Crisis Analysis Worksheet

Name - Zarni Tun ID - 6531503195

Case Study Analysis — The Peak-Hour Collapse

The Core Failure

The leading mobile banking application in Thailand became completely unusable during predictable peak-hour events due to millions of users across the country using their banking app at the same time.

Identify the Stakeholders

1. **Regular banking customers** – Unable to log in, transfer money, or pay for goods/services during peak hours.
2. **Merchants and online sellers** – Lost sales due to failed payments.
3. **The bank** – Suffered financial loss, reputational damage, and increased customer complaints.
4. **Government regulators (Bank of Thailand)** – Forced to intervene and impose new requirements.

Rank the Quality Failures

1. **Scalability/Performance** – unable to handle predictable peak-hour transaction loads.
2. **Reliability** – no fallback mechanisms; the entire app failed instead of isolating non-critical services.
3. **Usability** – generic error messages with no guidance or recovery options.
4. **Correctness** – risk of incomplete or failed transactions causing inconsistent account balances.

5. **Security** – no direct breach, but prolonged downtime could expose vulnerabilities.

Justification

Scalability was the most critical failure because the outage occurred during entirely predictable peak usage times. Without the ability to handle surges in transactions, all other service qualities became irrelevant — customers could not even access the system.

Root Cause Analysis

Chosen SDLC Phase: B) Architecture & Design

Explanation

The system was not architected for elastic scaling or fault isolation. The design failed to anticipate predictable peak loads, lacked load balancing across components, and did not plan for graceful degradation of non-critical services under stress.

Proactive Prevention Plan

Most Important Action during the DESIGN Phase

Implement a microservices architecture with autoscaling and load balancing to distribute traffic and isolate failures.

Most Important Action during the TESTING Phase

Conduct realistic stress and load testing simulating peak-hour traffic patterns and transaction volumes.

Case Study Analysis — The Registration Day Crash

The Core Failure

The national government registration website became unresponsive and failed to process submissions during the opening of a high-demand “first come, first served” program due to its inability to handle a predictable, massive spike in concurrent users.

Identify the Stakeholders

1. **Citizens** – Unable to register despite being ready on time; experienced frustration and sense of unfairness.

2. **Government agencies** – Lost credibility and faced operational backlogs due to duplicate entries and failed submissions.

3. **Media & public commentators** – Amplified negative public perception through widespread reporting.

4. **Program beneficiaries** – Faced delays in receiving benefits due to administrative chaos.

Rank the Quality Failures

1. **Scalability/Performance** – unable to process the predictable surge of millions of simultaneous requests.

2. **Reliability** – system crashed entirely without fallback mechanisms

3. **Usability** – no helpful feedback, leading to repeated submissions and user confusion.

4. **Correctness** – duplicate entries created risk of data inconsistencies.

5. **Security** – no direct breach occurred, but instability increases exposure risk.

Justification

The core failure was the inability to handle a known, scheduled surge. The government had advanced knowledge of the opening time and potential demand but failed to provision or scale resources accordingly, making scalability the critical weak point.

Root Cause Analysis

Chosen SDLC Phase: B) Architecture & Design

Explanation

The system lacked a design for handling flash-crowd events, such as implementing elastic scaling, distributed request throttling, and virtual queueing. This deficiency, combined with unrealistic load testing, made the crash inevitable.

Proactive Prevention Plan

Most Important Action during the DESIGN Phase

Incorporate a virtual queue and distributed load management system to control simultaneous access and preserve stability.

Most Important Action during the TESTING Phase

Simulate “Day One” flash crowd scenarios with millions of concurrent virtual users to validate capacity and response times.

Case Study 3 — The 55-Million Record Leak

The Core Failure

A massive data breach exposed 55 million Thai citizens' sensitive personal information due to weak security controls, poor data protection practices, and the absence of robust preventative measures, resulting in irreversible privacy loss and widespread fraud risks.

Identify the Stakeholders

1. **Thai citizens** – Suffer irreversible loss of privacy, increased risk of identity theft, fraud, and scams.

2. **Affected organization (government agency or company)** – Faces legal penalties, massive reputational damage, and loss of public trust.

3. **Law enforcement & regulators** – Required to investigate the breach and enforce data protection laws, creating resource strain.

4. **Businesses & financial institutions** – Must implement additional verification and fraud prevention measures due to compromised identities.

Rank the Quality Failures

1. **Security** – weak security controls, likely vulnerabilities, lack of encryption, and overly broad user access.

2. **Correctness** – failure to preserve integrity of sensitive data through secure storage and access controls.

3. **Reliability** – inability to maintain a trustworthy and resilient data protection environment.

4. **Usability** – indirectly impacted as citizens must navigate additional security checks and processes due to breach.

5. **Scalability/Performance** – not a direct cause but may have influenced choice of insecure shortcuts in design.

Justification

Security is the most critical failure because this incident involved the direct exposure of sensitive, immutable PII. Once compromised, this type of data cannot be revoked or changed, making the consequences long-term and severe for millions of people.

Root Cause Analysis

Chosen SDLC Phase: B) Architecture & Design

Explanation

The breach likely stemmed from architectural flaws and inadequate security-by-design principles — such as storing sensitive data in plaintext, not enforcing least privilege access, and lacking regular penetration testing. These weaknesses made exploitation via SQL injection, cloud misconfigurations, or stolen credentials possible.

Proactive Prevention Plan

Most Important Action during the DESIGN Phase

Implement **security-by-design** measures: encrypt all sensitive data with strong encryption, enforce least privilege access, and integrate multi-factor authentication for administrative accounts.

Most Important Action during the TESTING Phase

Perform **recurring penetration tests** and vulnerability scans to identify and patch security gaps before attackers can exploit them.