

MegaGlobalCorpSuperInc  
Project Management in Practice

Alan Casey 0721662  
JUN HE 0586374  
Eoin Murphy 9021221  
Yerenca Fernndez 11037199  
David O Neill 0813001

November 17, 2011

# Table Of Contents

<b>1</b>	<b>Project Charter</b>	<b>1</b>
1.1	Introduction . . . . .	1
1.2	Charter . . . . .	1
<b>2</b>	<b>Project Scope</b>	<b>3</b>
2.1	Scope Disclaimer . . . . .	3
2.2	Scope Statement . . . . .	3
2.3	Objectives . . . . .	3
2.3.1	Client server model . . . . .	3
2.3.2	Domain specific language . . . . .	3
2.3.3	Directory services schema . . . . .	4
2.3.4	Administrator front end . . . . .	4
<b>3</b>	<b>Work Breakdown Structure</b>	<b>5</b>
<b>4</b>	<b>Project Schedule</b>	<b>6</b>
4.1	Activities . . . . .	6
4.2	Gannt chart . . . . .	7
<b>5</b>	<b>Communications Plan</b>	<b>8</b>
5.1	Introduction . . . . .	8
5.2	Aims . . . . .	8
5.3	Target audience . . . . .	8
5.4	The communications plan . . . . .	8
5.4.1	Fyp Coordinator . . . . .	8
5.4.2	Project Supervisor Semester 1 . . . . .	8
5.4.3	Project Supervisor Semester 2 . . . . .	8
5.4.4	Student Semester 1 . . . . .	9
5.4.5	Student Semester 2 . . . . .	9
5.5	Evaluation and change . . . . .	9
<b>6</b>	<b>Risk Management Plan</b>	<b>10</b>
6.1	Introduction . . . . .	10
6.2	Risk Monitoring . . . . .	10
6.3	Risk mitigation and avoidance . . . . .	10
6.4	Qualitative and quantitative . . . . .	10
6.5	Register . . . . .	11
<b>7</b>	<b>Quality Plan</b>	<b>12</b>
7.1	Introduction . . . . .	12
7.2	Review process . . . . .	12
7.3	Quality Compliancy plan . . . . .	13
	<b>Bibliography</b>	<b>14</b>
	<b>List of figures</b>	<b>14</b>

# 1 Project Charter

## 1.1 Introduction

A project charter is a document that formally recognizes the existence of a project and provides direction on the projects objectives and management. It authorizes the project manager to use organizational resources to complete the project. Key project stakeholders should sign a project charter to acknowledge agreement on the need for and the intent of the project.

## 1.2 Charter

**Project Title:** Opening a new MegaGlobalCorpSuperInc branch in Ireland.

**Date of Authorization:** October 1

**Project Start Date:** October 1

**Project Finish Date:** October 1 2013

### **Key Milestones:**

- Office fully operational with 50 developers by April 1 2012
- First version of software released by October 1 2012
- Scale up operations to 200 developers after October 1 2013

### **Budget Information:**

The company has allocated 5 million for this project based on early budget estimates and more funds are available if necessary.

**Project Manager:** Yerenca Fernandez

### **Project Objectives:**

Our new Irish branch will be a software development house for MegaGlobalCorpSuperInc internal IT needs. It is a crucial project for our company to continue to grow in a competitive market. This is the first charter for the project and the objectives are to have the branch fully operational with 50 developers in 6 months, to release the first version of software to our internal customer by the end of year 1 and to scale up to 200 developers by the end of year 2.

### **Main Project Success Criteria:**

The branch must be fully operational with 200 developers by October 1 2013. The software must meet all specified criteria, be thoroughly tested and have its first major release within 12 months.

