SOFTWARE DEVELOPMENT PLAN

VERSION: PRELIMINARY

17 MARCH, 2006

NOOR HAFIZA BINTI MOHD RAWI

DATABASE DESIGN AND ANALYSIS FOR ENVIRONMENT DATA

CONTENTS

1. INTRODUCTION 2
1.1 PROJECT OVERVIEW 2
Objectives 2
1.2 PROJECT DELIVERABLES 3
1.3 EVOLUTION OF THE SOFTWARE PROJECT MANAGEMENT PLAN 3
1.4 REFERENCE MATERIALS 4
1.5 DEFINITIONS AND ACRONYMS 4

2. PROJECT ORGANIZATION 5
2.1 PROCESS MODEL 5
2.2 ORGANIZATIONAL STRUCTURE 5
2.3 ORGANIZATIONAL BOUNDARIES AND INTERFACES 5
2.4 PROJECT RESPONSIBILITIES 5

3. TECHNICAL PROCESS 6
3.1 METHODS, TOOLS, AND TECHNIQUES 6
3.1.1 Methodology 6
3.1.2 Method 8
3.1.3 Tools 8
3.1.4 Techniques 8
3.2 SOFTWARE DOCUMENTATION 9
3.3 PROJECT SUPPORT FUNCTIONS 9
3.3.1 Quality Assurance 9
3.3.2 Software Quality 9

4. WORK PACKAGES, SCHEDULE, AND BUDGET 10
4.1 WORK PACKAGES 10
4.2 DEPENDENCIES 12
4.3 RESOURCE REQUIREMENTS 12
4.3.1 Hardware Requirements 12
4.3.2 Software Requirement 12
4.3.3 Budget and Resource Allocation 12
4.4 SCHEDULE 12

5. APPENDICES 13
5.1 APPENDIX I 13
5.2 APPENDIX II 14
5.3 APPENDIX III – GANTT CART 15

1. INTRODUCTION

This document is to provide an overview of the entire document:

1. Describe the purpose, scope and methodology of this project.
2. To explain the main tasks undertaken by the developer.
3. To introduce software project management and to describe its distinctive characteristics.
4. To discuss project planning and the planning process.
5. To show how graphical schedule representations are used by project management.
6. Summarize (or give an abstract for) the contents of this document.

1.1 Project Overview

Environmental data such as temperature, pressure, humidity and radiation in Malaysia had been collected over several years. The data were obtained using various equipments in the field. Some parameters also have incomplete data. It is desirable to enable the data to be accessed globally and also to provide summaries of the data with trends and projections.

1.2 Objectives

- To design a database that can store the environment readings/data on a web server.
- To perform a specific query from the database that has been created in the previous step
  - On the client-side in a web-based platform.
- To generate a specific report periodically on a month basis or yearly basis as requested by the client.
- To analyze the data for diurnal (daily), seasonal and other trend/projections purposes
- Also look at method of filling up the missing data if necessary.

1.3 Project Deliverables

No.	Deliverables	Verified By	Date
1.	Software Development Plan	Supervisor	25/03/2006
2.	Use Case Diagram	Supervisor	01/04/2006
3.	Software Requirement Document (SRD) Version: Draft Version: Preliminary Version: Final	Supervisor	 01/04/2006 05/06/2006 23/09/2006
4.	Software Design Document (SDD) Version: Draft Version: Preliminary	Supervisor	 05/06/2006 23/09/2006
5.	Software Test Document		

1.4 Evolution of the Software Project Management Plan

This section will list what is expected to be done to this document. This chart contains a plan for the document's revisions. The entries below are provided for the purposes of illustration. The progression from "Draft" to "Preliminary" to "Final" and then through a section of "Revisions" is one typical approach.

Version	Primary Author(s)	Description of Version	Date Expected
Draft	Developer	Initial draft created for distribution and review comments	11/03/2006
Preliminary	Developer	Second draft incorporating initial review comments, distributed for final review	25/03/2006
Final	Developer	First complete draft, which is placed under change control	TBD
Revision 1	Developer	Revised draft, revised according to the change control process and maintained under change control	TBD
etc.	Etc	Etc.	TBD

For this project, project deliverables includes all the project's development life cycle.

