Risk management is carried out to:

1. Identify the risk
2. Reduce the impact of risk
3. Reduce the probability or likelihood of risk
4. Risk monitoring

Software risk management is all about risk quantification of risk. This includes:

1. Giving a precise description of risk event that can occur in the project
2. Defining risk probability that would explain what are the chances for that risk to occur
3. Defining How much loss a particular risk can cause
4. Defining the liability potential of risk

With impact we mean the consequence of a risk in case it happens. It is important to know about the impact because it is necessary to know how a business can get affected:

1. What will be the loss to the customer
2. How would the business suffer
3. Loss of reputation or harm to society
4. Monetary losses
5. Legal actions against the company
6. Cancellation of business license

Risk Management comprises of following processes:

1. Software Risk Identification
2. Software Risk Analysis
3. Software Risk Planning
4. Software Risk Monitoring

Risk identification:

* Tight timelines
* Undefined project scope
* Insufficient resources
* Unavailable prerequisites
* Continuously changing requirements
* Unresolved, misapplied, unrecognized metrics
* Note verifying test scripts, in this specific case

*Scheduling Risk*: Testing projects are not efficiently or completely scheduled to meet the deployment deadline. Inefficiency in scheduling can include:

* Inaccurate time estimates
* Improper assessment of required tool resources
* Improper assessment of required manpower resources
* Unanticipated expansion(s) in project scope
* Inaccurate identification of complexities, functionalities, or operations

*Budget Risk:* Required investment is inaccurately anticipated, including:

* *Inaccurate Cost Estimation:*Certain required items excluded from the estimation of costs
* *Cost Overruns:*Unanticipated expenses, or inaccurate estimation, have cause unanticipated expenses
* *Expansion of the project scope:*The project scope is expanded to include initially unanticipated expenses.

*Operational Risk:*Ineffective processing, system failures, or unanticipated circumstances define operational risk. Causes include:

* Failure to establish testing priorities
* Conflicting test priorities
* Insufficient resources
* Improper training
* Improper communication among team members
* Improper communication with enterprise stakeholders

*Technical Risk:*Technical risks often lead to functionality and performance failures. Some causes include:

* Continually changing requirements
* Lack of technical resources
* Product complexities

*General Risk:*

* Changes in market strategies
* Changes in government regulations
* Changes in customer demands and interests

**Treating Identified Risk**

At times risk is identified after-the-fact. When risk happens despite upfront assessments, it can possibly be treated in one of four ways:

* Risk mitigation – Renewed planning to avoid the risk.
* Risk acceptance – The risk was not eliminated in prior releases, and is therefore accepted in the current release.
* Risk transfer – Outsourcing the solution to a specialty risk-treatment company.
* Risk surrender – Postponing the application development for a later release, significantly impacting by reduction enterprise ROI.

Avoidance of loss, or risk mitigation, is best ensured by incrementally reviewing the testing plan, procedures, modules, and metrics to assess how precisely risk planning was executed:

* Were risk mitigation measures effectively defined and executed?
* How effectively did measures designated to risk mitigation ensure successful deployment outcomes?
* Were measures to reduce the impact of unmitigated risk defined and executed?
* How effective were the measures designated to reduce the impact of unmitigated risk in ensuring successful deployment outcomes?
* What steps can be taken to improve risk mitigation in future projects?