

Entrega

Enviado para calificar

Calificado

La tarea fue enviada 23 horas 12 mins después de la fecha límite

Los estudiantes pueden editar este envío

 [Exposición U2 Gestión del tiempo para el desarrollo de Sistemas de Información.pdf](#)

25 de marzo de 2025, 23:12

U2

[Comentarios \(0\)](#)

### Calidad en los Sistemas de Información U2

Calificación:

Rúbrica			
Formalidad de la presentación. <ul style="list-style-type: none"><li>Al inicio de la presentación se realizó la presentación individual de cada integrante del equipo.</li><li>Al inicio de la exposición, el equipo presentó el tema y dio una síntesis de la temática que se iba a abordar.</li></ul>	No cumple	Cumple parcialmente	Cumple adecuadamente
	<b>0 puntos</b>	<b>0 puntos</b>	<b>1 puntos</b>
Dominio del tema (por equipo). <ul style="list-style-type: none"><li>Todos los integrantes hablan con fluidez y demuestran conocimiento del tema.</li><li>Todos los integrantes manejan los materiales y recursos visuales presentados.</li></ul>	No cumple	Cumple parcialmente	Cumple adecuadamente
	<b>0 puntos</b>	<b>4 puntos</b>	<b>8 puntos</b>
Habilidades comunicativas (individual). <ul style="list-style-type: none"><li>Modula el tono de su voz, además es posible que toda la audiencia (compañeros de clase) lo</li></ul>	cumple solo	Cumple parcialmente	Cumple adecuadamente

<p>escuche. • Es capaz de hablar de forma natural y sin titubeos, haciendo fluido su mensaje. • Utiliza una velocidad adecuada para expresarse. • Utiliza una postura corporal adecuada, en la que muestra seguridad de lo que está hablando. • Crea contacto visual con su audiencia para referirse y explicar su mensaje.</p>	<p>algunos aspectos <b>5 puntos</b></p>	<p><b>10 puntos</b></p>	<p><b>15 puntos</b></p>
<p>Tiempo y forma. • La exposición se presentó el día y la hora establecidos. • Se respetó el tiempo de duración de la exposición.</p>	<p>No cumple <b>0 puntos</b></p>	<p>Cumple parcialmente <b>1 puntos</b></p>	<p>Cumple adecuadamente <b>2 puntos</b></p>
<p>PRESENTACIÓN ELECTRÓNICA Uso pertinente. • El material presentando sirve de apoyo para la exposición del tema o proyecto.</p>	<p>No cumple <b>0 puntos</b></p>	<p>Cumple parcialmente <b>1 puntos</b></p>	<p>Cumple adecuadamente <b>2 puntos</b></p>
<p>Elementos de diseño. • Se utiliza un fondo en color adecuado que permite resaltar la información e imágenes presentadas. • Toda la presentación utiliza colores apropiados para hacer legible el contenido. • La presentación utiliza una fuente de letra legible para leer el contenido, además utiliza un tamaño proporcional. • Se hace uso de imágenes como apoyo y refuerzo de la información presentada.</p>	<p>No cumple <b>0 puntos</b></p>	<p>Cumple adecuadamente <b>4 puntos</b></p>	<p>Cumple parcialmente <b>8 puntos</b></p>
<p>Contenido de la presentación • La presentación al inicio incluye una portada, con el nombre o los nombres de los participantes y se incluye el tema que se aborda. • Es posible visualizar una estructura en la temática, abordando los puntos de manera ordenada. • En cada lámina de la presentación</p>	<p>No cumple <b>0 puntos</b></p>	<p>Cumple parcialmente <b>2 puntos</b></p>	<p>Cumple adecuadamente <b>4 puntos</b></p>

se incluye un subtítulo. • Al final de la presentación se integran las referencias bibliográficas que fueron utilizadas para construir la temática.

Calificación actual en el libro

36.00

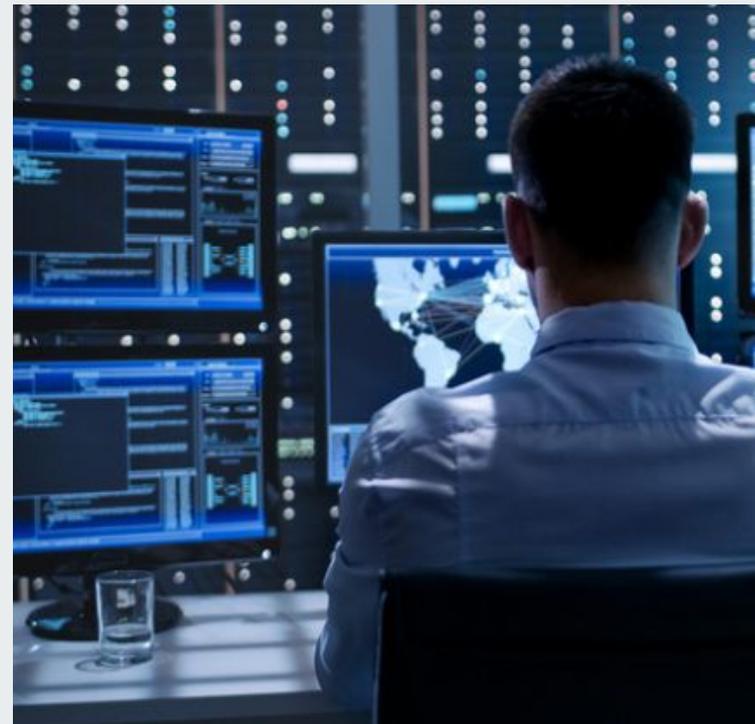


# Gestión del tiempo para el desarrollo de Sistemas de Información

*Calidad en los Sistemas de Información*  
MTI. Guadalupe Zetina Cruz



Presentan:  
J Díaz, G Caixba, A Montán y  
R Paxtian.  
Grupo: 610-A



# INTRODUCCIÓN



El tiempo en el ciclo de vida de un proyecto de desarrollo de sistemas de información, es sumamente importante, por lo que hay que mantener vigilado en todo momento los plazos y cumplimientos de las actividades necesarias para el desarrollo de estos.

Según [1], existen tres restricciones en un proyecto, las cuales son: **Tiempo, Costo y Alcance**:

Siendo estos los factores importantes y **no negociables** una vez realizada la acta de constitución, pues no podemos sobrepasar estos valores, los cuales una vez violados, se transformarán en **multas** para la empresa que realiza el proyecto, siendo nosotros los administradores de proyectos los principales responsables, lo que nos puede llevar desde un regaño, una multa económica o perder nuestro puesto hasta incluso, **cárcel**.

# ¿QUÉ ES LA GESTIÓN DEL TIEMPO EN PROYECTOS EN DESARROLLO DE SISTEMAS DE INFORMACIÓN?



Se trata de la adopción de técnicas, metodologías y herramientas las cuales nos permiten mantener el control en el plazo de entrega de nuestros proyectos, logrando realizar el **alcance**, el **tiempo**, **costo** y **calidad** esperada por nuestros clientes.

Para poder cumplir los objetivos en tiempo y forma, se necesita:

- **Definir el proyecto:** Establecer objetivos, estructura y planificación general.
- **Planificar tareas y recursos:** Identificar actividades, asignar recursos y fijar plazos.
- **Gestionar costos y seguimiento:** Establecer costos, compararlos con el presupuesto y analizar avances.
- **Coordinar y optimizar:** Relacionar tareas, prever problemas y aplicar acciones de corrección.
- **Comunicación en equipo:** Involucrar al equipo en la planificación y resolución de problemas, con instrucciones e informes claros.

