

INSTITUTO POLITÉCNICO
NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO
ESCOM

Trabajo Terminal
"Título del trabajo terminal"
2026 – A062

Presentan
Nombre completo de los y las participantes

Directores
Dr. Velez y Lulu Grado y nombre completo

Mes y año

INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO
SUBDIRECCIÓN ACADÉMICA

No. de TT: 20 -
Fecha de presentación de TT
Documento técnico

"Título del trabajo terminal"
20 -

Presentan
Nombre completo de los y las participantes

Directores
Dr. Velez y Lulu Grado y nombre completo

Resumen

En este proyecto...

Palabras clave: , , ,

Carta Responsiva, sólo para TT2

Advertencia

"Este documento contiene información desarrollada por la Escuela Superior de Cómputo del Instituto Politécnico Nacional, a partir de datos y documentos con derecho de propiedad y por lo tanto, su uso quedará restringido a las aplicaciones que explícitamente se convengan."

La aplicación no convenida exime a la escuela su responsabilidad técnica y da lugar a las consecuencias legales que para tal efecto se determinen.

Información adicional sobre este reporte técnico podrá obtenerse en:

La Subdirección Académica de la Escuela Superior de Cómputo del Instituto Politécnico Nacional, situada en Av. Juan de Dios Bátiz s/n Teléfono: 57296000, extensión 52000.

Agradecimientos

Contents

1	Capítulo 1. Introducción	11
1.1	Antecedentes	11
1.2	Planteamiento del problema	11
1.3	Propuesta de solución	12
1.4	Objetivo general	13
1.5	Objetivos específicos	13
1.6	Justificación	13
1.7	Organización del documento	13
2	Capítulo 2. Estado del arte	15
2.1	Aplicación X	15
2.2	Artículo Y	15
2.3	Trabajo Terminal Z	15
2.4	Tabla comparativa	15
3	Capítulo 3. Marco Teórico	17
3.1	Criptografía	17
3.1.1	Algoritmos criptográficos	17
3.2	Lenguaje de programación	17
4	Capítulo 4. Análisis	18
4.1	Requerimientos	18
4.1.1	Requerimientos funcionales	18
4.1.2	Requerimientos no funcionales	18
4.1.3	Reglas de negocio	18
4.2	Análisis de las herramientas a utilizar	19
4.2.1	Análisis de los sistemas operativos	19
4.3	Análisis de riesgos	19
4.4	Análisis de factibilidad	20
4.4.1	Factibilidad técnica	20
4.4.2	Factibilidad operativa	20
4.4.3	Factibilidad económica	21
4.4.4	Factibilidad legal	22
4.5	Análisis de sostenibilidad	22
5	Capítulo 5. Diseño	23
5.1	Diagrama de arquitectura del software, sistema, aplicación	23
5.1.1	Diagrama de contexto	23
5.1.2	Diagrama de contenedores	23

5.1.3	Diagrama de componentes	23
5.2	Casos de uso	23
5.2.1	Diagrama de casos de uso	23
5.2.2	Actores	24
5.2.3	Especificación de Casos de Uso	24
5.3	Diagramas de secuencia	25
5.4	Diagramas de código	25
5.4.1	Diagramas de clases	25
5.4.2	Modelo de datos	26
6	Capítulo 6. Desarrollo e Implementación	27
6.1	Desarrollo del proyecto	27
6.2	Implementación	27
7	Capítulo 7. Pruebas	28
7.1	Generación del plan de pruebas	31
7.1.1	Construcción y ejecución de pruebas	31
7.1.2	Creación de casos y escenarios de prueba al sistema	31
7.1.3	Ejecución de casos y escenarios de prueba al sistema	31
8	Capítulo 8. Conclusiones y trabajo a futuro	34

List of Tables

List of Figures

Glosario de términos

Una lista con los términos más utilizados en el documento. En orden alfabético.

Término	Definición
Algoritmo	Conjunto ordenado y finito de operaciones que permite hallar la solución de un problema.
Almacenamiento virtual	Técnica que simula más memoria que la que realmente existe y permite a la computadora ejecutar varios programas simultáneamente.

Abreviaturas

Una lista con las abreviaturas más utilizadas en el documento. Ten en cuenta que personas de otras profesiones tendrán acceso a tu documento. En orden alfabético.

Abreviatura	Significado
RISC	Reduced Instruction Set Computer (conjunto de instrucciones reducidas)
SMTP	Simple Mail Transfer Protocol (Protocolo simple de transferencia de correo)

Chapter 1

Capítulo 1. Introducción

1.1 Antecedentes

Esta idea surgió a partir de nuestra propia experiencia, ya que, al tener 21 años, nos dimos cuenta de lo fácil que es perder el control sobre los gastos cotidianos. Ambos, al igual que muchos de nuestros amigos y personas cercanas, solíamos ignorar la importancia de llevar un registro detallado de nuestros ingresos y egresos. A medida que nos adentramos en la vida adulta, nos dimos cuenta de que la gestión de los gastos y el ahorro era algo que muchas veces se pasaba por alto, lo que provocaba una falta de conciencia sobre cuánto dinero realmente se estaba gastando.

Comenzamos a investigar el impacto que tenía la falta de seguimiento financiero, y cómo las tecnologías actuales podrían ayudarnos a simplificar el proceso. Observamos que, a pesar de contar con una gran cantidad de herramientas digitales disponibles, la mayoría de estas eran demasiado complejas o no ofrecían un enfoque claro y accesible para personas sin experiencia en finanzas. Entonces recordamos nuestras propias experiencias como estudiantes, donde las dificultades para ahorrar o manejar el dinero se volvían recurrentes.

