pruebas.
-

25. Pruebas Unitarias

- ✓ **Descripción:** Probar cada componente individual del sistema para asegurarse de que cumplen con los requisitos.
 - ✓ **Duración:** 5.
 - ✓ **Responsable:** Tester/QA.
 - ✓ **Entregable:** Reporte de pruebas unitarias.
-

26. Pruebas de Errores del Software

- ✓ **Descripción:** Detectar y corregir errores en el código antes de la integración final del sistema.
 - ✓ **Duración:** 4.
 - ✓ **Responsable:** Desarrollador.
 - ✓ **Entregable:** Reporte de errores corregidos.
-

27. Pruebas de Conexión con la Base de Datos

- ✓ **Descripción:** Verificar que el sistema se conecta correctamente a la base de datos y realiza las consultas necesarias.
 - ✓ **Duración:** 3.
 - ✓ **Responsable:** Tester/QA.
 - ✓ **Entregable:** Reporte de pruebas de conexión.
-

28. Pruebas de Aceptación de Usuario

- ✓ **Descripción:** Realizar pruebas con los usuarios finales para asegurarse de que el sistema cumple con sus expectativas.
- ✓ **Duración:** 5.
- ✓ **Responsable:** Tester/QA.

Entregable: Reporte de pruebas de aceptación

29. Planificación

- ✓ **Descripción:** Definir el cronograma de trabajo y las dependencias entre actividades.
 - ✓ **Duración:** 3.
 - ✓ **Responsable:** Project Manager.
 - ✓ **Entregable:** Cronograma del proyecto.
-

30. Finalizar Documentación del Software Preliminar

- ✓ **Descripción:** Completar la documentación inicial del software, incluyendo especificaciones técnicas y manuales de usuario.
 - ✓ **Duración:** 3.
 - ✓ **Responsable:** Documentador técnico.
 - ✓ **Entregable:** Documentación preliminar.
-

31. Reportes de Pruebas

- ✓ **Descripción:** Documentar los resultados de todas las pruebas realizadas durante el desarrollo, detallando errores encontrados y su resolución.
 - ✓ **Duración:** 3.
 - ✓ **Responsable:** Tester/QA.
 - ✓ **Entregable:** Reporte final de pruebas.
-

32. Ambiente de Entrega

- ✓ **Descripción:** Preparar el entorno de producción donde se implementará el sistema para su uso final.
 - ✓ **Duración:** 3.
 - ✓ **Responsable:** Ingeniero de sistemas.
 - ✓ **Entregable:** Ambiente de entrega configurado.
-

33. Entrega de Documentación y Producto

- ✓ **Descripción:** Hacer la entrega formal del sistema terminado y toda la documentación relacionada al cliente.
 - ✓ **Duración:** 2.
 - ✓ **Responsable:** Project Manager.
 - ✓ **Entregable:** Documentación completa y sistema entregado.
-

8. MATRIZ DE PRECEDENCIA

N.º Actividad	Descripción De La Actividad	Precedencia	Dependencias
1	Levantamiento de Requerimientos	Ninguna	Ninguna
2	Estudios de Factibilidad	1	Requiere los requerimientos levantados
3	Análisis del Sistema Propuesto	1, 2	Requiere el levantamiento de requerimientos y estudio de factibilidad
4	Requerimientos Funcionales	1, 3	Necesita los requerimientos y análisis del sistema
5	Casos De Usos	4	Requiere los requerimientos funcionales
6	Diseño Funcional	4, 5	Basado en los casos de uso y requerimientos funcionales
7	Casos De Prueba	5, 6	Dependiente del diseño funcional y casos de uso
8	Diseño del Sistema	6	Basado en el diseño funcional

N.º Actividad	Descripción De La Actividad	Precedencia	Dependencias
9	Arquitectura Preliminar	8	Requiere el diseño del sistema
10	Diseño Técnico	9	Basado en la arquitectura preliminar
11	Creación De La Base De Datos (SQL.)	9, 10	Necesita el diseño técnico y arquitectura preliminar
12	Diagramas UML Que Documentan El Proyecto	9, 10	Dependiente del diseño técnico y arquitectura preliminar
13	Despliegue del proyecto a GitHub	9, 10, 11	Requiere el diseño técnico y base de datos
14	Convertir código a Python en model.py	9, 11, 12	Necesita los diagramas UML y diseño técnico
15	Creación del archivo serializar.py	14	Basado en la conversión de código
16	Creación del archivo views.py	14, 15	Depende del model.py y serializar.py
17	Creación del archivo url.py	16	Requiere el archivo views.py
18	Re-subir proyecto a GitHub / gitup commit	13, 17	Necesita el despliegue previo y archivos creados
19	Despliegue de la base de datos en Clever Cloud	11, 18	Basado en la base de datos creada y código subido

N.º Actividad	Descripción De La Actividad	Precedencia	Dependencias
20	Despliegue del backend en Railway	18, 19	Depende del código subido y base de datos en la nube
21	Creación de las páginas en Wix	20	Necesita el backend desplegado
22	Desarrollo de controladores y programación	14, 15, 16, 17	Basado en la estructura del código
23	Desarrollo de la interfaz de usuario	21, 22	Depende de las páginas de Wix y programación
24	Ambiente de pruebas	22, 23	Necesita controladores e interfaz finalizados
25	Pruebas Unitarias	24	Requiere el ambiente de pruebas
26	Pruebas de errores Del Software	24	Necesita el ambiente de pruebas
27	Pruebas De Conexión Con La Base De Datos	24	Requiere el ambiente de pruebas
28	Pruebas De Aceptación De Usuario	25, 26, 27	Basado en todas las pruebas previas
29	Planificación	1	Depende del levantamiento inicial de requerimientos

N.º Actividad	Descripción De La Actividad	Precedencia	Dependencias
30	Finalizar Documentación Del Software Preliminar	28	Requiere las pruebas de aceptación
31	Reportes De Pruebas	28	Basado en las pruebas de aceptación
32	Ambiente De Entrega	30, 31	Necesita la documentación final y reportes
33	Entrega De Documentación Y Producto	32	Depende del ambiente de entrega completado

Tabla 12 Matriz de Precedencia

9. MATRIZ DE TIEMPO. (TIEMPO EN HORAS)

N.º Actividad	Descripción	O (Óptimo)	M (Medio)	P (Pésimo)	T (Estándar) ($T = (O + 4M + P) / 6$)
1	Levantamiento de Requerimientos	2	4	7	4.2
2	Estudios de Factibilidad	1	2	3	2.0
3	Análisis del Sistema Propuesto	3	5	7	5.0
4	Requerimientos Funcionales	2	5	6	4.7
5	Casos De Usos	1	3	7	3.3
6	Diseño Funcional	3	5	8	5.2
7	Casos De Prueba	2	3	5	3.2
8	Diseño del Sistema	5	8	10	7.8
9	Arquitectura Preliminar	1	5	8	4.8
10	Diseño Técnico	3	5	7	5.0

N.º Actividad	Descripción	O (Óptimo)	M (Medio)	P (Pésimo)	T (Estándar) ($T = (O + 4M + P) / 6$)
11	Creación De La Base De Datos (SQL.)	2	5	8	5.0
12	Diagramas UML Que Documentan El Proyecto	1	3	5	3.0
13	Despliegue del proyecto a GitHub	1	1	2	1.2
14	Convertir código a Python en model.py	3	4	7	4.3
15	Creación del archivo serializar.py	5	7	9	7.0
16	Creación del archivo views.py	4	9	12	8.7
17	Creación del archivo url.py	1	1	3	1.3
18	Re-subir proyecto a GitHub / gitup commit	1	1	4	1.5

N.º Actividad	Descripción	O (Óptimo)	M (Medio)	P (Pésimo)	T (Estándar) (T = (O + 4M + P) / 6)
19	Despliegue de la base de datos en Clever Cloud	1	1	2	1.2
20	Despliegue del backend en Railway	4	9	15	9.2
21	Creación de las páginas en Wix	5	6	18	7.8
22	Desarrollo de controladores y programación	6	8	10	8.0
23	Desarrollo de la interfaz de usuario	4	5	11	5.8
24	Ambiente de pruebas	2	5	8	5.0
25	Pruebas Unitarias	4	5	10	5.7
26	Pruebas de errores Del Software	3	5	6	4.8
27	Pruebas De Conexión Con La Base De Datos	4	5	7	5.2

N.º Actividad	Descripción	O (Óptimo)	M (Medio)	P (Pésimo)	T (Estándar) ($T = (O + 4M + P) / 6$)
28	Pruebas De Aceptación De Usuario	2	5	5	4.5
29	Planificación	3	5	10	5.5
30	Finalizar Documentación Del Software Preliminar	4	5	20	7.3
31	Reportes De Pruebas	4	5	8	5.3
32	Ambiente De Entrega	2	5	7	4.8
33	Entrega De Documentación Y Producto	1	5	6	4.5

Tabla 13 Matriz de Tiempo

10. MATRIZ DE INFORMACIÓN. (TIEMPO EN HORAS)

N.º Actividad	Descripción	Secuencia	Tiempo
1	Levantamiento de Requerimientos	Ninguna	3.8
2	Estudios de Factibilidad	1	1.8
3	Análisis del Sistema Propuesto	1, 2	5
4	Requerimientos Funcionales	1, 3	4.6
5	Casos De Usos	4	3.3
6	Diseño Funcional	4, 5	5.1
7	Casos De Prueba	5, 6	3.1
8	Diseño del Sistema	6	7.8
9	Arquitectura Preliminar	8	4.8
10	Diseño Técnico	9	5.
11	Creación De La Base De Datos (SQL.)	9, 10	5
12	Diagramas UML Que Documentan El Proyecto	9, 10	3
13	Despliegue del proyecto a GitHub	9, 10, 11	1.1
14	Convertir código a Python en model.py	9, 11, 12	4.3

