NSW POLICE
PROJECT

PROJECT PLAN

Police Emergency Response System

CONTENTS

1 Introduction 2

2 Project Organisation 3

2.1 Group Organisation 3

2.2 Client Organisation 4

3 Risk Management 5

3.1 Risk Identification 5

3.1.1 Technology Risks 5

3.1.2 People Risks 5

3.1.3 Organisational Risks 5

3.1.4 Requirements Risks 5

3.1.5 Estimation Risks 5

3.2 Risk Analysis 6

3.3 Risk Planning 7

3.4 Risk Monitoring 8

4 Project Resources 9

4.1 Group resources 9

4.2 Hardware 9

4.3 Software 9

5 Project Schedule 10

5.1 Gantt Chart 10

5.2 Work Breakdown 11

5.3 Cost and Time 11

5.4 Total Work Hours and Wages 12

6 Tracking 13

6.1 Version Control 13

6.2 Tracking Progress 13

6.3 Communication 13

6.4 Quality Control 14

1 INTRODUCTION

The purpose of this document is to present a detailed project plan of the development of a Police Emergency Response System. Our task is to perform an upgrade of the current NSW Police Force emergency response system.

Currently, police call centres in NSW use a paper-based system to record information from an emergency call and then transmit the information over radio.

The project is being undertaken to improve the efficiency of the current procedure used by the NSW Police Force to respond to emergency calls. The goal is to bring all the NSW call centres onto a computer-based system, where data is efficiently stored and retrieved, without the need of a paper-based file system. The transmission of emergency response information between officers will be automated and sent through text rather than over the radio.

The system will be developed using an agile project framework, and as a result the requirements may change over time as more information becomes available from relevant stakeholders.

This project plan will cover:

- **Project Organisation** , detailing the roles and responsibilities of team members and group operation and communications;
- **Risk Management**, detailing the associated risks with the project and how they will be managed;
- **Project Resources**, including the people, hardware and software required as well as how tasks will be allocated;
- **Project Schedule**, including a Gantt Chart as well as outlining the tasks, schedules and deliverables associated with the project; and
- **Tracking**, this includes how the project will be monitored in regards to ensuring and maintaining a quality system.

This information in this document will be used to facilitate effective communication between team members and to the project stakeholders, as well as a basis for how we undertaken the project.