### **Approach:**

- Hold weekly progress review meetings with all the core project team.
- Within one month develop a clear work breakdown structure, scope verification plan and project schedule.
- Picking and setting up of Office, server and network hardware within 2 months.
- Purchase all required hardware within 3 months

- New Employee Orientation to be completed for initial employees and 50 developers by month 6
- Conduct thorough software testing per approved test plans

### **Roles and Responsibilities:**

<b>Name</b>	<b>Role</b>	<b>Position</b>	<b>Contract Information</b>
Vispi Shroff	Sponsor	CEO	Vispi.Shroff@lero.ie
Yerenca Fernandez	Team Member	Project Manager	dmz.oneill@gmail.com
Alan Casey	Team Member	Team Member	alancasey20@gmail.com
David O'Neill	Team Member	Team Member	yerenka@gmail.com
Jun He	Team Member	Team Member	ilovefjjq@gmail.com
Eoin Murphy	Team Member	Team Member	

## 2 Project Scope

### 2.1 Scope Disclaimer

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

### 2.2 Scope Statement

The intent of this project is to provide a Linux port of the Windows Group Policy Object (GPO) editor and a means to deploy these policies to a multitude of Gnus not Unix (GNU) systems through a common non-ambiguous domain specific language (DSL).

### 2.3 Objectives

#### 2.3.1 Client server model

Centralized management being the theme of this project establishes the inception of the concept of a central authority and therein a client server model. The implementation is REQUIRED to provide a client-server model that MUST exhibit the following.

1. The implementation MUST support at least two variants of linux to prove applicability (Redhat and Suse).
2. The design MUST support the ability to be extended to support a multitude of different systems with varying revisions of supporting packages.
3. The implementation SHOULD use the Practical Extraction and Reporting Language (PERL) for highest availability possible.
4. The server is REQUIRED to scale from two test clients to hundreds of clients.
5. This SHOULD be achieved via the Master Slave Design Pattern or similar pattern.
6. A master with multiple slave servers implementation is RECOMMENDED.
7. A compiled language MUST NOT be used as updates to the client SHALL be sent via the server which will be deployed on varying architectures.
8. The Server SHOULD support both Push and Pull; at least MUST implement support for pull requests.

#### 2.3.2 Domain specific language

The domain specific language SHOULD exhibit the following characteristics.

1. The domain specific language MUST be extensible.
2. As new modules SHALL be deployed to the client, these Modules MAY provide hooks to the interpreter which it MUST accept.
3. The Domain Specific Language(DSL) SHOULD be implemented as a hybrid language.
  - The interpreter SHALL parse the Domain Specific Language(DSL) and process it.
  - The interpreter SHALL execute embedded general purpose language instructions defined in the Client server model objective.

### **2.3.3 Directory services schema**

The directory services schema **MUST** provide a means of representing and storing the following elements.

1. Files, Permissions, Lvalue and Rvalue of elements within configuration files.
2. Services and with common names, which **MUST** be translated by the client into the distribution specific names.
3. A representation Organizational groups in hierarchy fashion compared to active directory **MUST** be implemented.
4. Computer objects **SHALL** be recognized by a globally unique identifier ( GUID ) represented as a 32-character hexadecimal string.

### **2.3.4 Administrator front end**

The Administrator Group Policy editor **SHOULD** exhibit the following characteristics.

1. **MUST** provide the ability of creating organizational units.
2. **MUST** provide the ability to move computers between organizational units.
3. **MUST** Implement a hierarchy of organizational units where by policies **MAY** be inherited and overridden where applicable.
4. **MUST** provide a means to modify policies.
5. On application or change of a policy the server **MUST** generate new domain specific language script.
6. The front end **MUST** provide a means of importing existing policies in the form of the DSL.
7. The User Interface **SHOULD** implement the ability to push updates to the client.

