

Informe FINAL del SISTEMA

BISOFT-32 PROCESO DE INGENIERIA DEL SOFTWARE I



19 de marzo de 2025

Sede San Pedro

**1. Informe de Pruebas de Control del Sistema de Información**

**1.1. Datos Generales del Informe**

* **Nombre del Sistema**: [Nombre del sistema que se prueba]
* **Versión del Sistema**: [Versión actual del sistema en pruebas]
* **Fecha del Informe**: [Fecha en que se realiza el informe]
* **Responsable de la Prueba**: [Nombre completo del responsable del informe, cargo y contacto]
* **Equipo de Pruebas**: [Lista de los miembros del equipo, incluyendo su rol y responsabilidades]
* **Entorno de Pruebas**: [Indicar si las pruebas se hicieron en entorno de desarrollo, pruebas, producción, etc.]

**1.2. Introducción**

**1.2.1 Descripción del Sistema**

Aquí debes proporcionar una breve descripción del sistema de información que se está evaluando. Esto incluye su propósito, la tecnología utilizada y cómo se integra con otras aplicaciones o sistemas.

* **Objetivo del Sistema**: [Objetivo general del sistema y su función principal].
* **Tecnología**: [Tecnologías, plataformas y herramientas utilizadas en el desarrollo del sistema].
* **Usuarios Objetivo**: [Grupo(s) de usuarios a los que está dirigido el sistema].

**1.2.2 Objetivo de las Pruebas de Control**

Se debe especificar el propósito de realizar las pruebas de control. Aquí se explican las razones por las que es necesario validar los controles en el sistema y qué aspectos se espera cubrir con estas pruebas.

* **Propósito**: Verificar la integridad, seguridad, accesibilidad, y confiabilidad del sistema.
* **Importancia**: Asegurar que el sistema cumpla con las normativas, estándares internos o externos y que no tenga vulnerabilidades que pongan en riesgo la confidencialidad, integridad y disponibilidad de la información.

**1.2.3 Alcance de las Pruebas**

El alcance de las pruebas define qué áreas del sistema fueron probadas y cuáles quedaron fuera de este informe.

* **Componentes Evaluados**: [Base de datos, interfaces, componentes de red, seguridad, accesos, etc.].
* **Áreas Excluidas**: [Si hay áreas que no se evaluaron por cualquier motivo, indícalo aquí].

**1.3. Metodología de Pruebas**

**1.3.1 Enfoque General de las Pruebas**

Aquí debes detallar el enfoque y tipo de pruebas que se utilizaron. Esto puede incluir el tipo de pruebas funcionales, de seguridad, de rendimiento, etc.

* **Pruebas Funcionales**: Evaluación de la correcta ejecución de las funcionalidades del sistema.
* **Pruebas de Seguridad**: Evaluación de la implementación de controles de seguridad, como autenticación, autorización, cifrado de datos, etc.
* **Pruebas de Integridad de Datos**: Verificación de que los datos se mantienen consistentes y correctos en todo momento.
* **Pruebas de Rendimiento**: Evaluación de la capacidad del sistema para soportar cargas esperadas o extremas.
* **Pruebas de Usabilidad**: Evaluación de la facilidad de uso y accesibilidad del sistema para los usuarios.

**1.3.2 Técnicas de Pruebas**

Describe las técnicas y herramientas utilizadas para realizar las pruebas.

* **Técnicas Manuales**: [Si se usaron pruebas manuales, como pruebas de accesos, revisión de logs, pruebas de interfaz].
* **Técnicas Automáticas**: [Herramientas automáticas utilizadas, como Selenium, JUnit, o cualquier framework de pruebas].
* **Pruebas de Caja Negra/Caja Blanca**: [¿Se realizaron pruebas de caja negra, donde se evalúa el sistema sin conocer su código, o caja blanca, donde se analiza el código interno.

**1.3.3 Criterios de Aceptación**

Define qué condiciones deben cumplirse para que las pruebas sean consideradas exitosas.

* **Criterios para Aprobación de la Prueba**: [Describir las condiciones que deben cumplirse, como la ausencia de vulnerabilidades graves, el cumplimiento de los requisitos de seguridad, la estabilidad del sistema, etc.].

**1.4. Descripción de los Controles Probados**

En esta sección se describen detalladamente los controles específicos que fueron evaluados. Esto incluye controles de acceso, seguridad, integridad de datos, etc.