N.º Actividad	Descripción	Secuencia	Tiempo
15	Creación del archivo serializar.py	14	7
16	Creación del archivo views.py	14, 15	8.6
17	Creación del archivo url.py	16	1.3
18	Re-subir proyecto a GitHub / gitup commit	13, 17	1.5
19	Despliegue de la base de datos en Clever Cloud	11, 18	1.1
20	Despliegue del backend en Railway	18, 19	9.1
21	Creación de las páginas en Wix	20	7.8
22	Desarrollo de controladores y programación	14, 15, 16, 17	8
23	Desarrollo de la interfaz de usuario	21, 22	5.8
24	Ambiente de pruebas	22, 23	5
25	Pruebas Unitarias	24	5.6
26	Pruebas de errores Del Software	24	4.8
27	Pruebas De Conexión Con La Base De Datos	24	5.1

N.º Actividad	Descripción	Secuencia	Tiempo
28	Pruebas De Aceptación De Usuario	25, 26, 27	4.5
29	Planificación	1	5.5
30	Finalizar Documentación Del Software Preliminar	28	7.3
31	Reportes De Pruebas	28	5.3
32	Ambiente De Entrega	30, 31	4.8
33	Entrega De Documentación Y Producto	32	4.5

Tabla 14 Matriz de Información

11. MATRIZ DE COSTOS.

N.º Actividad	Descripción	Costo (RD\$)	Tiempo (horas)
1	Levantamiento de Requerimientos	20,000	4.2
2	Estudios de Factibilidad	10,000	2.0
3	Análisis del Sistema Propuesto	15,000	5.0
4	Requerimientos Funcionales	15,000	4.7
5	Casos De Usos	10,000	3.3
6	Diseño Funcional	25,000	5.2
7	Casos De Prueba	15,000	3.2
8	Diseño del Sistema	25,000	7.8
9	Arquitectura Preliminar	30,000	4.8
10	Diseño Técnico	25,000	5.0
11	Creación De La Base De Datos (SQL.)	20,000	5.0
12	Diagramas UML Que Documentan El Proyecto	20,000	3.0
13	Despliegue del proyecto a GitHub	8,000	1.2

N.º Actividad	Descripción	Costo (RD\$)	Tiempo (horas)
14	Convertir código a Python en model.py	5,000	4.3
15	Creación del archivo serializar.py	5,000	7.0
16	Creación del archivo views.py	10,000	8.7
17	Creación del archivo url.py	8,000	1.3
18	Re-subir proyecto a GitHub / gitup commit	5,000	1.5
19	Despliegue de la base de datos en Clever Cloud	25,000	1.2
20	Despliegue del backend en Railway	20,000	9.2
21	Creación de las páginas en Wix	25,000	7.8
22	Desarrollo de controladores y programación	20,000	8.0
23	Desarrollo de la interfaz de usuario	30,000	5.8
24	Ambiente de pruebas	10,000	5.0
25	Pruebas Unitarias	15,000	5.7

N.º Actividad	Descripción	Costo (RD\$)	Tiempo (horas)
26	Pruebas de errores Del Software	10,000	4.8
27	Pruebas De Conexión Con La Base De Datos	10,000	5.2
28	Pruebas De Aceptación De Usuario	20,000	4.5
29	Planificación	10,000	5.5
30	Finalizar Documentación Del Software Preliminar	10,000	7.3
31	Reportes De Pruebas	10,000	5.3
32	Ambiente De Entrega	10,000	4.8
33	Entrega De Documentación Y Producto	15,000	4.5
Total		511,000	161.8

Tabla 15 Matriz de Costo⁴

⁴ Este precio es igual al Costos de Desarrollo.

12. MATRIZ DE RIESGO.

Riesgo	Probabilidad (%)	Impacto	Respuesta al Riesgo
Retrasos en el desarrollo debido a cambios en los requerimientos	30	Alto: Afecta plazos y recursos.	Establecer un proceso formal de gestión de cambios que incluya revisiones mensuales y aprobación de cambios significativos.
Fallas en la integración del backend (Django) con el frontend (Wix)	20	Alto: Puede afectar la funcionalidad completa.	Realizar pruebas de integración continuas durante el desarrollo, con pruebas unitarias previas a cada integración.
Falta de experiencia en el uso de Wix	40	Moderado: Baja eficiencia y problemas técnicos.	Organizar capacitaciones y workshops iniciales para el equipo de desarrollo, proporcionando documentación y recursos de aprendizaje.
Errores en la base de datos MySQL	25	Alto: Pérdida de datos importantes y posibles interrupciones.	Implementar un sistema de respaldo automático diario y realizar auditorías regulares de la base de datos.
Sobrecarga del servidor web	15	Alto: Inaccesibilidad del sistema durante picos de tráfico.	Utilizar servidores escalables en la nube, optimizar el código para un mejor rendimiento y monitorizar el tráfico en tiempo real.

Riesgo	Probabilidad (%)	Impacto	Respuesta al Riesgo
Problemas de seguridad cibernética (SSL)	10	Alto: Pérdida de confianza del usuario y violación de datos.	Implementar medidas de seguridad robustas, como la encriptación de datos y realizar pruebas de penetración periódicas.
Falta de comunicación entre los equipos de desarrollo	35	Moderado: Retrasos y errores de coordinación.	Fomentar una cultura de comunicación abierta, con reuniones diarias breves y un canal de comunicación (por ejemplo, Slack) para resolver dudas rápidamente.
Sobreestimación de los recursos técnicos	25	Moderado: Aumento en los costos y posibles retrasos.	Realizar una planificación realista basada en capacidades previas y hacer un seguimiento continuo del uso de recursos.
Desacuerdos con el cliente sobre las funcionalidades	30	Alto: Reversiones y retrabajo que afectan el cronograma.	Documentar todos los requisitos y cambios acordados, y realizar revisiones periódicas con el cliente para asegurarse de que las expectativas estén alineadas.
Problemas en las pruebas de usuario	20	Moderado: Necesidad de ajustes que podrían retrasar el lanzamiento.	Involucrar a los usuarios finales desde el inicio para obtener retroalimentación constante y realizar pruebas en fases tempranas del desarrollo.

Riesgo	Probabilidad (%)	Impacto	Respuesta al Riesgo
Cambios en el entorno tecnológico	15	Moderado: Puede requerir adaptaciones en el proyecto.	Mantenerse actualizado sobre las tendencias tecnológicas y ser flexible para realizar ajustes en la arquitectura y herramientas utilizadas.
Problemas de financiamiento	20	Alto: Puede llevar a la paralización del proyecto.	Preparar un plan de financiamiento claro y considerar fuentes alternativas de financiamiento en caso de necesidad.

Tabla 16 Matriz de Riesgo

13. DESCRIPCIÓN DE LAS LIMITACIONES DE RECURSOS.

1. Recursos Humanos

- ✓ **Especialización Técnica:** La necesidad de conocimientos específicos en tecnologías como Django, Wix puede limitar el número de candidatos disponibles. Si no se encuentran desarrolladores adecuados, el proyecto podría retrasarse o necesitar ajustes en su alcance.
- ✓ **Colaboración en Equipo:** Si los miembros del equipo no están familiarizados entre sí o con las herramientas de gestión de proyectos, esto puede generar conflictos y falta de comunicación, afectando el rendimiento del equipo.

2. Recursos Económicos

- ✓ **Ajustes Presupuestarios:** Cambios en las condiciones económicas pueden obligar a recortes en el presupuesto, lo que podría comprometer la calidad del software o limitar la capacitación del personal.
- ✓ **Incertidumbre Financiera:** Dependencia de aportes externos (como fondos gubernamentales o donaciones) puede crear incertidumbre en el flujo de caja del proyecto, haciendo difícil planificar gastos a largo plazo.

3. Recursos Materiales

- ✓ **Obsolescencia de Equipos:** Equipos de cómputo obsoletos pueden ralentizar el desarrollo y pruebas del sistema, afectando la productividad del equipo.
- ✓ **Limitaciones de Software:** Herramientas de desarrollo que no cumplen con los estándares requeridos o que son difíciles de usar pueden generar frustración y disminuir la eficiencia del equipo.

4. Recursos de Tiempo

- ✓ **Planificación de Tiempos:** La subestimación del tiempo necesario para completar ciertas actividades puede llevar a una acumulación de tareas y a un retraso general en el cronograma.
- ✓ **Pruebas Inadecuadas:** Si el tiempo dedicado a las pruebas es insuficiente, pueden surgir errores en producción que impacten la experiencia del usuario y requieran correcciones costosas y tardadas.

5. Recursos Externos

- ✓ **Dependencia de Proveedores:** La dependencia de proveedores para servicios como hosting o licencias de software puede crear vulnerabilidades si estos proveedores no cumplen con sus compromisos.
- ✓ **Cambios Normativos:** Cambios en la legislación o normativas que regulan la educación superior o el uso de datos personales pueden requerir modificaciones en el sistema, afectando el cronograma y los costos.

Recomendaciones para Mitigar Limitaciones

1. **Planificación Proactiva:** Realizar un análisis detallado de las competencias del equipo y asignar tareas según habilidades, además de planificar la capacitación necesaria desde el inicio.
2. **Presupuesto Flexible:** Incorporar márgenes de maniobra en el presupuesto para imprevistos y cambios de alcance.
3. **Revisiones Periódicas:** Implementar reuniones regulares para evaluar el progreso, ajustar tiempos y recursos según sea necesario.
4. **Establecimiento de Contingencias:** Tener planes de contingencia para los riesgos identificados, como la búsqueda de proveedores alternativos o la posibilidad de reubicar tareas dentro del equipo en caso de ausencias.
5. **Documentación Clara:** Mantener una documentación clara y accesible para todos los miembros del equipo, lo que facilitará la comunicación y la continuidad del trabajo, incluso ante cambios de personal.

