Introduction

Purpose of Plan

The Project Incident Mapping (Project IM) plan is to define the project's goals, objectives and scope which include an analysis of project risks involved and its management. It will also define the project schedule, project resources and managerial control plans such as reporting, scope control, monitoring, tracking, communication and quality control mechanisms. This project plan will define the Work Breakdown Structure, the stakeholders involved in doing the work and a timeline consisting of the Gantt chart.

Goals and Objectives

The client has imposed requirements so that "Officers will be able to log and retrieve information more quickly and easily, and switch seamlessly between applications, without having to enter the same data many times". As well as complying with this requirement, the system must be accessible in a majority of popular internet browsers which will allow easy integration into mobile vehicles and hardware devices for field use.

The system must also be able to filter information if it is required for the authorized officer, to prevent a display of irrelevant chunks of information to the field in which the officer works in.

The primary objective is to build and deliver a subsystem complying with these goals and objectives to the client within the timeframe of 3.5 months, as well as staying within the budget range.

Scope

Scope Definition

Project IM will provide an application to interface between the traffic and criminal incident departments of the NSW Police force. This will require the following technology:

Development

- HTML
- JavaScript
- CSS
- ASP

Client

- JavaScript and cookies enabled
- Major web browsers

The proposed Project IM delivered to the client will be:

- A web application of the interface for authorized officers to "retrieve information more quickly and easily", wherever they are
  - A full cross-browser functionality of the application for easy integration into a variety of hardware devices
  - Secure login system
- Seamless access to applications required in both traffic and criminal incident departments
- Current map of NSW including detailed streets
- Insertion of information into the map
  - By longitude and latitude
  - Address
- Filtering of information within the application according to:
  - Authorized user rank
  - Preferences
- Application will help view growing trends in incidents for:
  - Criminal incidents
  - Traffic incidents

Project IM will include the following interfaces:

- Google Maps for regular updates on the current map of NSW

Items Beyond Scope

- Other than providing noticeable trends in incidents, the application will not provide solutions to incident cases
- Triple zero (000) calls will not have direct access to the application
- Hardware solutions for mobile command vehicles or field officers such as the implementation of GPS and the system within vehicles and/or handheld devices
- Satellite access to develop and update the map of NSW
- Development and implementation of applications required by traffic and criminal incident departments
- Hardware upgrades required for implementation
- Additional software required for the use of the application

Projected Budget

Project IM is being developed by a team of 5 members, each of whom has high level of experience in their field of work. Projected budget stands as a maximum of $50,000 per member within the 3.5 month timeframe, paid on an hourly basis. A further $100,000 will be required for the license to use Google Maps as an interface for regular satellite updates of the NSW map. The total projected budget is equal to $350,000.

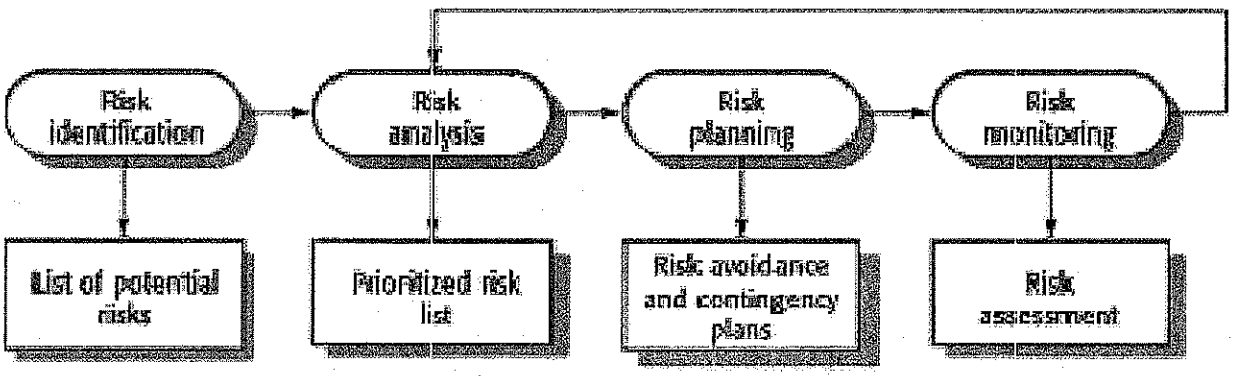
Assumptions

There are functional applications used by the current traffic and criminal departments which require integration to improve efficiency within those departments, rather than requiring frequent reauthorizations within the system.

Development Risks and Management

Risk Management Process

The risk management process as seen in Software Engineering 9th Edition, Sommerville. Risk – or uncertainty – management is essential during the lifecycle of the project; Project IM will be using the following risk management process until the system is delivered.



Scope

Scope Definition

The project will improve service by providing an information system that responds to the problems identified by the NSW Police in the field of forensic DNA testing. This includes reducing the backlog of existing data, reducing time between arrest and conviction for perpetrators, improved inter-branch communication and flexibility regarding system options and customisations. As well as building a good foundation for the integration of future applications.

The Project will introduce the following new technology:

- Web based interfaces (platform independent and can run on existing desktop/laptop/mobile hardware)
- Virtualisation architecture (with high scalability)
- Single sign-on
- Advanced reporting software
- Data prioritisation and escalation services
- Basis for integration of other applications

Items beyond Scope

The project does not include the following issues in the initial scope

- Upgrade or replace existing desktop or notebook hardware
- Decreased power costs due to virtualisation architecture
- Purchase of additional printers to render new reports
- Replacement of existing category 5 cabling or other infrastructure related communications materials

Project Resources

Taking the nature of the system into account it's essential for it to be secure and be available only to personnel of NSW police department. The existing data centre will be used as a method to keep security high. Minimal space will be required for the hardware.

1. Executive Summary

1.1. Statement of Purpose

The Police Mobile Command Deployment System (PMCDS) Project Plan will provide an outline of the project, including the project's goals and objectives and how it will achieve them. It will do this by defining the following:

- Scope of the project
- Resources
- Risks and their management
- Tasks including deliverables
- Roles and responsibilities
- Tracking, communication and quality control mechanisms

1.2. Scope

1.2.1. Project Purpose

The purpose of this project is to develop and implement a new Police Mobile Command Deployment system for the NSW Police Force. The system will need to allow for regular updates using GPS and be able to manage the deployment of the command vehicles based on location away from event in question and current assigned command. By reducing the amount of time that the vehicles spend idle, we can increase the effective size of the vehicle fleet.

1.2.2. Background

Presently, police mobile command vehicles provide a valuable asset for police commands across NSW – However, the demand for use of the vehicles presently outstrips their availability. Current NSW Liberals & Nationals policy dictates that the availability of these vehicles should be increased.

Their strategy to improve the availability of these resources to police commands consists of both investing in additional vehicles (increasing the fleet from 41 vehicles with an additional 15), and deploying the vehicles on the basis of need, so that these resources are available where they are most needed.

Presently we make the assumption that police mobile command vehicles are being assigned to specific police commands, and it is left to the individual commands to organise deployment of the vehicles in an optimal manner.

1.2.3. Objectives

The projects main objective is to help the NSW Police Force easily manage their new command vehicles, specifically help reduce their effective fleet size and to make it easier to organise commands and deployments. It will also need to identify current vehicle locations using GPS and current levels of activity. Using this information it will provide an informative recommendation on how to proceed with the current situation.

1.2.4. Key Stakeholders

NSW Police Force – All Staff need to be trained in the new system

Deborah Richards – Project Sponsor on behalf of NSW Police

[redacted] – Project manger

[redacted] – Head software engineer

[redacted] – System developer

[redacted] – Systems analyst

[redacted] – Systems analyst

1.2.5. Organisation Requirements

- Secure as it hold sensitive and vital information for the NSW police force.
- Higher than average quality, specifically in response time in relation to vehicle GPS location.

1.2.6. Approach

In order to satisfy these requirements, this project will need to create an application using Java, which integrates Google Maps and GPS locators to effectively deliver the required functionality of the software. Secure databases will also be used to provide added security.

1.2.7. Constraints

This project will need to be completed by the 11th of November 2011, there are no costs restraints, however testing such an application may prove difficult and may source testing constraints.

