Implementation of Dynamic Graphics in Problem Reporting System for ICT/BIS Academic Buildings

by

Diyana Bt. Rasid

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Information Technology (Hons)
(Information Technology)

JUNE 2006

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

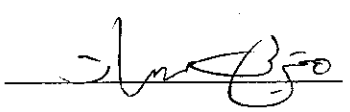
Implementation of Dynamic Graphics in Problem Reporting System for ICT/BIS Academic Buildings

by

Diyana Binti Rasid

A project dissertation submitted to the
Information Technology Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirements for the
BACHELOR OF INFORMATION TECHNOLOGY (Hons)
(Information Technology)

Approved by,

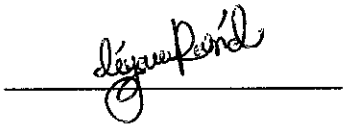


(Ms Mazlina Mehat)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
JUNE 2006

CERTIFICATION OF ORGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein has not been undertaken or done by unspecified sources or persons.



DIYANA BINTI RASID

ABSTRACT

This study deals with matter pertaining on developing a system that allows UTP community to have quick online problem reporting access to Property Management and Maintenance Department (PMMD) for ICT/BIS academic buildings. Currently, the paper-based forms and procedures employed by the department typically involve cumbersome reporting steps and result in inefficient organizational processes when attempting to use the information for overall improvement. The objectives of this project are to easily record and identify which area of ICT/BIS academic buildings are in need of attention; to produce excellent dynamic graphical problem reports and analysis; and to build a secured system through its design and engineering. The project revolves around getting to know the programmable graphics techniques such as automatic image map generation and dynamic charts rendering, and also building dynamic images driven by the database. To achieve the above objectives within seven months time frame, the author follows Software Development Life Cycle (SDLC) model and complement it with the use of Evolutionary Prototyping approach. In conclusion, the extensive usage of interactive mapping and dynamic charts can definitely increase the web navigation efficiency of this proposed system.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

2-D	Two-dimensional
3-D	Three-dimensional
API	Application Programming Interface
CGI	Common Gateway Interface
HREF	Hypertext Reference
HTML	Hypertext Markup Language
ICT/BIS	Information & Communication Technology/Business Information System
JDK	Java Development Kit
JSP	Java Server Page
PMMD	Property Management and Maintenance Department
SDLC	Software Development Life Cycle
SVG	Scalable Vector Graphics
UI	User Interface
UTP	Universiti Teknologi PETRONAS

CHAPTER 1
INTRODUCTION

1.1 BACKGROUND OF STUDY

Ever since its birth in the early 90's, World Wide Web has gained tremendous popularity in a manner that is beyond anyone's imagination. An important factor contributed to the Web phenomenon is the introduction of dynamic content. As a result, dynamic content opens doors for web e-commerce, individual-customizable site and interactive online gaming. Recently, web applications become more sophisticated, developers increasingly need to be able to incorporate dynamic graphics and animation. Dynamic graphics make the site visually interesting, add aesthetic value, and provide interactivity.

This project revolves around getting to know the programmable graphics techniques such as automatic image map generation and dynamic charts rendering, and also building dynamic images driven by the database. The main concern of the project is the spatial or graphic navigation, to illustrate the problem submitted by user graphically instead of to just show the problem information in text, forms and tables only.

A navigation style gives information about the content and structure of a site. It determines how easy the users can access information. One of these styles is the spatial or graphic navigation. Graphic navigation gives users the opportunity to move around the site through visual representations instead of a list of choices.

Statisticians and data analysts have always used pictorial representations of data (graphs and charts) to discover and explore relationships between variables, and to communicate their results to other people. Developments in computer hardware and software have now brought dynamically interactive graphs to everyone's desktop.

1.2 PROBLEM STATEMENT

In large academic buildings area, there are many things that can go wrong - light bulbs burn out, air conditioner not functioning, broken glass-wall, water/power supply interruption, and ceilings leak. Since UTP staffs and the facilities have grown in size, complexity and usage, the demands placed upon those responsible for ensuring the continued operation and security of these buildings has also increased.

Currently, UTP PMMD fully uses paper-based procedures in managing each reported problem. The paper-based forms and procedures typically involve cumbersome reporting steps and result in inefficient organizational processes when attempting to use the information for overall improvement. Thus, confidence in the effectiveness of the procedure may be diminished, perpetuating the cycle of poor problem reporting system.

The utilization of reliable online access and extensive graphical data representations are required to increase the web navigation efficiency of this proposed system. People like to see their complaints addressed and this system must be capable to speed up problems getting fixed by quickly notify people who need to do the work. This has lead to a demand for automated problem reporting and online tracking system with effective visualization of data.

1.3 OBJECTIVES AND SCOPE OF STUDY

The project aims to allow UTP community to have quick online access to PMMD. Purposely, this problem reporting system is for the new ICT/BIS Academic Buildings consist of Building 1 and Building 2. This project has the following objectives:

- To enable UTP staffs and administrators of PMMD to easily record and identify which area are in need of attention.
- To produce dynamic graphical problem reports and analysis conforming real situation/mapping.
- To build a reliable system to remain dependable in the face of malice, error or mischance through its design and engineering.

1.3.1 Feasibility of the Project within the Scope and Time Frame

There are three groups of user with different privileges – Administrator, Approver and Submitter. The system will report on each status of a problem graphically and by color coded scheme. Emails mechanism will be employed for the delivery of notifications and alerts. There will be dynamic tables and charts for the management to find out what kinds of problems are most frequent, how long it takes on average to fix problems, and which problems have been outstanding the longest.

For this project, the time frame given is around seven months from October 2005 until April 2006 (see Appendix I: Project Schedule). The author will spend two months to carry out research and writing paperwork. The remaining five months will be allocated in executing the project. The project will be divided into several phases in accordance to the problem solving process flow of UTP PMMD.

CHAPTER 2
LITERATURE REVIEW AND THEORY

2.1 PRESENTATION OF INFORMATION BY A GRAPHICAL LANGUAGE

Graphics can convey messages instantly. It also makes a multimedia title aesthetic in appearance and interesting. Besides, it also can capture user attention and mood more than textual presentation. The use of pictorial information or graphics should be consistent throughout the multimedia title so that user could easily identify them and not being confused (McGloughin, 2001).

Visual representations can be an important part of a successful website. When designing a website, it is important to determine if an image map or other navigation aid can be most appropriate for the user. The whole point of image map is to provide more graphical interaction without much overhead (Fleishman, 1996). Online mapping is driving a new breed of web sites that help users locate the services they need (Watson, 2005).

Spatial or graphic navigation means using visual representations of objects in space to discriminate among choices. Spatial representations can be either two or three dimensional, and pictorial or symbolic. Others called it hypermedia (McKnight, 1996) or image maps. With this navigation style, graphics act as buttons to link to other pages. Maps, plans, and elevations are examples of two dimensional spatial navigation. Surrogate travel is an example of three dimensional spatial navigation (Hoffman, 1999).

Does the presentation of editorial content in graphical format help readers understand and remember more story information? With many news organizations investing time, money, energy, and human resources into online multimedia presentations, the Eyetrack III team was curious. "Recall of Information Presented in Text vs. Multimedia Format" report is one of many from the Eyetrack III study of broadband-era websites. The results were published in September 2004.

Eyetrack III Team's (2004) study found the following:

> It is important to note that in the test there are two modes of communication for the multimedia format -- text and moving illustrations. Findings in the "Recall of Information Presented in Text vs. Multimedia Format" report reveal that information about a process or procedure that was unfamiliar to users was more correctly recalled when participants received it in a multimedia graphic format. When asked to recall information about a process or procedure or to define vocabulary related to a new process that few participants were likely to have ever heard of before, participants who received information in multimedia graphic formats appeared to learn more effectively and were more likely to answer questions correctly.

2.2 DYNAMIC GRAPHIC

Dynamic content, which was made available by technologies like CGI (Common Gateway Interface), is the first "killer" Web application, as it makes Web more interactive and possible for e-commerce. Now, dynamic image creation, which can be the next "killer" application, is pushing dynamic content to a new level. (Guang Yang, 2004)

There will be extensive usage of interactive mapping and dynamic charts in this project. This project will study the efficiency of problem reporting system by using visualization of data.

2.2.1 Generating Dynamic Image Map

One of the things that mystify newcomers to the Web is how to set up an image so that when you click on something in it, you are taken to specific location on the website. The answer: image mapping. Image maps allow a user to click on an image, have that click translated into a set of (x, y) coordinates in pixels relative to the image, and then have those coordinates translate into a location or resource on the site network. The process only seems mysterious. (Fleishman, 1996)

As mentioned by Fleishman G. (1996), there are server-side image mapping and client-side image mapping. In server-side image mapping, the coordinates are sent off to a server, where a program is run (through the HREF reference); the result is that you are taken to the location those coordinates point to. This takes longer and sucks up a little server juice in the process; it can be tedious. In client-side mapping, servers are omitted. Using a set of new HTML tags, you provide exactly the same kind of directives, but the browsers itself recognizes the locations that the image map points to as it loads the image-map page, so there is no delay when you click on a mapped object.

An example of dynamic online image mapping website can be retrieved from http://iwin.nws.noaa.gov/iwin/graphicsversion/rbigmain.html and is shown in Figure 1 below. It is taken from one of Interactive Weather Information Network (IWIN) websites. The website displays weather condition in each state in America. The weather results will be automatically updated every five minutes. Visitor can click at any specific map to get more detailed weather information (see Figure 2).

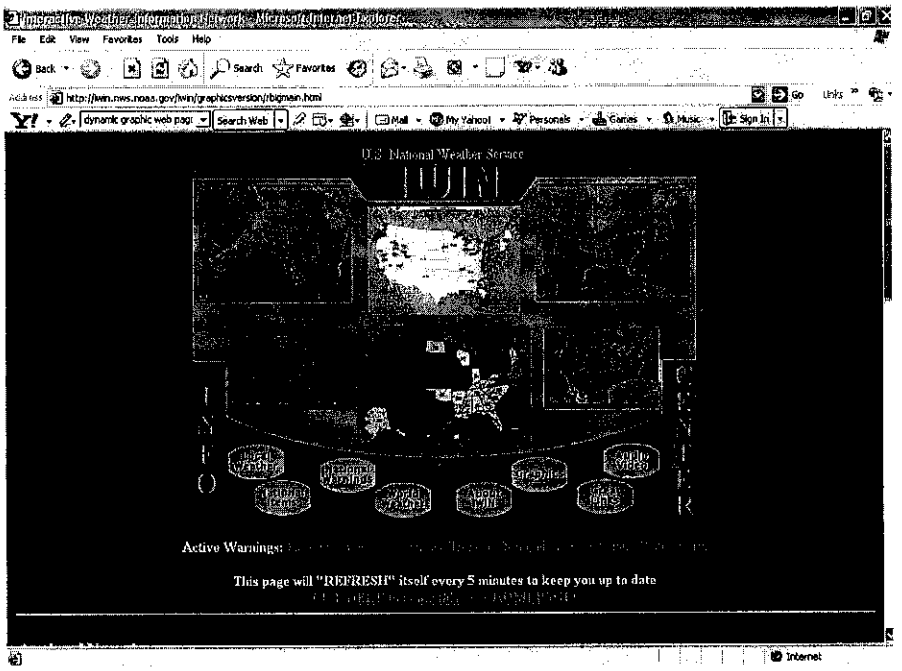


Figure 1: IWIN U.S National Weather Service Website (Homepage)

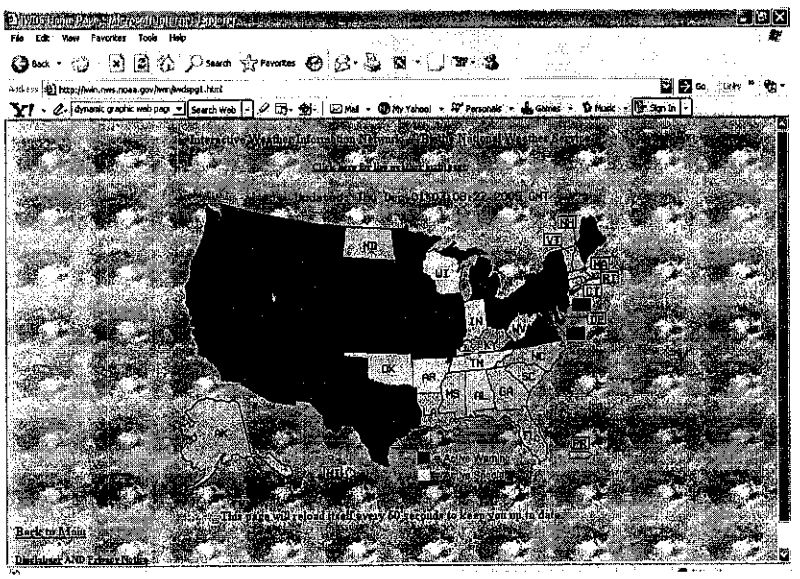


Figure 2: IWIN U.S National Weather Service Website (Color-coded State)

Innovative online mapping services are now available for just about any need. http://www.weatherbonk.com/weather/index.jsp links data from several weather services to display appropriate weather icons on an online map of the selected location. Problem reporting and tracking systems are also fairly common. In http://www.communitywalk.com/map/65, tourists can virtually tour London on sites by clicking on landmarks to get details and photos. However, the author intends to extend the usage of such systems beyond just linking maps to useful chunks of data.