14. CALENDARIO DE EJECUCIÓN DE PROYECTOS.

14.1 Fechas relevantes

Códigos	Entregables	Fecha De Entrega
E01	Documentación del proyecto (backend)	30 sept 2024
E02	Diseño de la interfaz de usuario (HTML)	03 oct 2024
E03	Código fuente del sistema	08 oct 2024
E04	Implementación del sistema	11 oct 2024
E05	Pruebas y validación	13 oct 2024
E06	Manual de usuario	21 oct 2024

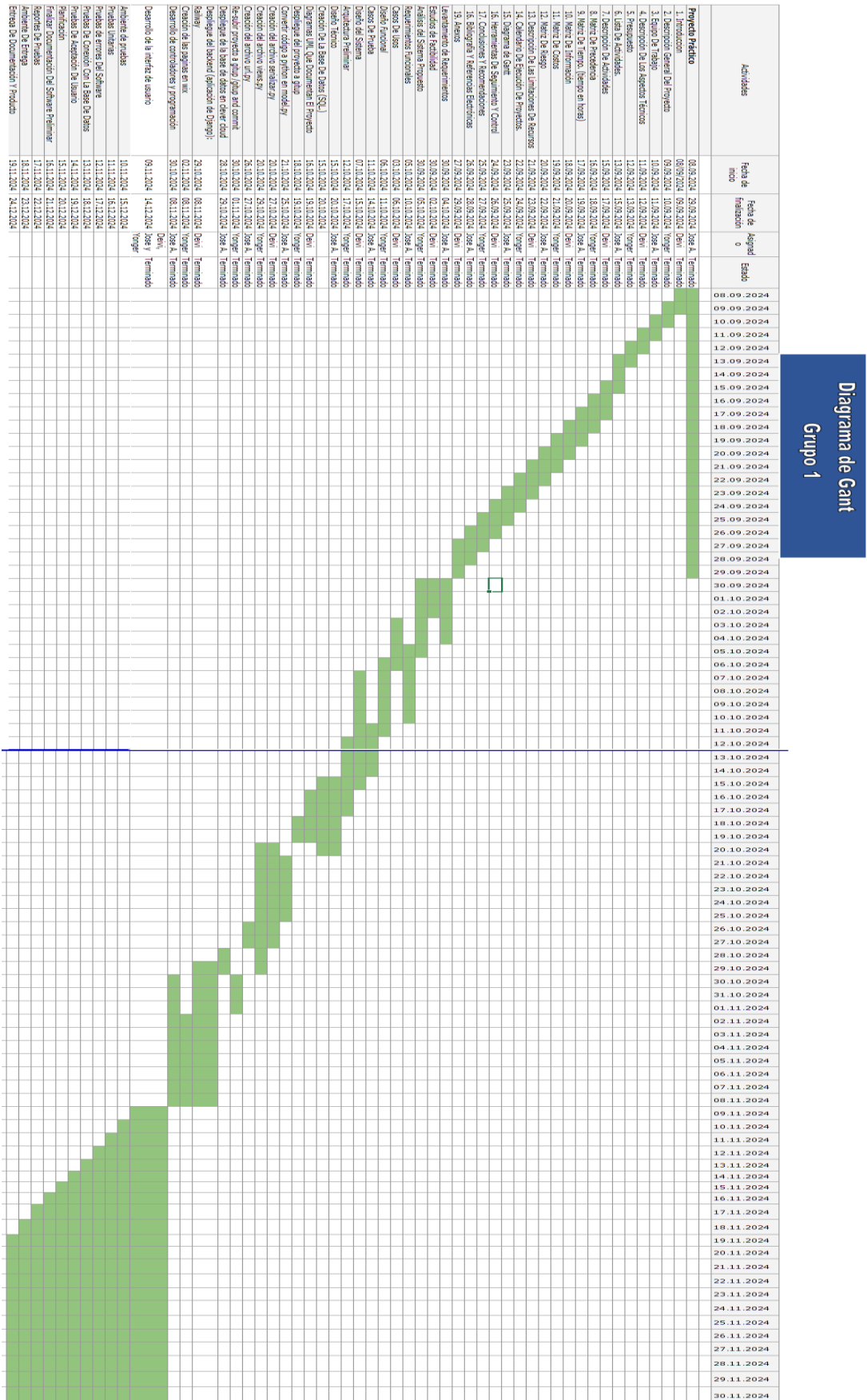
Tabla 17 Calendario de Ejecución

15. DIAGRAMA DE GANTT

Actividades	Fecha de inicio	Fecha de finalización	Asignado	Estado
Proyecto Práctico	08.09.2024	29.09.2024	Jose A.	Terminado
1. Introducción	08/09/2024	09.09.2024	Deivi	Terminado
2. Descripción General Del Proyecto	09.09.2024	10.09.2024	Yonger	Terminado
3. Equipo De Trabajo	10.09.2024	11.09.2024	Jose A.	Terminado
4. Descripción De Los Aspectos Técnicos	11.09.2024	12.09.2024	Deivi	Terminado
5. Presupuesto	12.09.2024	13.09.2024	Yonger	Terminado
6. Lista De Actividades.	13.09.2024	15.09.2024	Jose A.	Terminado
7. Descripción De Actividades	15.09.2024	17.09.2024	Deivi	Terminado
8. Matriz De Precedencia	16.09.2024	18.09.2024	Yonger	Terminado
9. Matriz De Tiempo. (tiempo en horas)	17.09.2024	19.09.2024	Jose A.	Terminado
10. Matriz De Información	18.09.2024	20.09.2024	Deivi	Terminado
11. Matriz De Costos	19.09.2024	21.09.2024	Yonger	Terminado
12. Matriz De Riesgo	20.09.2024	22.09.2024	Jose A.	Terminado
13. Descripción De Las Limitaciones De Recursos	21.09.2024	23.09.2024	Deivi	Terminado
14. Calendario De Ejecución De Proyectos.	22.09.2024	24.09.2024	Yonger	Terminado
15. Diagrama de Gantt	23.09.2024	25.09.2024	Jose A.	Terminado
16. Herramientas De Seguimiento Y Control	24.09.2024	26.09.2024	Deivi	Terminado
17. Conclusiones Y Recomendaciones	25.09.2024	27.09.2024	Yonger	Terminado
16. Bibliografía Y Referencias Electrónicas	26.09.2024	28.09.2024	Jose A.	Terminado
19. Anexos	27.09.2024	29.09.2024	Deivi	Terminado
Levantamiento de Requerimientos	30.09.2024	04.10.2024	Jose A.	Terminado
Estudios de Factibilidad	30.09.2024	02.10.2024	Deivi	Terminado
Análisis del Sistema Propuesto	30.09.2024	05.10.2024	Yonger	Terminado
Requerimientos Funcionales	05.10.2024	10.10.2024	Jose A.	Terminado
Casos De Usos	03.10.2024	06.10.2024	Deivi	Terminado
<i>Diseño Funcional</i>	06.10.2024	11.10.2024	Yonger	Terminado
Casos De Prueba	11.10.2024	14.10.2024	Jose A.	Terminado
Diseño del Sistema	07.10.2024	15.10.2024	Deivi	Terminado
Arquitectura Preliminar	12.10.2024	17.10.2024	Yonger	Terminado
Diseño Técnico	15.10.2024	20.10.2024	Jose A.	Terminado
Creación De La Base De Datos (SQL.)	15.10.2024	20.10.2024		Terminado
Diagramas UML Que Documentan El Proyecto	16.10.2024	19.10.2024	Deivi	Terminado
Despliegue del proyecto a gitup	18.10.2024	19.10.2024	Yonger	Terminado
Convertir código a python en model.py	21.10.2024	25.10.2024	Jose A.	Terminado
Creación del archivo serializar.py	20.10.2024	27.10.2024	Deivi	Terminado
Creación del archivo views.py	20.10.2024	29.10.2024	Yonger	Terminado
Creación del archivo url.py	26.10.2024	27.10.2024	Jose A.	Terminado
Re-subir proyecto a gitup /gitup and commit	30.10.2024	01.11.2024	Yonger	Terminado
Despliegue de la base de datos en clever cloud	28.10.2024	29.10.2024	Jose A.	Terminado
Despliegue del backend (aplicación de Django): Railway	29.10.2024	08.11.2024	Deivi	Terminado
Creación de las páginas en wix	02.11.2024	08.11.2024	Yonger	Terminado
Desarrollo de controladores y programación	30.10.2024	08.11.2024	Jose A.	Terminado

Desarrollo de la interfaz de usuario	09.11.2024	14.12.2024	Deivi, Jose y Yonger	Terminado
Ambiente de pruebas	10.11.2024	15.12.2024		
Pruebas Unitarias	11.11.2024	16.12.2024		
Pruebas de errores Del Software	12.11.2024	17.12.2024		
Pruebas De Conexión Con La Base De Datos	13.11.2024	18.12.2024		
Pruebas De Aceptación De Usuario	14.11.2024	19.12.2024		
Planificación	15.11.2024	20.12.2024		
Finalizar Documentación Del Software Preliminar	16.11.2024	21.12.2024		
Reportes De Pruebas	17.11.2024	22.12.2024		
Ambiente De Entrega	18.11.2024	23.12.2024		
Entrega De Documentación Y Producto	19.11.2024	24.12.2024		

Figura 2.0 Diagrama GANTT



16. HERRAMIENTAS DE SEGUIMIENTO Y CONTROL

En el desarrollo y ejecución del Sistema Web de Gestión de Pagos e Ingresos de la Unidad de Educación Continua, se emplearán diversas herramientas para asegurar un seguimiento efectivo del progreso del proyecto y un control adecuado de los recursos. A continuación, se describen las principales herramientas que se utilizarán:

Herramientas de Gestión de Proyectos

- **Monday:** Se utilizará para la planificación y gestión de tareas. Monday permite organizar actividades, asignar tareas a miembros del equipo y establecer plazos, facilitando el seguimiento del avance del proyecto en tiempo real.