# Metodologías y técnicas para la gestión del tiempo en el desarrollo de SI

Métodos:



- ❖ **Timeboxing:** Este método ayudará a dividir las tareas más complejas en otras más sencillas para facilitar el trabajo.
- ❖ **‘Getting Things Done’:** Lo primero que se debe hacer es escribir todas tus tareas pendientes.
- ❖ **No lo dejes para mañana:** Este método consiste en hacer lo peor al principio. Al llegar del trabajo, se debe encargarse de las tareas que son extensas o complejas.
- ❖ **El principio de Pareto:** Nos sugiere que realicemos primero las tareas más sencillas o fáciles.



## EJEMPLO

Un equipo de TI gestiona un sistema de información para la gestión de clientes

- ❖ El equipo define bloques de tiempo de 2 horas para tareas específicas.
- ❖ Cada mañana, el equipo revisa una lista de tareas pendientes: actualización de seguridad, corrección de errores y optimización del rendimiento.
- ❖ Inician el día con las tareas más críticas, como solucionar una falla en la base de datos que afecta a los clientes.
- ❖ Identifican que el 20% de los errores en el sistema son responsables del 80% de los problemas de rendimiento. Se enfocan en corregir estos errores primero.

## Técnicas para la gestión del tiempo:

- ❖ Establecer prioridades: Crear una lista de tareas diarias y marcar las más importantes.
- ❖ Planificación de tareas: Emplear herramientas de gestión de tiempo puede ayudar a poner orden.
- ❖ Eliminar distracciones: Pueden emplearse aplicaciones de bloqueo para redes sociales u otras distracciones con el fin de minimizarlas.
- ❖ Aprender a decir no: Hay que reconocer los límites y aprender a rechazar tareas adicionales que puedan afectar a la carga de trabajo actual.



## **EJEMPLO**

Un equipo de TI gestiona un sistema de información para la gestión de clientes

- ❖ El equipo crea una lista diaria de tareas y prioriza las más importantes.
- ❖ Utilizan herramientas como Trello para asignar responsabilidades.
- ❖ Los desarrolladores bloquean redes sociales y notificaciones innecesarias mientras trabajan.
- ❖ Un gerente de TI rechaza solicitudes no urgentes, como cambios menores en la interfaz.



# Herramientas y tecnologías para la optimización del tiempo en el desarrollo de SI

- ❖ **Scoro:** Se centraliza la administración de tareas, seguimiento de tiempos y control financiero en un solo lugar.
- ❖ **ProofHub:** Permite organizar tareas, hacer seguimiento del tiempo y facilitar la comunicación entre miembros del equipo.
- ❖ **Bitrix24:** Administra proyectos, comunica equipos y organiza el tiempo de forma eficiente en un entorno colaborativo.
- ❖ **Replicon:** Supervisa las horas laborales, administra proyectos y asegura el cumplimiento de normativas laborales.
- ❖ **Teamwork:** Asigna tareas, realiza seguimiento de tiempo y mejora la colaboración en proyectos complejos.



## Ejemplo Sistema de Información Hospital

- ❖ Se usa Scoro para definir las fases del proyecto, los gerentes asignan tareas a cada equipo, establecen los tiempos estimados para cada entrega
- ❖ Crear tableros con las tareas asignadas y sus fechas límite. Se configuran alertas y recordatorios para evitar retrasos.
- ❖ Se usa la función de chats y videollamadas para reuniones rápidas sobre integración de módulos, documentan errores encontrados y notifica a los desarrolladores.
- ❖ Cada desarrollador registra sus horas de trabajo. Se usa para asegurarse de que el personal respete horarios laborales y descansos.
- ❖ Dividir del proyecto en módulos: gestión de pacientes, administración de citas, facturación y reportes médicos. Se asignan responsables por módulo.



La gestión del tiempo es una habilidad crucial para cualquier desarrollador de sistemas, ya que puede afectar la calidad, el costo y la entrega de sus proyectos.



Herramientas para la gestión del tiempo pueden ser:

- ❖ Calendario
- ❖ Reloj registrador online
- ❖ Aplicación para tomar notas
- ❖ Seguimiento de hábitos
- ❖ Herramientas de prevención de distracciones
- ❖ Alertas y recordatorios



# CASOS DE ÉXITO Y BUENAS PRÁCTICAS EN LA GESTIÓN DEL TIEMPO PARA EL DESARROLLO DE SI

Un **caso de éxito** en la gestión del tiempo en el desarrollo de un SI es aquel en el que el proyecto se lleva a cabo dentro de los plazos establecidos, cumpliendo con los objetivos definidos, sin comprometer la calidad del producto final.

Una buena práctica en la gestión del tiempo en el desarrollo de un SI incluye:

- **Definición clara de objetivos:** Establecer metas y plazos claros desde el inicio del proyecto.
- **Planificación detallada:** Crear un cronograma realista, desglosando el proyecto en etapas y asignando tiempos adecuados para cada tarea.
- **Priorización de tareas:** Identificar las tareas más críticas y urgentes para abordarlas primero, evitando retrasos que afecten el cronograma general.
- **Monitoreo continuo:** Realizar revisiones periódicas para evaluar el progreso y realizar ajustes si es necesario.
- **Gestión de riesgos :** Prever posibles problemas y contar con planes de contingencia para evitar demoras importantes.
- **Comunicación efectiva :** Mantener una comunicación constante entre los miembros del equipo y las partes aceptadas para garantizar que todos estén alineados y resuelvan problemas rápidamente.



## Caso de éxito:

Un equipo logró cumplir un plazo ajustado para la entrega de un **sistema de gestión** de inventarios mediante la utilización de técnicas ágiles. La **clave** del éxito fue que el equipo ya tenía un conocimiento del sistema, lo que permitió una mejor estimación del tiempo necesario y una priorización eficiente de tareas.

### Buenas prácticas:

- **Planificación detallada:** Antes de comenzar, se definieron claramente los requisitos, se desglosaron las tareas por fases y se asignaron tiempos para cada una.
- **Uso de metodologías ágiles:** Las reuniones diarias de seguimiento y la iteración constante aseguran que cualquier desviación fuera detectada temprano.
- **Revisión continua de la calidad:** Se realizaron pruebas continuas para asegurar que el producto final cumpliera con los estándares sin retrasar el tiempo de entrega.



# Referencias



[1] “10 Herramientas esenciales para la gestión del tiempo”. Bitrix24 ist ein kostenfreier Online-Arbeitsplatz für Ihr Unternehmen: CRM, Aufgaben, Online-Termine und vieles mehr. Accedido el 28 de febrero de 2025. [En línea]. Disponible: <https://www.bitrix24.mx/articles/10-herramientas-esenciales-para-la-gestion-del-tiempo.php>

[2] Repsol. “Gestión del tiempo: qué es, beneficios y técnicas”. REPSOL. Accedido el 28 de febrero de 2025. [En línea]. Disponible: <https://www.repsol.com/es/energia-futuro/personas/gestion-del-tiempo/index.cshtml>

[3] *Time Management: How It Is Important for Project Delivery*. (s.f.). Resources Library. [https://www-saviom-com.translate.google/blog/time-management-how-it-is-important-for-project-delivery/?\\_x\\_tr\\_s=I=en&\\_x\\_tr\\_tl=es&\\_x\\_tr\\_hl=es&\\_x\\_tr\\_pto=sge#:~:text=Una%20gesti3n%20adecuada%20del%20tiempo,evitar%20demoras%20en%20los%20proyectos.](https://www-saviom-com.translate.google/blog/time-management-how-it-is-important-for-project-delivery/?_x_tr_s=I=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=sge#:~:text=Una%20gesti3n%20adecuada%20del%20tiempo,evitar%20demoras%20en%20los%20proyectos.)

