EXAMPLE: IT - SYSTEM DOWNTIME

PROBLEM: E-Commerce Platform Unplanned Downtime - Average 4.2 hours/month of unplanned downtime causing €180K monthly revenue loss

Plan

1 – Clarify the Problem

• Define Ultimate Goal, Current Situation, Standard and Gap/Problem

Ultimate Goal: 99.99% platform availability (max 4.3 min/month downtime)
Current Situation: 99.4% availability (4.2 hours/month downtime)
Standard: SLA commitment 99.9%, industry best 99.99%
Gap: 0.5% below SLA, significant revenue impact
Business Impact: €180K/month lost revenue, 12% customer churn

2 – Breakdown the Problem

• Breakdown the problem into sub-problems and prioritize
• Identify the Point of Occurrence

Sub-problems identified:
• Database connection pool exhaustion (35%)
• Memory leaks in payment service (25%)
• Cascade failures from third-party API (20%)
• Deployment-related incidents (15%)

3 – Set the Target

• Measurable targets: What? How Much? When?

WHAT: Unplanned downtime
HOW MUCH: Reduce from 4.2 hours to <5 minutes per month
WHEN: Within 8 weeks

Supporting Targets:
• MTTR: 45 min → 5 min

4 – Analyze the Root Cause

• Graphical display of the most likely direct root causes

5-Why Analysis (Database Issue):
1. Why downtime? → Database connections exhausted
2. Why exhausted? → Connections not released properly
3. Why not released? → Timeout too long, no circuit breaker
4. Why no circuit breaker? → Legacy architecture
5. Why legacy? → Technical debt not prioritized

Root Causes: No auto-scaling, missing circuit breakers, inadequate monitoring

Do

5 – Develop and Implement Countermeasures

• What? • Who? • When? • Where?

WHAT:
1. Implement circuit breaker pattern (Resilience4j)
2. Deploy auto-scaling with predictive algorithms
3. Set up comprehensive APM monitoring
4. Implement blue-green deployments with auto-rollback
5. Add chaos engineering tests

WHO: Platform Team Lead, SRE, DevOps, Architecture
WHEN: Week 1-4: Implementation | Week 5-6: Automation | Week 7-8: Testing
WHERE: Production environment (staging validation first)

Check

6 – Evaluate Results and Processes

• What measurable results did the solution achieve?
• Who is responsible for ongoing measurement?

RESULTS (after 8 weeks):
✓ Downtime: 4.2 hours → 3 minutes/month (99.99% achieved!)
✓ MTTR: 45 min → 4 min (auto-recovery)
✓ MTBF: 7 days → 120+ days
✓ Zero P1 incidents during last 3 promotions

BUSINESS IMPACT:
• Revenue recovered: €175K/month
• SLA penalties: €0 (was €25K/month)

Act

7 – Standardize Processes

• Where else can the solution be applied?
• How will the improved state be standardized?

WHERE ELSE (YOKOTEN):
• Mobile app backend services
• Internal tools and admin portals
• Partner API gateway

STANDARDIZATION:
• Architecture Decision Records updated
• New microservice template with resilience built-in
• Runbook library in Confluence (25 scenarios)

CULTURE CHANGE: "Reliability First" principle adopted

NEXT KAIZEN: Deployment frequency to 10x/day

Date:

Team: