ACKNOWLEDGEMENTS

I would like to thank my professor Mr. Themistokleous for his guidance and his assistance when needed, he was always on call whenever I needed him.

Additionally two people in particular, Marilia for her great support and my friend Thomas for his valuable and in deep knowledge of MS Visual Studio.

TABLE OF CONTENTS

1. INTRODUCTION

1.1 SUMMARY

Success in the business world, involves taking some risks. The systems that change the world today are very risky, but the payback is worth that risk and more. One needs to know how to manage risk. This includes how to identify risk sources, quantify risk parameters, and develop plans to handle risk. Risks are inevitable in IT projects and systems. The high failure rate of modern large IT projects and systems, such as those involving ERP, CRM and SCM, is largely due to senior management and project management's failure to assess risks up front and to mitigate the causes of the greatest risks at the very start.

An adequate analysis of potential risks can significantly increase the likelihood of success for a project and can justify money put aside for management reserves. Risk is the possibility of suffering loss. In IT, the loss may involve increased costs, longer completion times, reduced scope, reduced quality or reduced stakeholder satisfaction. Risk and opportunity are different sides of the same coin. Some IT projects advance the state of the art, and as such are more risky than those that do not. The opportunity for significant advancement cannot be done without significant risk. Risk is essential to progress, and failure is often a key part of learning. We must learn to balance the possible negative consequences of risk against the potential benefits of its associated opportunity.

In other words to lay a foundation for a discussion of the role of information systems in risk management, we must first define the business needs that drive organizations and firms to implement risk management functions and the Risk Management Process.

These needs appear below:

1. Initially, to better understand the risks it is taking, an organization wants to identify them and measure them. Risks that can be quantified are in most cases market, financial and credit risks.
2. To provide better incentives to its business units and to individual employees, an organization wants to reward good risk-adjusted performance. The organization must measure its risk or risks before it can adjust performance for risk
3. To provide its stakeholders with a consistent and optimal risk-return tradeoff over time, an organization has to accurately match the amount of capital it employs with the risk it takes.

Therefore the induction of Risk identification, Risk quantification, Risk response control and Risk response development as processes is crucial. Risk management information systems are designed to overcome the problem of aggregating data across diverse trading units. The design of an information system depends on the risk measurement methodology chosen. Both a risk management information system and a risk measurement methodology have to be well incorporated. It's a problem that should be addressed, that involves the accuracy of the resulting measures of risk/s and the way/method of computing them. Technical progress helped over time to implement and use more and more accurately various methodologies. The current and likely future improvements in risk management information systems make feasible new ways of collecting aggregate data on organizations' risk-taking activities.

Over the last years an increasing number and wider variety of risk, interface with organizations. Complicated financial risks which have grown in importance, constant change of business environments via globalization of enterprises, risks that have to do with insufficient understanding of future dynamics of a business etc. All the above and more drove organizations to quickly recognize the importance of managing all risks and their interactions, not just the familiar risks or the ones that are easy to quantify, but even risks that appear to be insignificant, but due to their interaction with others might cause greater damage.

Managers and stakeholders in general expect a Risk Management Information System (RIMS or RMIS) to provide them with the data they need to meet the above business needs. Most managers use RIMS to:

- Calculate Value at risk
- Perform scenario analyses
- Measure current and future exposure to each counterparty
- Do all three above at varying levels, across various grouping of risks and across product types

1.2 AIM

The aim of this dissertation is to study the area of risk management. In doing so, a model/tool will be developed and presented. The proposed model addresses the problems rise from managing Information Systems (IS). A Risk Tracking & Management tool will be developed, that will incorporate and materialize the model described helping to materialize various procedures of an organization and meet above goals. Challenges such as rapid alterations in system life cycle, is also addressed.

The model will use inputs, processes and outputs that make risk management more efficiently applied and being a useful tool for senior management of an organization. Additionally the model will also based on installation off control measures, on a balance between the cost of controls and the need to reduce or eliminate possible threats

The tools that will be implemented for controlling/m the risk and tracking the defects of a system will be a Risk Management and a Ticketing tool which will to perform below core functions:

1. Store, track and identify risks and potential defects
2. Calculate the possible magnitude of risks
3. Provide an effective tool in recording, maintaining, tracking and controlling identified risks in co-operation with the system that supports.
4. Quick access to full history of incidents

5. Prioritization/Criticality tracking
6. Easy access to proposed plans in order to reduce or avoid risks

1.3 OBJECTIVE

The objectives of the current dissertation are to identify, manage potential risks to a system and take preventive steps to handle the uncertainties. For this purpose a tool that is capable of:

1. On-time recognize risks that may attend attention depending on their priority
2. Minimize the chances of a risk to re-occur by tracking original risk and storing the actions taken to handle it
3. Organize risks by prioritizing them
4. Reporting

Once this knowledge is prioritized, stored and handled a manager or owner of this tool is capable of developing strategies and plans for risk management for the system or project this tool is applied to.

2. LITERATURE REVIEW

Abstract

Risk Management is increasingly approached from modern organizations and companies as an integral part of good management. Risk management should take a balanced view of decision problems by gathering, measuring and correctly analyzing all possible risks and rewards. All risk analyses are based on the same general principles, such as generation of alternatives, quantification of uncertainties, prepare for consequences and how these are going to be reduced. Unsurprisingly the factors which deserve the most attention vary from problem-to-problem. Therefore this dissertation is focused on how these uncertainties can be handled via risk management tools and procedures.

This paper identifies the risk management process which consists of several discrete steps. These processes are further analyzed in depth in order to acquire in depth knowledge of the theoretical background. Later on this background will enable me to introduce two different tools that potentially can assist professionals to apply risk management for a "live" Information System or a project. This study will also apply action research as the core methodology to approach the whole effort. The outcome/objective of this dissertation is to provide useful reference and an alternative proposition to managers to handle risk in the systems and organizations they have the authority.

2.1 INTRODUCTION

Information systems management is defined as a subset of the overall internal controls of a business covering the application of people, documents, technologies and procedures by the relative stakeholders of an organization/company to solve or control business problems such as costing, services, risk identification or a business strategy. Management information systems are distinct from regular information systems in that they are used to analyze other information systems applied in operational activities in the organization.

At the start, in businesses and other organizations, internal reporting was made manually and only periodically, as a by-product of the accounting system and with some additional statistic(s), and gave limited and delayed information on management performance. Previously, data had to be separated individually by the people as per the requirement and necessity of the organization. Later, data was distinguished from information, and instead of the collection of mass and mostly "useless" data was decided, that it was crucial to identify the important points (information) and then follow procedures for deciding what is important and what is not to be processed or stored.

In general systems/projects and enterprises are exposed to risks. The following steps show how these risks should be treated in order Risk Management to be successful. An information system consists of people, equipment, and procedures to gather, sort, analyze, evaluate, and distribute needed, timely, and accurate information to marketing decision makers.

This process consists of seven discrete steps (F**igure 2.1**) :

- Establish Context
- Identify Risks
- Analyze/Quantify Risks
- Integrate Risks
- Assess/Prioritize Risks
- Treat/Resolve/Exploit Risks and
- Monitor & Review

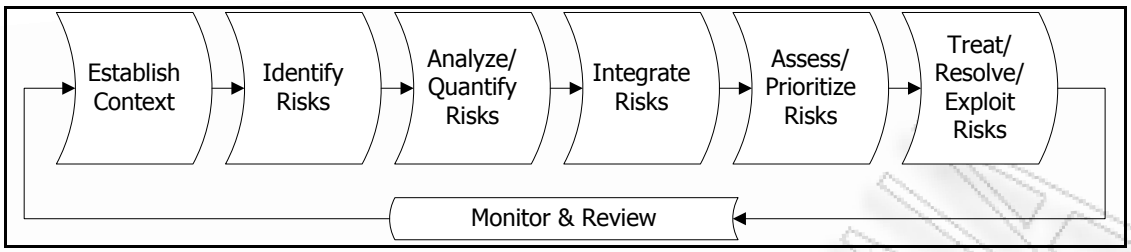


Figure 2.1 – The process for successful Risk Management

2.2 THE PROCESS

2.2.1 Establish Context

This step includes External, Internal and Risk Management Contexts

- The External Context starts with a definition of the relationship of the system/enterprise with its environment, including identification of the system's strengths, weaknesses, opportunities and threats (SWOT analysis). The context setting also identifies the various stakeholders (eg. Employees, customers, environment, community), as well as the communication policies with these stakeholders
- The Internal Context starts with an understanding of the overall objectiveness and it's key performance indicators. It also includes the organization's oversight and governance structure.
- The Risk Management Context identifies the risk categories of relevance to the system/enterprise and the degree of coordination throughout the organization, including the adoption of common risk metrics.

2.2.2 Identify Risks

After establishing the context, the next step in the process of managing risk is to identify potential risks. Risks are about events that, when triggered cause problems. Hence, risk identification can start with the source of problems or with the

problem itself. The Risk Identification process involves identifying all the risks that might impact the project/system, documenting them, and documenting their characteristics. Risk Identification is an iterative process that continually builds on itself. Progressing through the project, more risks may present themselves. Once a potential new risk is identified or discovered, analysis will determine if a response plan is needed. The risk management cycle starts with Risk Identification and progresses through the remaining risk processes to determine what to do about them. Several groups of individuals involved in the system can help identify risks, including project team members, stakeholders, subject matter experts, users and anyone else who you think may help in the process. In the first round of Risk Identification, only the project team and subject matter experts can be included. Then the rest of the stakeholders or risk management team can be involved during the second round of identification. The chosen method of identifying risks may depend on culture, industry practice and compliance. The identification methods are formed by templates or the development of templates for identifying source, problem or event

Risk Identification Inputs

The inputs to the Risk Identification process are:

- Enterprise environmental factors
- Organizational process assets
- Project scope statement
- Risk management plan
- Project management plan

The enterprise environmental factors input concerns things from outside the project that may help determine or influence project outcomes. The project scope statement contains a list of project assumptions. It's imperative during the risk-planning stages of a project and throughout the work of the project to revisit and revalidate various project assumptions. Paying particular attention to the roles and responsibilities section of the risk management plan, the budget and schedule for risk activities, and are crucial for the final success. The project management plan contains several other management plans (like schedule, quality, and cost) that can be helpful sources also when identifying risks.

Tools and Techniques Used to Identify Risk

The Risk Identification process is undertaken using five tools and techniques:

- Documentation reviews
- Information-gathering techniques
- Checklist analysis
- Assumptions analysis
- Diagramming techniques

Documentation reviews involve reviewing project plans, assumptions, and historical information from a total project perspective as well as at the individual deliverables or activities level. This review helps the project team identify risks associated with the project objectives. Key factor to the quality of the plans and the consistency between plans is the content completion or not. An exceptionally documented schedule is great, but if the budget isn't as well documented, potential risks may rise.

Information Gathering

Information gathering encompasses several techniques, including brainstorming, the Delphi technique, interviewing, root cause identification, and strength and weakness analysis. The goal of these techniques is to end up with a comprehensive list of risks at the end of the meeting.

- Brainstorming
- Delphi Technique
- Nominal Group Techniques
- Interviewing
- Root Cause Identification
- Strengths, Weaknesses, Opportunities, and Threats
- Checklist Analysis
- Assumption Analysis
- Diagram Techniques

Brainstorming

Brainstorming is probably the most often used technique of the Risk Identification process. Brainstorming involves getting subject matter experts, team members, risk management team members, and anyone else who might benefit the process and asking them to start identifying possible risk events. The trick here is that one person's idea might spawn another idea and so on, so that by the end of the session you've identified all the possible risks. The facilitator could start the group off by going through the categories of risks to get everyone thinking in the right direction.

Delphi Technique

The Delphi technique is a lot like brainstorming, only the people participating in the meeting don't necessarily know each other. In fact, the people participating in this technique don't all have to be located in the same place and can participate anonymously. What you do is assemble experts, from both inside and outside the company, and ask via a questionnaire to identify potential risks. They in turn send their responses back. All the responses are organized by content and sent back to the Delphi members for further input, additions, or comments. The participants then send their comments back one more time, and a final list of risks is compiled by the facilitator. The Delphi technique is a great tool that allows consensus to be reached very quickly. It also helps prevent one person from unduly influencing the others in the group and thus prevents bias in the outcome because the participants are usually anonymous and don't necessarily know how others in the group responded.

Nominal Group Technique

Another technique that is similar to the Delphi technique is the nominal group technique. It isn't a named tool and technique of this process, but it is a technique you might find useful and that you may find on the exam. This technique requires the participants to be together in the same room. Each participant has paper and pencil in front of them, and they are asked to write down what risks they think the project faces. Using sticky-backed notes is a good way to do this. Each piece of paper should contain only one risk. The papers are given to the facilitator, who sticks them up to the wall or a white board. The panel is then asked to review all the risks

posted on the board; rank them and prioritize them, in writing; and submit the ranking to the facilitator. Once this is done, you should have a complete list of risks.

Interviewing

Interviews are question-and-answer sessions held with others, including other project managers, subject matter experts, stakeholders, customers, the management team, project team members, and users. People involved provide possible risks based on their past experiences with similar projects. This technique involves interviewing people with previous experience on similar projects or those with specialized knowledge or industry expertise. Asking them about any risks that they've experienced or that they think may happen on the project.

Root Cause Identification

Looking at the symptoms and not at the problem is a common problem. That's the idea here. Root cause identification involves digging deeper than the risk itself and looking at what the cause of the risk is. This helps define the risk more clearly and it also helps later when it's time to develop the response plan for the risk.

Strengths, Weaknesses, Opportunities, and Threats (SWOT)

Strengths, weaknesses, opportunities, and threats (also known as SWOT analysis) is an analysis technique that examines through each of these viewpoints (SWOT) the project itself, project processes, resources, the organization, and so on. It also helps broaden your perspective of where to look for risks.

Checklist Analysis

Checklists used during the Risk Identification process are usually developed based on historical information and previous project team experience. Identical projects that are similar to nature can benefit from this approach, by compiling a list of risks. This can easily converted to a checklist that allows identifying risks on future projects quickly and easily. Relying solely on checklists for Risk Identification might cause missing important risks.

Assumptions Analysis

Assumptions analysis is a matter of validating the assumptions identified and documented during the course of the project Planning processes. Assumptions should be accurate, complete, and consistent. Assumptions should be examined for these qualities and are also used as a starting point to further identify risks.

The important thing to note about the project assumptions is that all assumptions are tested against two factors:

- The strength of the assumption or the validity of the assumption
- The consequences that may impact the project if the assumption turns out to be false

All assumptions that turn out to be false should be evaluated and scored just as risks.

Diagramming Techniques

There are three types of diagramming techniques used in Risk Identification:

- cause-and-effect
- system or process flowcharts and
- influence diagrams.

- **Cause-and-effect** diagrams show the relationship between the effects of problems and their causes. This diagram depicts every potential cause and sub-cause of a problem and the effect that each proposed solution will have on the problem. Figure 2.2 shows an example cause-and-effect diagram.

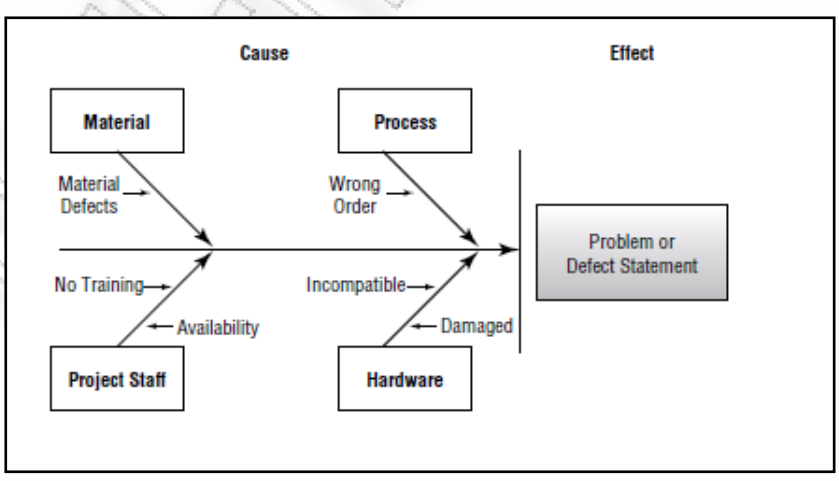


Figure 2.2 Cause-and-effect Diagram (Source: PMP Professional Study Guide, Kim Heldman, Wiley Publishing, 2005)

- **The system or process flowchart** shows the logical steps needed to accomplish an objective, how the elements of a system relate to each other, and what actions cause what responses. This flowchart is probably the one you're most familiar with. It's usually constructed with rectangles and parallelograms that step through a logical sequence and allow for "yes" and "no" branches.

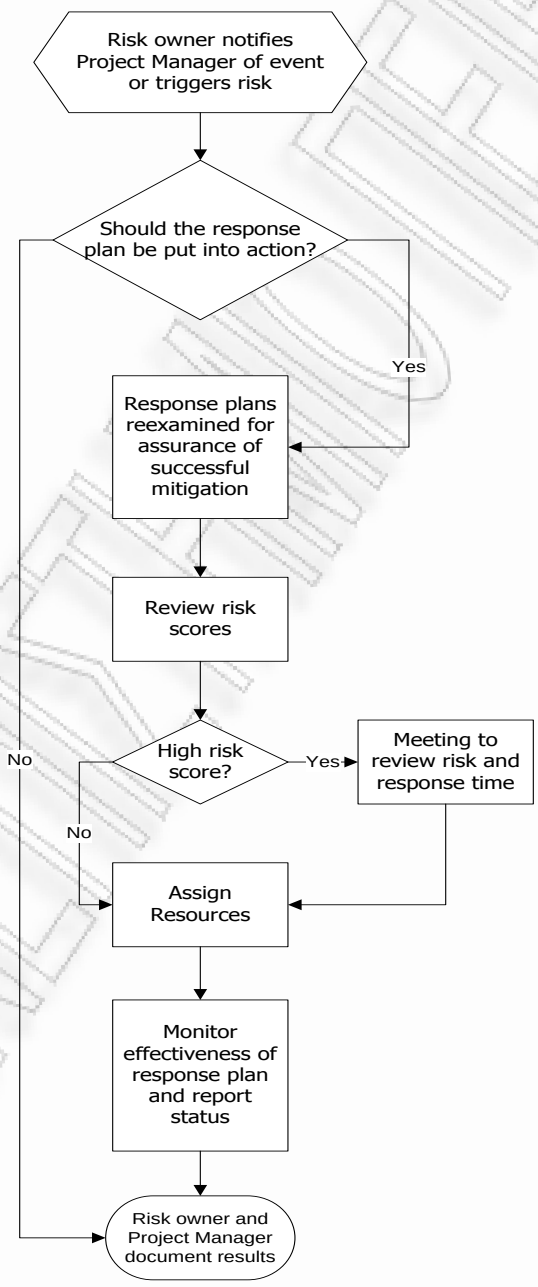


Figure 2.3 Process Flowchart

- A third diagramming technique used during Risk Identification is called **influence diagrams**. Influence diagrams typically show the casual influences among project/system variables, the timing or time ordering of events, and the relationships among other project variables and their outcomes. Simply put, they visually depict risks (or decisions), uncertainties or impacts, and how they influence each other. The weather is a variable that could impact delivery time and delivery time is a variable that can impact when revenues will occur.

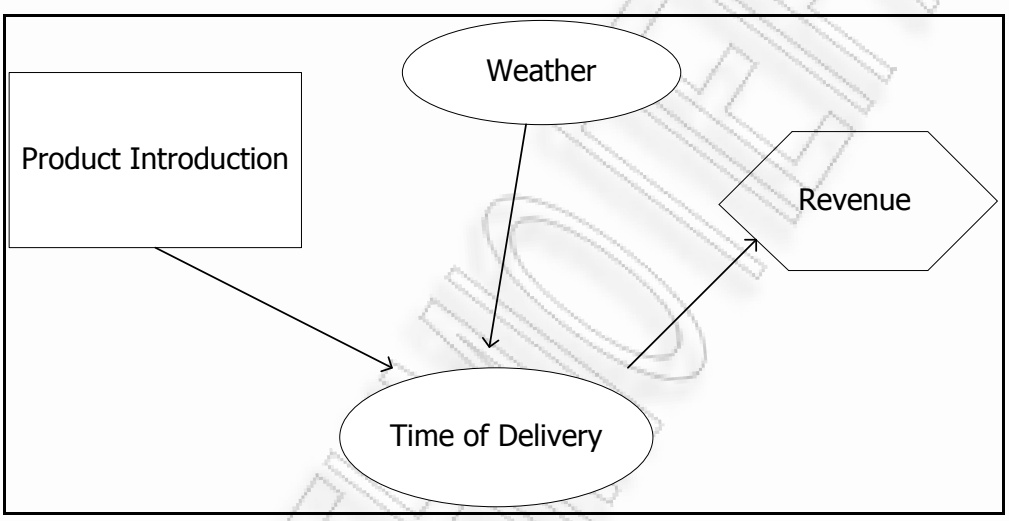


Figure 2.4 Influence Diagram

Risk Identification Outputs

The output of the Risk Identification process is the risk register. Everything you've done in this process up to this point will get documented here. The risk register contains the following elements:

- List of identified risks
- List of potential responses
- Root causes of risks
- Updated risk categories

We'll take a look at each one next. Understand that all risks should be documented, tracked, reviewed, and managed throughout the project.

List of Identified Risks

Risks are all the potential events and their subsequent consequences that could occur as identified by you and others during this process. You might want to consider logging your risks in a risk database or tracking system to organize them and keep a close eye on their status. This can easily be done in spreadsheet format or whatever method you choose. List the risks and assign each risk a tracking number. This gives you a means to track the risks, their occurrence, and the responses implemented.

List of Potential Responses

Potential responses to risk may be identified at the same time you're identifying the risks. Sometimes just identifying the risk will tell you the appropriate response. Document those responses here. You'll refer to them again in the Risk Response Planning process a little later in this chapter.

Root Causes of Risk

We talked about this one before. When you're identifying risks, be sure you go further than thatand try to identify the cause. Document those causes you've identified here in the risk register.

Updated Risk Categories

As a result of identifying and documenting your risks, you may well discover some categories of risk need changed or updated to reflect the risks for the current project. This might also include updating the RBS. It's a good practice to update those categories now, while you're in the midst of the process. Your next project will be all the better for it.

2.2.3 Analyze/Quantify Risks

This step involves calibrating and whether possible, creating probability distributions of outcomes for each material risk. This step provides necessary input for later steps, such as integrating and prioritizing risks. Analysis techniques range

along a spectrum from qualitative to quantitative, with sensitivity analysis, scenario analysis, and/or simulation analysis applied where appropriate.

Risk Categories

Risk categories are a way to systematically identify risks and provide a foundation for understanding. When determining and identifying risks, the use of risk categories helps improve the process by giving everyone involved a common language or basis for describing risk. Risk categories should be identified during this process and documented in the risk management plan. You will use these categories during the next process, Risk Identification, to help identify risks. The following list includes some of the categories you might identify during this process (or modify based on previous project information):

- Technical, quality, or performance risks
- Project management risks
- Organizational risks
- External risks

There are a couple of ways you can go about describing categories of risk. One way is simply listing them. You could, and should, review prior projects for risk categories and then tailor them for this project. You could also construct a **Risk Breakdown Structure** (**RBS**), which lists the categories and subcategories, figure 2.5.

Risk categories may reflect the type of industry or application area in which the project exists. For example, Information Technology projects will likely have many risks that fall into the technical category, whereas construction projects may be more subject to risks that fall into the external risks category. The categories do not have to be industry specific, however. Keep in mind that project management, for example, is a risk on every project in every industry. A description of each of the categories is shown next:

Technical/quality/performance risks

Technical, quality, or performance risks include risks associated with unproven technology, complex technology, or changes to technology anticipated during the course of the project. Performance risks may include unrealistic

performance goals. Perhaps one of the project deliverables concerns a component manufactured to specific performance standards that have never been achieved. That's a performance risk.

Project management risks

The project management risk category includes improper schedule and resource planning, poor project planning, and improper or poor project management disciplines or methodologies.