2 PROJECT ORGANISATION

Figure 2.1 Team Organisation Table

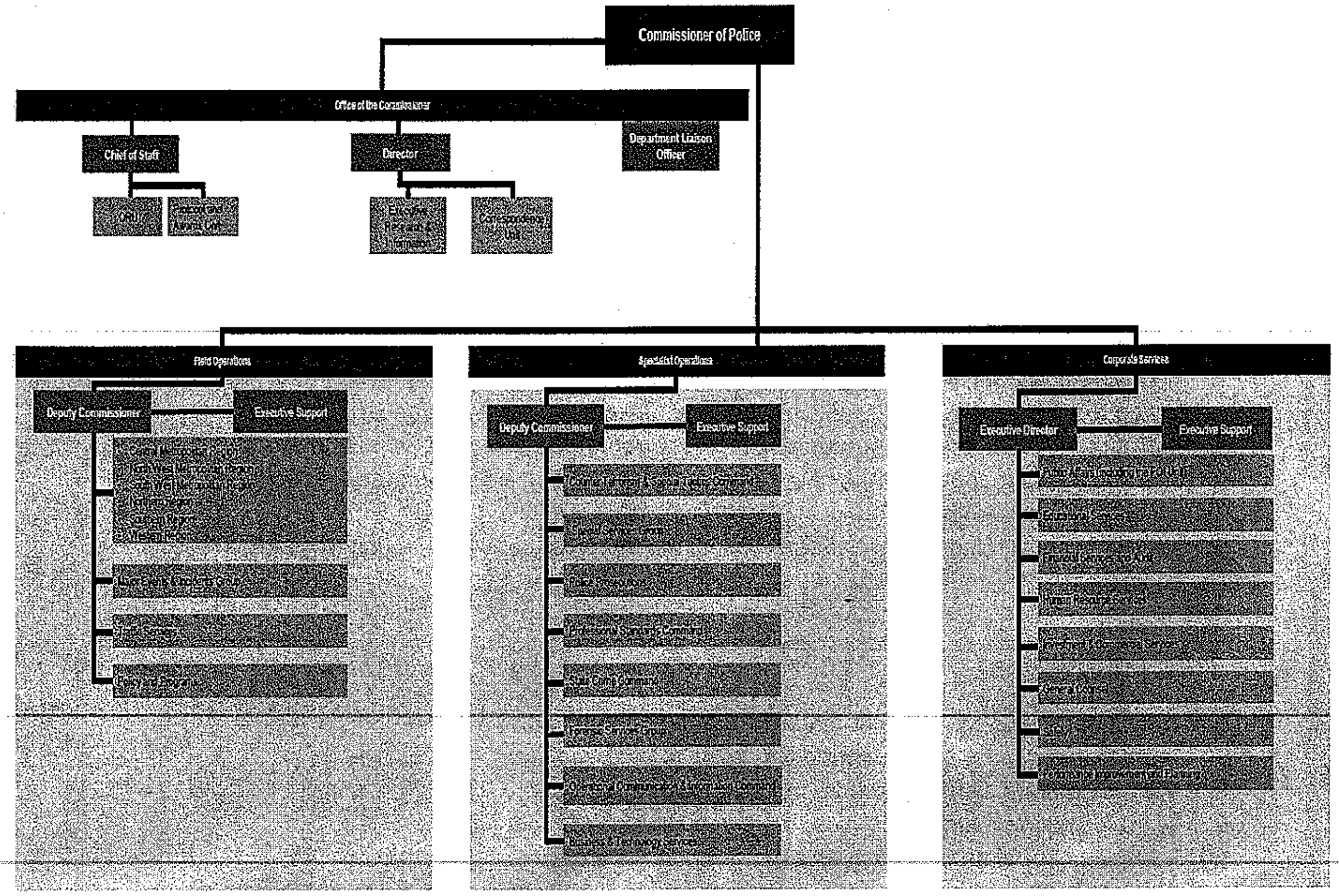
2.1 GROUP ORGANISATION

Name	Role	Responsibilities
[redacted]	Project Leader	- Lead and direct the team - Report project status to the client - Manage project activities - Ensure goals and objectives are met on time - Monitor project progress and state
[redacted]	Business Analyst	- Facilitate requirements gathering - Review deliverables - Monitor risks - Testing of the system
[redacted]	Lead Developer	- Create framework content - Developing the system source code - Testing of the system
[redacted]	System Designer	- Create framework content - Designing the core functionality of the system - Developing the system source code - Testing of the system
[redacted]	Developer	- Create framework content - Developing the system source code - Testing of the system

Above are the specific roles and responsibilities assigned to each team member. Roles and responsibilities that are common amongst the whole team include ensuring the system is developed in a quality environment; ensuring system meets all the requirements; making sure we deliver the project on time; and maintaining effective backup and tracking procedures to prevent data loss and delays.

2.2 CLIENT ORGANISATION

Figure 2.2 Client Organisation Structure



Shown above is the organisation structure of our client, the NSW Police Force. The team is contracted by the Office of the Commissioner, the primary contact being Deborah Richards, in order to perform an upgrade of the NSW Police Force emergency response system.

It is the responsibility of the project manager to lead the team in order to ensure that all the goals and objectives set from the office of the commissioner are met.

It is also the responsibility of the project manager to report the status to the client. The project manager acts as the voice of the whole team. Communication among the team is important in order to properly share information and knowledge with each other. Information and knowledge is shared across the team, using a variety of communication methods such as VOIP and TRAC, as well as by telephone. Having a small development team comprised of members from a diverse range of backgrounds and skills allows for very effective communication and decision-making, and ultimately enables the project to have a high chance of success.

While the project leader is the informal leader of the group, core decisions are made by the entire team to ensure that the team functions well together as a whole. No decisions relating to the system will be formally made, until the entire group reaches a consensus.

Each team member has strengths in different areas. This diverse range of skills enables a task to be allocated according to the most capable person. This process is decided by the whole team. This relaxed structure ensures that all team members have an equal say in decisions, which increases the chances of success.

3 RISK MANAGEMENT

We are applying four steps as part of our risk management to identify, analyse, plan and monitor risks that would affect our project, product or client's business.

