

# ***Risk Mitigation, Monitoring, and Management Plan***

## **I. Table of Contents**

<b>I. TABLE OF CONTENTS.....</b>	<b>1</b>
<b>1.0 INTRODUCTION .....</b>	<b>2</b>
1.1 SCOPE AND INTENT OF RMMM ACTIVITIES .....	2
1.2 RISK MANAGEMENT ORGANIZATIONAL ROLE.....	2
<b>2.0 FUNCTIONAL DATA DESCRIPTION .....</b>	<b>ERROR! BOOKMARK NOT DEFINED.</b>
2.1 RISK TABLE.....	3
2.1.1 Description of Risk m.....	3
2.1.2 Probability and Impact for Risk m.....	4
<b>3.0 RISK MITIGATION, MONITORING AND MANAGEMENT .....</b>	<b>6</b>
3.1 RISK MITIGATION FOR RISK M.....	6
3.1.1 Product Size .....	6
3.1.2 Business Impact.....	6
3.1.3 Customer (User) Risks .....	7
3.1.4 Process Risks.....	7
3.1.5 Technology Risks.....	7
3.1.6 Development Risks.....	7
3.1.7 Employee Risks (Teammates).....	8
3.2 RISK MONITORING FOR RISK M.....	8
3.2.1 Product Size .....	8
3.2.2 Business Impact.....	8
3.2.3 Customer (User) Risks .....	9
3.2.4 Process Risks.....	9
3.2.5 Technology Risks.....	9
3.2.6 Development Risks.....	9
3.2.7 Employee Risks (Teammates).....	9
3.3 RISK MANAGEMENT FOR RISK M .....	10
3.3.1 Product Size .....	10
3.3.2 Business Impact.....	10
3.3.3 Customer (User) Risks .....	10
3.3.4 Process Risks.....	10
3.3.5 Technology Risks.....	11
3.3.6 Development Risks.....	11
3.3.7 Employee Risks (Teammates).....	11
<b>4.0 SPECIAL CONDITIONS.....</b>	<b>11</b>



## 1.0 Introduction

This section gives a general overview of the Risk Mitigation, Monitoring and Management Plan for the **Waste Management Inspection Tracking System (WMITS)**.

Comment: The first sections are repeated from the Software Project Plan. This is acceptable, but they can also be included by reference.

### 1.1 Scope and intent of RMMM activities

We want the software to be free of any defects or errors, but it is hard or at times almost impossible to develop a system that is free of any defects. To be safe we would like to have a risk management plan to counter any difficulties that may impact the development or the creation of the software. Our goal is to assist the project team in developing a strategy to deal with any risk. For this we will take a look at the Possible risks, how to monitor them and how to manage the risk.

For software development to avoid any risk both the developer and client have to work together. Client has to spend time with the developer in the beginning phase of the software development. If client decides to change the software, meaning if client wants to add some more functions into the software or to change the requirement, this will have major effect on the development of the software.

### 1.2 Risk management organizational role.

Every one associated with the software has responsibility of managing the risk. That is if everyone participated and paid close attention to all the details during the early phase of the software development many risk can be avoided.

- Software development can avoid having risk by double-checking their schedule, product size, estimates regarding costs of the development etc.
- Customer can help avoid risk by providing all necessary software information during the early phase of the development.
- Software development team can avoid risk by getting all the details of the equipment that are provided or are accessible to them.
- Client can avoid risk by making all necessary business changes before initiating request for the software.

## **2.0 Risk Description**

This section describes the risks that are likely to be encountered during this project.

### **2.1 Risk Table**

The following table describes the risks associated with the project. The appropriate categories of the risks are also given, as well as probability of each risk and its impact on the development process.

#### **2.1.1 Description of Risk m**

##### **Business Impact Risk:**

This is the risk where concern is that of the not being able to come up or produce the product that has impact on the client's business. If the software produced does not achieve its goals or if it fails to help the business of clients improve in special ways, the software development basically fails.

##### **Customer Risks:**

This is the risk where concern is client's motivation or willingness in helping the software development team. If the client fails to attend meeting regularly and fails to describe the real need of the business the produces software will not be one that helps the business.

##### **Development Risks:**

If client fails to provide al the necessary equipment for the development and execution of the software this will cause the software to become a failure. So in other words customer has to be able to provide time and resources for the software development team. If all the requested resources are not provided to the software development team odds for the software development to fail rises greatly.

