CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION

Methodology refers to the analysis of the methods used appropriate to the field of study. It is a systematic way of accomplishing certain tasks and is defined as a collection of procedures, techniques, tools and documentation aids that helps a software developer to speed up and simplify the software development process (Pressman 2001).

Methodology acts as the backbone that provides guidance on how a system should be developed and the proper activities to be done. It ensures that a consistent and reproducible approach is used from the first activity of the software development process until the software is completed. It also helps in reducing the risk of omitting important activities and consistent documentation is produced, and helps the developers to speed up and simplify the development process.

This chapter contains an explanation on software development methodology that was used in T-UCD to plan, manage, control and evaluate the project. It follows by the next section
which examines the information gathering techniques that have been used for the purpose of data collection.

3.2 SOFTWARE DEVELOPMENT LIFE CYCLE

Software Development Life Cycle (SDLC) or sometimes known as the waterfall model is a phased and structured approach for systems development in many organizations (Deacon 2005). The phases include requirements feasibility, analysis, system design, coding, implementation, testing and maintenance. There are variations of these stated phases, with anywhere from three to almost twenty identifiable phases (Dennis, et al. 2001). Usually, each phase is performed sequentially, although potential for overlaps exists. The specific step and their sequence are meant to be adapted as required for a project, consistent with management approaches. At the end of each phase (and sometimes within phases for intermediate steps), a system development project reaches a milestone and as deliverables are produced, they are often reviewed by parties outside the project team.

3.2.1 SDLC of T-UCD: V-Shaped Model

In order to develop a T-UCD, the V-Shaped model process is used as the SDLC project development. However, it is simply ineffective in practice and thus unrealistic to assume it will work for this project. The choice of using the V-Shaped Model will makes sense only if the project is well defined and has fewer risks. The requirements must be well known and stable.
The V-Shaped model focused on the importance of testing during all stages of T-UCD development. Each phase in V-Shaped model has to be evaluated by the client with review and verified by specific deliverable before proceeding to the next phase during T-UCD development. Moreover, the flow of control in this model is iterative, which allows the T-UCD project software developer to return to the previous stage if required. Furthermore, this model has a number of advantages such as earlier receipt user feedback, earlier review and evaluation of requirements which helps in developing additional requirements and upfront monitoring of quality. This is one of the few available ways to generate a quality software product.

The software development life cycle phases with testing activity in the V-Shaped Model (refer to Figure 3.1) will be done in several phases for which each phase represents a complete development life cycle, with certain functionality of the system delivered at the end of each phase. The approach of using phases for project deliveries provides flexibility in what the developer will deliver; this allows both the developer and the user to change contents on any of the phases.
As for the project development, V-Shaped model has been referred as a guide to ensure the development of this system follows proper steps. Figure 3.2 is the modified V-Shaped model according to the development of T-UCD.
Figure 3.2: T-UCD V-Shaped model
The software development phase of T-UCD development can be broken down to the following tasks as follows:

i. Initial phase
ii. Requirement analysis phase
iii. System design phase
iv. Implementation phase
v. Testing phase
vi. Evaluation and acceptance phase

3.2.1.1 Initial Phase

Once the T-UCD project has been decided to be implemented, the problem statement, objective of the project, project significance, target user, methodology, project limitations, project scope and research question were identified. Details for all of these have been explained in Chapter 1 (Introduction). In addition, a time line indicating when the project developer will perform various step of the T-UCD project has also been prepared. This time line is presented in a form of Gantt chart that shows the start date, end date and duration as well as tasks implemented for each project step in each phase.

3.2.1.2 Requirement Analysis Phase

This phase is the most crucial part of this project. “A requirement is a description of what a system should do”, (Leszek 2001). In this phase, questionnaire survey analysis and focus group discussion have been conducted to collect user requirements for the system. After that, the results of the questionnaire and focus group were analyzed. This results to the
identification of functional and non-functional requirements. In terms of the functional requirement, use case diagrams have been drawn to represent the functionality of the system from the users’ perspective as well as to identify the scope of the system. All of these requirements were explained in detail in Chapter 4 (Requirement Analysis).

3.2.1.3 System Design Phase

The design phase builds on the knowledge obtained from the analysis phase for which all of the acquired system requirements are translated into a ‘blueprint’ for the software construction. The blueprint depicts a holistic view of the software. Analysis and design is a crucial phase in the entire development cycle. Any glitch in the design phase could be very expensive to solve in the later stage of the software development. Thus, T-UCD uses Unified Modeling Language (UML) as the specification and design method for this project.