Herramientas de Comunicación

- **Slack:** Se implementará como plataforma de comunicación interna. Slack facilita la comunicación en tiempo real, la creación de canales temáticos y la integración con otras herramientas, lo que ayuda a mantener al equipo conectado y al tanto de las actualizaciones del proyecto.
- **Microsoft Teams:** Además de la mensajería, Teams permitirá realizar videoconferencias y compartir archivos, lo que es fundamental para reuniones y colaboraciones remotas.

Herramientas de Documentación

- **Google Drive:** Para el almacenamiento y colaboración en documentos, hojas de cálculo y presentaciones. Google Drive permite a los miembros del equipo trabajar en documentos simultáneamente, facilitando la actualización y revisión de la documentación del proyecto.
- **Dropbox:** Se usará para la gestión y almacenamiento de documentos del proyecto, permitiendo el acceso fácil y seguro a la documentación por parte del equipo.

Herramientas de Desarrollo

- **GitHub:** Para el control de versiones del código fuente, GitHub permitirá gestionar cambios en el código, colaborar en el desarrollo y llevar un registro de las modificaciones realizadas por cada miembro del equipo.
- **Postman:** Para realizar pruebas de las API desarrolladas, Postman facilitará la verificación del correcto funcionamiento de los endpoints del sistema.

Herramientas de Monitoreo

- **UptimeRobot:** Se utilizará para monitorizar el estado y el rendimiento del sistema una vez implementado. UptimeRobot enviará alertas en caso de caídas del servicio, lo que permitirá una respuesta rápida para minimizar el tiempo de inactividad.

Herramientas de Pruebas

- **Selenium:** Para realizar pruebas automatizadas de la interfaz de usuario. Selenium ayudará a garantizar que las funcionalidades del sistema se comporten correctamente en diferentes navegadores y dispositivos.
- **Jira:** En caso de que se requiera un enfoque más robusto para la gestión de incidencias y pruebas, Jira puede ser utilizado para rastrear errores y gestionar el ciclo de vida de estos.

17. CONCLUSIONES Y RECOMENDACIONES

17.1 Conclusiones

El diseño y desarrollo del sistema web en Python para la gestión de pagos e ingresos de cursos y diplomados en la Unidad de Educación Continuada de la Facultad de Ciencias Económicas y Sociales (FCES) representa un avance significativo en la optimización de los procesos administrativos de la institución. A través de la implementación de tecnologías modernas como Django para el backend y Wix para el frontend, se ha logrado una solución eficiente y accesible que mejora la experiencia del usuario y fortalece la gestión interna.

El sistema desarrollado cumple con los objetivos planteados, abarcando aspectos clave como el diseño de la arquitectura, la implementación de un módulo de registro y seguimiento de pagos, la generación de reportes financieros y la integración de notificaciones. Además, se llevaron a cabo pruebas de rendimiento y seguridad para garantizar su estabilidad y funcionalidad.

El desarrollo del proyecto fue un proceso colaborativo en el que el equipo enfrentó y superó diversos retos técnicos y organizacionales mediante una adecuada planificación, distribución de tareas y comunicación constante. La adopción de este sistema no solo permite automatizar la gestión financiera, sino que también optimiza el uso de recursos y reduce errores administrativos.

17.2 Recomendaciones

- 1) **Ampliación de funcionalidades:** Se sugiere considerar la integración de módulos adicionales, como un sistema de retroalimentación de los estudiantes y un portal de seguimiento académico. Esto permitiría mejorar la interacción y satisfacción de los usuarios, expandiendo el impacto del sistema más allá de la gestión financiera.
- 2) **Capacitación del personal administrativo:** Para maximizar el aprovechamiento del sistema, es recomendable ofrecer capacitaciones periódicas al personal encargado de la gestión de pagos e ingresos, asegurando un uso eficiente de la plataforma.
- 3) **Mantenimiento y actualizaciones:** Se recomienda establecer un plan de mantenimiento continuo para garantizar el correcto funcionamiento del sistema, actualizar la seguridad y mejorar su rendimiento a largo plazo.
- 4) **Optimización de la experiencia del usuario:** Se pueden realizar mejoras en la interfaz gráfica y la navegabilidad de la plataforma para hacerla aún más intuitiva y accesible.
- 5) **Monitoreo y análisis de datos:** Implementar herramientas de análisis de datos para evaluar la eficiencia del sistema y su impacto en la gestión administrativa, permitiendo ajustes y mejoras basadas en información objetiva.

Estamos seguro de que estas recomendaciones van a evolucionar el sistema y consolidarse como una solución integral para la gestión de pagos e ingresos en la FCES, asegurando su sostenibilidad y efectividad a largo plazo.

16. BIBLIOGRAFÍA Y REFERENCIAS ELECTRÓNICAS

- ✓ Holovaty, A., & Willison, S. (21 de julio de 2005). django Framework. Obtenido de django: <https://www.djangoproject.com/>
- ✓ Maffulli, S. (8 de febrero de 1998). Open-Source Initiative. Obtenido de Open SourceInitiative: <https://opensource.org/licenses>
- ✓ Mateu, C. (2006). Desarrollo de aplicaciones web. En C. Mateu, Desarrollo de aplicaciones web (pág. 378). Barcelona: Eureka Media, SL.
- ✓ Otto, M., & Thornton, J. (19 de agosto de 2011). Bootstrap. Obtenido de Bootstrap: <https://getbootstrap.com/>

19. ANEXOS

A1 (Guía de entrevistas - Preguntas formuladas al cliente)

1. ¿Qué tipo de información quieren generar con el sistema?

- Estados financieros, evaluación del tiempo, mejorar los pagos.

2. ¿Tienen algún plazo o fecha límite para implementar la solución?

- 30 días, pero el tiempo real sería de dos meses.

3. ¿Cómo se diferencian los pagos con distintos cursos o diplomados?

- Son iguales en proceso, pero los cursos tienen diferentes tarifas, mientras que los diplomados están estandarizados.

4. ¿Cómo se manejan los pagos parciales?

- El solicitante de reembolso debe presentar una carta firmada. Se puede automatizar el proceso.

5. ¿Quién es responsable del control de los pagos?

- Un equipo de personal administrativo se encarga del seguimiento de los pagos.

6. ¿Cómo se confirman las inscripciones de un estudiante una vez realizado el pago?

- A través de un correo institucional conectado al área financiera.

7. ¿Necesitan enviar recordatorios de pago pendiente o notificaciones de ingresos?

- Sí, es necesario.

8. ¿Qué métodos de pago utilizan?

- Todos los métodos, incluyendo transferencias, Paypal y pagos manuales.

9. ¿Tienen algún requisito de seguridad específico para los datos de pago e información de los estudiantes?

- Sí, es importante asegurar la privacidad de la información.

10. ¿Cómo registran a los estudiantes en el curso?

- Es un proceso manual y en línea, se busca automatizar más.

11. ¿Cuáles son los datos más importantes que necesitan recopilar de los alumnos?

- Nombre, cédula, correo, número de teléfono y referencias.

12. ¿Cómo administran los contenidos y materiales de los cursos?

- No quieren que los materiales sean accesibles desde la plataforma para evitar problemas de ética.

13. ¿Qué método de pago utilizan?

- Utilizan manual, Paypal y transferencias.

14. ¿Tienen personal que puede administrar el sistema una vez esté en funcionamiento?

- Sí, hay personal capacitado para administrar el sistema.

15. ¿Con qué frecuencia necesitan actualizar la información sobre los cursos o pagos?

- Diario.

16. ¿Tienen planes de crecimiento en términos de cantidad de cursos o diplomados?

- Sí, siempre están creciendo.

17. ¿Cómo miden el éxito de esta nueva solución?

- A través del aumento de ingresos y la agilización de los trabajos de los profesores.

18. ¿Qué aspecto consideran más importante en el sistema?

- Que sea fácil de usar y altamente personalizado.

19. ¿Hay alguna otra plataforma o sistema que hayan visto que les gustaría tomar como referencia?

- Se mencionó que se admiraban sistemas existentes, pero no se especificó uno.

20. ¿Qué tipo de reportes necesitan generar regularmente?

- Reportes de ingresos, pagos pendientes y asistencia a cursos.

21. ¿Cómo planean realizar el seguimiento de las interacciones de los estudiantes con la plataforma?

- Se utilizarán herramientas de análisis para monitorear la actividad de los usuarios.

22. ¿Existen algunos datos que deben ser obligatorios para el registro de los estudiantes?

- Sí, el nombre, cédula y correo son obligatorios.

23. ¿Tienen alguna preferencia sobre la tecnología a utilizar para el desarrollo del sistema?

- No se mencionó una preferencia específica, pero buscan algo robusto y fácil de manejar.

24. ¿Qué tipo de soporte técnico necesitarán una vez que el sistema esté en funcionamiento?

- Necesitarán soporte continuo y un manual de usuario.

25. ¿Qué nivel de acceso tendrá cada usuario en la plataforma?

- Habrá diferentes niveles de acceso, dependiendo del rol (administrativo, docente, estudiante).

26. ¿Cómo piensan integrar el sistema con otras herramientas o plataformas que ya usan?

- Buscarán formas de integrar mediante APIs o importaciones/exportaciones de datos.

