**Risk Analysis**

**and Management**

**Plan**

Version 1.03

(18/04/2015)

**Table of contents**

1. Executive Summary (page 3)

2. Introduction (page 3)

3. Risks (page 4)

3.1. Loss of Team Members (page 4)

3.2. Loss of Access to Technology (page 4)

3.3. Loss of Documentation/Code (page 5)

3.4. Loss of Client (page 5)

3.5. Loss of Funding (page 5-6)

3.6. Suspension of the Project (page 6)

3.7. Changing of Requirements (page 6-7)

3.8. Changing of Clients (page 7)

3.9. Maintenance (page 8)

3.10. Over-Evolution of the System (page 8-9)

4. References (page 10)

**1. Executive Summary:**

The purpose of this document is to provide a management framework to ensure that levels of risk and uncertainty are properly managed for the remainder of the project. As risk management is an ongoing process over the life of a project, the Risk Register must be considered a ‘snap shot’ of relevant risks at one point in time.

This document will achieve this by defining the following:

- The process that will be/has been adopted by the project to identify, analyse and evaluate risks during the remainder of the project.

- How risk mitigation strategies will be developed and deployed to reduce the likelihood and/or impact of risks.

- How often risks will be reviewed, the process for review and who will be involved.

- Roles and responsibilities for risk management.

- How reporting on risk status and changes to risk status will be undertaken within the project and to the Steering Committee.

- A complete risk register, containing all risks identified for the project, their current gradings and the identified risk mitigation strategies to reduce the likelihood and seriousness of each risk.

**2. Introduction:**

The purpose of risk management is to ensure levels of risk and uncertainty are identified and then properly managed in a structured way, so that any potential thread to the delivery of outputs (level of resourcing, time, cost and quality) and the realisation of outcomes/benefits by the business owner is appropriately managed to ensure the project is completed successfully.

The objectives of the risk management approach in the Flight Booking Management System Project are to identify, assess and mitigate risks where possible and to continually monitor risks throughout the remainder of the project as other risks or threats emerge or a risk’s impact or likelihood changes.

As risk management is an ongoing process over the life of a project, this Risk Management Plan and Risk Register must be considered a ‘snap shot’ of relevant risks at one point in time.

Where required, the process of risk identification, assessment and the development of countermeasures will involve consultation with the Steering Committee members, the Flight Booking System Reference Group, other relevant stakeholders and the Project Team members.

**3. Risks:**

**3.1. Loss of Team Members:**

At any moment throughout the project, the team could lose members due to many possible reasons, such as dropping the subject, leaving university, leaving the country, death, world war 3, etc.

In such a case where one member of the team leaves, the rest of the team will have to do a bit more work to cover for the lost member. If 2 or more were to leave the team, that’s where it would become a serious problem in regards to brain power and working power.

We have one Manager, one Chief Programmer, one Chief GUI Designer/Tester and two Analysts/Designers. We have already lost one member of the team, but everyone agreed to do a bit more work and shift around to the areas that needed more work. If we were to lose one Analyst/Designer, it wouldn’t be critical, but developing a perfect system will become much harder as the workload will be doubled for the Analyst/Designer. If the GUI Designer were to leave, since it is a specialised field, another member would need to spend time and effort to learn it. If that is the case, we would most probably not develop a GUI unless the development was well ahead and we had the time to do so.

If the Tester were to leave, the Programmer could make notes of all the aspects in the system that need to be tested and hand them out to the rest of the team to work on. If the Programmer were to leave, since they are our strongest programmer, it will hurt out development quite a lot, but everyone else would have to work together in the programming to get the job done. If the manager were to leave, the documentation, organising and scheduling has already been done, so for another member to pick up that role wouldn’t be too much of a challenge.

In order to try and minimise any reason people may have for leaving the team, weekly meetings are held, the team is managed as organised as possible. Each team member must feel comfortable and motivated to work on this project.

We cannot prevent a member leaving the project, but we can prevent the loss of the work they contribute to the project with information sharing. During each meeting all information created in the previous week is shared with the group. This way in the event a member leaves, all his/her work with the project is not lost as we have a record of their work.

