PROJECT PLAN
FOR
VOLUNTEER MANAGEMENT SYSTEM (VMS)

CLIENT	**The Client**
PROJECT	**Volunteer Management System (VMS)**
TITLE	**Project Plan**
VERSION	**1.0**
DATE	**02 Apr 2011**

ORIGINATOR

Name / Signature Dio Phung

APPROVAL

Name / Signature Dio Phung

Institute of Systems Science
25 Heng Mui Keng Terrence
Singapore 119615

Document Reference
VMS/PM-1.2.1/1

Acceptance

Customer Signature	Customer Signature
Name & Designation	Name & Designation
Date & Company Stamp	Date & Company Stamp

Revision History

Date	Version	Description	Author
02 Apr 2011	1.0	Initial release	Dio Phung

Table of Contents

1. Introduction

The target customer is poverty elimination NGO. It has grown tremendously over the years and is attracting more and more volunteers both domestically and internationally. As it grows, there is an increasing need to coordinate and manage its volunteers, especially international volunteers, as they require more administrative and logistic efforts. The current ways of manually handling documents and volunteers requires a lot of works and limiting the organization efficiency.

From the situational analysis and problems identified, the team proposed an integrated system with the aim of addressing the problems of volunteer communication, documentation, recruitment and retention.

The project serves as the required MTECH project for students in ISS to complete the Master of Technology (Software Engineering) course.

1.1 Purpose

The purpose of this document is to provide a Project Plan – which define the generic strategy for successfully deliver all required deliverables for MTECH project on schedule.

The document will:

- Specify each team member's role and responsibility
- Provide the team with a plan for the activities that they are to perform
- Specify the deliverables the team will produce
- Indicate the required efforts for each tasks against the project timeline

1.2 Audience

The intended readers of this Project plan are the project team.

1.3 Purpose

The Project plan consists of these major parts:

- Work Breakdown Structure
- Project Effort Estimation
- Project Schedule and timeline
- Project Deliverables
- Project structure and staffing

1.4 References

To fully understand the background to this project, the reader should also refer to the project proposal (reference VMS/W-9.2).

2. Project Structure and Staffing

Below is the project structure and staffing – each resource will be assigned to a specific role. During the project timeline, they are expected to perform as defined in the role description. However, resource allocation can be modified and additional tasks may be assigned to each resource depends on the project needs.

Role	Responsibilities
Project Manager – Dio	• Steer the development of the project. • Monitor progress on a monthly basis through progress meetings. • Responsible for managing the project on a week–to-week basis by reviewing progress against the plan and instituting appropriate action. • Liaise with users on key project matters. • Responsible for reviewing and approving all deliverables produced by the team. • Ensure timely project deliveries, manage and mitigate risks, revise and track risk management at all time. • Liaise with client on the sign off and reporting of progress.
Quality Manager – Peishan	• Provide the structure and content of the deliverable project documents. • Assure that projects perform to the level of quality defined in the Quality Plan. • Advise on matters of quality relating to the Project Processes and Practices. • Perform general project administration activities. • Assist in technical design and system specification. • Under take the software development.
Technical Lead – Zaw Htet	• Provide guidelines and directions on all technical aspects of the project. • Drive the Architectural Proof of Concept to determine the architecture framework for system. • Provide a blueprint application framework based on Architectural Proof of Concept for one significant use case for the development team's reference. • Document best practices for the development team to ensure

	a consistent architectural is adopted throughout system. • Work with and report to the Project Manager on technical aspects of system. • Guide the whole team to rectify any design deficiencies discovered during the entire course of the project development life cycle. • Responsible for the overall design of the project. • Under take the software development. • Responsible for installation of software at client side.
Development Lead – Feng Yan	• Responsible for prototyping, system specification, user and programmer documentation. • Support in installation and user training activities. • Supervise the team members in development and other project activities. • Represent the project team in technical discussions with client on related modules. • Report progress, issues and risks at internal project meeting. Prepare monthly team progress report. • Under take the software development.
Test Lead & DBA – Jifa	• Responsible for Functional Test and Performance / Load Test. • Work with Project Team and the Client to finalize scope of the automated functional test and Performance/Load Test. • Manage the Test Teams in providing the required services. • Work with and report to Project Manager on test activities for test teams. • Propose Test Strategy, Reporting and acceptance on areas covered. • Analyse, prepare and present Test Results. • Assist in user requirements specification. • Under take the software development. • Review database design. • Administer, set-up and manage the database guidelines. • Perform database tuning and advise project teams on database design related issues.

	• Setup database backup and recovery operations and procedures. • Perform database backup and restore as and when required. • Assist in generating test data or loading if applicable.
Business Analyst Lead – Thida	• Responsible for undertaking user requirements specification, system specification, testing, user documentation and user training activities. • Record meeting minutes for the relevant meetings with the users. • Assist in software testing. • Under take the software development.
Business Analyst – Hazel	• Responsible for undertaking user requirements specification, system specification, testing, user documentation and user training activities. • Record meeting minutes for the relevant meetings with the users. • Assist in software testing. • Under take the software development.

Table 1 – Project Structure and Staffing

3. Project Approach

The VMS project will follow Waterfall SDLC model through 7 phases starting from Project Planning to Final Project Report.

In each phase, there will be a number of activities to be performed and required deliverables to be produced. Each phase will also involve different resources depends on the requirement in that phase – the details will be presented in Work Breakdown Structure (WBS) in section 4.

Project Manager and Quality Manager will be involved in all phases of the project for overseeing the activities and qualities of the deliverables.