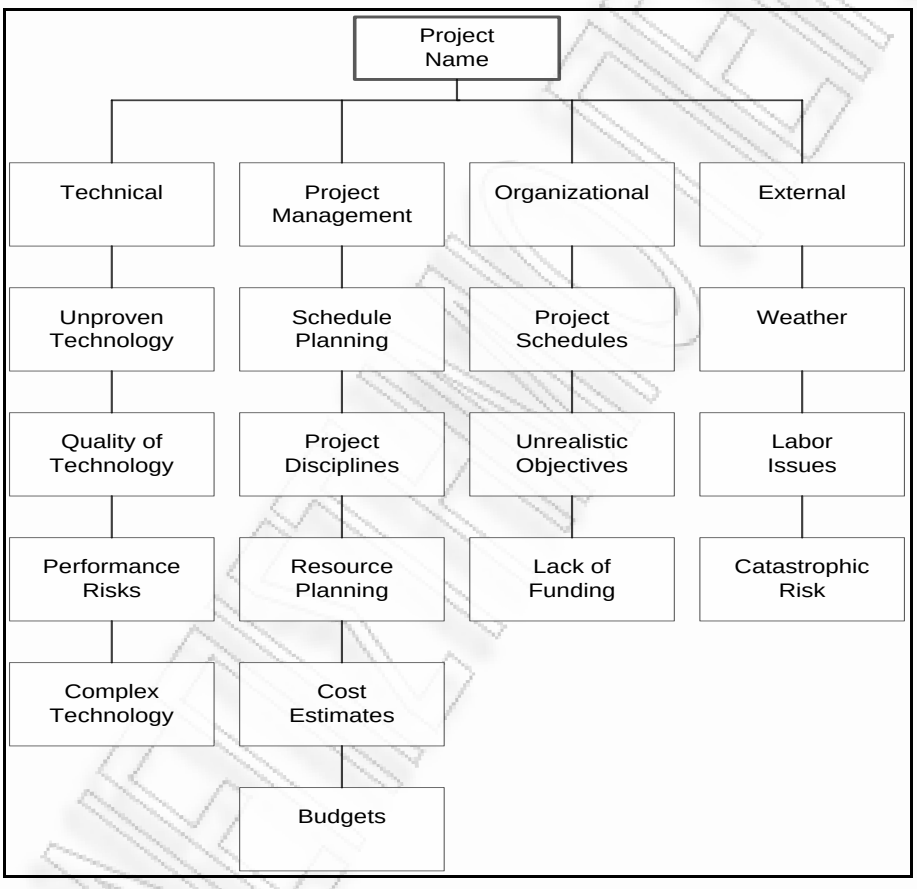


Figure 2.5 Risk Breakdown structure

Organizational risks

The organizational risk category can include resource conflicts due to multiple projects occurring at the same time in the organization; scope, time, and cost objectives that are unrealistic given the organization's resources or structure; and lack of funding for the project or diverting funds from this project to other projects.

External risks

The external risk category includes those things external to the project, such as new laws or regulations, labor issues, weather, changes in ownership, and foreign policy for projects performed in other countries. Catastrophic risks — known as force majeure — are usually outside the scope of risk management planning and instead require disaster recovery techniques. Force majeure includes events like earthquakes, meteorites, volcanoes, floods, civil unrest, terrorism, and so on.

Defining Probability and Impact

When you're writing the risk management plan, you'll want to document how probability and impact will be defined on the project. Probability describes the potential for the risk event occurring, while impact describes the impact, or consequences, the project will experience if the risk event occurs. This definition can be very sophisticated or simple. For example, you may use numeric values to define probability and impact or simply assign a high-medium-low rating to each risk. What's important to note now is that you don't use these probability and impact definitions here. You use these definitions later in the Qualitative Risk Analysis process. But you should define and document them here in the risk management plan.

a) **Qualitative** Techniques
b) **Quantitative** Techniques

A) ANALYZE USING **QUALITATIVE** TECHNIQUES

The Qualitative Risk Analysis process involves determining what impact the identified risks will have on the project objectives and the probability they'll occur. It also ranks the risks in priority order according to their effect on the project objectives. This helps the team determine if Quantitative Risk Analysis should be performed or if you can skip right to developing response plans. The Qualitative Risk Analysis process also considers risk tolerance levels, especially as they relate to the project constraints (scope, time, cost, and quality) and the time frames of the potential risk events.

The Qualitative Risk Analysis process should be performed throughout the system. This process is the most often when prioritizing project risk because it's fast,

relatively easy to perform, and cost effective. Two techniques can be used during this process to help correct biases that can occur in the data you've gathered: definitions of probability and impact and expert interviewing.

The Qualitative Risk Analysis process's tools and techniques are primarily concerned with discovering the probability of a risk event and determining the impact (or consequences) the risk will have if it does occur. The output of this process is a risk register update where you'll document the prioritized risks you've scored using these tools and techniques. All of the information you gather regarding risks and probability needs to be as accurate as possible. It's also important that you gather unbiased information so that you don't unintentionally overlook risks with great potential or consequences. The purpose of this process is to determine risk event probability and risk impact. You'll use the tools and techniques of this process to establish risk scores, which is a way of categorizing the probability and risk impact. The Qualitative Risk Analysis process includes the following tools and techniques:

- Risk probability and impact assessment
- Probability and impact matrix
- Risk data quality assessment
- Risk categorization
- Risk urgency assessment

We'll look at each of these tools and techniques next.

Risk Probability and Impact Assessment

This tool and technique assess the probability that the risk events you've identified will occur and it determines the effect their impacts have on the project objectives, including time, scope, quality, and cost. Analyzing risks in this way allows you to determine which risks require the most aggressive management. When determining probabilities and impacts, you'll refer back to the risk management plan element called "definitions of risk probability and impact."

Probability

Probability is the likelihood that an event will occur. The classic example is flipping a coin. There is a .50 probability of getting heads and a .50 probability of getting tails on the flip. One thing to note is that the probability that an event will occur plus the

probability that the event will not occur always equals 1.0. In this coin-flipping example, there is a .50 chance that you'll get heads on the flip. There is also, therefore, a .50 chance you will not get heads on the flip. The two responses added together equal 1.0. Probability is expressed as a number from 0.0—which means there is no probability of the event occurring—to 1.0—which means there is 100 percent certainty the risk will occur. Determining risk probability can be difficult because it's most commonly accomplished using expert judgment. In non-PMP terms, this means you're guessing (or asking other experts to guess) at the probability a risk event will occur. Granted, you're basing your guess on past experiences with similar projects or risk events, but no two risk events (or projects) are ever the same. It's best to fully develop the criteria for determining probability and get as many experts involved as you can. Carefully weigh their responses to come up with the best probability values possible.

Impact

Impact is the amount of pain (or the amount of gain) the risk event poses to the project. The risk impact scale can be a relative scale that assigns values such as high-medium-low (or some combination of these) or a numeric scale known as a cardinal scale. Cardinal scale values are actual numeric values assigned to the risk impact. Cardinal scales are expressed as values from 0.0 to 1.0 and may be stated in equal (linear) or unequal (nonlinear) increments. Figure 3 shows a typical risk impact scale for cost, time, and quality objectives based on a high-high to low-low scale. You'll notice that each of the "high-medium-low" value combinations on this impact scale have been assigned a cardinal value. We'll use these in the next section when we talk about the probability and impact matrix.

Objectives	Low-Low	Low	Medium	High	Critical
Cost	No significant Impat	Less than 6% increase	7% - 12% increase	13% - 18% increase	More than 18% increase
Time	No significant impact	Less than 6% increase	7% - 12% increase	13% - 18% increase	More than 18% increase
Quality	No significant impact	Few components impacted	Significant impact requiring customer approval to proceed	Unacceptable Quality	Product not usable

Figure 2.6 Example of Risk Impact Scale

Assessing Probability and Impact

The idea behind both probability and impact values is to develop predefined measurements that describe what value to place on a risk event. If these scales have not yet been determined, develop them as early on in the project as possible. You can use any of the techniques we talked about earlier in the section "Tools and Techniques Used to Identify Risk," such as brainstorming or the Delphi technique, to come up with the values for probability and impact. During the Qualitative Risk Analysis process, you'll determine and assess probability and impact for every risk identified during the Risk Identification process. You could interview or hold meetings with project team members, subject matter experts, stakeholder or others to help assess these factors. During this process, you should document not only the probability and impact but also the assumptions your team members used to arrive at these determinations. The next technique—probability and impact matrix—take the probability and impact values one step further by assigning an overall risk score.

Probability and Impact Matrix

A probability and impact matrix defines the combination of probability and impact that helps you determine which risks need detailed risk response plans. For example, a risk with a high probability of occurring and a high impact will likely need a response plan. This matrix is typically defined by the organization, during the planning meeting and analysis. You'll use this matrix in the Qualitative Risk Analysis process, and we'll talk more in depth in the section about this technique. Again, you want to define (or modify) the probability and impact matrix in the risk management plan.

A probability and impact matrix assigns an overall risk rating to each of the project's identified risks. The combination of probability and impact results in a classification usually expressed as high, medium, or low. A ranking system is also essential to be defined in this point. The values should be ranked-ordered from high to low. In practice, ordinal values may also include ranking by position. In other words, the risks are listed in rank order as the first, the second, the third, and so on.

An ***example*** to further understand the above will follow. You have identified a risk event that could impact project costs, and your experts believe costs could increase by as much as 9 percent. According to the risk impact rating matrix in figure 2.6

above, this risk carries a medium impact, with a value of 0.40. Keep this number because you're going to plug it into the probability impact matrix, along with the probability value, to determine an overall risk value next. You'll remember from our discussion previously that probability values should be assigned numbers from 0.0 to 1.0. The team has determined that there is a 0.2 probability of this risk event occurring. The risk impact scale shows a medium or 0.4 impact should the event occur. To determine whether the combination of the probability and impact of this risk is high, medium, or low, you'll need to check the probability impact matrix. Figure 2.7 below shows a sample probability and impact matrix. First look at the probability column. Your risk event has a probability of .2. Now follow that row until you find the impact score of .04. According to your probability and impact matrix values, this risk carries a score of 0.08 and falls in the low threshold, so this risk is assigned a low (or green condition) value. The values assigned to the risks determine how Risk Response Planning is carried out for the risks later during the risk-planning processes. Obviously, risks with high probability and high impact are going to need further analysis and formal responses. Remember that the values for this matrix (and the probability and impact scales discussed earlier) are determined prior to the start of this process and documented in the Risk Management Plan. Also keep in mind that probability and impact do not have to be assigned the same values as I've done here. You may use 0.8, 0.6, 0.4, and 0.2 for probability, for example, and assign .05, 0.1, .03, 0.5, and 0.7 for impact scales, as shown in figure 2.8 below.

	Impact Values				
Probability	**0.05**	**0.2**	**0.4**	**0.6**	**0.8**
0.8	0.04	0.16	0.32	0.48	0.64
0.6	0.03	0.12	0.24	0.36	0.48
0.4	0.02	0.08	0.16	0.24	0.32
0.2	0.1	0.04	0.08	0.12	0.16

Figure 2.7 Probability and Impact Matrix (example)

	Impact Scores				
Probability	**0.05**	**0.2**	**0.4**	**0.6**	**0.8**
0.8	0.04	0.16	0.32	0.48	0.64
0.6	0.03	0.12	0.24	0.36	0.48
0.4	0.02	0.08	0.16	0.24	0.32
0.2	0.1	0.04	0.08	0.12	0.16

Figure 2.8 Probability and Impact Scores

B) ANALYZE USING **QUANTITATIVE** TECHNIQUES

The Quantitative Risk Analysis process evaluates the impacts of risk prioritized during the Qualitative Risk Analysis process and quantifies risk exposure for the project by assigning numeric probabilities to each risk and their impacts on project objectives. This quantitative approach is accomplished using techniques like Monte Carlo simulation and decision tree analysis. The purpose of this process is to perform the following:

- Quantify the project's possible outcomes and probabilities.
- Determine the probability of achieving the project objectives.
- Identify risks that need the most attention by quantifying their contribution to overall project risk.
- Identify realistic and achievable schedule, cost, or scope targets.
- Determine the best project management decisions possible when outcomes are uncertain.

Quantitative Risk Analysis—like Qualitative Risk Analysis—examines each risk and its potential impact on the project objectives. You may choose to use both of these processes to assess risk or only one of them, depending on the complexity of the project and the organizational policy regarding risk planning. The Quantitative Risk Analysis process can follow either the Risk Identification process or the Qualitative Risk Analysis process. If you do use this process, be certain to repeat it every time the Risk Response Planning process is performed and as part of the Risk Monitoring

and Control process. We have already covered many of the inputs to the Quantitative Risk Analysis process in previous sections of this chapter. They are as follows:

- Organizational process assets
- Project scope statement
- Risk management plan
- Risk register
- Project management plan

The elements of the project management plan you'll want to pay close attention to as an input to this process are the project schedule management plan and the project cost management plan.

Quantitative Risk Analysis and Modeling Techniques

There are four techniques encompassed in this tool and technique: sensitivity analysis, expected monetary value analysis, decision tree analysis, and modeling and simulation. Let's take a brief look at each of them.

Sensitivity Analysis

Sensitivity analysis is a quantitative method of analyzing the potential impact of risk events on the project and determining which risk event (or events) has the greatest potential for impact by examining all the uncertain elements at their baseline values. One of the ways sensitivity analysis data is displayed is a tornado diagram. Figure 2.9 shows a sample tornado diagram.

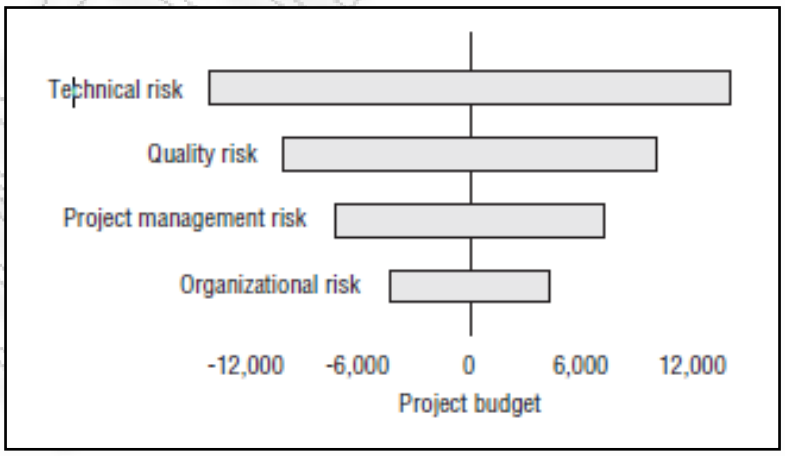


Figure 2.9 Tornado Diagram (Source: http://leadershipchamps.wordpress.com/2009/06/14/find-how-sensitive-is-your-project-against-variables-tornado-diagram/)

You can see by the arrangement of horizontal bars (each representing a sensitivity variable) how the diagram gets its name. The idea is that each sensitivity bar displays the low and high value possible for that element. (It's beyond the scope of this book to explain how these values are determined. The questions you may encounter on the exam are focused on the context of this type of analysis.) The variables with the greatest effect on the project are displayed at the top of the graph and decrease in impact as you progress down through the graph. This gives you a quick overview of how much the project can be affected by uncertainty in the various elements. It also allows you to see at a glance which risks may have the biggest impacts on the project and will require carefully crafted, detailed response plans. You can use tornado diagrams to determine sensitivity in cost, time, and quality objectives or for risks you've identified during this process. Sensitivity analysis can also be used to determine stakeholder risk tolerance levels.

Expected Monetary Value (EMV) Analysis

Expected monetary value (EMV) analysis is a statistical technique that calculates the average, anticipated impact of the decision. EMV is calculated by multiplying the probability of the risk times its impact and then adding them together. EMV is used in conjunction with the decision tree analysis technique, which is covered next. I'll give you an example of the EMV formula in the next section. Positive results generally mean the risks you're assessing pose opportunities to the project while negative results generally indicate a threat to the project.

Decision Tree Analysis

Unfortunately, this isn't a tree outside your office door that produces "Yes" and "No" leaves that you can pick to help you make a decision. Decision trees are diagrams that show the sequence of interrelated decisions and the expected results of choosing one alternative over the other. Typically, more than one choice or option is available when you're faced with a decision or, in this case, potential outcomes from a risk event. The available choices are depicted in a tree form starting at the left with the risk decision branching out to the right with possible outcomes. Decision

trees are usually used for risk events associated with time or cost. Figure 2.10 shows a sample decision tree using expected monetary value (EMV) as one of its inputs.
The expected monetary value of the decision is a result of the probability of the risk event multiplied by the impact and then adding their results. The squares in this figure represent decisions to be made, and the circles represent the points where risk events may occur.

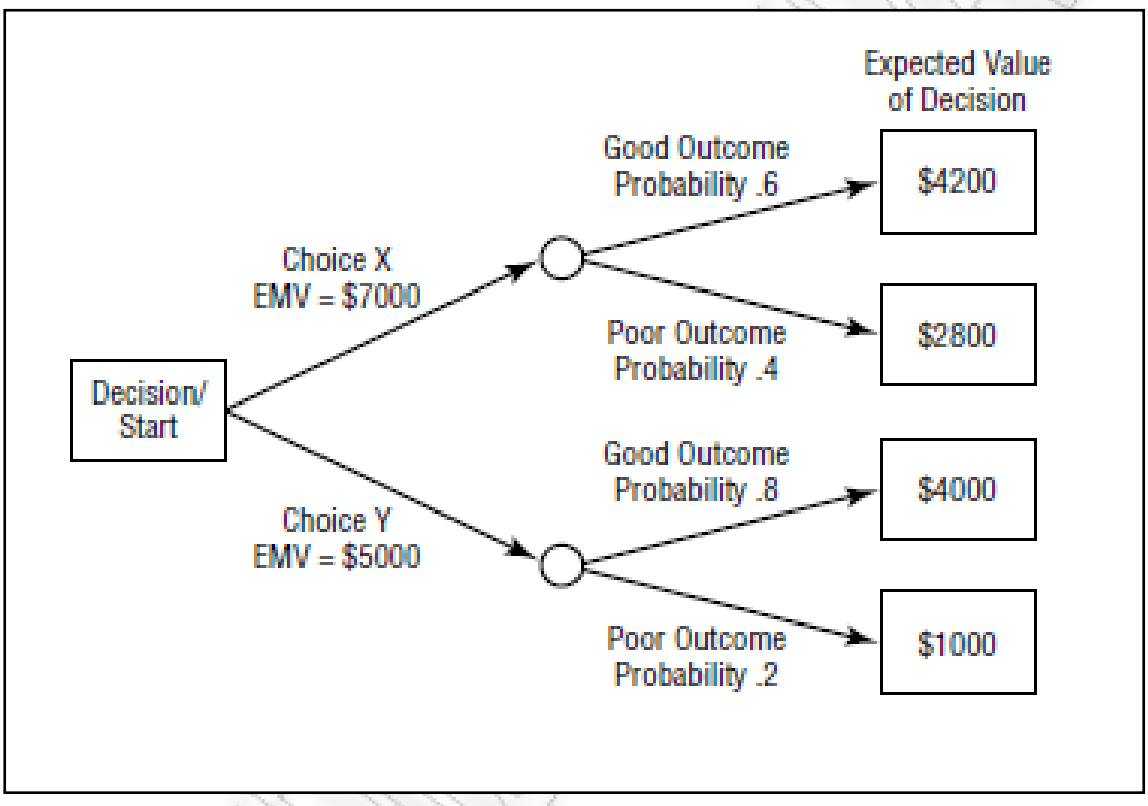


Figure 2.10 Decision Tree (Source: Induction Material from ISEB Certification, Software Testing Foundation)

The decision with an expected value of $7,000 is the correct decision to make because the resulting outcome has the greatest value.

Modeling and Simulation

Modeling and simulation techniques are often used for schedule risk analysis and cost analysis. For example, modeling allows you to translate the potential risks at specific points in the project into their impacts so you can determine how the project objectives are affected. Simulation techniques compute the project model using various inputs such as cost or schedule duration, to determine a probability

distribution for the variable chosen. (Cost risks typically use either a work breakdown structure or cost breakdown structure as the input variable. Schedule risks always use the precedence diagramming method as the input variable. We we'll cover schedule diagramming methods in Chapter 7.) If you used simulation techniques to determine project cost and use the cost of the project elements as the input variable, a probability distribution for the total cost of the project would be produced after running the simulation numerous times. Modeling and simulation techniques examine the identified risks and their potential impacts to the project objectives from the perspective of the whole project. Monte Carlo analysis is an example of a simulation technique. Monte Carlo analysis is performed many times, typically using cost or schedule variables. Every time the analysis is performed, the values for the variable are changed using a probability distribution for each variable. Monte Carlo analysis can also be used during the Schedule Development process.

Quantitative Risk Analysis Outputs

The output of the Quantitative Risk Analysis process is—I'll bet you can guess—risk register updates. As with the Qualitative Risk Analysis process, there are new elements you'll record in the risk register:

Probabilistic analysis of the project Probabilistic analysis of the project is the forecasted results

of the project schedule and costs as determined by the outcomes of risk analysis. These results include projected completion dates and costs, along with a confidence level associated with each. According to

Confidence levels can also be used to describe the level of confidence placed on the outcome of the forecasted results. For example, suppose the projected schedule completion date is July 12 and the confidence level is .85. This says that you believe the project will finish on or before July 12 and that you have an 85 percent level of confidence that this date is accurate.

Probability of achieving the cost and time objectives Using the tools and techniques of Quantitative Risk Analysis allows you to assign a probability of achieving the cost and time objectives

of the project. This output documents those probabilities and as such requires a thorough understanding of the current project objectives and knowledge of the risks.

Prioritized lists of quantified risks The prioritized list in this process is similar to the list produced during the Qualitative Risk Analysis process. The list of risks includes those that present the greatest risk or threat to the project and their impacts. It also lists those risks that present the greatest opportunities to the project. This list should also indicate which risks are most likely to impact the critical path and those that have the largest cost contingency.

Trends in Quantitative Risk Analysis results Trends in Quantitative Risk Analysis will likely appear as you repeat the risk analysis processes. This information is useful as you progress, making those risks with the greatest threat to the project more evident, which allows you the opportunity to perform further analysis or go on to develop risk response plans.

2.2.4 Integrates Risks

This step involves aggregating all risk distributions, reflecting possible interconnections among systems and possible risks that rise when attempting to integrate them. Once information passes from one system to another there is always a risk that should be calculated and taken into account. Figure below shows four possible factors that lead to increased quality risk for a system.

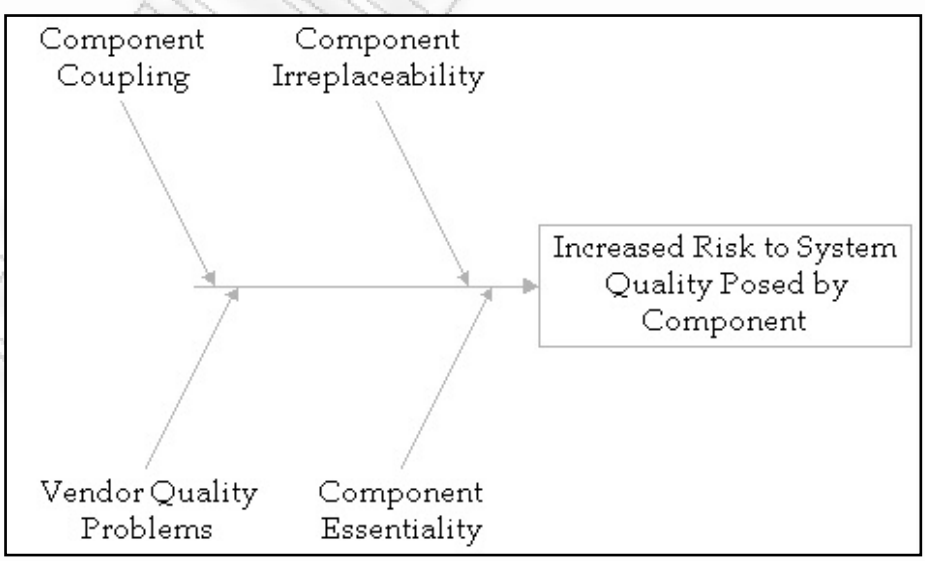


Figure 2.11 - Possible factors that lead to increased quality risk for a system

One factor that increases risk from coupling two or more systems and creates a strong interaction with between them is when a component fails. For example, suppose the customer table on the Web application database becomes locked and inaccessible under normal load. In such a case, most of the other components of the system, being unable to access customer information, will also fail. The database is strongly coupled to the rest of the system.

Another factor that increases risks is irreplaceability, when there are few similar components available or the replacement is costly or requires a long-lead time. If such a component creates quality problems for the system, unfortunately organizations cannot get rid of them. For example, the database package chosen might be replaceable, provided that you don't do anything non-standard with it. However, the development organization will want to be paid for the custom-developed Web application, and, when the organization attempt to replace it, there are not off-the-shelf products probably available.

Yet another factor that increases risks is essentiality, where some key feature or features of the system will be unavailable if the component does not work properly. For example, suppose you planned to include a pop-up loan planner on the first page of your application to allow customers to evaluate some payment scenarios. Should that component not work, you can still deliver the major features of your application. A pop-up loan planner is not essential to your system. However, if the subsystem that accesses a credit bureau to check customer credit scores does not work, you cannot process loan applications, since checking credit scores is essential.

2.2.5 Assess/Prioritize Risks

Although prioritization can be also addressed in previous steps, in this distinct step the Project Manager or the owner of the project/system ha s a more clear/overall picture of a possible risk that might arise.

The scope of the risk is very important when prioritization is attempted in this point. All stakeholders should prioritize by high-to-low risk. Additionally apart from criticality value is also considered. It would be a waste of effort if the delivery of a high risk functionality has little value. A dual approach should always be followed. Product/System owners should not prioritize on a single parameter. While prioritizing, you should consider both value and risk. Initially high-risk, high-value functionality or issues should be attacked first and then progress to the low-risk, high-value functionality. High-Risk, low-Value functionality should be avoided until such a time that it is deemed a high-value.