Nos dimos cuenta de que había una desconexión entre las herramientas existentes y las necesidades reales de las personas, sobre todo de quienes están comenzando a tener responsabilidades financieras y que aún no tienen los conocimientos necesarios para optimizar sus finanzas. En nuestras conversaciones con amigos, colegas y familiares, descubrimos que la falta de un sistema práctico de seguimiento de gastos era un problema común, y que muchas personas sentían que no podían controlar sus finanzas debido a la falta de visibilidad y de herramientas efectivas. Esta experiencia compartida fue el motor que nos impulsó a crear algo que pudiera ayudar a las personas a tomar el control de sus finanzas desde una edad temprana.

1.2 Planteamiento del problema

En México, la gestión inadecuada de las finanzas personales es una preocupación creciente. Según la Encuesta Nacional sobre Salud Financiera (ENSAFI) 2023, solo el 53.2% de la población realiza algún tipo de registro o control de sus gastos, y de este porcentaje, únicamente el 32.8% cumple con dicho registro. [1] Este comportamiento refleja una falta de conciencia y disciplina financiera que puede llevar a decisiones económicas desfavorables.

Además, un estudio del Instituto Nacional de Estadística y Geografía (INEGI) revela que el 45.9% de la población menciona que casi nunca o nunca le sobra dinero al final

del mes, lo que indica una limitada capacidad de ahorro y planificación financiera [1]. Esta situación se ve agravada por la creciente preocupación por el futuro financiero; un informe de Statista muestra que un alto porcentaje de personas en diversos países están preocupadas por su situación financiera futura [2].

A pesar de la disponibilidad de herramientas digitales, el uso de aplicaciones de finanzas personales ha sido limitado. Según un informe de Adjust, aunque las instalaciones de aplicaciones de finanzas aumentaron un 50% en comparación con 2022, la retención de usuarios sigue siendo un desafío, con una disminución constante en la retención después del día 7 de uso [3]. Esto sugiere que, aunque existe interés inicial, los usuarios enfrentan barreras para mantener el uso continuo de estas herramientas.

Estos datos evidencian una necesidad urgente de soluciones que fomenten hábitos financieros saludables, mejoren la educación financiera y faciliten el seguimiento de los gastos personales. La falta de registros precisos y la escasa utilización de herramientas digitales eficaces contribuyen a una gestión financiera deficiente, afectando la estabilidad económica de los individuos.

1.3 Propuesta de solución

Para abordar las problemáticas identificadas en la sección anterior, se propone utilizar un conjunto de tecnologías digitales y algoritmos de análisis de datos que faciliten la gestión financiera personal de manera intuitiva y accesible. Las herramientas seleccionadas se centran en automatizar la captura y categorización de gastos, ofrecer análisis visuales y generar alertas y recomendaciones basadas en los hábitos del usuario.

En primer lugar, la digitalización de los tickets de compra mediante Reconocimiento Óptico de Caracteres (OCR, por sus siglas en inglés) permitirá capturar de manera automática la información de gastos, evitando la dependencia de registros manuales, los cuales según la ENSAFI 2023 solo realizan el 53.2% de los mexicanos [1]. Esto contribuiría a que los usuarios tengan un registro más preciso de sus gastos, reduciendo la omisión de información crítica. Por otra parte, el OCR al no ser 100% efectivo, los usuarios podrán ingresar o corregir los datos del ticket de manera manual si esta falla.

Además, la aplicación integrará algoritmos de categorización automática que clasificarán los gastos en diferentes rubros (alimentación, transporte, entretenimiento, etc.).

Para mejorar la comprensión de los gastos, se emplearán tableros interactivos y gráficos dinámicos, basados en bibliotecas de visualización como ECharts, D3.js u otras alternativas similares, que permitirán al usuario identificar patrones de gasto y tendencias a lo largo del tiempo. Estudios de la OCDE (2023) destacan que la visualización de datos financieros mejora la toma de decisiones y la comprensión de los hábitos de consumo, aumentando la probabilidad de ahorro hasta en un 20% [4].

De igual manera, para mejorar la comprensión de los gastos y permitir un seguimiento financiero más estructurado, se implementará un sistema de generación de reportes automáticos, tanto mensuales como anuales, que consolide la información capturada por la aplicación móvil y la plataforma web. Estos reportes incluirán tablas de resumen, gráficos interactivos y análisis comparativos de categorías de gasto, ingresos y ahorro. Además, los reportes pueden ser exportados a formatos comunes como PDF o Excel, facilitando la revisión histórica, el seguimiento de objetivos y la planificación de presupuestos futuros.

Finalmente, se implementará un sistema de notificaciones y alertas inteligentes que informen al usuario sobre sus límites de gasto o posibles desviaciones de su presupuesto.

En conjunto, estas tecnologías permiten atacar todas las problemáticas detectadas: la falta de registro de gastos, la escasa conciencia sobre hábitos financieros, la baja retención en el uso de aplicaciones y la dificultad de análisis financiero. La integración de OCR, visualización interactiva y alertas inteligentes crea un ecosistema digital que facilita el seguimiento y mejora la educación financiera del usuario.

1.4 Objetivo general

Desarrollar una aplicación y plataforma web interactiva para la gestión de finanzas personales que registre automáticamente los gastos de tickets físicos y digitales, categorizándolos para facilitar el análisis y seguimiento del presupuesto del usuario.

1.5 Objetivos específicos

- Desarrollar el módulo de escaneo de tickets físicos mediante la aplicación móvil.
- Desarrollar el módulo de escaneo de tickets digitales mediante la plataforma web.
- Implementar el módulo de categorización automática de gastos.
- Implementar el módulo de alertas y notificaciones financieras.
- Desarrollar el módulo de generación de informes y reportes de gastos mensuales, trimestrales y anuales.
- Desarrollar el módulo de análisis de gastos con gráficas interactivas y visualización de tendencias.
- Desarrollar el módulo de ingreso y corrección manual de datos de gastos.
- Implementar el módulo de creación de planes financieros y seguimiento de deudas.