3.1 RISK IDENTIFICATION

Below we have identified and described a number of risks associated with the project. The risks have been numbered from 1 to 16.

3.1.1 TECHNOLOGY RISKS

1. Our scheduled backup of project work is somehow not executed, and a computer failure occurs where all data is lost (*affects project*)
2. Hardware essential for the system is not present or malfunctioned at a critical time (such as the laptop to simulate the police vehicle's laptops) (*affects project and product*)
3. Software tool(s) to develop our system do not perform to satisfaction (*affects product*)
4. The web browser does not have the expected capabilities to make the system function to specifications (*affects product*)
5. New software is released that already has all the features that we are trying to develop (affects *business*)

3.1.2 PEOPLE RISKS

6. We suddenly lose contact with team member(s) (*affects project*)
7. Team leader cannot attend a critical meeting (*affects project*)
8. Team member(s) cannot attend a meeting (*affects project*)
9. Team member(s) cannot complete their work on time (*affects project*)
10. Team member(s) drop the unit mid-way through the project (*affects project*)
11. We find we do not have the necessary programming skills required in our team to complete a particular feature of the system (*affects project and product*)
12. Conflict between team members on a particular issue which affects the project (*affects project*)

3.1.3 ORGANISATIONAL RISKS

13. A meeting on a particular day is affected by outside interference (e.g. location of meeting is on fire and evacuated) (*affects project*)

3.1.4 REQUIREMENTS RISKS

14. Client requests a requirements change (*affects project and product*)

3.1.5 ESTIMATION RISKS

15. Size and scope of the system is underestimated (*affects project and business*)
16. Time to develop the system is underestimated (*affects project and business*)

3.2 RISK ANALYSIS

The risk analysis table below *(Figure 1)* is the results of our analysis of the identified risks to determine their probability of occurring, the impact should they occur, and the priority of monitoring them.

Figure 3.1 Risk Analysis Table

Risk No.	Risk Description	Probability	Impact	Priority
1	Our scheduled backup of project work is somehow not executed, and a computer failure occurs where all data is lost	Low	Catastrophic	1
2	Hardware essential for the system is not present or malfunctioned at a critical time (such as the laptop to simulate the police vehicle's laptops)	Low	Serious	4
3	Software tool(s) to develop our system do not perform to satisfaction	Low	Tolerable	9
4	The web browser does not have the expected capabilities to make the system function to specifications	Low	Catastrophic	8
5	New software is released that already has all the features that we are trying to develop	Low	Tolerable	16
6	We suddenly lose contact with team member(s)	Low	Serious	12
7	Team leader cannot attend a critical meeting	Low	Tolerable	10
8	Team member(s) cannot attend a meeting	Moderate	Tolerable	6
9	Team member(s) cannot complete their work on time	Moderate	Serious	7
10	Team member(s) drop the unit mid-way through the project	Low	Serious	13
11	We find we do not have the necessary programming skills required in our team to complete a particular feature of the system	Low	Serious	11
12	Conflict between team members on a particular issue which affects the project	Low	Tolerable	14
13	A meeting on a particular day is affected by outside interference (e.g. location of meeting is on fire and evacuated)	Low	Tolerable	15
14	Client requests a requirements change	Moderate	Serious	5
15	Size of the system is underestimated	High	Tolerable	2
16	Time to develop the system is underestimated	High	Serious	3

3.3 RISK PLANNING

The risk strategy table below *(Figure 2)* outlines the preventative or reactive strategy for each risk.