1.5 Reference Materials

(Ignore)

1.6 Definitions and Acronyms

(Ignore)

2. PROJECT ORGANIZATION

In the next four subsections, developer will describe how the product will be developed. This development includes the view of software process and the structure of the project organization.

2.1 Process Model

For this project, process model in used is Rational Unified Process (RUP). This model is used because it is one of the modern process model derived from the work on the UML and associated process. RUP have four phases, which are Inception, Elaboration, Construction and Transition.

Also, for the project's major milestones, I prepare a draft for Gantt Cart. This cart describes a briefing for the project's schedule. See Gantt Cart in the Appendix II.

2.2 Organizational Structure

All the project development process will be develop by the developer, so there is no organizational structure.

2.3 Organizational Boundaries and Interfaces

(Ignore)

2.4 Project Responsibilities

All the project development is under developer responsibilities.

3. TECHNICAL PROCESS

This section describes the top-level technical processes used on the project including the technical methods, tools, and techniques; major software documents; and supporting activities such as configuration management and quality assurance.

3.1 Methods, Tools, and Techniques

3.1.1 Methodology

For this project, methodology in used is Rational Unified Process (RUP). All the phases in RUP are described as below:

i. **Inception**: Establish the business case for the system.
ii. **Elaboration**: Develop an understanding of the problem domain and the system architecture.
iii. **Construction**: System design, programming and testing.
iv. **Transition**: Deploy the system in its operating environment.

However, for this version, only Inception and Elaboration phase will be discuss. The next two phases will be discussed in the next version. The RUP diagram with the workflow is shown in Appendix I.

i. **Inception Phase**

In this phase, the business case which is includes project requirement is establish. To complement this business case, a basic use case model, project plan (includes the Gantt Cart) and project descriptions are generated. Activities in this phase are list as below:

Activities	Description
Analysis	- Analyze requirement
Architecture	- Identify the methods, tools and techniques
Implementation	- Identify the implementation issues
Installation and support	- Identify the installation's issues such as cost, technique and risk

If this project does not pass this milestone, this phase will be repeat until it is being redesigned to better meet the criteria.

ii. **Elaboration Phase**

The elaboration phase is where the project starts to take shape. In this phase the problem domain analysis is made and the architecture of the project gets its basic form. Activities for this phase are shown as below:

Activities	Description
Analysis	- Analyze the main requirement - Prepare a Sequence Diagram
Architecture	None
Implementation	None
Installation and support	- Install software and hardware

This phase must pass the milestone by meeting the following criteria:

- Use-case model in which the use-cases and the actors have been identified and most of the use-case descriptions are developed. The use-case model should be 80% complete.
- A description of the software architecture in a software system development process.
- Architecture prototype, which can be executed.
- Business case and risk list which are revised.
- A development plan for the overall project.

In this phase, system are designed depends on the software specification document. Then, system will be develop depends on the software document. After leaving this phase, the project transitions into a high-risk operation where changes are much more difficult and detrimental when made.

iii. **Construction Phase**

In this phase the main focus goes to the development of components and other features of the system being developed. This is the phase when the bulk of the coding takes place. This phase produces the first external release of the software.

iv. **Transition Phase**

In the transition phase the product has moved from the development organization to the end user. The activities of this phase include training of the end users and maintainers and beta testing of the system to validate it against the end users' expectations. The product is also checked against the quality level set in the Inception phase. If it does not meet this level, or the standards of the end users, the entire cycle in this phase begins again.

If all objectives are met, the Product Release (PR) milestone is reached and the development cycle ends.

3.1.2 Method

The RUP will be implementing using the Unified Modeling Language (UML) method.