##### **Employee Risk:**

This risk is totally dependent on the ability, experience and willingness of the software development team members to create the working product. If the team members are not experience enough to use the application necessary to develop the software it will keep pushing the development dates until it's too late to save the project. If one or more members of the software development team are not putting in all the effort required to finish the project it will cause the project to fail. Employee risk is one of the major risk to consider while designing the software.

##### **Process Risks:**



Process risk involves risks regarding product quality. If the product developed does not meet the standards set by the customer or the development team it is a failure. This can happen because of the customer's failure to describe the true business need or the failure of the software development team to understand the project and then to use proper equipment and employees to finish the project.

#### Product Size:

This risk involves misjudgment on behalf of the customer and also the software development team. If the customer fails to provide the proper size of the product that is to be developed it will cause major problems for the completion of the project. If software development team misjudges the size and scope of the project, team may be too small or large for the project thus spending too much money on project or not finishing project at all because of shortage of finances.

#### Technology Risk:

Technology risk involves of using technology that already is or is soon to be obsolete in development of the software. Such software will only be functional for short period of time thus taking away resources from the customer. Since the technology changes rapidly these days it is important to pay importance to this risk. If customer request use of software that soon to be obsolete software development team must argue the call and have to pursue customer to keep-up with current technology.

### 2.1.2 Probability and Impact for Risk m

The following is the sorted version of the above table by probability and impact:

Category	Risks	Probability	Impact
Employee Risks	Lack of training and experience	40%	1
Process Risk	Low product quality	35%	1
Product Size	Where size estimates could be wrong	30%	2
Development Risks	Insufficient resources	30%	2
Customer Risk	Customer may fail to participate	20%	3
Technology Risk	Obsolete technology	10%	2
Business Impact	Product may harm the business	10%	3

**Table - Risks Table (sorted)**



<u>Impact Values</u>	<u>Description</u>
1	Catastrophic
2	Critical
3	Marginal
4	Negligible

Above is the table that categorizes the risks involved in software development. It gives brief description of the risk in Risks column and also provides the probability of risk occurring in percentages in Probability column and also the impact of the risk in the Impact column.

The impacts values assigned to the each risk are described in the section below the risk table. It is very convenient way to look at the risk and derive the information of the risk.



## 3.0 Risk Mitigation, Monitoring and Management

This section in detail describes Risk Mitigation, Monitoring and Management for each of the possible risks. It will talk about ways to avoid, monitor and to have ways to manage the risks.

### 3.1 Risk Mitigation for Risk *m*

In this section several different software development risks will be identified, a plan will be created to avoid these risks. We will think of the risks possible and of the way to keep the software development process from encountering these risks. It is important to have mitigation plan to avoid risks once and for all. Goal is to attack the risk even before it comes into existence. The plan will help in identifying the possible risks and to monitor them.

#### 3.1.1 Product Size

In this risk concern is of under or overestimation (mainly underestimation) of the number of Function Points. If we estimate too few LOC (line of code) necessary for the project we may get wrong cost figures which can prove fatal to the software development plan. To avoid this from happening we will use conservative figures to reduce the probability of the risk. This means we will overestimate the LOC a little. If we end up finishing the project earlier than that will not create any troubles. If the software cost estimations are passed with higher than actual cost required to deliver the product, the software development team can get credited for finishing the product sooner. Also for any reason the delivery dates get pushed back development team can still deliver the product on the time. In normal circumstances companies are not picky about the product size, so increasing the number will not cause any troubles in getting approval for the project.

**Critique:** Although this strategy may be reasonable in this context, I'm not in favor of it. It is not possible to avoid the risk of underestimation. Rather, your best estimates should be submitted with the proviso that (1) changes will cause a re-evaluation of estimates; (2) the estimates will be revisited at regular (short) intervals and readjusted if new information surfaces; (3) management provides an indication that it understands that the estimate are subject to change as the project progresses.

#### 3.1.2 Business Impact

In this risk category we are concerned about the quality of the final product. As mitigation step we will spend more time with the users to understand their needs. This way we can gather all the information necessary for the project to be successful. We will try to understand business environment and can try to provide the user with help in defining software requirements. More the time team spends with the customer better the understating the team will have regarding the software. This will help in coming up with

just right product at right price for the customer. Team has to make sure that the palm size PC integration and the cost of the palm size PC is justified, meaning that it really improves inspection process.