Figure 3.2 Risk Strategy Table

Risk No.	Risk Description	Strategy
1	Our scheduled backup of project work is somehow not executed, and a computer failure occurs where all data is lost	We will make sure all our work is backed up on separate drives even after each minor development (*preventative*)
2	Hardware essential for the system is not present or malfunctioned at a critical time (such as the laptop to simulate the police vehicle's laptops)	We will acquire the necessary technology as early as possible (*preventative*)
3	Software tool(s) to develop our system do not perform to satisfaction	We will select a number of CASE tools and use them as backup should one not satisfy our needs (*preventative*)
4	The web browser does not have the expected capabilities to make the system function to specifications	We will try other web browsers should the main one not have the expected capabilities (*preventative*)
5	New software is released that already has all the features that we are trying to develop	We will try to persuade client that only a custom-built system is enough to satisfy their unique needs (*reactive*)
6	We suddenly lose contact with team member(s)	If all contact fails, we will report to advisor (*reactive*)
7	Team leader cannot attend a critical meeting	Team leader will direct them and communicate with them another way (Skype or phone) (*reactive*)
8	Team member(s) cannot attend a meeting	Team leader will tell them what to do, and tell them of any important information from the meeting (*reactive*)
9	Team member(s) cannot complete their work on time	Team leader will assign the work to someone else who is able to (*reactive*)
10	Team member(s) drop the unit mid-way through the project (affects project)	Report to advisor (*reactive*)
11	We find we do not have the necessary programming skills required in our team to complete a particular feature of the system	We will research the problem and find a solution, else the feature may have to be removed (*preventative*)
12	Conflict between team members on a particular issue which affects the project	Resolve the conflict peacefully, reminding them of importance of project (*preventative*)
13	A meeting on a particular day is affected by outside interference (e.g. location of meeting is on fire and evacuated)	We will meet elsewhere or reschedule meeting for another time (*reactive*)
14	Client requests a requirements change	Assess impact of the change via traceability matrix, maximise effort to accommodate change (*reactive*)
15	Size of the system is underestimated	Maximise effort to accommodate change, else ask client to revise system's features (*reactive*)
16	Time to develop the system is underestimated	Maximise effort to finish on time, else ask client for extension or revise system's features (*reactive*)

3.4 RISK MONITORING

We have identified a number of indicators which will assist us in monitoring each type of risk and continually checking whether a risk is more likely to happen. We will be very mindful of the risks we have prioritised but will still monitor the low priority risks as well. Some risks cannot be effectively monitored because of their unpredictable nature, and instead we will adopt the reactive strategies that we outlined in *Figure 2*. The indicators are shown in the risk indicators table *(Figure 3)* below.

Figure 3.3 Risk Indicators Table

Risk Type	Indicators
Technology Risks	- Computers that hold our project work behave unpredictably or are slow - Necessary software is not in our possession before development begins - Necessary hardware is not in our possession before development begins - Complaints about software performance by team members
People Risks	- Staff morale is low - Team member(s) stop attending meetings or replying to messages - Poor relationships among team members
Organisational Risks	- Lack of leadership by the team leader
Requirements Risks	- Initial feedback from client is poor - Continued feedback from client is poor - Many requirements changes are being made
Estimation Risks	- We miss milestones in development - We are behind schedule according to the Gantt Chart - We continually fail to hold meetings

4 PROJECT RESOURCES

4.1 GROUP RESOURCES

Every member in our team is an essential resource that will provide valuable input for the information system. In terms of special capabilities, Michael has an associate within the local police force that has and will continue to offer beneficial information in assisting the team with information gathering. Other people that may assist us in the development of the system may include the client, students, and tutors.

In terms of resources, all team members will use the hardware and software that we have outlined below for their work (as well as their own personal computers) in the development of the system. Resources to further assist us in the project will include material found on the internet.

We will be using these resources to develop the prototype which will be demonstrated at project's end.

4.2 HARDWARE

The following hardware will be used in the development and prototyping of our system:

- Desktop PC which will simulate the operator's computer as part of the prototyping
  - Dual core CPU, dedicated GPU, 4GB RAM and a 500GB hard drive
  - Keyboard and mouse
  - Monitor
- Portable Laptop which will simulate a laptop in a police vehicle
  - Similar specifications to the desktop PC

This hardware will be used for the development and the prototype. The hardware must be fast and reliable to improve efficiency and reduce chances of data loss. We have already acquired these items.

In the deployment phase there will be two logical sides to our system:

- Server side: Implemented by an online service, it will be hosting the information system (namely the database) on a dedicated network connection. It will simulate the PERS database on a police server.
- Client side: Clients (the hardware outlined above) will connect to the database to read and write data to it. In terms of the prototype, the desktop PC will act as the operator as they interface with the database through the WebApp. The laptop will also be able continuously check for updates from other clients.