The system design phase is divided to two phases which are preliminary design and detailed design. In the preliminary design, entity relationship diagram and class diagram were constructed to show a data design of the system. As for the architectural design, data flow diagram and interaction overview diagram have been designed. After the entire diagrams have been successfully designed, graphical user interfaces and subsequently component diagram which shows a component design of the system were designed.

In the detailed design phase, each module is explained in detailed and the algorithms of each module are designed. All diagrams can be referred in Chapter 5 (System Design).
3.2.1.4 Implementation Phase

The implementation stage of software development is the process of converting a system specification into an executable system. Therefore, during this phase a lot of focus has been given to the programming aspects as well as the testing activities for the T-UCD.

Once new requirements are comprehended, the initial design or a change in the design of a new requirement is done. A new prototype or a revised prototype is subsequently developed. The prototype is iteratively modified to respond to the initial design for new requirements. At the end of each prototyping cycle, the tested prototype is used for further system development.

3.2.1.5 Testing Phase

Testing phase is divided into three phases which are unit testing, integration testing and system testing. The goal of the unit testing is to fully test the logical and programming flows for each module. In this phase, inputs on an unexpected data value for testing during implementation have been provided. Unit testing is conducted to detect defects and fault that could exist in the system.

Integration testing has been carried out to test full version system on real time. In this phase, all T-UCD modules were combined and tested to ensure that the system functions correctly. The tested input consists of cases from user and administrator.
The main goal of system testing is to produce final version of the system. In this phase, requirement testing and non-requirement testing have been conducted before a complete system is run to detect possible errors.

### 3.2.1.6 Evaluation and Acceptance Phase

This is the final phase of the system development. The goal of this phase is to obtain feedback from user about the system to evaluate whether the user requirements are fulfilled. To achieve this goal, user acceptance testing has been performed by conducting a questionnaire survey. This survey involved 20 students and the result from the survey can be referred in Chapter 7 (System Testing and Evaluation).

### 3.3 Research Techniques

At the core of systems analysis is the collection of information. The designer must learn on about problems, opportunities, constraints and terminology of the information systems that are currently being used and must understand how users would like to improve the current systems and organizational operations with new or replacement information systems. Many ways exist to get this information using the fact-finding techniques. Common methods include reviews on documentation, questionnaires, interviewing, participant-observation, naturalistic enquiry and good old-fashioned research.

For the purpose of this project, research techniques, questionnaire survey analysis and focus group discussion have been used in order to gather all required information.
Considerable research had been performed to obtain information and news on latest and technologies through:

i. Computer trade journals and citations  
ii. Latest periodicals and books  
iii. Sites through the internet

### 3.3.1 Computer Trade Journals and Citations

Computer trade journals and citations provide information on previous researches especially on how others have solved similar problems. Most white papers or publications of the journals and citations which are related to this project are obtained from IEEE Xplore, ACM and CITESEER websites.

IEEE Xplore is one of the most popular online delivery system that provides full text access to the world's highest quality technical literature in electrical engineering, computer science and electronics. IEEE Xplore contains full text documents from IEEE journals, transactions, magazines, letters, conference proceedings, standards and IEE (Institution of Electrical Engineers) publications. Users are permitted to view, download, and print content found in IEEE Xplore for personal use. To access IEEE Xplore, users must use the IEEE Xplore URL which includes the IEEE domain name [http://www.ieee.org/ieeexplore](http://www.ieee.org/ieeexplore) or [http://www.ieeeexplore.ieee.org/](http://www.ieeeexplore.ieee.org/).

Other source for the journals and citations can be found in the portal of ACM. The fundamental components of the ACM Portal are an enhanced version of the ACM Digital Library plus an extended bibliographic database, consisting initially of more than a quarter-million citations of core works in computing. These works are of all types such as journals,
proceedings, books, technical reports, theses and they are from all the major publishers in the discipline. The ACM Portal thus provides an Online Guide to Computing Literature and a "reading room" for ACM's own literature in the ACM Digital Library.

Meanwhile, there are scientific literature digital library and search engine that focuses primarily on the literature in computer and information science, which are called Citeseer.IST (Scientific Literature Digital Library). Rather than creating just another digital library, CiteSeer provides algorithms, metadata, services, techniques and software that can be used in other digital libraries. CiteSeer indexes PostScript and PDF research articles on the Web. CiteSeer is capable to show the context of citations to a given paper, allows to quickly and easily seeing on what other researchers have to say about an article of interest.