Entrega

Enviado para calificar

Calificado

La tarea fue enviada 17 horas 57 mins después de la fecha límite

Los estudiantes pueden editar este envío

 [Resumen-CalidadSI U2.pdf](#)

26 de marzo de 2025, 17:57

[Comentarios \(0\)](#)

Calidad en los Sistemas de Información u2

Calificación:

<b>Rúbrica</b>			
Portada • El documento cuenta con hoja de presentación que incluye: nombre del alumno, nombre del docente, nombre de la signatura, nombre de la actividad, periodo escolar y fecha de entrega.	No cumple	Cumple parcialmente	Cumple adecuadamente
	<b>0 puntos</b>	<b>1 puntos</b>	<b>2 puntos</b>
Estructura • Se incluyen separadores para identificar con facilidad los diferentes temas del trabajo. • Se adjunta un índice para poder visualizar y localizar fácilmente los temas	No cumple	Cumple parcialmente	Cumple adecuadamente
	<b>0 puntos</b>	<b>2 puntos</b>	<b>4 puntos</b>
Organización. • Respeta la estructura y organización del texto original. (inicio, desarrollo y conclusiones). • Las ideas principales se presentan con orden y congruencia.	No cumple	Cumple parcialmente	Cumple adecuadamente
	<b>0 puntos</b>	<b>2 puntos</b>	<b>4 puntos</b>

Contenido • El documento cumple con los temas solicitados. • Recaba de manera coherente lo esencial del texto original. (No fragmenta las ideas). • Se presentan las ideas originales del autor. No presenta juicios de valor e interpretaciones. • De manera general el escrito se presenta completo y preciso.

No cumple	Cumple parcialmente	Cumple adecuadamente
<b>0 puntos</b>	<b>6 puntos</b>	<b>12 puntos</b>

Comprensión del tema • Se observa que el alumno comprendió el texto original y puede plasmar las ideas más significativas. (Capacidad de síntesis).

No cumple	Cumple parcialmente	Cumple adecuadamente
<b>0 puntos</b>	<b>1 puntos</b>	<b>2 puntos</b>

Participación. • El alumno realiza aportaciones en plenaria grupal y puede expresar con claridad y coherencia los temas indicados

No cumple	Cumple parcialmente	Cumple adecuadamente
<b>0 puntos</b>	<b>3 puntos</b>	<b>6 puntos</b>

Calificación actual en el libro

28.00



INSTITUTO TECNOLÓGICO SUPERIOR DE  
SAN ANDRÉS TUXTLA

# INSTITUTO TECNOLÓGICO SUPERIOR DE SAN ANDRÉS TUXTLA

---

---

## INGENIERÍA INFORMÁTICA

“Calidad enfocada al Desarrollo de Sistemas de Información ”

# RESÚMEN

Calidad en los Sistemas de Información

Presentan

**EMMANUEL DE JESÚS TEOBAL DÍAZ**  
**MARÍA GRISEL CAIXBA HERRERA**  
**RAFAEL PAXTIAN CAMPECHANO**  
**ANNETTE MONTAN MARTÍNEZ**

Profesor

*Guadalupe Zetina Cruz*



## **Introducción**

El control de calidad en el desarrollo de sistemas de información es un aspecto fundamental para garantizar la eficiencia, seguridad y confiabilidad de los productos. Para lograrlo, es necesario aplicar diversos métodos y herramientas que permitan identificar, registrar y corregir defectos durante el proceso de desarrollo. Entre estas herramientas se encuentra el cuaderno de registro de defectos, el cual facilita la organización y seguimiento de fallas encontradas, permitiendo una gestión eficaz de los errores y una mejora continua en el producto final.

Asimismo, la contabilización de defectos y errores es una práctica esencial para monitorear la calidad del software y tomar decisiones informadas para su corrección. El uso de métricas, listas de comprobación y una adecuada gestión del tiempo contribuyen a optimizar los procesos y minimizar el impacto económico de los defectos. En este contexto, la aplicación de estándares y metodologías especializadas asegura que los sistemas de información cumplan con altos niveles de calidad y satisfagan las necesidades de los usuarios.

### **2.2.1. El cuaderno de registro de defectos.**

Es un documento usado en procesos de control de calidad; Registra, clasifica y da seguimiento a los defectos identificados durante el desarrollo de un sistema. Este, detalla la descripción del defecto, ubicación, persona que encontró, la fecha de detección, el estado en que se encuentra y las acciones correctivas.

Sus objetivos pueden clasificarse en tres ámbitos importantes:

- Registro detallado de defectos: Documentación clara y organizada de cada defecto encontrado, incluyendo breve descripción, ubicación y causas posibles.
- Facilitación del seguimiento: Traza los defectos desde su detección hasta su corrección asegurando tomar decisiones necesarias para corregirlos.
- Asignación de responsabilidades: Identifica a los responsables de la corrección de cada defecto.

Un cuaderno de registro de defectos, debe contener las siguientes características:

- Descripción del defecto
- Fecha del defecto
- Ubicación del defecto
- Gravedad del defecto
- Acciones correctivas

Ventajas de tener un cuaderno de registro de defectos:

Nos permite llevar una mejor organización y control, ya que facilita el acceso a la información y el seguimiento de cada caso. Tiene una mayor eficiencia en la solución de problemas, porque reduce el tiempo de respuesta ante fallas recurrentes. Mejora la calidad del producto o servicio, garantizando el cumplimiento de estándares de calidad. Favorece la toma de decisiones basada en datos, pues permite tomar decisiones estratégicas para minimizar errores. Ofrece una asignación clara de responsabilidades, porque mejora la comunicación entre los equipos de trabajo. Y por último, favorece el cumplimiento de normas y auditorías.

### **2.2.2. Contabilización de defectos y errores.**

Un error es una acción humana que produce algún resultado incorrecto. El error es una equivocación de parte de los desarrolladores o analista, que puede llevarnos a generar uno o más defectos.

Los defectos, son aquellos que se encuentran en algún componente del sistema; Es la imperfección de un componente causado por un error. El analista de pruebas es quien debe reportar el defecto, puesto que su tarea es encargarse de ejecutar los casos de prueba y encontrar los mismos.

La contabilización de errores tiene como objetivos:

- Monitorear y evaluar la calidad en el proceso o producto mediante la identificación de los defectos.
- Establecer tendencias y patrones en la aparición de errores para tomar las decisiones de manera informada.
- Facilitar la mejora continua a través del análisis de causas y la implementación de medidas correctivas.
- Cumplir con normativas y estándares de calidad en la industria o disciplina.
- Optimizar la eficiencia operativa al reducir la repetición de errores y costos asociados.

Métodos de contabilización:

1. Conteo de defectos: Práctica usada en calidad y gestión de proyectos para identificar y registrar errores, fallas o problemas encontrados en un sistema o producto.
2. Conteo de errores: Implica identificar y registrar todos los errores que se encuentren durante el desarrollo del sistema. El objetivo principal es detectar estos errores y corregirlos, así como documentarlos para prevenir errores futuros.