4.3 SOFTWARE

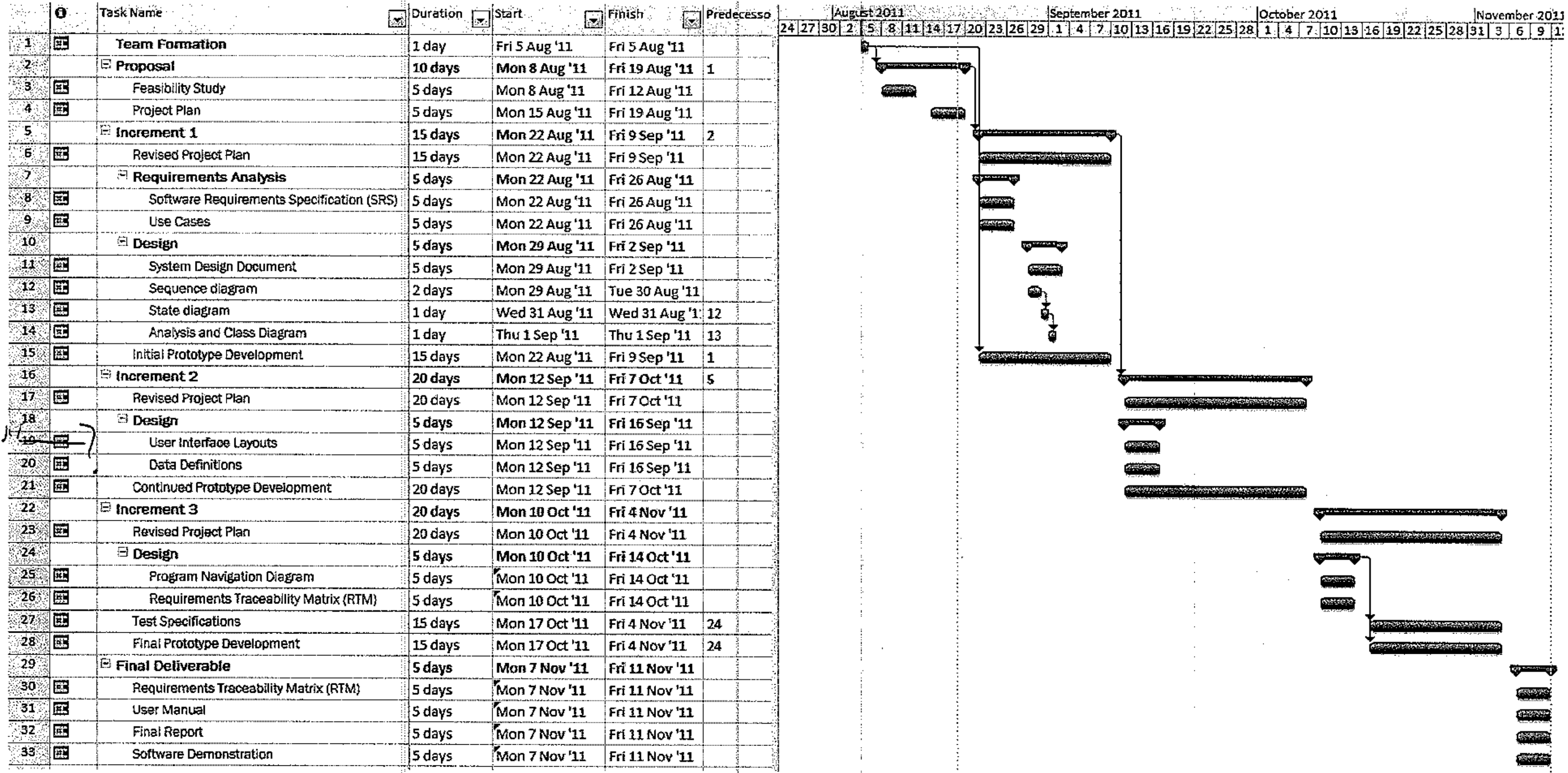
The following software will be utilised in the development and prototyping of our system:

- Microsoft Windows OS (Windows 7)
- Microsoft Visual Studio 2010 - Development IDE
- Microsoft SQL Server 2008
- Microsoft Office 2010 Suite (Word, Access, Project etc.)
- Microsoft .Net 4.0
- Microsoft IIS
- Mozilla Firefox Web Browser
- Sybase Power Designer
- TRAC (see section 6)

5 PROJECT SCHEDULE

This section outlines each of the tasks that are to be involved in the project for creating a more efficient system in the NSW Police Force Emergency Response System. The schedule also outlines the estimated starting and ending dates of the various tasks that will be completed, along with the forecasted duration of each tasks in days and hours, and also the associated team members who have been assigned to complete each task. With each increment, the schedule may change to reflect new changes or tasks.