3.3.2 Latest Periodicals and Books

To get an in-depth understanding on the software engineering and software maintenance itself, developer must refer to the latest periodicals and books. All relevant materials can be found in the University Malaya’s Library. To get more information about the library, they are suggested to browse the given url: www.umlib.um.edu.my. A lot of readings have been performed on the theory of software engineering by much emphasis given on the theoretical aspects of development process, software maintenance, maintenance tools as well as the theory and the example of possible scenarios available in many existing database management systems.
3.3.3 Sites Through the Internet

Some extensive researches on areas relevant to the T-UCD project have been done by attending to the online tutorials or guided tour found on web sites that serves as well known search engines for example Yahoo.com, Google.com and Altavista.com. Additional to this, existing algorithms and sample programs particular to those which implement maintenance in database system in VB.net have been analyzed through sites browsing to obtain guidelines for the coding of the T-UCD programs.

3.3.4 Questionnaires Survey

Often, projects fail because of resulting data are not useful to the development of a project. A measurement project can be more successful if it is designed with goals in mind. Due to this, to design a questionnaire, this project has been referred to the Goal Question Metrics (GQM) (Fenton and Pfleeger 1998) approach to ensure this project is archives its goal. The GQM is first suggested by Basili and his colleagues in 1980’s (Fenton and Pfleeger 1998).

By employing this approach, the first step usually express the overall goals of the project. Subsequently, questions which will be included into the questionnaire are generated. Target respondents of these questions are also identified to determine if the goals are met. Finally, each question is studied to define necessary measurements in order to answer each question.

To ensure the success of this project, a questionnaire survey had been carried out to attain necessary information and opinions from target respondents. Questionnaire is a quantitative technique and an effective method which can be used as part of a research especially when
this method is capable to gather numerous respondents in a limited time. By this way, time is saved because it does not require meeting and interviewing each and every one of her respondents.

The questionnaires are created to analyze the respondents’ opinions towards UML diagram and use case diagram. Questions such as what they know about use case diagram?, what is level of understanding on use case diagram?, what are the problems that occur in drawing use case diagram using existing systems?, and user opinion towards having system that is capable to help them in drawing use case diagram are a number of issues that should be considered.

This questionnaire survey is done to collect responses and opinions from respondents on use case diagram. The respondents were given a list of questions to answer. The questions include closed questions (yes/no), multiple choice questions (choose only one or choose all that apply) and open question (ask for comments). The questionnaires were printed out on paper and also through the online survey. The online survey is carried out using the Infopoll Designer free website.

A total of 200 respondents, from various backgrounds, race, age and gender had taken part in this questionnaire survey. The target respondents of this questionnaire are students from Faculty of Computer Science and Information Technology in University of Malaya and University College of Technology Twintech. Details of the questionnaires can be referred in Appendix A. The sampling was also chosen based on user’s knowledge in UML. The data and analysis compiled from the questionnaires can be referred at Chapter 4 (Requirement Analysis), and the details of the results are attached in Appendix B.
3.3.4.1 Goal-Question-Metric Approach

As previously mentioned in section 3.3.4, this research employs the Goal-Question-Metric (GQM) approach to make it more successful. The GQM approach provides a framework involving three steps which are:

i. List the major goals of the development or maintenance project.

ii. Derive from each goal the questions that must be answered to determine if the goals are being met.

iii. Decide what must be measured in order to be able to answer the questions adequately.

The main goal of the questionnaire survey is to identify the student’s problem in order to draw a use case diagram. After goals have been identified, several questions are selected to achieve the goal. Once these questions are identified, each question was analyzed to determine what must be measured in order to answer the question. Table 3.1 illustrates the Goal Question Metric for the questionnaire survey.
Table 3.1: Goal Question Metrics

**SECTION A: RESPONDENT INFORMATION**

<table>
<thead>
<tr>
<th>GOAL:</th>
<th>To identify the respondent information.</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUESTIONS:</td>
<td>What is your gender?</td>
</tr>
</tbody>
</table>

**SECTION B: OPINION ON UML**

<table>
<thead>
<tr>
<th>GOAL:</th>
<th>To examine the respondent proficiency in UML.</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUESTIONS:</td>
<td>What is your proficiency of UML?</td>
</tr>
<tr>
<td>METRICS:</td>
<td>Proficiency of respondents in UML.</td>
</tr>
</tbody>
</table>

**SECTION C: OPINION ON USE CASE DIAGRAM**

<table>
<thead>
<tr>
<th>GOAL:</th>
<th>To evaluate the respondents knowledge of use case diagram.</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUESTIONS:</td>
<td>What do you think about drawing a use case diagram?</td>
</tr>
<tr>
<td>METRICS:</td>
<td></td>
</tr>
<tr>
<td>METRICS:</td>
<td>Percentages of respondents thinking.</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------</td>
</tr>
</tbody>
</table>