Each phase in the VMS project will be discussed in details as below.

3.1 Project Planning

To initiate the project, the team will:

- Produce a formal Project Plan (this document) to define the strategy for the project team to successfully produce all required deliverables.

- Produce a Quality Assurance Plan beside the Project Plan – this document will provide guiding principle for the team to ensure they will deliver quality product.

- Set up a project filing system to store all management and technical documents of the project.

The Project Manager, Quality Manager will be involved in this phase. At the end of this activity, they will also prepare first Project Presentation and Quality Audit check.

3.2 Requirement Analysis

To initiate the project, the team will:

Due to the nature of the project, one member of the project team, who will act as the putative User Manager in the requirement analysis phase, has replaced the actual customer in the proposal.

The Business Analyst team (consist of one Business Analyst Lead and one Business Analyst) will complete this phase by:

- Working with the putative user to identify the user requirements;

- Producing a user requirements specification; and

- Conducting additional research and verification with the putative user to resolve any remaining requirements issues.

The Business Analyst Team and putative User Manager will be involved in this phase.

3.3 Analysis Modelling

3.3.1 Analysis

The next phase will be analysis and design. In this phase, the team will:

- Produce Use Case Realization Report (Analysis)

- Produce high level Design Specifications.

- Produce Transition Strategy from Analysis to Design

The Technical Lead, Quality Manager, and development lead will be involved in this phase. At the end of this activity, Project Manager and Quality Manager will also prepare second Project Presentation report and Quality Audit check.

3.3.2 Prototyping

Based on the high-level user requirements specification, a series of prototypes (approximately 2 rounds) will be produced by using HTML pages and wireframe.

Each prototype will be demonstrated to the putative UM (who was appointed by the team) to determine the acceptability of screen layouts, report formats and methods of operation (menus, function keys, etc).

Because of each prototype demonstration, changes will be agreed to be included in the next prototype. Hence, each prototype builds on the functionality of the previous until it is agreed with the putative UM that the last prototype represents the full detailed requirements for the software to be developed. These requirements will be documented in analysis and design specifications.

The development team will be involved and responsible to produce the prototype of VMS.

3.3.3 Design Modelling

After Analysis and Prototyping phase is completed, the team will proceed to produce detailed design specifications through a series of activities:

- Defining detailed objects specifications
- Defining detailed sequence diagrams

The Technical Lead and Development Lead will be involved in this phase.

3.3.4 Software Implementation

When the final prototype has been produced and requirements for the software have been agreed, the software will be implemented by:

- Produce detailed level design specification – this document will include all sequence diagrams and object specifications.
- producing user manual
- producing programmer documentation / guide;
- specifying the algorithms and other detailed processes to be implemented;
- develop the system;
- undertaking unit test;

The development team is the owner of this phase.

3.3.5 System Integration Testing

When the development and unit testing of the system are completed, the test team will start the testing phase by:

- defining test scripts and test data
- preparing testing environment
- performing system test;
- producing test log;

Project Manager, Quality Manager and the test team will be the owner of this phase. Notes: There will be no deployment activities as agreed with the putative User Manager.

3.3.6 User Acceptance Testing

After the System Integration Test is done and verified as passed, the team will proceed to perform UAT by:

- Preparing test scripts and test data for UAT
- Preparing the UAT environment
- Performing UAT and record the test results
- Performing corrective actions needed to pass UAT

The Project Manager, Putative User Manager and the test team will be involved in this phase. After the UAT is completed successful, the team will document the test results and the system is considered accepted by users. Due to the nature of the project, there will not be actual user sign-off.

3.3.7 Project Report and Closure

After the development and testing of the system is completed, the team will finish the project by:

- Produce the system user guide
- Produce End of Project report
- Prepare third Project Presentation and Audit Check.

4. Work Breakdown Structure (WBS)

The WBS will specify the list of tasks that need to be performed to successfully complete the VMS project. Together with the list of tasks, WBS also identifies the resources assigned to each tasks and the required deliverables of that activity.

The WBS is done by following these steps:

- Identifying all project deliverables
- Forming the needed tasks to deliver these items, arranging the tasks with respect to RUP methodology and Waterfall SDLC
- Assigning the corresponding resources based on the required deliverables,

4.1 WBS Legends

Roles	Resources
Project Manager - PM	Dio
Quality Manager - QM	Peishan
Technical Lead - TL	Zaw
Development Lead - DL	Feng Yan

Business Analyst Lead - BAL	Thida
Business Analyst - BA	Hazel
Test Lead - TSL	Jifa
Developer - DEV	All members
Putative User Manager - UM	Dio, Peishan

Table 2 – WBS Legends (Roles)

Deliverables	Abbreviation
Design Specifications	DS
Functional Specifications	FS
Project Management Plan	PMP
Project Report	PR
Prototyping Study Report	PRS
Quality Plan	QAP
Test Documentation	TD
Use Case Model Survey	UCMS
Use Case Realization Report	UCRR
User Requirement Specifications	URS

Table 3 – WBS Legends (Deliverables)

4.2 WBS Details

No.	Activity	Resources	Deliverables
1.	**Project Planning**		
1.1.	Produce Project Plan	PM	PMP
1.2.	Review Project Plan	PM, QM	PMP (final)
1.3.	Produce Quality Plan	QM	QAP
1.4.	Review Quality Plans	PM, QM	QAP (final)
1.5.	Prepare first Audit & Presentation	QM, PM, BL	First PR
1.6.	Setup Filing System	QM	Filing system
2.	**Requirement Analysis**		
2.1.	Research Requirements	PM, BA,BAL	
2.2.	Produce User Requirements Specifications	BA, BAL	URS
2.3.	Review User Requirement Specifications	PM, QM, BAL	URS (final)
3.	**Analysis Modelling**		
3.1.	Identify Analysis Objects/Ops/Attributes	BAL, TL, DL	

