**RIESGO 2: No detección de errores en ambiente de desarrollo**

• Tipo de riesgo: Técnico / Calidad  
• Categoría: QA / DevOps / Infraestructura  
• Descripción: El riesgo ocurre cuando los errores no se detectan en el entorno de desarrollo debido a diferencias con el entorno de producción, mala configuración o pruebas insuficientes. Esto genera defectos inesperados tras el despliegue y pérdida de confiabilidad del producto.

## 1. IDENTIFICACIÓN DEL RIESGO

|  |  |
| --- | --- |
| Nombre del riesgo | No detección de errores en ambiente de desarrollo |
| Categoría | QA / DevOps / Infraestructura |
| Causas probables | Ambientes mal configurados, diferencias entre entornos, falta de staging, pruebas solo en local. |
| Consecuencias | Errores en producción, retrabajo, pérdida de tiempo y confianza del cliente. |
| Fuente | Entorno de desarrollo, pipeline de integración, configuración del entorno. |

## 2. EVALUACIÓN DEL RIESGO

|  |  |
| --- | --- |
| Probabilidad de ocurrencia | Alta (4/5) |
| Impacto potencial | Alto (4/5) |
| Nivel de riesgo | Muy alto |
| Indicadores de riesgo | Errores no detectados localmente, fallos tras despliegue, diferencias de comportamiento. |

## 3. DEFINICIÓN DE MEDIDAS DE CONTROL

## A. Medidas Preventivas (Antes del riesgo)

- Crear ambientes de staging que simulen producción. DevOps Docker, Kubernetes  
- Configurar pipelines con validaciones previas al deploy. DevOps / QA CI/CD, GitHub Actions  
- Unificar configuraciones entre entornos. DevOps Dotenv, Variables de entorno  
- Hacer pruebas automáticas en staging. QA TestRail, Selenium  
- Documentar diferencias entre entornos y resolverlas. DevOps / QA Wiki interna, auditorías

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

- Repetir pruebas en staging con datos similares a producción. QA Base de datos anonimizada  
- Desplegar en entorno canario antes de producción. DevOps Feature flags, blue-green deploy  
- Hacer rollback rápido si se detecta el fallo. DevOps Versionado y backups

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

- Analizar causa raíz de los errores post-despliegue. QA / Dev RCA, logs  
- Actualizar documentación de configuración y diferencias. QA / DevOps Wiki técnica  
- Establecer revisión de configuración antes de cada release. QA / PMO Checklist técnico

## 4. Estimación de Pérdidas Económicas por Riesgo No Controlado

|  |  |  |
| --- | --- | --- |
| Escenario de materialización | Impacto directo | Costo estimado (MXN) |
| Errores críticos tras despliegue | Corrección urgente y retrabajo | $20,000 |
| Pérdida de datos o funcionalidad | Afectación al cliente y reputación | $30,000 |
| Tiempo extra de QA y Dev | Retrasos y corrección manual | $12,000 |
| Soporte post-lanzamiento | Atención al cliente y hotfix | $8,000 |

Total estimado de pérdidas económicas: → $70,000 MXN

## 5. Costo de Implementación de Estrategias de Control

A. Costos de Medidas Preventivas:  
- Configuración de ambiente staging: $5,000 MXN  
- Pruebas automatizadas en staging: $3,500 MXN  
- Sincronización de configuraciones y documentación: $2,500 MXN  
Total medidas preventivas: $11,000 MXN  
  
B. Costos de Medidas de Mitigación:  
- Validación en staging real: $3,000 MXN  
- Feature flags y despliegue controlado: $2,000 MXN  
Total medidas de mitigación: $5,000 MXN  
  
C. Costos de Medidas Correctivas:  
- Análisis postmortem y checklist: $2,500 MXN  
- Refuerzo en revisión de configuración: $1,500 MXN  
Total medidas correctivas: $4,000 MXN

📊 Resumen Económico del Plan de Contingencia  
- 🛡️ Prevención: $11,000 MXN  
- 🚨 Mitigación: $5,000 MXN  
- 🔄 Recuperación: $4,000 MXN  
- 💥 Costo de no hacer nada: $70,000 MXN  
  
📈 Análisis Costo-Beneficio  
Costo total de estrategias: $20,000 MXN  
Ahorro potencial: $70,000 – $20,000 = $50,000 MXN (≈ 250% de ROI)