27. ¿Tienen alguna preferencia sobre el diseño o la interfaz de usuario?

- Quieren un diseño intuitivo y amigable para todos los usuarios.

28. ¿Qué medidas de seguridad tienen en mente para proteger los datos?

- Implementar encriptación y accesos controlados.

29. ¿Tienen algún plan para capacitar al personal en el uso del sistema?

- Sí, se realizarán capacitaciones una vez esté implementado.

30. ¿Cómo planean manejar las quejas o problemas que puedan surgir con el uso del sistema?

- Se establecerá un canal de atención al usuario para resolver inquietudes.

31. ¿Qué características consideran esenciales para la gestión de pagos?

- Facilidad de uso, múltiples métodos de pago y reportes automáticos.

32. ¿Qué tan importante es la personalización de la experiencia del usuario?

- Muy importante, ya que desean que cada usuario tenga una experiencia adaptada a sus necesidades.

33. ¿Tienen en mente realizar pruebas piloto antes del lanzamiento del sistema?

- Sí, realizarán pruebas piloto con un grupo reducido de usuarios.

34. ¿Qué tipo de feedback esperan obtener de los usuarios una vez que el sistema esté en uso?

- Esperan comentarios sobre la facilidad de uso, funcionalidad y eficiencia.

35. ¿Tienen algún plan para actualizar el sistema en el futuro?

- Sí, desean implementar actualizaciones periódicas basadas en el feedback recibido.

36. ¿Qué tipo de contenido esperan que los estudiantes puedan acceder a través del sistema?

- Material de cursos, recursos adicionales y foros de discusión.

37. ¿Cómo piensan evaluar el éxito del sistema después de su implementación?

- A través de métricas de uso, satisfacción del usuario y rendimiento en la gestión de pagos.

38. ¿Qué rol jugarán los docentes en la plataforma?

- Los docentes tendrán la capacidad de cargar contenido, gestionar inscripciones y evaluar a los estudiantes.

39. ¿Planean incorporar alguna función de gamificación en el sistema?

- Están considerando elementos de gamificación para aumentar la motivación de los estudiantes.

40. ¿Cómo manejarán la comunicación entre estudiantes y docentes?

- A través de un sistema de mensajería interna y foros de discusión.

41. ¿Tienen un plazo definido para el desarrollo e implementación del sistema?

- Sí, buscan implementarlo en un plazo de seis meses.

42. ¿Qué recursos adicionales consideran necesarios para el éxito del proyecto?

- Recursos humanos, financieros y técnicos para asegurar un desarrollo fluido.

43. ¿Hay alguna regulación específica que deban cumplir con el sistema?

- Deben cumplir con las normativas de protección de datos y accesibilidad.

44. ¿Qué plan de contingencia tienen en caso de problemas técnicos durante el uso del sistema?

- Se establecerá un protocolo de emergencia para resolver problemas rápidamente.

45. ¿Qué aspectos consideran críticos para la formación de los usuarios?

- La capacitación en el uso del sistema y la comprensión de las políticas de datos.

46. ¿Cómo piensan promover el sistema entre los estudiantes y el personal?

- A través de campañas de información y sesiones de demostración.

47. ¿Tienen alguna visión a largo plazo para el sistema?

- Expandirlo para incluir más funciones y cursos en el futuro.

48. ¿Cómo planean recoger datos para la mejora continua del sistema?

- Mediante encuestas y análisis de uso regular.

49. ¿Qué elementos de diseño consideran esenciales para la interfaz?

- Simplicidad, accesibilidad y un diseño responsive.

50. ¿Cómo se asegurará la calidad del contenido que se sube al sistema?

- Se establecerán procesos de revisión para garantizar la calidad del material.

A2.0 (Factura cotización de equipos Mínimo)



COTIZACION (QUOTATION)

C/Luis E. Del Monte No. 32
Tel: (809) 524-2189, Fax:, E-mail: info@exatech.com.do, Web: www.exatech.com.do
RNC 1-3022773-1

Fecha: 2024-09-25
Cotización #: 69723-1225
Cliente: 2624

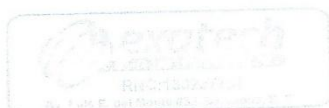
!Exatech! Tu Computienda

Cliente: Yonger Suero
Dirección:
Ciudad: BARAHONA
País: REPUBLICA DOMINICANA
Teléfono:
Fax:
Mail:
RNC:

Enviar a:
Dirección:
Ciudad:
País:
Teléfono:
Fax:
Mail:

Envío (Shipping)	Terminos de Envío (Shipping Terms)	Representante	Fecha de Envío (Delivery Date)

Código	Cantidad	Descripción	Precio Unitario	ITBIS	Total
1	11090	1.00 CPU MINI I5 6TH 8GB/256GB SSD	7,886.06	1,419.49	9,305.56



MERCANCIA SUJETA A CAMBIOS SIN PREVIO AVISO

1. Queda entendido que esta de acuerdo con los precios, términos, forma de envío y especificaciones mas arriba descritas.

2. Envíe toda la correspondencia a:

EXATECH COMPUTER S.R.L

Autorizado por

Recibido por

Fecha

Página 1 de 1

C/Luis E. Del Monte No. 32
Tel: (809) 524-2189, Fax:, E-mail: info@exatech.com.do, Web: www.exatech.com.do

Subtotal	7,886.06
Descuento 0.00%	0.00
ITBIS	1,419.49
Total DOP	9,305.56

A2.1 (Factura cotización de equipos Media)



COTIZACION (QUOTATION)

C/Luis E. Del Monte No. 32
Tel:(809) 524-2189, Fax:, E-mail:info@exatech.com.do, Web: www.exatech.com.do
RNC 1-3022773-1

Fecha: 2024-09-25
Cotización #: 69724-1225
Cliente: 2624

¡Exatech! Tu Computiendita

Cliente: Yongel suero
Dirección:
Ciudad: BARAHONA
País: REPUBLICA DOMINICANA
Teléfono:
Fax:
Mail:
RNC:

Enviar a:
Dirección:
Ciudad:
País:
Teléfono:
Fax:
Mail:

Envío (Shipping)	Términos de Envío (Shipping Terms)	Representante	Fecha de Envío (Delivery Date)

Código	Cantidad	Descripción	Precio Unitario	ITBIS	Total
1	11060	1.00 CPU DELL OPTIBLE 3020 I3 3.6GHz 4gb 128gb ssd SFF	15,243.23	2,743.78	17,987.01

Estudian UASD

MERCANCIA SUJETA A CAMBIOS SIN PREVIO AVISO

1. Queda entendido que esta de acuerdo con los precios, términos, forma de envío y especificaciones mas arriba descritas.

2. Envíe toda la correspondencia a:

EXATECH COMPUTER S.R.L

Autorizado por

Recibido por

Fecha

Página 1 de 1

C/Luis E. Del Monte No. 32
Tel:(809) 524-2189, Fax:, E-mail:info@exatech.com.do, Web: www.exatech.com.do

Subtotal	15,243.23
Descuento 0.00%	0.00
ITBIS	2,743.78
Total DOP	17,987.01

A2.2 (Factura cotización de equipos Máximo)



COTIZACION (QUOTATION)

C/Luis E. Del Monte No. 32
Tel: (809) 524-2189, Fax:, E-mail: info@exatech.com.do, Web: www.exatech.com.do
RNC 1-3022773-1

Fecha: 2024-09-25
Cotización #: 69725-1225
Cliente: 2624

!Exatech! Tu Computiende

Cliente: Yongel suero
Dirección:
Ciudad: BARAHONA
País: REPUBLICA DOMINICANA
Teléfono:
Fax:
Mail:
RNC:

Enviar a:
Dirección:
Ciudad:
País:
Teléfono:
Fax:
Mail:

Envío (Shipping)	Términos de Envío (Shipping Terms)	Representante	Fecha de Envío (Delivery Date)

Código	Cantidad	Descripción	Precio Unitario	ITBIS	Total
1	3519	1.00 TARJETA GRAFICA NVIDIA QUADRO P2000 GDDR5 5GB	20,622.24	3,712.00	24,334.24
2	2351	1.00 CPU DELL I7-7700 16GB/512GB SSD REFURBISHED	31,963.25	5,753.39	37,716.64

Estudian UASD

MERCANCIA SUJETA A CAMBIOS SIN PREVIO AVISO

1. Queda entendido que esta de acuerdo con los precios, términos, forma de envío y especificaciones mas arriba descritas.

2. Envíe toda la correspondencia a:

EXATECH COMPUTER S.R.L



Subtotal	52,585.49
Descuento 0.00%	0.00
ITBIS	9,465.39
Total DOP	62,050.88

Autorizado por

Recibido por

Fecha

Página 1 de 1

C/Luis E. Del Monte No. 32
Tel: (809) 524-2189, Fax:, E-mail: info@exatech.com.do, Web: www.exatech.com.do

A2.3 (Factura cotización de equipos Recomendada)



COTIZACION (QUOTATION)

C/Luis E. Del Monte No. 32
Tel: (809) 524-2189, Fax: , E-mail: info@exatech.com.do, Web: www.exatech.com.do
RNC 1-3022773-1

Fecha: 2024-09-25
Cotización #: 69726-1225
Cliente: 2624

!Exatech! Tu Computienda

Cliente Yongel Suero
Dirección
Ciudad BARAHONA
País REPUBLICA DOMINICANA
Teléfono
Fax
Mail
RNC

Enviar a:
Dirección:
Ciudad:
País:
Teléfono:
Fax:
Mail:

Envío (Shipping)	Términos de Envío (Shipping Terms)	Representante	Fecha de Envío (Delivery Date)

Código	Cantidad	Descripción	Precio Unitario	ITBIS	Total
1	10646	1.00 UPS FORZA SL-1501UL 1500VA - 900 WATTS 8 ENTRADAS,	11,673.13	2,101.16	13,774.29
2	3702	2.00 MEMORIA RAM A-Tech 16GB PC4-21300 Desktop DDR4 266	2,966.10	1,067.80	7,000.00
3	2627	1.00 PROCESADOR AMD PHENOM 9550+ X4	6,133.26	1,103.99	7,237.25

Estudian UASD

MERCANCIA SUJETA A CAMBIOS SIN PREVIO AVISO

1. Queda entendido que esta de acuerdo con los precios, términos, forma de envío y especificaciones mas arriba descritas.