5.1 GANTT CHART



		Task Name	Duration	Start	Finish	Predecesso
1		**Team Formation**	1 day	Fri 5 Aug '11	Fri 5 Aug '11	
2		⊟ **Proposal**	**10 days**	**Mon 8 Aug '11**	**Fri 19 Aug '11**	1
3		Feasibility Study	5 days	Mon 8 Aug '11	Fri 12 Aug '11	
4		Project Plan	5 days	Mon 15 Aug '11	Fri 19 Aug '11	
5		⊟ **Increment 1**	**15 days**	**Mon 22 Aug '11**	**Fri 9 Sep '11**	2
6		Revised Project Plan	15 days	Mon 22 Aug '11	Fri 9 Sep '11	
7		⊟ **Requirements Analysis**	**5 days**	**Mon 22 Aug '11**	**Fri 26 Aug '11**	
8		Software Requirements Specification (SRS)	5 days	Mon 22 Aug '11	Fri 26 Aug '11	
9		Use Cases	5 days	Mon 22 Aug '11	Fri 26 Aug '11	
10		⊟ **Design**	**5 days**	**Mon 29 Aug '11**	**Fri 2 Sep '11**	
11		System Design Document	5 days	Mon 29 Aug '11	Fri 2 Sep '11	
12		Sequence diagram	2 days	Mon 29 Aug '11	Tue 30 Aug '11	
13		State diagram	1 day	Wed 31 Aug '11	Wed 31 Aug '1:	12
14		Analysis and Class Diagram	1 day	Thu 1 Sep '11	Thu 1 Sep '11	13
15		Initial Prototype Development	15 days	Mon 22 Aug '11	Fri 9 Sep '11	1
16		⊟ **Increment 2**	**20 days**	**Mon 12 Sep '11**	**Fri 7 Oct '11**	5
17		Revised Project Plan	20 days	Mon 12 Sep '11	Fri 7 Oct '11	
18		⊟ **Design**	**5 days**	**Mon 12 Sep '11**	**Fri 16 Sep '11**	
19		User Interface Layouts	5 days	Mon 12 Sep '11	Fri 16 Sep '11	
20		Data Definitions	5 days	Mon 12 Sep '11	Fri 16 Sep '11	
21		Continued Prototype Development	20 days	Mon 12 Sep '11	Fri 7 Oct '11	
22		⊟ **Increment 3**	**20 days**	**Mon 10 Oct '11**	**Fri 4 Nov '11**	
23		Revised Project Plan	20 days	Mon 10 Oct '11	Fri 4 Nov '11	
24		⊟ **Design**	**5 days**	**Mon 10 Oct '11**	**Fri 14 Oct '11**	
25		Program Navigation Diagram	5 days	Mon 10 Oct '11	Fri 14 Oct '11	
26		Requirements Traceability Matrix (RTM)	5 days	Mon 10 Oct '11	Fri 14 Oct '11	
27		Test Specifications	15 days	Mon 17 Oct '11	Fri 4 Nov '11	24
28		Final Prototype Development	15 days	Mon 17 Oct '11	Fri 4 Nov '11	24
29		⊟ **Final Deliverable**	**5 days**	**Mon 7 Nov '11**	**Fri 11 Nov '11**	
30		Requirements Traceability Matrix (RTM)	5 days	Mon 7 Nov '11	Fri 11 Nov '11	
31		User Manual	5 days	Mon 7 Nov '11	Fri 11 Nov '11	
32		Final Report	5 days	Mon 7 Nov '11	Fri 11 Nov '11	
33		Software Demonstration	5 days	Mon 7 Nov '11	Fri 11 Nov '11	

Req5 Analy

5.2 WORK BREAKDOWN

Task 1 involved the formation of the group, team bonding and figuring out each other's roles and responsibilities, skills and proficiencies. (5 August 2011)

Task 2 involved barnstorming ideas on what the system will be that we will build. We researched a number of different areas of the NSW Police Force and selected the emergency response system. We then created the feasibility study and the project plan. (8 August to 19 August 2011)