3.2.	Produce Class/Collaboration Diagrams	BAL, TL, DL	
3.3.	Produce User Case Model Survey (Analysis)	BAL, TL, DL	UCMS (Analysis)
3.4.	Write Use Case Realization Report (Analysis)	BAL. TL, DL	UCRR (Analysis)
3.5.	Review & Finalize Use Case Realization Reports (Analysis)	QM, BA, BAL	UCRR (Analysis - final)
3.6.	Produce UI Specification	TL, DL	UI Specs
3.7.	Review UI Specifications (2 rounds)	TL, PM, UM	UI Specs (final)
3.8.	Produce Prototype UI	TL, DL	Prototype
3.9.	Review Prototype UI (2 rounds)	TL, PM, UM	Prototype (final)
3.10.	Produce Prototyping Study Report	TL, DL, PM	Prototyping Report
3.11.	Produce FS based on UCRR and Prototype UI	TL, DL	FS
3.12.	Review FS	TL, QM, PM	FS (final)
3.13.	Produce High-level DS	TL, DL	High-level DS
3.14.	Review High-level DS	TL, QM, PM	High-level DS (final)
3.15.	Prepare second Audit & Presentation	PM, QM, TL	Second PR
4.	**Design Modelling**		
4.1.	Adjust objects to implementation architecture	TL, DL	
4.2.	Define Object Associations	TL, DL	
4.3.	Produce Sequence Diagrams	TL, DL	Sequence Diagrams
4.4.	Specify Object Attributes/Operations	TL, DL	Object Specifications
4.5.	Structure Object Model for Implementation	TL, DL	
4.6.	Produce detailed DS	TL, DL	Detailed DS
4.7.	Review & finalize detailed DS	PM, QM, TL	Detailed DS (final)
5.	**Software Implementation**		
5.1.	Implement Code Components	TL, DL, DEV	System code
5.2.	Peer Review and Unit Test	TL, DL, DEV	Unit test results
5.3.	Integrate System Modules	TL, DL, DEV	Integrated System code
6.	**System Integration Testing**		
6.1.	Identify Integration Test Approach and	PM, TSL, TL	Test plan

	Test Plan		
6.2.	Review & finalize Integration and Test Plan	PM, TSL, TL	Test plan (final)
6.3.	Identify Test Cases based on Use Cases, prepare test script	TSL, TL	Test cases
6.4.	Implement Test Environment	TL, TSL	Test environment
6.5.	Prepare test data	BA, BAL	Test data
6.6.	Perform System Test	TSL, TL	Test results
6.7.	Additional Testing: stress test, performance test	TSL, TL	Test results
6.8.	Analyze Results and Correct Defects	PM, TSL , TL	Corrected system code, test results
7.	**User Acceptance Testing**		
7.1.	Prepare UAT environment	TL, DL, QM	UAT environment
7.2.	Prepare UAT data	BAL, BA	UAT data
7.3.	Conduct UAT	UM, TSL, PM, QM	UAT test results
7.4.	Review Test Results/ Corrective Action	UM, QM, TSL, TL, DL	System code (final) & Test Results (final)
8.	**Project Report and Closure**		
8.1.	Produce Final Project report	PM, QM	End of PR
8.2.	Produce User Guide	TL, DL	User Guide
8.3.	Review User Guide	UM, TL, PM	User Guide (final)
8.4.	Produce Project Presentation & Audit	PM, QM	Third PR

Table 4 – WBS Details

5. Project Effort Estimation

The VMS project will follow Waterfall SDLC model through 7 phases starting from Project Planning to Final Project Report.

In each phase, there will be a number of activities to be performed and required deliverables to be produced. Each phase will also involve different resources depends on the requirement in that phase – the details will be presented in Work Breakdown Structure (WBS) in section 4.

Project Manager and Quality Manager will be involved in all phases of the project for overseeing the activities and qualities of the deliverables.

Each phase in the VMS project will be discussed in details as below.

5.1 Resource Legends

Roles	Resources
Project Manager - PM	Dio
Quality Manager - QM	Peishan
Technical Lead - TL	Zaw
Development Lead - DL	Feng Yan
Business Analyst Lead - BAL	Thida
Business Analyst - BA	Hazel
Test Lead - TSL	Jifa
Developer - DEV	All members
Putative User Manager - UM	Dio, Peishan

Table 5 – Resource Legends

PROJECT EFFORT ESTIMATION (man-day)