### 3 Work Breakdown Structure

#	Major Milestones	#	Minor Milestones	#	Constituents	Description
1	Project Initiation	1	Project conception			Formalize the idea and define rough idea of scope.
		2	Project acceptance			Describe the project t potential supervisors and get one to sign off on it.
		3	Project submission			Sign documents and submit them to the fyp office.
2	Evaluation Phase	1	Develop risk plan	1	Identify risks	A Identify the risks in the project description.
				2	Analyze risks	Use these risks to develop scope and intended work.
				3	Document risks	Document these risks for the report.
		2	Plan for quality	1	Define quality requirements	Define the areas where non functional requirements should be catered for.
				2	Define procedures for quality	Design timelines for peer review, walkthroughs and inspections.
				3	Document quality management plan	Document these meetings for the quality plan.
		3	Define Scope	1	Requirements capture	I identify the needs and conditions for project success.
				2	Develop strategy plan	I identify the miles stones and how the project should process to meet these expectations.
				3	Research previous experience	Research previous projects in the area and their outcomes.
				4	Define scope	Define scope to better realize the expectations of the product.
				5	High level work breakdown	Break down the project into constituent parts.
				6	Deliverables and acceptance criteria	Identify milestones and required fidelity prototyping.
				7	Document assumptions	Document any assumptions made during the design process.
3	Design Phase	1	Design back end	1	Client framework	Design the client framework that handles the interpretation of the DSL and the distribution specific actions.
				2	Client	Design the Client to connect the the server for updates.
				3	Server	Design the Server to handle client connections.
				4	DSL specification	Design concepts of the DSL specification and document structure.
		2	Design front end	1	Policy editor	Design the visual policy editor.
4	Development Phase	1	Back end	1	Client framework	Implement interpreter and class representations.
				2	Client	Implementation.
				3	Server	Implementation.
				4	DSL specification	Implement DSL specification.
		2	Front end	1	Policy editor	Implement the visual policy editor.
5	Testing phase	1	Assure quality	1	Participate in walk-throughs	Walk through code with supervisor.
				2	Conduct inspection and audits	Conduct finite analysis of code .
				3	Conduct project reviews	Conduct project review and timelines goals and progress.
				4	Document tests	Define test cases and results.
6	Deployment phase	1	Installation			Conduct installation test.
		3	Demo			Conduct high level presentation.

# 4 Project Schedule

## 4.1 Activities

#	Info	Title	Given Plan ned Work	Expected Start	% Complete	Flag Status
0		▼ Linux Group Policy		05/09/2011	23%	
1		▼ Project Initiation		05/09/2011	100%	
2		Project Conception	5 days ?	05/09/2011	100%	
3		Project Acceptance	4 days ?	09/09/2011	100%	
4		Project Submission	1 day ?	15/09/2011	100%	
5		▼ Evaluation Phase		19/09/2011	100%	
6		▼ Develop Risk Plans		26/09/2011	100%	
7		Identify risks	1 day	26/09/2011	100%	
8		Analyze risks	1 day	27/09/2011	100%	
9		Document risks	1 day	28/09/2011	100%	
10		▼ Plan for Quality		26/09/2011	100%	
11		Define quality requirements	1 day	26/09/2011	100%	
12		Set up standards and procedures for quality management	1 day	27/09/2011	100%	
13		Document quality management plan	1 day	28/09/2011	100%	
14		▼ Define Scope		19/09/2011	100%	
15		Requirements capture	8.5 days ?	19/09/2011	100%	
16		Develop strategies and plans	1 day ?	19/09/2011	100%	
17		Conduct Planning Workshop	1 day ?	20/09/2011	100%	
18		Research previous experience	1 day ?	21/09/2011	100%	
19		Define scope	1 day ?	22/09/2011	100%	
20		Develop high level work breakdown	1 day ?	23/09/2011	100%	
21		Specify deliverables and acceptance criteria	1 day ?	26/09/2011	100%	
22		Document assumptions	1 day ?	27/09/2011	100%	
23		End of evaluation		28/09/2011	100%	
24		▼ Design Phase		03/10/2011	46%	
25		▼ Design Back end		03/10/2011	60%	
26		Client Framework	12.5 days ?	03/10/2011	100%	
27		Client Design	12.5 days ?	10/10/2011	100%	
28		Server Design	12.5 days ?	19/10/2011	50%	
29		DSL Specification	15 days ?	26/10/2011	0%	
30		▼ Design Front End		02/01/2012	0%	
31		Policy Editor	15 days ?	02/01/2012	0%	
32		Specification complete		22/01/2012	0%	
33		Project Presentation		14/10/2011	50%	
34		▼ Development Phase		17/10/2011	4%	
35		▼ Back End		17/10/2011	7%	
36		Client Framework	20 days ?	17/10/2011	20%	
37		Client	20 days ?	31/10/2011	0%	
38		Server	20 days ?	14/11/2011	0%	
39		▼ Front End		16/01/2012	0%	
40		Policy Editor	30 days ?	16/01/2012	0%	
41		End of Development		01/03/2012	0%	
42		▼ Testing Phase		16/11/2011	0%	
43		▼ Assure Quality		16/11/2011	0%	
44		Participate in walk-throughs and reviews	55 days	16/11/2011	0%	
45		Conduct inspections and audits	12 days	16/01/2012	0%	
46		Conduct project reviews	14 days	01/02/2012	0%	
47		Documentation	13.5 days ?	15/02/2012	0%	
48		▼ Deployment Phase		27/02/2012	0%	
49		Installation	2.5 days ?	27/02/2012	0%	
50		Launch finished	3 days ?	29/02/2012	0%	
51		Demo	1 day ?	10/04/2012	0%	