1.6 Justificación

{Describir la justificación del proyecto, ¿por qué es importante realizar el proyecto? (No es porque va a resolver las problemáticas, no repetir las problemáticas ni la propuesta de solución). Describir ¿por qué utilizar las tecnologías propuestas? ¿Qué beneficios le traerá a los usuarios el resolver las problemáticas? Incluir referencias para sustentar sus argumentaciones} [Mínimo media cuartilla, máximo dos]

1.7 Organización del documento

{Una muy breve descripción de cada capítulo, un párrafo por cada capítulo, utilizar diferentes formas de redacción por cada capítulo. Se recomienda usar la taxonomía incremental de Marzano y Kendall}

En el capítulo 1, se describe la problemática junto con la ...

En el capítulo 2, se revisan los trabajos similares al proyecto ...

En el capítulo 3, se describen los conceptos básicos que se ...

En el capítulo 4, se analizan ...

} [Un párrafo por cada capítulo]

Chapter 2

Capítulo 2. Estado del arte

{Describir cada uno de los proyectos, artículos, aplicaciones, software y demás similares al proyecto. No necesariamente iguales, puede ser que resuelvan de forma parcial o total las problemáticas. Un capítulo por cada trabajo (organizados por tipo) mínimo 10 trabajos máximo 20}

2.1 Aplicación X

{Incluir una breve descripción de la aplicación, software, sistema, etc. Incluir las características, ventajas, desventajas, dónde se utiliza, estadísticas, opinión personal, se pueden incluir imágenes de la aplicación, software, sistema. Que se note que se probó la aplicación y a partir de ahí se obtuvo esa información no a partir de comentarios de terceros.} [Por cada trabajo, mínimo una cuartilla, máximo tres]

2.2 Artículo Y

{Incluir una breve descripción del artículo (no es el resumen del artículo). Incluir las características, ventajas, desventajas, estadísticas, opinión personal, se pueden incluir imágenes o gráficas con los resultados. Que se note que leyeron el artículo y a partir de ahí se obtuvo esa información no a partir de comentarios de terceros.} [Por cada artículo, mínimo una cuartilla, máximo tres]

2.3 Trabajo Terminal Z

{Incluir una breve descripción del TT (no es el resumen). Incluir las características, ventajas, desventajas, dónde se utilizaría, opinión personal, se pueden incluir imágenes o gráficas del TT, software, sistema. Que se note que se probó la aplicación y a partir de ahí se obtuvo esa información no a partir de comentarios de terceros.} [Por cada trabajo, mínimo una cuartilla, máximo tres]

2.4 Tabla comparativa

{Al final del capítulo incluir una tabla comparativa, comparando los trabajos descritos anteriormente, la tabla deberá comparar las características principales de todos los tra-

bajos con la propuesta. La tabla contendrá en las filas los trabajos similares (ordenados de la misma forma en que se presentaron en los capítulos anteriores y en las columnas enumerar las características más importantes. Las celdas de la tabla contendrán un **p** si contempla esa característica o un **y** si no la contempla. Evitar poner el precio o si es gratuita o no. Colocar la propuesta en la última fila, esta deberá contener en todas las celdas un **p** Ver el siguiente vídeo: https://youtu.be/I_lKPXJ05Wg

Título del trabajo	Característica 1	Característica 2	Característica 3	Característica 4
Trabajo 1	p	y	p	p
Trabajo 2	y	p	y	p
Trabajo 3	p	p	p	y
Trabajo 4	p	y	y	y
...	y	y	y	p
Propuesta	p	p	p	p

Tabla 1. Tabla comparativa entre las principales ...

{Al final, incluir una breve conclusión de esta comparativa. Máximo un párrafo.}

Chapter 3

Capítulo 3. Marco Teórico

{Incluir todas las definiciones importantes que se usarán en el trabajo, agrupadas por temática en cada subcapítulo. Todo el contenido debe estar referenciado, NO es copy/paste de la o las fuentes. Se tiene que redactar lo que entendieron de varias definiciones; por eso deben usar varias fuentes, no solo una. Las definiciones deben ir acompañadas de figuras, tablas, ilustraciones, gráficas, etc. bien referenciadas. Este capítulo debe abarcar el 35% del documento final}

{Este capítulo solo contiene definiciones, no incluir parte del análisis, como "por eso en este trabajo se optó por utilizar este o este algoritmo"}

3.1 Criptografía

De acuerdo con [X] la palabra criptografía proviene del latín

3.1.1 Algoritmos criptográficos

Según [Z] se pueden clasificar a los algoritmos

DES

El primer algoritmo criptográfico, de acuerdo con [Y], es

3.2 Lenguaje de programación

En [W] se define a un lenguaje de programación como ...

Chapter 4

Capítulo 4. Análisis

{Como su nombre lo indica, en este capítulo se debe incluir toda la fase de análisis. Todo debe ir acorde con la metodología utilizada}

4.1 Requerimientos

{Incluir los requerimientos funcionales, no funcionales y las reglas de negocio}

4.1.1 Requerimientos funcionales

{Son aquellos requerimientos que son esenciales para que funcione el sistema, si uno de los requerimientos aquí descritos no se realiza, el sistema no será funcional. Enumerar todos estos requerimientos con la siguiente nomenclatura: RF01, RF02, RF03, etc. Incluir una descripción de cada requerimiento. Comenzar cada requerimiento con: "El sistema permitirá ..."}
...