**1.4.1 Control de Acceso**

* **Descripción**: Control de roles y permisos, autenticación y autorización de usuarios.
* **Pruebas Realizadas**: [Pruebas sobre diferentes roles de usuarios, verificando que los accesos estén restringidos adecuadamente].
* **Resultado**: [Aprobado/No Aprobado]
* **Observaciones**: [Detalles sobre posibles problemas con accesos indebidos o vulnerabilidades encontradas].

**1.4.2 Control de Integridad de Datos**

* **Descripción**: Verificación de que los datos se almacenan correctamente y no se alteran sin autorización.
* **Pruebas Realizadas**: [Validación de que las transacciones de base de datos son correctas, pruebas de auditoría, revisiones de registros, etc.].
* **Resultado**: [Aprobado/No Aprobado]
* **Observaciones**: [Detalles sobre cualquier alteración no autorizada o posible corrupción de datos].

**1.4.3 Control de Seguridad**

* **Descripción**: Evaluación de las protecciones de seguridad implementadas.
* **Pruebas Realizadas**: [Revisión de autenticación, encriptación, protección contra ataques como inyección SQL, CSRF, XSS, etc.].
* **Resultado**: [Aprobado/No Aprobado]
* **Observaciones**: [Detallar vulnerabilidades encontradas o áreas de mejora en la seguridad del sistema].

**1.4.4 Control de Backup y Recuperación**

* **Descripción**: Evaluación de los procedimientos de copia de seguridad y recuperación ante desastres.
* **Pruebas Realizadas**: [Verificación de que los datos se respalden de manera regular y que puedan ser recuperados con éxito].
* **Resultado**: [Aprobado/No Aprobado]
* **Observaciones**: [Deficiencias en la frecuencia de backups, tiempos de restauración, etc.].

**1.4.5 Control de Monitoreo**

* **Descripción**: Revisión de los sistemas de monitoreo de eventos y alertas.
* **Pruebas Realizadas**: [Revisión de logs, configuración de alertas ante eventos críticos].
* **Resultado**: [Aprobado/No Aprobado]
* **Observaciones**: [Problemas en la configuración de alertas o en la recolección de logs].

**1.5. Resultados de las Pruebas**

**1.5.1 Resumen de Resultados Generales**

Proporciona un resumen global de cómo han resultado las pruebas. Indica si se lograron los objetivos y si el sistema está en condiciones adecuadas.

* **Porcentaje de Controles Aprobados**: [Porcentaje de controles probados que fueron aprobados].
* **Deficiencias Críticas**: [Si se encontraron problemas graves que afectan el funcionamiento del sistema].
* **Áreas de Mejora**: [Si hubo áreas que necesitan ajustes menores o mejoras, indícalas aquí].

**1.5.2 Incidencias y Deficiencias Encontradas**

Detalles específicos de las incidencias y deficiencias encontradas durante las pruebas.

| **Incidencia** | **Descripción** | **Impacto** | **Estado** | **Recomendaciones** |
| --- | --- | --- | --- | --- |
| Incidencia 1 | [Descripción de la incidencia encontrada] | [Alto/Medio/Bajo] | [Resuelta/Pendiente] | [Acción recomendada para corregir] |
| Incidencia 2 | [Descripción de la incidencia encontrada] | [Alto/Medio/Bajo] | [Resuelta/Pendiente] | [Acción recomendada para corregir] |

**1.5.3 Recomendaciones Generales**

Proporciona recomendaciones sobre cómo mejorar los controles, procedimientos y configuraciones del sistema.

* **Recomendación 1**: [Mejorar seguridad mediante la implementación de un sistema de autenticación multifactor].
* **Recomendación 2**: [Optimizar los procesos de backup y establecer políticas claras de restauración ante desastres].

# ****2. Plan Detallado para Realizar un Presupuesto para un Sistema de Información****

## ****2.1. Definición de los Requisitos del Sistema****

El primer paso clave para construir un presupuesto sólido es comprender los **requisitos funcionales** y **no funcionales** del sistema de información. Esto es esencial para definir el alcance y los recursos que necesitarás.

### ****Actividades necesarias****:

* **Reuniones con stakeholders**: Reunirse con todas las partes interesadas (gerentes, usuarios finales, personal técnico) para definir qué espera cada uno del sistema.
* **Análisis de requisitos funcionales**: Documentar todas las funciones que el sistema debe realizar. Ejemplo: ingreso de datos, reportes, gestión de usuarios, etc.
* **Análisis de requisitos no funcionales**: Considerar factores como rendimiento, escalabilidad, seguridad, integridad de los datos, usabilidad y requisitos legales.