1.2.8. Assumptions

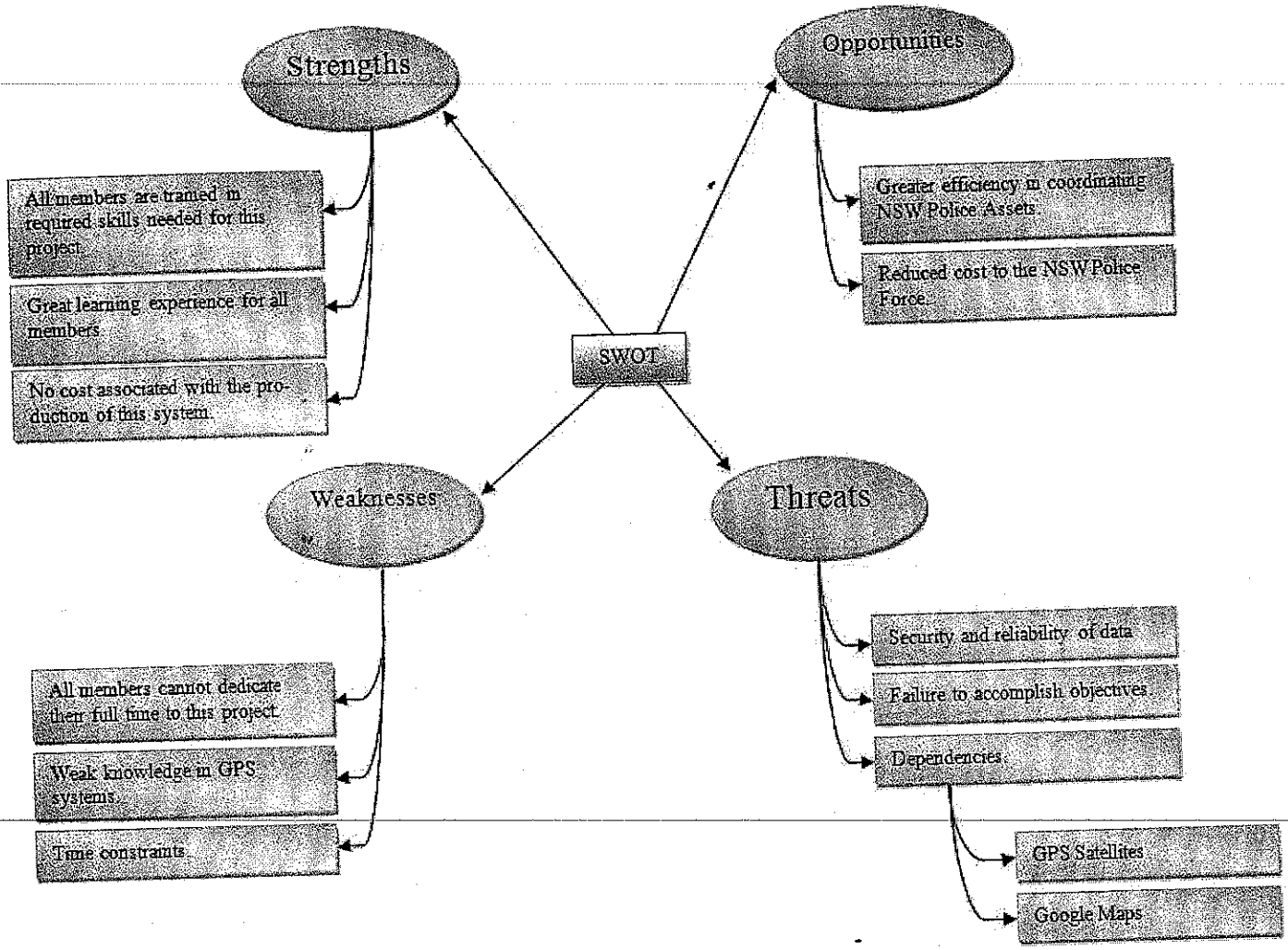
We assume the following:

- Each Police Command Vehicle is fitted with a GPS locator.
- Each Police Command Vehicle can easily communicate with NSW Police Force Headquarters to update status, current situation and acknowledge new commands.

1.2.9. Scope Verification

After each incremental milestone, a meeting will be organised with the project sponsor - Deborah who then provide feedback on current project direction and satisfaction on meeting expectations.

1.3. SWOT Analysis



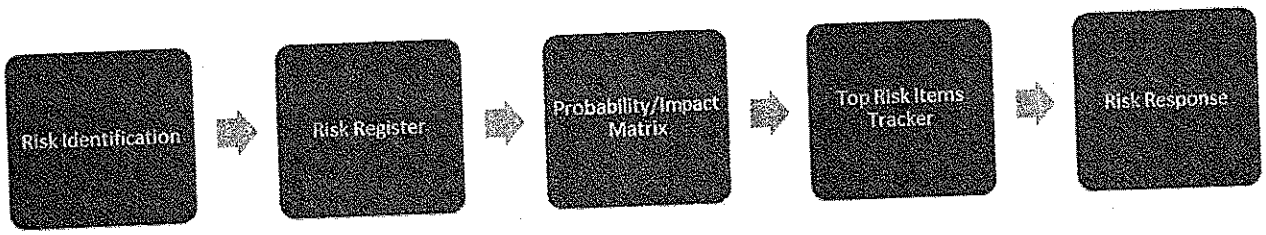
2. Risk Management

Risk Management Roles and Responsibilities

- Project Manager chairs the risk assessment meetings
- Project team participates in risk assessment meetings and members serve as meeting recorder and timekeeper
- Project team participates in the regular update of the risk register, probability/impact matrix and top risk item tracker

2.1. Risk Management Methodology and Documentation

The approach we have taken to manage risks for this project included a methodical process by which the project team identified, logged, scored, and ranked the various risks. The most likely and highest impact risks were added to the top risk items tracker to ensure that the assigned risk managers take the necessary steps to implement the mitigation response at the appropriate time during the schedule. The project team will also document risk information using a risk register, probability/impact matrix, and a top risk item tracker in either paper or electronic form. The following is a graphical representation of the risk management process for our project team. This process is repeated and updated to our risk trackers every time a new requirement is introduced to the project plan.



2.2. Risk Identification

For the project, risk identification will be conducted in the initial project risk assessment meeting. The project manager chaired the risk assessment meeting and distributed notepads to each member of the team and allowed 10 minutes for all team members to record as many risks as possible. These risks are then updated to the risk register, probability/impact matrix, and the top risk item tracker. The results from our project risk assessment meeting are represented in the risk register in the section below.

2.3. Risk Register

The Risk Register for this project is a log of all identified risks, their probability and impact to the project, risk ranking and their unique identification number. The register is created through the initial project risk management meeting led by the project manager. During this meeting, the project team identified and categorized each risk. Additionally, the team assigned each risk with a score based on the probability of it occurring and the impact it could potentially have. The information gathered in this register is then used by the project team in the analysis and prioritization of the risks.

In order to effectively integrate categories of risk into the numbering system, the first identifier in the numbering system should correspond to the first letter of the risk category name. For example:

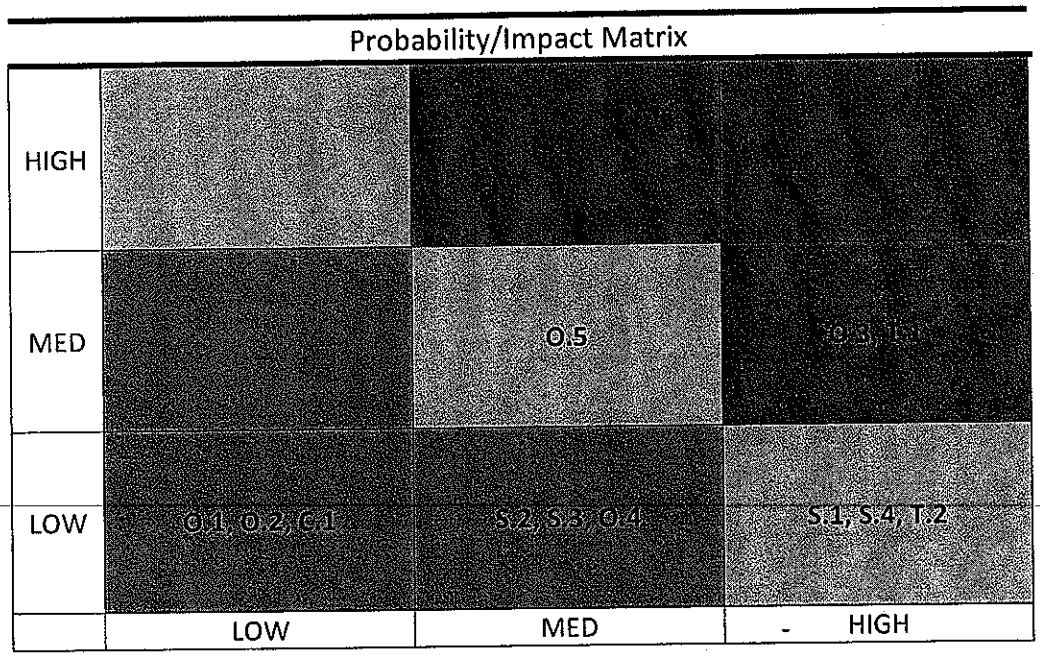
- S.1 = Schedule risks
- O.1 = Operational risks
- T.1 = Technical risks
- C.1 = Cost risks

Below is an example of a risk register use by our team members. It also contains the list of risks the project team have identified in the earlier stages.

Risk	No.	Rank	Chance (H/M/L)	Impact (H/M/L)	Cause of risk
Missed deadlines and over-runs	S.1	4	L	H	Wrong time estimation, bad time management, miscommunication of scheduled tasks, unexpected scope expansion, lack required skills and knowledge to complete certain components on time.
Accidental exclusion of required components	S.2	6	L	M	Ineffective communication and tracking of scheduled tasks.
Resource overruns	S.3	7	L	M	Ineffective tracking of resources, underestimate skills of staff and incorrectly delegate tasks.
Incompletion of individual tasks	S.4	5	L	H	Ineffective management of human resources, ineffective delegation of tasks.
Incorrect use of system leading to decrease productivity of system	O.1	9	L	L	Ineffective or lack of police training for the use of newly developed system.
Unexpected vehicle down time and equipment failure.	O.2	11	L	L	Did not accommodate for possible unforseen events in project plan or user documentation.
Inability to achieve a quantifiable benefit from the system	O.3	1	M	H	The system was not able to manage the MPCCs in a way where it was able produce quantifiable benefit by using fewer vehicles.
Lack of police human and technical resource to	O.4	8	L	M	Shortage of manpower, difficulty with physical integration of system, unforseen events such as strikes.
Difficulty with operation of system by normal staff	O.5	6	M	M	User interface is badly designed rendering difficult to use for police staff with average standard of computer literacy.
Difficulty or dead end with development of certain functionalities	T.1	2	M	H	Failure to identify complex functionalities and knowledge required to develop those functionalities.
Development of function is not compatible with rest of the system	T.2	3	L	H	Unexpected addition of scope requirements that is not compatible with developing system, did not devise project properly.
Unexpected additional running costs such as servicing of MPCCs and training	C.1	10	L	L	Ineffective communication to end users of any additional costs.

2.4. Risk Qualification and Prioritization

In order to determine the severity of the risks identified by the team, a probability and impact factor will also be included into the risk register. This information aids the project team in prioritizing risks based upon the probability and effect they may have on the project. The project managers utilize a probability/impact matrix to facilitate the team in recognising risks that require priority. The probability/impact matrix below helps the project team in identifying the high priority risks in the red areas. The remaining colours orange and green represent medium and low priority respectively.



2.5. Top Three Risk Item Tracker

The risk analysis process should ultimately output to the top risk item tracker where team members will ideally respond with a prompt solution. This table lists the top priority risks including information on current rankings as well as previous rankings, date it was last updated, a summary of progress made and the person it was checked by.

Upon analysis of probability versus impact of risks, we have concluded that the three risks with the highest priority:

- Unable to achieve quantifiable benefit by using the system to manage MPCCs. As mentioned in the feasibility study, a major part of project success relies on the benefit derived from the deployment of the newly developed system. It is simple logic, if there is insufficient or even no benefit, there is no point of developing the system. We will attempt to overcome

this problem with carefully devised algorithms to ensure effective deployment of MPCCs. Ultimately aiming to produce the same productivity with fewer vehicles.

- Unable to produce certain functionalities or deliver on time due to lack of knowledge or over optimistic expectation of proposed system. As a project team, we can lower the risks by ensuring all members thoroughly understand all requirements and skills required to carry out the developments. It is also important to carry out on going coaching sessions to ensure team members are equipped to complete their delegated tasks on time.

- Development of new required functionality is incompatible with rest of system. This is caused by the introduction of new scope requirements which is common in an agile software development life cycle. To ensure this does not occur, we aim to design and use scalable solutions to cater for any future changes and developments. Although the impact of such a scenario is disastrous, the likelihood of it happening is rated low due to the nature of the project being a controlled assessment.

Shown below, is an example of the actual template that team members have to update on a regular basis.

Top 3 Risk Item Tracker					
Risk No, Name	**Rank**	**Last Rank**	**Progress**	**Last Updated**	**Checked by**
Risk Manager:					
Signature:					

2.6. Risk Response - Mitigation/Avoidance Planning

The project manager leads the project team in developing responses to each identified risk. As more risks are identified, they will be qualified and the team will develop avoidance and mitigation strategies accordingly. These risks will also be updated to the Risk Register and the project plan to ensure they are monitored at the appropriate times and are deal with in a prompt manner.

The risks for this project will be managed and controlled within the constraints of time, scope, and resources. All identified risks will be evaluated in order to determine how they affect these constraints. The project manager, with the assistance of the project team, will determine the best way to respond to each risk to ensure compliance with these constraints.

3. Schedule

3.1. Tasks

For a list of all tasks please see appendix 1

3.2. Project Deliverables

Deliverable 1: Feasibility Study (Friday 12th of August 2011)

Deliverable 2: Project Plan (Friday 19th of August 2011)

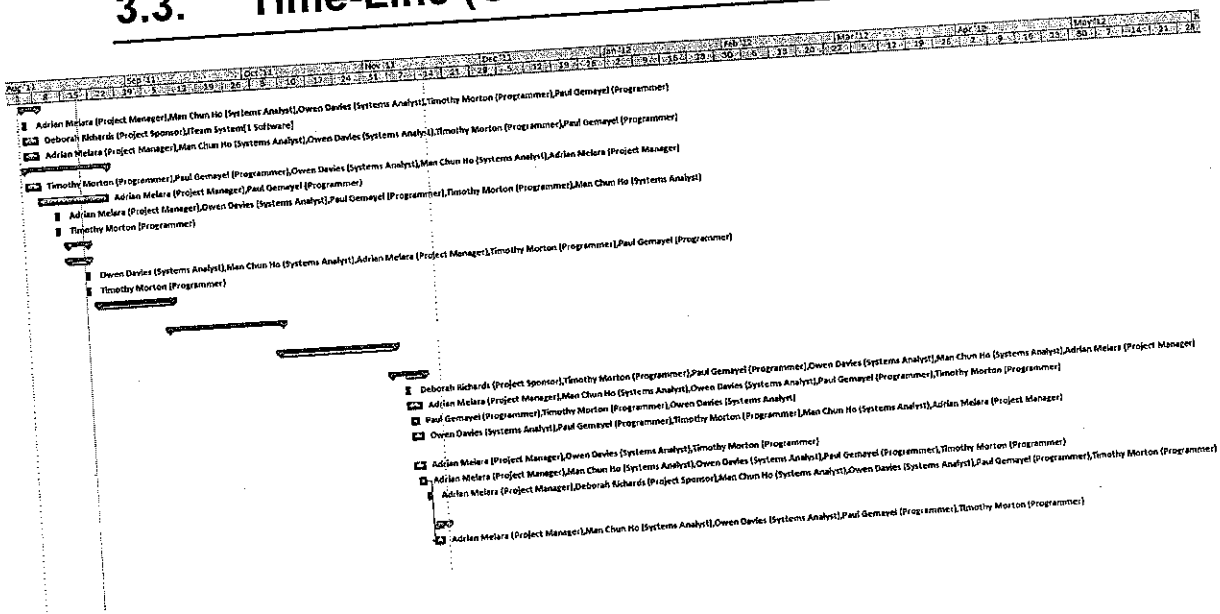
Deliverable 3: Updated Plan, Requirements, Design, Test Specification, Prototype (Friday 9th September 2011)

Deliverable 4: Updated Plan, Requirements, Design, Test Specification, Prototype (Friday 7th October 2011)

Deliverable 5: Updated Plan, Requirements, Design, Test Specification, Prototype (Friday 4th November 2011)

Final Deliverable: Software Demonstration, RTM, User Manual and Final Report (Friday 11th November 2011)

3.3. Time-Line (Gantt chart)



Risk Management

Risk management is the documentation, calculation, and ranking of risks. Risks can arise from doubt in financial markets, legal responsibilities, project failures, credit risk, natural causes and accidents. Numerous risk management strategies contain relocating the risk to alternative group, dodging the risk, minimize the negative consequence of the risk, and tolerant some or all of the penalties of a particular risk. Certain features of risk management principles have come under analysis for having no measurable enhancement on risk even though the confidence in approximations and decisions grow.