2.2.2 Generating Dynamic Chart

When it comes to dynamically creating a graphic for a web page, even an image of the simplest form becomes tricky. Only a few options are available for a web designer to create graphics dynamically. For instance, a developer can write a Java Servlet to generate images with Java 2D API. Apparently, not all web designers have the technical expertise to do that. In addition, modifying an image created by a program requires more time and effort. (Guang Yang, 2004)

Currently, the author is considering two methods for generating dynamic chart in the online system – JFreeChart and Java & SVG. Gilbert (2005) described JFreeChart as a free chart library for the Java™ platform. It is designed for use in applications, applets, servlets and JSP. JFreeChart is distributed with complete source code subject to the terms of the GNU Lesser General Public License. JFreeChart can generate pie charts (2D and 3D), bar charts (regular and stacked, with an optional 3D-effect), line and area charts, scatter plots and bubble charts, time series charts (including moving averages, high-low-open-close charts and candlestick plots), Gantt charts, meter charts (dial, compass and thermometer), symbol charts, wind plots, combination charts and more. JFreeChart is written entirely in Java, and should run on any implementation of the Java 2 platform (JDK 1.2.2 or later). Shown in Figure 3 is an example of JFreeChart output.

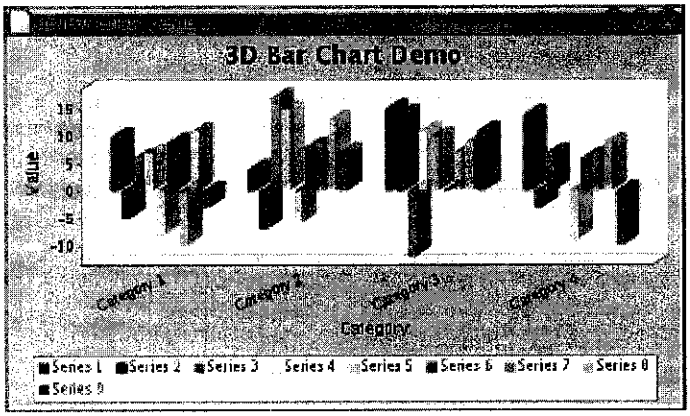


Figure 3: JFreeChart Sample Chart

Gilbert (2005) affirmed JFreeChart is free software, so anyone can extend it and add new features to it. Already, more than 80 developers from around the world have contributed code back to the JFreeChart project. It is likely that many more chart types will be developed in the future as developers modify JFreeChart to meet their requirements.

Another alternative is building 2-D graphics applications using Java and SVG software. For brevity and to stick closely to the concept, business logic is embedded in the SVGController servlet's service method, rather than using a separate model layer. The application shows how to integrate SVG into the UI generated by the JSP (Refer to Figure 4).

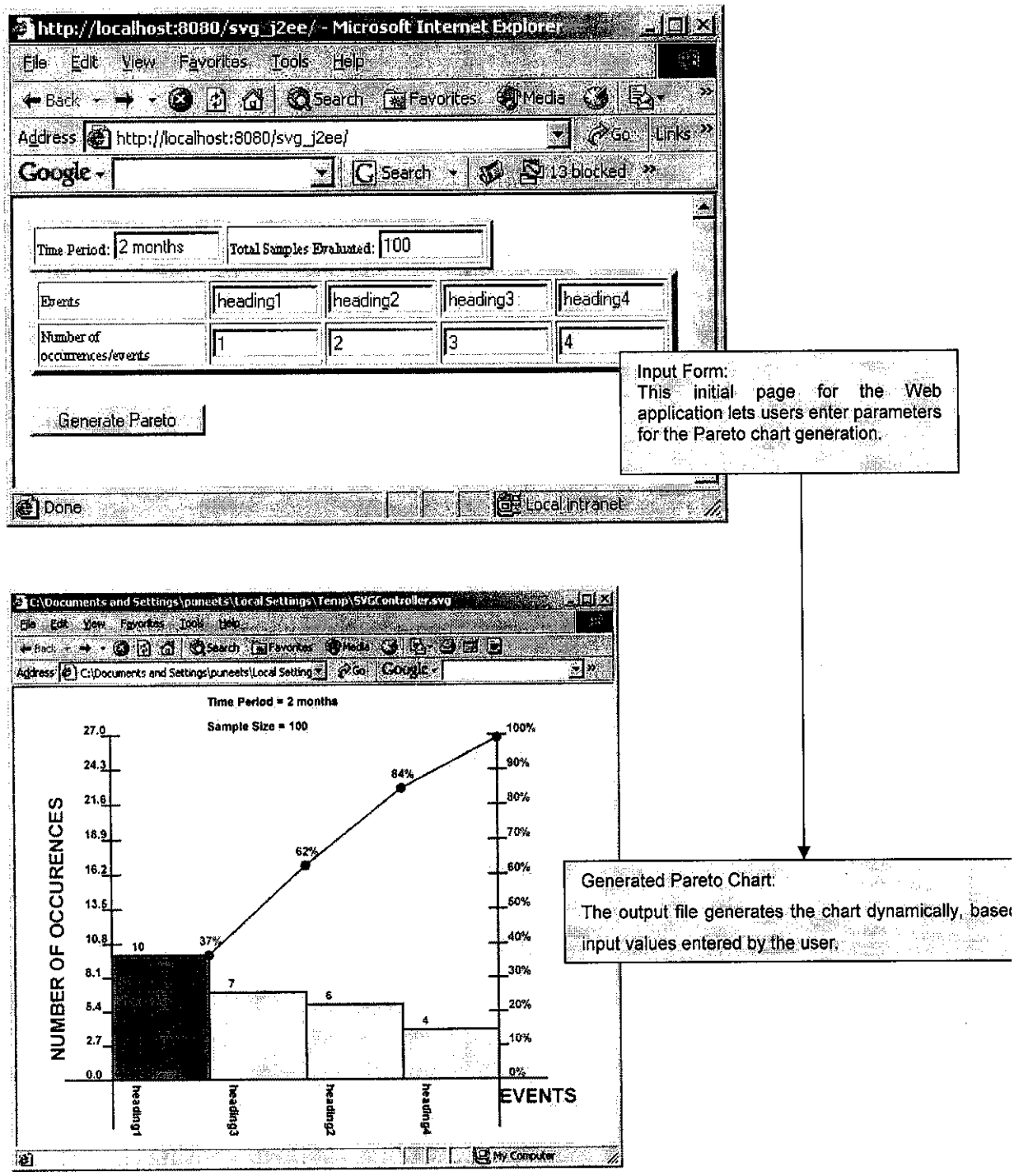


Figure 4: Java & SVG Sample Chart

SVG is robust enough for use in heavy 2-D graphics applications such as the printed circuit board industry. However, at this stage, SVG's 3-D capabilities are still somewhat limited. SVG technology is already mature enough that there are a few books available - and there is plenty of information on the Web if you want to learn more. (Puneet M. Sangal, 2005)

CHAPTER 3
METHODOLOGY/PROJECT WORK

3.1 PROCEDURE IDENTIFICATION

From the view of Pressman (1997), SDLC is "an organizational process of developing and maintaining system" (p. 32). System development life cycle consist of combination of various activities.

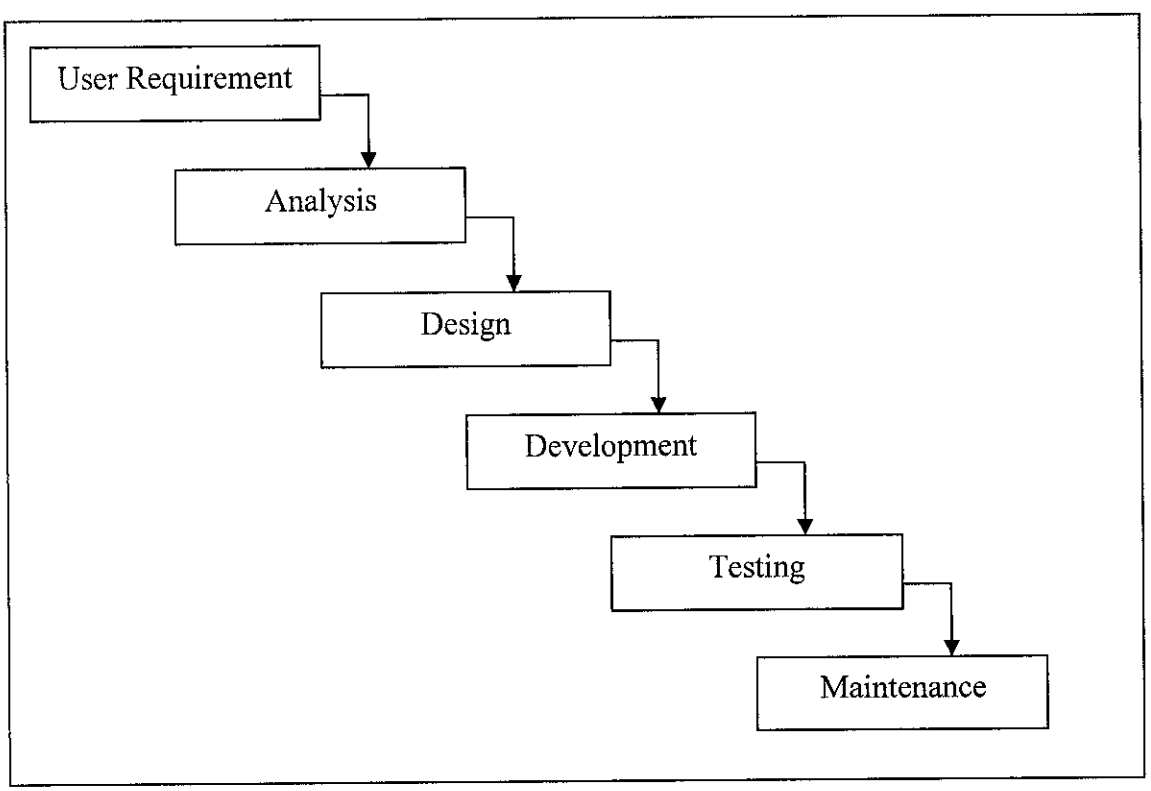


Figure 5: System Development Life Cycle (SDLC) Diagram

For this project, the author is going to implement the SDLC model and use the Evolutionary Prototyping approach. In general, these are the SDLC phases for this project:

1) **User Requirement**
2) **Analysis**
3) **Design**
4) **Development**
5) **Testing**
6) **Maintenance**

Issues about going through the SDLC center around two interrelated main concerns. The first concern is the extended time required to go through the development life cycle. As the investment of analyst time increases, the cost of delivered system rises proportionately. The second concern about using the SDLC is that user requirements change over time. The two concerns are interrelated, because they both pivot on the time required to complete the SDLC and the problem of falling out of touch with user requirements during subsequent development phases.

To overcome these problems, Evolutionary Prototyping is an approach used to complement the SDLC model. According to Sommerville (2001), "Evolutionary Prototyping is based on the idea of developing an initial implementation, exposing this to user comment and refining this through many stages until an adequate system has been developed. This way, it effectively shortens the time between ascertainment of information requirements and delivery of a workable system. With prototype, users can actually see what is possible and how their requirements translate into hardware and software." (p. 175).

3.2 TOOLS REQUIRED

3.2.1 Software

- Apache Tomcat 5.x Web Server
- MySQL Database
- Macromedia Dreamweaver MX 2004
- Java Compiler j2sdk1.4.2_03
- Internet Explorer 5.5 and above

3.2.2 Hardware

- A well equipped and fairly new Windows or Macintosh PC

- At least 3 GB of free space on hard drive

- Enough RAM to run at least three applications at once, such as a Web browser, Macromedia Dreamweaver MX, and Microsoft Word

- A good color monitor supporting at least 800 x 600 pixels and 16bit color; 1024 x 768 pixels or more is recommended.

- A removable disk drive or other reliable way of backing up work

3.3 USER REQUIREMENT

User requirement is very important in any project. The developer would be able to know what user wants and the problem that user had faced.

Identifying problems, opportunities and objectives

Valuable information is gathered by interviewing the prospective users and by reviewing websites and journals. The existing problem reporting process flow of PMMD is evaluated. Deficiencies are identified. These are done by interviewing

intended user of the system (see Appendix II: Interview Details) and consulting with support personnel.

The author managed to interview the personnel of UTP PMMD, Mr. Hazizi Laili. The interview session was held in his office on 11th October 2005 at 10.45 am. List of interview questions are prepared a day before the interview (see Appendix II: Interview Details).

The purpose of the interview is to determine two issues:

- To determine the level of acceptance of UTP PMMD on the proposed idea: Dynamic Graphical Problem Reporting System
- To determine the scope of the online system: ICT/BIS Academic Buildings

Determining information requirements

Strive to understand what information users need to perform their jobs. This segment serves to fill in the picture that the author has of the organization and its objectives. The author gets a clear picture of what actually the courseware being developed would likely be. The author has identified background of the project, the scope, objectives and problems with current system.

The users need a new way in reporting and interpreting building's property problems instead of using forms and tables as the main representation. Therefore the completion of this Dynamic Graphical Online PMMD Problem Reporting System would give a new dimension in property problem management whereby it is not only used for its interactive and attractive features, it also provides convenience to users in conditions of time constraint and work demand.