Fig. 4.1: Activity schedule



## 4.2 Gantt chart

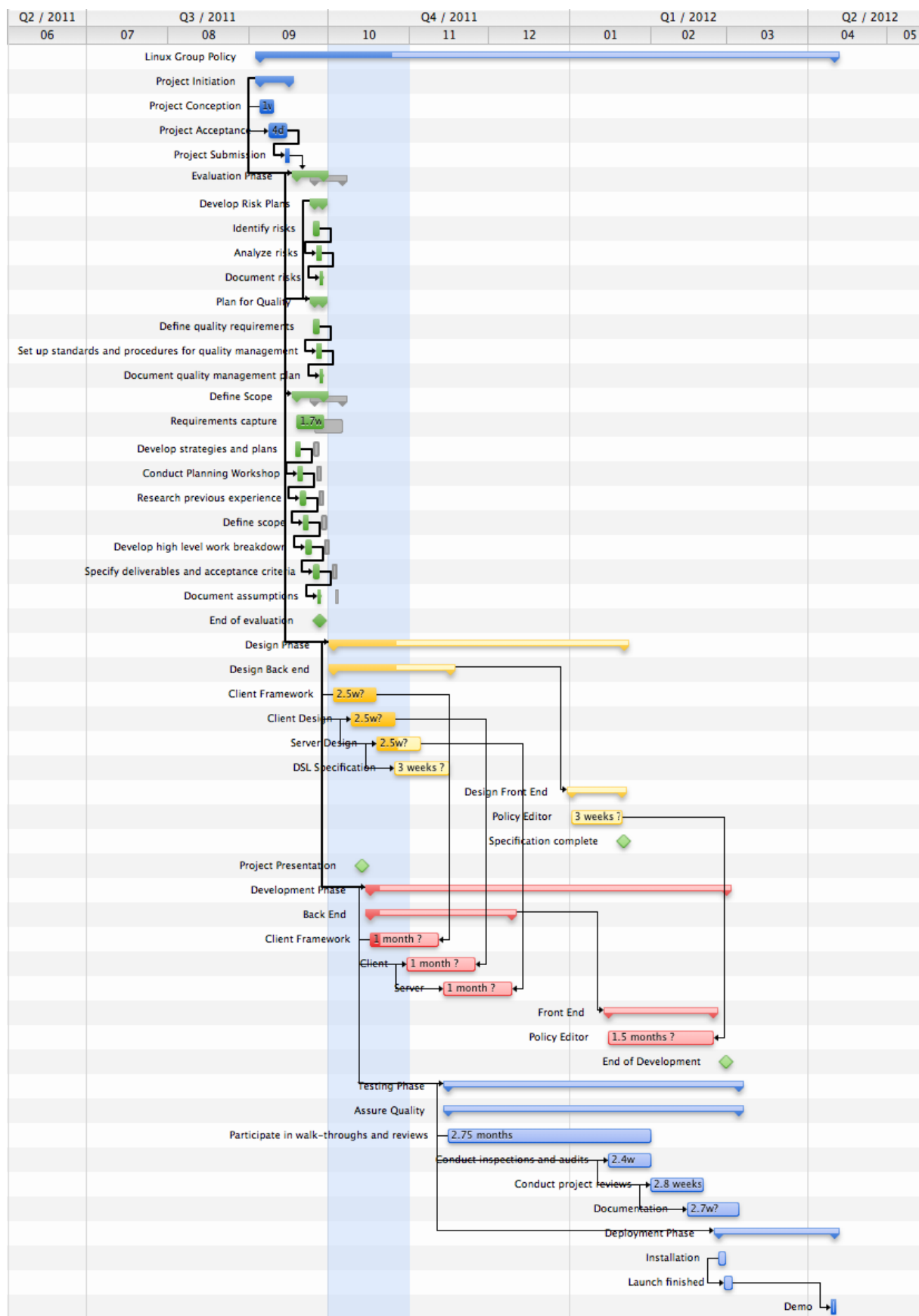


Fig. 4.2: Gantt chart

# 5 Communications Plan

## 5.1 Introduction

This communications plan presents the information pertinent to Student and the relevant parties during the process of the final year project. It outlines roughly how each party is responsible for administrating and delivering operational programmes.

## 5.2 Aims

The aim of this communication plan is to present an overview of the series of events or programmes as part of the final year project project management plan. This overview will highlight the roles and contributions of the parties involved and the timelines for deliveries therein.

## 5.3 Target audience

The targeted audience or stakeholders for the communications plan include the following

- The final year project coordinator
- Supervisor
- Second reader
- Student
- Project manager reviewer

## 5.4 The communications plan

### 5.4.1 Fyp Coordinator

Activity	Stage	Finish Date	Medium	Responsibility	Audience	Frequency
Fyp Guidelines	Understand	Week 12 Semester 1	Meeting	Coordinator	Students	weekly