### ****Presupuesto asociado****:

* **Reuniones y Consultas**: Costos relacionados con la participación de consultores o analistas para realizar las reuniones de definición de requisitos.
* **Documentación de Requisitos**: Los costos asociados con la creación de la documentación técnica (por ejemplo, la contratación de un analista de sistemas).

## ****2.2. Identificación de Componentes del Sistema****

### ****Componentes esenciales a identificar y presupuestar****:

#### 2.2.1 **Desarrollo del Software**

El desarrollo del software incluye el diseño, la programación, las pruebas y la implementación del sistema. Esto requiere un equipo adecuado de desarrollo y herramientas especializadas.

**Actividades y costos**:

* **Diseño del sistema**: Requiere de arquitectos de software o diseñadores para crear la estructura del sistema, bases de datos, interfaces de usuario, etc.
  + Costo estimado: $3,000 - $8,000 (dependiendo de la complejidad).
* **Desarrollo**: Los desarrolladores trabajarán en el código del sistema.
  + **Programadores**: Dependerá del lenguaje, la complejidad y las horas que se necesiten para programar.
    - Ejemplo: Si un programador cobra $40/hora y se requieren 1,200 horas (3 meses de trabajo completo para 1 programador):
      * Costo estimado: $48,000 (para 1 desarrollador).
    - Si el equipo tiene 4 desarrolladores, el costo se multiplicará.
* **Pruebas de desarrollo**: Las pruebas de integración y pruebas de validación que aseguran la calidad del sistema.
  + Costo estimado: $5,000 - $15,000.
* **Revisión del código y documentación**: Asegurarse de que el código sea de alta calidad y esté bien documentado.
  + Costo estimado: $3,000 - $5,000.

#### 2.2.2 **Licencias de Software y Herramientas**

**Tipos de licencias necesarias**:

* **Lenguajes de programación y frameworks**: Si se utilizan herramientas comerciales como bases de datos o plataformas específicas de desarrollo.
  + Ejemplo: Licencia de Microsoft SQL Server o herramientas de desarrollo como JetBrains, etc.
  + Costo estimado: $500 - $2,000 anuales por licencia.
* **Herramientas de desarrollo y testing**: Si utilizas herramientas como JIRA, Selenium, Jenkins, etc., hay que incluir las licencias y costos de estas herramientas.
  + Costo estimado: $1,000 - $10,000 dependiendo del tamaño del equipo.

#### 2.2.3 **Infraestructura Tecnológica**

La infraestructura es uno de los componentes más grandes en términos de costos. Necesitarás servidores, redes, almacenamiento, seguridad, etc.

**Costos involucrados**:

* **Servidores** (físicos o en la nube):
  + Ejemplo: Servidor dedicado o servicios en la nube (Amazon AWS, Microsoft Azure).
    - Costo estimado: Entre $500 y $5,000 mensuales dependiendo de la configuración (por ejemplo, 12 meses de alquiler de 1 servidor en la nube con características medias podría costar $2,000 al año).
* **Bases de datos**: Si el sistema requiere bases de datos especializadas como Oracle o bases de datos en la nube.
  + Costo estimado: $2,000 - $10,000 anuales por licencia de base de datos.
* **Redes y Comunicaciones**:
  + Costos de configuraciones de red, equipos como routers, switches y equipos de firewall.
  + Costo estimado: $2,000 - $10,000 dependiendo de la configuración.

#### 2.2.4 **Seguridad del Sistema**

Las medidas de seguridad son cruciales para proteger la información sensible. Los costos incluyen herramientas de protección contra ataques y auditorías externas.

**Actividades y costos**:

* **Software de seguridad**: Firewall, antivirus, herramientas de protección contra ataques (como DDoS, inyecciones SQL, etc.).
  + Costo estimado: $1,000 - $5,000 anuales para herramientas de seguridad.
* **Auditorías de seguridad**: Exámenes de seguridad para identificar vulnerabilidades.
  + Costo estimado: $3,000 - $10,000 dependiendo de la complejidad y el tamaño del sistema.

## ****2.3. Estimación de Costos Detallados****

Aquí, elaboramos una tabla para hacer un presupuesto detallado. Este presupuesto debe considerar todas las categorías mencionadas anteriormente.