3.4 ANALYSIS

This phase involves analyzing system needs and structured decision made. Based on the interview, the scope of this Dynamic Graphical Online PMMD Problem Reporting System is identified. The author decided to cover only ICT/BIS new academic buildings as preliminary effort for the PMMD first online problem reporting system. A broad range of problem types crossing academic buildings environments are described by Mr. Hazizi Laili. Some examples of common problems are related to air-conditioning system, lighting, furniture, building's fabric, building's lift, and toilet.

The current paper-based problem report management flow is discussed thoroughly with Mr. Hazizi and can be illustrated in Figure 6 below:

Step 1: Entry	Step 2: Event Received	Step 3: Investigation	Step 4: Action	Step 5: Feedback

Figure 6: Current PMMD Problem Report Management Flow

See Appendix VI for the current PMMD problem report management process flowchart.

Description of Current PMMD problem report management flow:

Firstly, problem information is filled in the PMMD Work Request/Defect Form (see Appendix III). The event received by PMMD personnel is segregated between mechanical & electrical engineering group or civil engineering group. Then, the problem will be investigated and solved by the assigned group within planned time limit. Finally, feedback or verification is obtained from either the problem submitter or problem solver.

To add to understanding of the worth of the idea, the author conducted a predevelopment survey. It was conducted by distributing questionnaires randomly to students and staff. The objective of the questionnaire is to gather information, opinion and feedback for the research work. With the help of the result gathered from this survey, the author will be able to come out with an effective Online PMMD Problem Reporting System.

3.5 DESIGN

The author uses the information collected earlier to accomplish the logical design of the system. In this phase, the online system design is divided into three parts:

- System Flow and Architecture
- System Interface or Screen Design
- The Database Structure

3.5.1 System Flow and Architecture

A few system modeling are created to get a clear view of the proposed system especially in term of its workflow. Likewise, the system modeling facilitates the author in understanding the system for the development phase. The author developed Use Case Diagram that shows the interaction between external entities (users) and the system. The Use Case Diagram is concerned with the functions provided by the online system to each type of users – Submitter, Administrator, and Approver. It is a depiction of a system's behavior or functionality under various conditions as the system responds to requests from users. Please refer to Appendix IV.

Online PMMD Problem Reporting System (ICT/BIS Buildings) Architecture:

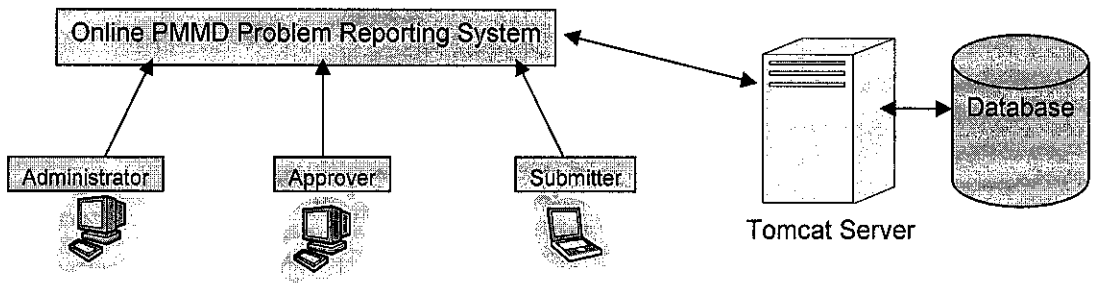


Figure 7: System Architecture

The system architecture diagram (refer Figure 7) illustrates the interaction between the client and the server. It can be seen that the client (user) issues a request to the Java Tomcat server, the server then passes the request to the JSP engine to dynamically construct the data the user requested. This data is then passed back to the server, which in turn issues the client with the response. This is the structure on which will base the author's project. It can be seen that the user can enter queries via the browser (for examples: submit a new problem report and search a list of a certain type of property problems).

3.5.2 System Interface or Screen Design

The interface or front end of the system is carefully designed. The design will guide the author in the development phase. It is critical to determine where the content will reside in the online system, how to construct the interface and how to construct the navigation graphically. All the required information which had been identified before are included in the design. The arrangement of images, forms, text boxes, buttons, and labels are identified in order to get a user-friendly interface.

The usages of graphics are found to be good for representing information, but their overuse can be distracting and congesting. In this Online PMMD Problem Reporting System (ICT/BIS Buildings) graphics are widely used to represent each problem location, problem status, background and buttons. Therefore, the author decided to use graphics that are simple enough to convey complex problem status and locations.

3.5.3 The Database Structure

The key for uncomplicated development stage of this Online PMMD Problem Reporting System (ICT/BIS Buildings) is a good database design. The database is organized around problem reports and suitable data structure to record entries. It is more important to get the database right than to write lengthy programs to answer infinite questions.

The relational database management system that the author uses is MySQL. The Structured Query Language (SQL) is an ANSI standard (ANSI SQL92) method facilitating communication with relational databases. SQL uses expressions derived from a reserved list of English words to manipulate and store data. The steps in creating and interfacing with an Internet accessible database involve:

- Database Design
  - Deciding where to put appropriate information in tables, columns & fields
- Connecting to the database server
- Creating the database space on the database server
  - Achieved by declaring a name for the individual database on the server
- Creating the tables with their appropriate column headings
  - As specified in the database design, for the database
- Input the information (or data) to the tables

- Querying the database and presenting the results obtained
- Editing and updating the information contained within the database

3.6 DEVELOPMENT

Application code, databases, interfaces, and code to integrate PMMD requirements into the online reporting system are constructed during the software construction. Hardware, software, and documentation components which have been individually tested are assembled into functional subsystems and systems. Application components must be integrated. Either part or all of an entire application may be built by assembling components. The author completed and updated all system design and support documentation. Integration is performed iteratively with increasingly larger and more complex combinations of components. The steps in which integration will occur shall be documented in the project's test plans. Integration is completed when entire parts have been assembled and integration level testing has been performed.

3.7 TESTING

Before the system can be used, it must be tested. It is less costly to catch problems before the system is signed over to users. The author follows a cycle to testing:

- Requirements Analysis: Testing should begin in the requirements phase of the Software Development Life Cycle (SDLC).
- Design Analysis: During the design phase, author confers with targeted users in determining what aspects of a design are testable and under what parameter those tests work.

- Test Planning:
    - Test Strategies that includes Functional Testing, Integration Testing and User Acceptance Testing.
    - Test Plans creation.
- Test Development: Test Procedures, Test Scenarios and Test Cases.
- Test Execution: Testers execute the software based on the plans and tests and report any errors found.
- Test Reporting: Once testing is completed, the author determines whether or not the online system is ready for release.
- Retesting the Defects

Not all errors or defects found must be fixed immediately. Some may be caused by errors in configuring the test software to match the development or production environment. Some defects can be handled by a workaround in the production environment. Others might be deferred to future releases of the online system, or the deficiency might be accepted by the business user. There are yet other defects that may be rejected by the author (of course, with due reason) if author deems it inappropriate to be called a defect.

3.8 MAINTENANCE AND CLOSING

Evaluation takes place in every single phase to ensure that one phase of the system development is finished and can proceed with the next phase. Providing user support is an ongoing activity. New users will require training. The emphasis of this phase will be to ensure that the user needs are met and the system continues to perform as specified in the operational PMMD environment. At the closing stage, activities involved finalizing the system, presenting the system, and preparing final documentation and lesson learnt.

CHAPTER 4
RESULTS AND DISCUSSION

The prototype of Online PMMD Problem Reporting System performs as expected. This chapter will explain the result of predevelopment survey, functional and integration testing. It also will represent the screen captures and its brief descriptions.

4.1 PREDEVELOPMENT SURVEY

A predevelopment survey was conducted to establish the worth of the project. The author distributed questionnaires randomly to UTP students, staff (targeted users) and non-targeted users. The objective of the questionnaire is to gather information, opinion and feedback for the research work.

The questionnaire (see Appendix V: Questionnaire) consisted of seven (7) questions asking the preferable features of a problem reporting system. This survey was distributed on paper to 80 people in January and February, 2006. An electronic version of the survey was also distributed through email; 16 were returned. Therefore, total number of returned surveys analyzed in this data set: 96.

According to the survey results, all respondents have an Internet-connected computer and are using Windows operating system. Microsoft Internet Explorer is the most preferred browser as voted by 88 (92%) of the respondents. Interestingly, 94 (98%) of them never use any online based system for property problem reporting, but fully support the idea of having one. Majority of the respondents find that an online based system would definitely help the property problem reporting process.

Shown below (Figure 8) is a pie chart of respondent's consent on how they find an online based system would help for the property problem reporting process:

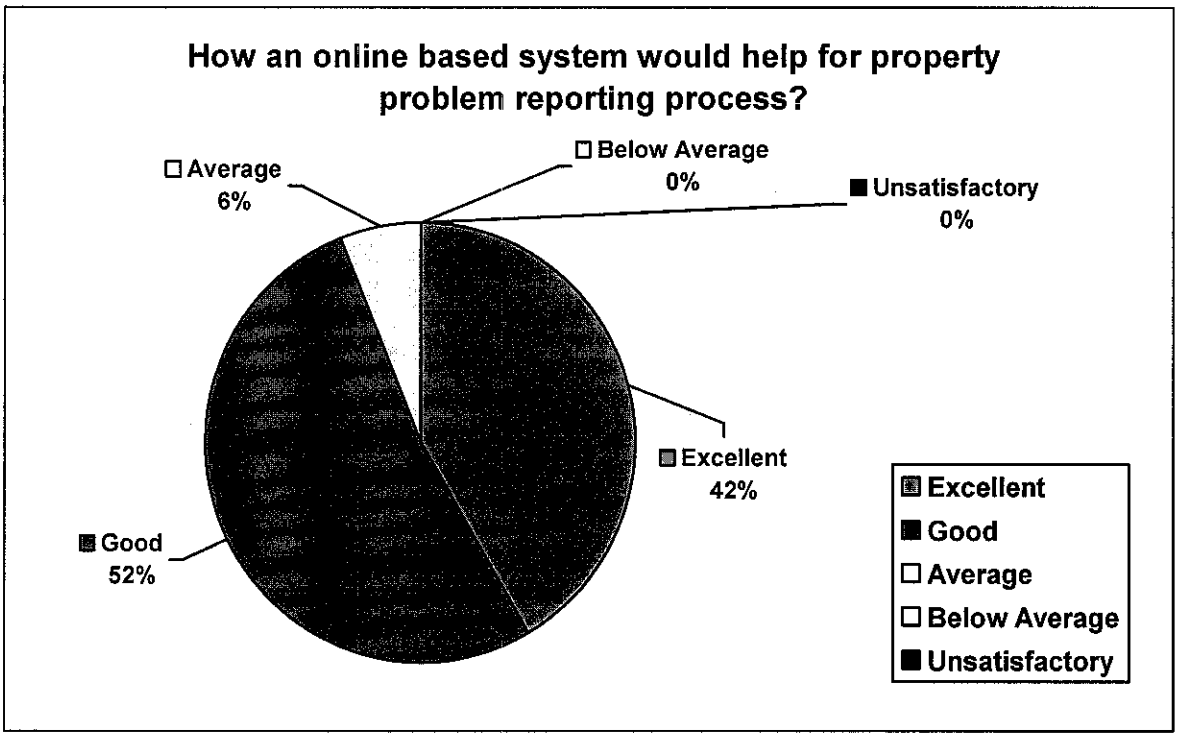


Figure 8: Pie Chart – Question No. 3 Survey Result

There is apparent need of a convenient online based reporting system because currently, 78 (81%) of them will report their property problem to liable department, 11 (12%) of them do not care to report and fix the problem, and the rest will try to fix the problem without reporting it to the department. When asked for additional opinion for this particular question, a number of people who choose to report the problem confessed that they are not satisfied with current procedures that undesirably require their time and energy. Respondents that do not care to report or fix the problem justified that they do not want to involve in the hassle and the problem is not big to worry about.

The most important result of this survey is that 88 (92%) respondents prefer information of reported problems are displayed in combination of both graphic and text, but minimal text. 14 (8%) other respondents prefer reported problems to be displayed in combination of both graphic and text, but minimal graphic. This outcome matches author's project objectives and is shown in pie chart (Figure 9) below:

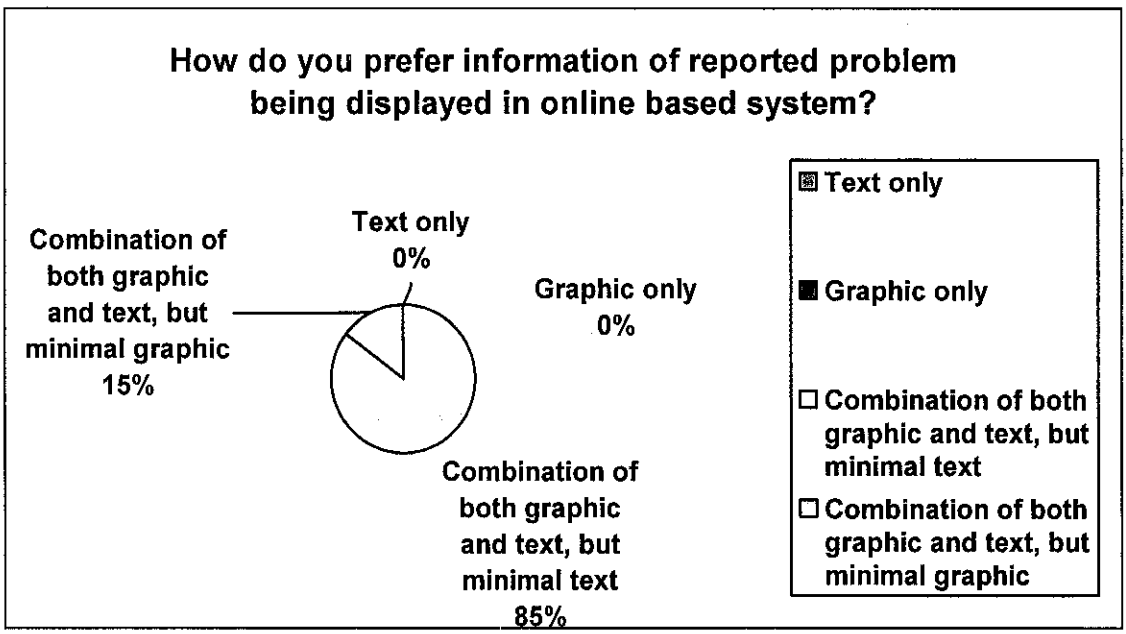


Figure 9: Pie Chart – Question No. 4 Survey Result

During this analysis phase, all important information that needs to be included in the design phase is identified. Project analysis could capture the requirements and hopefully could solve the problems mentioned in the problem statement section.