4.1.2 Requerimientos no funcionales

{Son aquellos requerimientos que no son esenciales para que funcione el sistema, pero hacen que el sistema funcione correctamente, sea más seguro, más rápido, se vea más bonito, etc. si uno de los requerimientos aquí descritos no se realiza, el sistema seguirá funcionando, pero puede que arroje datos erróneos, sea muy lento, no se vea bien, etc. Enumerar todos estos requerimientos con la siguiente nomenclatura: RNF01, RNF02, RNF03, etc. Incluir una descripción de cada requerimiento.}

RNF01 – La contraseña se debe guardar cifrada en la base de datos.

El sistema guardará las contraseñas de todos los usuarios de forma cifrada en la base de datos, de manera que no sea visible en caso de que haga la petición correspondiente.

RNF02 – El sistema ...
...

4.1.3 Reglas de negocio

{Las reglas de negocio son reglas que el programador impone al usuario, para que la funcionalidad del sistema se realice correctamente. El no acatar alguna de estas reglas

producirá mensajes de error al operar el sistema. Enumerar todas estas reglas con la siguiente nomenclatura: RN01, RN02, RN03, etc. Incluir una descripción de cada regla}

RN01 – Datos obligatorios

Al registrar los datos de su perfil de usuario, los siguientes son obligatorios: Nombre, apellidos, teléfono y código postal

RN02 – Número máximo de hijos registrados

Cada usuario puede registrar un máximo de 5 hijos al sistema.

RN03 - ...

...

4.2 Análisis de las herramientas a utilizar

{En esta sección deberá realizar un análisis (comparación) de las herramientas que utilizará al desarrollar el proyecto. Lenguaje de programación, sistema operativo, servicio en la nube, framework, etc. el análisis deberá estar basado en datos objetivos y reales, no a la percepción personal de los integrantes. Aquí NO incluir ni definiciones, ni historia, ni características, ni ventajas o desventajas. Todo eso va en el Marco teórico. Aquí únicamente se va a hacer una comparación. Pueden usar tablas comparativas o compararlas con una característica en particular. Toda la información debe ir referenciada}

4.2.1 Análisis de los sistemas operativos

Se analizaron los siguientes sistemas operativos:

Sistema operativo	Característica 1	Característica 2	Característica 3	Característica 4
Windows [X]	p	y	p	p
Linux [Y]	y	p	y	p
Solaris [Z]	p	p	p	y
OS-X [W]	p	y	y	y
...	y	y	y	p

Tabla 2. Análisis de los sistemas operativos

De acuerdo con la tabla 2, se optó por utilizar el sistema operativo Solaris; ya que ofrece ...

4.3 Análisis de riesgos

{Identificar todos los riesgos (peligros) que pueden ocurrir durante la realización del proyecto, realizar una evaluación de estos riesgos y aplicar las acciones pertinentes para reducir o eliminar la probabilidad de que se produzcan.

Se recomienda realizar una matriz de riesgos. Clasifica los riesgos en Bajos, Moderados, Altos y Extremos. Numéralos como R01, R02, R03, etc. Y agrúpalos por categorías.

Ejemplo de matriz de riesgos:}

Riesgo	Categoría	Probabilidad	Gestión del riesgo
R01 - Pérdida del código	Técnico	Alta	Realizar un respaldo de todo el código cada viernes a las 8:00 pm

Riesgo	Categoría	Probabilidad	Gestión del riesgo
R02 - Un integrante deja el equipo	Personal	Baja	Replantear los objetivos, No se re-alizarán los módulos menos importantes, como graficación y generación de reportes

4.4 Análisis de factibilidad

{Determinar la factibilidad técnica, económica, operativa y jurídica (en algunos casos) del proyecto. En este apartado se deberá establecer la duración y tamaño del proyecto, determinar costos y beneficios, analizar si la tecnología para realizar el proyecto existe y está al alcance de los integrantes, además en caso de que sea necesario, analizar las leyes o permisos que permiten desarrollar el proyecto}

4.4.1 Factibilidad técnica

{En esta sección se evaluará si el equipo y software están disponibles, son accesibles y tienen las capacidades técnicas requeridas para el diseño que se esté planificando.

Así mismo, se debe considerar si se cuenta con el personal que posee la experiencia técnica requerida para diseñar, implementar, operar y mantener el sistema, software o aplicación propuesta.}

4.4.2 Factibilidad operativa

{Aquí se debe determinar la posibilidad que el sistema, software, aplicación propuesta se utilice y, además, que ese uso sea el adecuado.

Se recomienda realizar un instrumento de recopilación de información (cuestionario, encuesta, entrevista) en donde se obtenga datos verídicos de usuarios potenciales que realmente estén dispuestos a utilizar el software, sistema, aplicación. Aquí se debe indicar el plan de aplicación de instrumentos (tipo y tamaño de muestra, insumos requeridos, periodo y momento de aplicación) y el tipo de instrumento que se va a utilizar.

En la sección de anexos se incluirá el instrumento completo, aquí se debe hacer un análisis de cada pregunta o grupo de preguntas; cada una en un subcapítulo junto con el resultado del análisis (un gráfica preferentemente y un párrafo con el análisis personal sobre los resultados obtenidos)}

Sobre si los adultos mayores conviven con animales de compañía.

Del total de la muestra, se analizaron los adultos mayores (personas con 60 años o más) y se obtuvo que, casi un 60% de ellos cuentan con algún animal de compañía (ver figura 4.W)

Figura 4.W. Porcentaje de adultos mayores que tienen animales de compañía en casa. Elaboración propia.

De esto se puede afirmar que un porcentaje muy alto de los adultos mayores encuestados podría utilizar la aplicación propuesta...

4.4.3 Factibilidad económica

{Aquí se incluye un análisis del costo y beneficios asociados con el proyecto. Se deben determinar los costos de los recursos tecnológicos, humanos y materiales tanto para el desarrollo como para la implantación del sistema, software, aplicación.

