RISK MANAGEMENT: What is it?

- Analysis and management are a serious of steps that helps a software team to understand and manage uncertainty.
- Many problems can plague a software project.
- A risk is a potential problem - it might happen, it might not.
But, regardless of the outcomes, it’s a really good idea to identify it, assess its probability of occurrence, estimate its impact, and establish a contingency plan should the problem actually occur...
WHY IS IT IMPORTANT?

Think about the boy scout motto:

Be prepared software is a difficult under taking. Lots of things can go wrong, and frankly, many often do.

It’s for this reason that being prepared—understanding the risk and taking proactive measures to avoid or manage them—is a key element of good software project management...
WHAT ARE THE STEPS OF RISK MANAGEMENT?

Recognizing what can go wrong is the first step, called “risk identification”. Next, each risk is analyzed to determine the likelihood that it will occur. Once this information is established, risks are ranked, by probability and impact. Finally, a plan is developed to manage those risks with high probability and high impact.
Reactive risk strategies have been laughingly called the “Indiana Jones school of risk management”. In the 1980s-era movies that carried his name, Indiana Jones, when faced with overwhelming difficulty, would invariably say, “don’t worry, I’ll think of something!” Never worrying about problems until they happened, Indy would react in some way.
“If you don’t actively attack they risk, they will actively attack you”
TOM GILLB...
Sadly, the average software project manager is not Indiana Jones, and the members of the software project team or not his trusty sidekicks. The majority of software teams rely solely on reactive risk strategies.
At best, a reactive strategy monitors the project for likely risks. Resources are set aside to deal with them, should they become actual problem. More commonly, the software team does nothing about risks until something goes wrong.
Then, the team files into action in an attempt to correct the problem rapidly. This is often called a firefighting mode. When this fails, "crisis management"[CHA92] takes over and the project is in real jeopardy.
A considerably more intelligent strategy for risk management is to be proactive. A proactive strategy being long before technical work is initiated. Potential risks are identified, their probability and impact are assessed, and they are ranked by importance.
Then, the s/w team establish a plan for managing risk. The primary objective is to avoid risk, but because not all risks can be avoided, the team works to develop a contingency that will enable it to respond in a controlled and effective manner.
Throughout reminder of this chapter, we discuss a proactive strategy for risk management.
To Mitigate the risk

- Project management must develop a strategy for reducing turnover.

- Possible steps to be taken are:
  - Meet with current staff to determine causes for turnover.
  - Once the project commences, assume turnover will occur and develop techniques to ensure continuity when people leave.
• Organize project teams so that information about each development activity is widely dispersed.

• Define documentation standards and establish mechanisms to ensure that documents are developed in a timely manner.
Conduct peer reviews of all work.

Assign a backup staff member for every critical technologist.

**NOTE:**
If RE for a specific risk is less than the cost of risk mitigation, don’t try to mitigate the risk but continue to monitor it.
RMMM

• **Definition:**

  - The Risk Mitigation, Monitoring and Management, RMMM, plan documents all work performed as part of risk analysis and is used by the project manager as part of overall project plan.
  - Once RMMM has been documented and the project has begun, risk mitigation and monitoring steps commence.
Risk Mitigation covers efforts taken to reduce either the probability or consequences of a threat.

Risk monitoring and control is the process of identifying, analyzing, and planning for newly discovered risks and managing identified risks.

Risk management is the identification, assessment, and prioritization of risks.
RMMM PLAN

• The RMMM PLAN documents all work performed as part of risk analysis and used by the project manager as part of the overall project plan

• Some software teams do not develop a formal RMMM document, rather each risk is documented individually using a Risk information sheet.
• In most cases, **RIS** is maintained using a database system.

• So Creation and information entry, priority ordering, searches and other analysis may be accomplished easily.

• The format of **RIS** is describe in diagram:-
## Risk information sheet

**Risk ID:** P02-4-32  
**Date:** 5/9/04  
**Prob:** 80%  
**Impact:** high

### Description:
Only 70 percent of the software components scheduled for reuse will, in fact, be integrated into the application. The remaining functionality will have to be custom developed.

### Refinement/context:
- **Subcondition 1:** Certain reusable components were developed by a third party with no knowledge of internal design standards.
- **Subcondition 2:** The design standard for component interfaces has not been solidified and may not conform to certain existing reusable components.
- **Subcondition 3:** Certain reusable components have been implemented in a language that is not supported on the target environment.

### Mitigation/monitoring:
1. Contact third party to determine conformance with design standards.
2. Press for interface standards completion; consider component structure when deciding on interface protocol.
3. Check to determine number of components in subcondition 3 category; check to determine if language support can be acquired.

### Management/contingency plan/trigger:
- **RE** computed to be $20,200. Allocate this amount within project contingency cost.
- Develop revised schedule assuming that 18 additional components will have to be custom built; allocate staff accordingly.
- **Trigger:** Mitigation steps unproductive as of 7/1/04

### Current status:
- **5/12/04:** Mitigation steps initiated.

**Originator:** D. Gagne  
**Assigned:** B. Loster
A good way to describe a risk is to represent the risk in Condition-transition-consequences (CTC) format.

The risk is stated in the following form:

*Given that* <condition> *then there is concern that (possibly) <consequence>*
• Using the **CTC** format for the reuse risk, we can write:-

• Given that all reusable components must conform to specific design standards and that some do not conform, then there is concern that (possibly) only 70% of the planned reusable modules may actually be integrated into the as-built system, resulting in the need to custom engineer the remaining 30% of components.
• The general condition can be refined in the following manner:

• **Sub-condition 1:**
  • Certain reusable components were developed by a third party with no knowledge of internal design standards.
**Sub-condition 2:**
The design standard for component interface has not been solidified and may not confirm to certain existing reusable components.

**Sub-condition 3:**
Certain reusable components have been implemented in a language that is not supported on the target environment.
• The method for identifying risk is to create a risk item checklist. The checklist can be used for risk identification and focuses on some subset of known and predictable risk in the following generic subcategories:

- **Product size** - Risk associated with the overall size of the software to be built or modified.