Risks should be prioritized in any step of the process. If it is done effectively, all stakeholders can focus their time and effort on the most important risks.

2.2.6 Treat/Resolve/Exploit Risks

The Risk Response Planning process is the last process covered in this chapter. Risk Response Planning is a process of deciding what actions to take to reduce threats and take advantage of the opportunities discovered during the risk analysis processes. This process also includes assigning departments or individual staff members the responsibility of carrying out the risk response plans you'll outline in this process. These folks are known as risk owners. Plans for risks of low severity or insignificant impact is not efficient or a good use of the project team's time. Spend your time planning responses that are appropriate given the impact the risk itself poses (or the opportunity the risk presents) and don't spend more time, money, or energy to produce a response than the risk event itself would produce if it occurred. Several strategies are used in this process to reduce or control risk. It's important that you choose the right strategy for each risk so that the risk and its impacts are dealt with effectively. After deciding on which strategy to use, you'll develop an action plan to put this strategy into play should the risk event occur. You may also choose to designate a secondary or backup strategy.

RISK RESPONSE PLANNING TOOLS AND TECHNIQUES

There are four tools and techniques for the Risk Response Planning process and each one of
them involves a strategy. The tools and techniques are as follows:

- Strategies for negative risks or threats
- Strategies for positive risks or opportunities
- Strategies for both threats and opportunities
- Contingence response strategy

Strategies for Negative Risks or Threats

There are three strategies to deal with negative risks or threats to the project objectives. These refer to how to avoid, transfer, and mitigate a risk.

Avoid

To avoid a risk means you'll evade it altogether, eliminate the cause of the risk event, or change the project plan to protect the project objectives from the risk event. Let's say you're going to take a car trip from your home to a point 800 miles away. You know—because your friends who just took the same trip told you—that there is a long stretch of construction on one of the highways you're planning on using. To avoid the risk of delay, you plan the trip around the construction work and use another highway for that stretch of driving. In this way, you change your plans, avoid the risk of getting held up in construction traffic, and arrive at your destination on time. With risk avoidance, you essentially eradicate the risk by eliminating its cause. Here's another example: Suppose your project was kicked off without adequate scope definition and requirements gathering. You run a high probability of experiencing scope creep—ever-changing requirements—as the project progresses, thus impacting the project schedule. You can avoid this risk by adequately documenting the project scope and requirements during the Planning processes and taking steps to monitor and control changes to scope so it doesn't get out of hand. Risks that occur early in the project might easily be avoided by improving communications, refining requirements, assigning additional resources to project activities, refining the project scope to avoid risk events, and so on.

Transfer

The idea behind a risk transfer is to transfer the risk and the consequences of that risk to a third party. The risk hasn't gone away, but the responsibility for the management of that risk now rests with another party. Most companies aren't willing to take on someone else's risk without a little cash thrown in for good measure. This strategy will impact the project budget and should be included in the cost estimate exercises if you know you're going to use it. Transfer of risk can occur in many forms but is most effective when dealing with financial risks. Insurance is one form of risk transfer. You are probably familiar with how insurance works. Car insurance is a good example. You purchase car insurance so that if you come upon an obstacle in the road and there is no way to avoid hitting it, the cost to repair the damage to the car is paid by the insurance company. Okay, minus the deductible and all the calculations for the age of the car, the mileage, the color and make of the car, the weather conditions the day you were driving—but I digress.

Another method of risk transfer is contracting. Contracting transfers specific risks to the vendor, depending on the work required by the contract. The vendor accepts the responsibility for the cost of failure. Again, this doesn't come without a price. Contractors charge for their services, and depending on the type of contract you negotiate, the cost might be quite high. For example, in a fixed-price contract, which we'll talk more about in the Procurement Planning section, the vendor (or seller) increases the cost of the contract to compensate for the level of risk they're accepting. A cost reimbursable contract, however, leaves the majority of the risk with you, the buyer. This type of contract may reduce costs if there are project changes midway through the project.

Keep in mind that contracting isn't a cure-all. You might just be swapping one risk for another. For example, say you hire a driver to go with you on your road trip and their job is to do all the driving. If the driver becomes ill or in some way can't fulfill their obligation, you aren't going to get to your destination on time. You've placed the risks associated with the drive on the contract driver; however, you've taken on a risk of delay due to nonperformance, which means you've just swapped one risk for another. You'll have to weigh your options in cases like this and determine which

side of the risk coin your organization can more readily accept. Other forms of transference include warranties, guarantees, and performance bonds.

Mitigate

When you mitigate a risk, you attempt to reduce the probability of a risk event and its impacts to an acceptable level. This strategy is a lot like defensive driving. You see an obstacle in the road ahead, survey your options, and take the necessary steps to avoid the obstacle and proceed safely on your journey. Seeing the obstacle ahead (identifying risk) allows you to reduce the threat by planning ways around it or planning ways to reduce its impact if the risk does occur (mitigation strategies).

The purpose of mitigation is to reduce the probability that a risk will occur and reduce the impact of the risk to a level where you can accept the risk and its outcomes. It's easier to take actions early on that will reduce the probability of a risk or its consequences than it is to fix the damage once it's occurred. Some examples of risk mitigation include performing more tests, using less-complicated processes, creating prototypes, and choosing more reliable vendors.

Strategies for Positive Risk or Opportunities

There are three strategies for dealing with opportunities or positive risks on the project: exploit, share, and enhance.

Exploit

When you exploit a risk event, you're looking for opportunities for positive impacts. This is the strategy of choice when you've identified positive risks that you want to make certain will occur on the project. Examples of exploiting a risk include reducing the amount of time to complete the project by bringing on more qualified resources or by providing even better quality than originally planned.

Share

The share strategy is similar to transferring because you'll assign the risk to a third-party owner who is best able to bring about the opportunity the risk event presents. For example, perhaps what your organization does best is investing. However, it isn't so good at marketing. Forming a joint venture with a marketing firm to capitalize on a positive risk will make the most of the opportunities.

Enhance

The enhance strategy closely watches the probability and or impact of the risk event to assure that the organization realizes the benefits. This entails watching for and emphasizing risk triggers and identifying the root causes of the risk to help enhance impacts or probability.

Strategies for Both Threats and Opportunities

The third tool and technique of the Risk Response Planning process, strategies for both threats and opportunities, is called the acceptance strategy. Acceptance of a risk event is a strategy that can be used for risks that pose either threats or opportunities to the project. Passive acceptance is a strategy that means that you won't make any plans to try to avoid or mitigate the risk. You're willing to accept the consequences of the risk should it occur. Acceptance may also mean the project team was unable to come up with an adequate response strategy and must accept the risk and its consequences. Active acceptance may include developing contingency reserves to deal with risks should they occur. (We'll look at contingency reserves in the next section.) Let's revisit the road trip example. You could plan the trip using the original route and just accept the risk of running into construction. If you get to that point and you're delayed, you'll just accept it. This is passive acceptance. You could also go ahead and make plans to take an alternate route but not enact those plans until you actually reach the construction and know for certain that it is going to impede your progress. This is active acceptance and may involve developing a contingency plan.

Contingency Planning

The last tool and technique of the Risk Response Planning process is called the contingent response strategy, better known as contingency planning. Contingency planning involves planning alternatives to deal with the risks should they occur. This is different than mitigation planning in that mitigation looks to reduce the probability of the risk and its impact whereas contingency planning doesn't necessarily attempt to reduce the probability of a risk event or its impacts. Contingency planning says the risk may very well occur and we better have plans in place to deal with it. Contingency comes into play when the risk event occurs. This implies you need to plan for your contingencies well in advance of the threat. After the risks have been identified and quantified, contingency plans should be developed and kept at the ready. Contingency allowances or reserves are a common contingency response. Contingency reserves include project funds that are held in reserve to offset any unavoidable threats that might occur to project scope, schedule, cost, or quality. It also includes reserving time and resources to account for risks. You should consider stakeholder risk tolerances when determining the amount of contingency reserves. Fallback plans should be developed for risks with high impact or for risks with identified strategies that may not be the most effective. In practice, you'll find that identifying, prioritizing, quantifying, and developing responses for potential threats may happen simultaneously. In any case, you don't want to be taken by surprise, and that's the point of the risk processes. If you know about potential risks early on, you can very often mitigate them or prepare appropriate response plans or contingency plans to deal with them.

RISK RESPONSE PLANNING OUTPUTS

As you've no doubt concluded, the purpose of the Risk Response Planning process is to develop risk responses for those risks with the highest probability and impact on the project objectives. There are three outputs of the Risk Response Planning process: risk register updates, project management plan updates, and risk-related contractual agreements.

Risk Register Updates

Again, the risk register is updated at the end of this process with the information you've discovered during this process. The response plans are recorded in the risk register. You'll recall that the risk register lists the risk in order of priority (those with the highest potential for threat or opportunity first), so it makes sense that the response plans you have for these risks will be more detailed than the remaining lists. Some risks may not require response plans at all, but you should put them on a watch list and monitor them throughout the project.

After Risk Identification, Qualitative Risk Analysis, and Quantitative Risk Analysis are preformed, the following elements should appear in the risk register:

List of identified risks, including their descriptions, what WBS element they impact (or area of the project), categories (RBS), root causes, and how the risk impacts the project objectives

- Risk owners and their responsibility
- Risk triggers
- Response plans and strategies, including the steps to take to implement the strategy
- Cost and schedule activities needed to implement risk responses
- Contingency reserves for cost and time
- Contingency plans
- Fallback plans
- List of residual and secondary risks
- Probabilistic analysis of the project and other outputs of the Qualitative and Quantitative

Risk Analysis processes

The only elements in the preceding list we haven't talked about so far are residual and secondary risks. A residual risk is a leftover risk so to speak. After you've implemented a risk response strategy—say mitigation, for example—some minor risk may still remain. The contingency reserve is set up to handle situations like this.

Secondary risks are risks that come about as a result of implementing a risk response. The example given previously where you transferred risk by hiring a driver to take you to your destination but they became ill along the way is an example of a secondary risk. Their illness delayed your arrival time, which is risk directly caused by hiring the driver or implementing a risk response. When planning for risk, identify and plan responses for secondary risks that could occur as well.

Risk-Related Contractual Agreements

If you're planning on using strategies such as transference or sharing, for example, you may need to purchase services or items from third parties. The contracts for those services can be prepared now and discussed with the appropriate parties. Risks exist on all projects, and risk planning is an important part of the project Planning processes. Just the act of identifying risks and planning responses can decrease their impact if they occur. Don't take the "What I don't know won't hurt me" approach to risk planning. This is definitely a case where not knowing something can be devastating. Risks that are easily identified and have planned responses aren't likely to kill projects or your career. Risks that you should have known about but ignored could end up costing the organization thousands or millions of dollars, causing schedule delays, or ultimately killing the project. There could be a personal cost as well, as cost and schedule overruns due to poor planning on your part are not easily explained.

2.2.7 Monitor & Review

The step involves continual gauging of the risk environment and the performance of the risk management strategies. It also provides a context for considering risk that is scalable over a period of time. The results of the ongoing reviews are fed back into the context-setting step and the cycle repeats.

The best way to support monitoring and reviewing the risks are best described below:

- Provide preventive guidance, best practices, simulation, and testing;
- Provide and operate indications, alert, and warning capabilities;

- Provide and operate operation centers and teams;
- Provide and participate in information sharing, situational awareness, and information fusion activities; and
- Coordinate and provide response, recovery, and reconstitution.

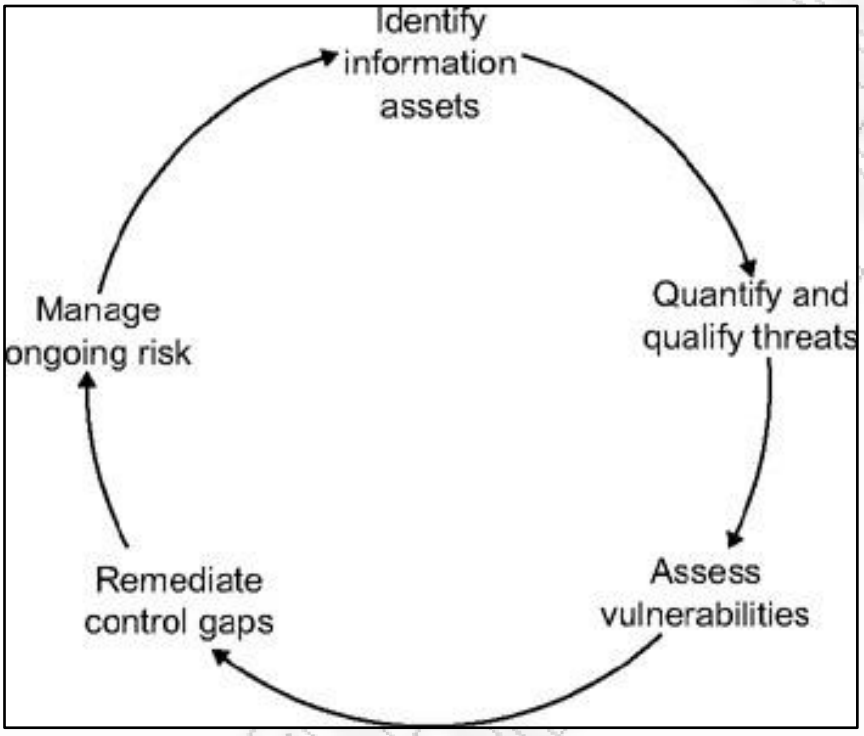


Figure 2.12 - Incident Management Lifecycle (Source: http://www.itservicestrategy.com/five-basic-phase-it-risk-management-life-cycle)

Continuous monitoring and review of risks ensures new risks are detected and managed, and that action plans are implemented and progressed effectively. Review processes are often implemented as part of the regular management meeting cycle, supplemented by major reviews at significant project phases and milestones. Monitoring and review activities link risk management to other management processes. They also facilitate better risk management and continuous improvement. The main input to this step is the risk watch list of the major risks that have been identified for risk treatment action. The outcomes are in the form of revisions to the risk register, and a list of new action items for risk treatment.

IMPLEMENTATION

Implementation follows all of the planned methods for mitigating the effect of the risks. Purchase insurance policies for the risks that have been decided to be transferred to an insurer, avoid all risks that can be avoided without sacrificing the entity's goals, reduce others, and retain the rest.

REVIEW AND EVALUATION OF THE PLAN

Initial risk management plans will never be perfect. Practice, experience, and actual loss results will necessitate changes in the plan and contribute information to allow possible different decisions to be made in dealing with the risks being faced.

Risk analysis results and management plans should be updated periodically. There are two primary reasons for this:

1. to evaluate whether the previously selected security controls are still applicable and effective, and
2. to evaluate the possible risk level changes in the business environment. For example, information risks are a good example of rapidly changing business environment.

BUSINESS RISK MANAGEMENT

Successful risk management in today's online world requires organizations to build appropriate threat management and vulnerability-management programs to manage risks and monitor the systems deployed to support critical business processes. A complete technology-management and riskmanagement program incorporates the following four principles:

- The enterprise must understand the requirements of the business process being assessed. These can include concerns over financial loss, damage to reputation, loss of intellectual property, devaluation of goods, and regulatory requirements (a critical driver), among other business-specific risks.
- The enterprise must understand failure modes, including knowledge of how specific system compromises or failures can affect a business process and its relative risk. These risks need to be aligned with a management strategy: funding corrective measures if plausible, developing compensating controls,

insuring the risk, and, in most cases, developing a detection method for these failure modes.

- The enterprise must map failure modes to a specific response. This procedure is critical to managing risks that require response, such as disclosure of data that may have reporting requirements in terms of the Notice of Security Breach Act (California Civil Code 1798.82,1 formerly California Senate Bill 1386), the failure of a system that may require administrative maintenance to return to service, or a specific failure mode that requires interruption of some activity to prevent financial loss.
- The enterprise must put in place detective controls and operational monitoring so that, when a failure mode occurs, it is detected without delay and the appropriate response is enacted. When this framework is practiced, systems risk—including vulnerabilities, design flaws, and/or weakness in strength of controls—can be better described. An understanding of the risk involved—failure modes in particular—begins with a clear definition of terms and an effort to ensure that the language is well developed. As a result, when a security incident warrants an executive decision such as a "go forward" strategy, the risk-management plan is already in place to mitigate the threat. This framework includes development of language to describe business-process risk and operation of supporting programs with the right levels of operational and capital spending, resulting in successful yet cost-effective security programs.

CONSIDERATIONS BEFORE MOVING ON

In software engineering, a piece of software is assembled from components or modules and these components in turn can be made up from smaller sub-components. The management of an information system involves tracking the further development and maintenance of the individual components. Though tools exist to track component dependencies and historical changes, the key software management hurdle is the manual evaluation of the trade-offs in leveraging current components vs. investing in new software development.

Consider a typical software project management scenario. The requirement specification is acquired from the customer and then, the solution architects devise

the Work Breakdown Structure of the problem to identify the different tasks breaking them down into smaller pieces. This information is input to a project management tool like Microsoft Project along with estimates on time and resources for each task. The tool may have elaborate guidelines on how to reason about a project - find the critical path in the project, compute the amount of time an individual task can be delayed (slack time), evaluate tasks to identify over-allocated resources, etc. It is not hard to see that the user, who may be a project manager or software architect, has to evaluate the relevance of existing components *manually* based on the project objectives like expected software functionality, performance and development time. This analysis also helps user scope out new development in the information system and estimate the overall integration effort involved.

Now consider the case when a software has been released and is now being maintained. If any updates/patches are available for the software components that were reused in the project, their impact is *manually* evaluated to decide if a new build of the software is necessitated. Though there are some tools to track component dependencies and record history of component releases, the key software management hurdle still remains that the trade-off choices have to be *manually* evaluated. Our contribution relates to this general area of software project management.

The project management processes, practices, and methods that are the key to this IT maturity model are based upon critical success factors. All too often in IT, project and system management do not allocate enough time to do the project/system work right the first time, but later they are forced to find the time and resources to do it over again. Completing IT projects successfully and delivering the appropriate IT system, for the first time, requires the identification and understanding of all the critical success factors of such projects. Once these factors are itemized and fully appreciated, then effective management and technical methods and metrics can be formulated for project performance, risk, and quality control. Theoretically and statistically, project/system success probability decreases as the size of an IT project/system grows. Many factors, such as the interaction of project/system stakeholders and the interaction of technical components, increase in complexity in ratio to the square of the number of such items. Therefore, subdividing large IT projects/systems into smaller parts decreases complexity and thus increases

the likelihood of success; however, this subdivision needs to be consistent with the metrics and methods to monitor and control all identified critical success factors.

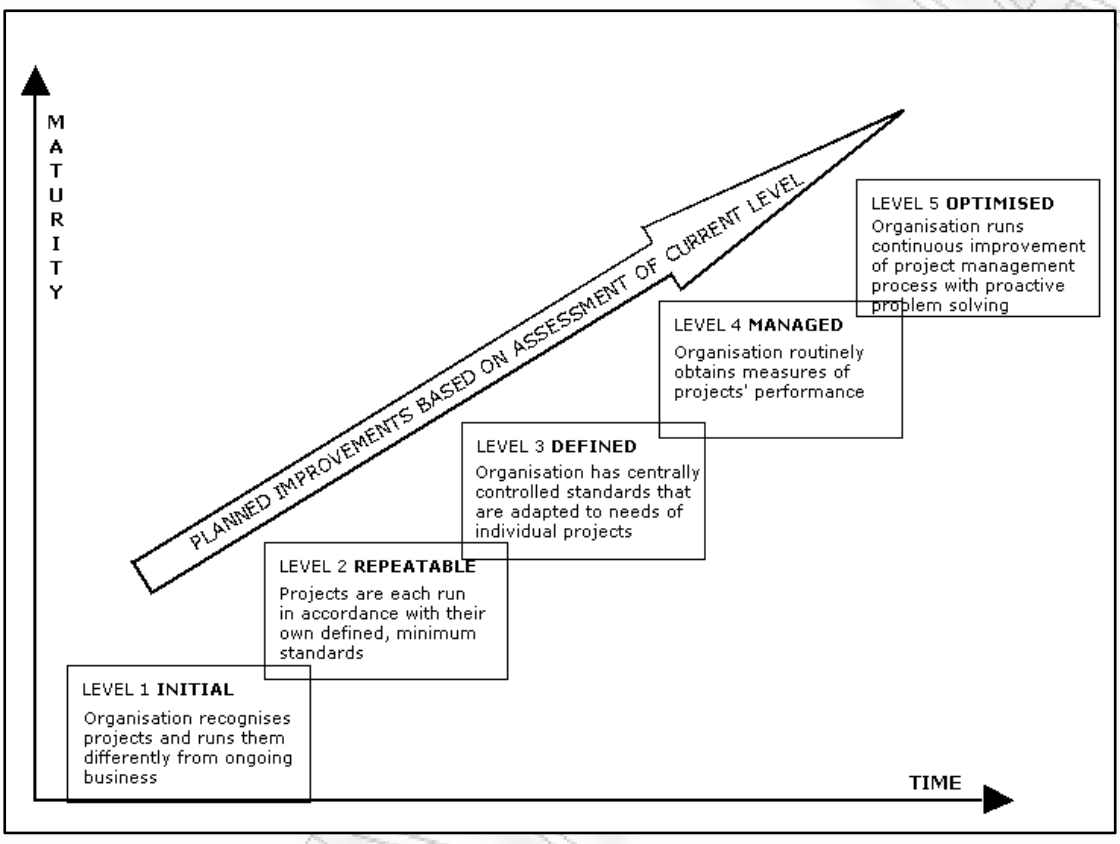


Figure 2.13. IT Project Management Maturity (Source: http://www.quality-projects.co.uk/main/maturity.html)

3. PROPOSITION

The proposition to address, and handle above issues is the induction of a combination of two individual tools. One **Risk Management Tool** and another supportive **Ticketing Tool**. These to tools implemented provide to relative user an overview of the projects, which can be monitored and stakeholders can be coordinated more efficiently and effectively. In more detail the basic functionalities of the application are described below:

1. Store and track identified project risks and possible issues (defects).
2. Submit and track suggestions/propositions/additions to the system. This provides client with a formal path to express and further develop alterations to functionality which wasn't initially predicted.
3. Ensure that preventive and tracking measures are in place to cope with risks.
4. Identify possible gaps in preventive measures.
5. Provide an effective tool in recording, maintaining, tracking and controlling identified risks in co-operation with the project plan.
6. Link risks with relative issues risen during operation of a system or development of a project
7. Identify the risk factor and the priority assigned to each risen issue in relevance with the project/system.

3.1 RISK MANAGEMENT TOOL

Two separate applications will assist incorporate all above. The **Risk Management Tool** will handle the following factors and will perform as explained below:

1. **RM Ticket Number:** Unique identification of the risk in the form of a number, where this number is a sequential number to avoid unnecessary complexity and be able to handle issues efficiently.
2. **Severity:** Value obtained by considering the likelihood of the risk materializing into a problem and its impact to the normal operation of the system. Values range from 1 to 5, five been the highest. Advice Figure 2.7.

Risk prioritization seeks to choose most critical risks to address from the full list of identified risks. Risk items may correspond to an event, to a number of alternative events – anyone of which may occur – or to a continuous distribution of possible outcomes. The primary choice for the most important items may theoretically be those with the largest risk exposure but, in reality, the decisions are more complex.

Below, there is a list of factors influencing the choice of risks for the prioritized list:

- Size of risk exposure of the risk item relative to the size of the target
- Size of the risk exposure of the risk item relative to the largest risk exposure
- Compound risks where a risk item is dependent upon another risk item that is ranked high on the prioritized list
- Maximum number of risk items that can be interpreted and acted upon by project management

In drafting a risk priority list, the objective is to choose those risk items which are prime candidates either for action while they are still risks or are of such concern that a close eye should be kept on them to trigger the necessary actions if they become more threatening.

The system operator will be responsible for defining the priority of a risk. The priority will depend on the likelihood (L) of the risk materializing to a problem, and its impact (I) to the project execution. Priority will be calculated by multiplying the Likelihood

rating by the Impact rating. As such, the more likely a risk may materialize into a problem and the more impact it has to the project, the higher priority the risk has for reduction/resolution and contingencies.

