EXPERIMENT – 04

Aim: Analyze the risk related to the project and prepare RMMM plan

Objective: Analyzing the risks while developing
Attendance Management system and preparing RMMM plan

Theory:

The proactive management of risks throughout the software development lifecycle is important for project success

- The risk management practice, which involves risk identification, analysis, prioritization, planning, mitigation, monitoring, and communication
- Software development risks that seem to reoccur in educational and industrial projects
- A risk-driven process for selecting a software development model

Risk in itself is not bad; risk is essential to progress, and failure is often a key part of learning. But we must learn to balance the possible negative consequences of risk against the potential benefits of its associated opportunity.

A risk is a potential future harm that may arise from some present action (Wikipedia, 2004), such as, a schedule slip or a cost overrun. The loss is often considered in terms of direct financial loss, but also can be a loss in terms of credibility, future business, and loss of property or life.

THE RISK MANAGEMENT

The risk management process can be broken down into two interrelated phases, risk assessment and risk control, as outlined in Figure 1. These phases are further broken down. Risk assessment involves risk identification, risk analysis, and risk prioritization. Risk control involves risk planning, risk

mitigation, and risk monitoring.(Boehm, 1989). It is essential that risk management be done iteratively, throughout the project, as a part of the team's project management routine

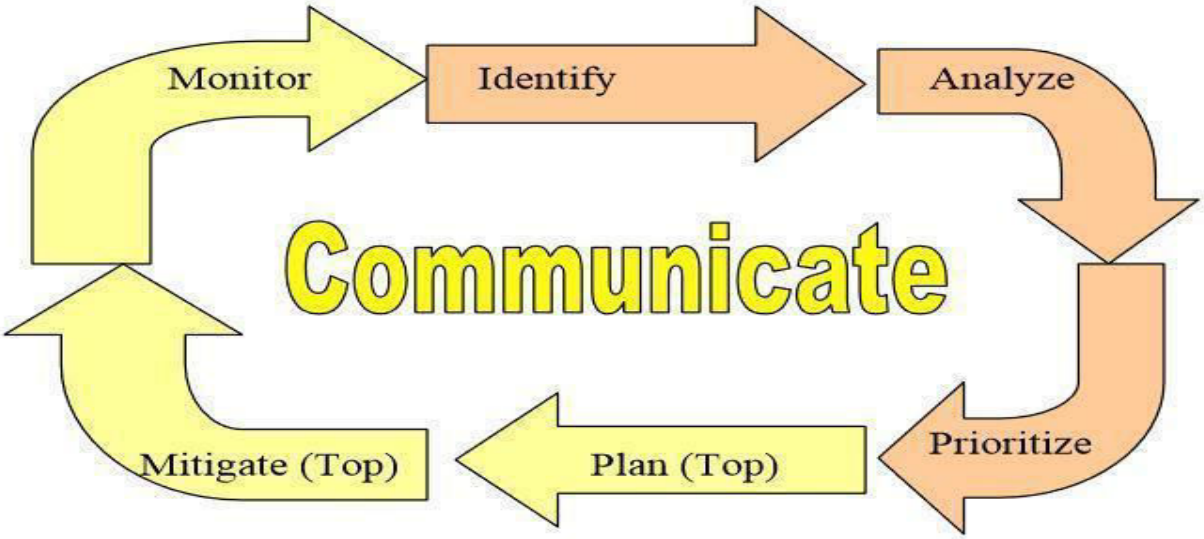

Figure 1: The Risk Management Cycle.

RISK IDENTIFICATION

In the risk identification step, the team systematically enumerates as many project risks as possible to make them explicit before they become problems. There are several ways to look at the kinds of software project risks, as shown in Table 1. It is helpful to understand the different types of risk so that a team can explore the possibilities of each of them. Each of these types of risk is described below.

Table 1: General Categories of Risk

Generic Risks		Product-Specific Risks
Project Risks	Product Risks	Business Risks
Factors to consider: People, size, process, technology, tools, organizational, managerial, customer, estimation, sales, support		

Generic risks are potential threats to every software project. Some examples of generic risks are changing requirements, losing key personnel, or bankruptcy of the software company or of the customer. It is advisable for a development organization to keep a checklist of these types of risks. Teams can then assess the extent to which these risks are a factor for their project based upon the known set of programmers, managers, customers, Identify Analyze Prioritize Plan (Top) Mitigate (Top) Monitor Risk Management and policies.

.

There are some specific factors to consider when examining project, product, and business risks. Some examples of these factors are listed here, although this list is meant to stimulate your thinking rather than to be an all-inclusive list.

TYPES OF RISKS INVOLVED

- **People risks** are associated with the availability, skill level, and retention of the people on the development team.

- **Size risks** are associated with the magnitude of the product and the product team. Larger products are generally more complex with more interactions. Larger teams are harder to coordinate.

- **Process risks** are related to whether the team uses a defined, appropriate software development process and to whether the team members actually follow the process.

- **Technology risks** are derived from the software or hardware technologies that are being used as part of the system being developed. Using new or emerging or complex technology increases the overall risk.

- **Tools risks**, similar to technology risks, relate to the use, availability, and reliability of support software used by the development team, such as development environments and other Computer-Aided Software Engineering (CASE) tools.