3.1.3 Tools

All the tools are lists as below:

- **Software**
  i. Rational Rose: to prepare the UML techniques
  ii. Macromedia Dreamweaver MX: to develop the software
  iii. PHP: as the programming language
  iv. mySQL: as the database
  v. Windows XP: as the operating system
  vi. Internet Information Services (IIS): web server

- **Hardware**
  (Ignore)

3.1.4 Techniques

All techniques in UML will be implemented for this project. The techniques are lists as below:

- Analysis Modeling – use case diagram, sequence diagram, activity diagram
- Design Modeling – package

3.2 Software Documentation

(Ignore)

3.3 Project Support Functions

3.3.1 Quality Assurance

To ensure the system development process is quality; a few things need to be concern:

i. The documentation needs to follow the IEEE standard.
ii. Always check the system to make sure the system follows the standard.
iii. Prepare report for test case design.

3.3.2 Software Quality

i. Reliability - the system should be reliable to any cases of failure.
ii. Maintainability – the system should be design based on identified components.

4. WORK PACKAGES, SCHEDULE, AND BUDGET

In the next five subsections, developer will describe the work packages, dependencies, resource requirements and budget.

4.1 Work Packages

This subsection describes the work package (i.e., tasks) that must be completed to complete the software. Each work package was present in a diagram showing the breakdown of work packages.

Work package 1: Establish the business case for the system

ID	WBS	Tasks
1	1	**Project's Proposal**
2	1.1	Identify problem
3	1.2	Determine the objective, scope and constrains
4	1.3	Send proposal
5	2	**Selection of Methods, Tools and SDLC**
6	2.1	Literature of methods, tools and SDLC
7	2.2	Discussion with supervisor
8	3	**Inception Phase**
9	3.1	**Preliminary Iterations**
10	3.1.1	Iterations strategy
11	3.1.2	Identify requirement, scope and constrains
12	3.1.3	Preparation of Software Project Development Plan
13	3.1.4	Software Project Development Plan document
14	3.2	**Preliminary Requirements Analysis**
15	3.2.1	Feasibility Study
16	3.2.2	Analysis of risk and domain problem
17	3.2.3	Identify Functional requirement
18	3.2.4	Identify non-Functional requirement
19	3.3	**Design**
20	3.3.1	Use case design
21	3.4	**Test**
22	3.4.1	Functional and non-Functional testing
23	3.4.2	Use case testing
24	3.4.3	Use Case Diagram
25	4	**Elaboration Phase**
26	4.1	**Preliminary Iterations**
27	4.1.1	Analysis of domain requirement
28	4.1.2	Design
29	4.1.3	Implementation
30	4.1.4	Definition and requirement specification
31	4.1.5	Requirement definition
32	4.1.6	Requirement specification
33	4.1.7	Preparation of Software Requirement Specifications document
34	4.2	Software Requirement Specifications Document (SRS)

35	4.3	**Secondary Iterations**
36	4.3.1	Analysis of requirement
37	4.3.2	Design
38	4.3.3	Implementation
39	4.3.4	Test
40	4.3.5	Preparation of Sequence Diagram
41	4.3.6	Sequence Diagram
42	5	Preparation of project's documentation

Work package 2: Development of the system / Software Design

ID	WBS	Tasks
43	6	**Construction Phase**
44	6.1	**Preliminary Iterations**
45	6.1.1	Construction phase planning
46	6.1.2	Basis design
47	6.2	**Secondary Iterations**
48	6.2.1	**Design specification**
49	6.2.1.1	Workflow design
50	6.2.1.2	Architectural design
51	6.2.1.3	Abstract specification
52	6.2.1.4	Interface design
53	6.2.1.5	Component design
54	6.2.1.6	Data structure design
55	6.2.1.7	Algorithm design
56	6.2.1.8	Database design
	6.2.1.9	System development
57	6.2.2	**Software Design Document (SDD)**
58	6.3	**Third Iterations**
59	6.3.1	**Server-side development**
60	6.3.1.1	Database development
61	6.3.1.2	Data Access Layer development
63	6.3.2	**Client-side development**
64	6.3.2.1	Workflow development
65	6.3.2.2	Web Application development
	6.3.3	System development
66	7	**Transition Phase**
67	7.1	**Preliminary Iterations**
68	7.1.1	Basis Testing Plan
69	7.1.2	System Test
70	7.1.2.1	White-box Testing
71	7.1.2.2	Black-box Testing
72	7.2	**Software Test Document (STD)**
73	7.3	**Secondary Iterations**
74	7.3.1	Installation Plan
75	7.3.2	System implementation
76	7.3.3	Acceptance Testing

4.2 Dependencies

(Ignore)

4.3 Resource Requirements

4.3.1 Hardware Requirements

(Ignore)

4.3.2 Software Requirement

See section 3.1.3.