Priority Rating	Level of priority	For Project Execution	For the Product (e-Procurement system)
5	High	A risk of rating 5 can result to a critical issue/incident, which can form a "Show Stopper" for the operation of the system. For instance, a rating 5 risk may relate to crucial input expected by the project team in order to progress to the project implementation (i.e. delayed acceptance of a project deliverable).	A risk of rating 5 can relate to a critical system incident which can hinder the use of the system for its purpose or the use of a critical functionality
4 – 4.9	Medium to High	A risk of rating 4 is typically associated to issues/incidents that can result to considerable problems for the execution of the project but they are not considered as "blocking factors" For instance, if the schedule for submission of deliverables is not respected by both parties, the project could miss its objectives within the period defined in the Contract	A risk of rating 4, if materialized, can significantly interfere with the proper development and deployment of the system and thus reduce its functionality or the acceptance rate by end-users
3 – 3.9	Medium	A risk of rating 3 can result to a	A risk of rating 3 can

		major issue/incident, which can relate to an issue which affects the scheduled and pre-defined execution of project tasks. For instance, a rating 3 risk may relate to a critical technical decision that the Project Manager and the Developer must make in order for a specific system functionality to be implemented (i.e. delayed definition of user requirements for a specific operation of the system)	result to a major incident, which can hinder the use of one or more important functionalities of the system
2 – 2.9	Low to Medium	A risk of rating 2 can result to issues that are not considered critical for the execution of the project. For instance, missing information for deliverables that are not in the critical path of the project could result in minor system defects	A risk of rating 2 can result to minor problems in the operation of the system, mainly regarding the usability level
0.1 – 1.9	Low	A risk of rating 1 can result to a minor issue/incident, which although cannot impede the project execution, it can create obstacles to the smooth and on-time execution of the project	A risk of rating 1 can result to a minor incident which cannot hinder the user to use the business critical functionalities of the system. However, the implementation of that functionality is considered to be faulty and requires modification

0	No priority	The risk is closed (i.e. either the rating of its Likelihood or Impact is set to 0).	

3. **Likelihood:** Value representing an estimate of the probability of the risk actually materializing into a problem. Ratings range 1.0, 0.8, 0.6, 0.4, 0.2 & 0, 1.0 been the risk that will definitely materilize into an actual problem.
4. **Impact:** Level of impact that would occur if the Risk identified actually happens. Values range from 1 to 5, five means that if risk materialized into a problem, there would be critical impact to the operation of the system.
5. **Risk Area:** The grouping that the identified risk belongs to e.g. timeline, scope, effort
6. **Source of risk:** Description of cause (source) of the risk
7. **Identified threats:** Problems in the system behavior if source of risk materializes into an actual problem
8. **Risk:** The actual risk resulting from the respective sources of risks and identified threats
9. **Time Created / Identification Date:** Time stamp of the identified/potential risk
10. **Risk reduction plan:** Description of the plan for reducing the probability of the risk to materialize into an actual threat
11. **Risk contingency plan:** Description of the plan and the events that trigger them to handle the risk in case it materialize
12. **Date Closure:** The date when risk is closed (if applicable)
13. **Assigned ticket:** User can link already submitted tickets from the supportive tool "Incident Ticket Tracking Tool", such as defects or Suggestions.

This Risk Management Tools will also provide a link to and view to more specific issues raised, that interconnect with the particular risk. With this linkage user is capable to further evaluate the risk and relatively decide whether to rise or lower the severity/priority of the Risk and further adjust, refine the Reduction Plan or the Risk Contingency Plan. Details on the linked tickets are described below.

3.2 TICKETING TOOL

A supportive to the Risk Management tool will be implemented as well. An **Incident Ticketing Tool** for reporting defects and possible suggestions, for further linkage to the Risk Management Tool Ticket, in the scope to keep a to-do list as well as to prioritize, schedule and track dependencies. In the context of Risk Control, the Ticketing Tool will play the role of managing software-related problems that were initially identified as risks. This tool also includes a list that tickets appear linked with the RM tool.

A short description of the fields appearing in the **Ticketing Tool** main page is provided below:

1. **Synopsis:** Short Description of the issue risen
2. **Description:** Detailed description of the issue risen
3. **Type:** A selection from a drop-down menu having two values. The first and most important is the value "defect" which indicates that this ticket is risen due to software/system failure. The second selection would be "Suggestion" indicating that a new functionality should be implemented that was not described in original functional specifications of the system but now the client decided it is crucial to be implemented.
4. **Status:** Description of the status for the issue risen. The potential values appear below:
    - New: communicated informally from client, but not yet examined.
    - In progress / evaluated: A Developer is assigned to work on the issue but has not completed the task yet
    - Fixed: Developer has Fixed the defect risen
    - As Designed: the submitted issue was not a defect after all. An explanation to the client has been communicated to explain relatively. False alarm. No impact on the system
    - Closed Fixed: Client verifies that no defect exist any more
    - Closed As Design: Client confirms that issue risen wasn't a defect
5. **Priority:** Selection from a drop-down menu to indicate whether this issue is Low, Medium, High or Critical (show-stopper)

6. **Assign to:** Selection from a list of developers to link the issue for resolution/development

Both tools will provide the possibility to view all previously submitted Risks and Defects or Suggestions. The combination of these two tools will provide accurate feedback to a project manager when trying to identify risks on time. This will assist the organization to provide most of the time to its client quality products on time, on budget and efficiently.

4. METHODOLOGY

Over the years there has been a large amount of controversial discussion and arguments surrounding research methodologies. Much of this debate has centered on the issue of which one is best, qualitative or quantitative methods. Different methodologies become more or less popular depending on different environments they are developing. The normative literature indicates that, both methodologies have their specific strengths and weaknesses. These should be acknowledged and addressed by the individual performing a research.

Deciding the methodology

Taking as De facto that quantitative is better that qualitative research is the mistake most of researchers do. Neither is better than the other – they are just different and both have their, advantages and disadvantages depending on the model we test. A typical qualitative research involves words and a quantitative involves numbers. Qualitative analysis is inductive and quantitative is deductive.

One of the most basic questions is also one which is often overlooked. What kind of information is needed? Qualitative or quantitative? The most basic question in whether to do qualitative or quantitative research is whether the research needs to produce projectable data. Organizations frequently make the mistake of using methods and processes focusing on gathering numerical data. This is not always the case.

Quantitative research generates statistics through the use of large-scale survey research, using methods such as data collection methods. For example a researcher would have asked someone to fill in a questionnaire and then gather as many as he can. Later on he would have analyzed his data and by using relative tools he would have produced results through statistical analysis. This method falls under the umbrella of quantitative research.

Qualitative research explores attitudes, behavior and experiences through such methods as the actual use of the object investigated. It attempts to get an in-depth opinion from participants. As it is attitudes, behavior and experiences which are

important, fewer people tend to take part in such a research, but the contact with these people tends to last a lot longer. Under the umbrella of qualitative research there are many different methodologies.

A figure follows that outlines the basic and more distinct features that separate these two kind of methods, when using several criteria:

CRITERIA	QUALITATIVE	QUANTITATIVE
Purpose	To understand & interpret interactions.	To test hypotheses, look at cause & effect, & make predictions.
Group Studied	Smaller & not randomly selected.	Larger & randomly selected.
Variables	Study of the whole, not variables.	Specific variables studied
Type of Data Collected	Words, images, or objects.	Numbers and statistics.
Form of Data Collected	Qualitative data such as open – ended responses, interviews, participant observation, field notes & reflections	Quantitative data based on precise measurements using structured & validated data-collection instruments
Type of Data Analysis	Identify patterns, features, themes.	Identify statistical relationship
Objectivity and Subjectivity	Subjectivity is expected.	Objectivity is critical.
Role of Researcher	Researcher & their biases may be known to participants in the study, & participant characteristics may be known to the researcher.	Researcher & their biases are not known to participants in the study, & participant characteristics are deliberately hidden from the researcher (double blind studies).
Results	Particular or specialized findings that are less likely to get generalized.	Findings that can be generally applied to other populations.

Scientific Method	Exploratory or bottom–up: the researcher generates a new hypothesis and theory from the data collected.	Confirmatory or top-down: the researcher tests the hypothesis and theory with the data.
View of Human Behavior	Dynamic, situational, social, & personal.	Regular & predictable.
Most Common Research Objectives	Explore, discover, & construct.	Describe, explain, & predict.
Focus	Wide-angle lens; examines the breadth & depth of phenomena.	Narrow-angle lens; tests a specific hypotheses.
Nature of Observation	Study behavior in a natural environment.	Study behavior under controlled conditions; isolate causal effects.
Subjectiveness	Subjective.	Single reality, objective.
Final Report	Narrative report with contextual description & direct quotations from research participants.	Statistical report with correlations, comparisons of means, & statistical significance of findings.

4.1 THE APPROACH

This section is dedicated to describe my approach to apply qualitative research in Information Systems. As also mentioned above, Qualitative research involves the use of qualitative data, such as interviews, documents, and participant observation data, to understand and explain the systems been developed. In Information Systems, there has been a general shift in IS research away from technological to managerial and organizational issues. Therefore there is been an increasing interest in the application of qualitative research methods to such systems. The goal is to provide as much as possible useful information on qualitative research in ISs.

4.2 QUALITATIVE RESEARCH METHODS

A research method is a strategy of inquiry which moves from various and underlying assumptions to research design and data collection. The choice of research method influences the way in which the researcher collects data. Specific research methods also imply different skills, assumptions and research practices. Three fundamental research methods will be discussed below. These are:

- action research
- case study research and
- grounded theory

4.2.1 Action Research

Action research is an established research method in use in the social and medical sciences since the mid-twentieth century, and has increased in importance for information systems toward the end of the 1990s. Action research has developed a history within information systems. Action research varies in form, and responds to particular problem domains. The most typical form is a participatory method based on a five-step model, which will be explained later on. The method produces highly relevant research results, because it is grounded in practical action, aimed at solving an immediate problem situation while carefully informing theory.

Adapting Hult and Lennung's definition [1980] four major characteristics of IS action research are distinguishable:

1. Action research aims at an increased understanding of an immediate social situation, with emphasis on the complex and multivariate nature of this social setting in the Information System domain.
2. Action research simultaneously assists in practical problem solving and expands scientific knowledge. This goal extends into two important process

characteristics: First, there are highly interpretive assumptions being made about observation; second, the researcher intervenes in the problem setting.

3. Action research is performed collaboratively and enhances the competencies of the respective actors. A process of participatory observation is implied by this goal.

4. Action research is primarily applicable for the understanding of change processes in social systems.

Action research refers to a class of research approaches, rather than a single, monolithic research method. As a class, the various forms of action research share some agreed characteristics, and these characteristics distinguish action research from other approaches to social enquiry. A careful survey of the action research literature finds widespread agreement by action research authorities on four common characteristics:

- an action and change orientation;
- a problem focus;
- a process involving systematic and sometimes iterative stages and
- collaboration among participants

Action research has been described as a technique characterized by intervention experiments that operate on problems or questions perceived by practitioners within a particular context. The type of learning created by action research represents enhanced understanding of a complex social-organizational problem. The domain of information systems action research is clearest where the human interacts with information systems.

The ideal domain of the action research method is characterized by a social setting where:

1. the researcher is actively involved, with expected benefit for both researcher and organization,

2. the knowledge obtained can be immediately applied, there is not the sense of the detached observer, but that of an active participant wishing to utilize any new knowledge based on these observations,

3. the research is a cyclical process linking theory and practice

One clear area of importance in the ideal domain of action research is new or changed systems development methodologies. Studying new or changed methodologies involves the introduction of such changes. Theoretically, the study of a newly invented technique is impossible without intervening in some way to system been changed or developed and apply new technique into this environment. Action research is one of the few valid research approaches that we can study the effects of specific alterations in systems development methodologies in Information Systems.

The most prevalent action research description [Susman and Evered, 1978] details a five phase, cyclical process. The approach first requires the establishment of a client-system infrastructure or research environment. Then, five identifiable phases are iterated:

1. Diagnosing,
2. Action planning,
3. Action taking,
4. Evaluating and
5. Specifying learning.

The figure below illustrates this action research structural cycle.

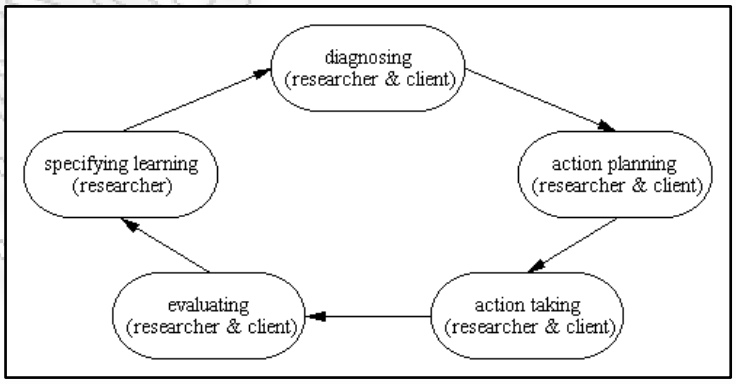


Figure 4.1 - Action Research Structural Cycle (Source: http://www.scu.edu.au/schools/gcm/ar/arr/arow/kms.html)

Diagnosing

Diagnosing corresponds to the identification of the primary problems that are the underlying causes of the organization's desire for change. Diagnosing involves self-interpretation of the complex organizational problem, not through reduction and simplification, but rather in a holistic context. This diagnosis will develop certain theoretical assumptions about the nature of the organization and its problem domain.

Action Planning

Researchers and practitioners then collaborate in the next activity, action planning. This activity specifies organizational actions that should relieve or improve these primary problems. The discovery of the planned actions is guided by the theoretical framework, which indicates both some desired future state for the organization, and the changes that would achieve such a state. The plan establishes the target for change and the approach to change.

Action Taking

Action taking then implements the planned action. The researchers and practitioners collaborate in the active intervention into the client organization, causing certain changes to be made. Several forms of intervention strategy can be adopted. For example, the intervention might be directive, in which the research "directs" the change, or non-directive, in which the change is sought indirectly. Intervention tactics can also be adopted, such as recruiting intelligent laypersons as change catalysts and pacemakers. The process can draw its steps from social psychology, e.g., engagement, unfreezing, learning and re-framing.

Evaluating

After the actions are completed, the collaborative researchers and practitioners evaluate the outcomes. Evaluation includes determining whether the theoretical effects of the action were realized, and whether these effects relieved the problems. Where the change was successful, the evaluation must critically question whether the action undertaken, among the myriad routine and non-routine organizational

actions, was the sole cause of success. Where the change was unsuccessful, some framework for the next iteration of the action research cycle (including adjusting the hypotheses) should be established.

Specifying Learning

While the activity of specifying learning is formally undertaken last, it is usually an ongoing process. The knowledge gained in the action research (whether the action was successful or unsuccessful) can be directed to three audiences:

- First, what Argyris and Schön [1978] call "double-loop learning," the restructuring of organizational norms to reflect the new knowledge gained by the organization during the research.
- Second, where the change was unsuccessful, the additional knowledge may provide foundations for diagnosing in preparation for further action research interventions.
- Finally, the success or failure of the theoretical framework provides important knowledge to the scientific community for dealing with future research settings.

The action research cycle can continue, whether the action proved successful or not, to develop further knowledge about the organization and the validity of relevant theoretical frameworks. As a result of the studies, the organization thus learns more about its nature and environment, and the constellation of theoretical elements of the scientific community continues to benefit and evolve.

4.2.2 Case Study

The term "case study" has multiple meanings. It can be used to describe a unit of analysis or to describe a research method. In our case, "Case study" refers to a research method. Case study research is the most common qualitative method used in information systems (Orlikowski and Baroudi, 1991; Alavi and Carlson, 1992). The scope of a "Case Study" can be defined as follows:

A case study is an empirical inquiry that:

- Investigates a contemporary phenomenon within its real-life context, especially when
- The boundaries between phenomenon and context are not clearly evident (Yin 2002).

Case study research method is well-suited to IS research, since the object of our discipline is the study of information systems in organizations, and "interest has shifted to organizational rather than technical issues".

4.2.3 Grounded Theory

Grounded theory is a research method that seeks to develop theory that is grounded in data systematically gathered and analyzed. According to Martin and Turner (1986), grounded theory is "an inductive, theory discovery methodology that allows the researcher to develop a theoretical account of the general features of a topic while simultaneously grounding the account in empirical observations or data." The major difference between grounded theory and other methods is its specific approach to theory development - grounded theory suggests that there should be a continuous interplay between data collection and analysis.

Grounded theory approaches are becoming increasingly common because the method is extremely useful in developing context-based, process-oriented descriptions and explanations of the phenomenon.

4.3 QUANTITATIVE RESEARCH METHODS

Researchers and Educators give various definitions to “quantitative research.” Quantitative research is the numerical representation and manipulation of observations for the purpose of describing and explaining the phenomena that those observations reflect. It is used in a wide variety of natural and social sciences, including physics, informatics, biology, psychology, sociology and geology (Wikipedia Encyclopedia, 2005).

In addition, according to Cohen (1980), quantitative research is defined as social research that employs empirical methods and empirical statements. He states that an empirical statement is defined as a descriptive statement about what "is" the case in the "real world" rather than what "ought" to be the case. Typically, empirical statements are expressed in numerical terms. Another factor in quantitative research is that empirical evaluations are applied. Empirical evaluations are defined as a form that seeks to determine the degree to which a specific program or policy empirically fulfills or does not fulfill a particular standard or norm.

Moreover, Creswell (1994) has given a very concise definition of quantitative research as a type of research that is `explaining phenomena by collecting numerical data that are analyzed using mathematically based methods (in particular statistics).'

The first element of this definition is explaining phenomena. This is a key element of all research, either quantitative or qualitative. When we set out to do some research, we are always looking to explain something. The next part of the definition refers that quantitative research is primarily collecting numerical data. This is closely connected to the final part of the definition: analysis using mathematically-based methods. In order to be able to use mathematically based methods our data have to be in numerical form. This is not the case for qualitative research. Qualitative data are not necessarily or usually numerical, and therefore cannot be analyzed using statistics. The last part of the definition refers to the use of mathematically based methods, in particular statistics, to analyze the data. This is what people usually think when they think of quantitative research, and is often seen as the most important part of quantitative studies. This is a bit of a misconception. While it is important to use the right data analysis tools, it is even more important to use the right research design and data collection instruments. However, the use of statistics to analyze the data is the element that puts a lot of people off doing quantitative research, because the mathematics underlying the methods seems complicated.

Therefore, because quantitative research is essentially about collecting numerical data to explain a particular phenomenon, particular questions seem immediately suited to being answered using quantitative methods. For example,

- How many students learning "Mathematics I" get A's in the first semester?

• What percentage of the students learning "Mathematics I" has negative attitudes towards the course?

• On average, is there any significant difference between students learning "Mathematics I" in University of Athens and those from University of Piraeus?

These are all questions we can look at quantitatively, as the data we need to collect are already available to us in numerical form. However, there are many phenomena we might want to look at, but which don't seem to produce any quantitative data. In fact, relatively few phenomena occur in the form of `naturally' quantitative data. Many data that do not naturally appear in quantitative form can be collected in a quantitative way.

Examples of this are attitudes and beliefs. These attitudes obviously do not naturally exist in quantitative form. However, we can develop a questionnaire that asks pupils to rate a number of statements (for example, `"Mathematics I" is an exciting lesson') as either agree strongly, agree, disagree or disagree strongly, and give the answers a number (e.g. 1 for disagree strongly, 4 for agree strongly). Now we have quantitative data on pupil attitudes to school. In the same way, we can collect data on a wide number of phenomena, and make them quantitative through data collection instruments like questionnaires or tests.

The number of phenomena we can study in this way is almost unlimited, making quantitative research quite flexible. However, not all phenomena are best studied using quantitative methods. While quantitative methods have some notable advantages, they also have disadvantages. This means that some phenomena are better studied using qualitative methods.

In short, quantitative research generally focuses on measuring social reality. Quantitative research and/or questions are searching for quantities in something and to establish research numerically. Quantitative researchers view the world as reality that can be objectively determined so rigid guides in the process of data collection and analysis are very important.

4.4 DECISION

Upon the completion of defying all methods, qualitative and quantitative, I came to a conclusion of choosing as ideal method, the "Action Research" one. This method is commonly used when studying or evaluating information systems. Due to the fact that there are no numerical data to analyze, quantitative analysis was impossible to apply. Qualitative methods mainly analyze visual and not structured data.

Action research provided both observation and participation in the system developed. It also emphasizes on the increased understanding of the situation or root cause for which the information system should be developed for. Additionally it simultaneously assists in practical problem solving and expanding knowledge for the system.

Since the knowledge obtained can also be applied and the observer can interact with the system and involved actively on demand the decision for applying this method was more than obvious.

5. DATA ANALYSIS

5.1 Introduction

Software developers used to write much lines of code and various logical functions and procedure to support reuse of their code. In this chapter we will look in-depth the architectural design and all tools needed and used for the development of this thesis. This project used a 3-Tier Architecture. In software engineering multi-tier architecture is a client-server architecture in which the presentation, the application processing and the data management are logically separate processes. Breaking up an application into separate tiers, developers only have to modify or add a specific layer, rather than have to rewrite the entire application over. There should be a presentation tier, a business tier, and a data tier. Sample of the architectural design is shown in following figure.

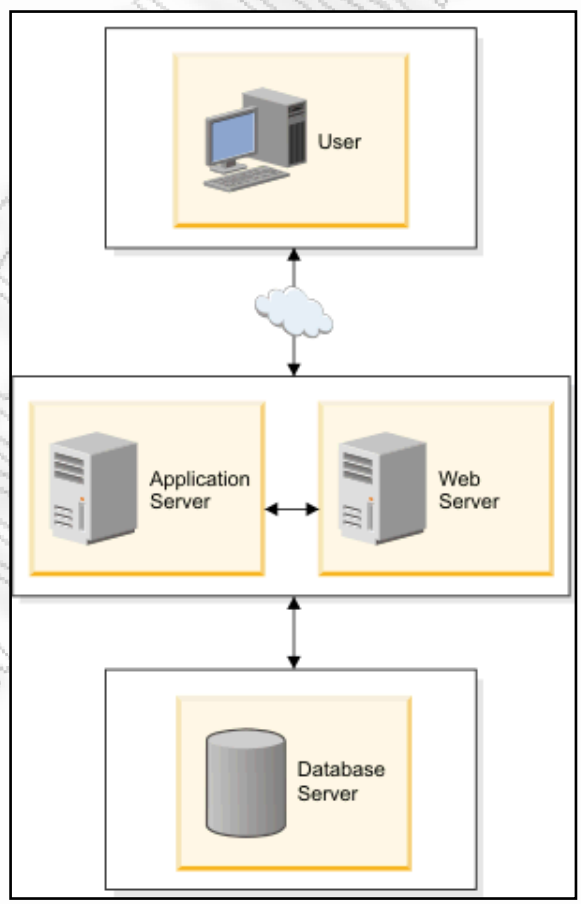


Figure 5.1 – Typical example of a 3-Tier Architecture

There has been an enormous success of the World Wide Web. Today many applications are developed for the web, in such different areas as banking and finance, e-commerce, education, government and entertainment. The complexity and sophistication of web applications grows day-by-day. The development cycles for Web Applications are much shorter than for more tradional software systems due to their reduced complexity. As a consequence Web Applications have a higher probability of failure during operation.

5.2 ARCHITECTURE DESIGN

5.2.1 Database Design

Since we have decided to follow the two-tier architectural design path the first element to be examined is the setup and the architectural approach of the database. The database should be designed in such a way that it can evolve, changing to meet the future information needs of the organization. This evolution is possible when the designer develops a true conceptual model of the organization with the following characteristics:

- The model faithfully mirrors the operations of the organization
- It is flexible enough to allow changes as new information needs arise
- It is independent of physical implementation

A well-designed conceptual (abstract) database model protects the data resource by allowing it to evolve so that it serves both today's and tomorrow's information needs. Even if the database management system chosen for implementation is replaced, the logical model may change, but the conceptual model of the enterprise can survive. The staged database design approach is a top-down method that begins with general statements of needs, and progresses to more and more detailed consideration of problems. Different problems are considered at different phases of the project. These phases are distinctively presented in stages below.

1. Actual Needs/ Analyze environment
2. Conceptual design
3. Logical Database design – Choose DBMS
4. Physical design
5. Build
6. Try/Test

1. Analyze your Needs / Environment

The first step in designing a database is to determine the current user environment. The designer determines inputs and outputs by asking the key users of a system. They help him out into how they intend to use the system. The designer should additionally identify their needs in order to apply the correct procedures and methods. The designer considers not only present needs but possible new applications or future uses of the database. The result of this analysis is a model of the user environment and requirements.

In our case the environment which the application is going to be used upon, is an MS Window® based environment running various browser applications (IE, Firefox, Chrome), for ease of use. Users in this stage are questioned in order the designer to understand their business needs and comprehend how to develop the conceptual design of the database that follows. Additionally the designer decides on the database and the technology which is going to use. In our case what I have used is an MS SQL Server. Since the architecture is a 3-tier one I have decided for the application/business layer to user MS .NET as a development tool. The final two decisions are based on current market research that indicated that most software houses are Windows oriented and additionally and more crucially these tools were in my disposal for free via the Piraeus University MSDNAA network. The needs of this system appear in the list below:

- A ticketing tool to submit and track new software defects and enhancements
- A ticketing tool that the user is able to submit and track Risks (from now on called Risk Management Tool)
- The ticketing tool must have the capability to store information such as:
    - Synopsis of the ticket

    - Description
    - Type
    - Status
    - Priority
- The Risk Management Tool must have the capability to store information such as:
    - Risk Area
    - Source of the risk
    - Identified threads
    - Risk Synopsis
    - Risk Reduction Plan
    - Risk Contingency Plan
    - Risk Severity (Likelihood * Impact = Priority), matrix shown in Figure 2.7
    - Event for reducing risk
    - Risk Status
    - A field to select tickets from Ticketing Tool to link
- The Risk Management Tool tickets are linked with the Ticketing Tool individual tickets in a relationship one-to-many.
- Each ticket in both tools should be unique
- There should be possibility to view the tickets from each tool individually, and for each one, the relationship between them (Ticketing and Risk Management)
- The tools should be web applications for ease of access of the managers (One common UI preferably).