Los costos deben ir en moneda nacional y se deben incluir las referencias de dónde se obtuvieron esa información (estas referencias deben incluir la fecha de consulta)}

Recursos humanos

Para la realización del proyecto se contempla...

Núm.	Cargo	Costo individual	Costo total
1	Ing. Sistema (Líder del Proyecto)	\$ 24,000.00	\$ 24,000.00
2	Analista/Diseñador	\$ 19,200.00	\$ 38,400.00
1	Ingeniero del Software	\$ 16,800.00	\$ 16,800.00
1	Programador	\$ 14,440.00	\$ 14,440.00
Total			\$ 93,640.00

Tabla 4.Z. Recursos humanos necesarios para realizar el proyecto. Elaboración propia

Recursos tecnológicos

Para la realización del proyecto se contempla...

Cantidad	Descripción	Costo/hora	Total
Hardware			
2	140 horas Computadora HP Pavillion x360	\$ 16.00	\$ 2,240.00
1	Impresora Lexmark X3350 (depreciación 240/16*1)		\$ 3,000.00
Software			
1	Licencia Microsoft Office		\$ 800.00
Total			\$ 6,040.00

Tabla 4.W. Recursos tecnológicos necesarios para realizar el proyecto. Elaboración propia

Recursos materiales

Para la realización del proyecto se contempla...

Cantidad	Descripción	Costo	Total
1	Resma de Papel A4	\$ 60.00	\$ 60.00
2	Cartuchos para Impresora	\$ 300.00	\$ 600.00
40	Transporte a la empresa	\$ 10.00	\$ 400.00
20	Viáticos	\$ 100.00	\$ 2,000.00
Total			\$ 3,060.00

Tabla 4.Y. Recursos materiales necesarios para realizar el proyecto. Elaboración propia

Flujo de pago

El total requerido para

Recursos	Costo
Recursos Humanos	\$ 93,640.00
Recursos Tecnológicos	\$ 6,040.00
Recursos Materiales	\$ 3,060.00
Imprevistos (10%)	\$ 10,274.00
Total	\$ 113,014.00

Tabla 4.Y. Recursos materiales necesarios para realizar el proyecto. Elaboración propia

4.4.4 Factibilidad legal

{Si procede, de debe verificar si el proyecto por desarrollar no atenta o incumple alguna ley o norma de carácter municipal, estatal o mundial. De lo contrario no puede implementarse porque estará en contra de las disposiciones legales y por lo tanto no resultará viable.

El análisis debe incluir las referencias correspondientes}

4.5 Análisis de sostenibilidad

{Es imperativo que el desarrollo del proyecto y el sistema, software, aplicación cuenten con un análisis de sostenibilidad. Aquí se debe incluir, bien fundamentado, ese análisis.

Se debe incluir una evaluación del impacto ambiental para determinar cómo minimizar las repercusiones del proyecto sobre el medio ambiente.}

Chapter 5

Capítulo 5. Diseño

5.1 Diagrama de arquitectura del software, sistema, aplicación

{El diagrama de arquitectura presenta de manera gráfica una forma de comunicar cómo se planea construir un sistema, software, aplicación. Se recomienda seguir los primeros tres niveles del modelo C4 (Contexto, Contenedor, Componente y Código). <https://c4model.com/>}

5.1.1 Diagrama de contexto

{En este diagrama se debe mostrar el sistema, software, aplicación a desarrollar y sus interacciones en términos de las personas que lo utilizan y otros sistemas, software, aplicaciones con los que interactúa.}

5.1.2 Diagrama de contenedores

{Este diagrama amplía el sistema, software, aplicación; en él se muestran los diferentes contenedores con que se cuentan (aplicaciones, almacenamiento de datos, microservicios, etc.) que componen el sistema, software o aplicación.}

5.1.3 Diagrama de componentes

{Este diagrama expande un contenedor individual y muestra los componentes que contiene. Estos componentes se deben asignar a abstracciones reales en función de su código.}

5.2 Casos de uso

5.2.1 Diagrama de casos de uso

{El Diagrama debe seguir la notación para casos de uso establecida por UML, incluyendo los elementos del modelo de casos de uso: actores, casos de uso y relaciones.

Un caso de uso se define como un conjunto de acciones realizadas por el sistema que dan lugar a un resultado observable El caso de uso especifica un comportamiento que el sujeto puede realizar en colaboración con uno o más actores, pero sin hacer referencia a su estructura interna.}

5.2.2 Actores

{Un actor es cualquier entidad externa al sistema modelado que interactúa con él.

No necesariamente coincide con los usuario, pues un mismo usuario puede desempeñar distintos roles que correspondan con varios actores. Además, un mismo actor puede desempeñar varios papeles según el caso de uso con que interactúa.

Para cada uno de los actores involucrados en el documento y representados en el diagrama, debe completarse la siguiente ficha. Si existe más de un actor, se copia el título (Nombre del Actor) y la ficha tantas veces sea necesario.}

[Nombre de Actor 1]

Actor	[Nombre del Actor]
Identificador:	[Identificador único]
Descripción	[Breve descripción del Actor]
Características	[Características que describen al actor]
Relación	[Describe la relación de este actor con otros actores del sistema]
Referencias	[Elementos del desarrollo en los cuales este actor interviene, incluyendo por ejemplo Casos de Uso, Diagramas de Secuencia, entre otros.]

Atributos

Nombre	Descripción	Tipo

[En este cuadro se colocará un listado de los atributos principales del actor; por ejemplo, para un actor "Cliente" podría ser: Nombre, Apellido, Número de Identificación (DNI), y otros datos de interés]

Comentarios

[Aquí se incluirán comentarios adicionales sobre el actor]

5.2.3 Especificación de Casos de Uso

{Para cada uno de los casos de uso mostrados en los diagramas de caso de uso, se debe incluir una especificación completa del mismo.