4.2 FUNCTIONAL TESTING

As described earlier, each subsystem will be tested once developed. This is to detect and debug any flaws before it is made up as a whole system. It is also to ensure each subsystem is well-functioning. All subsystems are tested by using functional testing. A successful functional testing is when expected result or output achieved from the respective input. The test results are described in Table 1 until Table 3 in Appendix VII.

Each subsystem as listed above is functioning as expected. In addition, testing on error-checking functionality for each subsystem is also being carried out. In case of unfilled fields, a pop-up error window will notify the user. Overall, during the testing, the whole system is functioning well without much redesign has been made.

4.3 INTEGRATION TESTING

Integration testing is conducted when each subsystem completely developed in the system as a whole. It is to ensure there is no flaw or error each time integration of subsystems is performed. In case of error found, debugging will be carried out. Under this testing, the systems linkages are also being tested. It is to ensure each link or hyperlink in the system is well-functioning. Besides, this testing is used for ensuring the successfulness of connection between the system and other system components, including database. Testing is done without outsider interference. The results of integration testing can be found in Appendix VIII.

As stated in previous, the integration testing is also being used to test the connection between the system and outside components including the database. The results of this testing are shown in Table 5 in Appendix VIII.

Overall, the integration testing is successfully carried out and the system is performed successfully as whole. There was not much redesign being made along the testing.

4.4 USER ACCEPTANCE TESTING

Even though the system has been put to a rigorous testing in both functional and integration earlier, the author feels that it is also important for the user acceptance test to be also conducted. This is in line with the objective of software development is to develop the software that meets the true needs of the user, not just the system specifications. Another objective of the test is to actually test on the business process flow of the system.

Tasks Scenario:

To make it a success, a few scenarios have been designed by the author, which will allow different kind of user to be involved in the testing of the system. For the purpose of this user acceptance testing, a total of 3 users from PMMD are randomly selected to participate in the testing procedure. Different type of users of the system has been selected to test the system to match the future real users of the system.

Participants are supposed to fill the online report form and submit it, and then browse through the submitted report through its graphical representation. These tasks are selected because they represent common activity a user may do with the Online PMMD Problem Reporting System. The test cases lead the participants (as Submitter/Approver/Administrator) to step through all the important features and procedures of the system, giving it a thorough test in almost all important functional aspects. The results of user activities provide reasonable and accountable measurement of main system parameters.

User activities are observed. After finishing the tasks, participants are asked to answer questions based on their fresh experience with the system. They are asked to rate performance of various aspects of the system and give opinions on overall usability and enjoyment with the system. The answers are analyzed to find out what aspects of the system contribute to the overall satisfaction of the system user.

The usability testing is successfully carried out and the system is performed successfully as whole. Users disclosed they have comfortable experience with the online systems. There are not much different in each user behavior pattern when interacting with the system. They agreed that the online system can productively assists the operations of PMMD.

4.5 SYSTEM INTERFACE

The following section will capture screenshots involved and describe what would be each interface does in the system. Figure 10 shows the screenshots of the Login page. It is the first page of the system. User needs to key in an appropriate Staff ID and password in order to login into the system.

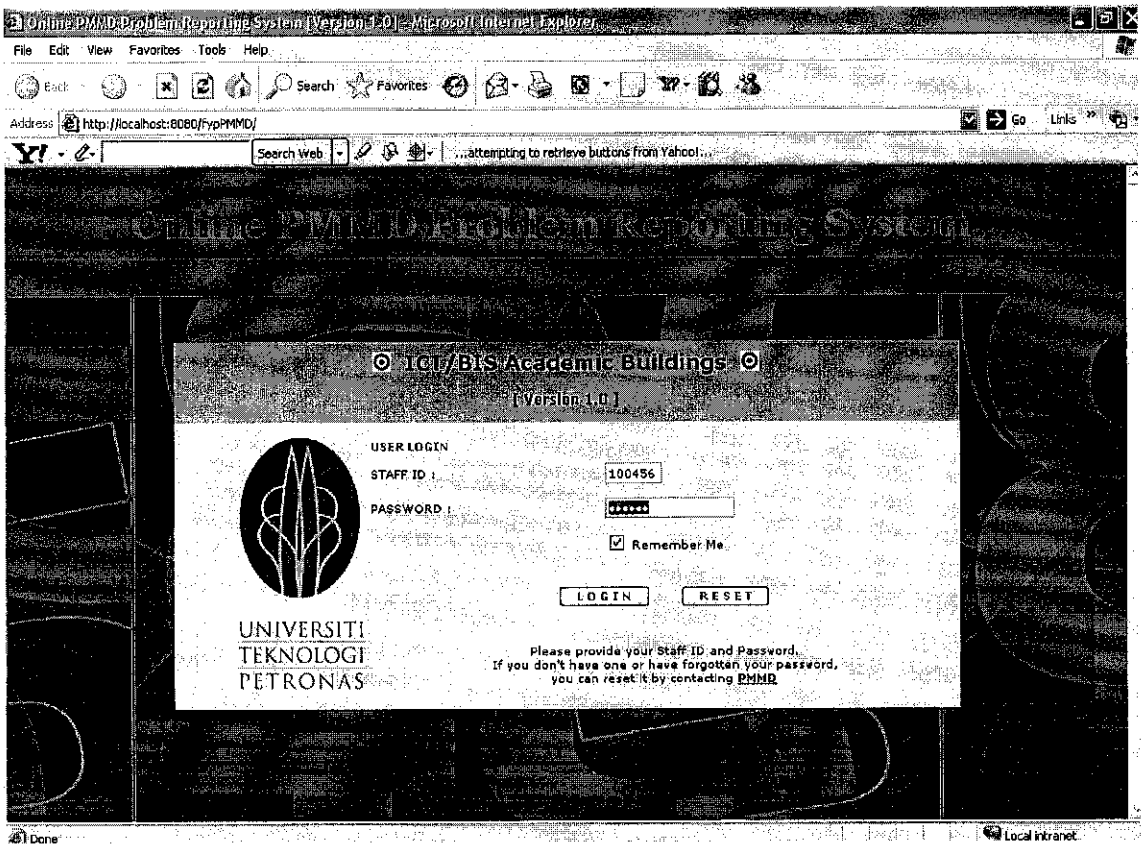


Figure 10: Online PMMD Problem Reporting System Login Page

The event is user logon triggered by user ID and password. The activity here is to find user record and verify password, then redirect user to Welcome Web page (Homepage).

Figure 11 shows Online PMMD Problem Reporting System homepage which is the default page once user successfully logged-in. At top right of the page, there are three links – UTP Link, User Tracker and Logout. Below of that, there are five main menus of the system:

- Home
- Report Form
- Report Analysis

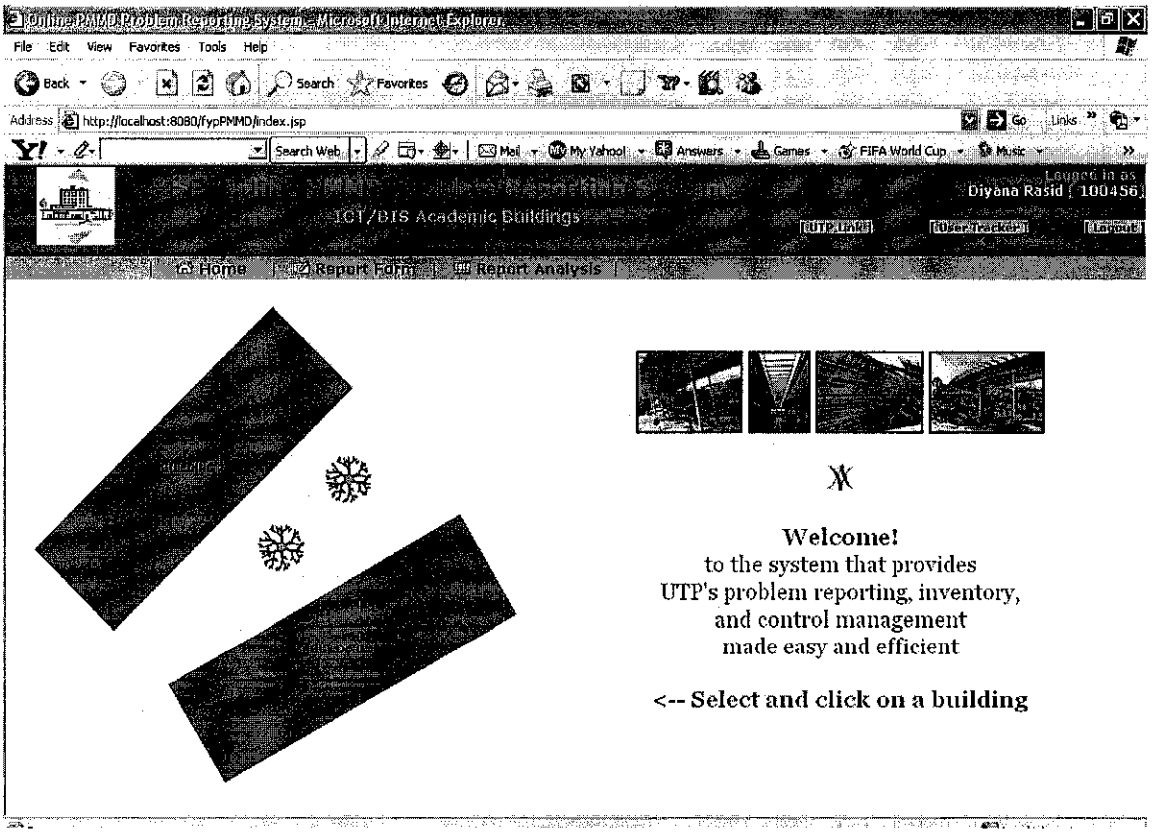


Figure 11: Online PMMD Problem Reporting System Homepage

When user clicks on 'Building 1' of the Homepage, user will be directed to this page:

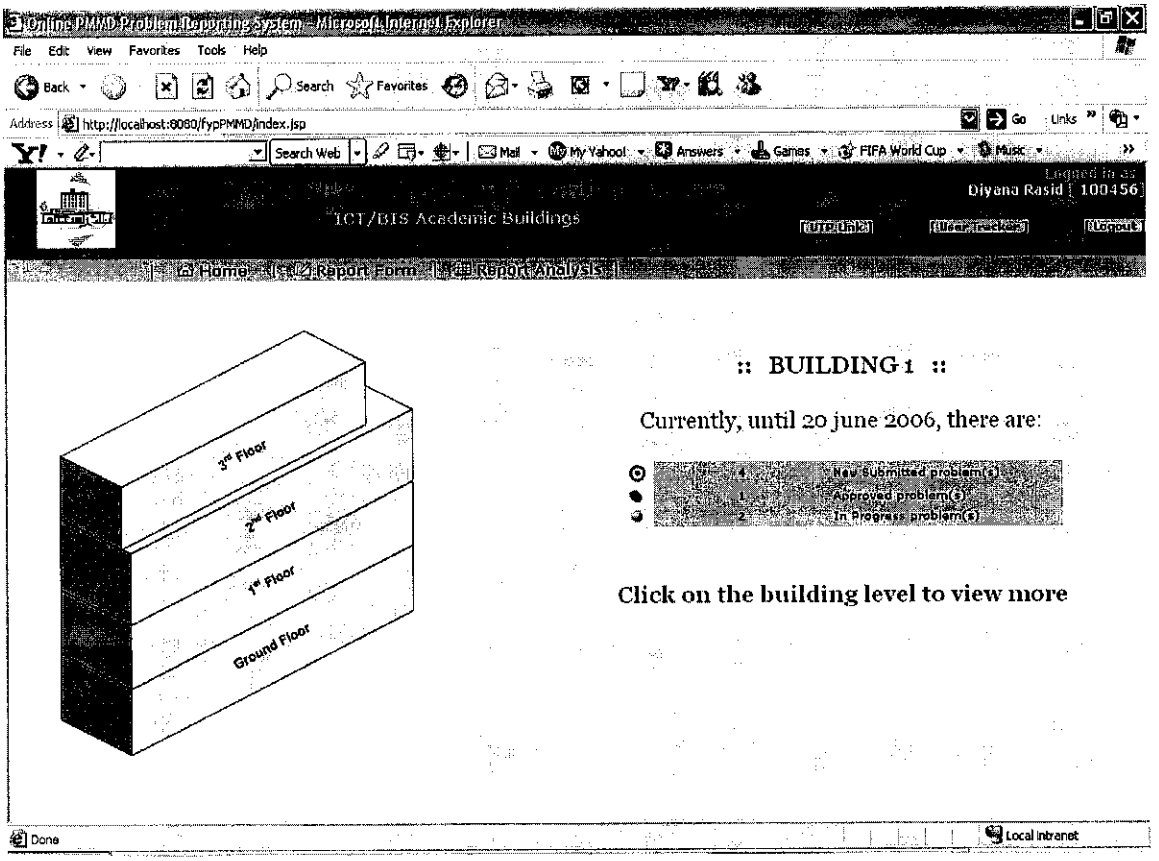


Figure 12: Building 1 Graphical Representation

When user click on selected building level (example: Ground Floor of Building 1), there are color-coded blinks for each submitted report (see Figure 13):