Fyp Guidelines	Understand	Week 8 Semester 2	Meeting	Coordinator	Students	weekly

### 5.4.2 Project Supervisor Semester 1

Activity	Stage	Finish Date	Medium	Responsibility	Audience	Frequency
Meet students	Envision	Week 1	Meeting	Students	Supervisor	once
Accept students	Acceptance	Week 2	Document	Students	Supervisor	once
Supervise Students	Development	Week 12	Meeting	Students	Supervisor	weekly
Agreement Form	Submission	Week 10 Thursday	Document	Student	Supervisor	once
Marking Scheme	Acceptance	Week 10 Friday	Document	Students	Supervisor	once
Interim report	Acceptance	Week 16 Tuesday	Document	Students	Supervisor	once

### 5.4.3 Project Supervisor Semester 2

Activity	Stage	Finish Date	Medium	Responsibility	Audience	Frequency
Supervise students	Envision	Week 10	Meeting	Supervisor	Students	once
Draft Report	Acceptance	Week 7 Thursday	Document	Students	Supervisor	once
Product Acceptance	Acceptance	Easter break Monday	Email	Students	Supervisor	once
Report Submission	Acceptance	Week 11 Thursday	Document	Students	Supervisor	once
Cut-off Submission	Acceptance	Week 12 Thursday	Email / Document	Students	Supervisor	once