2. Envíe toda la correspondencia a:

EXATECH COMPUTER S.R.L

Autorizado por

Recibido por

Fecha

Página 1 de 1

C/Luis E. Del Monte No. 32
Tel: (809) 524-2189, Fax: , E-mail: info@exatech.com.do, Web: www.exatech.com.do

Subtotal	23,738.59
Descuento 0.00%	0.00
ITBIS	4,272.95
Total DOP	28,011.53

A2.4 (Factura cotización de equipos Recomendado)

COPYCENTRO SHADDAY SRL

COTIZACION (QUOTATION)

RNC 1-17-01312-3

Fecha (Date): 2024-09-25
Cotización (Quotation) #: 71682-1570
Cliente (Customer) ID: 1

COPYCENTRO SHADDAY SRL
ALTA TECNOLOGIA

Cliente (Client) YONGER SUERO
Dirección (Address)
Ciudad (City) SIN CIUDAD
País (Country) República Dominicana
Teléfono (Phone)
Fax (Fax)
Mail (Mail)
RNC

Enviar a (Ship to)
Dirección (Address)
Ciudad (City)
País (Country)
Teléfono (Phone)
Fax (Fax)
Mail (Mail)

Envío (Shipping)	Términos de Envío (Shipping Terms)	Representante	Fecha de Envío (Delivery Date)

Item #	Cantidad (Qty)	Código (Code)	Descripción (Description)	Precio Unitario (Unit Price)	ITBIS (Taxes)	Total (Total Price)
1	1.00	600053	DISCO DURO SSD 2000GB 2TB 2.5 IMATION	8,758.47	1,576.53	10,335.00
			0			
2	2.00	210092	MEMORIA RAM PC XCON DDR4 16GB 3000MHZ 1024M*8/16C	1,923.73	692.54	4,540.00
			0			
3	1.00		PROCESADOR AMD RYZEN TM 5-4500 6-NUCLEOS 12-DESBLO	9,788.14	1,761.86	11,550.00
4	1.00		TARJETA GRAFICA ELSA-RX-550 4GB GDDR5 1XHDML-1XDVI	12,669.49	2,280.51	14,950.00

PRECIOS SUJETOS A CAMBIOS

1. Queda entendido que esta de acuerdo con los precios, términos, forma de envío y especificaciones mas arriba descritas.

(Enter this order in accordance with the prices, terms, delivery method, and specifications listed above.)

2. Envíe toda la correspondencia a:

(Send all correspondence to:)

COPYCENTRO SHADDAY SRL

Página 1 de 1



Autorizado por
(Authorized by)

Recibido por
(Received by)

Fecha
(Date)

URUGUAY 35, BARAHONA

Tel: 809-524-3797 Fax: 809-524-4341 E-mail: M.GUTIERREZ@SHADDAY.SRL.COM Web: https://www.facebook.com/shaddaymonteplata

Subtotal 35,063.56
Descuento 0.00% 0.00
ITBIS 6,311.44
Total DOP 41,375.00

A2.5 (Factura cotización de equipos Mínimo)

COPYCENTRO SHADDAY SRL

COTIZACION (QUOTATION)

RNC 1-17-01312-3

Fecha (Date): 2024-09-25
Cotización (Quotation) #: 71680-1570
Cliente (Customer) ID: 1

COPYCENTRO SHADDAY SRL
ALTA TECNOLOGIA

Cliente (Client) YONGER SUERO
Dirección (Address)
Ciudad (City) SIN CIUDAD
País (Country) República Dominicana
Teléfono (Phone)
Fax (Fax)
Mail (Mail)
RNC

Enviar a (Ship to)
Dirección (Address)
Ciudad (City)
País (Country)
Teléfono (Phone)
Fax (Fax)
Mail (Mail)

Envío (Shipping)		Términos de Envío (Shipping Terms)		Representante	Fecha de Envío (Delivery Date)	
Item #	Cantidad (Qty)	Código (Code)	Descripción (Description)	Precio Unitario (Unit Price)	ITBIS (Taxes)	Total (Total Price)
1	1.00	0	CPU 3.20GHZ DELL/HP CORE I7 6TH 8GB/500GB OPTIPLIX	10,868.64	1,956.36	12,825.00
2	1.00		CPU DELL I3 8GEN 8GB/500DD OPTIPLIX 3010/7010	7,440.68	1,339.32	8,780.00
3	1.00		CPU DELL OPTIPLIX 3.20GHZ I5 6TH 8GB/256	8,898.31	1,601.69	10,500.00



PRECIOS SUJETOS A CAMBIOS

1. Queda entendido que esta de acuerdo con los precios, términos, forma de envío y especificaciones mas arriba descritas.

(Enter this order in accordance with the prices, terms, delivery method, and specifications listed above.)

2. Envíe toda la correspondencia a:

(Send all correspondence to:)

COPYCENTRO SHADDAY SRL

Página 1 de 1

URUGUAY 35, BARAHONA

Tel: 809-524-3797. Fax: 809-524-4341. E-mail: M.GUTIERREZ@SHADDAYSRL.COM. Web: https://www.facebook.com/shaddaycomputienda

Subtotal	27,207.63
Descuento 0.00%	0.00
ITBIS	4,897.37
Total DOP	32,105.00

AutORIZADO POR
(Authorized by)

RECIBIDO POR
(Received by)

FECHA
(Date)

A3.0 (-Base de datos conectada)

-- Crear la base de datos para la gestión de pagos y cursos

```
CREATE DATABASE gestion_pagos_cursos;
```

-- Usar la base de datos recién creada

```
USE gestion_pagos_cursos;
```

-- Tabla Usuarios: almacena información básica de los usuarios del sistema

```
CREATE TABLE Usuarios (
```

```
    id INT AUTO_INCREMENT PRIMARY KEY, -- Identificador único para cada  
    usuario
```

```
    nombre_completo VARCHAR (255) NOT NULL, -- Nombre completo del usuario
```

```
    cedula VARCHAR (20) NOT NULL UNIQUE, -- Cédula única del usuario
```

```
    correo VARCHAR (255) NOT NULL UNIQUE, -- Correo electrónico único del  
    usuario
```

```
    telefono VARCHAR (15), -- Número de teléfono (opcional)
```

```
    celular VARCHAR (15), -- Número de celular (opcional)
```

```
    direccion TEXT, -- Dirección del usuario (opcional)
```

```
    contraseña VARCHAR (255) NOT NULL, -- Contraseña encriptada del usuario
```

```
    fecha_creacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP, -- Fecha de  
    creación del registro
```

```
    estado ENUM ('activo', 'inactivo') DEFAULT 'activo', -- Estado del usuario
```

```
    UNIQUE (cedula, correo) -- Asegura que la combinación de cédula y correo sea  
    única
```

```
);
```

-- Tabla Estudiantes: hereda de Usuarios y almacena información específica de
estudiantes

```

CREATE TABLE Estudiantes (

    id INT PRIMARY KEY, -- Identificador único (heredado de Usuarios)

    matricula VARCHAR (20) NOT NULL UNIQUE, -- Matrícula única del estudiante

    fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP, -- Fecha de
registro del estudiante

    FOREIGN KEY (id) REFERENCES Usuarios(id) ON DELETE CASCADE --
Relación con la tabla Usuarios

);

-- Tabla Docentes: hereda de Usuarios y almacena información específica de
docentes

CREATE TABLE Docentes (

    id INT PRIMARY KEY, -- Identificador único (heredado de Usuarios)

    especialidad VARCHAR (255) NOT NULL, -- Especialidad del docente

    fecha_contratacion DATE NOT NULL, -- Fecha de contratación del docente

    facultad VARCHAR (255) NOT NULL, -- Facultad a la que pertenece el docente

    escuela VARCHAR (255) NOT NULL, -- Escuela dentro de la facultad

    campus VARCHAR (255) NOT NULL, -- Campus donde trabaja el docente

    estatus ENUM ('activo', 'jubilado') NOT NULL DEFAULT 'activo', -- Estado laboral
del docente

    codigo ENUM ('invitado', 'oficial') NOT NULL, -- Tipo de código del docente

    FOREIGN KEY (id) REFERENCES Usuarios(id) ON DELETE CASCADE --
Relación con la tabla Usuarios

);

```

-- Tabla Administrativos: hereda de Usuarios y tiene permisos de acceso

CREATE TABLE Administrativos (

id INT PRIMARY KEY, -- Identificador único (heredado de Usuarios)

departamento VARCHAR (255) NOT NULL, -- Departamento al que pertenece el administrativo

cargo VARCHAR (255) NOT NULL, -- Cargo del administrativo

fecha_ingreso DATE NOT NULL, -- Fecha de ingreso del administrativo

acceso ENUM ('solo_ver', 'ver_agregar', 'superusuario') NOT NULL DEFAULT 'solo_ver', -- Nivel de acceso del administrativo

FOREIGN KEY (id) REFERENCES Usuarios(id) ON DELETE CASCADE --
Relación con la tabla Usuarios

);