There are four basic risk management strategies applicable: Mitigation, transference, acceptance and avoidance.

- Mitigation: project team members can try to mitigate the risk by fixing any flaw attainable or provide compensatory control to reduce the likelihood/ impact associated with the flaw.
- Transference: another option is to transfer the risk to a third party to reduce the overall impact on the project. However, this option does not fix the existing flaws in the project.
- Acceptance: in some circumstances, it might be possible for the project team members to do nothing and accept the risk.
- Avoidance: It is possible for the project to remove any vulnerability or threats identified by changing the initial plan

1. Risk Register

Risk/ ID	Description	Chance (H/M/L)	Impact (H/M/L)	Priority (H/M/L)	Trigger	Response	Responsibility
Work Overload **RK01**	The work assigned to any one individual is overwhelming at that point in time.	L	M	H	Amount of work allocated to each team member is poorly distributed.	The project manger will hold regular meetings with key project workers, at least bi-weekly, and create Gantt charts (WBS).	Project Manager
Exceeding Budget **RK02**	The total hours invested exceeds the allowable budget.	L	H	H	The project schedule is poorly designed and lack of calculation.	Project manger holds meetings with the accountant to review the finances.	Project Manager Accountant
Unsatisfactory Solution **RK03**	The software does not meet specific goals of the project.	M	H	H	Solution is not on par to client's expectations.	Will work with the software worker on project to ensure that define performance goals that are realistic and achievable.	Project Manager Software developer
Scope Creep **RK04**	The client requires more features from the solution.	M	H	M	Solution is not on par to client's expectations.	Re-analyze the requirements, attempt to capture most requirements early.	Project Manager Team Members
Communication **RK05**	A lack in communication in the team, which could result in a poor quality project or the software not even, is created.	M	M	M	Poor communication between the team members.	Create an environment in the workspace that will allow workers communicate to one another and be friendly to each other.	Project Manager Team Members

Hacker **RK06**	An unapproved individual could break into the system or sever.	L	H	M	A hacker is able to hack into the system and does so.	All security measures recommended by the Admin Support security plan will be put in place also monitor security recommendations and patch any software as security patches become available. Finally ensure that all passwords are changed monthly and replaced with strong passwords.	Admin Support
Implementation **RK07**	Failure to implement the project.	L	H	H	During the implantation stage a problem occur and the system it unable to be implemented.	Analysis four main implantation strategies (parallel, pilot, phase and direct) and assets which strategy is best suited for the software.	Project Manger Admin Support
Absent members **RK08**	Members not showing up during team meetings.	L	M	M	Level of commitment, health, lack of motivation.	Team building, use of expectancy theory	Team members
Poor Allocation of Tasks **RK09**	The tasks are assigned to members from incompatible background and knowledge.	M	H	M	Assignee is poorly informed/ aware of the incompatibility, poor resource management.	Keep a record of members' background and capabilities, re-allocating task.	Project Manager
Lack of Skills	Members lack skills and experience with	L	H	M	Management hires an inexperienced	Use top talent, team building, provides training.	Team members

Schedule

Tasks

The project was divided into 6 main phases as follows:

- Initiating (Deliverable 1)
- Planning (Deliverable 2)
- Requirements Analysis, Design, Test (Increment 1)
- System Design, Testing, Requirements (Increment 2)
- Detailed Design, Requirements, Testing (Increment 3)
- Completion (Final Deliverable)

The duration of each task and their sub-tasks were determined based on their complexity and importance for the overall success of the project. We also note the use of Agile SDLC in this project which will involve re-examining of these aspects at each stage of the project to ensure any additional change of the requirements are implemented in the final solution.

The responsibilities for the accomplishment of these tasks were allocated according to the group roles as follows:

- Steering Committee
- Core Management
- Customer Project Director
- IT Project Lead
- Project Manager
- Team Members
- Project Sponsor

For more detail on these roles see 'Organisation, Roles and Responsibilities', page 12.

Deliverables

Project-related deliverables:
The project deliverables include the Project Purpose, Project Scope, Risk Identification, Risk Management, Schedule Project Resources, Group Roles and Responsibilities, Tracking, Communications, and Quality Control Mechanisms, Test Specifications, Requirement Analysis, Design Documentation, and any other documents relevant to the project. This deliverables are presented to Debbie who is our project sponsor.

Product-related deliverables:

Software Demonstration: This includes the demonstration of the functionalities of the System to the client.

User Manual: This document is intended to provide assistance and presents all the functionalities of the system to the different types of users it will have.

Final Report: This contains the project planning, requirements and analysis, design, implementation, and learning outcomes.

4.0 Deliverables

The project will be divided into four individual deliverables. Each deliverable will contain variable tasks which have been allocated to the team. These deliverables will also act as milestones in the development process.

The deliverables include:

Tasks	Date Due
Feasibility Study	12/08/2011
Project Plan	19/08/2011
Increment One: - Updated Plan - Design - Test Specification - Prototype - Requirements	09/09/2011
Increment Two: - Updated Plan - Requirements - Test Specification - Design - Prototype	07/10/2011
Increment Three: - Updated Plan - Requirements - Design - Test Specification - Prototype	04/11/2011
Final Deliverable: - Software Demonstration - RTM , User Manual - Final Report	11/11/2011

4.0 Project Resources

This section of the document provides cost, effort and time estimates for the project to get JusticeKrew up and running:

4.1 Historical data used

Historical Data needs to be backed up and any information currently in process must be saved.

In order to do this, Group [redacted] members [redacted] will be heavily involved in backing up database and putting checkpoints in place to have an exit strategy if something goes wrong.

4.2 People

The team of 5 skilled individuals [redacted] will be carrying out task of getting the infrastructure and the application built.

However, there will be other key people required to complete the project. This may involve a few current users (to create a user group), staff with IP, key members from other applications/departments and finally the IT team for all the other departments/applications involved to bring them all together.

4.3 Hardware

The main hardware for basic project plan and implementation will be PCs and Laptops used by Group [redacted] team members.

Mainly, the web services will be installed using current servers that NSW Department are using.

The beauty of a SaaS application is that it has a low requirement for any PC to run the application, online – anytime, anywhere!

4.4 Software

We have used a number of software applications to run with this project. Some of them are:

- Microsoft Project – for project planning
- Microsoft Word – for document planning & writing
- Sybase Power Designer – to design the architecture of the software
- Microsoft Visual Studio – to build the application in .Net (C#)
- Microsoft SQL Management Studio – to manage the database
- Other applications to create HTML/CSS files and manage the application

4.5 Other Tools & Resources

Some of the other resources and tools used will be:

- iPad – to show the prototype and designs to stakeholders
- Stationary – modelling, designing and printing
- Computer Labs – to meet and collaborate ideas and implement the design

6.3 Member's roles and responsibilities

Each team member has been assigned with a primary role based on their skill set and primary and past experience. Each team member will focus on their role throughout the project. To share the workload evenly each member will find themselves having to take on other roles to assist other members. The table below outlines the individual roles and responsibilities of each group member of the project.