#### 5.4.4 Student Semester 1

Activity	Stage	Finish Date	Medium	Responsibility	Audience	Frequency
Project conception	Envision	Week 1	Meeting	Student	Potential Supervisor	once
Project acceptance	Confirm	Week 2	Meeting	Student	Supervisor	once
Project submission	Submission	Week 2 Thursday	Document	Student	Fyp office	once
Develop risk plan	Define	Week 3	Meeting	Student	Supervisor	once
Plan for quality	Evaluate	Week 4	Meeting	Student	Supervisor	once
Design analysis 1	Design	Week 5	Meetings	Student	Supervisor	weekly
Presentation	Presentation	Week 6 Friday	Meeting	Student	Supervisor Panel	once
Agreement Form	Submission	Week 10 Thursday	Document	Student	Supervisor	once
Marking Scheme	Submission	Week 10 Friday	Document	Student	Supervisor	once
Development analysis 1	Develop	Week 12	Meeting	Student	Supervisor	weekly
Interim report	Submission	Week 16 Tuesday	Document	Student	Supervisor	once

#### 5.4.5 Student Semester 2

Activity	Stage	Finish Date	Medium	Responsibility	Audience	Frequency
Design analysis 2	Design	Week 2	Meetings	Student	Supervisor	weekly
Development analysis 2	Develop	Week 6	Meeting	Student	Supervisor	weekly
Draft Report	Submission	Week 7 Thursday	Document	Student	Supervisor	once
Demo	Presentation	Week 10 Wednesday	Presentation	Student	Public presentation	once
Product Submission	Submission	Easter break Monday	Email	Student	Supervisor	once
Report Submission	Submission	Week 11 Thursday	Document	Student	Supervisor	once
Cut-off Submission	Submission	Week 12 Thursday	Email / Document	Student	Supervisor	once

### 5.5 Evaluation and change

As the deadline approaches this communications plan is subject to change. Communications between the supervisor and the student may be subject to change in order to ensure a quality product delivery on time.

# 6 Risk Management Plan

## 6.1 Introduction

As the product being delivered is highly conceptual, a new and unique product in the problem domain, chance is a significant element leading to risk of uncertainty of delivery. The purpose of this risk management plan is to identify risks and develop strategies to help mitigate these risks.

## 6.2 Risk Monitoring

During the ongoing meetings with the supervisor the main risks prevalent should be readdressed, discussed and the status of each risk readjusted. It is the responsibility of the risk manager(s) to provide these risk status updates, the trigger conditions and risk response.

## 6.3 Risk mitigation and avoidance

As more risk become apparent during the development these should be addressed immediately and discussed with the supervisor. These risks should be qualified by the student and avoidance and mitigation strategies should be developed with the supervisor.

## 6.4 Qualitative and quantitative

### Risk Likelihood Rating

Low	Medium	High	Extreme	Not Assessed
L	M	H	E	NA

### Risk Impact Rating

Grade	Mitigation action	Description
N	No Action	No action required unless grading increases over time
A	Alternative action	Have alternative action plan
P	Produce Minimum	Mitigation plan to provide minimum accepted
I	Implemented Execution	Identified and avoided during execution
C	Catastrophic	Identified and avoided on commencement as a priority

### Combined impact / Likelihood

Likelihood	Impact				
		Low	Medium	High	Extreme
	Low	N	N	A	C
	Medium	A	P	I	C
	High	I	I	C	C
	Extreme	C	C	C	C

### Grade change assessment

Identifier	Description
new	New Risk
—	No change
↓	Decreased risk
↑	Increased risk