**3.2. Loss of Access to Technology:**

It is possible that there are technical problems that occur, whether it be the hardware dies, a fire destroys the entire office or something else. In those situations, development will be required to halt until the necessary funding and/or replacement technology/equipment is obtained. This may set the project back quite a long time depending on the severity of the incident. If all of our access to the required technology/equipment for this project was to disappear, we would not be able to do anything until the client supplies us with new technology/equipment. This will also bring in additional project costs, which if get too big for the budget or scope of the project, will have the project terminated.

**3.3. Loss of Documentation/Code:**

Just like with the loss of access to technology, if documentation and/or code were to be lost, the project would most likely be set back much further since it cannot be replaced as easily with a sum of money. The corruption of files and or human error which results in the misplacement/deletion of files must be re-done either from scratch or the latest available checkpoint/version.

To help minimise these risks, all of the files that are worked on are uploaded to a github server which each team member has access to. Each member also has a copy of the server downloaded on their home desktops/laptops. There are even more copies of files stored on other data storage devices such as USBs. Therefore, in the case of the loss of documentation and/or code, the time and work lost will be marginally minimalised and easily re-done.

**3.4. Loss of Client:**

In the case that the client disappears due to family issues, health issues or other, there will no longer be a project to do. The worst-case scenario would be that no one a part of the project will receive anything. The entire project will have been a waste of time, money and effort.

In order to help minimise this, a weekly meeting is scheduled with the client with well-planned questions and deliverables to present to them. If the client feels confident in the project team, their will to work with us will increase. The weekly meetings also give us a better sense on whether the client is serious about this project and whether they will go through with it all the way with us or not. Ensuring that we fulfil all of the client’s requirements and goals is our most important task, as in the end, the project is alive and we are on the project team because of the client.

**3.5. Loss of Funding:**

Similar to the loss of a client, without any funding, the project will have nowhere further to go and the team will halt the project. The loss of funding could happen for one of many reasons. Some of those reasons may be that the client becomes bankrupt or decides to abandon important portions if not the entire project due to budget and financial instability.

If this were to happen, all the time and effort spend on the project would be null and void. Although this situation is not entirely under our control, there are factors which could help minimise the chances even a little bit.

It is up to the Analysts and Designers to understand what is required, plan and design the most efficient possible way to advance the project. A thorough and optimised design/plan will bring a higher success rate to the project and minimise the chances of financial failure.

**3.6. Suspension of the Project:**

It is possible that the project gets suspended if not cancelled due to one of many reasons as mentioned in the loss of the client or the loss of funding. Ultimately, this is not something anyone a part of the team wants to occur at any moment throughout the project. It could happen at the beginning or towards the end, in some cases being worse than others. The goal is to see the project through to the end.

The suspension of the project will make all the time and effort spent up until that point completely wasted. The team will have nothing further to do and depending on the reason and the way the project got suspended or cancelled, the team may never receive their pay or work certificate as opposed to just losing time and effort.

The only way the project team could minimise the chances of the project being suspended or terminated is to make everyone involved confident and keep their morale high. This is done with a well-done analysis, which leads to the planning, design and then implementation of the project. Weekly meetings are conducted between both the client and project team, as well as a meeting for the project team themselves.

The client meeting acts as a way for the project team to suggest ideas and show prototypes while receiving the requirements of the client to make sure both parties agree to what is being worked on. The project team meetings are held to discuss any issues and to set tasks on what has to be done as well as how. The meetings put everyone involved in the project at the same level when it comes to the progress and status of the project.

**3.7. Changing of Requirements:**

The project is susceptible to constant change, and that is expected to happen throughout the life of this project. The flow of the project greatly depends on how these changes come into play. If the base requirements were established at the beginning and the project was planned out well, then small changes and added specifications to the requirements will not be as great of an addition as a complete overhaul or many conflicting requirements.

The changing of requirements means that analysis, design, implementation and testing must be done constantly over the duration of the entire project. This brings the project to the issue of having more obstacles to be handled and work to be done, resulting in a higher cost and longer schedule of the project. If the requirements are too far-fetched because the client is inexperienced or lacks knowledge of how project development works, the project will be hard to work on in an efficient manner.