Costo de calidad: El objetivo del cálculo de este, es comprender cómo repercute la calidad en el balance final. El costo de calidad ofrece a los fabricantes la oportunidad de analizar y mejorar sus operaciones de calidad.

Tasa de defectos: Es una métrica que se utiliza para evaluar la calidad. Se calcula dividiendo el número de defectos por la cantidad de unidades examinadas,

multiplicando este por cien para obtener un porcentaje. Nos ayuda a comprender cuántos errores o problemas hay en comparación con el número total de unidades producidas o examinadas.

Ventajas de la contabilización:

- Claridad financiera: Los empresarios pueden entender mejor su posición económica e identificar áreas de mejora y tomar decisiones informadas.
- Cumplimiento legal: Mantiene los registros contables precisos y al día facilita la presentación de informes ante las autoridades fiscales.
- Toma de decisiones fundamentadas: Desde la expansión hasta la contratación de personal. La contabilidad proporciona datos esenciales para evaluar la viabilidad y riesgos asociados en cada decisión.

### **2.2.3. Formas de encontrar y corregir defectos.**

Se trata de la calidad de los sistemas de información, particularmente enfocándose en el proceso de encontrar y corregir defectos en el desarrollo de software. La existencia de defectos puede ocasionar problemas de funcionalidad, seguridad y rendimiento, lo que hace de esta tarea una de las más críticas en el ciclo de vida del software.

### **Pasos para la Identificación de Errores**

Se presentan varios pasos fundamentales para la identificación y corrección de errores en programas:

1. **Identificación de los síntomas del defecto:** Reconocer los indicadores de que algo no está funcionando como debería es el primer paso hacia la solución.
2. **Localizar el defecto:** Una vez identificados los síntomas, se reduce el enfoque para localizar más específicamente el defecto en el código fuente.
3. **Entender el error:** Analizar qué parte del programa está causando el fallo permite desarrollar una comprensión más profunda del problema, facilitando

la corrección.

4. **Decidir cómo corregir el defecto:** Aquí se elige un enfoque para solucionar el problema detectado, teniendo en cuenta las posibles soluciones y sus implicaciones.
5. **Hacer la corrección:** Implementar la solución decidida en el código.
6. **Verificación de la solución:** Finalmente, es crucial verificar que el defecto ha sido efectivamente corregido y que el programa ahora funciona como se espera.

## Herramientas para Encontrar Defectos

El documento describe varias herramientas y métodos que pueden emplearse para identificar defectos:

- **Compiladores:** Utilizados para detectar errores sintácticos y algunos lógicos, son herramientas que pueden eliminar hasta un 90% de estos tipos de errores antes de que el código sea ejecutado.
- **Revisiones de código:** Este método consiste en que los desarrolladores revisen manualmente el código fuente para identificar errores. Se considera la forma más efectiva de detectar problemas antes de la fase de prueba, pues permite detectar y corregir problemas desde la base del código, antes de que sean ejecutados.
- **Pruebas:** Involucran la ejecución del software con el objetivo de identificar errores. Existen diferentes tipos de pruebas, como pruebas unitarias, de integración y del sistema, cada una con su propia efectividad en la detección de defectos.

## 2.2.4. El costo de encontrar y corregir defectos.

### Costos de Corrección

El documento también aborda el costo asociado con la corrección de defectos en diferentes fases del desarrollo. A medida que avanza el proyecto, la corrección de errores se vuelve más costosa:

- **Revisión de código:** 1-2 minutos por defecto.
- **Pruebas unitarias:** 10-20 minutos por defecto (10 veces el costo de la revisión de código).
- **Pruebas de integración:** 1 hora o más (100 veces el costo de la revisión de código).
- **Pruebas de sistema:** 10-40 horas por defecto (1000 veces el costo de la revisión de código).
- **Corrección tras entrega:** Puede costar 88 horas o más por defecto, lo que es extremadamente costoso y puede impactar en la reputación de la empresa.

### Impacto Económico

Además de los costos directos en horas de trabajo, también se abordan las repercusiones económicas que afectan a la empresa, como los retrasos en las entregas y la pérdida de confianza del cliente. Se menciona un caso real donde una empresa pequeña quedó significativamente retrasada en la entrega de un software debido a defectos difíciles de integrar, causando costos adicionales y una pérdida de credibilidad ante el cliente.

Para terminar, la detección y corrección de defectos son procesos críticos y que es esencial implementar una combinación de estrategias, como la revisión de código, pruebas automatizadas y la retroalimentación del usuario, para mejorar la calidad del software. Estas prácticas no solo ayudan a reducir costos, sino que también mejoran la confiabilidad del producto final, destacando la importancia de una buena gestión en el desarrollo de software.

### 2.3. LISTAS DE COMPROBACIÓN.

Las listas de comprobación son herramientas esenciales en diversos sectores, son utilizadas para asegurar que todas las tareas o pasos necesarios en un proceso sean realizados correctamente. Permite mejorar la eficiencia y sin omisiones, proporcionando un método organizado para el cumplimiento de actividades. Son aplicables en distintos ámbitos, desde la seguridad aérea, la ingeniería de software, las finanzas y la industria, demuestra su versatilidad y eficacia en garantizar la calidad y seguridad basado en diferentes contextos.

Una lista de comprobación es una herramienta donde se enumeran tareas o acciones específicas que deben realizarse en un proceso. Cada tarea marca como completada, lo que permite asegurar que no omitan pasos y que el proceso se ejecute de manera eficiente.

**Seguridad Aérea:** Existen listas previas al despegue para garantizar que todos los sistemas del avión funcionen, antes de iniciar el vuelo, son obligatorios para la seguridad de los pasajeros y tripulación y así evitar accidentes por fallos técnicos.

**Ingeniería de Software:** Son esenciales para garantizar la calidad del código, ayudan a prevenir los errores y aseguran el cumplimiento de estándares.

**Finanzas:** Se utilizan listas de comprobación para tomar decisiones informadas en inversiones, así permiten reducir riesgos y aumentar la precisión de las evaluaciones económicas.

El formato de una lista de comprobación se basa en la descripción de cada tarea es un párrafo explicando la acción a realizar, se encuentra un cuadrado vacío en la parte izquierda de cada tarea y se marca la verificación una vez completada, las marcas de verificación se utilizan para indicar las tareas realizadas, puede variar dependiendo del modelo sector, en aviación, las listas son divididas por sistema y acción separadas por puntos, se suelen leer alto y no se utilizan marcas de verificación.

Las reglas para la creación de listas de comprobación se dividen en 4: la primera es basarse en reglas específicas del producto y cada pregunta debe referirse a una

norma, la segunda se actualizan las listas cuando se descubran defectos no contemplados inicialmente, la tercera las preguntas deben ser claras limitándose a una sola página, la cuarta las preguntas deben estar orientadas a identificar defectos importantes y no deben generar nuevas reglas.

## **2.4. GESTIÓN DEL TIEMPO PARA EL DESARROLLO DE SISTEMAS DE INFORMACIÓN.**

La gestión del tiempo es crucial para el desarrollo de Sistemas de Información (SI), permite cumplir con los plazos establecidos y cumplimientos de las actividades. Existen tres restricciones clave: Tiempo, Costo y Alcance, las cuales deben respetarse para evitar multas o consecuencias negativas para la empresa y el equipo encargado.