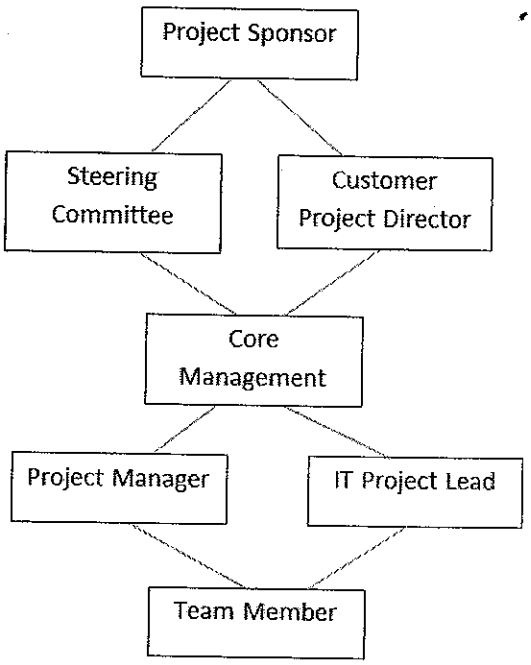
Member:	Roles:	Primary Responsibilities:	Skill set:
[redacted]	Project Manager	- Managing conflict within the group -Ensuring that features set out in the system scope are eventually implemented. -Managing the addition of features towards the application in a way such that no unnecessary additions outside of scope are made. - Identifying the skills and experience of the members of the team.	Programming: - C++ - Python - Java - SQL Software Experience: - Ms Office Applications - MS Project - Photoshop
[redacted]	Software Developer	- Primary Software Developer - Design Architect - Tests and tries the solution - Monitoring the system development to ensure it meets all requirements	Programming: - C++ - Python - Java - SQL - HTML/ CSS Software Experience: - Ms Office Applications - MS Project - Photoshop - Eclipse
[redacted]	Software Developer	- Primary Software Developer - Design Architect - Tests and tries the solution - Monitoring the system development to ensure it meets all requirements	Programming: - C++ - Java - HTML/ CSS - Python - SQL Software Experience: - Ms Office Applications - Eclipse - SQL Developer - Power Designer

	Software architect	- Developing the architecture of the system - Identify the project requirements	Programming: - C++ - HTML Software Experience: - Ms Office Applications - MS Project - Power Designer - Photoshop - BPM WorkBench
	Software architect	- Developing the architecture of the system - Identify the project requirements	Programming: - C++ Software Experience: - Ms Office Applications - MS Project - Enterprise Architect - Power Designer

Organisation

Group Roles and Responsibilities

Team Structure



Project Sponsor –	The ultimate authority over the project
Steering Committee –	Overview and maintain project direction
Customer Project Director –	Interface with the project sponsor
Core Management –	Team leader position, brings all components together
Project Manager –	Responsible for each sub-task of a milestone
IT Project Lead –	Responsible for each sub-task's technology required
Team Member –	General execution of tasks

**Our team will rotate between these roles throughout the milestones*

Outline of Roles and Responsibilities

Role	Responsibilities
Project Sponsor *(NSW Police: Debbie)*	• Ultimate authority over the project's direction • Provides the necessary funding for project • Approves milestones, and project timeline • Converses with their own organisation to finalise requirements
Steering Committee [redacted]	• Responsible for providing overall guidance to the project's direction • Advises and resolves issues escalated by the Core Manager
Core Management [redacted]	• Meet regularly to review and monitor project progress • Review and approve/deny change requests (Agile SDLC) • Review and monitor project risks • Provide project status updates to sponsor • Quality Control responsibilities (final documents, non-conformance)

Customer Project Director [redacted]	• Manages the development of the project plan • Works with project sponsor to acquire funding • Coordinates functional resources for the team • Review procurement contracts • Review and resolve obstacles to project progress • Quality Control responsibilities (requirements research, inspection testing)
IT Project Lead [redacted]	• Obtains approval for IT related purchases • Ensures technology available for the team • Coordinates the development of the IT project • Responsible for communicating decisions to Core Management • Monitor IT project development and report status • Collaborates with Customer Project Director and steering team • Work with project manager to ensure schedule deadlines are met • Quality Control responsibilities (audits, record keeping, design control)
Project Manager [redacted]	• Reports and receives direction from IT Project Lead, Customer Project Director, and Steering Team • Maintains the project plan • Motivates the team members • Participates in development of the deliverables • Ensures documentation is being developed concurrently with the IT systems • Provides ongoing cost and time analysis • Quality Control responsibilities (audits, record keeping, design control)
Team Member *(Whole Team)**	• Contributes to project schedule developments • Contributes to team deliverables • Participates in team meetings • Provides updates to Project Manager • Maintains records of work • Communicate any problems to Project Manager

**Our team will rotate between these roles throughout the milestones*

Roles and Responsibilities Matrix

	Feasibility	Scope/ WBS	Schedule	Resource allocation	Task work	Organisation	Status Report	Communication	Support /Testing	Budget	Quality
Project Sponsor/ Steering	B,C	A,F	E	E						A,C	
CPD	E	B,F	E	E			E	A,C,D		B,C,D,F	
Core Mgmt.	F	E	A,C	C,E	D,E		A,C,D, E,F	C,D,E,F		E	A,C,D, E,F
Project Manager	C,D,E	D	B,C	E,F	D,E	A,C,D, E,F	E	D,E		E	B,C,D, E,F
IT Lead	C,D	D,E	D,E	D	C,D,E, F	D,E	B,C,D	D	A,C		C,D,E, F
Team Member	D	D,E	D,E	D	C,D,E	D,E	C,D	D	B,E		C,D,E, F
					D				D		

Roles and Responsibilities Legend

A	Create (Primary)
B	Administer (Secondary)
C	Initiating, Planning
D	Implementing
E	Controlling, Monitoring
F	Closing

- Unit testing- Creating tests that ensure that each function works as specified. These small tests can be run every time the code is changed and helps ensure that even after changes, the system will still work correctly.
- Verification – This is verifying that each of the requirements are met as well as meeting the quality metric.

Quality Standards

Quality Area	Low Quality Identifiers	High Quality Identifiers
Security	A low level officer has the same level of access as a high level officer. Anyone has access to the database information.	User permissions and accounts are implemented in such a way that those in upper level management of the police force can set or specify the permission level of individual officers.
Usability	Time for any police officers to find options they require exceed 1 minute. A first time user has to refer to the user manual more than 3 times before finding the option they require.	The system needs to be user-friendly with a clear GUI, where an employee takes less than 10 seconds to find any function they are looking for. A first time user only has to look at the user manual once.
Reliability	System has greater than 3 days downtime a year or takes in excess of 30 seconds to load.	System has less than 30 seconds downtime a year with online support systems and a response time under 2 seconds.
Efficiency	The system cannot handle more than 1500 concurrent users.	The system must be able to handle 3000 concurrent users.
Interoperability	The system is tailor made for one browser and only renders correctly in one browser.	This system will be considered excellent if it can run on all browsers and render correctly.

Tracking

Tools

- Microsoft Project (Gantt charts and Tracking Gantt).– Allows us to track the work being done and overall progress of the project
- Trac – Allows us to track the progress of each iteration of the project's documentation and also to allow us to better manage incremental changes to the project documentation
- Version Control – Allows us to make backups of documents and to compare the changes between the different versions of the project's deliverables and also to have a previous copy to fall back on in the event that something happens to the latest copy.

Change mgt.?
Other means of tracking

Quality Control Mechanisms

Quality Assurance

These are the process that will be used to prevent faults or bugs before they occur.

- Management review – Management will review the project at the end of every deliverable.
- Pair programming – Helps to reduce bugs that could occur in the prototype as well as develop new ideas of how to implement the solution.
- Checklists - As requirement are finished they are to be checked off the check list as well as reviewed for quality.
- Quality gates – Making sure that any task or deliverable is up to standard before proceeding to the next deliverable.

Quality Control

These are processes that will be used to detect faults in systems that are developed.

- Review peer – A member of the group will review any work or code written. The group member that checks the work must be different from the member that wrote it.
- Prototyping – The agile approach is used. This implies that prototyping is part of the development process. This will help define the requirements and debug possible issues.
- Quality metric – Quantifiable standard that if the system achieves, will verify the system as a quality system.