Furthermore the type of users, for these tools, are two. The administrator that have the responsibility of the maintenance and the managers who are the actual users that request access grant from administrators.

2. Development of the conceptual data model

Using the model of the user environment, the designer develops a detailed conceptual model of the database — identifying the entities, attributes, and

relationships that will be represented. In addition to the conceptual model, the designer has to consider how the database is to be used. The types of applications and transactions, the kinds of access are also considered. The result of this phase is a conceptual data model. A conceptual data model identifies the highest-level relationships between the different entities. Features of conceptual data model include:

- Includes the important entities and the relationships among them.
- No attribute is specified.
- No primary key is specified

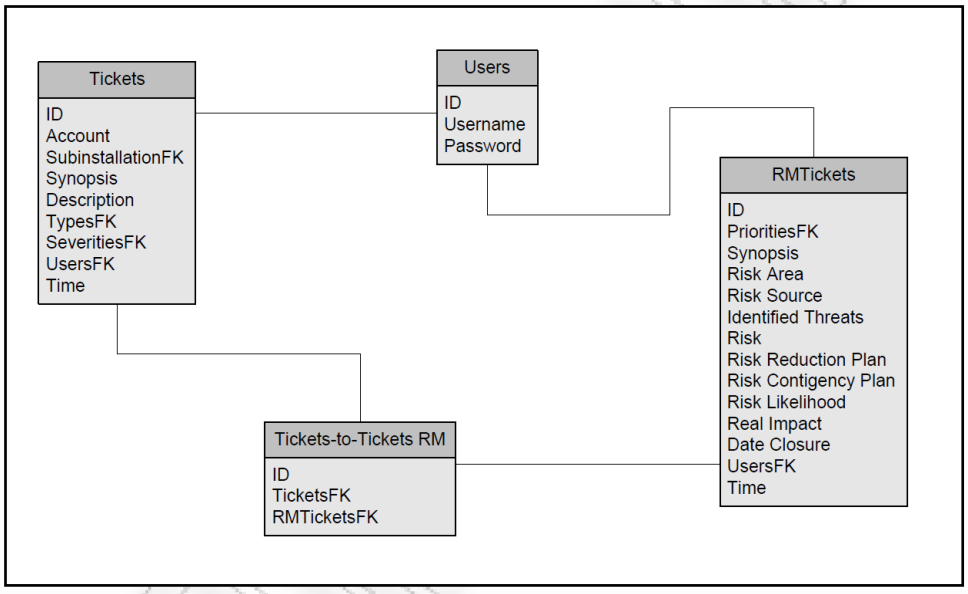


Figure 5.2 Conceptual Database Model

The Database Specifications are intended to support program coding and database generation by the developer. A brief system overview description is required as a point of reference for all stakeholders.

3. Choosing DBMS / Logical Database Design

The designer uses the above knowledge and the available software and hardware to determine the correct Database management system. He also maps the conceptual model to the data model to create the logical model. In my case the

database management system chosen was the MS SQL Management Studio. It is a comprehensive model or view of the workings of the organization in the mini-world. All the entities, with their attributes and relationships, are represented in the logical model using the data model that the DBMS supports. The model includes any constraints on the data and semantic information about the data meanings. The logical model supports the external views, in that any data available to any user must be present in or derivable from the logical model. The logical model is relatively constant. When the DBA originally designs it, he or she tries to determine present and future information needs and attempts to develop a lasting model of the organization. Therefore, as new data needs arise, the logical model might already contain the objects required. If that is not the case, the DBA expands the logical model to include the new objects. A good logical model will be able to accommodate this change and still support the old external views. Only users who need access to the new data should be affected by the change. The logical schema is a complete description of the information content of the database. The DBMS uses the logical schema to create the logical record interface, which is a boundary below which everything is invisible to the logical level and which defines and creates the working environment for the logical level. No internal or physical details such as how records are stored or sequenced cross this boundary. The logical model actually a collection of logical records.

The logical model is presented in the below figure 5.3

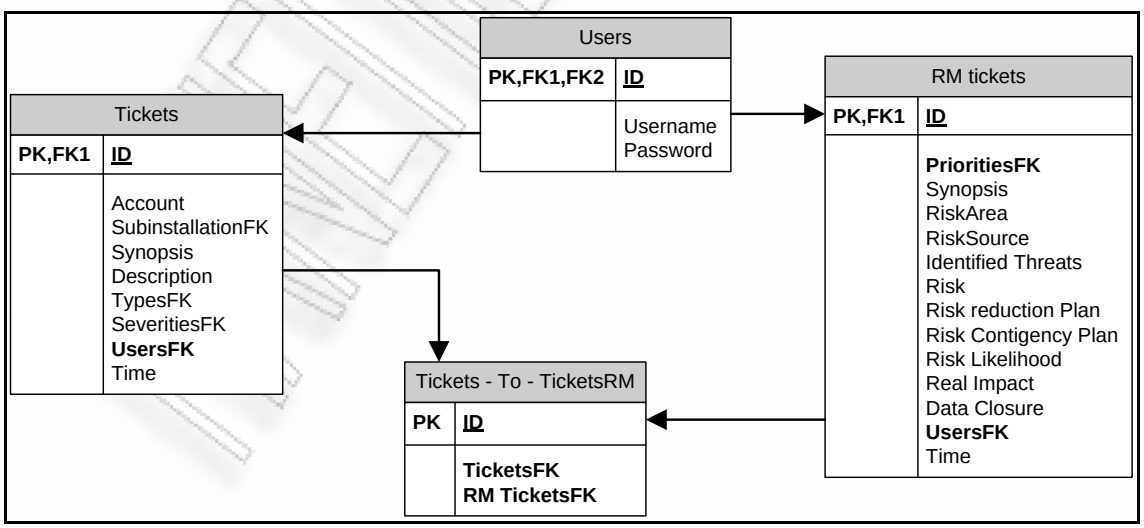


Figure 5.3 Logical Model DB Layout

4. Develop the physical model

In this stage the designer plans the layout of data considering the structures supported by the DBMS. The below figure was automatically generated by Microsoft SQL Management Studio.

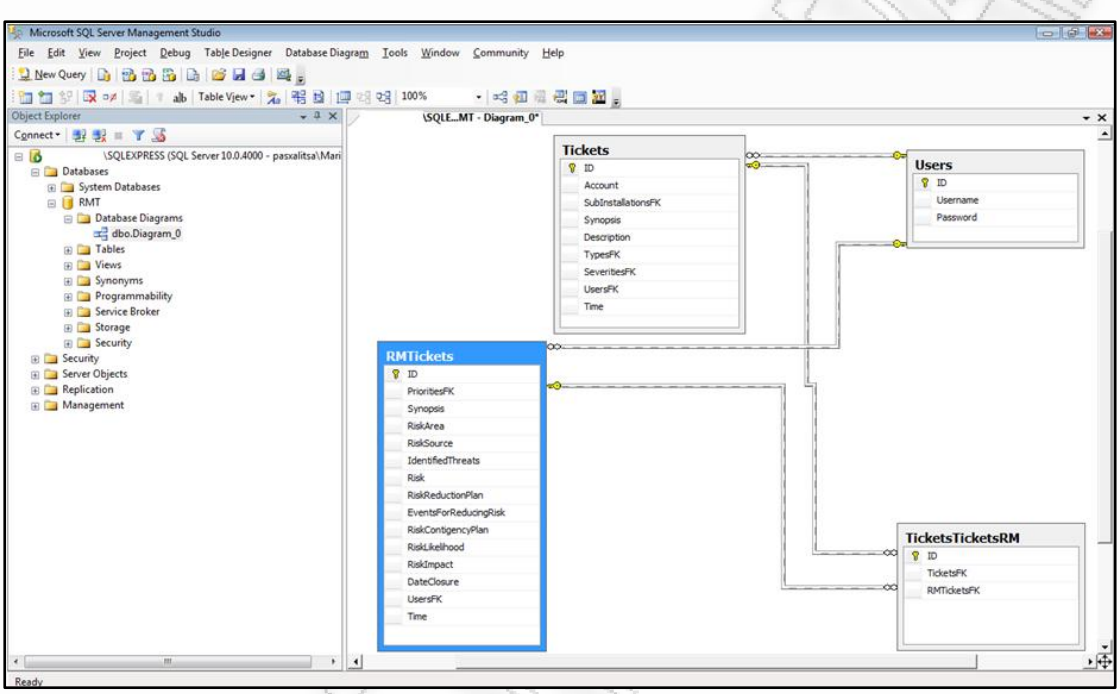


Figure 5.4 Physical Model

5. Build

This stage involves the actual development of the database and the final implementation of all above stages. Upon the completion of the physical model, the architect of the database creates the structure of the database using the data definition language from the chosen DBMS. Therefore physical datasets, libraries and data loading are established. Additionally standards should be enforced since the database is shared, accesses and probably developed by various individuals. Users who are responsible for inserting and updating data must follow the standard format for entering data. Typical standards are default values, to show acceptable ranges of items to users, some data standards are specifications for null values, codes, punctuation, and capitalization.

6. Try/Test/Evaluate

At this point the database is completed and a tester is assigned to perform various function and generally interfere with database to check if all above designs have been implemented as originally was designed.

5.2.2 Web Application Design

The design of the application is all about decisions which influence the characteristics of the arising system. The modern Web architecture emphasizes scalability and component interactions and independencies. Architecture determines how system elements are identified and allocated, how the elements interact to form a system, the amount of communication needed for interaction and the interface protocols used for communication. When designing this Web application, the goal was to minimize complexity to ensure adequate performance.

Referring to web applications it is crucial to know that due to the nature of this application the browser creates an HTTP request for specific URL/s that map to resources on a Web Server (like the one designed in above section). The server renders and returns HTML pages to the client, which the browser can display. The core of a web application is its server side logic. The application can contain several distinct layers. The illustration below show the web architecture used for this application

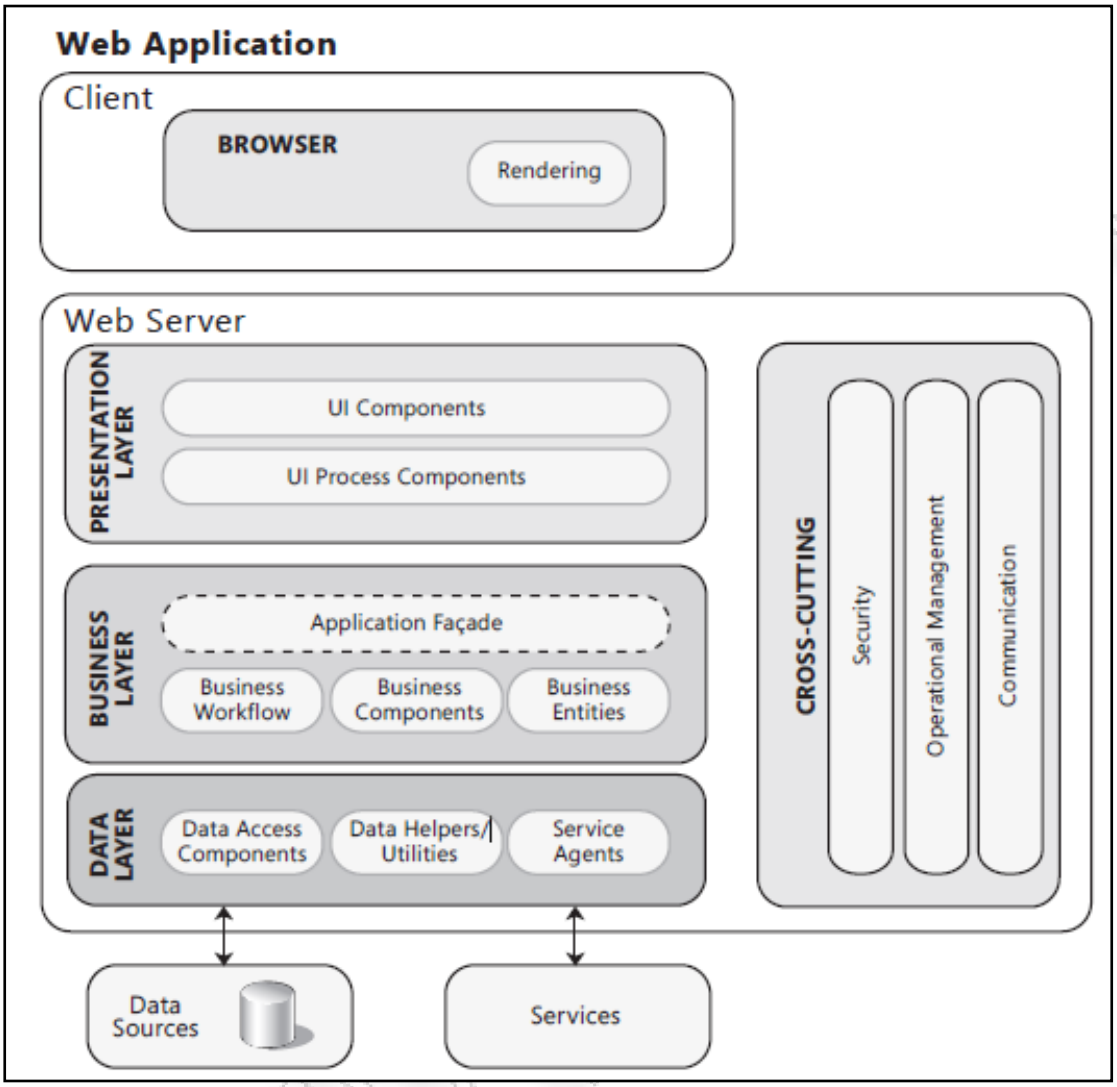


Figure 5.5 – An example of Web Application Design (Source: http://blogs.msdn.com/b/jmeier/archive/2008/11/24/application-architecture-diagrams-added-to-codeplex.aspx)

Designing a web application there are various guidelines that should be followed and several common issues that the designer must consider as the procedure develops. The key principles to follow are:

- **Separation of concerns**. Divide your application into distinct features with as little overlap in functionality as possible. The important factor is minimization of interaction points to achieve high cohesion and low coupling. However, separating functionality at the wrong boundaries can result in high coupling and complexity between features even though the contained functionality within a feature does not significantly overlap.

- **Single Responsibility principle**. Each component or module should be responsible for only a specific feature or functionality, or aggregation of cohesive functionality.
- **Principle of Least Knowledge.** A component or object should not know about internal details of other components or objects.
- **Don't repeatyourself**. You should only need to specify intent in one place. For example, in terms of application design, specific functionality should be implemented in only one component; the functionality should not be duplicated in any other component.
- **Minimize upfront design.** Only design what is necessary. In some cases, you may require upfront comprehensive design and testing if the cost of development or a failure in the design is very high. In other cases, especially for agile development, you can avoid big design upfront (BDUF). If your application requirements are unclear, or if there is a possibility of the design evolving over time, avoid making a large design effort prematurely.

Specific Design Issues

Designing an application there are also several common issues that must be considered. These issues can be categorized into specific areas of design. The section below highlights the areas software mistakes occur, with the most probability.

- **Authentication**
  Designing an effective authentication strategy for the business layer is important for the security and reliability of the application. Failure to do so can leave your application vulnerable to spoofing attacks, dictionary attacks, session hijacking, and other types of attacks.
- **Authorization**
  Designing an effective authorization strategy for the business layer is important for the security and reliability of the application. Failure to do so can leave the application vulnerable to information disclosure, data tampering, and elevation of privileges.

- **Caching**

  Designing an appropriate caching strategy for the business layer is important for the performance and responsiveness of the application. Caching is often used to optimize reference data lookups, avoid network round trips, and avoid unnecessary and duplicated processing. As part of the caching strategy, when and how to load the cache data mst be initially determined

- **Exception Management**

  Designing an effective exception management solution for the business layer is important for the security and reliability of an application. Failing to do so can leave the application vulnerable to Denial of Service (DoS) attacks, and may allow it to reveal sensitive and critical information about the application. Raising and handling exceptions is an expensive operation, so it is important that the exception management design takes into account the impact on performance.

- **Logging & Auditing**

  Designing a good logging and auditing solution for the business layer is important for the security and reliability of the application. Failing to do so can leave your application vulnerable to repudiation threats, where users deny their actions. Log files may also be required to prove wrongdoing in legal proceedings. Auditing is generally considered most authoritative if the log information is generated at the precise time of resource access, and by the same routine that accesses the resource. System monitoring tools can use this instrumentation, or other access points, to provide administrators with information about the state, performance, and health of an application.

- **Validation**

  Designing an effective validation solution for the business layer is important for the usability and reliability of the application. Failure to do so can leave the application open to data inconsistencies and business rule violations, and a poor user experience. In addition, it may leave the application vulnerable to security issues such as cross-site scripting attacks, SQL injection attacks, buffer overflows, and other types of input

attacks. There is no comprehensive definition of what constitutes a valid input or malicious input. In addition, how the application uses input to influence the risk of the exploit.

5.2.3 AJAX

One of the technologies used in the web application is AJAX. This was due to the one of the problems with web applications face. Typically every time a page gets data dynamically from the server, the data is displayed in a new page. Therefore the user has to wait for this new page to load before continuing. This can often take time, and users can get frustrated, and the experience is more disjointed than using a typical GUI application. Then the usability of such a web application is not good.

AJAX is is **A**synchronous **Ja**vaScript + **X**ML, and is one solution to this problem. In an AJAX application, an HTML page, makes asynchronous calls to the server using JavaScript and loads the data in bits and pieces as needed. Asynchronous means that the browser does not wait around doing nothing until the data arrives. This means that a page can get data from a server and display the data, without having to refresh the page.

Generally an AJAX application uses the following stages:

1. JavaScript in an HTML page creates an XMLHttpRequest object, which sends an HTTP request to a server
2. The server processes the request and sends back the result, of a database query for example, as XML.
3. JavaScript in the page (using a callback function) detects when the result has been received, and uses the browser to display the data

5.2.4 Microsoft® .NET® Framework & Visual Studio

By definition Microsoft® .NET® Framework is

- Common Language Runtime – provides an abstraction layer over the operating system
- Base Class Libraries – pre-built code for common low – level programming tasks
- Development frameworks and technologies – reusable, customizable solutions for larger programming tasks

And allows developers to:

- Apply common skills across a variety of devices, application types, and programming tasks
- Integrate with other tools and technologies to build the right solution with less work
- Build compelling applications faster

According to Microsoft, Microsoft Visual Studio is a powerful tool that ensures quality code throughout the entire application lifecycle, from design to deployment. Whether you're developing applications for the web, Windows, Windows Phone, and beyond, Visual Studio is your ultimate all-in-one solution. This development tool was used because it is a capable, reliable and there are plenty of references in the web for optimum support. Additionally this software is provided for free by the University of Piraeus in collaboration with Microsoft via MSDNAA (Microsoft Developer Network Academic Alliance) for all students (http://msdnaa.cs.unipi.gr/login.php). This service also provided the SQL Management Studio used for the design of the database

5.2.5 ADO.NET

Most applications and more specifically web applications, as the one developed for this thesis, need data access most of the times, making it a crucial component when working with such applications. Data access is making the application interact with a database, where all the data is stored. Different applications have different requirements for database access. .NET uses ADO

.NET (**A**ctive_X **D**ata **O**bject) as it's data access and manipulation protocol which also enables developers to work with data on the Internet.

Evolution of ADO.NET

The first data access model, DAO (**D**ata **A**ccess **M**odel) was created for local databases with the built-in Jet engine which had performance and functionality issues. Next came RDO (Remote Data Object) and ADO (Active Data Object) which were designed for Client Server architectures but, soon ADO took over RDO. ADO was a good architecture but as the language changes so is the technology. With ADO, all the data is contained in a record-set object which had problems when implemented on the network and penetrating firewalls. ADO was a connected data access, which means that when a connection to the database is established the connection remains open until the application is closed. Leaving the connection open for a long time, raises concerns about database security and network traffic. Also, as databases are becoming increasingly important and as they are serving more people, a connected data access model makes us think about its productivity. For example, an application with connected data access may do well when connected to two clients, the same may do poorly when connected to 10 and might be unusable when connected to 100 or more. Also, open database connections use system resources to a maximum extent making the system performance less effective.

Why ADO.NET?

To cope up with some of the problems mentioned above, ADO .NET came into existence. ADO .NET addresses the above mentioned problems by maintaining a disconnected database access model which means, when an application interacts with the database, the connection is opened to serve the request of the application and is closed as soon as the request is completed. Likewise, if a database is updated, the connection is opened long enough to complete the Update operation and is closed. By keeping connections open for only a minimum period of time, ADO .NET conserves system resources and provides maximum security for databases and also has less impact on system performance. Also, ADO .NET when interacting with the database uses XML and converts all the data into XML format for database related operations making them more efficient.

The Architecture

Data Access in ADO.NET relies on two components: DataSet and Data Provider.

DataSet

The dataset is a disconnected, in-memory representation of data. It can be considered as a local copy of the relevant portions of the database. The DataSet is persisted in memory and the data in it can be manipulated and updated independent of the database. When the use of this DataSet is finished, changes can be made back to the central database for updating. The data in DataSet can be loaded from any valid data source like Microsoft SQL server database, an Oracle database or from a Microsoft Access database.

Data Provider

The Data Provider is responsible for providing and maintaining the connection to the database. A Data Provider is a set of related components that work together to provide data in an efficient and performance driven manner. The .NET Framework currently comes with two Data Providers: the SQL Data Provider which is designed only to work with Microsoft's SQL Server 7.0 or later and the OleDb Data Provider which allows us to connect to other types of databases like Access and Oracle. Each Data Provider consists of the following component classes:

- The Connection object which provides a connection to the database
- The Command object which is used to execute a command
- The DataReader object which provides a forward-only, read only, connected recordset
- The DataAdapter object which populates a disconnected DataSet with data and performs update

The data access with ADO .NET can be summarized as follows. A connection object establishes the connection for the application with the database. The command object provides direct execution of the command to the database. If the

command returns more than a single value, the command object returns a DataReader to provide the data. Alternatively, the DataAdapter can be used to fill the Dataset object. The database can be updated using the command object or the DataAdapter. The below image illustrates the components of ADO .NET architecture.

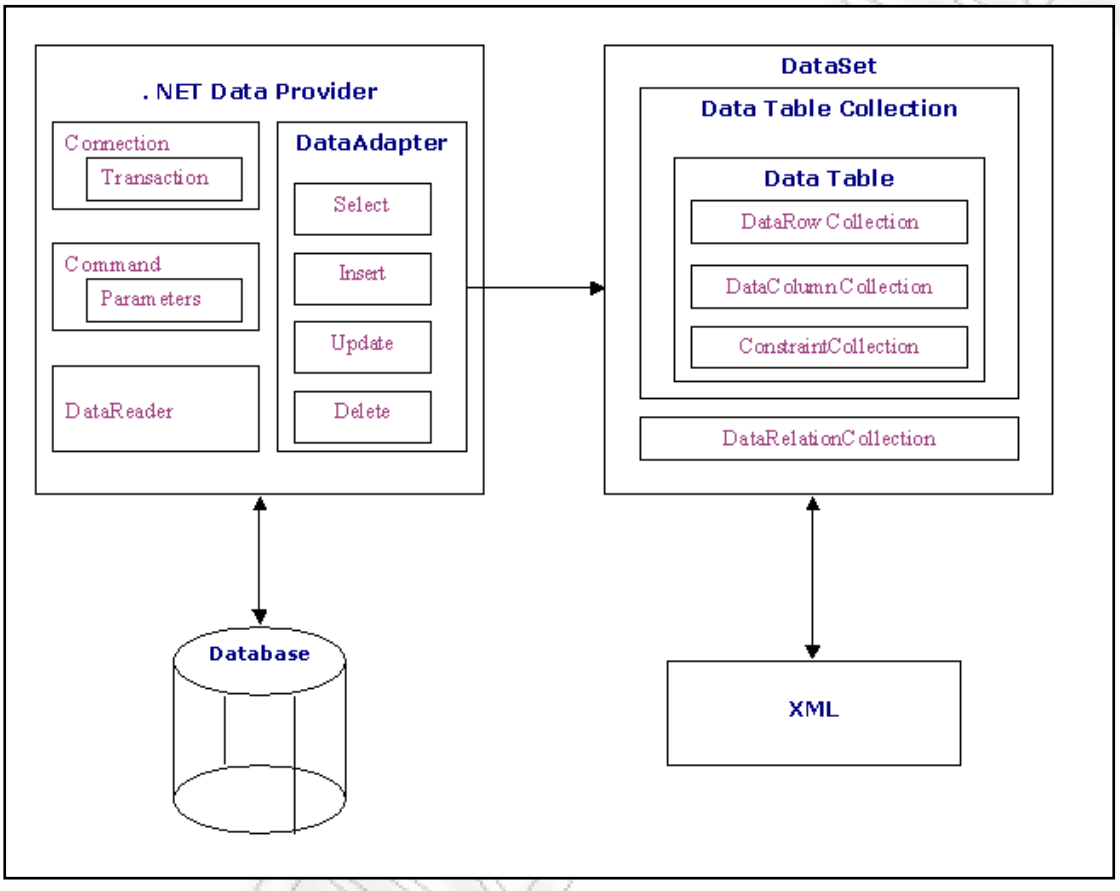


Figure 5.6 – Typical ADO .NET Data Architecture (Source: http://msdn.microsoft.com/en-us/library/27y4ybxw(v=vs.71).aspx)

5.3 APPLICATION BUSINESS SCENARIOS

This section of the document provides a sample of a use case used and various business scenarios from which the application architecture was based on.