La gestión de tiempo es el uso de técnicas, metodologías y herramientas para controlar los tiempos de entrega en proyectos, se debe cumplir en tiempo y forma, se necesita definir el proyecto, planificar tareas y recursos, gestionar costos y seguimiento, coordinar y optimizar y comunicación en equipo.

Tenemos cuatro metodologías la primera es Timeboxing dividen las tareas complejas en pequeñas partes para facilitar su ejecución, la segunda Getting Things Done anota y organiza todas tus tareas pendientes, la tercera no lo dejes para mañana se hace primero lo más difícil y la cuarta principio de Pareto se prioriza tareas sencillas con mayor impacto. Las cuatro técnicas son establecer prioridades listar tareas diarias y marcar las más importante, planificar las tareas se usan herramientas para la gestión del tiempo, evitar distracciones se bloquean las redes sociales y otras interrupciones, aprender a decir no es rechazar tareas adicionales que afecten la productividad.

Las cuatro tecnologías son Scoro que gestiona tareas, tiempos y finanzas, ProofHub organiza tareas y mejora la comunicación, Bitrix24 coordina proyectos y equipos eficientemente, Replicon supervisa horas laborales y normativas y Teamwork asigna las tareas y optimiza la colaboración. La gestión del tiempo es clave en el desarrollo de sistemas, puede afectar la calidad, costo y entrega, algunas herramientas útiles

son calendarios, reloj registrador online, apps de notas, seguimiento de hábitos, bloqueadores de distracciones, alertas y recordatorios.

Los casos de éxito y buenas prácticas en la gestión del tiempo se deben seguir estas estrategias: Definición clara de objetivos es establecer metas y plazos desde el inicio, planificación detallada crean un cronograma realista con tareas bien definidas, priorización de tareas abordan primero las más críticas y urgentes, monitoreo continuo evalúan los avances y ajustar estrategias si es necesario, gestión de riesgo tiene planes de contingencia para evitar retrasos, comunicación efectiva mantiene contacto constante entre el equipo y stakeholders.

Un equipo entregó un sistema de gestión de inventarios a tiempo usando tecnologías ágiles. Su éxito se debió a un conocimiento del sistema, mejor estimación del tiempo y priorización eficiente. Aplicaron planificación detallada, uso de metodologías ágiles y revisión continua de la calidad

## **2.5. Obtener calidad en los sistemas de información (métodos, métricas, metodologías, estándares)**

La calidad en los sistemas de información es un factor clave para garantizar la precisión, eficiencia y seguridad de los datos y procesos. Su correcta implementación permite que los sistemas sean confiables y alineados con las necesidades del negocio. Para lograrlo, es fundamental aplicar métodos rigurosos de prueba y revisión, utilizar métricas precisas para evaluar el desempeño, adoptar metodologías efectivas para la mejora continua y cumplir con estándares reconocidos internacionalmente.

Uno de los pilares de la calidad en los sistemas de información es la aplicación de **métodos** que permitan detectar y corregir errores en el software. Entre estos métodos se encuentran las pruebas de software, que incluyen pruebas unitarias para verificar el correcto funcionamiento de cada componente individual, pruebas de integración para garantizar la interacción adecuada entre los distintos módulos, pruebas de sistema para evaluar el software en su totalidad y pruebas de aceptación que validan si el producto cumple con las expectativas del usuario final. Además, las inspecciones de código ayudan a identificar errores y mejorar la calidad del software, mientras que las auditorías de calidad aseguran que los procesos cumplan con los estándares establecidos.

Para medir y evaluar la calidad del software y su desempeño, se emplean **métricas** especializadas. Entre las métricas de calidad del software, se encuentran la densidad de defectos, que mide el número de fallos por unidad de tamaño del software; el tiempo medio entre fallos (MTTF), que indica la confiabilidad del sistema; el tiempo medio de reparación (MTTR), que mide la rapidez con la que se

soluciona un fallo; y la satisfacción del cliente, que evalúa la experiencia del usuario final. Por otro lado, las métricas de desempeño incluyen el tiempo de respuesta del sistema ante una solicitud y el rendimiento (throughput), que mide la cantidad de transacciones procesadas en un período determinado.

Para garantizar una mejora continua en la calidad del software, se aplican diversas **metodologías**. Six Sigma se enfoca en reducir defectos y variaciones en los procesos mediante herramientas estadísticas y estrategias de optimización. Lean Six Sigma combina estos principios con Lean Manufacturing, eliminando desperdicios y mejorando la eficiencia. Total Quality Management (TQM) promueve la participación de todos los miembros de una organización en la mejora de la calidad, fomentando una cultura de excelencia. Por otro lado, Agile es una metodología ampliamente utilizada en el desarrollo de software, basada en la entrega incremental, la colaboración constante con los clientes y la capacidad de adaptación a los cambios.

El cumplimiento de **estándares** internacionales es otro aspecto crucial para la calidad en los sistemas de información. ISO/IEC 9126 establece un modelo de evaluación del software basado en características como funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad. Su evolución, ISO/IEC 25010, amplía este modelo para evaluar tanto la calidad del producto como su uso. CMMI (Capability Maturity Model Integration) es un modelo de mejora de procesos que ayuda a desarrollar productos y servicios de alta calidad. Además, existen marcos de gestión y gobierno de TI como ITIL (Information Technology Infrastructure Library), que optimiza la alineación de los servicios de TI con las necesidades del negocio, y COBIT (Control Objectives for Information and Related Technologies), que proporciona herramientas para desarrollar, implementar, monitorear y mejorar las prácticas de TI en una organización

## **2.6. Controlar la calidad del sistema de información.**

Debemos entender la calidad como la búsqueda de la excelencia en todo lo que el hombre, la sociedad y las organizaciones realizan.

Este concepto, aplica al desarrollo de sistemas de información, debido a que las investigaciones de hoy en día en el área de ingeniería de software, se centran en el desarrollo de metodologías que garanticen y controlen la calidad en el software construido. El principal problema, es la concepción de calidad que tenemos todos y cada uno de nosotros. El control de calidad se fundamenta en el principio de que la calidad se construye a través del proceso de mejora continua.

El Software Quality Assurance, se fundamenta en:

1. Uso de métodos y herramientas de análisis, diseño, codificación y prueba.

2. Revisiones técnicas.
3. Estrategia de escalación.
4. Control en la documentación del software.
5. Procedimientos que aseguren un ajuste a los estándares de desarrollo.

Algunos de los factores que determinan la calidad en el software son:

- a) Factores medibles.
- b) Factores no medibles

Según el autor McCall, la calidad puede agruparse de acuerdo con tres aspectos fundamentales:

- La características operacionales
- La capacidad de soportar cambios
- La adaptabilidad a nuevos entornos.

## **2.7. Costo de calidad de los sistemas de información.**

El desarrollo de software es una actividad muy compleja ya que éste es un producto intangible que depende principalmente del esfuerzo intelectual y creatividad de personas que lo realizan. Los errores humanos están presentes en todas las etapas de un proyecto de este tipo y puede llegar a ser muy costosa su corrección.

Costos para asegurar la calidad o costos de conformidad:

- Costos de prevención: Son aquellos que se encargan de asegurar la calidad del software y prevenir defectos en todas las fases del desarrollo de software.
- Costos de evaluación: Aquellos que descubren la condición de la calidad del software.

Costos por falta de calidad o costos de no conformidad:

- Costos de fallas internas: El esfuerzo para detectar y corregir problemas previos a que el usuario los detecte.
- Costos de fallas externas: Son del esfuerzo para corregir problemas que son detectados por los usuarios.