**SECTION C: OPINION ON USE CASE DIAGRAM (cont...)**

**GOAL:** To evaluate the respondents knowledge of use case diagram.

<table>
<thead>
<tr>
<th>QUESTIONS:</th>
<th>Do you find that you had to rely on your own interpretation (at least, in part) of the differences between <em>include</em> and <em>extend</em> relationship?</th>
<th>If yes, how much did you have to rely on your own interpretation?</th>
<th>How of often did you make mistake with the <em>include</em> and <em>extend</em> relationship?</th>
<th>Do you think that you tried to avoid these relationship because of the uncertainty of their meaning?</th>
<th>Do you find that you tried to ignore the system boundary for the use case diagram because of the uncertainty of their importance?</th>
<th>Please write 1 to 6 into the column according to the steps that you follow while drawing a use case diagram.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>METRICS:</th>
<th>Percentages of respondents’ interpretation.</th>
<th>Respondents’ knowledge about <em>include</em> and <em>extend</em> relationships.</th>
<th>Respondents’ knowledge about system boundary.</th>
<th>Sequence of steps.</th>
</tr>
</thead>
</table>
### SECTION D: OPINION ON CURRENT UML CASE TOOL

**GOAL:** To determine the weaknesses of current UML CASE tools.

<table>
<thead>
<tr>
<th>QUESTIONS</th>
<th>METRICS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Have you ever drawn a use case diagram using other UML CASE tools?</td>
<td>Experience of respondents in using current UML CASE tools.</td>
</tr>
<tr>
<td>Did you encounter any problem when using those tools?</td>
<td>Problems of current UML CASE tools.</td>
</tr>
<tr>
<td>What type of problem?</td>
<td></td>
</tr>
</tbody>
</table>

### SECTION E: SUGGESTION FOR PROPOSED SYSTEM

**GOAL:** To collect the respondents suggestions for proposed system.

<table>
<thead>
<tr>
<th>QUESTIONS</th>
<th>METRICS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Are you interested in having a system that is capable of helping you in drawing use case diagram?</td>
<td>Types of respondents’ suggestions.</td>
</tr>
<tr>
<td>What information should be included in the system to help you in drawing a use case diagram?</td>
<td></td>
</tr>
<tr>
<td>Which one of the following process do you want this new system to tutor you in the process of drawing a use case diagram?</td>
<td></td>
</tr>
</tbody>
</table>
3.3.5 Focus Group Discussion

In order to satisfy the qualitative method, focus group discussions have been conducted to get more information on issues involved in drawing use case diagram among university students. This focus group discussion involved two lecturers and 40 postgraduate students from Faculty of Computer Science and Information Technology, University of Malaya. The lecturers are Madam Siti Hafizah Ab. Hamid and Madam Nazean Jomhari. They are teaching a UML subject in the faculty. The students are in Master of Computer Science program and they are taking Object-Oriented Technique course of the 2005 / 2006 session. The students are given one complex case study on Car Park System and then the students’ answers are discussed with the lecturer. The case study came from a compact disk (CD) which is attached with the Object-Oriented System Analysis and Design Using UML text book (Bennett, et al. 2002). The students were asked to do the case study in a group of five students. Altogether, eight samples of students’ answers were discussed in this focus group. Result of the discussion can be referred in Chapter 4 (Requirement Analysis) and details of the case study can be referred in Appendix C.

3.4 SUMMARY

In this chapter the methodology used to build up the system which consists of software life cycle model and fact finding techniques are discussed. Particular for this research, the goals, tasks and output for each phase in the system had integrated with the V-
Shaped software lifecycle model. This research uses triangulation, a mixed method of quantitative and qualitative approaches. In the qualitative method, focus group discussions had been conducted while in the quantitative method; findings from the questionnaire designed in collaboration with the focus group had been explored. The result analysis from the questionnaire was summarized into graphical statistics and was captured as the basis of user requirement for the system. The questionnaire result analysis also solves one of the research problem statements which were to study and analyze the users’ need in drawing UML diagram and to find out possible problem that could occur in drawing UML diagram especially in use case diagram. Based on the questionnaire result and analysis, a few number of important information in tools of drawing UML diagram were identified. Based on the questionnaire survey analysis, the main features that should be included in the proposed system had identified. In the focus group discussion, few issues related to use case diagram among university students have found. By combining the result from both methods, this research aims to arrive at a more nuanced understanding of the problem in drawing a use case diagram among university students.