#	Activity Description	Dio (PM)	Peishan (QM)	Thida (BAL)	Hazel (BA)	Zaw (TL)	Feng Yan (DL)	Jifa (TSL)
1	**Project Planning**							
1.1.	Produce Project Plan	4						
1.2.	Review Project Plan	1.5	1.5					
1.3.	Produce Quality Plan		4					
1.4.	Review Quality Plans	1	1					
1.5.	Prepare first Audit & Presentation	2	2	1	1	1	1	1
2	**Requirement Analysis**							
2.1.	Research Requirements	2.5		5.5	5.5			
2.2.	Produce User Requirements Specifications			3	2.5			
2.3.	Review User Requirement Specifications	1.5	1.5	1.5				
3	**Analysis Modeling**							
3.1.	Identify Analysis Objects/Ops/Attributes			5	5	5	5	
3.2.	Produce Class/Collaboration Diagrams			5	5	5	5	
3.3.	Produce User Case Model Survey (Analysis)			5	5	5	5	
3.4.	Write Use Case Realization Report (Analysis)			2.5	2.5	2.5	2.5	
3.5.	Review & Finalize Use Case Realization Reports (Analysis)	2	2	2		2		
3.6.	Produce UI Specification					3	3	
3.7.	Review UI Specifications (3 rounds)	2	2	2				
3.8.	Produce Prototype UI						5	5
3.9.	Review Prototype UI (3 rounds)	1.5	1.5					
3.10.	Produce Prototyping Study Report			1.5		1.5	1.5	
3.11.	Produce FS based on UCRR and Prototype UI					5	5	5
3.12.	Review FS	2.5	2.5					
3.13.	Produce High-level DS					2.5	2.5	

3.14.	Review High-level DS	2	1			0.5		
3.15.	Prepare second Audit & Presentation	2	2	1	1	1	1	1
4	**Design Modeling**							
4.1.	Adjust objects to implementation architecture					3	3	
4.2.	Define Object Associations			2	2			2
4.3.	Produce Sequence Diagrams	3	3	3	3			3
4.4.	Specify Object Attributes/Operations			4	4	4.5	4	4
4.5.	Structure Object Model for Implementation					1.5	1.5	
4.6.	Produce detailed DS			2.5	2.5	5	5	2.5
4.7.	Review & finalize detailed DS	1	1			1		
5	**Programming**							
5.1.	Implement Code Components	10	10	10	10	10	10	10
5.2.	Peer Review and Unit Test	5	5	5	5	5	5	5
5.3.	Integrate System Modules	2.5	5	5	2.5	5	5	5
6	**Integration System Test**							
6.1.	Identify Integration Test Approach and Test Plan	1				1		1
6.2.	Review & finalize Integration and Test Plan	1	1					1
6.3.	Identify Test Cases based on Use Cases, prepare test script			2.5	2.5			5
6.4.	Implement Test Environment					2	2	2
6.5.	Prepare test data	1	1	3	3			
6.6.	Perform System Test	2.5	2.5			1		2.5
6.7.	Additional Testing: stress test, performance test	1	1	1				1
6.8.	Analyze Results and Correct Defects	1	1		5	5	5	5
7	**User Acceptance Testing**							
7.1.	Prepare UAT environment					2	2	2
7.2.	Prepare UAT data			2.5	2.5			
7.3.	Conduct UAT	2.5	2.5			1		2.5
7.4.	Review Test Results/ Corrective Action	1	1		5	5	5	5

8	**Project Report and Closure**							
8.1.	Produce Final Project report	4	4					
8.2.	Produce User Guide					1	1	
8.3.	Review User Guide	1	1			1		
8.4.	Produce Project Presentation & Audit	2	2	1	1	1	1	1
	Individual total	64	62	76.5	75.5	89	86	71.5
	Project Total	**524.5**						

6. Project Milestones and Timeline

Major milestones have been identified, as shown in the below figure:

No.	Activity	Approx. start date	Approx. end date
1.	**Project Planning**		
1.1.	Produce Project Plan	21 Jan 2011	02 Apr 2011
1.2.	Produce Quality Plan	21 Jan 2011	02 Apr 2011
1.3.	Prepare First Audit & Presentation	28 Mar 2011	09 Apr 2011
2.	**Requirement Analysis**		
2.1.	Produce User Requirements Specifications	22 Jan 2011	02 Apr 2011
3.	**Analysis Modelling**		
3.1.	Produce Functional Specifications : UCMS and UCRR (Analysis)	10 Apr 2011	07 Jul 2011
3.2.	Produce High-level Design Specifications	1 Jul 2011	1 Aug 2011
3.3.	Produce Prototyping Study Report	10 Apr 2011	12 Jun 2011
3.4.	Prepare Second Audit & Presentation	01 Aug 2011	12 Aug 2011
4.	**Design Modelling**		
4.1.	Produce detailed DS	02 Aug 2011	13 Sep 2011
5.	**Software Implementation**		
5.1.	Produce System Code	14 Sep 2011	26 Nov 2011
5.2.	Integrate System Code	14 Sep 2011	26 Nov 2011
6.	**System Integration Testing**		
6.1.	Produce SIT plan, test script and test data	13 Aug 2011	13 Sep 2011
6.2.	Perform System Integration Test and corrective actions	27 Nov 2011	11 Dec 2011
7.	**User Acceptance Testing**		
7.1.	Prepare UAT plan, test script and test data	13 Aug 2011	13 Sep 2011
7.2.	Perform UAT	12 Dec 2011	30 Dec 2011
8.	**Project Report and Closure**		
8.1.	Produce Final Project report	20 Dec 2011	07 Jan 2012
8.2.	Produce User Guide	13 Aug 2011	07 Jan 2012
8.3.	Produce Project Presentation & Audit	20 Dec 2011	07 Jan 2012

Table 6 – Project Milestones and Timeline

7. Project Deliverables

Major milestones have been identified, as shown in the below figure:

By undertaking the work described in this plan, the following deliverables will be produced by the project:

- Project Plan.
- Quality Plan.
- User Requirement Specifications.
- First Presentation and Audit
- Functional Specification: Use Case Model Survey and User Case Realization Report (Analysis) – one per use case
- High-level Design Specifications: Proposed Software Architecture and Transition Strategy from Analysis to Design
- Prototyping Study Report
- Second Presentation and Audit
- Detailed Design Specifications
- Source and Executable Code.
- Test Plan and other test documents
- User Guide
- End of Project Report
- Final Project Presentation

8. Resources

The supporting resources required to enable the project team to undertake the activities specified in Section 4 (WBS) are described as below:

8.1 Computer Hardware and Software

The team will use these following hardware and software during the whole project timeline:

- Laptop or desktop with Windows environment
- Eclipse IDE
- Java Programming Language

- Hibernate Framework
- Spring Framework
- Microsoft SQL Server
- Microsoft Office
- And other components required for the VMS development.