-- Tabla Cursos: almacena información sobre los cursos disponibles

CREATE TABLE Cursos (

id INT AUTO_INCREMENT PRIMARY KEY, -- Identificador único para cada curso

nombre VARCHAR (255) NOT NULL, -- Nombre del curso

descripcion TEXT, -- Descripción del curso

tipo ENUM ('curso', 'diplomado') NOT NULL, -- Tipo de curso (curso o diplomado)

tarifa DECIMAL (10, 2) NOT NULL, -- Tarifa del curso

estado ENUM ('activo', 'inactivo') DEFAULT 'activo', -- Estado del curso

fecha_inicio DATE NOT NULL, -- Fecha de inicio del curso

fecha_fin DATE NOT NULL, -- Fecha de finalización del curso

capacidad INT DEFAULT 0, -- Capacidad máxima del curso

docente_id INT, -- Relación con Docentes

modulos INT NOT NULL, -- Número de módulos del curso


```

    horas INT NOT NULL, -- Total de horas del curso

    codigo VARCHAR(20) NOT NULL, -- Código del curso

    profesor VARCHAR(255) NOT NULL, -- Nombre del profesor (alternativa a
docente_id)

    facultad VARCHAR(255) NOT NULL, -- Facultad a la que pertenece el curso

    telefono VARCHAR(15) NOT NULL, -- Teléfono de contacto para el curso

    imagen_url VARCHAR(255), -- URL de la imagen del curso

    FOREIGN KEY (docente_id) REFERENCES Docentes(id) ON DELETE SET
NULL -- Relación con la tabla Docentes

);

-- Tabla Inscripciones: almacena las inscripciones de los estudiantes en los cursos

CREATE TABLE Inscripciones (

    id INT AUTO_INCREMENT PRIMARY KEY, -- Identificador único para cada
inscripción

    estudiante_id INT, -- Relación con la tabla Estudiantes

    curso_id INT, -- Relación con la tabla Cursos

    fecha_inscripcion TIMESTAMP DEFAULT CURRENT_TIMESTAMP, -- Fecha de
inscripción

    estado ENUM('inscrito', 'pendiente') NOT NULL, -- Estado de la inscripción

    FOREIGN KEY (estudiante_id) REFERENCES Estudiantes(id) ON DELETE
CASCADE, -- Relación con la tabla Estudiantes

    FOREIGN KEY (curso_id) REFERENCES Cursos(id) ON DELETE CASCADE --
Relación con la tabla Cursos

);

```

-- Tabla Pagos: almacena información sobre los pagos realizados por los estudiantes

CREATE TABLE Pagos (

id INT AUTO_INCREMENT PRIMARY KEY, -- Identificador único para cada pago

inscripcion_id INT, -- Relación con la tabla Inscripciones

metodo_pago ENUM('transferencia', 'Paypal', 'manual') NOT NULL, -- Método de pago utilizado

monto DECIMAL(10, 2) NOT NULL, -- Monto del pago

estado_pago ENUM('completado', 'pendiente', 'parcial') NOT NULL, -- Estado del pago

fecha_pago TIMESTAMP DEFAULT CURRENT_TIMESTAMP, -- Fecha de realización del pago

fecha_vencimiento DATE, -- Fecha de vencimiento del pago

FOREIGN KEY (inscripcion_id) REFERENCES Inscripciones(id) ON DELETE CASCADE -- Relación con la tabla Inscripciones

);

-- Tabla Historial de Pagos: para auditar cambios en los estados de pago

CREATE TABLE Historial_Pagos (

id INT AUTO_INCREMENT PRIMARY KEY, -- Identificador único para cada registro de historial

pago_id INT, -- Relación con la tabla Pagos

estado_pago_anterior ENUM('completado', 'pendiente', 'parcial'), -- Estado anterior del pago

estado_pago_nuevo ENUM('completado', 'pendiente', 'parcial'), -- Nuevo estado del pago

fecha_cambio TIMESTAMP DEFAULT CURRENT_TIMESTAMP, -- Fecha del cambio de estado

comentario TEXT, -- Comentario sobre el cambio de estado

FOREIGN KEY (pago_id) REFERENCES Pagos(id) ON DELETE CASCADE --
Relación con la tabla Pagos

);

-- Tabla Reembolsos: almacena información sobre solicitudes de reembolso

CREATE TABLE Reembolsos (

id INT AUTO_INCREMENT PRIMARY KEY, -- Identificador único para cada solicitud de reembolso

usuario_id INT, -- Relación con la tabla Usuarios

pago_id INT, -- Relación con la tabla Pagos

motivo TEXT NOT NULL, -- Motivo de la solicitud de reembolso

estado ENUM('pendiente', 'aprobado', 'rechazado') NOT NULL, -- Estado de la solicitud

fecha_solicitud TIMESTAMP DEFAULT CURRENT_TIMESTAMP, -- Fecha de solicitud de reembolso

fecha_resolucion TIMESTAMP NULL, -- Fecha de resolución del reembolso (si aplica)

FOREIGN KEY (usuario_id) REFERENCES Usuarios(id) ON DELETE CASCADE, -- Relación con la tabla Usuarios

FOREIGN KEY (pago_id) REFERENCES Pagos(id) ON DELETE CASCADE --
Relación con la tabla Pagos

);

-- Tabla Notificaciones: almacena notificaciones enviadas a los usuarios

CREATE TABLE Notificaciones (

id INT AUTO_INCREMENT PRIMARY KEY, -- Identificador único para cada notificación

usuario_id INT, -- Relación con la tabla Usuarios

tipo_notificacion ENUM('recordatorio_pago', 'ingreso_registrado') NOT NULL, --
Tipo de notificación

mensaje TEXT NOT NULL, -- Mensaje de la notificación

fecha_envio TIMESTAMP DEFAULT CURRENT_TIMESTAMP, -- Fecha de
envío de la notificación

estado ENUM('enviado', 'pendiente') DEFAULT 'pendiente', -- Estado de la
notificación

FOREIGN KEY (usuario_id) REFERENCES Usuarios(id) ON DELETE CASCADE
-- Relación con la tabla Usuarios
);

-- Tabla Reportes: almacena reportes generados por el sistema

CREATE TABLE Reportes (

id INT AUTO_INCREMENT PRIMARY KEY, -- Identificador único para cada
reporte

tipo_reporte ENUM('ingresos', 'pagos_pendientes', 'asistencia') NOT NULL, --
Tipo de reporte

fecha_generacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP, -- Fecha de
generación del reporte

descripcion TEXT NOT NULL -- Descripción del reporte
);

A4.0 (-Diccionario de Datos)

Python: Es un lenguaje de programación ampliamente utilizado en las aplicaciones web, el desarrollo de software, la ciencia de datos y el machine learning (ML).

Optimizar: La optimización de software es el proceso de mejorar el rendimiento, la eficiencia y la calidad de las aplicaciones de software.

Sistema web: Son todos aquellos softwares que son utilizados accediendo a través de un servidor web, sin necesidad de tener una aplicación en el escritorio.

Dispositivo: Un **dispositivo informático** es un ordenador *que* proporciona uno o más servicios.

Django: Es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como modelo–vista–controlador.

Backend: Son los datos y la infraestructura que permiten que la aplicación funcione. Almacena y procesa los datos de las aplicaciones para los usuarios.

FronTend: El desarrollo web front-end se refiere a la práctica de construir y diseñar la interfaz de usuario de un sitio web o aplicación.

MySQL: Según el autor: Arizbé Ken en su artículo Backen: ¿Qué es y para qué sirve? Publicado 3 de octubre 2023 dice que, “MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual”

Interfaz de usuario: Es el punto de interacción y comunicación humano-computadora en un dispositivo

Wix: Es una compañía SaaS que ofrece servicios de desarrollo web basados en la nube

Diseñador UI/UX: Es el profesional que se encarga de supervisar la experiencia de usuario, la UX (User Experience).

QA: El aseguramiento de la calidad es el conjunto de actividades planificadas y sistemáticas aplicadas en un sistema de gestión de la calidad para que los requisitos de calidad de un producto o servicio sean satisfechos.

Servidor web: Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente y generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente.

Hardware: El hardware, equipo o soporte físico en informática se refiere a las partes físicas, tangibles, de un sistema informático, sus componentes eléctricos, electrónicos y electromecánicos.

Software: Según Luis Olarte Gervacio en su artículo Clasificación de software de sistemas y aplicaciones de fecha 23 abril, 2018, en la plataforma conogasis.org, define que, “es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados, que forman parte de las operaciones de un sistema de computación.”

Licencia de software: Es un contrato entre el licenciante y el licenciataro del programa informático, para utilizarlo cumpliendo una serie de términos y condiciones.

La nube: Es una metáfora para designar una red mundial de servidores remotos que funciona como un único ecosistema, normalmente asociada a Internet.

Mantenimiento: Todas las acciones que tienen como objetivo preservar un artículo o restaurarlo a un estado en el cual pueda llevar a cabo alguna función requerida.

Automatización: Consiste en usar la tecnología para realizar tareas con muy poca intervención humana.

Autenticación: La autenticación o autentificación es el acto o proceso de confirmar que algo es quien dice ser. A la parte que se identifica se le llama probador.

UptimeRobo: Monitorea sitios web, servidores, API y servicios en línea para verificar su disponibilidad y disponibilidad

Arquitectura del sistema: Según la Universidad UNIR, en su publicación titulada Arquitectura de software y sistemas: características y perfiles profesionales sin fecha "describe el proceso más avanzado de diseño, desarrollo y estructura de los sistemas informáticos, incluidos subsistemas, dispositivos de red, hardware y software. Sus modelos conceptuales describen la estructura y el comportamiento de una gran variedad de componentes y subsistemas físicos y lógicos conectados entre si"

Navegador web: Es un programa que permite ver la información que contiene una página web.