Task 3 is the first of three increments. It will involve revising the project plan, and creating the Software Requirements Specification (SRS) and the use cases. We will also begin the initial prototype development. (22 August 2011 to 9 September 2011)

Task 4 is the second increment. We will be revising the project plan and continuing the development of the prototype. We will also continue the design of the user interface and data definitions. (12 September – 7 October 2011)

Task 5 is the third increment. The project plan will be revised to its final version and we will finalise the design aspects of the system, create test specifications and complete the prototype. (10 October – 4 November 2011)

Task 6 is the final deliverable. We will include the RTM, user manual, the final report and the prototype demonstration to the client. (7 November – 11 November 2011)

5.3 COST AND TIME

The NSW Police Force Emergency Response System is designated to begin on the 5th of August 2011 and will end on the 4th of November 2011. This will be including the system demonstration. The estimated work hours and cost for each member of this team varies according to each person's role and position, but we have estimated the base work hours for each member as the same.

5.4 TOTAL WORK HOURS AND WAGES

Figure 5.1 Task Hours

Task	Time (hours)
Task 1: Team Formation	2
Task 2: Proposal	12
Task 3-5: Requirements, Design, Testing, Prototype and Project Plan Updating	52
Task 6: Final Deliverables, Software Demonstration	12
Total	94

Figure 5.2 Team Wages

Member Name	Position/ Role	STD Pay/hr	OVT Pay/hr
[redacted]	Project Leader	$80/hr	$160/hr
[redacted]	Business Analyst	$50/hr	$90/hr
[redacted]	Lead Developer	$55/hr	$90/hr
[redacted]	System Designer	$55/hr	$90/hr
[redacted]	Developer	$50/hr	$90/hr

Figure 5.3 Team Total Hours and Wages

Member Name	Total Base Work Hours (hrs)	Total Base Pay ($)
[redacted]	94	$ 7,520
[redacted]	94	$ 4,700
[redacted]	94	$ 5,170
[redacted]	94	$ 5,170
[redacted]	94	$ 4,700
	Total	$ 27,260

6 TRACKING

Tracking for this project will be primarily achieved using the software package TRAC. TRAC will enable team members to share project resources and documents between each other.

6.1 VERSION CONTROL

TRAC provides an interface to a version control system of our project documents and of the system source code. This will be used to manage and keep track of project builds.

Version control is important in any software project as it allows us to revert back to a previous version of the project, which can be vital to solving problems that arise after changes and the implementation of new components to the system.

The documents will also be hosted on an external server, so should our primary hard drives crash, we will be able to retrieve the documents from TRAC.

6.2 TRACKING PROGRESS

TRAC will also provide a way for the team to be able to keep track of project progress and complete tasks according to the project timeline, allowing the project as a whole to meet its milestones and ultimately keep on schedule.

The project manager will track progress via the Gantt chart, which will be updated to reflect changes in the schedule if they occur.

6.3 COMMUNICATION

Communication between team members will be carried out through team meetings, video-conferences (VOIP) and TRAC. As TRAC is a wiki it allows for a collaborative way of sharing individual tasks with the rest of the team.

Video Conferences - the majority of team communication will be carries out using video conferencing; the team will meet weekly to discuss upcoming milestones and to discuss which team members will complete which tasks. These weekly conferences will also be to ensure that all team members haven't strayed away from project goals and to ensure that all components of a deliverable are consistent with one another.

Team Meetings - team meetings will take place weekly to ensure that not only all team members have completed their parts but also to bring all tasks together as which will be put together and checked as a team.

TRAC - TRAC will be used by team members to share their contribution with the rest of the team. This also helps with team consistency as all members will be able to view each others work which means that team collaboration can occur without the need for daily team meetings.

6.4 QUALITY CONTROL

As this project is to be completed within three months quality control will be limited to the following areas:

Error Checking – verifying that the code compiles and contains no errors. The team members writing the changes to the code will carry this out.

Stress Testing – the application will be evaluated to ensure it meets the project requirements with no critical faults, as we have a limited time frame and budget stress testing will occur in an automated environment.

Documentation – ensure that all supplied documentation covers all areas of the application, is consistent and comprehendible by an end-user.

Managing Change – when project changes arise all team members as well as the client will be notified of the changes. As we are using an agile approach the client will most likely be provided with an incremental prototype of the system and supporting documentation.