Extra Slide : Risk in Software Development

Md. Mehedi Hasan Rafy

1. What is Software Risk?

- ***Definition****:* The probability of an *unwanted event* occurring in software development that can negatively impact cost, schedule, or quality.
- Risk combines two factors:
  - ***Uncertainty*** – probability that the event will occur.
  - ***Impact*** – consequences if the event occurs.
- ***Formula****:* $Risk = Probability \times Impact\ of\ Loss$

2. Why Perform Software Risk Analysis?

- To anticipate problems before they occur.
- To prioritize risks based on severity and probability.
- To make informed decisions about design, scheduling, budgeting, and resource allocation.
- To minimize surprises and ensure smoother project execution.
- Industry insight: Many projects fail not because of bad coding, but because risks (like scope creep or unrealistic deadlines) weren't managed.

3. Types of Software Risks

i. ***Security Risk***

- Unauthorized access, data leaks, or attacks.
- Example: Poor authentication → system breach.

ii. ***Performance Risk***

- System fails to meet speed, scalability, or reliability needs.
- Example: High load causes server crashes.

iii. ***Budgetary Risk***

- Project cost exceeds estimates.
- Example: Underestimating testing costs.

iv. ***Contractual & Legal Risk***

- Breach of agreements, licensing violations, compliance issues.
- Example: Using third-party software without proper license.

v. ***Operational Risk***

- Risks from day-to-day operation of the system.
- Example: System downtime due to lack of monitoring.

vi. ***Schedule Risk***

- Delays due to unrealistic timelines, resource shortage, or scope creep.
- Example: Adding features mid-project without adjusting schedule.

4. Risk Management

- A structured process of identifying, analyzing, monitoring, and controlling risks in software development.
- ***Goal****:* Minimize both the *probability* and the *impact* of risks.
- ***Steps****:*
    1. ***Identify risks*** **–** brainstorm, checklists, past projects.
    2. ***Assess risks*** **–** evaluate likelihood & impact.
    3. ***Plan responses*** **–** decide mitigation strategies.
    4. **Monitor –** track risks continuously during project.

5. Risk Assessment

- Evaluating risks based on risk exposure which is calculated using likelihood (probability) of happening and impact (severity) of the risk.
- Represented often using a Risk Table.

Risk Exposure (RE)

- Quantifies risk in terms of *expected loss*.
- ***Formula****:* $RE = P \times C$

  where, **RE =** Risk Exposure, **P** = probability (0–1), **C** = cost/impact of risk

Example 1: Schedule Delay	***Example 2:*** Security Breach
• Probability = 0.3 • Impact = $60,000 • RE = 0.03 × 60,0000 = $18,000	• Probability = 0.1 • Impact = $500,000 • RE = 0.1 × 500,000 = $50,000

So, if risk is less likely to happen but causes more damage, it is taken more seriously than the other risk based on risk exposure.

Risk Table

- A tabular method to organize risks systematically.
- Helps in comparing risks and choosing priorities.
- Typical columns: Risk | Probability | Impact | Risk Exposure | Response

Example Risk Table – E-commerce System

Risk	Probability	Impact	Exposure (P×C)	Mitigation
Payment gateway failure	High (0.7)	High	Critical	Backup gateway, monitoring
Data breach / cyberattack	Medium (0.5)	High	Significant	Encryption, penetration testing
Server overload in sales	High (0.8)	Medium	High	Load testing, auto-scaling
Team attrition (resignation)	Medium (0.4)	Medium	Medium	Cross-training, documentation
Delay in vendor API	Low (0.2)	High	Low-Medium	SLA agreements, backup API

6. Risk Control

- Steps taken to minimize or eliminate risks.
- Techniques:
    - Avoidance (remove risky feature).
    - Mitigation (add safeguards, e.g., extra testing).
    - Transfer (insurance, outsourcing).
    - Acceptance (live with it if cost > benefit).

7. Benefits of Risk Analysis

- Better decision-making in planning & execution.
- Higher quality and reliability of software.
- Reduced chances of project failure.
- Efficient resource allocation (time & money spent on the right risks).
- Greater stakeholder confidence in the project.

8. Example Case Analysis

Case: Online Banking System.

- Risks:
    - Security → hacking attempts.
    - Performance → slow response during peak hours.
    - Schedule → deadline pressure from regulators.
- **Risk management**:
    - Security → multi-factor authentication + penetration testing.
    - Performance → load testing + cloud scaling.
    - Schedule → buffer time + agile releases.