Roles: Es un conjunto de permisos que se comparten entre uno o más usuarios.

Encriptar: Según el portal o plataforma proofpoint, “es el proceso de codificar un mensaje o información de modo tal que solo los individuos autorizados sean capaces de acceder a esta.” En otras palabras, es como ponerle un candado a un mensaje para que solo quienes tienen la llave puedan leerlo.

Código fuente: Es un archivo o conjunto de archivos, que contienen instrucciones concretas, escritas en un lenguaje de programación, que posteriormente compilan uno o varios programas.

Código QR: Según el portal web latiendadelasbarras.com y Global Barcodes, S.L. con sede en Madrid, dice que “es la evolución del código de barras. Es un módulo para almacenar información en una matriz de datos o en un código de barras bidimensional”

Backups: Según el blog y portal clavei.es/blog/backup-que-es en su artículo ¿Qué son las copias de seguridad? Y, ¿qué beneficios tiene realizarlas? publicado por el autor Aitor Coves Martínez en Jun 3, 2024, "es el respaldo de información a mayor o menor escala que se realiza en cualquier dispositivo físico o nube."

Api: Es una pieza de código que permite a dos aplicaciones comunicarse entre sí para compartir información y funcionalidades.

Linux Ubuntu: Ubuntu es una distribución GNU/Linux basada en Debian GNU/Linux, que incluye principalmente software libre y de código abierto.

Ghz: El gigahercio es un múltiplo de la unidad de medida de frecuencia hercio y equivale a 10^9 Hz. Por lo tanto, tiene un período de oscilación de 1 nanosegundo.

Frameworks: Según Wikipedia y el Blog de nombre Fazt Blog en su publicación titulada ¿Qué es un Framework de Programación de fecha septiembre 06, 2017 dice que, “es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar”

XAMPP: Es un paquete de software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl.

Apache 2.4: Es un servidor web de código abierto, el más popular de los que encontramos en el mundo de la web. Como todo servidor web, maneja solicitudes HTTP

Nginx: Es un servidor web/Proxy inverso ligero de alto rendimiento y un proxy para protocolos de correo electrónico.

Rsync: Es una aplicación libre para sistemas de tipo Unix y Microsoft Windows que ofrece transmisión eficiente de datos incrementales, que opera también con datos comprimidos y cifrados.

Uncomplicated Firewall ò Firewall: Es un dispositivo de red, que puede ser software (host-based) o un dispositivo de hardware, utilizado para monitorear el tráfico que pasa a través de él, con el fin de bloquear o permitir paquetes, basándose en un conjunto de reglas y capacidades.

Endpoints apis: Son los puntos en los que el cliente y el servidor se comunican, lo que permite que dos aplicaciones compartan recursos.

Http: Es el protocolo de comunicación que permite las transferencias de información a través de archivos en la World Wide Web.

SSL/TLS: Seguridad de la capa de transporte y su antecesor Secure Sockets Layer son protocolos criptográficos, que proporcionan comunicaciones seguras por una red, comúnmente Internet.

Cross-Site Scripting (XSS): Es un tipo de ataque informático que permite a un actor de amenazas ejecutar código malicioso en el navegador de otro usuario.

Certificado SSL: Es un tipo de documento digital firmado por una entidad llamada Autoridad de certificación que asocia una clave pública a unos datos que representan la identidad de una entidad que posee la clave privada asociada a dicha clave pública.

Diagrama UML: El lenguaje unificado de modelado es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad, respaldado por el Object Management Group.

Diagrama de Gantt: Según el Blogger de nombre Excel Interactivo en su artículo Diagrama de Gantt en Excel Cronograma de un proyecto de fecha septiembre 20, 2020 dice que “es una herramienta gráfica cuyo objetivo es exponer el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado.”

A5.0 (-Cotización)

DESARROLLO DE TECNOLOGIA CORPORATIVA

RNC:132355131

Cotización:

DISEÑO Y DESARROLLO DE UN SISTEMA WEB ENPYTHON PARA LA GESTIÓN DE PAGOS E INGRESOS DECURSOS Y DIPLOMADOS EN LA UNIDAD DE EDUCACIÓNCONTINUADA DE LA FACULTAD DE CIENCIASECONÓMICAS Y SOCIALES (FCES)

Cliente: Licda. Maribel

Descripción General del Proyecto

Desarrollo de un sistema web en Python para la gestión de pagos e ingresos de cursos y diplomados en la unidad de educación continuada de la Facultad De Ciencias Económicas Y Sociales (FCES), con el objetivo de mejorar el registro, administración de pagos, y otras transacciones estudiantiles. El sistema permitirá la gestión de pagos, generación de reportes financieros, y contará con acceso seguro para administradores y personal autorizado de la universidad.

Tecnologías:

-Backend: Django (Python)

-Frontend: Wix

-Base de datos: MySQL

- Diseño UI/UX:

- Control de calidad: Testeos funcionales y de usabilidad

Personal Involucrado

1. Desarrolladores Backend (Django) - 2

2. Desarrolladores Frontend (Wix) - 2

3. Diseñador UI/UX - 1

4. Tester/QA - 1

5. Administrador de Base de Datos (DBA) - 1

Alcance del Proyecto

1. Módulo de Administración de Usuarios: Registro, autenticación y gestión de roles.
2. Módulo de Pagos: Generación de pagos, pasarela de pagos en línea, gestión de pagos pendientes.
3. Módulo de Reportes: Reportes financieros, exportación de datos a formatos PDF/Excel.
4. Interfaz de Usuario: Diseño intuitivo y accesible para personal universitario administrativo.
5. Seguridad: Configuración de permisos, cifrado de datos sensibles, y protección de datos personales.

Presupuesto Detallado

Cargo	Cantidad	Tasa (USD/Hora)	Horas Estimadas	Subtotal (USD)
Desarrollador Backend (Django)	2	18	83	1494,000
Desarrollador Frontend (Wix)	2	15	91	2,730,000
Diseñador UI/UX	1	15	91	1,365,000
Tester/QA	1	12	17	204,000
DBA	1	16	17	272,000
Total Estimado	7	76	299	6,061.00

Detalle de Entregables

1. Análisis y Diseño del Sistema

- Análisis de requerimientos: 2 semanas
- Diseño UI/UX y especificaciones técnicas: 2 semanas

2. Desarrollo y Codificación

- Backend (APIs, lógica de negocio, seguridad): 6 semanas
- Frontend (Interfaz en Wix, integración con backend): 5 semanas
- Base de Datos (Esquema y optimización): 2 semanas

3. Pruebas y Control de Calidad

- Pruebas unitarias y funcionales: 3 semanas
- Pruebas de integración y aceptación: 2 semanas

4. Implementación y Entrenamiento

- Implementación en entorno de producción: 1 semana
- Entrenamiento del personal universitario: 1 semana

Términos y Condiciones

- Forma de Pago: 50% al inicio, 25% a la entrega de los módulos funcionales, 25% a la entrega final.
- Tiempo de Desarrollo: Aproximadamente 5 meses desde la fecha de inicio del proyecto.
- Mantenimiento: Un mes de soporte técnico post-lanzamiento, incluido en el costo.

Para cualquier aclaración, no dude en contactarnos. Nos complace la oportunidad de colaborar con **Licda. Maribel Lorenzo Linares** y contribuir al mejoramiento de su sistema administrativo.

Revisado por: Ing. Carlos García

Cell: 809-657-1414

Fecha:08/10/2024



A6.0 Informe Final Análisis Software Antiplagio

<https://api.justdone.ai/api/v1/plagiarism/pdf-report/90a422d6-b104-4702-a5b3-a7138265b88f>

justdone Analysis Report

Plagiarism Detection Report

JustDone AI

Scan details

Scan Time:
February 28th, 2025 at 21:32 UTC

Total Pages:
60

Total Words:
14869

Plagiarism Detection






Plagiarism Types	Text Coverage	Words
Identical	0%	0
Minor Changes	0.2%	24
Excluded		
Omitted Words		0

Plagiarism



Results (1)

 Repository N/A	 Internal Database 0
 Internet Sources 1	 Current Batch 0

Plagiarism Types	Text Coverage	Words
 Identical	0%	0
 Minor Changes	0.2%	24
Excluded		
 Omitted Words		0

About Plagiarism Detection

Our AI-powered plagiarism scans offer three layers of text similarity detection: Identical, Minor Changes, and Paraphrased. Based on your scan settings we also provide insight on how much of the text you are not scanning for plagiarism (Omitted words).

Identical

One to one exact word matches. [Learn more](#)

Minor Changes

Words that hold nearly the same meaning but have a change to their form (e.g. "large" becomes "largely"). [Learn more](#)

Omitted Words

The portion of text that is not being scanned for plagiarism based on the scan settings. (e.g. the 'Ignore quotations' setting is enabled and the document is 20% quotations making the omitted words percentage 20%) [Learn more](#)

Copyleaks Internal Database

Our Internal Database is a collection of millions of user-submitted documents that you can utilize as a scan resource and choose whether or not you would like to submit the file you are scanning into the Internal Database. [Learn more](#)

Filtered and Excluded Results

The report will generate a complete list of results. There is always the option to exclude specific results that are not relevant. Note, by unchecking certain results, the similarity percentage may change. [Learn more](#)

Current Batch Results

These are the results displayed from the collection, or batch, of files uploaded for a scan at the same time. [Learn more](#)

Plagiarism Detection Results: (1)

 Metodologías de desarrollo de software

0.2%

<https://repositorio.uca.edu.ar/bitstream/123456789/522/1/metodologias-desarrollo-software.pdf>

Maida

Maida, Esteban Gabriel ; Pacienza, Julián Metodologías de desarrollo de software Tesis de Licenciatura en Sistemas y Computación Facul...