4.3.3 Budget and Resource Allocation

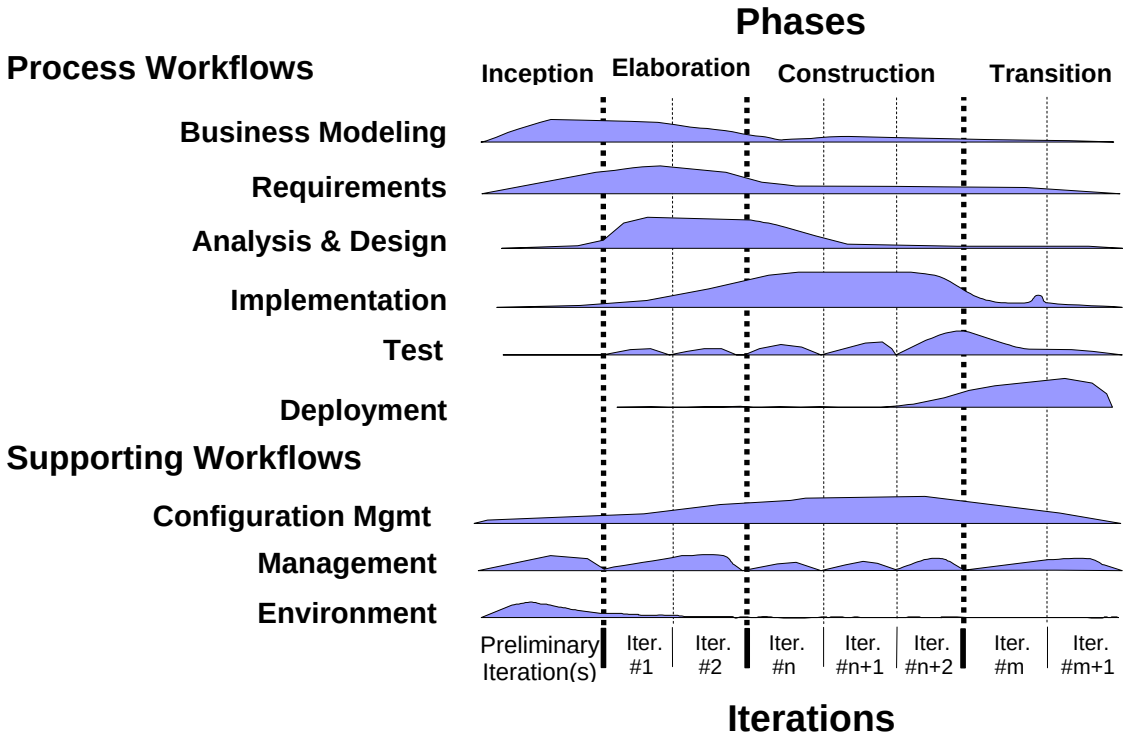
(Ignore)

4.4 Schedule

See Appendix III – Gantt Cart

5. APPENDICES

5.1 Appendix I



5.2 Appendix II

The static workflow for the RUP are descript in the table below:

Workflow	**Description**
Business modeling	The business processes are modelled using business use cases.
Requirements	Actors who interact with the system are identified and use cases are developed to model the system requirements.
Analysis and design	A design model is created and documented using architectural models, component models, object models and sequence models.
Implementation	The components in the system are implemented and structured into implementation sub-systems. Automatic code generation from design models helps accelerate this process.
Test	Testing is an iterative process that is carried out in conjunction with implementation. System testing follows the completion of the implementation.
Deployment	A product release is created, distributed to users and installed in their workplace.
Configuration and change management	This supporting workflow managed changes to the system.
Project management	This supporting workflow manages the system development.
Environment	This workflow is concerned with making appropriate software tools available to the software development team.

5.3 Appendix III – Gantt Cart