Communication Matrix

Communication Type	Objective of Communication	Medium	Frequency	Participants	Owner
Kickoff Meeting	• Introduce project team and outline project. • Discuss project objectives. • Assign roles and responsibilities.	• Face to face.	Project Inception	Project Team	Project Manager
Project Team Meetings	• Review project plan and project status with entire project team.	• Face to face. • Email. • IM.	Weekly	Project Team	Project Manager
Project Partial Meetings	• Discussion and collaboration between project team members working on related tasks.	• Face to face. • Email. • IM.	Regularly	Project Team	Project Manager
Software Design Meetings	• Discuss and develop software design solutions for the project.	• Face to face. • Email. • IM.	Weekly	Project Team	Software Engineer
Client Meetings	• Report project status to client. • Submit project documentation. • Receive feedback and advice from client.	• Face to face.	Weekly	Project Team, Client	Project Manager
Final Presentation	• Conduct software demonstration. • Submit final project documentation.	• Face to Face.	Project Close	Project Team, Client	Client

Communication

Communication Methods

Weekly face to face meetings	The team will hold a weekly meeting in order to update each other about the current status of the project and also to discuss about any problems that may arise.
Twice daily contact via email	Emails will be used when team member need to inform each other of work that has been finished or other problems if they are not available to meet face to face or are unreachable by phone.
Instant messaging	Instant messaging on services such as MSN and Google talk will occur on the occasion members need to contact each other for extended periods while being unable to meet face to face.
Partial meetings	Partial meetings of less than the full team will occur during the week where the members that are working on related parts can discuss the problems and solutions that they have developed.
Trac	Trac will be used to distribute documents and information.
SVN	SVN will be used for version control on software development.
Phone calls	Only used when a team member needs to be contacted urgently but cannot be reached through other means.

Communications Management

A key factor in the success of any project is its team members cooperating and working together. One of the common reasons that projects fail is because of lack of communication. It is important as the team communicates so that no assumptions are made about one team, communication also leads the team to work together and become unified instead of distant from each other. With this said proper communication between different parts of the team will lead each member to work together to produce a higher quality project.

Communication can take form in a few ways such as face to face meetings, mail/email, and even teleconferencing. Each one of these forms of communication has its ups and downs such as some communication requires a set amount of time and each to complete, so each form of communication should be used appropriately to ensure that all team members benefit from communication with each other and not only just a few.

Face to face meetings

Face to face meetings is the best way to communicate with a team, as all members would have the opportunity to receive a firsthand explanation of what needs to be done and to ask questions from the team leader. A face to face meeting would also be a great time to start brain storming ideas and also allow members to voice their opinions and receive feedback instantly.

The downside to face to face meetings is that it takes a lot of time and its hard to find a time when everyone is free to attend the meeting. With this in mind, meetings should take place at important times such as at the start and the end of the project and or at regular intervals say once a month. Another issues is that face to face meetings take a lot of time and preparation to hold, as there is a small amount of planning that must be done to ensure that only relevant information is voiced at these meetings. Information isn't the only thing that must be relevant, the choice of members to attend the meeting must also be relevant as well to ensure that each member receives only information that is useful or relevant to them.

For the purpose of our project, we have held an initial face to face meeting with all members to discuss our project and brain stormed ideas about features we could add to the disaster plan. We also held a second meeting halfway through to update each other on our current progress and also to see how each member was going with their assigned task. We plan to have one final meeting to bring all our work together and to help each other with any last minute changes.

Mail/Email

Mail or to be more specific email, is a good way to communicate with your team members about important updates that have happened throughout the project. Email a fast and reliable way of passing on important information to groups of people very fast, it also doesn't require the recipient to stop what they are doing to answer this form of communication, allowing them to read the email in their own time. This type of communication is reliable when trying to update others on changes that might have happened throughout the project or to notify people on times and location for the next face to face meeting. The downside to email is that it is not a great media to communicate, as it is only text based, meaning that emotions and thoughts are often hard to relay in an email. Another point is that members of the team often forget to check their email and respond too slowly.

Email has been the main type of communication throughout our team. Many of our members send each other updates on each other's progress and we are also able to arrange meeting times to

discuss more important issues that email can't express. As email can be checked at any time and is an easy way to communicate with each other, we have used this media to keep the team in touch with each other and it has proven to be highly effective as each member knows exactly what they need to do next and is updated on each other's progress.

Teleconferencing

Teleconferencing can be anything from a group audio call or video call with each other. This is an extremely useful way to communicate with each other, especially if members of the team live far away and are unable to make it to the meeting. This form of communication can take place over the phone, in a conference call, where members use their phone and take turns talking to express their feelings and opinions over the phone to the others. Audio calls can also take place over the internet with the proper program people from all over the world can join the conference call.

Teleconferencing can also take form of video calls as well which can prove to be more useful than audio calls as members could see each other and easily express their ideas and thought as if they were having a face to face meeting. The downside to teleconferencing it requires each member to have a good internet connection as video/audio calls to each other require a lot of internet usage. Many members may drop from the call or not be able to connect at all due this problem.

This form of communication has been implemented in our team, so members of the team could talk and discuss important issues at the end of each week. However it was hard for some of us to use this method, as there were problems of capped internet, failing hardware and also the problem of members not being able to attend each conference.

Scrum meetings (Sprints)

Our team is using an agile approach to create the system and so we opted to have daily scrum. These meetings will have specific objectives that we will discuss and will have to be cover such as

- What was completed yesterday?
- What are the planned tasks for today?
- Has anyone encountered any problems?

Although problems should be controlled by the head of the scrum meeting, most of the problems should be resolved after the meeting as solutions take a lot of time to resolve and there isn't enough time in the meeting.

Each sprint will be held for only 15-20 minutes a day at 11am every day and those who can not attend the meetings will be sent the notes made in each scrum meeting. The meeting will be held at uni or failing that will be held online via teleconferencing.

Since this is a small team of 5 we have planned for everyone to be a part of all meetings as each person on the team would be able to deliver vital information and updates on their field of work.

Quality Management

It is crucial to ensure quality in not just at creating the project but at every step of the project from the basic documents such as defining the problem to the designing the software to testing and maintenance. Quality however is not a measurement we can compare to a graph but rather a quality project is one that meets all of the client's requests and specification and runs without error.

Quality control

To ensure that a project is of high quality to the client what we will do is perform regular testing to ensure that our product that we will produce at the end of the day meets all of the client's specification and that there are no errors. There will not just be one test that will be set up but a series of tests that will be designed to test different aspects of the code. These tests are

- Unit testing
- Integration testing
- System testing
- User acceptance testing

Some of these tests can be performed by a single programmer while other tests will require the cooperation of the entire programming team or in some of the tests will require the aid of the clients themselves.

Unit testing

Unit testing is the smallest test that the programmers can run. This test will be run on individual components or modules of the program to see if they function properly. Tests are very hard to write for yourself so the software expert and the assistant programmer would write tests for each other.

These types of tests are small and easy to write and wouldn't take very long to write and run and should be run everytime a new component is completed (this is mainly every day).

Integration testing

Once certain modules have been completed we would have to start seeing if they are capable of interacting with each other and so we will run integration tests every now and again. This type of test doesn't really have a planned time and will only run once certain modules have been completed (we have to give allowance for the possibility that some modules will take longer than others to complete).

This type of test would take a bit longer and would require the cooperation of both the programmers and would take a bit longer than unit testing to complete. This test should still be run very often though to ensure that both programmers have modules that are compatible with each other.

System testing

This is one of the bigger tests that would simulate an entire system as a test and would require all the modules to be completed and ready. This would require the full cooperation of the programmer team as they would have to simulate an entire environment.

This type of testing requires the cooperation of the entire team and would take a lot of time to set up, run and to look for any bugs or potential bugs in the system. This type of test shouldn't be run as often and should only be done once the team is confident that the project is nearing an end.

User acceptance testing

This test is one of the most risky type of test and requires the users to come in and test the system every now and again so we can see if the client is satisfied with the system. We would also be looking for any bugs they may find and also their input on what can be improved or taken out of the system.

This type of test takes a bit of time and shouldn't be done so often especially since it would take up the time of the user in addition to the people on the team.