### **2.7.1. Cálculo del costo de la calidad. .**

Los costos asociados a la calidad, son aquellos gastos asegurados a garantizar que estos sistemas funcionen correctamente y satisfagan las necesidades de nuestros usuarios.

Clasificación:

- Costos de Prevención: Son gastos destinados a evitar que ocurran problemas en el sistema de información, incluyen actividades como planificación, capacitación, prevención y corrección de procesos.
- Costos de evaluación: Destinados a aquellos asociados a la medición de la calidad. Incluye inspecciones, auditorias, pruebas y verificaciones.
- Costos de fallos internos: Gastos derivados de defectos detectados antes de que el sistema sea puesto en producción.
- Costos de fallos externos: Surgen cuando los problemas se detectan después de que el sistema sea entregado o en uso.

Pasos para el cálculo del costo de la calidad:

1. Identificación de actividades de calidad
2. Clasificación de Costos
3. Recopilación de datos
4. Calculo de costos totales por categoria
5. Análisis de resultados
6. Implementación de mejoras.}

## **Conclusión**

El mantenimiento de la calidad en los sistemas de información es un proceso continuo que requiere la implementación de estrategias adecuadas para la detección y corrección de defectos. Herramientas como el cuaderno de registro de defectos, la contabilización de errores y las listas de comprobación son clave para identificar y solucionar fallas de manera eficiente. Además, una buena gestión del tiempo y el cumplimiento de estándares internacionales permiten mejorar la calidad del software y reducir costos asociados a errores en fases avanzadas del desarrollo.

La inversión en metodologías de aseguramiento de calidad no solo optimiza los recursos y tiempos de desarrollo, sino que también fortalece la confianza del usuario en el producto final. Por lo tanto, la aplicación de buenas prácticas y estrategias de control de calidad es esencial para el éxito de los sistemas de información en un entorno cada vez más competitivo y exigente.

Entrega

Enviado para calificar

Calificado

La tarea fue enviada 19 mins 47 segundos después de la fecha límite

Los estudiantes pueden editar este envío

 [Reporte-Investigación U2.pdf](#)

26 de marzo de 2025, 00:19

[Comentarios \(0\)](#)

Calidad en los Sistemas de Información U2

Calificación

### Rúbrica

Portada • El documento cuenta con hoja de presentación que incluye: nombre de los integrantes del equipo, nombre del docente, nombre de la signatura, nombre de la actividad, periodo escolar y fecha de entrega.

No cumple	Cumple parcialmente	Cumple adecuadamente
<b>0 puntos</b>	<b>1 puntos</b>	<b>2 puntos</b>

Desarrollo de la Información. • La información presentada incluye, introducción, desarrollo y conclusión cumpliendo con el objetivo de la unidad. • Realiza de manera adecuada la interpretación de los resultados. • Se incluyen evidencias de la actividad realizada.

No cumple	Cumple parcialmente	Cumple adecuadamente
<b>0 puntos</b>	<b>10 puntos</b>	<b>20 puntos</b>

Redacción y ortografía. • Las palabras están escritas correctamente y los signos de puntuación se usan de manera adecuada, de acuerdo con las normas académicas vigentes.

No cumple	Cumple parcialmente	Cumple adecuadamente
<b>0 puntos</b>	<b>1 puntos</b>	<b>2 puntos</b>

Participación. • Participó activamente compartiendo su investigación en plenaria grupal. • La participación se encuentra sustentada con argumentos sólidos. • Se expresa claramente y presenta dominio de la información logrando captar la atención de sus compañeros.

No cumple	Cumple parcialmente	Cumple adecuadamente
<b>0 puntos</b>	<b>3 puntos</b>	<b>6 puntos</b>

Calificación:

Calificación actual en el libro

30.00



# INSTITUTO TECNOLÓGICO SUPERIOR DE SAN ANDRÉS TUXTLA

---

---

## INGENIERÍA INFORMÁTICA

“Aplicación de Métodos y Herramientas de Medición de  
Calidad a un Sistema de Información”

# REPORTE DE INVESTIGACIÓN

Calidad en los Sistemas de Información

Presenta

**EMMANUEL DE JESÚS TEOBAL DÍAZ**  
**MARÍA GRISEL CAIXBA HERRERA**  
**RAFAEL PAXTIAN CAMPECHANO**  
**ANNETTE MONTAN MARTÍNEZ**

Profesor

*Guadalupe Zetina Cruz*



San Andrés Tuxtla, Veracruz

25 de marzo del 2025

## Contenido

<b>INTRODUCCIÓN</b> .....	3
<b>MEDICIÓN DE LA CALIDAD</b> .....	4
<b>APLICACIÓN EN UN SISTEMA DE INFORMACIÓN</b> .....	4
<b>HERRAMIENTAS DE MEDICIÓN</b> .....	4
<b>TÉCNICAS DE EVALUACIÓN</b> .....	5
<b>EJEMPLO DE APLICACIÓN</b> .....	6
<b>CARACTERÍSTICAS</b> .....	6
<b>PROBLEMÁTICAS</b> .....	6
<b>APLICACIÓN DEL DIAGRAMA DE PARETO PARA LA MEDICIÓN DE LA CALIDAD</b> .....	7
<b>RECOLECCIÓN DE DATOS</b> .....	7
<b>ORDENAMIENTO</b> .....	8
<b>GRAFICACIÓN</b> .....	9
<b>RESULTADOS</b> .....	9
<b>ENTORNO DE ANÁLISIS</b> .....	10
<b>CONCLUSIÓN</b> .....	12

## INTRODUCCIÓN

El análisis de Pareto es una herramienta clave en la medición de la calidad, ya que permite identificar y priorizar los problemas que tienen mayor impacto en un sistema. En este caso, se ha aplicado al sistema de información Grafiati, con el objetivo de mejorar la experiencia de los usuarios al gestionar citas y referencias bibliográficas.

A partir de una encuesta realizada a los usuarios, se identificaron cinco principales problemáticas, las cuales fueron ordenadas según la cantidad de quejas recibidas. Utilizando la regla 80/20, se realizó un Diagrama de Pareto para determinar cuáles de estos problemas generan el mayor porcentaje de insatisfacción y, por ende, requieren una atención prioritaria.

## MEDICIÓN DE LA CALIDAD

La medición de la calidad es el proceso de evaluar y cuantificar el nivel de excelencia de un producto, servicio o sistema con base en criterios predefinidos. En el contexto de los sistemas de información, se emplean métricas y herramientas especializadas para analizar aspectos como el rendimiento, la

## APLICACIÓN EN UN SISTEMA DE INFORMACIÓN

La medición de la calidad en un sistema de información se aplica a través de diversos métodos y herramientas que permiten evaluar su desempeño y adecuación a los requisitos establecidos. Algunos enfoques clave incluyen:

## HERRAMIENTAS DE MEDICIÓN

**Diagrama de Flujo:** Ayuda a descomponer los pasos de un proceso en una secuencia visual.

**Diagrama Causa-Efecto (Ishikawa o Espina de Pescado):** Identifica y muestra las posibles causas de un problema.

**Diagrama de Pareto:** Prioriza problemas o causas mediante una representación gráfica basada en la regla 80/20.

**COQ (Coste de la Calidad):** Se utiliza para identificar problemas potenciales y cuantificar los costos asociados a la falta de calidad en los procesos. Incluye:

- Diagramas de flujo detallados del proceso.
- Identificación de fases con costos de calidad (inspección, reparación, control de daños).
- Estimación de costos por fallos y por acciones correctoras o preventivas.
- Evaluación de la viabilidad de acciones correctoras.