5.3.1 Use Case Specification example – "Submit new Ticket"

5.3.1.1 Short Description

This Use Case "Submit New Ticket" describes in detail the way a new ticket to the "Ticketing Tool" (NOT the Risk Management tool) is created.

5.3.1.2 User Roles

- All Stakeholders

5.3.1.3 Prerequisites

User must exist in relative database having a unique username and password. Additionally user should use a web browser to access the application

5.3.1.4 Additional Conditions

In case of a successful submission, the application database is updated with the changes made by user.

In case of unsuccessful submission, the application database remains intact.

5.3.1.5 Actions Flow

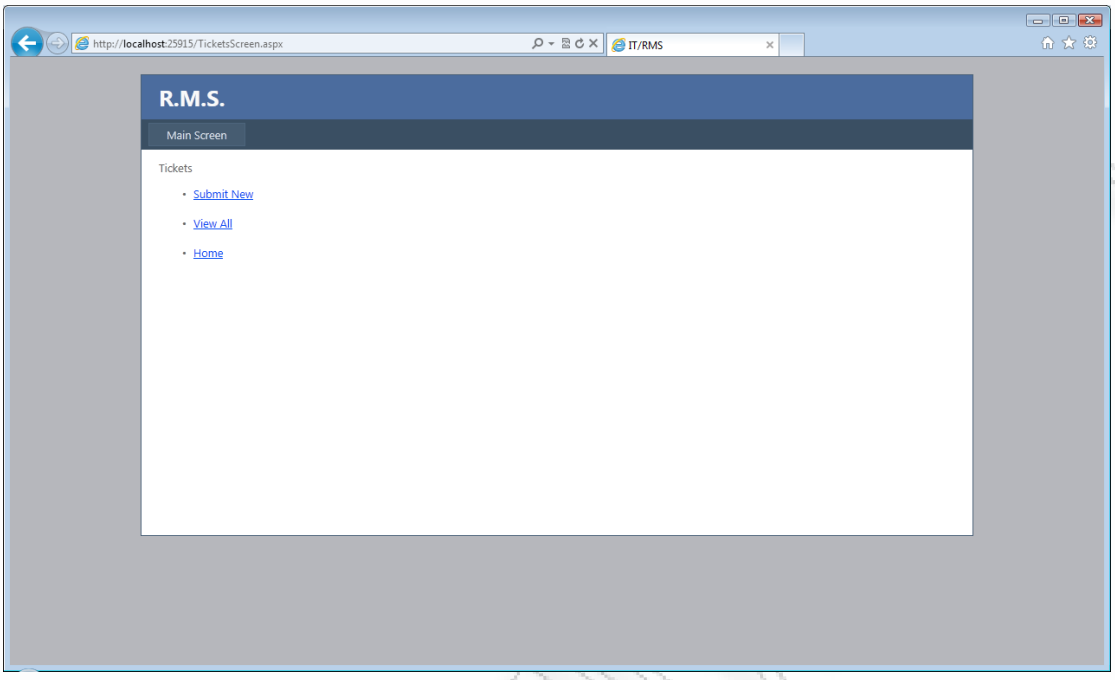


Figure 5.7 – Select to submit new ticket screen

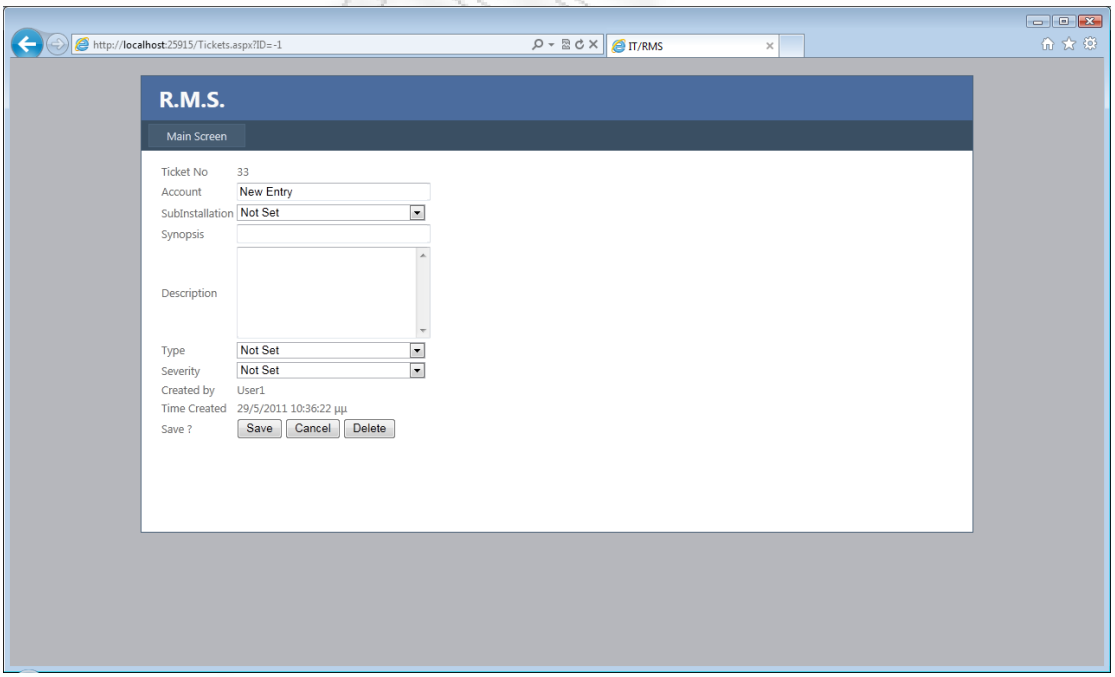


Figure 5.8 – Form to complete the submission of new ticket

Field Name	Type of Field	Additional Information
Ticket Number	Numeric	Automatically generated, upon creation of the form, indicating the unique ticket number.
Account	Alpha-Numeric	The client name this ticket refers to
Subinstallation	List of Value	User selects three different values. This refers to the environment this issue is risen
Synopsis	Alpha-Numeric	Synopsis of the ticket
Description	Alpha-Numeric	Description of the ticket
Type	List of Value	User select whether this ticket is a Defect or a Suggestion/Question
Severity	List of Value	User selects if the ticket is low, medium or high criticality
Created By	Alpha-Numeric	Automatically generated upon creation of the form indicating the name of the user, as appear in the application database
Time Created	Date	Automatically generated upon creation of the form indicating date this ticket created.

Figure 5.9 – New ticket submission form elements

5.3.1.6 Basic Flow

1. User selects link "Submit New" from the screen which appears in figure 5.7

2. The application navigates user to the screen which appears in figure 5.8
3. User completes form bearing in mind the table in figure 5.9
4. User selects "Save" button to conclude

5.3.2 Business Scenarios

5.3.2.1 Login to Risk Management Application

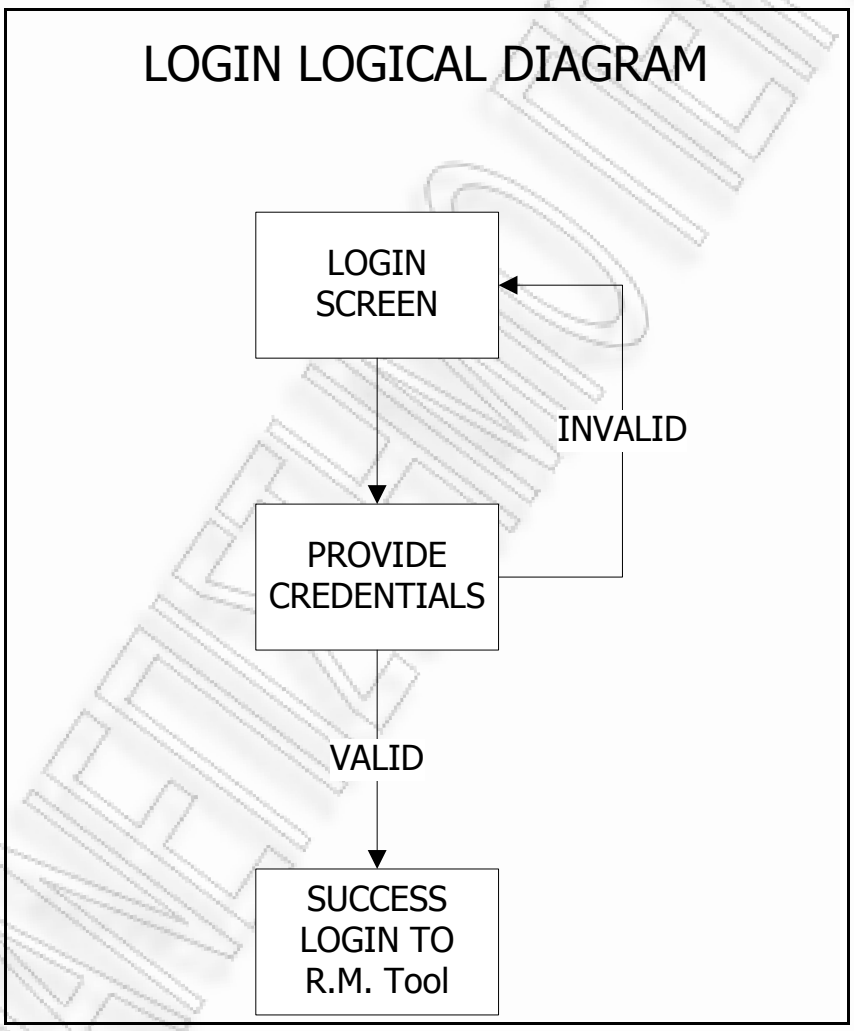


Figure 5.10 – Login Scenario

5.3.2.2 Submit a new Ticket (Defect or Suggestion)

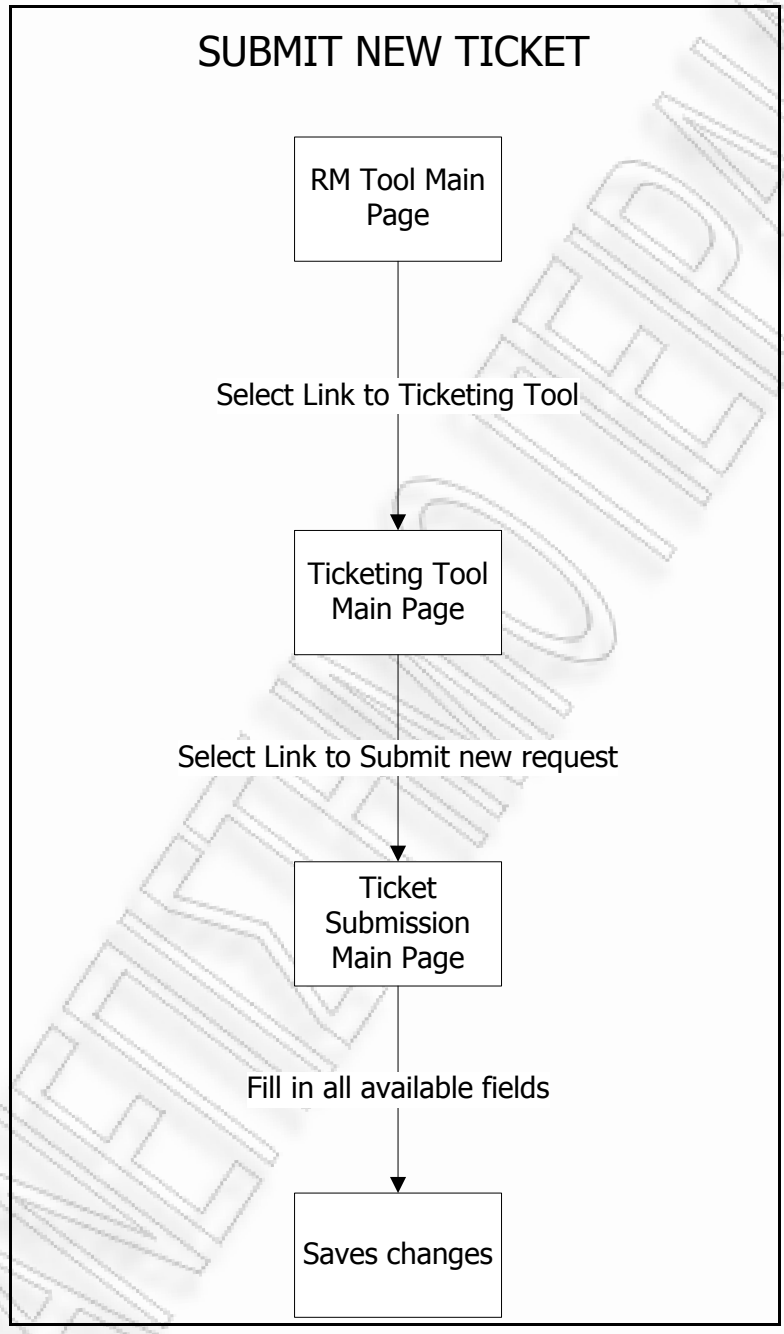


Figure 5.11 – New Ticket submission scenario

5.3.2.3 View submitted tickets

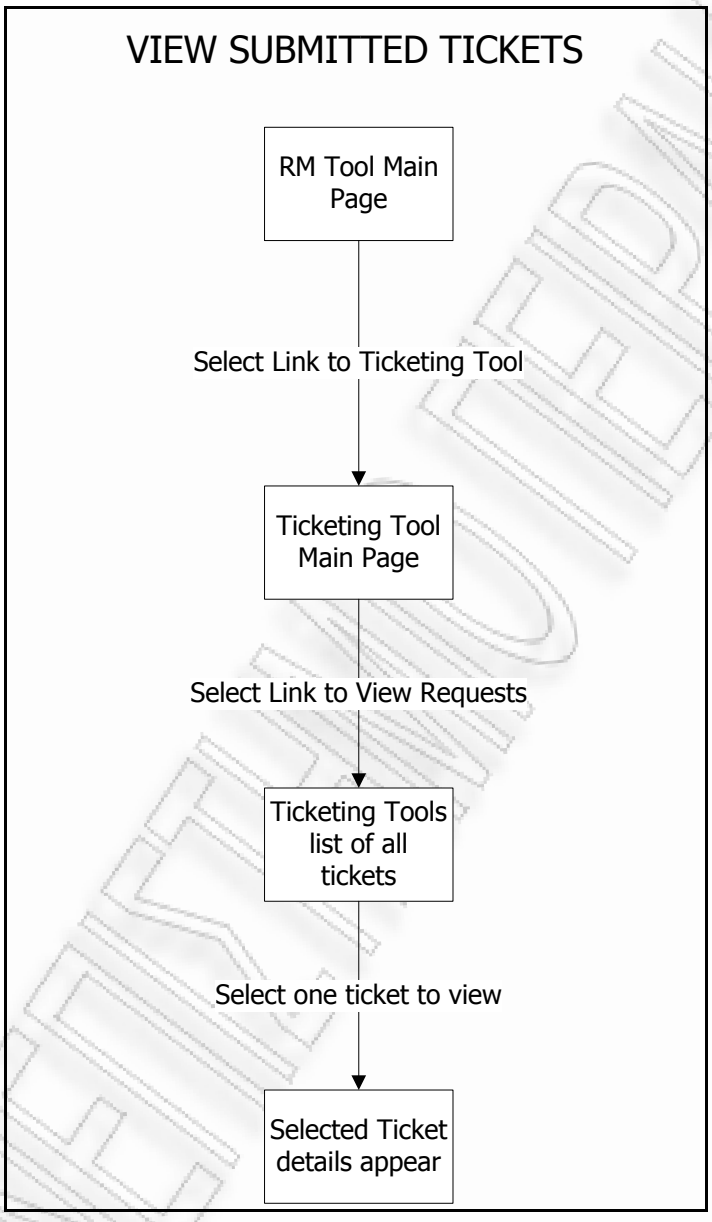


Figure 5.12 – View Submitted Tickets scenario

5.3.2.4 Submit a new Risk Management Ticket

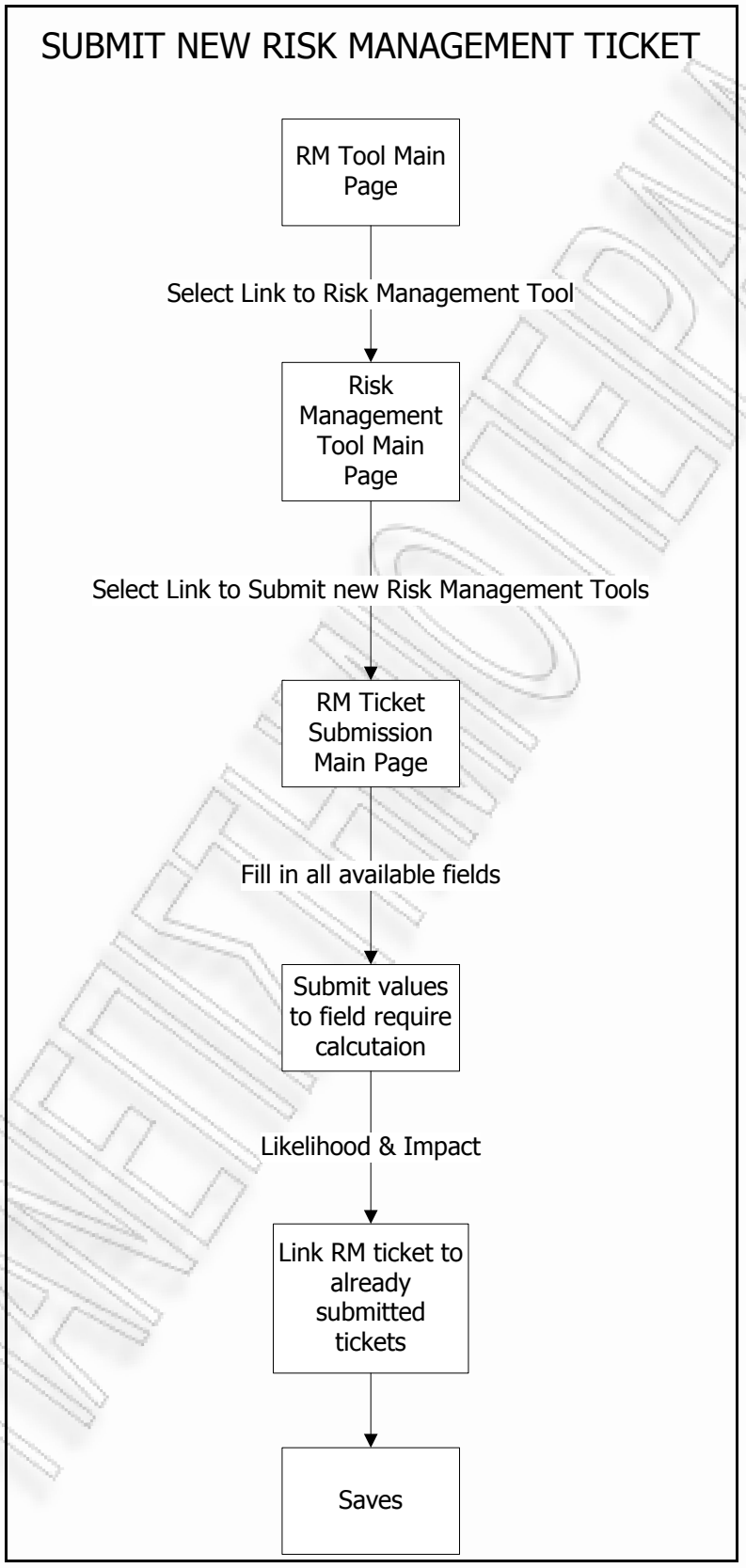


Figure 5.13 – Submit New RM Ticket scenario

5.3.2.5 View Risk Management Submitted tickets

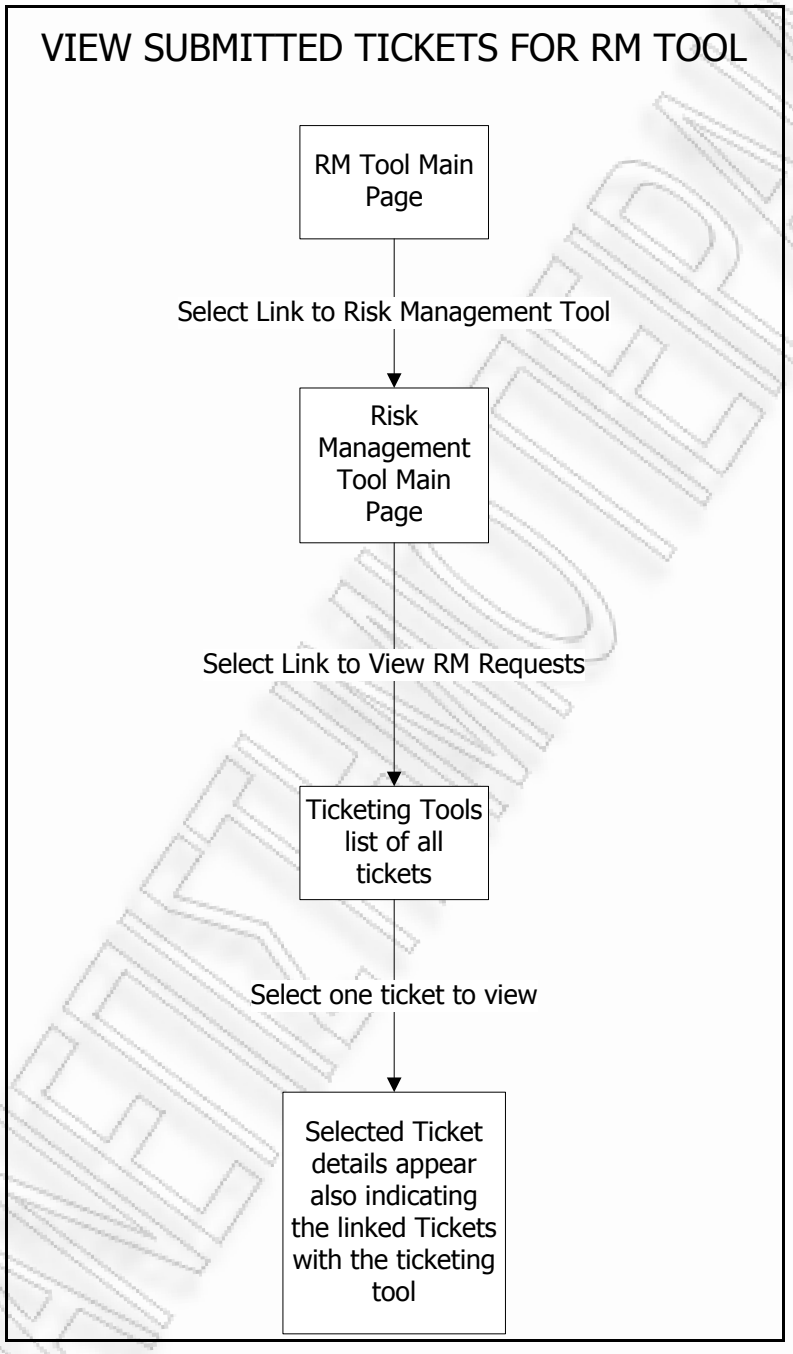


Figure 5.14 – View RM tickets Scenario

5.4 APPLICATION IMPLEMENTED

Upon the selection of the elements needed to implement the project, such as the application for development, for DBMS, the business scenarios and the desired structure of the application (as described in previous chapters) the application took the final form. Below is a sequence of print-screens that imprint and finally materialize the whole project described above.