Prototype

This will require us to create a small simulation of the system to present to the user, this is similar to user acceptance testing in the way that the user has to be satisfied with the prototype before we move on. This will be an evolutionary type of prototype so we will be continuously adding more and more functions to the system before we are ready to finish the project.

There are 3 meetings opportunities to present out prototype to the user and they will be able to have a first hand look at our work and tell us if they are happy or not with the system, they will also be able to tell us what else they are looking for in the system and if we are on the right track. These meetings with the user are crucial as our goal is to satisfy their needs for the system and we must listen carefully to what they have to say, because we can't move on without their approval and or ideas on the prototype.

Quality assurance

Quality can be assured at every step of the project and we should make it a point to remember that quality needs to be assured not only in the system itself but it should also be kept in the documentation as well. To ensure quality at every step of the project we plan to implement a few methods that are run throughout the entire team.

Quality gates

Quality gates will be used before every new phase in the development of the project. The use purpose of a quality gate would be to check the documentation of the previous stage/process to ensure all the documents are up to date.

By checking the documents at the end of every process, it will ensure that all the documents are up to date and of high quality. This is good because otherwise documents are left unrevised and sometimes out of date meaning they are obsolete.

Checklists

Checklists are useful to keep track of what needs to be done. This is a good way to check what processes have yet to be done and what code has to be tested.

We would use checklists to check that nothing was left out when coding, documenting or testing. Often when we code we would leave out certain functions but with the use of a check list we can ensure that all functions are added in.

By using a checklist to check off all the functions in a module, we would be ensuring quality in all the modules we make (so we don't have to go back and modify them later).

Audits

We have our own IT auditor that will check up on the system from time to time. The purpose of the IT auditor (as mentioned in the roles and responsibilities) is to check up on the system to ensure that the requirements of the system are met in the project. This can be done by an auditor checking the system at regular intervals.

Once an auditor does an assessment on the system he/she will create a report on the system that will inform us of systems quality and its closeness to the user's requirements. The report will be available to view at the end of every milestone (say we pass a successful integration test).

The purpose of this is to ensure that the system is keeping to the user's requirements and therefore ensuring a high quality project is delivered at the end.

Tracking, Communication, Quality Mechanisms

Tracking Mechanisms

MS Project

- Initially in Microsoft Project our Team Leader will setup a detailed breakdown of all the project milestones and their due dates
- From each milestone there will be a number of sub-tasks which will be allocated specific time during the work week to be completed. Each sub-task will also have a due date which is before the actual due date. Each sub-task will have a person assigned responsibility.
- The current Team Leader's responsibility involves updating the MS Project task list with status (as status reports are passed from Project Manager/IT Team Lead to the core management of our team).
- MS Project will then identify tasks or sub-tasks which are not on track to be completed or are running over time. They are highlighted in Red. The current team leader can then take corrective action.
- Corrective action may be to allocate more resources to the current sub-tasks, such as additional team members, more time, or other resources such as computers. If the leader cannot correct the action without causing the milestone's date to slip. Then an emergency team meeting is called and a plan is developed by the Core team to get the project back on track.
- MS Project acts as a tracking mechanism by checking our team's progress through each of the milestones, sub-tasks and helps us quickly identify if we are slipping on the final delivery date. Once this is identified corrective action can then be taken.

SVN

- Sub Version Control through Macquarie's Trac system allows a central repository for all our team's work to be stored. This means there is a central location for our current team leader to visit and check on the work progress.
- After each user has completed some work, they must commit their changes to the repository. This was set out at the initial team meeting as a protocol for our team (also acting as a means of backup for our project).
- All changes to files are monitored by the SVN system and at any time, the team leader can view which files have been updated by which user and also the content of those files.
- The revision number indicates the current version of the files uploaded, and when the repository was last updated. The team leader can also use that number to indicate when work has been completed and committed.
- SVN acts as another central tool for the team leader and team members to track the project's progress by monitoring the files committed and by whom.

Wiki

- The Wiki is a collaboration tool which will be used extensively throughout the project. Any important announcements regarding team tracking and subsequent progress will be updated to the Wiki homepage.
- The current team leader is responsible for keeping the tracking of our project status current on the Wiki. But any team member can also contribute to the status page to help give the team leader a better idea as to their current status on subtasks.

Status Reports

- As mentioned, the Project Manager, and IT Project Lead in change of each sub-task will be required to send 'status reports' to core management, to determine the current status on the milestones.
- Each status report will be a concise but descriptive report on the current status and will include the current date, the sub-task name, the milestone name, and a description. The Project Manager will be responsible for delivering this report to core management. The report may be delivered on the Wiki page (in a sub-page).

- Core management will use this information to update the MS project database, and any other subsequent tracking mechanisms which are in use.

Communication Mechanisms

Email

- Initially when the group first met, Email was the primary contact mechanism.
- However the pitfalls of Email are apparent including the complexity of managing group emails without losing valuable responses, due to a newer important Email being delivered over the top of older ones.
- Email will still be used throughout the project, but only as an alerting mechanism for urgent aspects of the project which require immediate response.
- Our primary communication mechanism will be the Wiki. Sometimes an email may be sent to point out new postings on the Wiki page.

Wiki

- Wiki has the advantage of being a fully collaborative experience, and is perfect for use throughout group work.
- It allows simultaneous viewing and editing of pages and allows great communication – as it has fewer pitfalls than group Email.
- It allows a central communication space for all team members and leaders to view, and provides the most up-to-date information – without losing anything important.

Trac SVN

- Communication through the SVN is less collaborative than the Wiki, but still allows for communication – since as other team members update their repository, they are pulling the most recent versions of the files and these files may contain comments or sections highlighted for attention by other team members.
- This aspect will be especially important for code/binary files.

Telephone

- Upon our initial meeting, we all exchanged telephone numbers in the event we lost or didn't have access to other communication mechanisms.
- Telephone would be only a backup communication mechanisms for our team

iLearn discussion boards

- The online forums available at iLearn provide a chance for our team to collaborate with other teams which are doing similar projects and a chance to share knowledge with other developers working on systems ours may be able to interface with.

Face-to-Face

- Face-to-face are in-person meetings which may occur outside of our planned official core team meetings.

Meetings

- Official team in-person meetings which occur at regular intervals, or emergency meetings
- These enable the team to collaborate and share ideas which may be too difficult to do through electronic means
- They also allow a chance to learn about other team members.
- Core Managers will organise the official meetings at a time convenient for everyone.

Stakeholder Communication Analysis
-Provides an overview of regular stakeholder communication documentation

Stakeholder	Document Name	Document Format	Contact Person	Due
Core Management	Status Report	Soft-copy via Wiki	Project Manager / IT Team Lead	Wednesday of each week
Team Member / Project Manager	Updated schedule	Soft-copy via Wiki & Alert via Email	Core Team	Thursday of each week
Project Sponsor	Status Report	Hard-copy	Customer Project Director	Friday of each week
Team Member	Daily Status update	Soft-copy via Wiki	Project Manager	Daily (Mon-Fri)
Project Sponsor	Minutes from Meetings	Soft-copy via Wiki/SVN	Customer Project Director	Within 24 hours of meeting

Communication matrix
-Identifies the communication requirements for the project

Communication Type	Objective	Medium	Frequency	Audience	Owner	Deliverable
Regular official team meetings	Review progress, plan out future tasks	In-person	Weekly	Project team (Steering, CPD, Core, PM, Lead & Team)	Current team leader	All
Wiki Status Updates	Review daily progress	Wiki	Daily (Mon-Fri)	Team Members	Project Manager	Increment 1, 2, 3 & Final
Wiki Collaboration	Communication among team members	Wiki	As required	Team members	Initiating Team Member	Increment 1, 2, 3 & Final
SVN communication	Highlight areas of program/ code which need attention	Code via SVN	As required	Team members	Initiating Team member	Increment 1, 2, 3 & Final
Unofficial Face-to-face	Track project progress, discuss issues	In-person	As required	Team members	Initiating Team member	Increment 1, 2, 3 & Final
Emergency Meetings	Allow for last minute changes to specification (change mgmt.), slipping schedule resolutions	In-person	As required (As changes are introduced)	Project team (Steering, CPD, Core, PM, Lead & Team)	Current Core team Leader	Increment 1, 2, 3 & Final
Important Project Alerts	Communicate time sensitive project alerts	Email, Telephone	As required	Team members	Current core team leader	Increment 1, 2, 3 & Final
iLearn Discussion with other teams	Collaborate with other teams / share knowledge	iLearn forums	Fortnightly	Other teams	Current core team leader	Increment 1, 2, 3 & Final