- Red dot – New submitted report
- Yellow dot – Approved report but no action taken yet
- Green dot – In Progress of solving the problem

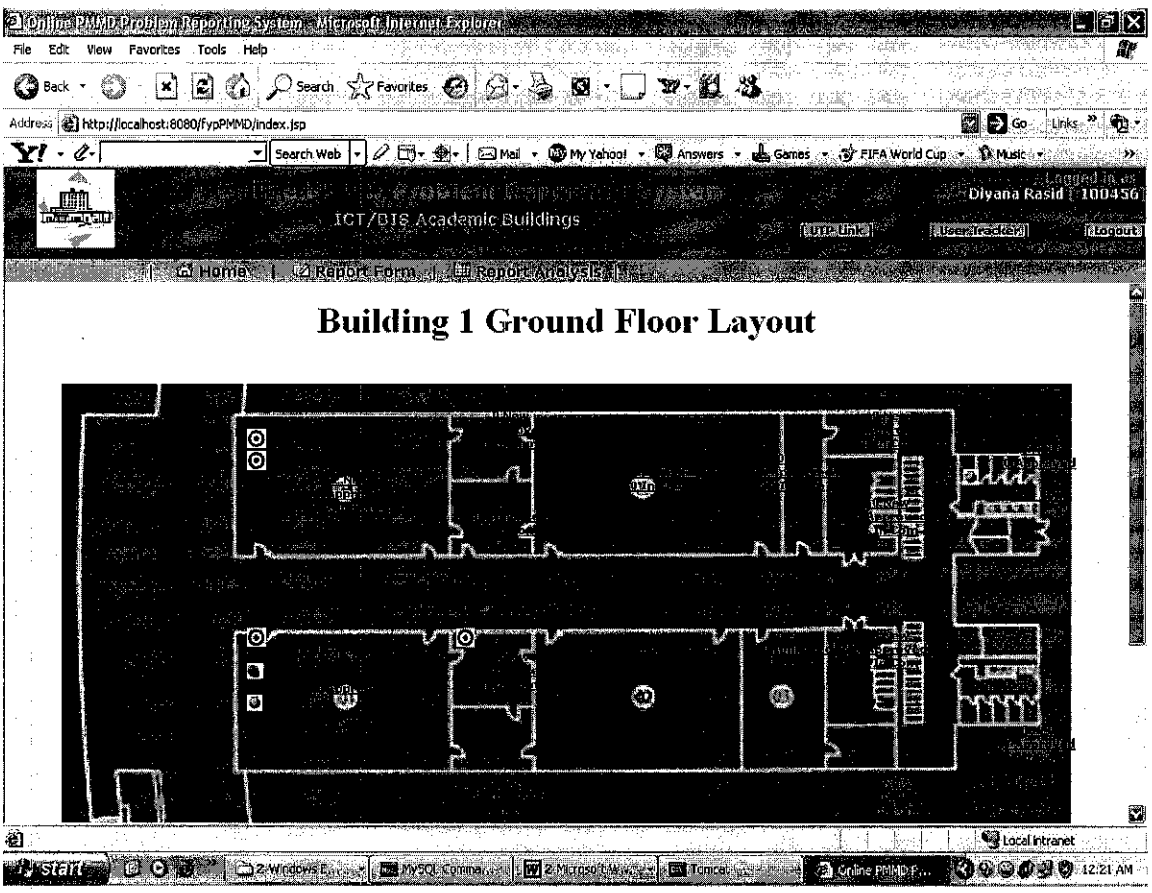


Figure 13: Ground Floor Building 1 Graphical Representation

Figure 14 shows the 'Report Form' section. The report form is different for each type of users. The three types of users are Approver, Administrator, and Submitter (common user).

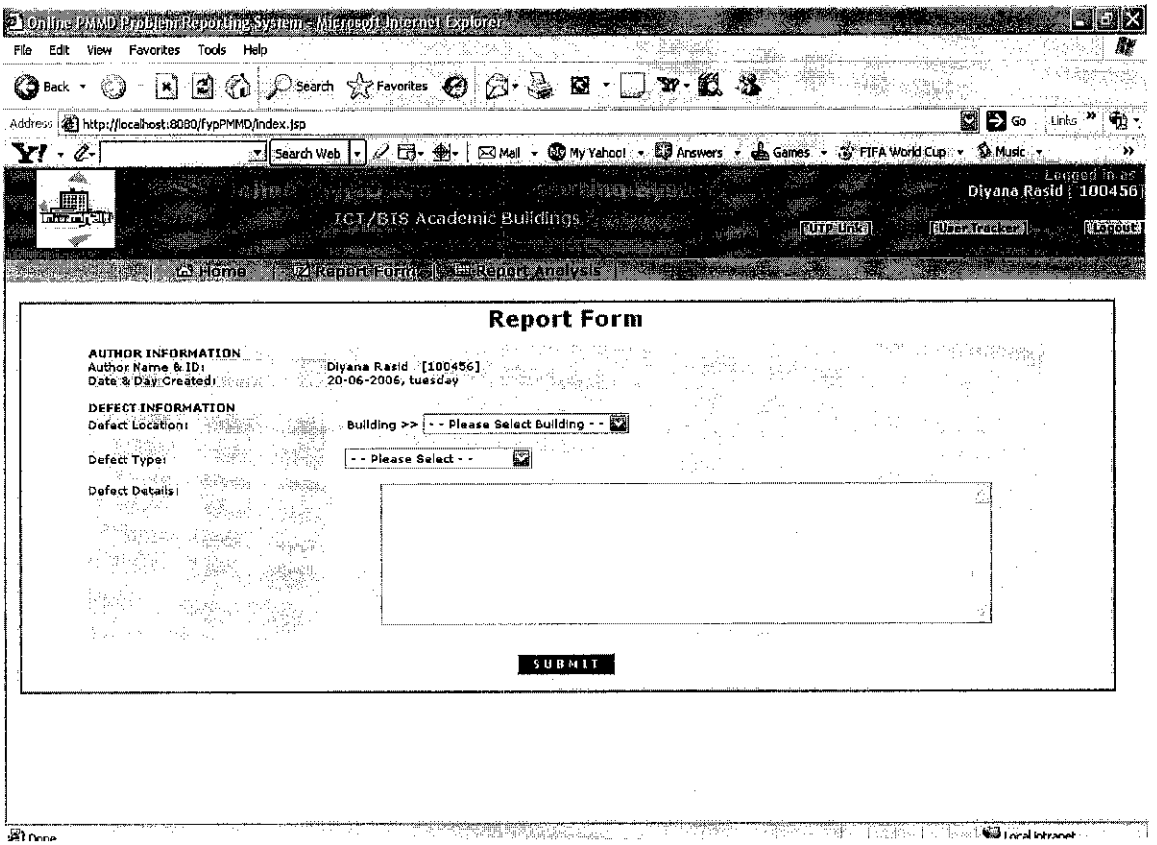


Figure 14: General View of Online PMMD Problem Reporting System 'Report Form'

Figure 15 below shows 'Report Form' for an Approver's view:

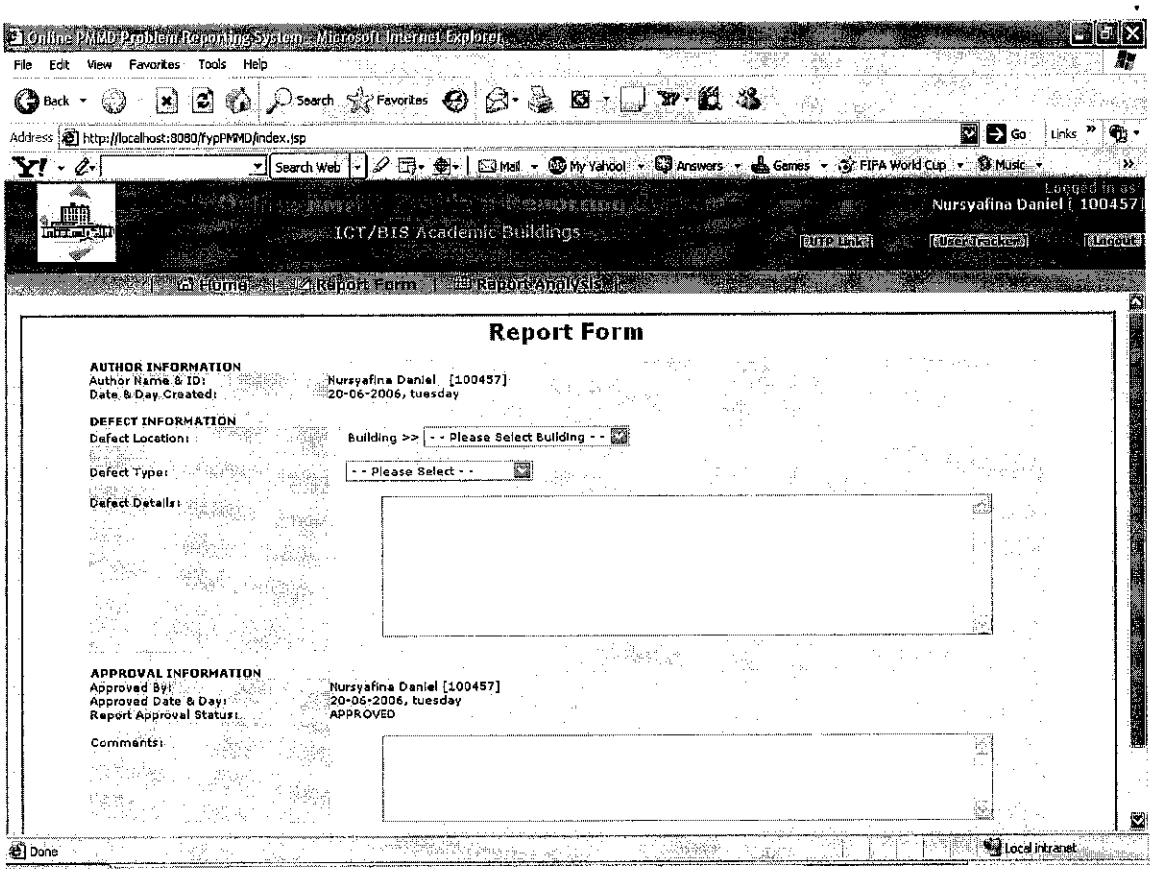


Figure 15: Approver's View of Online PMMD Problem Reporting System 'Report Form'

There are several options in 'Report Analysis' section where user can select from the dropdown menus. There are five selections:

- View by Problem Status
- View by Problem Location
- View by Problem Type
- View by Problem Submitter
- View Cancelled Problem

Figure 16 is an example: 'View by Problem Status'.