### **3.1.3 Customer (User) Risks**

If the users of the product fail to participate during the different phases of the software development we fail to recognize problems with the software. To avoid this in the mitigation phase we will try to meet the customer frequently and present software in phases so that customer and we can have better understanding the software being developed. More the time team spends with the customer better the understating the team will have regarding the software. This will help in coming up with just right product at right price for the customer. If customer fails to mention some special operations that have to performed with totally separate checklist and have to be stored separately, software development team does not know anything about it thus leaving big problem in the software.

Comment: The approach noted above is reasonable, but it is essential to get a written commitment from customer management that indicates that customer staff will be tasked with liaison with software developers. It's also necessary to inform customer management immediately if the liaison does not occur.

### **3.1.4 Process Risks**

We want quality of the product to be as high as possible. To achieve this we will set up guidelines to be followed for each of the team members during all the phases of the software development cycle. The standard will be set and defined for all of the software development. This will help the team in delivering the high quality product thus increasing our reputation in the market. This will help bring in more clients in the future. It will also save customer from getting low quality product. For example, palm size PC checklists are easy to get used too. If inspector can not get used to the forms in the PC they may go back to using pen and paper which is not good for the reputation of the team.

### **3.1.5 Technology Risks**

To avoid risk of using technology that may become obsolete in few years after the product have been developed. We will do excessive research on what technology to use for software development and will use the latest technology (programming languages etc.) to avoid this risk. Software development team has to make sure that the equipment requested (i.e. Palm-size PC) are current and will not be obsolete in near future.

### **3.1.6 Development Risks**

If the necessary tools are not provided to all of the team members, their work will lack quantity and quality. As mitigation phase we will make sure that the budget includes cost for latest technology and tools needed to achieve the desired product. For example if the



government refused to deliver the Palm-size PC to the DEQ the PC integration part will be useless until the units are actually bought.

### 3.1.7 Employee Risks (Team members)

This risk concerns the knowledge and of the employees and their willingness to help make the project succeed. As mitigation step of this risk we will make sure that some one in all of the project development phases knows exactly what to do and the tools to use to achieve the goals. If the employees that have little knowledge in the main software implementation language fails to learn it, it may cause big problems when coding part begins.

## 3.2 Risk Monitoring for Risk *m*

In this section we will identify the conditions to monitor to determine whether risk *m* is becoming more or less likely.

### 3.2.1 Product Size

To monitor the risk here, we will keep track of the amount of functions necessary for the program throughout the entire development cycle. This will tell us if the project may come across risk in future. We will keep track of the Visual Basic code produced during the coding. We will also keep track of the code required for the integration of the Palm-size PC. We will also be concerned about the size of the database. The number of table and number of tuple in each table. All of the above will help us in monitoring the size of the product.

**Comment:** The approach noted above is worthwhile, but more information should be provided. Specifically, where in the development cycle will a reevaluation of product size be done? What mechanism will be used to effect the reevaluation (e.g., a spreadsheet model)? What happens if product size appears to be increasing?

### 3.2.2 Business Impact

As monitoring step in this risk we will setup user meetings to show them the work that has been completed and to get user input on the work. We will have meetings every other week to present the work that has been done from the time of the last meeting. This will help team in staying in touch with the customers and will also be very efficient way to derive customer's input on the progress made. It will also be a way to get customer insight on the project, which will help us determine the changes that we may have to make to the software upon customers request.

**Comment:** The approach noted above is worthwhile, but additional steps might be taken. A 2 week gap in evaluation may be too large. Possibly, interim work products





might be posted on a Web site for evaluation by customers on a more frequent basis. This would not require a meeting, but would require a time commitment by the customer.

### **3.2.3 Customer (User) Risks**

To monitor this risk we will monitor the successes of the meeting by keeping track of people that have attended the meeting. If the out come of the meeting is low we can contact responsible person to help us overcome this problem. We will also have the login charts to show the customer who is attending the meetings and who needs to be reminded to start attending meetings.

### **3.2.4 Process Risks**

To monitor the risk here we will review each other's work to find the problems and to help each other in achieving better product quality. We will also have the general guidelines set for all of the work to be carried on for the software development. Software development team will constantly check each others work; will compare it with the set guidelines, and will inform a team member who is failing to participate in following the guidelines.

### **3.2.5 Technology Risks**

For monitoring phase during the development of the software we will keep eye on new technology. This will help us to keep up with new technology. For example plan will be drawn to use multiple set of the Palm-size PC so team can see if the different brand PC with same operation system is capable of carrying the task needed to perform checklist operations.