La Especificación del caso de uso, describe la forma en que el actor interactúa con el sistema, listando las funciones o tareas realizadas, los datos de entrada, información que necesita recibir el actor del sistema, información sobre eventos o cambios inesperados, entre otros.

Repetir la siguiente ficha junto con los títulos para cuantos casos de uso se tengan en el modelo.}

[Nombre de Caso de Uso Nro. 1]

Caso de Uso	[Nombre del Caso de Uso]
Identificador:	[Del caso de uso]
Actores	[Listado de los actores que tienen participación en el caso de uso]
Tipo	[Tipo de caso de uso, primario, secundario, opcional]

Referencias	[Requerimientos o funcionalidades incluidas en este caso de uso. Casos de uso relacionados.]
Precondición	[Condiciones sobre el estado del sistema que deben cumplirse para iniciar el caso de uso]
Postcondición	[Efectos inmediatos que tienen la ejecución del caso de uso sobre el estado del sistema]
Descripción	[Descripción del caso de uso]
Resumen	[Resumen de alto nivel del funcionamiento]

Curso Normal

Nro.	Ejecutor	Paso o Actividad
[Nro. de paso]	[Actor ejecutor o especifica si es el sistema o subsistema]	[Descripción del paso actividad]

[Se describe el proceso o secuencia de pasos ejecutadas usando frases cortas] [Cada paso del proceso puede ser ejecutado por los Actores o por el sistema] [Se describe la secuencia de acciones realizadas por los actores y la secuencia de actividades realizada por el sistema como respuesta].

Cursos Alternos

Nro.	Descripción de acciones alternas
[Número de paso]	[Descripción de la secuencia de acciones alternas para el número de actividad indicado. Debe hacer referencia al número de paso en el curso normal]

[Cada paso descrito en el curso normal, puede tener actividades alternas, según la distribución de escenarios que ocurra en el flujo de procesos, en esta ficha se completa para cada actividad (haciendo referencia a su número) las posibles secuencias alternas]

5.3 Diagramas de secuencia

{Estos diagramas muestran cómo un grupo de objetos se interactúan o se comunican entre sí a lo largo del tiempo. Los objetos se muestran mediante líneas verticales y los mensajes como flechas conectando objetos. Los mensajes son dibujados cronológicamente desde arriba hacia abajo. Los rectángulos en las líneas verticales representan los periodos de actividad de los objetos}

5.4 Diagramas de código

{El nivel 4 del modelo C4, en estos diagramas se amplía cada componente individual mostrando la forma en que se implementan.}

5.4.1 Diagramas de clases

{En estos diagramas se trazan de manera clara la estructura del sistema, software, aplicación al modelar sus clases, atributos, operaciones y relaciones entre objetos}

5.4.2 Modelo de datos

{Se debe incluir un modelo gráfico de las diferentes bases de datos que se usarán en el proyecto. Diagrama Entidad-Relación si se utilizan bases de datos relacionales. Diagramas de árbol si se utilizan bases de datos jerárquicas}

Chapter 6

Capítulo 6. Desarrollo e Implementación

6.1 Desarrollo del proyecto

{Se deben incluir los pasos necesarios, con una explicación detallada, del procedimiento que se utilizó para la realización de cada módulo del proyecto. Un subcapítulo por cada módulo o prototipo}

6.2 Implementación

{Incluir una explicación de cómo y dónde se implementó el proyecto terminado. Incluir las características que tendrá la plataforma, servidor, etc. En donde se montó el proyecto}

Chapter 7

Capítulo 7. Pruebas

{Incluir un plan de pruebas con todas las pruebas necesarias. Se deben probar los requerimientos funcionales y no funcionales}

{Se debe especificar el tipo de prueba que se va a documentar, pudiendo ser, funcional, integral, carga, rendimiento o de seguridad.}

{Se puede usar la siguiente plantilla}

Registro de revisiones

Versión	Fecha	Autor	Revisor	Observaciones

MATRIZ DE DATOS – ESCENARIO DE PRUEBA [FUNCIONAL, INTEGRAL, CARGA, RENDIMIENTO, SEGURIDAD]

[ACRÓNIMO DEL SISTEMA] M/[No. De Módulo] - CU/[No. De Caso de Uso] [Nombre del caso de prueba]

INFORMACIÓN DETALLADA

Identificador del caso de prueba	[ACRÓNIMO] - M/[No. De módulo] – CU/[No. De Caso de – ECP/[No. Del escenario del caso de prueba] –[Nombre del de prueba] – [Nombre del escenario] CC – M/01 - CU/001 -ECP/01 - Producto– Agregar Producto CC – M/01 - CU/001 -ECP/02 - Producto– Eliminar Producto
Ruta	Ruta correspondiente para la ejecución del escenario de prueba. El usuario inicia sesión, accede al módulo de productos y da clic a agregar producto
Precondiciones	Precondiciones que se deben cumplir antes de iniciar la ejecución del escenario del caso de prueba
Versión del sistema	Versión del sistema que se va a probar

ID	Acción	Datos de entrada	Tipo de prueba	Resultado esperado	Resultado final	Corrección

ID	Acción	Datos de entrada	Tipo de prueba	Resultado esperado	Resultado final	Corrección
[ACRÓNIMO - M/[No. De módulo] – CU/[No. De Caso de Uso] – ECP/[No. Del escenario del caso de prueba] – C [No. De caso]	{Descripción de lo que se realizará en el escenario}	{Información que se utilizará en campos donde se requiera información}	[Optimista / No optimista]	{Descripción del resultado esperado al ejecutar el escenario de prueba}	[Correcto/Incorrecto]	[SI/NO/APLICA] {En caso de no aplicar, es importante una breve descripción}

{Definición de los principales elementos en esta plantilla:

Registro de versiones: Contendrá información relacionada con la administración del documento, se especificarán fechas, autores del documento (estudiantes), persona que aprobó el mismo (director o directora) y observaciones si fueran necesarias.