### ****Ejemplo de presupuesto detallado****:

| **Categoría** | **Descripción** | **Costo Estimado** | **Observaciones** |
| --- | --- | --- | --- |
| **Desarrollo de Software** | Incluye diseño, desarrollo, pruebas y revisión | $150,000 | Para equipo de 4 desarrolladores durante 6 meses |
| **Licencias de Software** | Licencias de herramientas, bases de datos y frameworks | $10,000 | Licencias de SQL, IDEs y herramientas de testing |
| **Infraestructura** | Servidores en la nube, bases de datos y redes | $30,000 | Servidores en AWS y licencias de base de datos |
| **Seguridad** | Herramientas de seguridad y auditorías externas | $12,000 | Firewall, antivirus y auditorías de seguridad |
| **Capacitación** | Entrenamiento de usuarios y técnicos | $8,000 | Incluye capacitación para 20 usuarios finales |
| **Mantenimiento Anual** | Soporte, actualizaciones y mantenimiento regular | $25,000 | 20% del costo total anual |
| **Contingencia (10%)** | Fondo de contingencia para imprevistos | $23,500 | Para cubrir cambios no previstos |
| **Total Estimado** |  | **$258,500** |  |

## ****2.4. Cronograma de Implementación y Desembolso****

El cronograma es importante para distribuir los pagos en el tiempo y entender cómo se ejecutarán los recursos durante el proyecto.

**Fase de Planificación (1 mes)**:

* Reuniones de definición de requisitos.
* Documentación de especificaciones.
* Estimación de costos.

**Fase de Desarrollo (4-6 meses)**:

* Diseño del sistema.
* Programación.
* Pruebas de integración.
* Revisión del código.

**Fase de Implementación (1 mes)**:

* Instalación y configuración de servidores.
* Despliegue del sistema en producción.
* Capacitación a usuarios finales.

**Fase de Mantenimiento (Continuo)**:

* Soporte técnico.
* Actualizaciones periódicas.

**Cronograma de pagos**:

* Fase de planificación: 5% del total.
* Fase de desarrollo: 50% del total.
* Fase de implementación: 20% del total.
* Fase de mantenimiento anual: 25% del total.

## ****2.5. Revisión y Aprobación del Presupuesto****

Una vez elaborado el presupuesto detallado, presenta la propuesta a las partes interesadas para obtener su retroalimentación y aprobación.

* **Revisión interna**: Analiza el presupuesto con los gerentes de tecnología, finanzas y otros departamentos relevantes.
* **Aprobación final**: Una vez que todas las partes interesadas estén de acuerdo, procederás con la ejecución del presupuesto aprobado.

# ****3. Plan Detallado para Medir los Riesgos de un Sistema de Información****

## ****3.1. Definición del Alcance y los Objetivos del Plan de Gestión de Riesgos****

### ****1.1 Definición del Alcance****

Antes de empezar, es crucial determinar el **alcance del sistema** de información y qué componentes se incluirán en la evaluación de riesgos. Esto es clave para enfocar el esfuerzo y evitar que se omitan áreas críticas.

#### **Actividades específicas**:

* **Identificar componentes del sistema**: Identificar todos los componentes involucrados (hardware, software, bases de datos, redes, usuarios, interfaces).
  + **Ejemplo**: El sistema de gestión de inventarios puede incluir módulos de compras, ventas, gestión de almacenes, interfaces con proveedores y sistemas ERP, servidores, bases de datos SQL, etc.
* **Establecer límites**: Determinar qué parte del sistema se incluirá en la gestión de riesgos y cuáles estarán fuera de este alcance.
  + **Ejemplo**: Si el sistema se integra con otros sistemas de la empresa, pero solo el módulo de ventas está en producción, solo ese módulo se evaluará en esta fase.

### ****3.1.2 Establecimiento de Objetivos****

El objetivo del plan es identificar, evaluar, mitigar y monitorear riesgos que puedan afectar al sistema. Los objetivos deben alinearse con las metas de negocio y asegurar que se protejan los activos más críticos.

#### **Actividades específicas**:

* **Establecer metas claras**:
  + **Ejemplo**: Proteger la confidencialidad de los datos del cliente, garantizar la disponibilidad del sistema de ventas 24/7, asegurar la integridad de los registros de inventarios.
* **Alineación con los objetivos de la organización**:
  + **Ejemplo**: Si el objetivo de la empresa es reducir el tiempo de inactividad, el plan de riesgos debe poner especial atención a los riesgos que podrían afectar la disponibilidad del sistema.