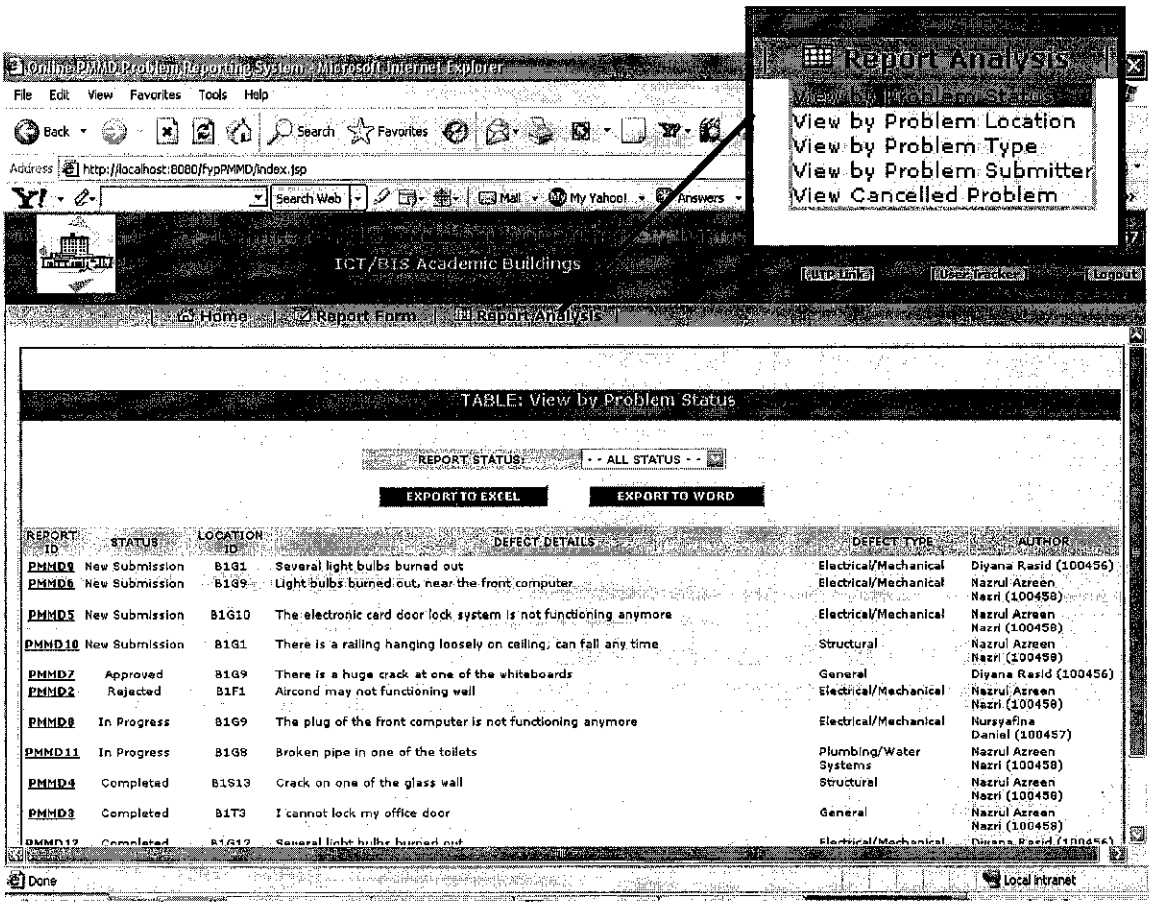


Figure 16: Report Analysis – View by Problem Status

To change the status of each problem report, an Approver or Administrator can click at the Report ID link in the table. When clicked, a new window will appear and shows the 'View Individual Report Info' (see Figure 17). Approver or Administrator can add new additional information in yellow coloured fields provided in the form.

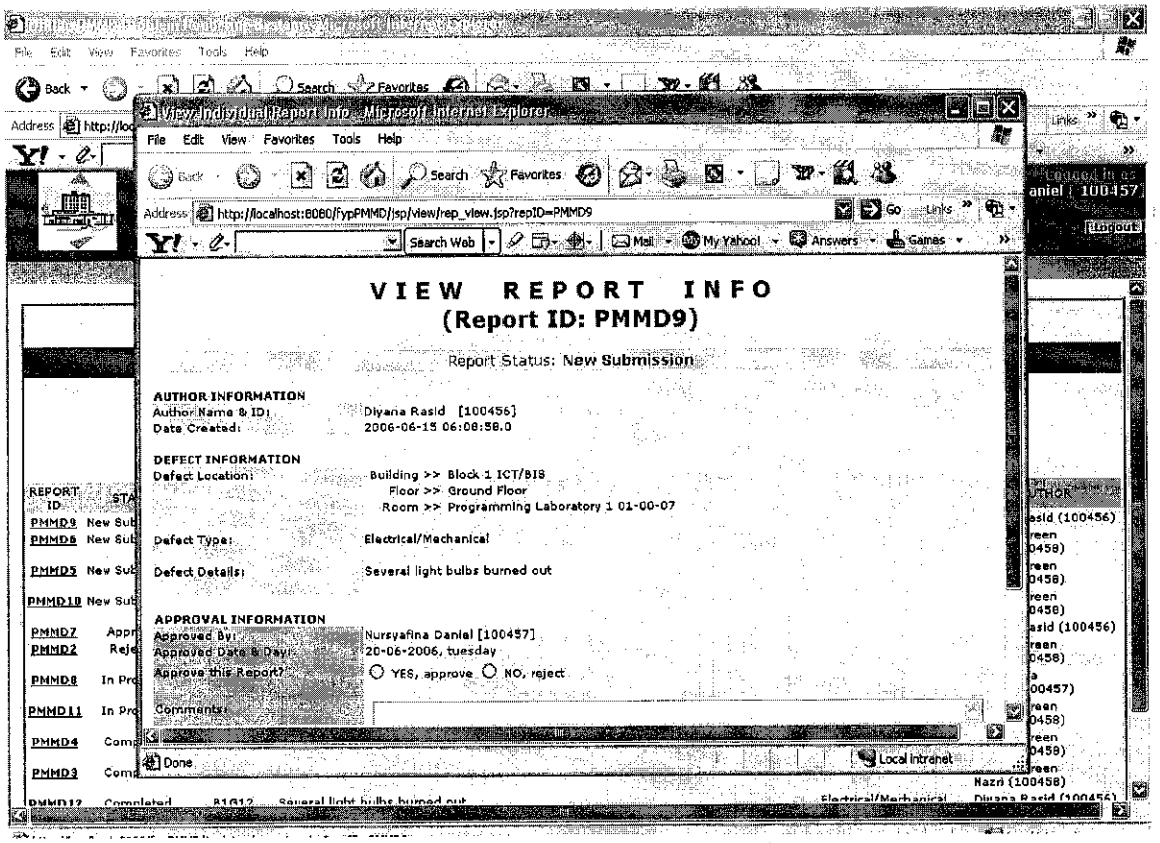


Figure 17: View Individual Report Page

Figure 18 below shows the User Tracker window that comes into view when user clicked the 'User Tracker' link. It displays the complete list of system's user history. The system is able to track user who logged-in and or modifications, and the access date.

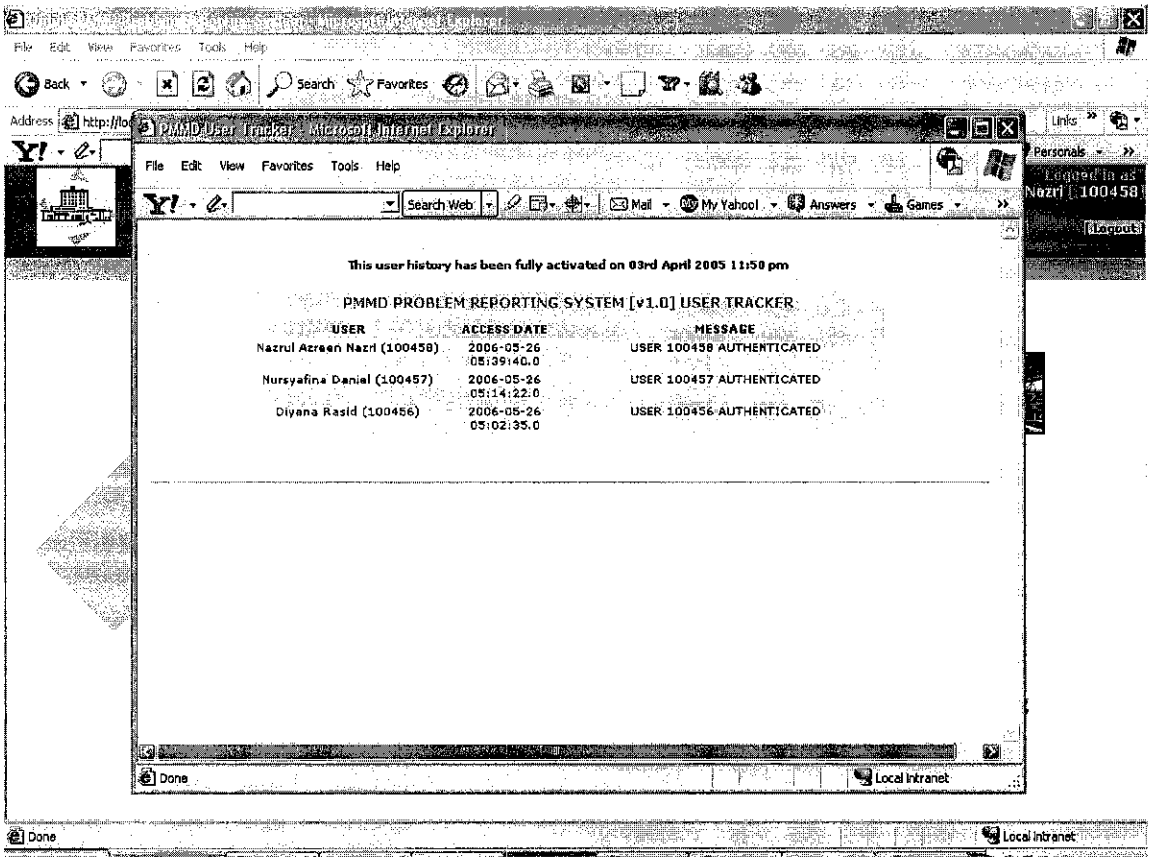


Figure 18: Online PMMD Problem Reporting System User Tracker

CHAPTER 5
CONCLUSION AND RECOMMENDATIONS

5.1 CONCLUSION

As a conclusion, the project has successfully met its objectives. The online system enables UTP staffs and administrators of PMMD to easily record and identify which area are in need of attention. It produces useful dynamic graphical problem reports and analysis conforming real situation/mapping of Building 1 and Building 2. The system also made reliable to remain dependable in the face of malice, error or mischance through its design and engineering. It is worth mentioning that the author develops the project using methods explained and within the time frame.

The design requirements of the project drive the author choice of technology. By implementing this project, UTP community specially the Property Management and Maintenance Department definitely will be able to increase the efficiency of problem reporting process flow and deliver higher quality of service.

5.2 RECOMMENDATIONS

There are a few recommendations that can be done to the system so that it can be expanded in the future to be more reliable and practical. The limitations of the system should be solved or improved.

5.2.1 Problem Report Email Notification

The function of email notification is not provided in the Online PMMD Problem Reporting System. For future purposes, this method of notification could be added as each UTP staff has an individual corporate email account. It would be feasible to integrate this Online PMMD Problem Reporting System with the current UTP Lotus Notes Email.

Email is a practical way to alert the status of problem report being submitted to responsible PMMD staff and problem submitter, whether the problem is in its completion phase or is completed. In case of cancellation of a problem report, then the system can also automatically send the notification individually to all associated people in the specific record.

5.22 Expansion of the System Scope

Since the project only focus on two of UTP new academic buildings, Building 1 and Building 2, there might be very comprehensive if the system can cover all buildings of the UTP academic area. Therefore, the PMMD would have a complete online system to handle its most important area. The expansion will definitely require a lot of time and effort and preferably developed by a team of several people.

5.23 Implementation of the Idea in Different Scope

Coming up with good idea is hard enough, but convincing others to do something with them is even harder. The author has developed the prototype of an implementation of the idea on using graphical site navigation and visual representations of data. The proof of this concept can be expanded to a different scope, for example, the UTP Residential Buildings.

When a different scope of the idea has been identified, there is a need to do some research on how similar scope went about it. The author probably not the first person to pitch something on the UTP Residential Building's scope, so it is better to go find out what other people did, and what kind of success they had. Learn from their mistakes.

REFERENCES

Eyetrack III Team. Recall of Information Presented in Text vs. Multimedia Format. Retrieved Nov 22, 2005, from the World Wide Web: *http://www.poynterextra.org/EYETRACK2004/*

Fleishman, G. Image Mapping. Retrieved Oct 14, 2005, from the World Wide Web: *http://glennf.com/pdfs/maps.pdf*

Gilbert, D. (2005). *The JFreeChart Class Library Installation Guide Version 1.0.0-rc1.* Object Refinery Limited.

Guang Yang. Programmable Graphics: The Next Killer Web Application. Retrieved Oct. 13, 2005, from the World Wide Web: *http://www.sunwesttek.com/wp_di.html*

Hoffman, B. (1999). Educational Technology 541 Multimedia Development. Retrieved Oct 13, 2005, from the World Wide Web: *http://edweb.sdsu.edu/courses/EDTEC541/*

IWIN U.S National Weather Service Website. Retrieved Dec 1, 2005, from the World Wide Web: *http://iwin.nws.noaa.gov/iwin/graphicsversion/rbigmain.html*

McGloughlin, S. (2001). *Multimedia Concepts and Practice, 5th Edition.* Prentice Hall International (UK) Limited.

McKnight, C.; Dillon, A.; Richardson, J. (1996). User-centered design of hypertext/hypermedia for education. In Kozma, R. B. (Ed.), Handbook of Research for Educational Communications and Technology, Section III: Hard Technologies: Media-

Related Research. pp. 622-633. IN: Association for Educational Communications and Technology.

Pressman, R. (2001). *Software Engineering A Practitioner's Approach.* McGraw-Hill International.