Matriz de datos: Se debe especificar el tipo de prueba que se va a documentar, pudiendo ser, funcional, integral, carga, rendimiento o de seguridad.

No. De Módulo: Correspondiente a la clasificación de los casos de uso, requisitos o clasificación que se tenga del sistema. En caso de no existir, el o la estudiante podrá proponer una numeración.

No. De Caso de Uso: Usar el mismo identificador que se usó en el apartado 5.2.2.

Nombre del caso de prueba: Nombre que se utilizará para identificar el caso de prueba, podrá ser propuesto por el o la estudiante.

Ruta: Para ejecutar un escenario de prueba específico, es necesario indicar cuál fue la ruta que siguió para ejecutar el experimento, con la finalidad de reducir tiempos y poder replicar en cualquier momento el escenario.

Precondiciones: Un aspecto relevante en la ejecución de pruebas son las precondiciones, en este apartado debe especificarse cuáles serán las actividades previas que se deben cumplir para ejecutar un escenario de pruebas. En caso de no cumplirse con alguna de ellas, no se podrán llevar a cabo las validaciones.

Versión del sistema: Toda aplicación debe tener una versión estable a la cual se le aplicarán las pruebas, es indispensable que las o los estudiantes indiquen a cual se deberán hacer todos los experimentos necesarios con la finalidad de detectar errores.

ID: Una vez que se tiene la información fundamental, se crearan los escenarios específicos con cada combinación necesaria para probar el sistema. Para ello se requiere tenga un identificador único para un seguimiento correcto.

[ACRÓNIMO DEL SISTEMA] Siglas que identifican el proyecto.

M/[No. De Módulo] Número del módulo correspondiente a la clasificación del sistema.

CU/[No. De Caso de Uso] Número del caso de uso del apartado 5.2.2.

[Nombre del caso de prueba] Se recomienda que el nombre a utilizar sea significativo al módulo que se está probando.

ECP/[No De caso de Prueba] Se recomienda utilizar numeración consecutiva para futuras consultas.

[Nombre del escenario de prueba] El nombre del escenario hace referencia a la operación específica que se va a realizar. Por ello es indispensable sea significativo el nombre que se ocupe.

C/[No. De caso] Este número será consecutivo para los experimentos que se tengan en el escenario de prueba.

Acción: Se debe describir de forma concreta cual será la acción que se llevará a cabo en el escenario, si es necesario realizar más de un paso, deberá describirse y numerarse para evitar confusiones.

Datos de entrada: Para toda acción se requiere información para ejecutar el escenario, así como se enumeraron los pasos, se deberán especificar los nombres y los valores que se ocuparán en cada campo, con la finalidad de evitar confusiones y realizar las acciones de forma correcta.

Tipo de prueba: Existen dos tipos de pruebas que se deben realizar en cada escenario, las optimistas y las no optimistas. Las pruebas optimistas son aquellas en la que se espera un resultado satisfactorio al ejecutarla, las pruebas no optimistas son aquellas en las que se espera un error al aplicar una acción.

Resultado esperado: debe describirse de manera concreta cuál será la consecuencia después de haber ejecutado el escenario, esto implica describir uno o más datos resultantes que arrojará la acción.

Resultado final: Para este punto existen dos estados, correcto e incorrecto. El resultado correcto será solo si la prueba fue satisfactoria en su totalidad, caso contrario tendrá que colocarse incorrecto y será un error que se tendrá que corregir antes de la presentación del TT.

Corregido: Finalmente, la última columna de la tabla se utilizará y actualizará cuando se haya realizado la corrección del error. Se tendrá que volver a ejecutar el escenario para validar que fue corregido de forma adecuada, caso contrario se tendrá que volver a corregir. Para este apartado existen tres estados: SI, NO y NO APLICA. En el caso de SÍ, se aplica cuando el defecto fue solucionado, caso contrario se utilizará NO. Para NO APLICA, se utilizará cuando el error que se registró no es un error, es decir, que la funcionalidad o resultado de la acción es correcta y no debe modificarse su funcionamiento, esto puede llegar a ocurrir por confusión por parte del o la estudiante al no tener bien definido y entendido los requisitos del sistema.}

7.1 Generación del plan de pruebas

{Esta etapa gobernara toda la etapa de pruebas}

Registro de revisiones

Versión	Fecha	Autor	Revisor	Observaciones
1.0	dd/mm/aaaa	Benjamín Cruz (estudiante)	Benjamín Cruz (director o directora)	Desarrollo de sesión de datos
1.1	dd/mm/aaaa	Benjamín Cruz (estudiante)	Benjamín Cruz (director o directora)	Desarrollo de pruebas

7.1.1 Construcción y ejecución de pruebas

{Generar una lista de los componentes que se van a probar, hay que tomar en cuenta los requerimientos, así como los módulos de los que estará conformado el sistema para poder obtener dicha información. Clasificar las pruebas en: pruebas funcionales, pruebas integrales, pruebas de carga y pruebas de seguridad. Verificar que no haya casos de prueba redundantes en la misma clasificación.}

Componentes que se probaron	Tipo de prueba que se realizó
Módulo de registro de usuario	Funcional
Módulo de inicio de sesión	Funcional, de seguridad
Módulo de recuperación de contraseña	Integral

7.1.2 Creación de casos y escenarios de prueba al sistema

{Una vez identificados y clasificados los casos de prueba, se procede a crear los escenarios. Primero se describen los casos de prueba utilizados para la ejecución de los experimentos y la nomenclatura para el nombrado de archivos.}

