**RIESGO 13: No considerar escalabilidad**

* **Tipo de riesgo:** Técnico / Estratégico
* **Categoría:** Arquitectura de software / Planeación / Tecnología
* **Descripción:** El riesgo se presenta cuando el sistema o producto no ha sido diseñado para crecer en capacidad (usuarios, transacciones, módulos, integraciones) sin afectar significativamente su rendimiento, costo o estabilidad. Esto puede provocar cuellos de botella, reestructuras tardías, pérdida de competitividad y elevados gastos de reingeniería.

**1. IDENTIFICACIÓN DEL RIESGO**

|  |  |
| --- | --- |
| Elemento | Detalle |
| Nombre del riesgo | No considerar escalabilidad |
| Categoría | Arquitectura / Tecnología / Planeación estratégica |
| Causas probables | Enfoque exclusivo en MVP, presión por tiempo, falta de experiencia técnica, omisión de arquitectura modular, subestimación del crecimiento futuro. |
| Consecuencias | Reestructuración del sistema, degradación de rendimiento, sobrecostos, pérdida de usuarios, limitaciones de expansión. |
| Fuente | Fase de análisis técnico, diseño arquitectónico, backlog técnico, documentación del producto. |

**2. EVALUACIÓN DEL RIESGO**

|  |  |
| --- | --- |
| Criterio | Valoración |
| Probabilidad de ocurrencia | Media-Alta (especialmente en entornos de startups o productos iterativos) |
| Impacto potencial | Crítico (afecta directamente sostenibilidad técnica y económica del producto) |
| Nivel de riesgo | Muy alto |
| Indicadores de riesgo | Saturación de procesos, código difícil de escalar, incompatibilidad con nuevos módulos, lentitud bajo carga. |

**3. DEFINICIÓN DE MEDIDAS DE CONTROL**

**A. Medidas Preventivas (Antes del riesgo)**

|  |  |  |
| --- | --- | --- |
| Acción | Responsable | Herramientas / Recursos |
| Diseñar una arquitectura basada en microservicios, modular o escalable desde el inicio. | Arquitecto / CTO | Docker, Kubernetes, Clean Architecture |
| Considerar patrones de escalabilidad horizontal y vertical en infraestructura. | DevOps / Arquitecto | AWS, Azure, GCP, Autoscaling |
| Evaluar crecimiento proyectado y casos de uso futuros desde la fase de requisitos. | PO / Cliente / PM | Benchmark, roadmap |
| Aplicar principios de desarrollo desacoplado y APIs bien definidas. | Backend / Arquitecto | REST, GraphQL, documentación Swagger |
| Realizar revisiones arquitectónicas periódicas. | CTO / QA / Dev | Revisión técnica, auditorías internas |

**B. Medidas de Mitigación (Durante el riesgo)**

|  |  |  |
| --- | --- | --- |
| Acción | Responsable | Herramientas / Recursos |
| Reestructurar parcialmente componentes críticos que dificultan la escalabilidad. | Arquitecto / Dev | Refactorización modular |
| Desplegar infraestructura temporal de mayor capacidad. | DevOps | Escalado vertical, instancias on-demand |
| Aplicar soluciones temporales: balanceo, cachés, colas de mensajes. | Backend / DevOps | Redis, RabbitMQ, Load Balancer |
| Priorizar módulos para escalabilidad en próximos sprints. | PO / Dev | Sprint planning, técnica MoSCoW |

**C. Medidas Correctivas (Después del riesgo)**

|  |  |  |
| --- | --- | --- |
| Acción | Responsable | Herramientas / Recursos |
| Realizar migración progresiva hacia arquitectura escalable. | Arquitecto / CTO | Service mesh, refactor |
| Documentar el impacto técnico y económico de la falta de escalabilidad. | QA / PM | RCA, postmortem técnico |
| Incluir escalabilidad como criterio obligatorio en futuras decisiones técnicas. | PMO / QA | Políticas internas, checklist QA |
| Auditar la arquitectura actual para detectar cuellos de botella. | QA / DevOps | Monitoreo, pruebas de carga |

**1. Estimación de Pérdidas Económicas por Riesgo No Controlado**

|  |  |  |
| --- | --- | --- |
| Escenario de materialización | Impacto directo | Costo estimado (MXN) |
| Colapso del sistema ante aumento de usuarios | Pérdida de servicio, fallos críticos | $30,000 MXN |
| Reingeniería completa del sistema para hacerlo escalable | 3 semanas de trabajo de 3 desarrolladores | $54,000 MXN (3 × $6,000 × 3 semanas) |
| Pérdida de usuarios / oportunidades por rendimiento bajo | Abandono de clientes, baja en conversiones | $15,000 MXN |
| Costos de soporte, incidencias y correcciones urgentes | Tiempo de QA, Dev y atención | $10,000 MXN |
| Total estimado de pérdidas económicas: |  | → **$109,000 MXN** |

**2. Costo de Implementación de Estrategias de Control**

**A. Costos de Medidas Preventivas (Antes del riesgo)**

|  |  |
| --- | --- |
| Acción | Costo estimado |
| Diseño de arquitectura escalable desde el inicio | $4,000 MXN |
| Estimación de escalabilidad y tráfico potencial | $2,000 MXN |
| Auditoría técnica temprana | $2,500 MXN |
| Documentación técnica de APIs y estructura modular | $2,500 MXN |
| 🟦 Total medidas preventivas: | **$11,000 MXN** |

**B. Costos de Medidas de Mitigación (Durante el riesgo)**

|  |  |
| --- | --- |
| Acción | Costo estimado |
| Refactorización parcial y desacoplamiento de módulos críticos | $6,000 MXN |
| Escalado de infraestructura temporal | $3,000 MXN |
| Implementación de cachés y colas | $2,500 MXN |
| Total medidas de mitigación: | **$11,500 MXN** |

**C. Costos de Medidas Correctivas (Después del riesgo)**

|  |  |
| --- | --- |
| Acción | Costo estimado |
| Migración técnica progresiva hacia arquitectura escalable | $7,000 MXN |
| Auditoría y documentación de la situación técnica | $2,000 MXN |
| Políticas internas de revisión arquitectónica | $1,500 MXN |
| Total medidas correctivas: | **$10,500 MXN** |

**Resumen Económico del Plan de Contingencia**

|  |  |
| --- | --- |
| Categoría | Costo estimado |
| 🛡️ Prevención | $11,000 MXN |
| 🚨 Mitigación | $11,500 MXN |
| 🔄 Recuperación | $10,500 MXN |
| 💥 Costo de no hacer nada | $109,000 MXN |

**📈 Análisis Costo-Beneficio**

**Costo total de implementar todas las estrategias:**  
$11,000 + $11,500 + $10,500 = **$33,000 MXN**

**Ahorro potencial si se previene o controla el riesgo:**  
$109,000 – $33,000 = **$76,000 MXN**

**(≈ 230% de retorno sobre la inversión en gestión del riesgo)**