ANNEX A: Effort Estimation by FPC & COCOMO

COCOMO Detail report

Estimate1 - Detail Report

Costar 7.02 04/09/2011 13:02:56 Page: 1

Estimate Name:	Estimate1	Estimate ID:	
Model Name:	Early Design 2000	Model ID:	2000
Process Model:	COCOMO II Model	Phases:	Waterfall

Component Name:	Component1	Component ID:	
Increment:	1	Level:	1
Developed Size:	5,359	EAF:	1.0067

Phase	Effort (Person-Months)	Cost (K$)	Duration (Months)	Staffing
RQ -- Requirements	1.3	0.0	1.6	0.8
PD -- Product Design	3.2	0.0	2.3	1.4
DD -- Detailed Design	5.0	0.0	2.1	2.3
CT -- Code & Unit Test	6.8	0.0	2.9	2.3
IT -- Integration & Test	3.9	0.0	2.0	1.9
Development (PD+DD+CT+IT)	18.9	0.0	9.4	
Totals (RQ+PD+DD+CT+IT)	20.3	0.0	11.0	
MN -- Maintenance (per year)	0.0	0.0		0.0

COCOMO Effort Report

Estimate1 - Effort Report

Costar 7.02 04/09/2011 13:03:15 Page: 1

Estimate Name:	Estimate1	Estimate ID:	
Model Name:	Early Design 2000	Model ID:	2000
Process Model:	COCOMO II Model	Phases:	Waterfall

Effort per Component (Person-Months)

Component Name	RQ	PD	DD	CT	IT	Total RQ to IT
Component1	1.3	3.2	5.0	6.8	3.9	20.3

Effort Summary

Component Totals	1.3	3.2	5.0	6.8	3.9	20.3
Grand Total	1.3	3.2	5.0	6.8	3.9	20.3

COCOMO Cost Driver Report

Estimate1 - Cost Driver Report

Costar 7.02 04/09/2011 13:03:30 Page: 1

Estimate Name: Estimate1 Estimate ID:
Model Name: Early Design 2000 Model ID: 2000
Process Model: COCOMO II Model Phases: Waterfall

Component Name	EAF	RUSE	SCED	RCPX	PDIF	PERS	PREX	FCIL	
Component1	1.0067	N	N	H	N	N	H	H	

Data functions

Internal Logical File/External Interface File				Element of complexity		
#	**Type**	**Reference**	**Description**	**DET**	**RET**	**Complexity**
1.	ILF	URS 3.1.2.2	**Volunteer Account:** - Title(Mr. , Mrs. , Ms, Dr, Other) - First Name - Last Name - Address - Country - Email Address (Read Only Field) - Phone(Home) - Phone(Mobile) - Date of Birth - Hobbies/ Interests - Special Skills - Qualifications attained (Education) - Qualifications attained (Others) - Languages - Volunteer Interests (Admin, Befriending, Events (organizing), General Services, IT, Research, Marketing/Publicity, Skilled Services, Others) - Join Date(Read Only)	16	1	L
2.	ILF	URS 3.1.4.1	**Project**: - Name - Location - Project introduction - Start date	7	1	L

			- End date - Status (Finished/in progress/ Coming)			
3.	ILF	URS 3.1.2.8	**Feedback**: - Author - Content - Created Time - Status (approved/not approved)	4	1	L
4.	ILF	URS 3.1.2.9	**Story/Experience** - Author - Topic - Content - Created Date - Status (approved/not approved)	5	1	L
5.	ILF	URS 3.1.4.5	**Project Preference (of volunteer):** - Volunteer Name - Preference details.	2	1	L
6.	ILF	URS 3.1.2.11	**Request for certificate:** - Requestor (Volunteer name) - Project name - Authorizer (the authority to approved this request) - Created Date.	4	1	L
7.	ILF	URS 3.1.3.5	**Staff account:** - Title(Mr. , Mrs. , Ms, Dr, Other) - First Name - Last Name - Address - Country - Email Address (Read Only Field) - Phone(Home) - Phone(Mobile) - Date of Birth - Job Title - Role(Read Only Field) - Join Date	12	1	L
8.	ILF	URS 3.1.4.1	**Project Proposal:** - Author / Proposer - The approving authority - Proposal details - Created date - Status of proposal.	5	1	L
9.	ILF	URS 3.1. 5.1	**Itinerary:** - Project ID - Start date - End date - Itinerary details (when do what, by whom) - Status (approved/not approved).	5	1	L

10.	ILF	URS 3.1.4	**Project Progress:** - Expected Start date - Expected End date - Actual Start date - Actual End date - Status.	5	1	L
11.	ILF	URS 3.1.4.7	**Project Calendar:** - Project name - Calendar details.	2	1	L
12.	ILF	URS 3.1.7.3	**Certificate:** - Certificate name - Volunteer name - Project name - Certificate content.	4	1	L
13.	ILF	URS 3.1.8.1	**System parameter:** - Parameter name - Parameter value	2	1	L