Nombre del documento	Nombre del caso de prueba
PR – 01 – Pruebas _funcionales – Inicio de sesión	PR – M/01 – CU/02 – Inicio de sesión
PR – 02 – Pruebas _de _seguridad – Inicio de sesión	PR – M/01 – CU/02 – Seguridad del Inicio de sesión
PR – 03 – Pruebas _funcionales – Registro de usuario	PR – M/01 – CU/01 – Registro de usuarios

7.1.3 Ejecución de casos y escenarios de prueba al sistema

{Una vez que se tengan los escenarios, el siguiente paso es ejecutar las pruebas a cada uno de los módulos existentes en el sistema}

ID	Acción	Datos de entrada	Tipo de prueba	Resultado esperado	Resultado final	Corrección

ID	Acción	Datos de entrada	Tipo de prueba	Resultado esperado	Resultado final	Corrección
PR – M/01 – CU/01 – ECP/01 – C 01	Se intentará iniciar sesión con un usuario registrado y una contraseña correcta	Usuario: "benji@ipn.mx" Contraseña: "miPass-word123"	Optimista	El sistema muestra un mensaje de bienvenida	Correcto	NO APLICA Se obtuvo el comportamiento esperado
PR – M/01 – CU/01 – ECP/01 – C 02	Se intentará iniciar sesión con un usuario registrado y una contraseña incorrecta	Usuario: "benji@ipn.mx" Contraseña: "OtroPasss-word789"	No Optimista	El sistema muestra un mensaje de error y no se tiene acceso al sistema	Correcto	NO APLICA Se obtuvo el comportamiento esperado
PR – M/01 – CU/01 – ECP/01 – C 02	Se intentará iniciar sesión con un usuario no registrado y una contraseña incorrecta	Usuario: "nadie@ipn.mx" Contraseña: "miPasss-word567"	No Optimista	El sistema muestra un mensaje de error y no se tiene acceso al sistema	Correcto	NO APLICA Se obtuvo el comportamiento esperado

La fase de pruebas es una de los más importantes, aquí será donde se generarán todas las posibles combinaciones de los campos del sistema para poder identificar alguna anomalía o mal funcionamiento. Para ello es importante tomar en cuenta los siguientes aspectos en cuanto a campos a probar:

Probar las longitudes máximas y mínimas que se establecieron en los requisitos y base de datos del sistema. Por ejemplo, si un campo está determinado por aceptar un mínimo de 2 caracteres y un máximo de 10, las pruebas abarcarían probar dicho campo con 1 carácter y con 11 caracteres para validar que en realidad esta longitud no la acepta.

Probar con diferentes tipos de dato, es decir, números enteros, números decimales, números negativos, letras, caracteres especiales, etc., para verificar que no tenga problemas con el tipo de codificación con el que fue programado, algunos ejemplos de ello son los siguientes *\$<!"'"/;¿#\$\$%&()[].; 123457 1.23 -12.

Si existieran campos opcionales, validar al igual que los obligatorios si al dejarlos en blanco el sistema realiza la operación o marca alguna excepción.

Los campos en los cuales se realizarán cálculos, introducir valores incorrectos, números exponenciales, o con signos negativos con la finalidad de evaluar el tipo de valor que acepta y que puede trabajar dicho campo.

Verificar en campos donde pide cargar imágenes los diferentes formatos como, .png, .jpg, .gif, .nmp, .jpeg, .tif, .tiff, etc., así como el tamaño y resolución de estas que pudieran

afectar el almacenamiento de este tipo de archivos.

En caso de que se requieran adjuntar documentos se deberán validar extensiones como, .doc, .xlsx, .xls, .pdf, .doctx, .dot, .rtf, .txt, .htm, .docm, .xml, .thmx o algún otro archivo con extensiones diferentes a las que pide, esto para validar que el campo en realidad acepta solo lo señalado.

En caso de tener vínculos, es importante validar que apunte y direcciona a la ruta correcta según lo descrito por el enlace, así como identificar la ausencia de alguno.

Si el sistema permite descargar archivos como imágenes, documentos o carpetas, es importante verificar la legibilidad y que el documento completo se haya descargado, así como validar que en realidad descargue el documento que menciona.

Si el sistema permite realizar impresión de documentos, es indispensable verificar que el documento resultante cumpla con lo que el sistema mostraba en su versión digital.

Para las pruebas de carga, verificar la cantidad de usuarios que pueden ingresar al sistema al mismo tiempo, esto con la finalidad de validar el número total o estimado que puede soportar el sistema.

El tiempo de respuesta es un factor relevante e importante, por lo que se debe validar que el sistema no tarde más de .10 o .3 segundos para dar respuesta, caso contrario puede resultar inoportuno para el usuario.

Es importante que cada función y acción dentro del sistema tenga un éxito o fallo, por lo que se debe validar en primera instancia la acción y en segundo que el mensaje corresponda a la operación que se está realizando

Chapter 8

Capítulo 8. Conclusiones y trabajo a futuro

{Las conclusiones no son un resumen del proyecto. Una conclusión de un proyecto es la revisión reflexiva de los resultados de este. Se trata de un conjunto de ideas sintetizadas que explican de manera clara y directa las soluciones a los problemas planteados antes y durante la ejecución del proyecto.

En otras palabras, las conclusiones del proyecto recogen las respuestas a las preguntas que, tanto el equipo como el Director y los involucrados en el proyecto, se han ido planteando desde que se decidiera a comenzar un proyecto determinado.}

Referencias

{En formato IEEE o APA. Se recomienda usar la herramienta de estilos en Word}

Anexos

{Incluir todos los contenidos que sirven para ampliar la información presentada, pero sin resultar imprescindibles para la comprensión del proyecto. Si bien los anexos constituyen un complemento para la investigación, su inclusión se considera un valor agregado, ya que aportan datos relevantes que no están mencionados en el cuerpo del documento.}