Puneet M. Sangal. Building 2-D Graphics Applications Using Java and SVG. Retrieved Oct 14, 2005, from the World Wide Web: *http://www.devx.com/webdev/Article/28708/0/page/3*

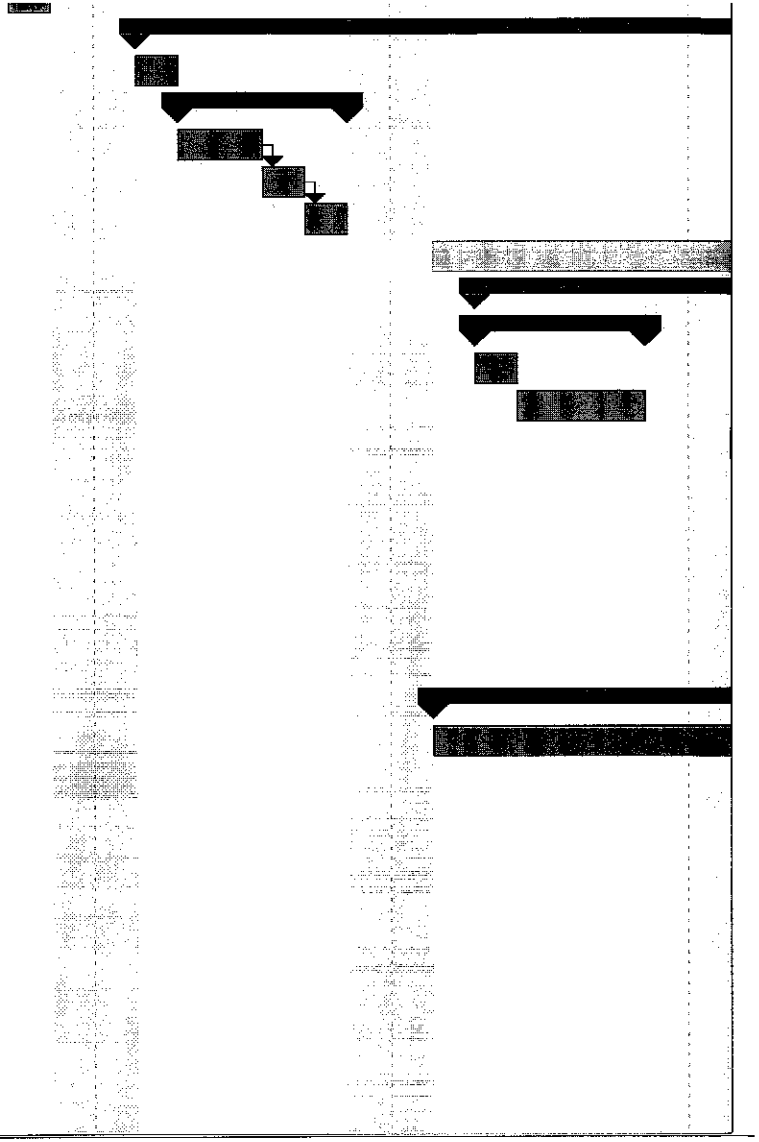
Sommerville, I. (2001). *Software Engineering, 6th Edition.* Essex: Pearson Education Limited.

Watson, J. Map sites chart online treasures. Retrieved Oct 14, 2005, from the World Wide Web: *http://www.vnunet.com/computing/analysis/2143335/map-sites-chart-online*

APPENDIX

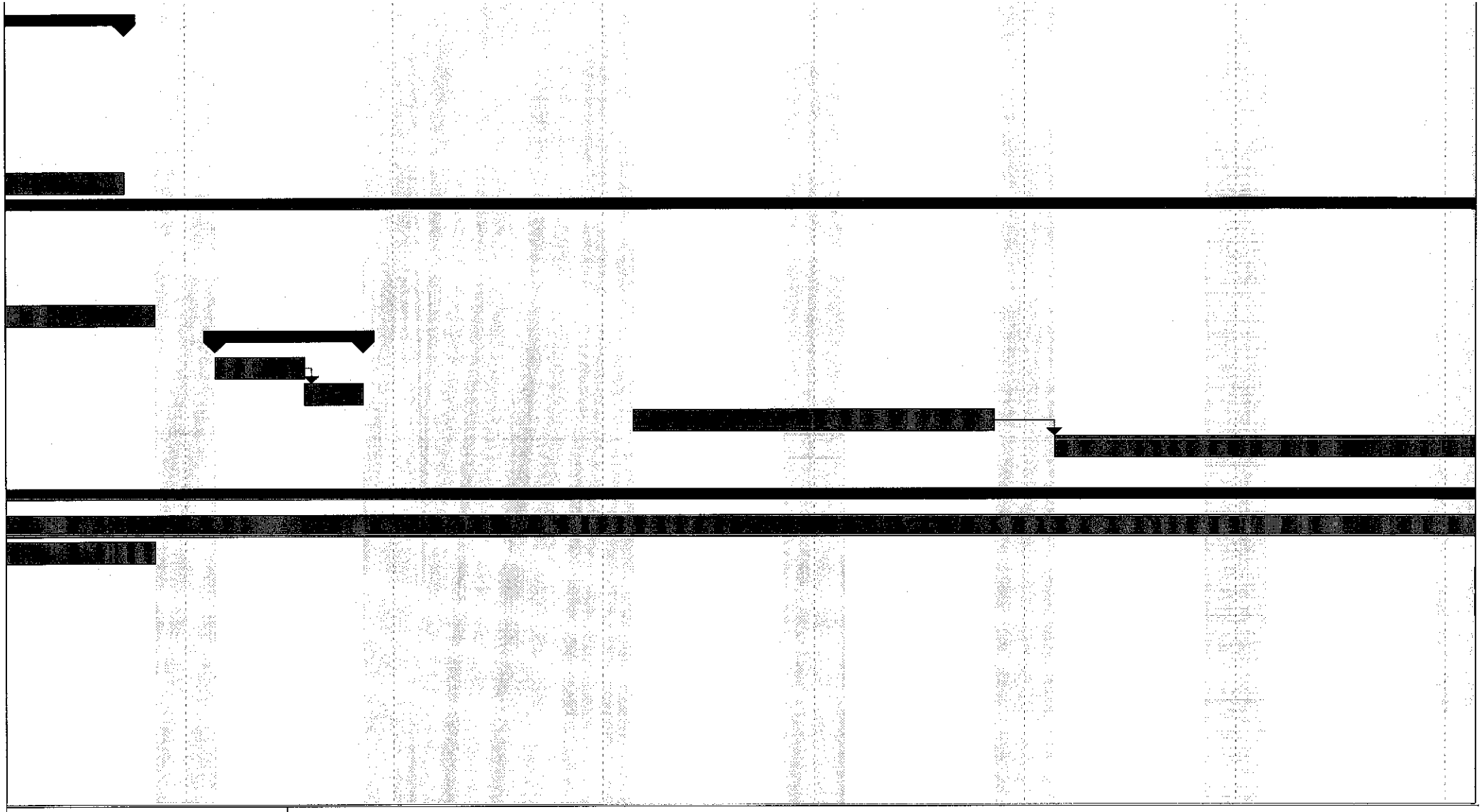
APPENDIX I: PROJECT SCHEDULE

4		**2. Planning**	**14 days**	**Mon 10/3/05**	**Thu 10/20/05**
5		2.1 Project kickoff meeting	1 day	Mon 10/3/05	Mon 10/3/05
6		**2.2 Prepare schedule**	**4 days**	**Tue 10/4/05**	**Fri 10/7/05**
7		2.2.1 Determine task resources, duratior	2 days	Tue 10/4/05	Wed 10/5/05
8		2.2.2 Create draft Gantt chart	1 day	Thu 10/6/05	Thu 10/6/05
9		2.2.3 Review & finalize Gantt chart	1 day	Fri 10/7/05	Fri 10/7/05
10		2.3 Preliminary research work	9 days	Mon 10/10/05	Thu 10/20/05
11		**3. Executing**	**127 days**	**Tue 10/11/05**	**Wed 4/12/06**
12		**3.1 Identifying problems, opportunities & (**	**4 days**	**Tue 10/11/05**	**Fri 10/14/05**
13		3.1.1 Interview PM&M personnel	1 day	Tue 10/11/05	Tue 10/11/05
14		3.1.2 Survey problem reporting flow	3 days	Wed 10/12/05	Fri 10/14/05
15		3.2 Determining information requirements	5 days	Mon 10/17/05	Fri 10/21/05
16		**3.3 Analyzing system needs**	**5 days**	**Mon 10/24/05**	**Fri 10/28/05**
17		3.3.1 Tools and equipments	3 days	Mon 10/24/05	Wed 10/26/05
18		3.3.2 Site setup	2 days	Thu 10/27/05	Fri 10/28/05
19		3.4 Designing recommended system	10 days	Mon 11/7/05	Fri 11/18/05
20		3.5 Developing and documenting system	100 days	Mon 11/21/05	Fri 4/7/06
21		3.6 Testing the system	3 days	Mon 4/10/06	Wed 4/12/06
22		**4. Controlling**	**135 days**	**Mon 10/10/05**	**Fri 4/21/06**
23		4.1 Weekly reports	135 days	Mon 10/10/05	Fri 4/21/06
24		4.2 Submission of Preliminary Report	5 days	Mon 10/17/05	Fri 10/21/05
25		4.3 First Semester Oral Presentation	5 days	Mon 12/5/05	Fri 12/9/05
26		4.4 Submission of Interim Report	5 days	Mon 12/5/05	Fri 12/9/05
27		4.5 Submission of Progress Report	5 days	Mon 1/23/06	Fri 1/27/06
28		4.6 Exhibition	5 days	Mon 4/3/06	Fri 4/7/06
29		4.7 Submission of Final Draft	5 days	Mon 4/10/06	Fri 4/14/06
30		**5. Closing**	**12 days**	**Thu 4/13/06**	**Fri 4/28/06**
31		5.1 Preparation for Final Oral Presentation	2 days	Thu 4/13/06	Fri 4/14/06
32		5.2 Final Oral Presentation	5 days	Mon 4/17/06	Fri 4/21/06
33		5.3 Preparation & submission of Project Diss	5 days	Mon 4/24/06	Fri 4/28/06



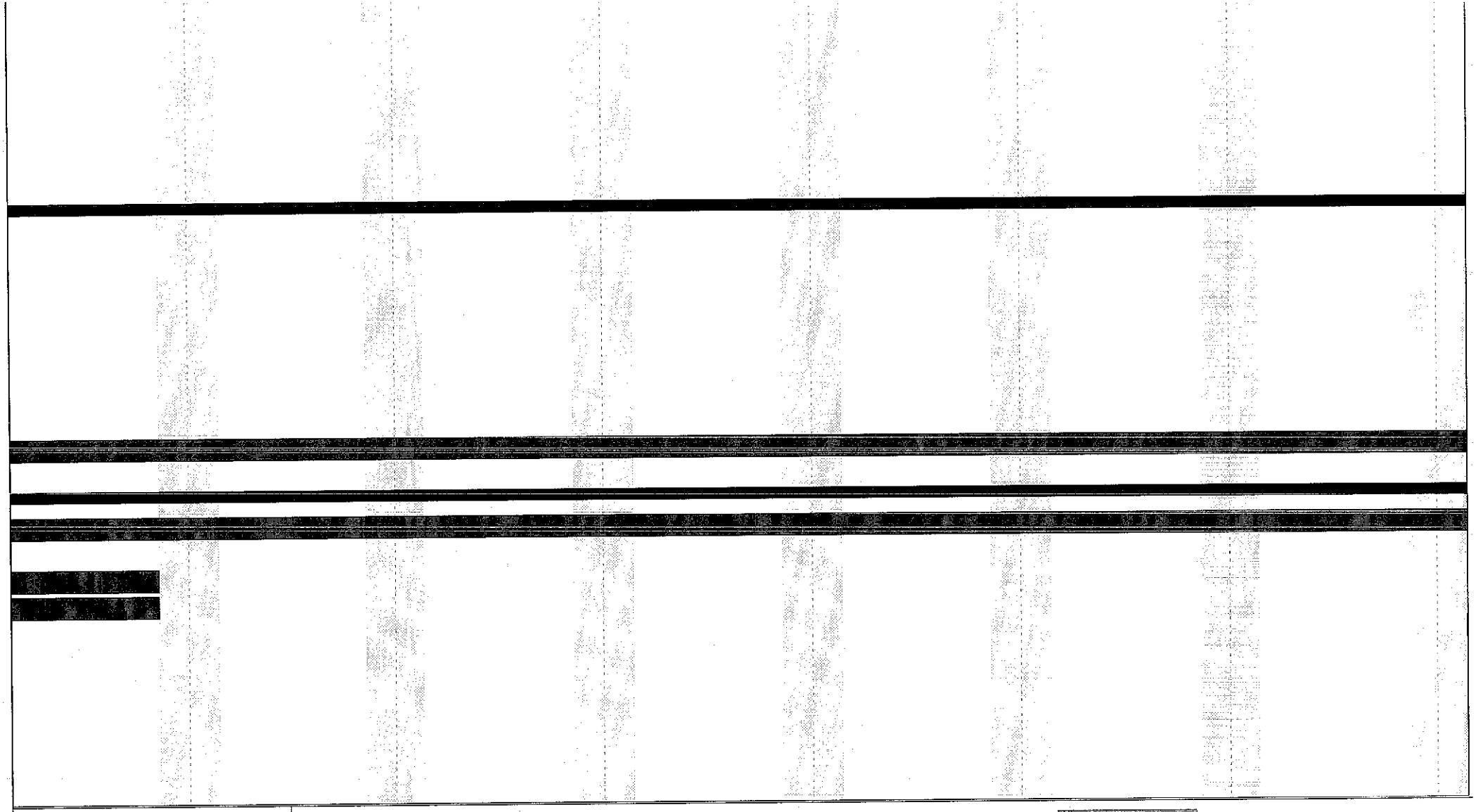
Project: FYP Project Schedule
Date: Tue 6/20/06

Task
Split
Progress
Milestone
Summary
Project Summary
External Tasks
External Milestone
Deadline