Transactional Functions

External Input (EI)/ External Output (EO)/External Enquiry (EQ)				**Element of complexity**		
#	**Type**	**Reference**	**Description**	**DET**	**FTR**	**Complexity**
1.	EI	URS 3.1.2.1	**Register Volunteer Account:** - User name - Password - DOB - Email - Phone	5	1	L
2.	EI	URS 3.1.2.2	**Update Volunteer Profile:**	5	1	L
3.	EI	URS 3.1.2.3	**Retrieve Forgotten Password**	2	1	L
4.	EI	URS 3.1.2.4	**Change Password**	2	1	L
5.	EI	URS 3.1.2.8	**Post Feedback to Project**: - Created By - For which project - Content - Created Time - Status (approved/not approved).	5	1	L
6.	EI	URS 3.1.2.9	**Share Experience and Inspiring stories:** - Author - Topic - Content - Created Date - Status (approved/not approved)	5	1	L
7.	EI	URS 3.1.2.6	**Raise Interest to Project** - Volunteer Name - Preference details	2	1	L

8.	EI	URS 3.1.2.11	**Send Request for Certificate** - Requestor (Volunteer name) - Indicated project - Authorizer (the authority to approved this request) - Created Date	4	1	L
9.	EI	URS 3.1.2	**Delete/Deactivate Volunteer Account**	5	1	L
10.	EI	URS 3.1.3.1	**Register Staff Account By Admin** - Staff name - Password - Department - Position / Role (Project director, PM, lead…etc.) - DOB - Mobile phone - Email	7	1	L
11.	EI	URS 3.1.3.2	**Update Profile By Staff**	7	1	L
12.	EI	URS 3.1.2.3	**Retrieve Forgotten Password**	2	1	L
13.	EI	URS 3.1.2.4	**Change password**	2	1	L
14.	EI	URS 3.1.3	**Manage Staff Role**	2	1	L
15.	EI	URS 3.1.3	**Delete/ Deactivate Staff Account**	7	1	L
16.	EI	URS 3.1.4.1	**Propose Project** - Author / Proposer - The approving authority - Proposal details - Created date - Status of proposal	5	1	L
17.	EI	URS 3.1.4.2	**Approve/Reject project proposal**	5	1	L
18.	EI	URS 3.1.4.3	**Setup project details** - Name - Location - Start date - End date - Status	5	1	L
19.	EI	URS 3.1.4.5	**Invite Volunteer By Email**	10	2	A
20.	EI	URS 3.1.4.6	**Assign project member and roles**	6	2	A
21.	EI	URS 3.1.5.4	**Setup itinerary plan**	5	1	L
22.	EI	URS 3.1.4.7	**Setup project calendar** - Project name - Calendar details	2	1	L
23.	EI	URS 3.1.4	**Update project progress** - Expected Start date - Expected End date - Actual Start date - Actual End date - Status	5	1	L
24.	EI	URS 3.1.5.1	**Prepare Itinerary Plan** - Project ID	5	1	L

			- Start date - End date - Itinerary details (when do what, by whom) - Status (approved/not approved)			
25.	EI	URS 3.1.5.2	**Submit Itinerary Plan**	5	1	L
26.	EI	URS 3.1.5.3	**Approve/Reject Itinerary Plan**	5	1	L
27.	EI	URS 3.1.7	**Upload Certificate Template** - Certificate name - Volunteer name - Certificate content - Project name	4	1	L
28.	EI	URS 3.1.7.1	**Request certificate by volunteer**	4	1	L
29.	EI	URS 3.1.8.1	**Configure System Parameter** - Parameter name - Parameter value	2	1	L

External Enquiry

External Input (EI)/ External Output (EO)/External Enquiry (EQ)				**Element of complexity**		
#	**Type**	**Reference**	**Description**	**DET**	**FTR**	**Complexity**
1.	EQ	URS 3.1.1.1	**Logon** **Input:** - ID - Password - Output: - Security level	 2 1	 1 1	L
2.	EQ	URS 3.1.2.5	**Browse for project** **Input:** - Start date - Location **Output:** - Name - Location - Start date - End date - Status	 2 5	 1 1	L
3.	EQ	URS 3.1.2	**Display Volunteer Account** **Input:** - Volunteer name **Output:** - User name - DOB - Role (Team lead/member) - Address - Email - Phone	 1 7	 1 1	L
4.	EQ	URS 3.1.3	**Display Staff Account** **Input:**	 1	 1	L

			- Staff name - Output: - Staff name - Department - Position / Role (Project director, PM, lead…etc.) - DOB - Mobile phone - Email	7	1	
5.	EQ	URS 3.1.6.2	**Project Tracking Report** **Input:** - Project ID **Output:** - Project calendar - Project itinerary - Project progress	1 3	1 1	L
6.	EQ	URS 3.1.6.1	**Volunteer Specific Report** **Input:** - Volunteer name **Output:** - No of Projects participated in - No of days spend in each project - Position in each project - Project preference	1 5	1 1	L

External Output

External Input (EI)/ External Output (EO)/External Enquiry (EQ)				**Element of complexity**		
#	**Type**	**Reference**	**Description**	**DET**	**FTR**	**Complexity**
1.	EO	URS 3.1.2.3	**New password email** - User name - New password - Hyperlink to activate new password	3	1	L
2.	EO	URS 3.1.4.5	**Volunteer Email invitation** - Project name - Project details - Hyperlink to join project	3	1	L
3.	EO	URS 3.1.7.1	**Generated Certificate** - Certificate name - Volunteer name - Project name - Certificate content	4	1	L