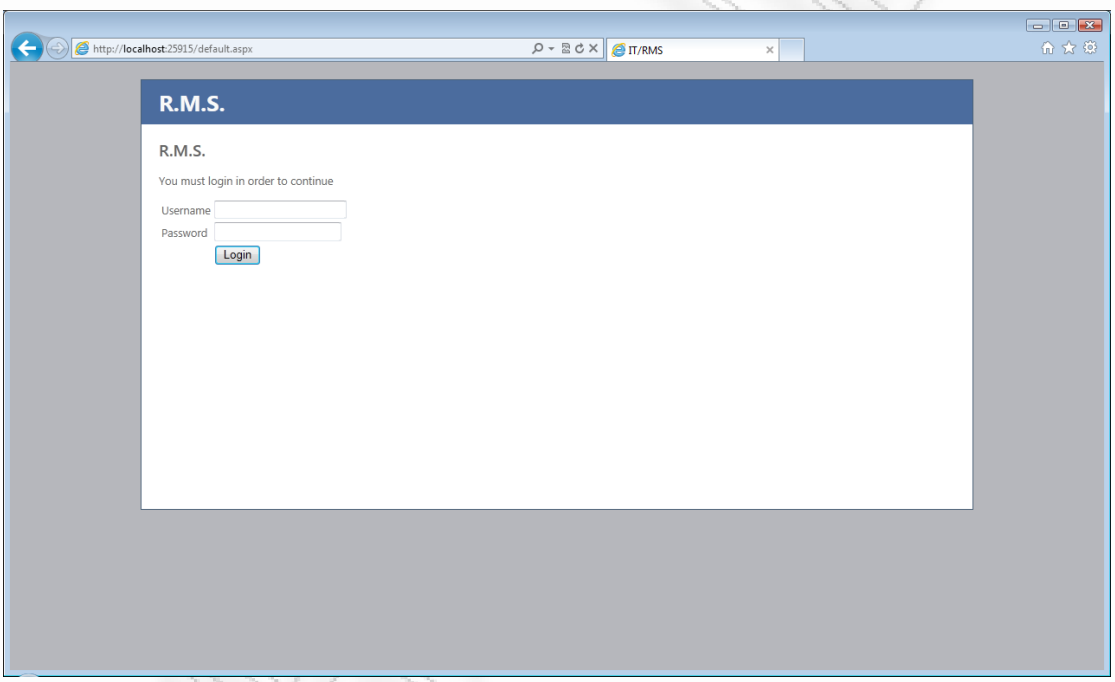


Figure 5.15 – Intro Screen

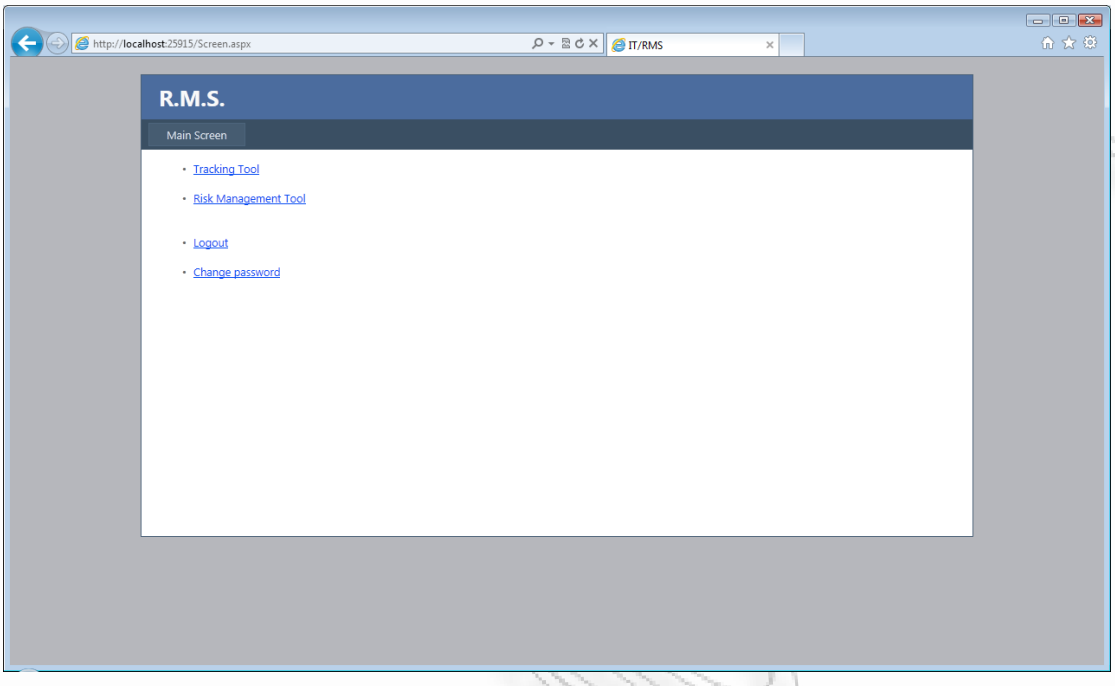


Figure 5.16 – Initial Selection Screen (Ticketing Tool, RM Tool, Change Password)

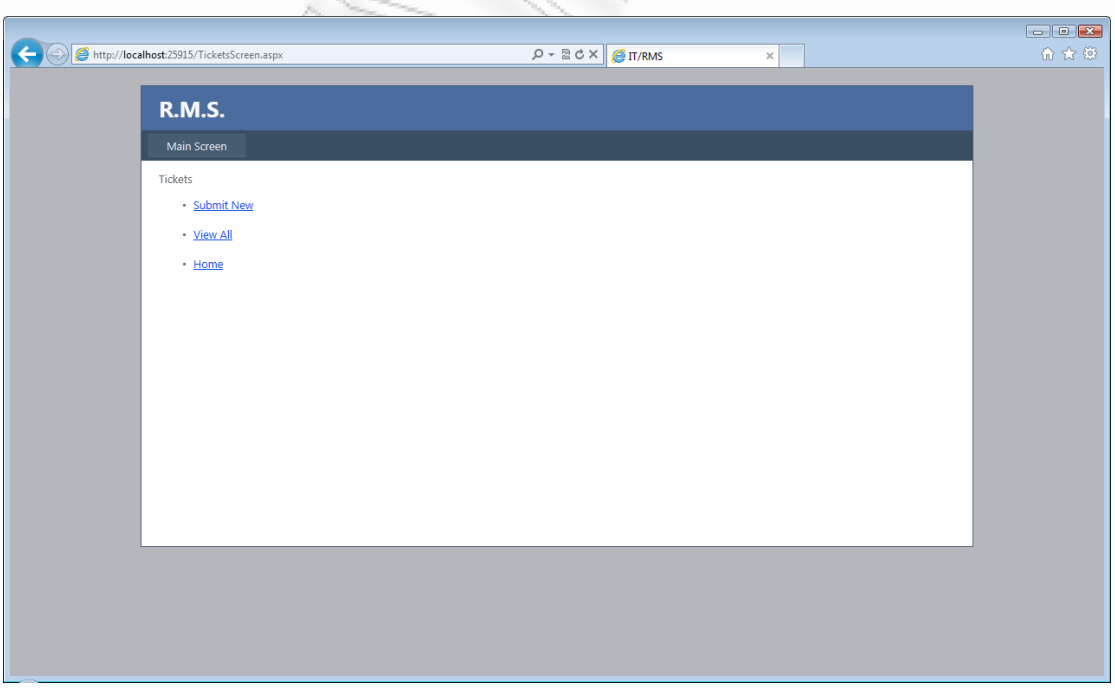


Figure 5.17 – Ticketing Tool initial screen

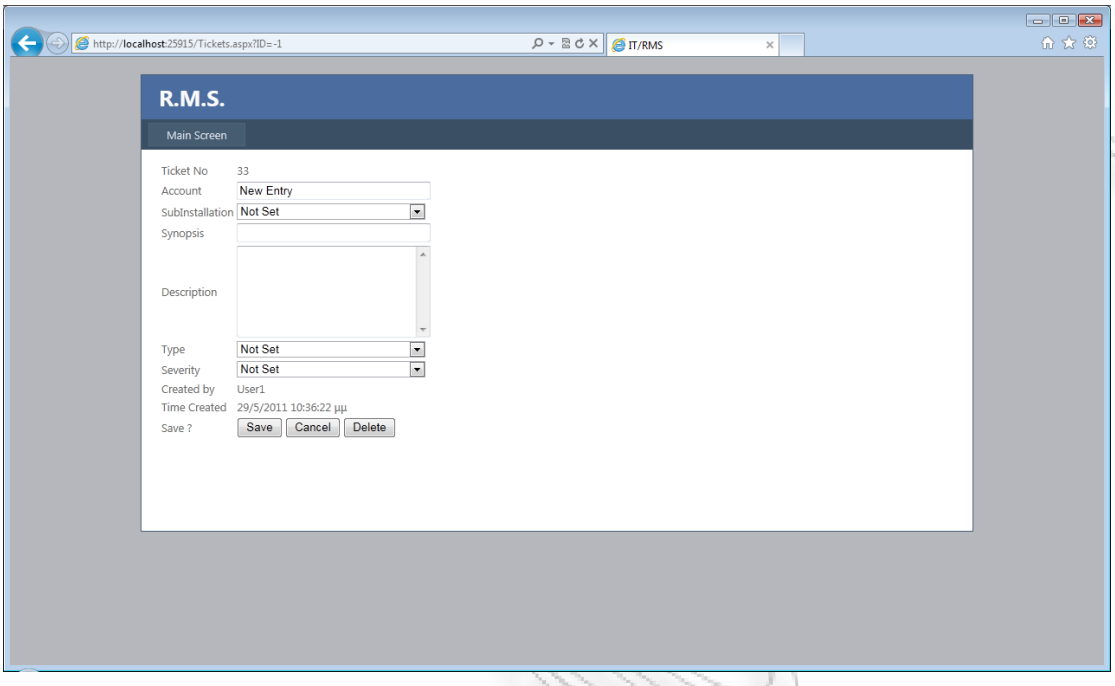


Figure 5.18 – New Ticket Submission Screen

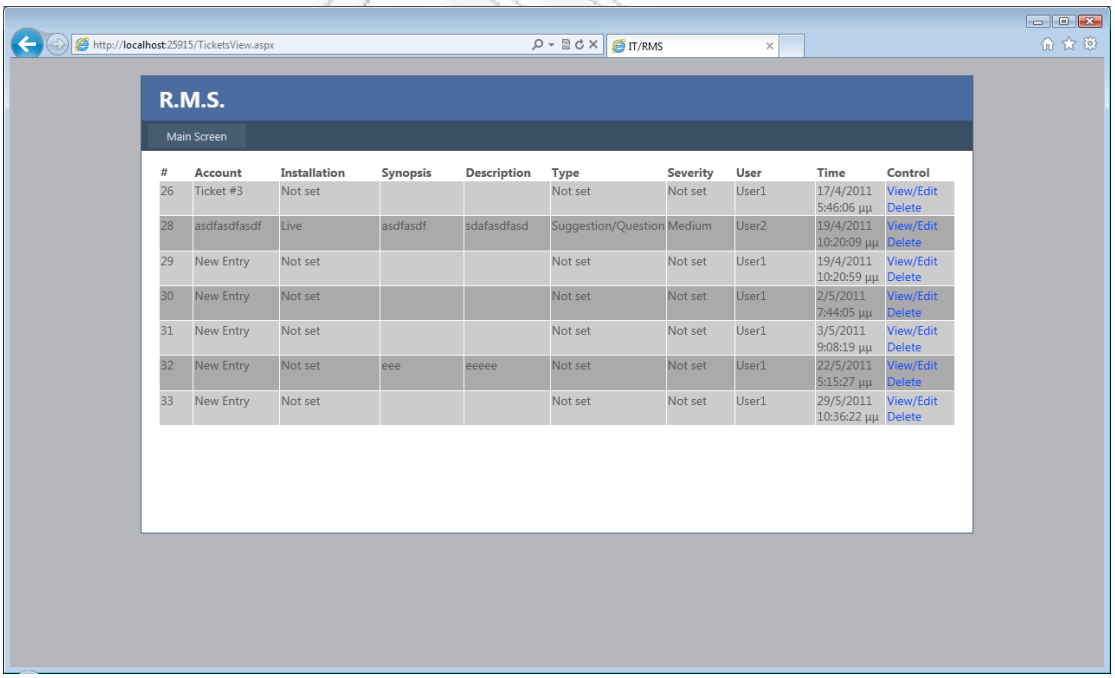


Figure 5.19 – View All Submitted Tickets

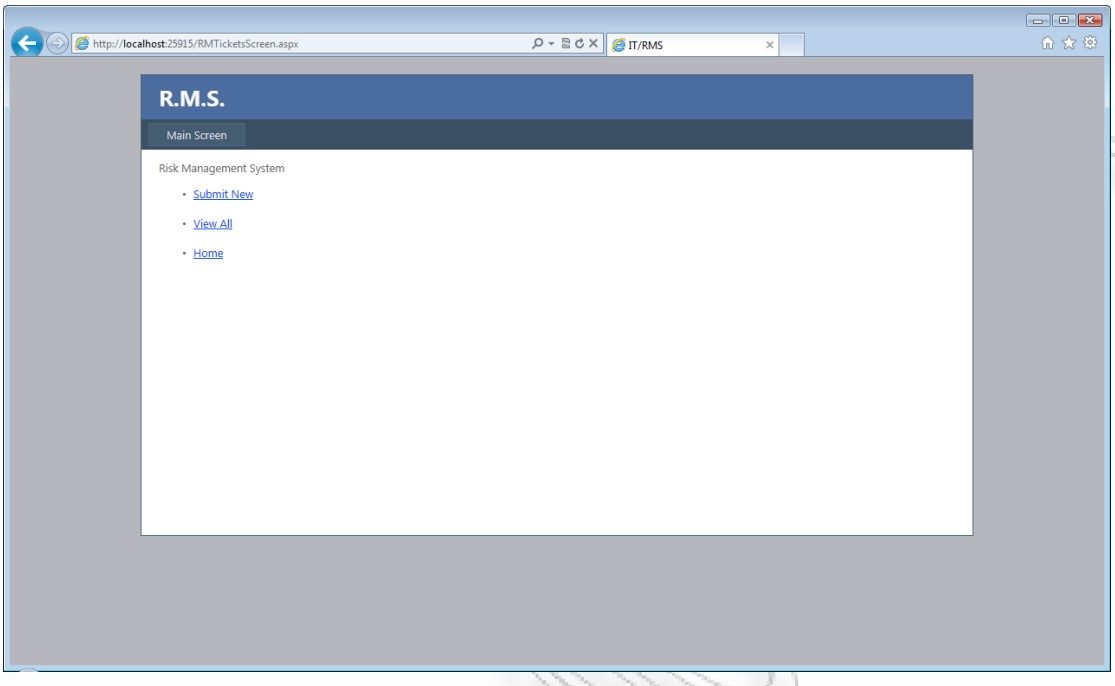
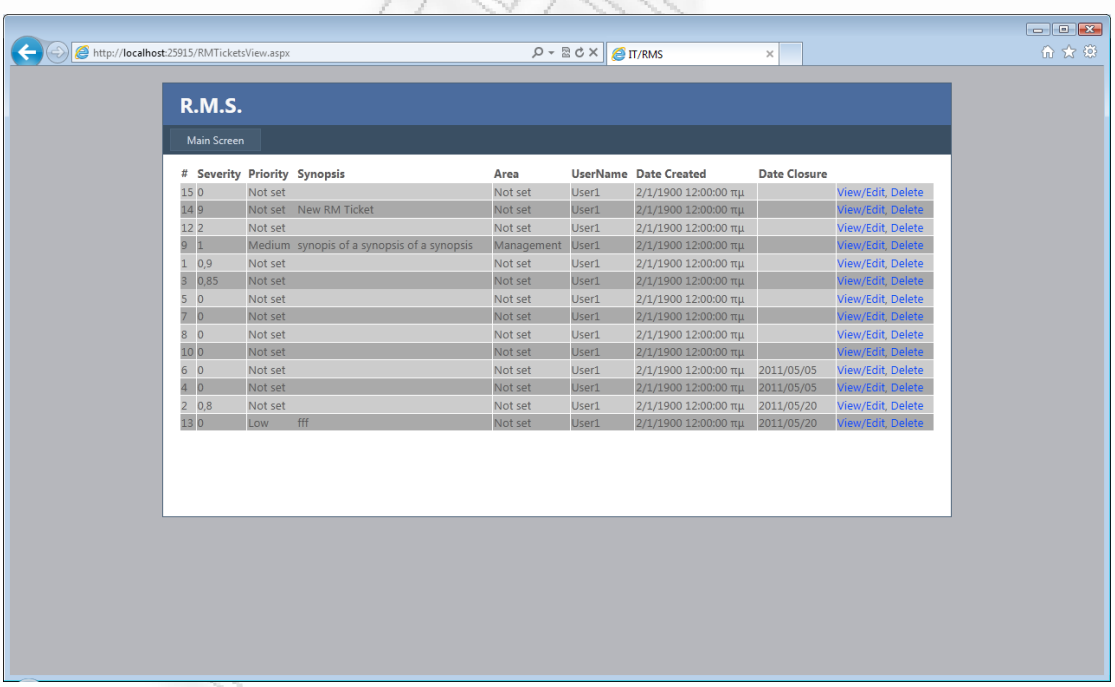


Figure 5.20 – Risk Management Tool initial screen



#	Severity	Priority	Synopsis	Area	UserName	Date Created	Date Closure	
15	0	Not set		Not set	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
14	9	Not set	New RM Ticket	Not set	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
12	2	Not set		Not set	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
9	1	Medium	synopis of a synopsis of a synopsis	Management	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
1	0,9	Not set		Not set	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
3	0,85	Not set		Not set	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
5	0	Not set		Not set	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
7	0	Not set		Not set	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
8	0	Not set		Not set	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
10	0	Not set		Not set	User1	2/1/1900 12:00:00 πμ		View/Edit, Delete
6	0	Not set		Not set	User1	2/1/1900 12:00:00 πμ	2011/05/05	View/Edit, Delete
4	0	Not set		Not set	User1	2/1/1900 12:00:00 πμ	2011/05/05	View/Edit, Delete
2	0,8	Not set		Not set	User1	2/1/1900 12:00:00 πμ	2011/05/20	View/Edit, Delete
13	0	Low	fff	Not set	User1	2/1/1900 12:00:00 πμ	2011/05/20	View/Edit, Delete

Figure 5.21 – View All Submitted RM Tickets

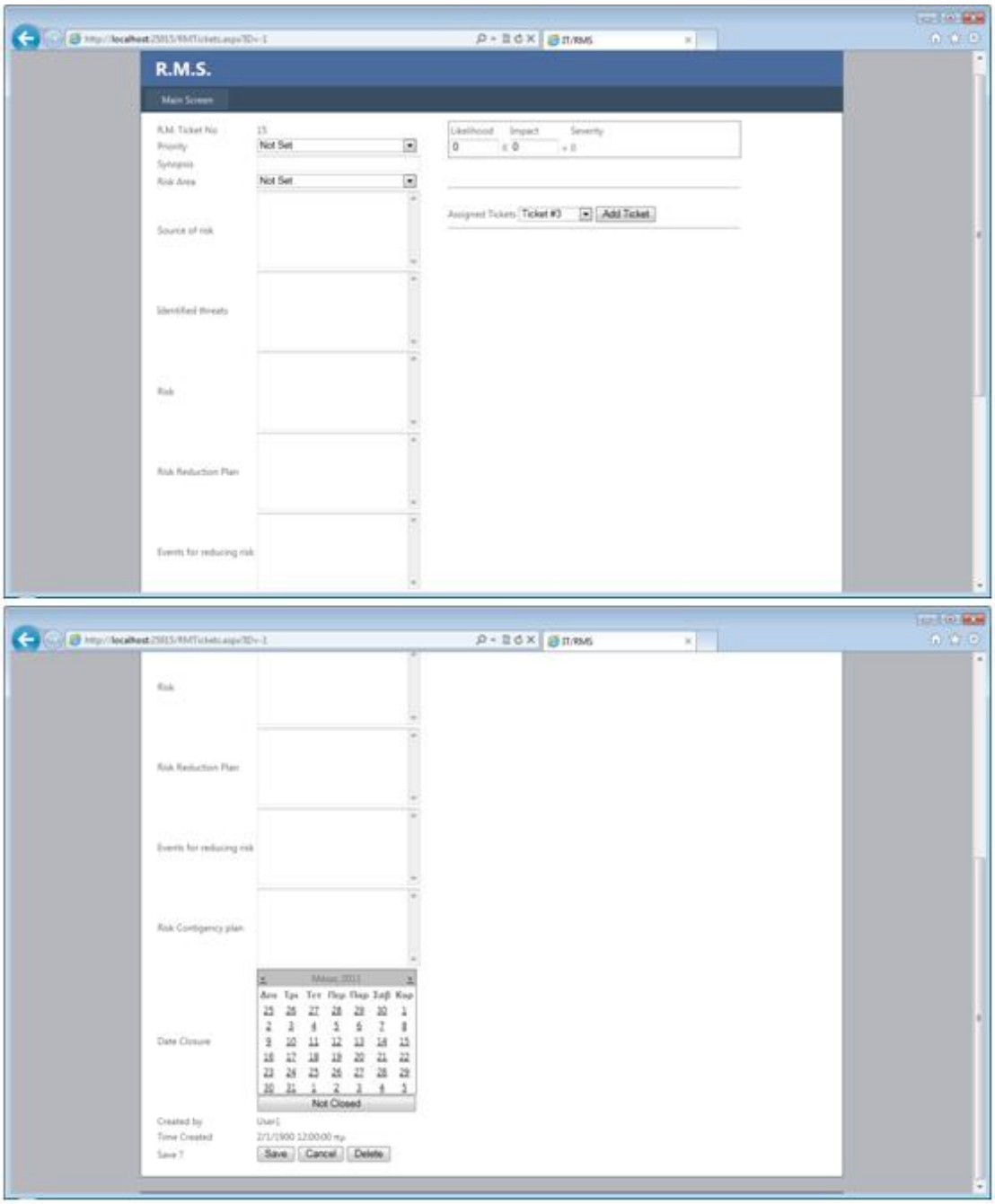


Figure 5.22 – Submit new RM ticket

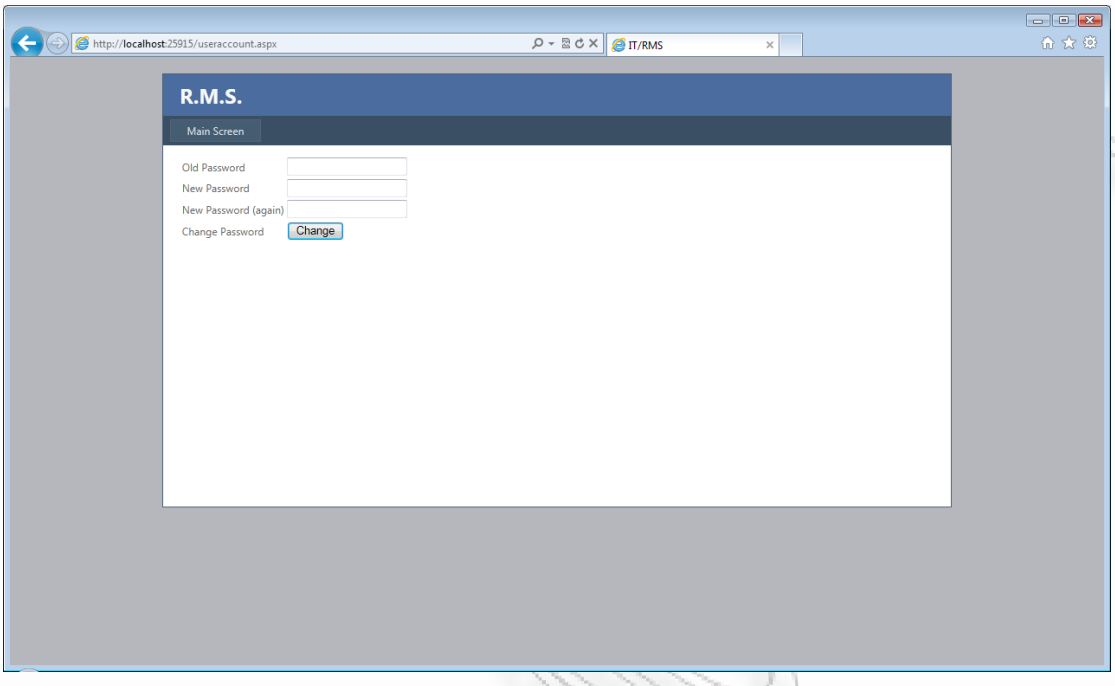


Figure 5.23 – Change Password

5.5 TESTING

Once the application is deployed the testing procedure as part of a software lifecycle initiated. Of course as realized the structure and complexity of the current application is relatively low, although testing should be performed in any case. Testing is never a waste of time and effort when it comes to software testing. Documented procedures must take place prior to actual testing of an application. Due to the nature of the project the testing was performed from a bias source, the developer of the project. Even so this is a procedure cannot be ignored or skipped. In order to establish procedures for testing relative test cases must be written down and performed.

First of all, for an application to operate is to try to replicate all basic business scenarios, as mentioned in **5.3.2** paragraph. Once these scenarios are implemented then the second phase of testing for defects can proceed. Once the business scenarios work, tester should actually try to implement scenarios eg submitting invalid values to fields and generally tries to follow non business scenarios bearing in

mind what can make the application to respond in an undesirable way. A plan should be prepared, including the test strategy and the objectives of testing. Entry criteria and Exit criteria are essential to further testing the application.