Therefore, frequent meetings are conducted between both the client and the project team, as well as frequent meetings between the project team themselves. The requirements can be discussed by both parties real-time, utilising good communication to clarify any options and/or issues with the requirements as soon as possible. After the client meetings conclude with a successful understanding in feasible changes, then project team goes through their own meeting to discuss and plan the technical side of the changes. Constant planning and re-design will be required to effectively handle this project.

**3.8. Changing of Clients:**

Just like in the changing of the requirements, the changing of clients also leads to re-evaluation and re-designing to be needed. The changing of clients contains all the element in the changing of requirements. Although, the changing of clients usually leads to a bit more technical changes.

The changing of a client is usually done from company to company as opposed to a single, small-scale client. If the current client were to pass on or sell the project to another person or company, they may have a different view and goal to the previous client. This leads to further changes in the requirements of the project. Although, with the changing of clients, it may not always be the same case that they have the same budget and/or technology as the previous client. This could be a good or bad aspect of client change.

If the new client has broader and greater views of the future of the implemented project, that may be the reason why they have taken over the project and are able to fund it with a greater budget, greater technology and crew. This would then require the system to be re-designed based off of new requirements, whether it be technical or the client’s.

However, if it happens to be the opposite and instead, the new client is has a smaller view and goal for this project, negative changes could be put into place. Also, with the changing of clients, the project team does not have as easy a choice of deciding to work on the project with the client or not as it’s already underway. Since the new client may desire a complete overhaul of the system to be designed, this could lead to the changing of the team as well. Team members may be removed from the project team and this would go back to the problems of the loss of team members.

In the chance that the client were to change, the best approach would be to immediately establish a client meeting and brief them on the entire project. The point is to show them what has been done and that the analysis of and planning done on the project are effective and do not need to be changed. As it is a new client, they may be inexperienced and unknowledgeable about what they may want, so they must be reassured with the future of the system we have designed.

**3.9. Maintenance:**

After the project has been completed as per the requirements of the client, they may still desire future upgrades to the system. In the case where maintenance is required, whether it be from the project team or a future team, the design of the system must be developer-friendly.

Maintenance could involve the fixes of problems which are encountered while the system is implemented and live, but assuming our design was done properly, this will not be a problem except for in specific cases which were purposely disregarded based on the feasibility studies of the project. For example, in the future, all computers may run a 128-bit operating system or all executables get a complete structure change. These are not factors that will be an issue in the near future, so designing the system to fit such different requirements is not the most efficient design available.

In terms of the maintenance of desired upgrades in the near future, which could come from the client’s new visions or feedback of customers, the system must be easy to understand and work on down the track. This requires all documentation, plans, sketches, user manuals and code to be well-done. Well-done documentation is easy to understand, consistant and well-formatted.

As for the code to allow for easy amendment in the future, the code is detailed and legible. This is achieved through the documentation of the system, such as diagrams and a data dictionary. The code itself also allows for commenting, which allows for complete English to be used to describe what parts of the system do exactly what. It eases the work required of any programming in the future, especially if done by another person who has never seen the system before.

**3.10. Over-Evolution of the System:**

Similar to maintenance, the issue of over-evolution or lacking capabilities of the system can be answered by a different design, but it would have been less feasible. However, this issue could also come from straight-out poor planning and design. An example issue is that in 5 years time, the company using the system may have been so successful, that their infrastructure and/or the system is unable to cope with the amount of data being processed.

The data requirements or executable structure may change in the future, causing a lack of processing ability by the system to operate with such types or amounts of data. There may be a new language required or the database may run out of space, or the system may not be able to handle that much memory.

The measures taken against this are done with feasible and near-future ideals set in mind. The system is tested throughout its development lifecycle and is ensured to be stable and efficient for the requirements at the time of implementation. Once again, the code is designed in such a way, that should this issue ever need to be solved in the future, it can be done so as easily as possible.

**4. References:**

<http://www.egovernment.tas.gov.au/__data/assets/pdf_file/0004/78367/Realising_Project_Benefits_Project_Risk_Management_Plan.pdf>

(Accessed: Thursday, 26/03/2015, 3:20pm)