## 6.5 Register

11

id	Description	Project Impact	Likelihood	Impact	Combined	Grade change	Review Date	Mitigation actions	Responsibility	Timeline
1	Supervisor unavailable	Poor planning and delivery	L	H	A	↓	Weekly	Identify next 2 milestones and required work	Student	Weekly
2	Work schedule interruptions by other academic deadlines	Delayed product delivery grade penalized	M	H	C	↑	Weekly	Plan in advance Don't schedule work during weeks 7-12 of semester 1 & 2	Student	Weekly
3	Failure to define requirements	Poor scope definition unattainable goals	L	M	N	↓	21/1/2012	No action required at this time	Supervisor Student	3/1/2012
4	Failure to define good scope Unattainable goals and expectations	Product that does not meet the requirements, quality expected	M	H	I	↓	21/1/2012	Hammer out scope with supervisor and plan for second reader expectations	Supervisor Student	1/2/2012
5	Poor design	Software that does not exhibit quality attributes	M	H	A	—	Ongoing	Plan for change in grading scheme rebalance report / code awarding marks	Student Supervisor	2/2/2012
6	Poor implementation	Software that does not realize the design	L	L	N	New	Ongoing	Refactor design as part of the iterative development	Student	1/3/2012
7	Technology cacophony Too many technologies to learn	Danger of spending too much time on one area	M	M	P	New	Ongoing	Produce the minimum requirements for demo day focus on delivery of written report expectations quantify attainable scope	Student Supervisor	1/3/2012
8	Presentation Poor explanation of project	Failure to show merit of project to supervising panel, Loss of 10% of overall marks	L	M	N	↓	Week 6	Properly define scope, applicability and intent	Student	completed
9	Agreement form submission	Failure to submit form on time could result in project failure	L	E	C	↑	Week 10	Make reminder and schedule agreement form signage with supervisor	Student	Week 10
10	Marking scheme	Marking agreement to be submitted	L	L	N	↑	Week 10	Make reminder and schedule agreement form signage with supervisor	Student	Week 10
11	Interim report	Failure to submit interim report on time will result in 10% loss	L	H	C	↑	Week 13	Make reminder and have pre interim report review meeting	Student	Week 14
12	Draft report	Failure to submit interim report on time will result in 10% loss	L	H	C	new	Week 6(2)	Make reminder and have pew draft report review meeting	Student	Week 7(2)
13	Demo day Product Launch Presentation	Result of loss of marks, if not properly presented	L	L	N	new	Week 9(2)	Have meeting with supervisor to showcase product	Student	Week 10(2)
14	Product submission	Penalty for late submissions	L	L	N	new	Week 8(2)		Student	Week 9(2)
15	Report Submission	Penalty for late submissions	L	L	N	new	Week 10(2)		Student	Week 11(2)

# 7 Quality Plan

## 7.1 Introduction

The project must comply with all of the requirements described in project requirements specification. The projects conformity to software requirements specification will be checked throughout the lifecycle as defined in the rational unified process. Upon the supervisors verification that all of the tests have been satisfactorily completed / passed, the project is considered to be of satisfactory quality. Meaning that the project complies with all requirements and is accepted by the supervisor.