- **Organizational and managerial risks** are derived from the environment where the software is being developed. Some examples are the financial stability of the company and threats of company reorganization and the potential of the resultant loss of support by management due to a change in focus or a change in people.

- **Customer risks** are derived from changes to the customer requirements, customers' lack of understanding of the impact of these changes, the process of managing these requirements changes, and the ability of the customer to communicate effectively with the team and to accurately convey the attributes of the desired product.

- **Estimation risks** are derived from inaccuracies in estimating the resources and the time required to build the product properly.

- **Sales and support risks** involve the chances that the team builds a product that the sales force does not understand how to sell or that is difficult to correct, adapt, or enhance.

RISK MITIGATION

Related to risk planning, through risk mitigation, the team develops strategies to reduce the possibility or the loss impact of a risk. Risk mitigation produces a situation in which the risk items are eliminated or otherwise resolved.

Risk avoidance. When a lose-lose strategy is likely, the team can opt to eliminate the risk is an example of a risk avoidance strategy is the team opting not to develop a product or a particularly risky feature.

Risk protection. The organization can buy insurance to cover any financial loss should the risk become a reality. Alternately, a team can employ fault-tolerance strategies, such as parallel processors, to provide reliability insurance.

Risk planning and risk mitigation actions often come with an associated cost. The team must do a cost/benefit analysis to decide whether the benefits accrued by the risk management steps outweigh the costs associated with

implementing them. This calculation can involve the calculation of risk leverage.

Risk Leverage = (risk exposure before reduction – risk exposure after reduction)/cost of risk reduction

If risk leverage value, rl, is ≤1, clearly the benefit of applying risk reduction is not worth its cost. If rl is only slightly > 1, still the benefit is very questionable, because these computations are based on probabilistic estimates and not on actual data. Therefore, rl is usually multiplied by a risk discount factor ρ< 1. If ρ rl > 1, then the benefit of applying risk reduction is considered worth its cost. If the discounted leveraged valued is not high enough to justify the action, the team should look for other, less costly or more effective, reduction techniques.

RISK MONITORING

After risks are identified, analyzed, and prioritized, and actions are established, it is essential that the team regularly monitor the progress of the product and the resolution of the risk items, taking corrective action when necessary. This monitoring can be done as part of the team project management activities or via explicit risk management activities.
Often teams regularly monitor their “Top 10 risks.”

Risks need to be revisited at regular intervals for the team to reevaluate each risk to determine when new circumstances caused its probability and/or impact to change. At each interval, some risks may be added to the list and others taken away. Risks need to be reprioritized to see which are moved

“above the line” and need to have action plans and which move “below the line” and no longer need action plans. A key to successful risk management is that proactive actions are owned by individuals and are monitored.

RMMM PLAN:

The RMMM plan is a document in which all the risk analysis activities are described. Sometimes project manager includes this document as a part of overall project plan. sometimes specific RMMM Plan is not created, however each risk can be described individually using risk information sheet. Typical template for RMMM Plan or Risk Information sheet can be as follows.

<table>
<tr><th colspan="4">Risk Information sheet</th></tr>
<tr><td colspan="4">Project Name < Enter Name of the project for which risk has to be identified></td></tr>
<tr><td>Risk id < # ></td><td>Date
<date at which risk is identified></td><td>Probability
<Risk Probability % ></td><td>Impact
<low/medium/high></td></tr>
<tr><td colspan="2">Origin
< The person who have identified the risk></td><td colspan="2">Assigned to
<Who is responsible for mitigating the risk></td></tr>
<tr><td colspan="4">Description
<Description of risk identified></td></tr>
<tr><td colspan="4">Refinement/Context
<Associated information for risk refinement></td></tr>
<tr><td colspan="4">Mitigation/ Monitoring
<Enter the mitigation / Monitoring steps taken></td></tr>
<tr><td colspan="4">Trigger/Contingency Plan
<if Risk Mitigation fails then the pan for handling the risk></td></tr>
<tr><td colspan="4">Status
<Running status that provides a history of what is being done for the risk and changes in the risk. Include the date the status entry was made></td></tr>
<tr><td colspan="2">Approval
< Name & Signature of person approving closure></td><td colspan="2">Closing Date
<Date></td></tr>
</table>

Sample RMMM:

Risk information sheet			
Risk ID: P02-4-32	Date: 5/9/02	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 7/1/02			
Current status: 5/12/02: Mitigation steps initiated.			
Originator: D. Gagne		Assigned: B. Laster	

Example: RMMM Plan for Attendance Maintenance system

Risk Information sheet			
Project Name < Attendance Maintenance system>			
Risk id < 1 >	**Date** < 13/11/15 >	**Probability** < 20 % >	**Impact** <medium>
Origin < Yajat Surya>		**Assigned to** <Bhakti Shelar>	
Description <Summation of attendance is not working after each month>			
Refinement/Context <Update the software program and check for the summation function>			
Mitigation/ Monitoring <See that every module is in working condition or not>			
Trigger/Contingency Plan <Call the developer of the software and ask him/her to correct the module which is not working properly>			
Status <1. Contacted the developer of the software purchased : 15/11/15 2 Attendance is taken manually till the module gets worked : 17/11/15 3 Developer corrects the module : 18/11/15>			
Approval < Srinadh Swamy >		**Closing Date** <18/11/15>	