## ****3.2. Identificación de Riesgos****

### ****3.2.1 Fuentes de Riesgos****

En esta etapa, el objetivo es identificar todos los posibles riesgos que pueden afectar al sistema. Existen diversas fuentes de riesgos, que se deben evaluar de manera exhaustiva.

#### **Actividades específicas**:

* **Evaluación de procesos críticos**: Analizar los procesos de negocio asociados al sistema.
  + **Ejemplo**: Si el sistema de ventas está conectado a la plataforma de pagos, cualquier interrupción en el procesamiento de pagos podría ser un riesgo crítico para la operación.
* **Análisis de amenazas externas**:
  + **Ejemplo**: Riesgos como **ataques de ransomware**, **phishing** y **DDOS** que podrían afectar la seguridad del sistema.
* **Análisis de amenazas internas**:
  + **Ejemplo**: Los errores humanos, como una mala configuración de la base de datos o un acceso no autorizado a datos confidenciales por parte de un empleado.
* **Condiciones externas**:
  + **Ejemplo**: Condiciones del mercado que pueden afectar el software que estás usando (como una descontinuación o desactualización de la plataforma en la nube).

#### **Tipos comunes de riesgos**:

1. **Riesgos Tecnológicos**:
   * Fallas en la infraestructura (ej., servidor caído).
   * Vulnerabilidades en el software.
   * Incompatibilidad de versiones de software.
2. **Riesgos de Seguridad**:
   * **Acceso no autorizado**: Los usuarios malintencionados podrían obtener acceso a información sensible.
   * **Ataques cibernéticos**: Phishing, malware, inyecciones SQL.
   * **Pérdida de datos**: Por corrupción de archivos o errores humanos.
3. **Riesgos Operacionales**:
   * **Errores humanos**: Como cargar datos incorrectos en el sistema.
   * **Fallas en procesos de negocio**: Como una caída del sistema de pago en línea.
4. **Riesgos Regulatorios**:
   * **Cumplimiento de normativas**: Leyes de privacidad (como GDPR), normativas de seguridad de datos.
   * **Requisitos industriales**: Cumplimiento con estándares como ISO 27001 para la seguridad de la información.

### ****3.2.2 Herramientas para Identificar Riesgos****:

Utilizar diferentes **herramientas de gestión de riesgos** como:

* **Análisis FODA (SWOT)**: Para identificar fortalezas, debilidades, oportunidades y amenazas que puedan impactar en el sistema.
* **Brainstorming**: Sesiones con equipos clave para identificar posibles riesgos.
* **Entrevistas**: Hablar con usuarios y partes interesadas clave para entender los riesgos.

## ****3.3. Evaluación de Riesgos****

### ****3.3.1 Evaluación de Probabilidad e Impacto****

Después de identificar los riesgos, es crucial **evaluar la probabilidad** de que ocurran y **el impacto** que tendrían en el sistema. La evaluación debe basarse en hechos y datos históricos, pero también en la experiencia del equipo.

#### **Escala de Probabilidad e Impacto**:

1. **Probabilidad**:
   * **Alta**: Riesgo casi seguro que sucederá en el futuro cercano.
   * **Media**: Existe una probabilidad moderada, pero no es seguro que ocurra.
   * **Baja**: Riesgo poco probable que ocurra en el corto plazo.
2. **Impacto**:
   * **Crítico**: El riesgo tiene consecuencias severas, como pérdida de datos o interrupción del servicio.
   * **Alto**: Causaría un daño significativo, pero no necesariamente impediría que el sistema funcione.
   * **Moderado**: Tiene un impacto limitado o temporal.
   * **Bajo**: Afecta mínimamente el rendimiento del sistema.

#### **Matriz de Riesgo**:

| **Impacto / Probabilidad** | **Alta** | **Media** | **Baja** |
| --- | --- | --- | --- |
| **Crítico** | **1** (Muy alto) | **2** (Alto) | **3** (Moderado) |
| **Alto** | **2** (Alto) | **3** (Moderado) | **4** (Bajo) |
| **Moderado** | **3** (Moderado) | **4** (Bajo) | **5** (Muy bajo) |
| **Bajo** | **4** (Bajo) | **5** (Muy bajo) | **6** (Bajo) |
|  |  |  |  |

### ****3.3.2 Herramientas de Evaluación de Riesgos****:

* **Matriz de Riesgos**: Para categorizar y priorizar los riesgos.
* **Análisis de impacto en el negocio (BIA)**: Evaluar cómo un riesgo afectaría las operaciones comerciales clave.