Unadjusted Function Point Count Summary

ITEM	LOW	AVERAGE	HIGH	TOTAL
External Input	27x3=81	2x4=8	0x6=0	89
External Output	3x4=12	0x5=0	0x7=0	12
Internal File	13x7=91	0x10=0	0x15=0	91
Interface File	0x5=0	0x7=0	0x10=0	0
External Inquiry	6x3=18	0x4=0	0x6=0	18
			Unadjusted Function Points	**210**

Calculate Value Adjustment Factor

CHARACTERISTIC	DI	Reasons
Data Communications	**2.5**	VMS is an online system with moderate number of users
Distributed Functions	**4**	The system will be built with multi-tier architecture where system components will be on different boxes
Performance	**1**	There is no special requirements for performance
Heavily Used Configuration	**0**	Configuration not mentioned in the URS
Transaction Rate	**1**	There are only moderate number of end users
On-Line Data Entry	**4**	System is an online system which deal moderately with data input
End User efficiency	**2**	There are no special requirements of end user efficiency except for some browsers support
On-line Update	**2**	System is an online system which deal moderately with data update
Complex Processing	**1**	There is no complicated processing of data in VMS
Re-usability	**2**	The code should be moderately reusable
Installation Ease	**0**	It is not mentioned in the URS
Operational Ease	**0**	It is not mentioned in the URS
Multiple Sites	**2.5**	It is not mentioned in the URS
Facilitate Change	**1**	There is just few requirements for further growth
Total Degree of Influence	**23**	

ADJUSTED FUNCTION POINT COUNT SUMMARY

Adjusted Factor = 0.65 + 0.01 * 23	0.88
Adjusted FPC = Unadjusted FPC * Adjusted Factor	**185**

CHOSEN MODEL AND COCOMO II SCALE FACTOR

We shall use the EARLY DESIGN 2000 WATERFALL model when using the COSTART 7 software tool

Scaling Factors	Description	Rating
Precedentedness	The team have never developed a volunteer management system	Somewhat Unprecedented
Development Flexibility	Strictly follow project plan and ISS audit process	Rigorous
Architecture/Risk Resolution	It seems to be a straightforward system	Often (60%)
Team Cohesion	Having worked together for one year	Basically co-operative
Process Maturity	RUP Process	SEI CMM Level2

COSTAR COST DRIVER

Cost Drivers	Product Attributes	Description
RCPX	Required Software Reliability: • High Product Complexity: • Nominal Documentation: • Nominal Database Size: • Assume that each user record contains 500 bytes • Assume that each project record contains 1M bytes • Assume total no of new users is 1000 per year • Assume total no of new projects is 10 per year • So total no of new records per year is roughly 0.5M • The existing user record over the past years can be 30M • Total databse size = 40M • 40M/5000 (High)	High
RUSE	Required Re-usability: • There is no intention to reuse the code in the future	Nominal
PDIF	Platform Difficulty: • No obvious constraints or stability problem	Nominal
PERS	Personnel Capability • Nothing said	Nominal
PREX	Combined Application, Platform, and Tool Experience: • The team is very familiar with Java framework	High
FCIL	Level of Tool and Site support: • The system will be	High

	developed using Java framework, GUI tools	
SCED	Required Development Schedule • The development shall follow the project plan strictly	Nominal

COSTART 7 EFFORT AND SCHEDULE ESTIMATES

Programming language	Java
Translation factor	29
Lines of Code = *Translation factor * FPC* *= 29 * 185 * (1 + Breakage)* *= 29 * 185* *Note:* *(i) % Breakage = 0 (It is assumed that requirement will not change)* *(ii) The system will not reuse any existing software*	5365 SLOC
Development Effort =	20.3 man-months
Schedule =	11.0 months

ANNEX B: Risk Questionnaire

Risk Questionnaires and Risk Management Techniques

S/N	Risk Item	Control Type	Specific Details
1.	No Project management skills set	Risk Localization Risk Insurance	Gain PM skills by on the job training Shadow a few people to work with common tasks and knowledge
2.	Team member may not be available during the project timeline	Risk Insurance Risk Avoidance	Each key project role will have backup personnel Schedule carefully to anticipate all unavailability upfront
3.	Customer may lose commitment and availability	Risk Avoidance Risk Transfer Risk Insurance	Look for alternative customer Appoint team member as putative customer Strengthen and ensure customer commitment
4.	Not domain expert and not much knowledge in Volunteer Management	Risk Transfer Risk Localization	Assign team members to gain domain knowledge by attending similar voluntary activities Research and study similar Volunteer Management systems
5.	Not all team member have technical knowledge for: - Presentation Layer (YUI2, DWR) - Data Access Layer (Hibernate)	Risk Localization Risk Insurance	Technical Architect to do training for team members to be familiar with the technology Do early prototype to study the technology upfront
6.	Not sure of the User Interface requirements	Risk Localization Risk Minimization	Produce early prototype to gather UI requirements Request customer to define thorough UI requirements Update UI specs as and when requested
7.	Portability risk : system need to support multiple browsers, multiple platforms	Risk Localization Risk Minimization	Do prototype to test the compatibility/portability against browsers Perform research for each key components to ensure maximum compatibility Aim to at least support IE7+ (other browsers for later)