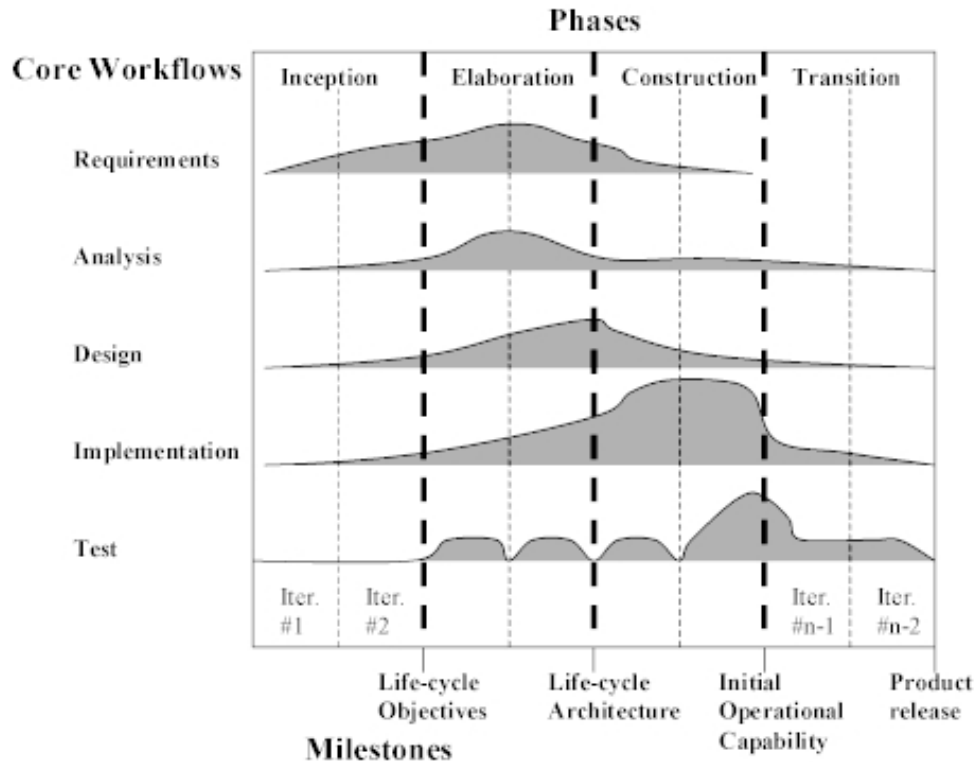


Fig. 7.1: Rational Unified Process

## 7.2 Review process

The communications plan outlines the meeting times when software reviews are scheduled to take place. The software reviews are in place to detect any defects in the current fidelity prototypes during its construction. The determination of new enhancements, or regressions should be documented and the risk register updated accordingly. This should be completed for each of the 4 individual components as outlines in the objectives.

The supervisor in conjunction with the student will attend each review, and critique software prototypes to ensure that the maximum number of possible defects are accounted for, moreover identify risks in the current direction.

This review process is subject to the work completed, accounting for disruptions in the work scheduled due to other academic commitments. The risk and schedule log should to addressed and adjusted accordingly based upon work interruptions. Each meeting will consist of no more than a one hours. The current prototype should be provided in advance of this meeting. Any enhancement found to be too difficult or unnecessary for product completion will be noted by the student.

### 7.3 Quality Compliancy plan

Deliverable	Quality Event	Quality Materials	Quality Metrics	Purpose	Presenters	Validators	Resolution
Preliminary Project Conception	High level Analysis	Project template Conformance with fyp research expectations	Area of research shows academic value	Compliance with academic expectations prior to submission.	Student	Supervisor	Re-scope project
Project Description Finalization	Accept project	Academic value	Timelines and expectations are realistic	Project submittal	Student	Supervisor	
Requirements	Business modeling Inception	Support for non functional requirements	Each requirement tracked back to project goal	Support in Definition scope	Student	Supervisor	Redefine requirements
Scope	Scope definition Inception	Properly defined scope	Clearly state whats in / out	identify clear objectives	Student	Supervisor	Redefine scope requirements
Objectives	Objectives inception	Broken down objectives that realize the scope	Objective components realize scope	Define work breakdown structure	Student	Supervisor	Redefine Objective scope
Work breakdown structure	Process inception	Each objective broken down	Clear milestones and finite process	Define Milestones and schedule	Student	Supervisor	Redefine Objective scope
Schedule	Schedule delivery inception	Timelines or other chart ( gannt )	Milestones WBS Dates	Project progress analysis	Student	Supervisor	Redefine structure milestones
Design	Analysis and design Elaboration	UML Diagrams	Class / Sequence diagrams	Realize requirements	Student	Supervisor	
Test Cases	Testing Construction	Test harnesses to test business rules	Code passes tests	Meet business rules	Student	Supervisor	Redefine design
Change Request	Business rule Conformance	Change request	Clear requirement change	Conformance with expectations	Supervisor	Student	Redefine scope requirements design
Implementation	Milestones Construction	Code	Code passes test cases	Meet design requirements	Student	Student	Fix bugs
Demo	Deployment	Deployment test harness	Product realizes requirements	Show proof of concept	Student	Supervisor	

# List of Figures

4.1	Activity schedule . . . . .	6
4.2	Gantt chart . . . . .	7
7.1	Rational Unified Process . . . . .	12