### **3.2.6 Development Risks**

For monitoring phase during the development of the software we will keep eye on tools being used and their effectiveness. This will help us to keep up with new technology. We will keep tack of the new equipment being brought into use at DEQ. We will also look for the availability of the inexpensive equipment that may provide help in completion of this project.

Critique: This discussion is vague and should be corrected. Specifically what development risks are likely to be encountered and specifically what can be monitored to determine whether they are becoming more or less likely?

### **3.2.7 Employee Risks (Teammates)**

Monitoring and managing of this risk will include looking out for each other, that is if some team-member is having difficulties in performing some tasks or using particular tool or technique other members of that team will help him out. This is where team members may have to spend little time with each other learning or teaching what others know.



### 3.3 Risk Management for Risk *m*

In this section we will identify several different software development risks and will try to create a plan to manage these risks if they do occur.

#### 3.3.1 Product Size

After careful monitoring of the process, if we still end up with underestimation of the FP, we will put more man-hours into the project. This is the only way that we think we can manage the risk.

Comment: There are other options, if staffing is inflexible. The team could deliver less functionality or opt to deliver in an incremental fashion.

#### 3.3.2 Business Impact

If a mistake has been made, user input on the completed work will provide us with information to fix or improve the software. We have done very many meeting with the clients and plan to do meeting every two weeks; this should clear any misunderstandings between the software development team and customers. This is the best way to go at since the work that is done on the project is revealed during the meetings and customer gets chance to make adjustments necessary.

#### 3.3.3 Customer (User) Risks

If the turn out in the meetings is not encouraging we will pass out questionnaire to easily gather customers' view. We will ask them question rather than waiting for them to ask us questions. We will also talk to the manager at the DEQ to help us come up with plan that will increase the attendance during the meeting. If the outcomes of the meetings is satisfactory there should not be any major difficulties regarding customer risks.

Comment: See earlier comment about the possibility of posting interim work products on a Web site for evaluation by customers on a more frequent basis.

#### 3.3.4 Process Risks

If the problem exists with the quality of the work, the quality assurance plan will be revised in the risk management phase. Other team member will attempt to take over or swap the work of the member whose work does not meet the quality standards.

### **3.3.5 Technology Risks**

For monitoring phase during the development of the software we will keep eye on new technology. If we spot new techniques that can be implement without major changes in our project we will include such techniques in the development of the project. We will also keep a look out for major shifts in the technology and how it affects the software that we are working on. If there is a need change in the technology will be discussed among team members and will be presented to the client. If client agrees necessary changes will be made with the existing technology.

### **3.3.6 Development Risks**

In the management phase if the funding for the technology and tools are not enough we will have to reschedule the phases of software development cycle to allow more time to coding phase. We will provide information on the several different Palm-size PC's and will let the customer to choose the one that is most appropriate for the customer to buy. We will also make sure that the equipment is allowed to be purchased under government controls and contracts.

### **3.3.7 Employee Risks (Team members)**

Monitoring and managing of this risk will include looking out for each other, that is if some team-member is having difficulties in performing some tasks or using particular tool or technique other members of that team will help him out. This is where team members may have to spend little time with each learning or teaching what others know. If team member lacks ability to use certain programming language or application, other team members will take some time off to teach the team member basics related to that application.

## **4.0 Special Conditions**

Special conditions that are associated with the software are as follows.

- **Use of the Palm-size PC:**  
We need to make sure that all the inspectors at the facility are comfortable with the use of the PC. It is hard at times to write certain characters since the PC has its own writing software. Use of the software that recognizes the handwriting of different personal will be used. It is necessary to provide training for the entire group of inspectors to use the writing software.
- **Saving Check-lists:**  
We need to make sure that as an inspector goes through a facility to be inspected using the PC, he or she does not have to save each and every entry made in the



Palm-size PC. The PC will automatically save the data after every entry is made or change has occurred. Save button will be used only to finalize the product.

- Login:  
Since we are using modular login we need to make sure that the person logged in will only have access to certain part of the application, this depend on the rights granted to the users. We have to explain to each user why he or she is not able to use certain parts of the applications. We also need to make sure that users with read only write understand why they are unable to make changes to other users reports.

Comment: Although these special conditions seem important, they do not seem appropriate for inclusion as part of the RMMM document. They would be better placed in the requirements specification.