**Benchmarking:** Método estructurado para comparar las mejores prácticas de diferentes organizaciones. Incluye:

- **Planificación:** Definir objetivos, estudiar procesos internos y seleccionar referentes.
- **Recopilación de datos:** Recolección de información cualitativa y cuantitativa mediante cuestionarios, entrevistas o visitas.
- **Análisis:** Comparación de datos y detección de brechas de rendimiento.
- **Adaptación:** Desarrollo de objetivos, planes de acción e implementación de mejoras.

**Encuestas:** Determinan la naturaleza de los procesos mediante:

- **Interrogación directa:** Entrevista verbal con registro de respuestas
- **Interrogación indirecta:** Uso de cuestionarios escritos.

## TÉCNICAS DE EVALUACIÓN

- **Encuesta de seguimiento:** Con este método se evalúa la calidad de los servicios a través de una encuesta por correo electrónico, por ejemplo, a través de Google Forms.
- **Encuestas y entrevistas:** Se pueden usar para recopilar información sobre conocimientos, creencias, intereses
- **Auditorías:** Permiten revisar si los procesos cumplen con los requisitos de la ISO 9001 y con las políticas internas de la organización.
- **Compras misteriosas:** Consiste en contratar a un "cliente encubierto" para que pruebe la calidad de un servicio.

## **EJEMPLO DE APLICACIÓN**

Grafiati es un sistema de información basado en web, cuya principal funcionalidad es ser un generador y gestor de citas y referencias bibliográficas el cual se adapta a diferentes tipos de estilos de citación.

## **CARACTERÍSTICAS**

- Permite crear citas y referencias en estilos como APA, MLA, Chicago, Harvard, y más
- Generar bibliografías automáticamente
- Permite crear referencias en diferentes idiomas
- Permite editar, modificar, copiar, y exportar referencias
- Cuenta con catálogos de proveedores de metadatos
- Permite convertir referencias entre diferentes estilos de cita

## **PROBLEMÁTICAS**

- De acuerdo con una encuesta realizada a los usuarios que utilizan este sistema de información, nos hemos percatado de que existen las siguientes problemáticas en el:
- El estilo de citación es incorrecto, 5 personas han presentado quejas.
- Publicidad invasiva: Los anuncios ocupan una buena parte del espacio visual y pueden distraer o hacer clic por error, interrumpiendo el proceso de agregar referencias, 30 personas presentaron quejas sobre este problema.
- Falta de validación de campos: No se observa una indicación clara de cuáles campos son obligatorios o si hay validación automática para formatos como la fecha o la URL, por lo menos 20 personas presentaron quejas.
- Función de autocompletado dependiente de URL: Si la URL no completa correctamente los campos (algo común con sitios mal estructurados), se tendría que llenar todo manualmente sin orientación clara, al menos 7 personas han presentado quejas.

- Confusión con la etiqueta "Nuevo" al elegir que estilo de referencia: Aunque resalta, no queda claro si "Nuevo" indica una actualización importante o solo que es una adición reciente, 10 personas han presentado quejas.

## APLICACIÓN DEL DIAGRAMA DE PARETO PARA LA MEDICIÓN DE LA CALIDAD

El Diagrama de Pareto se basa en la regla 80/20, que sugiere que el 80% de los problemas suelen ser causados por el 20% de las causas. En este caso, aplicarlo ayudaría a priorizar las problemáticas del sistema de información Grafiati, enfocándose en aquellas que tienen mayor impacto en la experiencia del usuario.

## RECOLECCIÓN DE DATOS

Se han identificado cinco problemáticas con la cantidad de quejas correspondientes:

- **Publicidad invasiva** – 30 quejas
- **Falta de validación de campos** – 20 quejas
- **Confusión con la etiqueta "Nuevo"** – 10 quejas
- **Función de autocompletado dependiente de URL** – 7 quejas
- **Estilo de citación incorrecto** – 5 quejas

	Categoría	Valor
1	Estilo de citación incorrecto	5
2	Publicidad Invasiva	30
3	Falta de validación de campos	20
4	Función de autocompletado dependiente de URL	7
5	Confusión de etiquetas	10
6		
7		
8		
9		
10		

**Figura 1. Software de análisis**

## ORDENAMIENTO

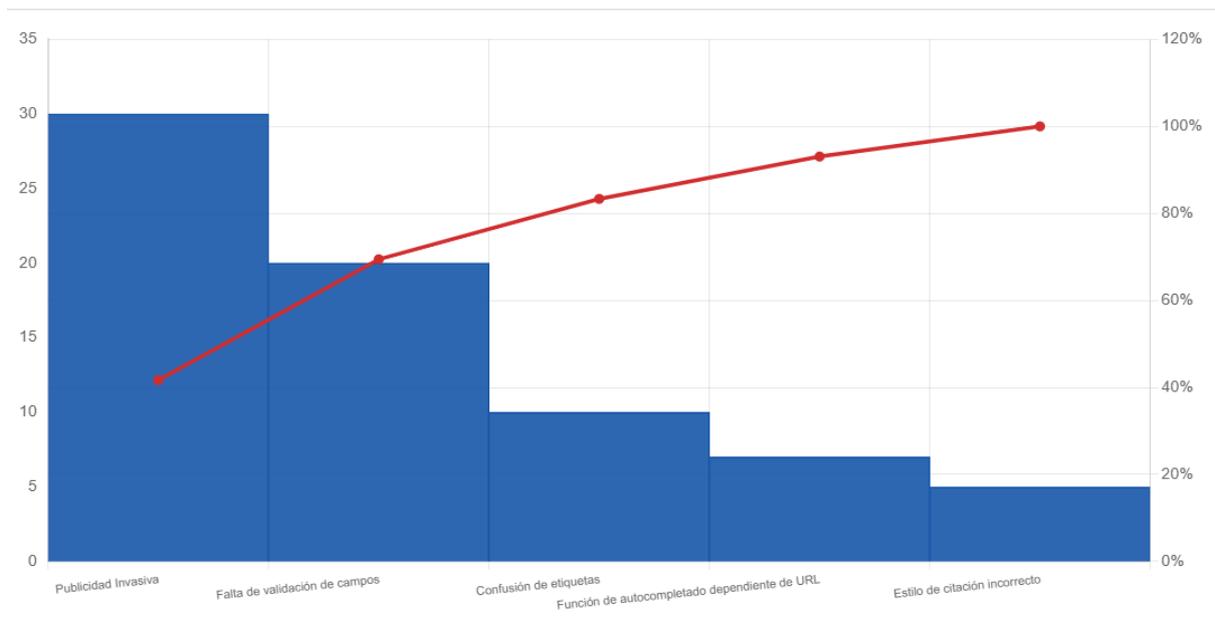
Tabla 1. Ordenamiento y Análisis

Problema	Quejas	% Individual	% Acumulado
Publicidad invasiva	30	41.667%	41.67%
Falta de validación de campos	20	27.778%	69.44%
Confusión con etiqueta "Nuevo"	10	13.889%	83.33%
Función de autocompletado	7	9.722%	93.06%
Estilo de citación incorrecto	5	6.944%	100%

CATEGORÍA	FRECUENCIA ABSOLUTA	FRECUENCIA RELATIVA	FRECUENCIA ABS. ACUMULADA	FRECUENCIA REL. ACUMULADA
Publicidad Invasiva	30	41.667%	30	41.67%
Falta de validación de campos	20	27.778%	50	69.44%
Confusión de etiquetas	10	13.889%	60	83.33%
Función de autocompletado dependiente de URL	7	9.722%	67	93.06%
Estilo de citación incorrecto	5	6.944%	72	100%

Figura 2. Ordenamiento y Análisis en Software

## GRAFICACIÓN



**Figura 3. Gráfica 80/20 del Diagrama Pareto**

## RESULTADOS

Los dos principales problemas (Publicidad invasiva y Falta de validación de campos) representan 62.5% de las quejas.