Quality Control Mechanisms

Quality control procedure table

Quality Control Activity	Procedure	Reviewer Position Responsible
Proofreading of final documents	• Check that final documents associated with deliverable match what is being delivered • Check against original specification sheet & project plan • Identify any typographical or grammatical errors • Sign-off document as complete	Core manager
Collaborate with Project Sponsor	• Check that requirements are actually what the sponsor requires by conducting own research • Check that proposed functionality meets requirements • Check that project changes can be incorporated as required	Customer Project Director
Design Control	• Check code is being written in the form required by the organisation in terms of in-code documentation / writing functions • Check functions are producing the correct output data • Oversee the construction of sample data to test functions	Project Manager & IT Project Lead
Document Control	• Check each document version follows the correct conventions • Check formatting of document is clear and follows conventions	Core Management
Inspection testing	• Checks the product is ready for acceptance testing by conducting testing with sample data • Check product is at a high standard ready for delivery	Customer Project Director
Non-conformance	• If any document or design fails QC (Quality Control), it is non-conformance • Identify what the QC issue is • Formulate a plan to resolve QC • Incorporate future testing to ensure this document/design flaw does not re-occur	Core Manager

Quality Records	• Check each quality control test has a record on the Wiki as either passed or failed, and comments about the testing • Each record should also include who conducted the test, what sample data was used, and who the test was referred to if it failed	Project Manager & IT Team Lead
Quality Audits	• Regular audits of specific aspects of the project will be performed weekly to ensure QC is at highest standard • Choose a component from Design & a component from Documentation • Check QC testing procedures meet standards • Add to QC record	Project Manager & IT Team Lead

Some mechanisms which will allow us to conduct quality control

- TRAC/ SVN – allows the current core manager to check project status, and allows the QC reviewer to access all current project material
- Quality Control section on Wiki – our team will have an area of the wiki setup which allows us to note any quality control concerns identified from the above table (or from anywhere else in the project). We can collaborate on the wiki to find a suitable solution
- MS Project – The actual time a project sub-task takes must include the time taken for a quality control review to examine and report on the quality of sub-deliverables. MS Project will help us identify any areas which have run over-time, or significantly under-time – highlighting these sections for QC review.

looking at ProjectManager as our means of managing our project, aswell as Wrike and CentralDesktop but none has been chosen as of yet.

5.2. Communications Management

The greatest threat to failure of many past projects was the lack of an effective communication. This is why it is important to devise a management plan to monitor, document as well as provide guidelines to ensure a quality standard for effective communication. Prior to the development of such a management plan, we have to first establish the number of communication channels amongst the project team members. This is done using the following formula: n(n-1)/2. We currently have 5 team members, hence the numbers of communication channels are: 5(5-1)/2 = 10. By using this formula we have calculated that there are 10 communication channels between our team members. Due to the high number of channels, the probability for miscommunication also increases; this is why the effective management of these channels is important to the success of our project.

In order to prevent communication from being a problem, our team has created an initial contract stating that each member must be available to communicate through at least three of the following methods including group meetings:
Facebook/Google group (created 8th August 2011)
E-mail
Skype
Phone calls/messages

5.2.1. Methods for Gathering and Storing Information

This section is for identifying both the formal and informal communication and how this information will be stored and shared across team members on both sides. Formal communication includes weekly status meetings and the dissemination of information through weekly status reports, revised project schedules and issues and risks lists. Informal project communications include email and phone ad-hoc transactions required to update, clarify and disseminate relevant project status information.

5.2.2. Distribution Structure

The distribution structure section of the Communications Plan identifies how the formal communication on the project will happen and who will be involved. The distribution structure contains subsections for each type of formal communication and outlines specific information for each type...Project Status Meetings, Project Status Reporting, Project Schedule, and any shared distribution or posting site such as a sharepoint site or wiki.

For each of these formal communication items, the plan identifies how each is delivered, who receives them and how often they are delivered and reviewed.

5.2.3.Formal Project Communication Matrix

The formal project communication matrix is basically a visual representation of the distribution structure for any formal project communications. This can be through the use of a graphic or table that provides the delivery team and the customer with a quick reference of the communications that happen on the project.

At a minimum, the matrix should include:

- The type of communication
- It's originator
- Who receives the communication or attends the meeting
- The frequency that the communication or meeting occurs
- And the source of the communication or meeting

5.2.4. Signoff Page

We believe that a formal signoff is important as it sets the stage for all communications and information expectations for both project teams going forward. Therefore, the final page of the Communications Plan should be a signoff page for the Project Manager and this document should be retained, managed and modified as needed (with signoff on any changes) as the project progresses and any necessary communication methods are added or changed.

5.3. Quality Control

5.3.1. Purpose and Overview

The purpose is to ensure the quality of system to be implemented in every aspect of the life-cycle development. Quality control activities monitor and verify that project deliverables meet defined quality standards.

In the following, we will describe the steps we intend to undertaken in order ensure quality such as scope, roles and roles and responsibilities, metrics and tools, review and audit plan and corrective.

5.3.2.Scope

The following are guidelines for the scope of the project:

- Identify Quality objectives
- Key project deliverables and processes to be reviewed for satisfactory quality level

- Create Quality standards
- Identify Quality control and assurance activities
- Allocate Quality roles and responsibilities
- Identify Quality tools
- Plan for reporting quality control and assurance problems

5.3.3. Roles and Responsibilities

- Project Manager: are ultimately responsible for the quality management of the entire project
- Project Team: determines how quality management will be applied to the project using various standards and tools that will be provided.
- Project Sponsor: is responsible for providing the funds and tools to allow for quality management

Meetings will be held weekly to discuss the progress of the quality management plan and all parties must attend.

Should be outlined / included in this document.

5.3.4. Metrics and Tools:

- Management Responsibility: Describes the quality responsibilities of all stakeholders.
- Documented Quality Management System: This refers to the existing Quality Procedures that have been standardized and used within the organization.
- Design Control: This specifies the procedures for Design Review, Sign-Off, Design Changes and Design Waivers of requirements.
- Document Control: This defines the process to control Project Documents at each Project Phase.
- Purchasing: This defines Quality Control and Quality Requirements for sub-contracting any part / whole part of the project.
- Inspection Testing: This details the plans for Acceptance Testing and Integration Testing.
- Non-conformance: This defines the procedures to handle any type of non-conformance work. The procedures include defining responsibilities, defining conditions and availability of required documentation in such cases.
- Corrective Actions: This describes the procedures for taking Corrective Actions for the problems encountered during project execution.
- Quality Records: This describes the procedures for maintaining the Quality Records during project execution as well as after the project completion.
- Quality Audits: An internal audit should be planned and implemented during each phase of the project.
- Training: This should specify any training requirements for the project team

5.3.5. Review and Audit Plan

5.3.5.1. Audit Criteria

Audits may be conducted by any of the following methods: reviewing existing data records/reports, attending meetings, interviewing employees or customers, conducting surveys, or participating in process improvement projects.

5.3.5.2. Audit Preparation

Prior to conducting any audit, the Auditor shall review the necessary documentation. Preparation documents may include but are not limited to: financial or technical spreadsheets, employee resume, employee training records, current project plans, current quality plans, previous audit reports, previous or current corrective action records, current or previous preventive action records, customer satisfaction or dissatisfaction records, manager requests, current or previous improvement reports, or previous surveillance audit reports. Audit checklists may be used as deemed necessary by the Auditor.

5.3.5.3. Corrective Action Plan

PURPOSE: This Plan describes IT Security Audit findings; documents responsibility for addressing the findings; and describes progress towards addressing the findings.

5.3.5.4. Audit Name:

Audit Finding No. & Agency Concurrence	Short Title	Summary	Risk	Responsible Person(s) and Due Date	Status	Status Date	Concurs: Planned Action & Status Does Not Concur: Mitigating Controls & Risk Acceptance