## ****3.4. Planificación de Respuestas a los Riesgos****

### ****3.4.1 Estrategias de Respuesta****

Una vez que se han evaluado los riesgos, es necesario definir cómo **responder** a cada uno de ellos de manera eficaz.

#### **Respuestas típicas a los riesgos**:

1. **Evitar**: Cambiar el enfoque o las prácticas del proyecto para evitar el riesgo.
   * **Ejemplo**: Si se identifica un riesgo de que un servidor se caiga durante una actualización, se puede optar por realizar la actualización durante horarios de menor tráfico.
2. **Mitigar**: Reducir la probabilidad o el impacto del riesgo.
   * **Ejemplo**: Si hay un riesgo de pérdida de datos debido a fallos en el sistema, se puede implementar un sistema de copias de seguridad diarias.
3. **Transferir**: Pasar el riesgo a un tercero.
   * **Ejemplo**: Contratar un seguro para cubrir posibles pérdidas por daños a la infraestructura.
4. **Aceptar**: Aceptar que el riesgo ocurra y planificar cómo manejarlo.
   * **Ejemplo**: Aceptar el riesgo de caídas menores en un servidor, pero tener protocolos de recuperación rápida.

### ****3.4.2 Planes de Contingencia****

Desarrollar planes para mitigar los efectos si el riesgo se materializa. Estos planes deben incluir **procedimientos de emergencia**, **comunicaciones** y **acciones correctivas**.

## ****3.5. Monitoreo y Control de Riesgos****

### ****3.5.1 Herramientas de Monitoreo****:

* **Sistema de monitoreo de infraestructura**: Herramientas como **Nagios** o **Zabbix** para monitorear el estado del servidor y la red.
* **Auditorías de seguridad**: Para verificar vulnerabilidades y asegurar la integridad del sistema.

### ****3.5.2 Revisiones Periódicas****:

Programar **revisiones periódicas** del plan de riesgos, donde se analicen los riesgos emergentes y se ajusten las respuestas en función de la evolución del sistema.

## ****3.6. Documentación y Comunicación de Riesgos****

### ****3.6.1 Registro de Riesgos****:

* Crear un **registro de riesgos** que contenga toda la información sobre los riesgos identificados, su evaluación, las respuestas planeadas y el estado actual.

### ****3.6.2 Reportes Regulares****:

* Informes periódicos a **la alta dirección** sobre el estado de los riesgos y las acciones tomadas para mitigarlos.

## ****3.7. Revisión y Mejora Continua****

El plan de riesgos debe ser **dinámico** y adaptarse a los cambios en el sistema y en el entorno operativo. Es esencial una **mejora continua** basada en las lecciones aprendidas.

#### **Actividades**:

* **Revisión post-incidente**: Si un riesgo se materializa, realizar un análisis post-mortem para identificar lecciones y mejorar la gestión de riesgos futura.

### ****3.8 Conclusión****

**3.8.1 Resumen de las Pruebas Realizadas**

* **General**: [Resumen de qué pruebas fueron realizadas, destacando si se cumplieron los requisitos del sistema].
* **Condición del Sistema**: [Evaluación global de la efectividad de los controles, con recomendaciones para la continuidad o la mejora].

**3.8.2 Acciones a Tomar**

* **Acciones Inmediatas**: [Si se identificaron deficiencias graves, qué acciones deben tomarse de manera urgente].
* **Acciones a Largo Plazo**: [Sugerencias sobre la mejora continua del sistema y las pruebas].

El presupuesto de un sistema de información debe ser detallado, cubrir todas las áreas necesarias para garantizar el éxito del proyecto y permitir un monitoreo eficaz durante la implementación y el mantenimiento. Tener en cuenta tanto los costos de desarrollo iniciales como los gastos recurrentes de mantenimiento es crucial para un proyecto a largo plazo.

Este plan detallado te permitirá gestionar los riesgos de un sistema de información de manera estructurada y proactiva. Implementarlo de forma eficiente reducirá los impactos negativos, mejorará la seguridad y garantizará la continuidad del negocio.

**3.9. Anexos**

Incluye documentación adicional como logs de pruebas, capturas de pantalla, resultados detallados de herramientas automáticas, etc.

* **Anexo 1**: [Descripción del anexo, si es relevante].
* **Anexo 2**: [Descripción del anexo, si es relevante].