To test this web application black box testing was chosen as a method. Black-box testing is a method of software testing that tests the functionality of an application as opposed to its internal structures or workings. Specific knowledge of the application's code/internal structure and programming knowledge in general is not required. Test cases are built around specifications and requirements, i.e., what the application is supposed to do. It uses external descriptions of the software, including specifications, requirements, and design to derive test cases. These tests can be functional or non-functional, though usually functional. The test designer selects valid and invalid inputs and determines the correct output. There is no knowledge of the test object's internal structure.

Figure 5.24 – Black Box Testing concept

The advantages of black box testing include:

- The test is unbiased because the designer and the tester are independent of each other.
- The tester does not need knowledge of any specific programming languages.
- The test is done from the point of view of the user, not the designer.
- Test cases can be designed as soon as the specifications are complete.

The disadvantages of this type of testing include:

- The test can be redundant if the software designer has already run a test case.
- The test cases are difficult to design.

- Testing every possible input stream is unrealistic because it would take an inordinate amount of time; therefore, many program paths will go untested.

An example of a testing scenario is shown below to demonstrate how, one of the business functionalities is tested.

Title :	Testing business functionality - Ticket submittion
Revision History :	1
Author :	Petros
Reviewed by:	Petros
Application	Ticketing Tools
Functional Area:	Submit new Ticket
Client:	XXX
Tested By:	PPE
Tested On:	DD/MM/YYYY
Application Version:	1.XX
Description :	
A test scenario to help expoit business inconsistencies of the application	
Pre-Requisites/ Conditions to reproduce	
1. User exist in test environment	
Test Execution	

Step	Description	Expected Result/comments	Actual Result	Comments
1	User logs in Risk Management Application by providing correct username and password.	User succesfully logs in the application. User verifies that Main screen appear	P	
2	User selects hyperlink "Tracking Tool"	User verifies new screen appears to select between: - Submit new ticket - View All tickets - Home		
3	User Selects hyperlink "Submit new ticket"	Verifies new screen appears that shows all available fields as described in 3rd chapter of the documentation		
4	User fills in all available fields and presses "Submit"	Verifies that application has succesfully saved data by navigating to "View ALL ticket requests" of the Ticketing application		

Top

Figure 5.25 – Example of a Test Scenario

6. CONCLUSIONS

To measure risk, an organization needs sophisticated information systems. The information systems must combine data from various inputs in a structured way to estimate the aggregate risk of the organization. In this thesis we have outlined some of the organizations firms face, when deciding to set up such systems. Initially we had an overview of the literature most modern systems base their operation and logic. We have described risk management methodologies currently in use in the market, and shown how this methodology, business processes and information system design interact and complement each other to build a tool to be able to handle in a respectable magnitude the risen risks of an organization.

In summary, successful and effective risk management is the basis of successful and effective management of recourses. Due to the reality of limited resources and large amount of threats, a reasonable decision must be made concerning the allocation of resources to protect systems. Risk management practices allow the organization to protect information and business process and therefore the organization's profile and value. To ensure the maximum value of risk management, it must be consistent and repeatable, while focusing on measurable reductions in risk. Establishing and utilizing an effective, high quality risk management process and basing the information security activities of the organization on this process will lead to an effective information system application for an organization.

Completing this dissertation the conclusions extracted, reminded me that measuring risks is not an easy task. To measure risk for an information system is a difficult task to perform, this is due to the fact that risk data must be combined across all levels of an organization or project or an information system. Information Systems are facing a rising number of risks that a project manager or the upper management of an organization or a company must tackle. Although this dissertation introduces a tool to raise issues, link them and track risks altogether, there is always the possibility, due to their abstract nature, an additional effort from the stakeholders to be performed in order to keep up with all possible risks a system might face. This effort is not easily tracked or documented and therefore not

measured. All stakeholders facing risks during an information system operational period and most of these issues have to do with time constrain. Unfortunately, no tool is capable to deal with it. Additionally to time constrains there are issues that have to do with the limited ability to impose coordination across an organization. That issue alone raises new challenges and problems. The optimum way to overcome these problems is left to the relative stakeholders, and how well they are organized.

FURTHER ACTIONS

To further upgrade and improve the efficiency of the tools developed, a number of additional propositions can be implemented. A tool that totally handles and incorporates all organization procedures and interacts with relative subsystems can be developed on-demand. When handling risks information is critical. Below there are several propositions that could have been incorporated to increase the level of data gathered for an organization or company.

Therefore we can outline the following actions:

- Implement a module that tracks the availability of a system stakeholder. For example a feature that can track holiday periods for these stakeholders, so that manager of the system knows all the time of the realistic capacity he can base his effort on.
- Link in each submitted ticket, for all stakeholders, depending on their roles toward the system, their time spent on each issue risen. This will assesses the owner of the project (eg an Account Manager or Project Manager) to track if a possible risk may rise from over pricing services, or when too much time is spent to an issue that the initial planned budget was very low.
- Capability to distinguish various projects from one-another by introducing the notion/idea of installations and client differentiation for a certain project.
- The introduction of the functionality to add/upload various files (texts, images etc) to better describe the risen issue in case of a defect, or further clarify with a document a new functionality.

- Integration with an established, internal to the organization, Software Change and Configuration Management Tool. For further reporting capabilities
- Characterization of the tickets submitted as released based or not so that the client can further assessed and know on time, if he can get his fix waiting a new release or not.
- A discrete field that only internal production is aware of, such as internal comments to further optimize internal communication
- Historical data stored for each issue submitted
- Different roles for clients and project stakeholders with capability to add users from the UI
- A portal, providing forum-like capabilities so that stakeholders and clients would have the opportunity to communicate more direct and formal, without the use of mailing services, just using their web browser.

7. REFERENCES

Kerrie Holley, Dr. Ali Arsanjni (2011): "**100 SOA Questions Asked And Answered**", Pearson Education Inc.

Biplav Srivastava: "**A Planning-based Decision Support Tool for Software Project Management**" IBM India Research Laboratory

Channu Kambalayal: "**3-Tier Architecture**", pdf over internet

Cheng-Few Lee, Alice C. Lee, John Lee (2009): "**Handbook of Quantitative Finance and Risk Management**", Springer (Dec. 2009)

Michael Gibson (1997): "**Information Systems for Risk Management**", Federal Reserve Board

University of Virginia (2010): "**Information Technology Security Risk Management (ITS-RM)**", University of Virginia USA

Riccardo Rebonato: "**Theory and Practice of Model Risk Management, Quantitative Research Center of the Royal Bank of Scotland**", Oxford University

(August 2009): "**Information Technology Sector Baseline Risk Assessment**", Homeland Security

Casualty Actuarial Society (2003): "**Enterprise Risk Management**", Casualty Actuarial Society

Kim Heldman (2005): "**Project Management Professional, Study Guide**", Wlley Publishing Inc.

Suphat Sukamolson PhD.: "**Fundamentals of quantitative research**", Chulalongkorn University

Zahir Irani and Peter Love (2008): "**Evaluating Information Systems Public and Private Sector**", Elsevier

Ernst & Young: "**Managing Information Technology Risk**", Ernst & Young, Financial Services

Angappa Gunasekaran (2008): "**Techniques and Tools for the Design and Implementation of Enterprise Information Systems**", IGI Publishing

Bryan C.Smith, C.Ryan Clay (2008): "**SQL Server 2008 MDX**", Microsoft Press

Karli Watsonet (2006): "**Beginning Visual C#**", Wrox Press

Jayaram Krishnaswamy (2007): "**Beginners Guide to SQL Server Integration Services Using Visual Studio 2005**", Packt Publishing

8. APPENDIX – SOURCE CODE

8.1 Application Source Code

"DAL.cs" file

```
using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Data;

using System.Data.SqlClient;



namespace srm.DataUtils {

    public static class DAL {

        public static String ConnectionString = "";

        public static Boolean UseSingleConnection = false;



        public static SqlConnection Connection = null;



        public static SqlConnection GetConnection () {

            if (UseSingleConnection) {

                if (Connection == null) Connection = new SqlConnection(ConnectionString);

                return Connection;

            } else {

                return new SqlConnection(ConnectionString);
```

```
        }
    }



    // Insert
    public static Int32 ExecInsertQuery(SqlCommand aSqlCommand) {
        if (aSqlCommand.Connection == null) aSqlCommand.Connection = GetConnection();
        aSqlCommand.CommandText = aSqlCommand.CommandText + " SELECT CAST(scope_identity() AS int);";
        Int32 Result = -1;
        aSqlCommand.Connection.Open();
        try  {
            Result =  Int32.Parse(aSqlCommand.ExecuteScalar().ToString());
        } finally {
            aSqlCommand.Connection.Close();
        }
        return Result;
    }



    //Update / Delete (/ Insert without return value)
    public static void ExecNonQuery(SqlCommand aSqlCommand) {
        if (aSqlCommand.Connection == null) aSqlCommand.Connection = GetConnection();
        aSqlCommand.Connection.Open();
        try  {
            aSqlCommand.ExecuteNonQuery();
        } finally {
            aSqlCommand.Connection.Close();
        }
    }
```

```
public static DataTable ExecSelectTableQuery(SqlCommand aSqlCommand) {
    if (aSqlCommand.Connection == null) aSqlCommand.Connection = GetConnection();
    DataTable Table = new DataTable();
    SqlDataAdapter DataAdaptor = new SqlDataAdapter(aSqlCommand);
    DataAdaptor.Fill(Table);
    return Table;
}
/*
public static SqlDataReader ExecSelectReaderQuery(SqlCommand aSqlCommand) {
    if (aSqlCommand.Connection == null) aSqlCommand.Connection = GetConnection();
    aSqlCommand.Connection.Open();
    SqlDataReader Result = null;
    try {
        Result = aSqlCommand.ExecuteReader();
    } finally {
        aSqlCommand.Connection.Close();
    }
    return Result;
}
*/
public static Int32 ExecInsertQuery(String aQuery) {
    return ExecInsertQuery(new SqlCommand(aQuery));
}

public static void ExecNonQuery(String aQuery) {
    ExecNonQuery(new SqlCommand(aQuery));
}
```

```
        public static DataTable ExecSelectTableQuery(String aQuery) {
            return ExecSelectTableQuery(new SqlCommand(aQuery));
        }
        /*
        public static SqlDataReader ExecSelectReaderQuery(String aQuery) {
            return ExecSelectReaderQuery(new SqlCommand(aQuery));
        }
        */
    }
}
```

"Global.asax" file

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Security;
using System.Web.SessionState;

namespace petridis {
    public class Global : System.Web.HttpApplication {

        void Application_Start(object sender, EventArgs e) {
            // Code that runs on application startup
```

```
    }

    void Application_End(object sender, EventArgs e) {
        //  Code that runs on application shutdown

    }

    void Application_Error(object sender, EventArgs e) {
        // Code that runs when an unhandled error occurs

    }

    void Session_Start(object sender, EventArgs e) {
        // Code that runs when a new session is started
        Session["ConnectionString"] = "Data Source=PASXALITSA\\SQLEXPRESS;Initial Catalog=RMT;Integrated Security=True";
        //Session["ConnectionString"] = "Data Source=BUGS\\SQLEXPRESS;Initial Catalog=RMT;Integrated Security=True";
        srm.DataUtils.DAL.ConnectionString = Session["ConnectionString"].ToString();
        Session["UserID"] = -1;
        Session["UserName"] = "";
    }

    void Session_End(object sender, EventArgs e) {
        // Code that runs when a session ends.
        // Note: The Session_End event is raised only when the sessionstate mode
        // is set to InProc in the Web.config file. If session mode is set to StateServer
        // or SQLServer, the event is not raised.
```

```
    }


  }

}
```

"Default.aspx" file

```
<%@ Page Language="C#" MasterPageFile="~/Site.master" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="petridis.Default" %>


<asp:Content ID="HeaderContent" runat="server" ContentPlaceHolderID="HeadContent">

</asp:Content>

<asp:Content ID="BodyContent" runat="server" ContentPlaceHolderID="MainContent">

   <h2>

      R.M.S.

   </h2>

   <p>

      You must login in order to continue

   </p>

   <table>

    <tr> <td> Username </td> <td> <asp:TextBox runat="server" ID="tbUsername"> </asp:TextBox> </td> </tr>

    <tr> <td> Password </td> <td> <asp:TextBox TextMode="Password" runat="server" ID="tbPassword"> </asp:TextBox>  </td> </tr>

    <tr> <td> </td> <td> <asp:Button runat="server" ID="btnLogin" OnClick="btnLogin_OnClick" Text="Login"></asp:Button> </td> </tr>

   </table>


</asp:Content>
```

"RM Tickets.aspx" file

```
<%@ Page Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="RMTickets.aspx.cs" Inherits="petridis.RMTickets" %>


<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">
  <style type="text/css">
    .Invalid {
        color :Red;
        font-weight:bolder;
    }
  </style>
  <script type="text/javascript">
    function HandleNums() {
        var NumX = $("#MainContent_tbRiskLikelihood").val();
        var NumY = $("#MainContent_tbRiskImpact").val();
        var Num = NumX * NumY
        $("#NumsMult").text(Num);
        if (Num > 1) {
            $("#NumsMult").addClass('Invalid');
        } else {
            $("#NumsMult").removeClass('Invalid');
        }


    }


    function RMTickets_Init() {
        $("#MainContent_tbRiskLikelihood").bind('keyup', function() {HandleNums();} );
```

```
        $("#MainContent_tbRiskImpact").bind('keyup', function() {HandleNums();} );

        HandleNums();

    }


    $(document).ready(RMTickets_Init);



  </script>

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">

  <div style="float:left; width:400px;">

  <table>

  <tr> <td> R.M. Ticket No  </td> <td> <asp:Literal runat="server" ID="alID"> </asp:Literal>  </td> </tr>


  <tr> <td> Priority       </td> <td>

     <asp:DropDownList ID="ddlPriority" runat="server" Width="218px">

     <asp:ListItem Value="0">Not Set</asp:ListItem>

     <asp:ListItem Value="1">Low</asp:ListItem>

     <asp:ListItem Value="2">Medium</asp:ListItem>

     <asp:ListItem Value="3">High</asp:ListItem>

     </asp:DropDownList>

  </td> </tr>


  <tr>  <td>  Synopsis                    </td>  <td>  <asp:TextBox  runat="server"  ID="tbSynopsis"
Width="218px"></asp:TextBox>  </td> </tr>


  <tr> <td> Risk Area       </td> <td>

     <asp:DropDownList ID="ddlRiskArea" runat="server" Width="218px">
```

```
    <asp:ListItem Value="0">Not Set</asp:ListItem>

    <asp:ListItem Value="1">Management & Technical</asp:ListItem>

    <asp:ListItem Value="2">Management</asp:ListItem>

    <asp:ListItem Value="3">Technical</asp:ListItem>

    </asp:DropDownList>

 </td> </tr>


 <tr> <td> Source of risk          </td> <td> <asp:TextBox runat="server" ID="tbRiskSource" Width="218px"
Height="100px" TextMode="MultiLine"></asp:TextBox>  </td> </tr>

 <tr> <td> Identified threats          </td> <td> <asp:TextBox runat="server" ID="tbIdentifiedThreats"
Width="218px" Height="100px" TextMode="MultiLine"></asp:TextBox>  </td> </tr>

 <tr> <td> Risk                        </td> <td> <asp:TextBox runat="server" ID="tbRisk" Width="218px"
Height="100px" TextMode="MultiLine"></asp:TextBox>  </td> </tr>

 <tr> <td> Risk Reduction Plan         </td> <td> <asp:TextBox runat="server" ID="tbRiskReductionPlan"
Width="218px" Height="100px" TextMode="MultiLine"></asp:TextBox>  </td> </tr>

 <tr> <td> Events for reducing risk </td> <td> <asp:TextBox runat="server" ID="tbEventsForReducingRisk"
Width="218px" Height="100px" TextMode="MultiLine"></asp:TextBox>  </td> </tr>

 <tr> <td> Risk Contigency plan       </td> <td> <asp:TextBox runat="server" ID="tbRiskContigencyPlan"
Width="218px" Height="100px" TextMode="MultiLine"></asp:TextBox>  </td> </tr>


 <tr> <td> Date Closure          </td> <td>

  <asp:UpdatePanel runat="server">

    <ContentTemplate>

     <asp:Calendar              runat="server"              ID="clDateClosure"              Width="218px"
OnSelectionChanged="clDateClosure_OnSelectionChanged" ></asp:Calendar>

     <asp:Button runat="server" ID="btnClearDate" OnClick="btnClearDate_OnClick" Text="Not Closed"
Width="218px"> </asp:Button> <br/>

      <span style="display:none">

      <asp:Literal runat="server" ID="alDateClosure" > </asp:Literal>

      </span>

    </ContentTemplate>
```

```
  </asp:UpdatePanel>

 </td> </tr>



 <tr> <td> Created by       </td> <td> <asp:Literal runat="server" ID="alUserName"> </asp:Literal>  </td>
</tr>

 <tr> <td> Time Created    </td> <td> <asp:Literal runat="server" ID="alTime"> </asp:Literal>  </td> </tr>


 <tr> <td> Save ? </td> <td>

                <asp:Button     runat="server"     ID="btnSave"     OnClick="btnSave_OnClick"     Text="Save">
</asp:Button>

                <asp:Button          runat="server"          ID="btnCancel"          OnClick="btnCancel_OnClick"
Text="Cancel"></asp:Button>

                <asp:Button          runat="server"          ID="btnDelete"          OnClick="btnDelete_OnClick"
Text="Delete"></asp:Button>

               </td></tr>

 </table>

 </div>


 <div style="float:left; width:400px;">


  <div style="border:1px solid gray;">

   <table>

   <tr> <td> Likelihood </td> <td> </td> <td>Impact </td> <td> </td> <td>Severity </td> </tr>

   <tr>

    <td> <asp:TextBox runat="server" ID="tbRiskLikelihood" Width="60px" Text="0.0">  </asp:TextBox> </td>

    <td> X </td>

    <td><asp:TextBox runat="server" ID="tbRiskImpact"      Width="60px" Text="0.0"></asp:TextBox> </td>
<td> = </td> <td><span id="NumsMult"> </span> </td>

   </tr>
```

```
    </table>
  </div>

  <br/>
  <br/>
  <hr/>
  <br/>

  Assigned Tickets

  <asp:DropDownList runat="server" ID="ddlTickets" /> <asp:Button runat="server" ID="btnAssignTicket" OnClick="btnAssignTicket_OnClick" Text="Add Ticket"> </asp:Button>

  <hr/>

  <asp:Repeater runat="server" ID="arAssignedTickets">
    <ItemTemplate>
      <%#DataBinder.Eval(Container.DataItem, "Account") %> <asp:LinkButton runat="server" ID="lbRemove" OnClick="lbRemove_OnClick" CommandArgument='<%# DataBinder.Eval(Container.DataItem, "ID")%>' Text="Remove" > </asp:LinkButton> <br/>
    </ItemTemplate>
  </asp:Repeater>
 </div>

 <div style="clear:both;"/>
</asp:Content>
```

"RM Ticket Screen. aspx" file

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="RMTicketsScreen.aspx.cs" Inherits="petridis.RMTicketsScreen" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">

 Risk Management System

 <ul>

   <li><a href="/RMTickets.aspx?ID=-1">Submit New</a></li> <br/>

   <li><a href="/RMTicketsView.aspx">View All</a>   </li>  <br/>

   <li><a href="/Screen.aspx">Home</a></li>

 </ul>

</asp:Content>
```

"RM Tickets View.aspx" file

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true"
CodeBehind="RMTicketsView.aspx.cs" Inherits="petridis.RMTicketsView" %>

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">


   <table style="width:100%;" cellspacing="1px;" class="list" >

     <thead>

     <tr>

     <td> #                  </td>

      <td> Severity            </td>

      <td> Priority           </td>

      <td> Synopsis            </td>

      <td> Area                </td>

      <!--
```

```
      <td> Source of risk          </td>


      <td> Identified threats      </td>
      <td> Risk                   </td>
      <td> Risk Reduction Plan     </td>
      <td> Events for reducing risk </td>
      <td> Risk Contigency plan    </td>
      -->



      <td> UserName               </td>
      <td> Date Created           </td>
      <td> Date Closure           </td>
      <td>                       </td>
    </tr>
    </thead>
    <tbody>
     <asp:Repeater runat="server" ID="arRMTickets">
      <ItemTemplate>
      <tr>
       <td> <%# DataBinder.Eval(Container.DataItem, "ID") %>                                    </td>
       <td> <%# DataBinder.Eval(Container.DataItem, "Severity") %>                              </td>
       <td>    <%#    Priorities[ParseIntDef(DataBinder.Eval(Container.DataItem,    "PrioritiesFK"),    0)]    %>
</td>
       <td> <%# DataBinder.Eval(Container.DataItem, "Synopsis") %>                              </td>
       <td>    <%#    RiskAreas[ParseIntDef(DataBinder.Eval(Container.DataItem,    "RiskArea"),    0)]    %>
</td>
       <!--
       <td> <%# DataBinder.Eval(Container.DataItem, "RiskSource") %>                           </td>
```

```
        <td> <%# DataBinder.Eval(Container.DataItem, "IdentifiedThreats") %>                    </td>

        <td> <%# DataBinder.Eval(Container.DataItem, "Risk") %>                                 </td>

        <td> <%# DataBinder.Eval(Container.DataItem, "RiskReductionPlan") %>                    </td>


        <td>        <%#        DataBinder.Eval(Container.DataItem,        "EventsForReducingRisk")        %>
</td>

        <td> <%# DataBinder.Eval(Container.DataItem, "RiskContigencyPlan") %>                   </td>

        -->

        <td> <%# DataBinder.Eval(Container.DataItem, "UserName") %>                             </td>

        <td> <%# DataBinder.Eval(Container.DataItem, "Time") %>                                 </td>

        <td> <%# DataBinder.Eval(Container.DataItem, "DateClosure") %>                          </td>



        <td> <a href="RMTickets.aspx?ID=<%# DataBinder.Eval(Container.DataItem, "ID") %>">View/Edit</a>,

         <a      href="RMTickets.aspx?Action=delete&ID=<%#      DataBinder.Eval(Container.DataItem,      "ID")
%>">Delete</a></td>

       </tr>

       </ItemTemplate>

      </asp:Repeater>

   </tbody>

   </table>


</asp:Content>
```

"Site.Master" file

```
<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site.master.cs" Inherits="petridis.SiteMaster"
%>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">

<head runat="server">

    <title>IT/RMS</title>

    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.5.2/jquery.min.js" type="text/javascript"></script>


    <link href="~/Styles/Site.css" rel="stylesheet" type="text/css" />


    <style type="text/css">

        table.list > tbody > tr:nth-child(even) {background: #aaaaaa;}

        table.list > tbody > tr:nth-child(odd)  {background: #cccccc;}

        table.list > tbody > tr:hover           {background: #eeeeee;}

        table.list > thead {font-weight:bolder;}

        table.list {color : #595959;font-size:14px;letter-spacing:0px;}

        table.list a {text-decoration:none;}

        table.list a:hover {text-decoration:underline;}

        table.list > tbody> tr > td {vertical-align:top;}


        .fl {float:left;}

        .fc {float:right;}

    </style>


    <asp:ContentPlaceHolder ID="HeadContent" runat="server">

    </asp:ContentPlaceHolder>


</head>

<body>
```

```
<form runat="server">

<asp:ScriptManager runat="server"></asp:ScriptManager>

<div class="page">

  <div class="header">

    <div class="title">

      <h1>

        R.M.S.

      </h1>

    </div>

    <div class="clear hideSkiplink">

      <asp:Menu ID="NavigationMenu" runat="server" CssClass="menu" EnableViewState="false"
IncludeStyleBlock="false" Orientation="Horizontal">

        <Items>

          <asp:MenuItem NavigateUrl="~/Screen.aspx" Text="Main Screen"/>

        </Items>

      </asp:Menu>

    </div>

  </div>

  <div class="main">

    <asp:ContentPlaceHolder ID="MainContent" runat="server"/>

  </div>

  <div class="clear">

  </div>

</div>

<div class="footer">



</div>

</form>
```

```
</body>

</html>
```

"User Account.aspx" file

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site.Master" AutoEventWireup="true" CodeBehind="useraccount.aspx.cs" Inherits="petridis.useraccount" %>

<asp:Content ID="Content1" ContentPlaceHolderID="HeadContent" runat="server">

</asp:Content>



<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">

   <asp:literal runat="server" ID="alServerStatus"> </asp:literal>

   <table>

    <tr> <td> Old Password </td> <td> <asp:TextBox runat="server" ID="tbPassword" TextMode="Password"> </asp:TextBox> </td> </tr>

    <tr> <td> New Password </td> <td> <asp:TextBox runat="server" ID="tbNewPassword" TextMode="Password"> </asp:TextBox> </td> </tr>

    <tr> <td> New Password (again) </td> <td> <asp:TextBox runat="server" ID="tbNewPassword2" TextMode="Password"> </asp:TextBox> </td> </tr>

    <tr> <td> Change Password </td> <td> <asp:Button runat="server" ID="btnLogin" OnClick="btnChange_OnClick" Text="Change"></asp:Button> </td> </tr>

   </table>

</asp:Content>
```