Risk Impact Analysis

No.	Risk Item	Impact						Likelihood					
		1	2	3	4	5	Total	1	2	3	4	5	Total
1.	No Project management skill set		x	x			2	x		x	x		3
2.	Team member may not be available during the project timeline	x	x	x	x		4			x	x	x	3
3.	Customer may lose commitment and availability	x	x	x			3			x	x		3
4.	Not domain expert and not much knowledge in Volunteer Management			x	x		2			x	x		2
5.	Not all team member have technical knowledge		x	x	x	x	4			x	x	x	3
6.	Not sure of the User Interface requirements			x		x	2				x		1
7.	Portability risk : system need to support multiple browsers, multiple platforms		x	x	x		3			x	x	x	3

Impact Legend

1: Reduced Functionality
2: Increased Cost
3: Schedule Slippage
4: Quality Cost
5: Performance

Likelihood Legend

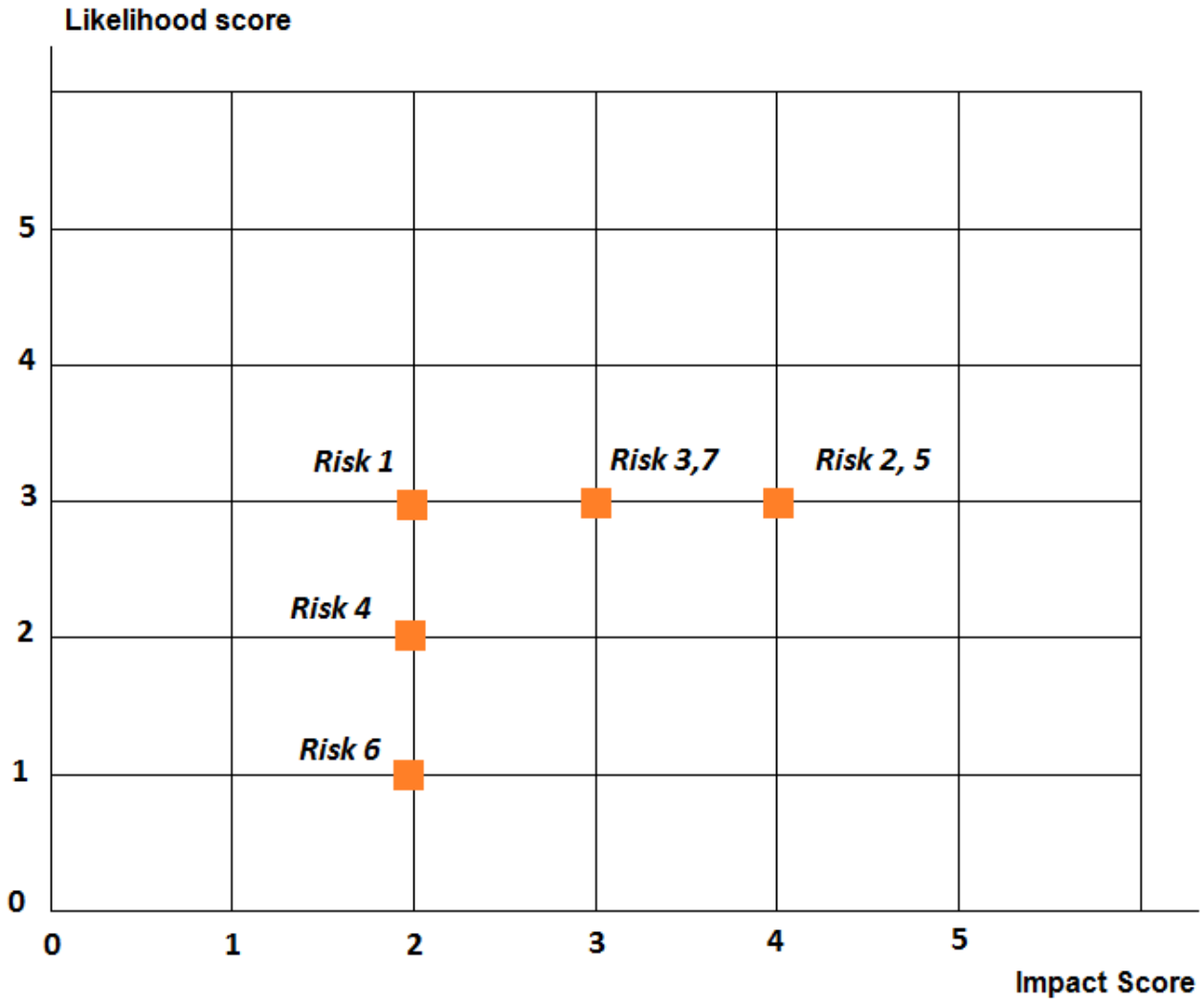
1: Novelty
2: Product History
3: Project Team History
4: Recognised Authority
5: Expectation

ANNEX C: List of Prioritized Tasks and Risk Prioritization Chart

Prioritized Tasks

#	Risk	Impact score	Likelihood score	Total score
2	Team member may not be available during the project timeline	4	3	7
5	Not all team member have technical knowledge	4	3	7
3	Customer may lose commitment and availability	3	3	6
7	Portability risk : system need to support multiple browsers, multiple platforms	3	3	6
1	No Project management skill set	2	3	5
4	No domain expert and not much knowledge in Volunteer Management	2	2	4
6	Not sure of the User Interface requirements	2	1	3

Risk Prioritization Chart



ANNEX D: Amendment History

This is a record of all amendments made to the Document after sign-off

Version	Chapter / Section	Page No	Notes of Amendments or Changes
			•
			•
			•
			•
			•
			•
			•
			•
			•
			•
			•
			•
			•