Si se resuelven estos dos problemas, se eliminaría una gran parte de la insatisfacción de los usuarios.

Se recomienda dar prioridad a la eliminación de la publicidad invasiva y mejorar la validación de campos, ya que estas dos problemáticas generan la mayor cantidad de quejas y afectan más a los usuarios.

# ENTORNO DE ANÁLISIS

Tabla 2. Entorno de Análisis

Componente	Ejemplos / Descripción
Sistema operativo	Windows 10/11
Editor de código	No aplica
Lenguajes de programación	No aplica
Frameworks o librerías	Excel, Herramienta WEB
Base de datos	No aplica
Servidor local	No aplica
Control de versiones	Docs
Herramientas colaborativas	Meet, Docs, Grafiati, Whatsapp

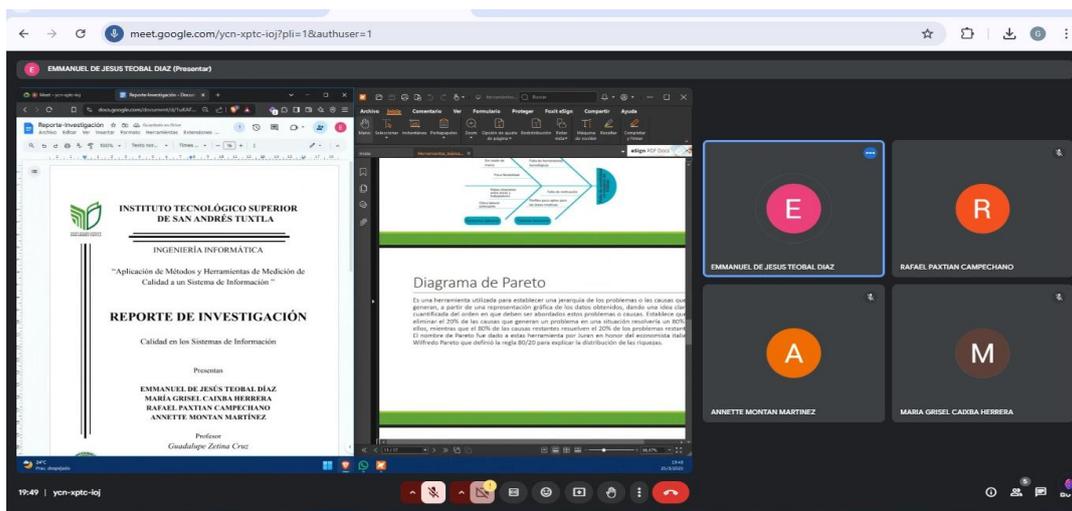
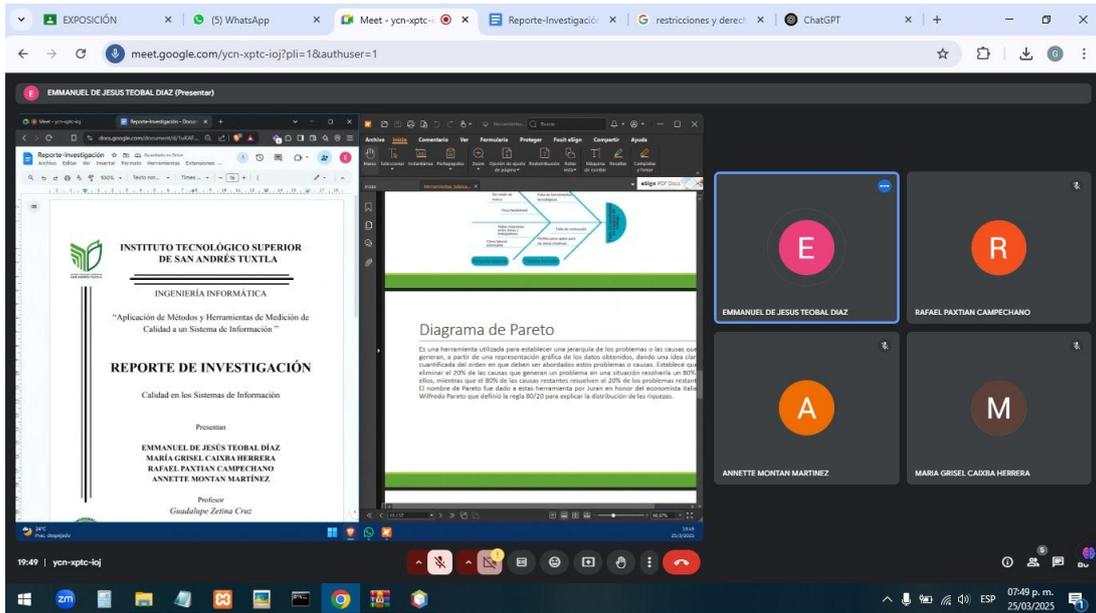
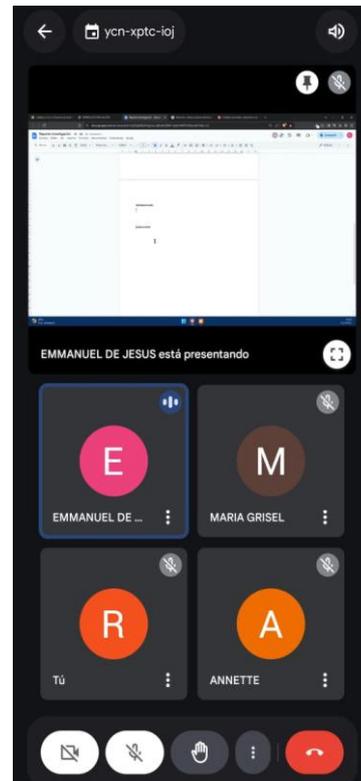


Figura 4. Herramienta de colaboración Google Meet



**Figura 5. Herramienta de colaboración Google Meet, aplicación de la herramienta**



**Figura 5. Herramienta de colaboración Google Meet Documentación**

## **CONCLUSIÓN**

El análisis de Pareto permitió identificar que el 41.67% de las quejas provienen de la publicidad invasiva, seguida por la falta de validación de campos con un 27.78%, acumulando entre ambas el 69.44% de los problemas reportados. Esto demuestra que, si se resuelven estos dos aspectos, se logrará una mejora significativa en la calidad del sistema y en la satisfacción de los usuarios.

Por lo tanto, se recomienda que las estrategias de mejora se enfoquen inicialmente en la reducción o reestructuración de la publicidad dentro del sistema, así como en la optimización de la validación de campos. Una vez abordados estos problemas prioritarios, se pueden considerar las demás mejoras para seguir optimizando la experiencia de usuario en Grafiati.

## **FUENTES**

Elaboración propia