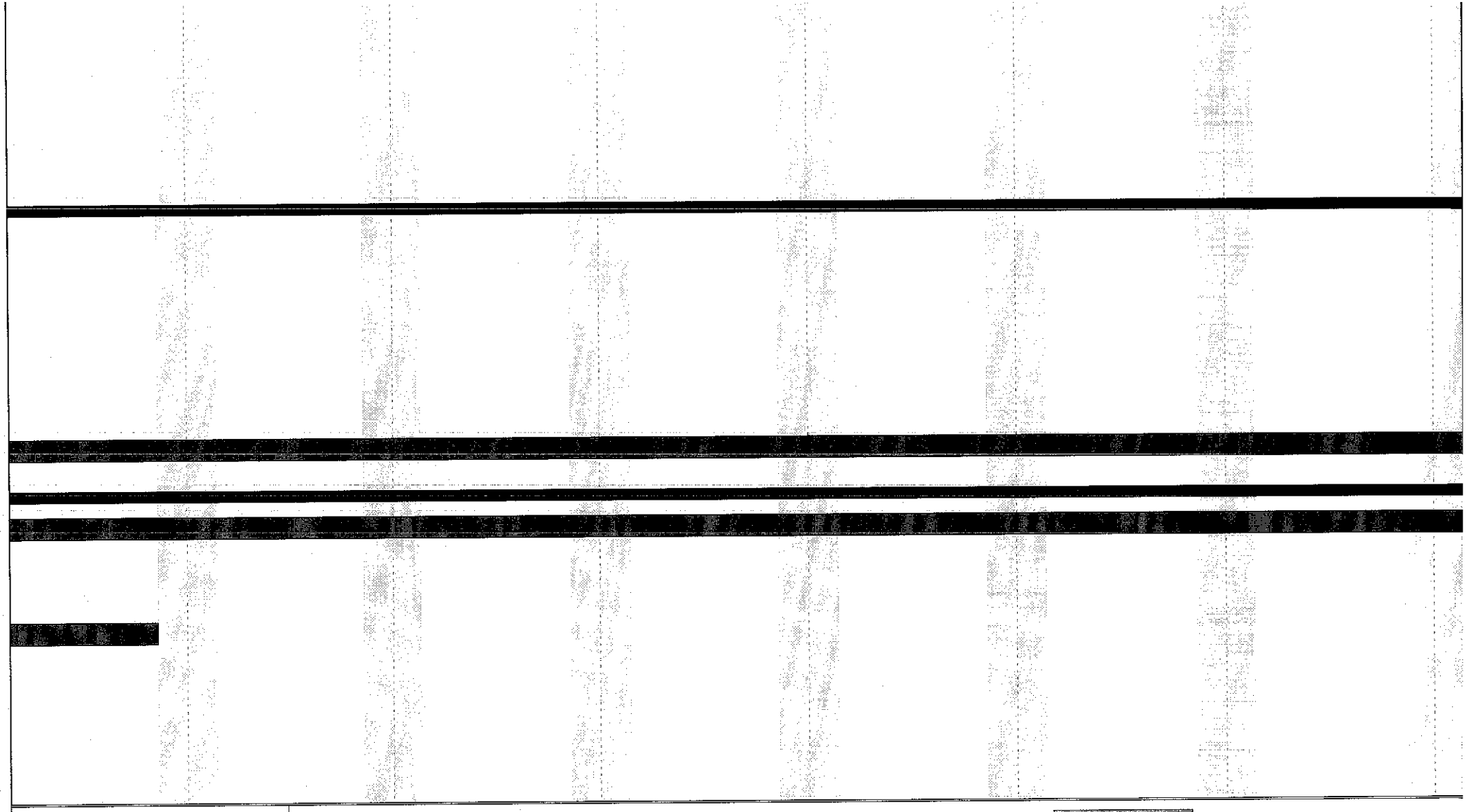
Project: FYP Project Schedule
Date: Tue 6/20/06

Task		Milestone		External Tasks	
Split		Summary		External Milestone	
Progress		Project Summary		Deadline	



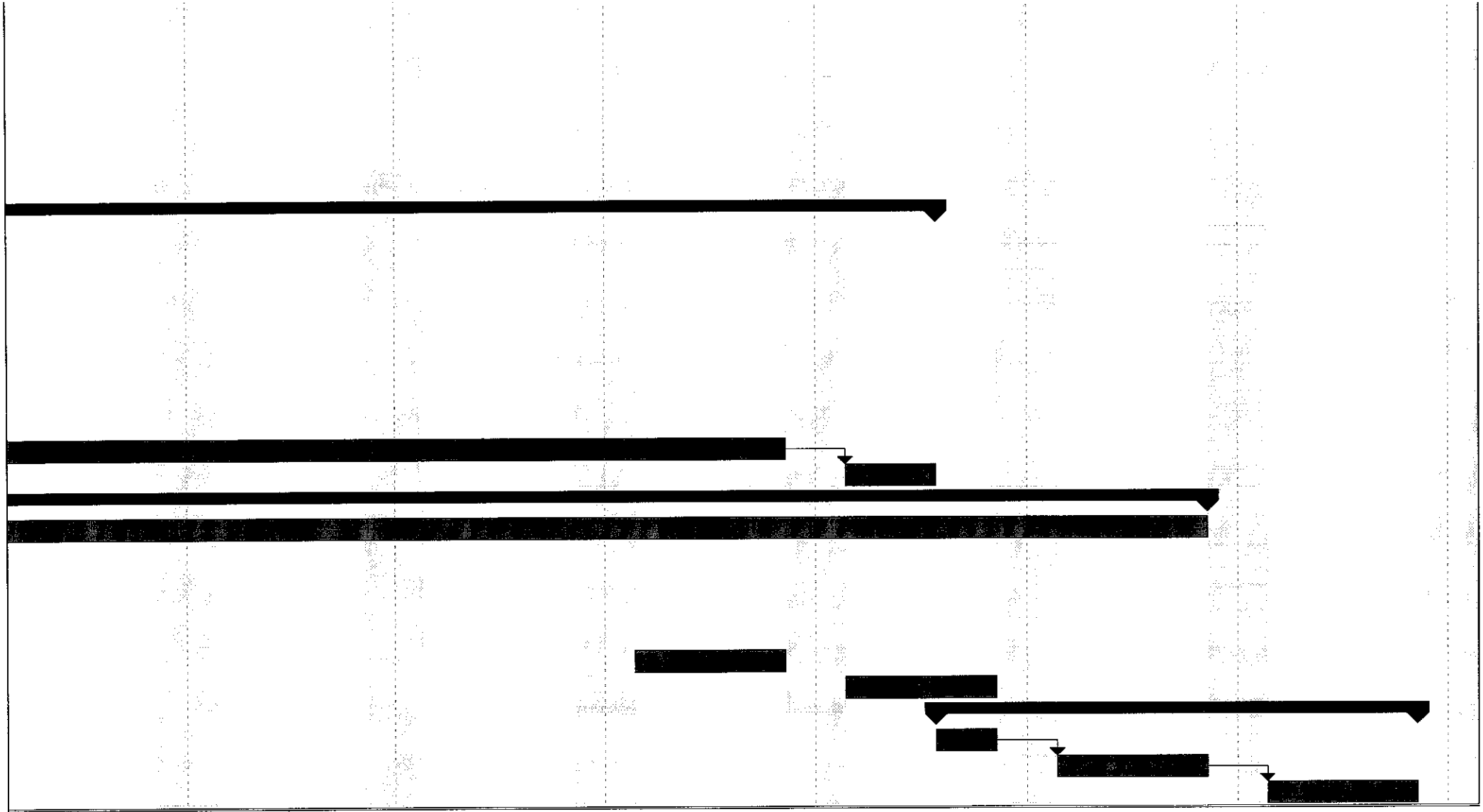
Project: FYP Project Schedule
Date: Tue 6/20/06

Task
Split
Progress
Milestone
Summary
Project Summary
External Tasks
External Milestone
Deadline



Project: FYP Project Schedule
Date: Tue 6/20/06

Task
Split
Progress
Milestone
Summary
Project Summary
External Tasks
External Milestone
Deadline



Project: FYP Project Schedule
Date: Tue 6/20/06

Task
Split
Progress
Milestone
Summary
Project Summary
External Tasks
External Milestone
Deadline

APPENDIX II: INTERVIEW DETAILS

INTERVIEW DETAILS

Mode of Interview: Meeting

Date of Interview: October 11, 2005

Time of Interview: 10.45 am – 11.30 am

Interviewee Information

Name: Mr. Hazizi B. Laili

Designation: Executive (Civil Engineering)

Organization: Universiti Teknologi PETRONAS

Department: Property Management and Maintenance

List of Interview Questions

1. What are the responsibilities of the Property Management and Maintenance Unit? What are the differences between property management and property maintenance?
2. What are your individual work responsibilities in this unit?
3. What are the broad ranges of problem types crossing all academic building environments?
4. Could you explain to me the step by step procedures of a problem reporting?

Step 1: Entry	Step 2: Event Received	Step 3: Investigation	Step 4: Action	Step 5: Feedback

5. As system administrator, what would you want to be able to do in the system?
6. How would you want the system to create internally defined communication rules?
7. Can you give me the floor plan for the ICT/BIS academic buildings?

APPENDIX III: PMMD Work Request/Defect Form

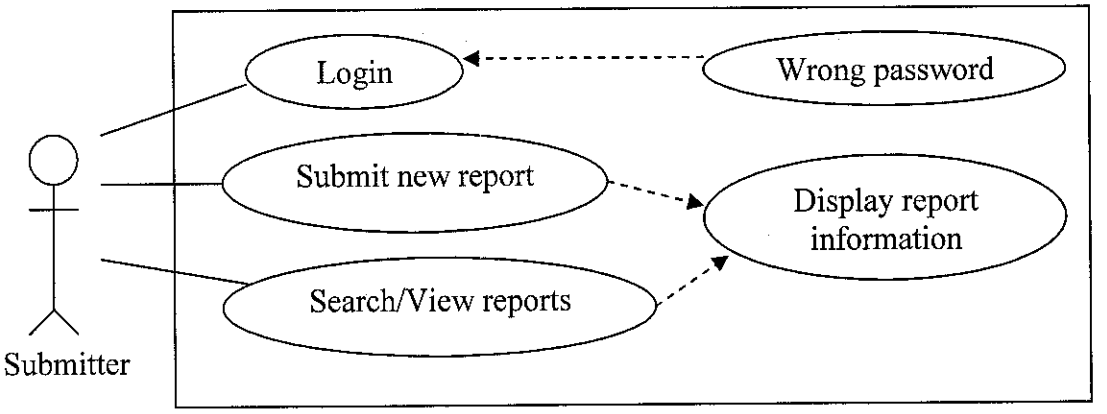
PMMD WORK REQUEST/ DEFECT FORM

UNIVERSITI TEKNOLOGI PETRONAS

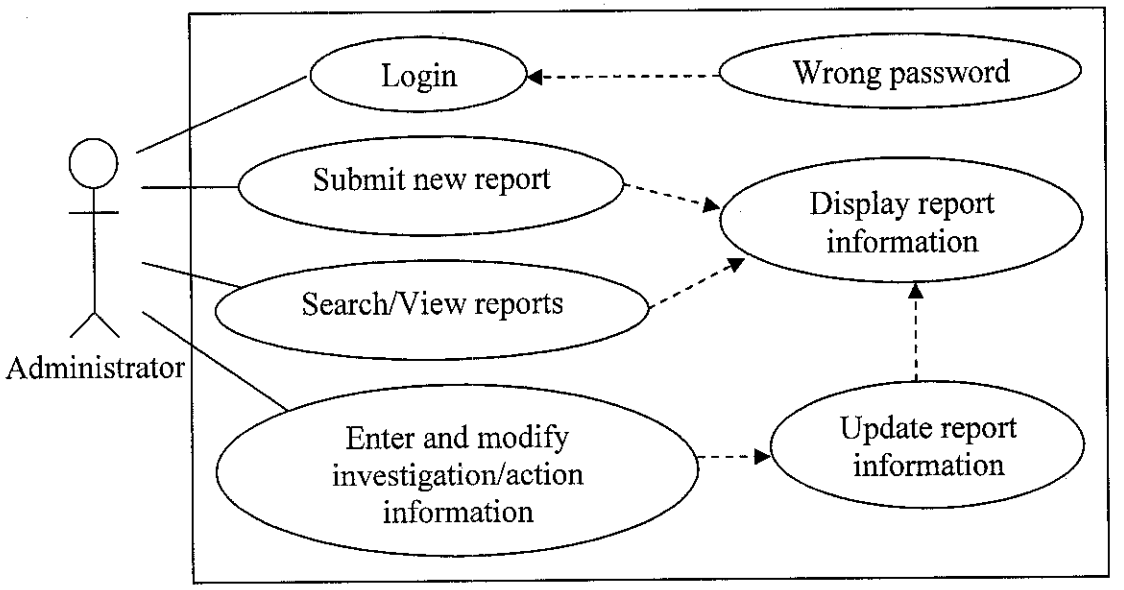
DATE:							
APPOINTED CONTRACTOR							
LOCATION							
PRIORITY LEVEL (TO BE FIILED BY PMMD ENGINEER)	URGENT REPAIR WORKS		NORMALREPAIR WORKS		PLAN WORKS		
JUSTIFICATION							
	To be fill by User/ Assitant Engineer/ Executive/ Engineer		To be fill by Appointed Contractor			To be fill by the User	
NO	BUILDING/ ROOM	REQUEST/ DEFECT IN DETAILS	ACTION TAKEN/ REMARKS	DATE	REPLACED PARTS (IF ANY)	DATE	VERIFICATION BY

	Prepared By	Endorsed By	Approved By
	Date	Date	Date

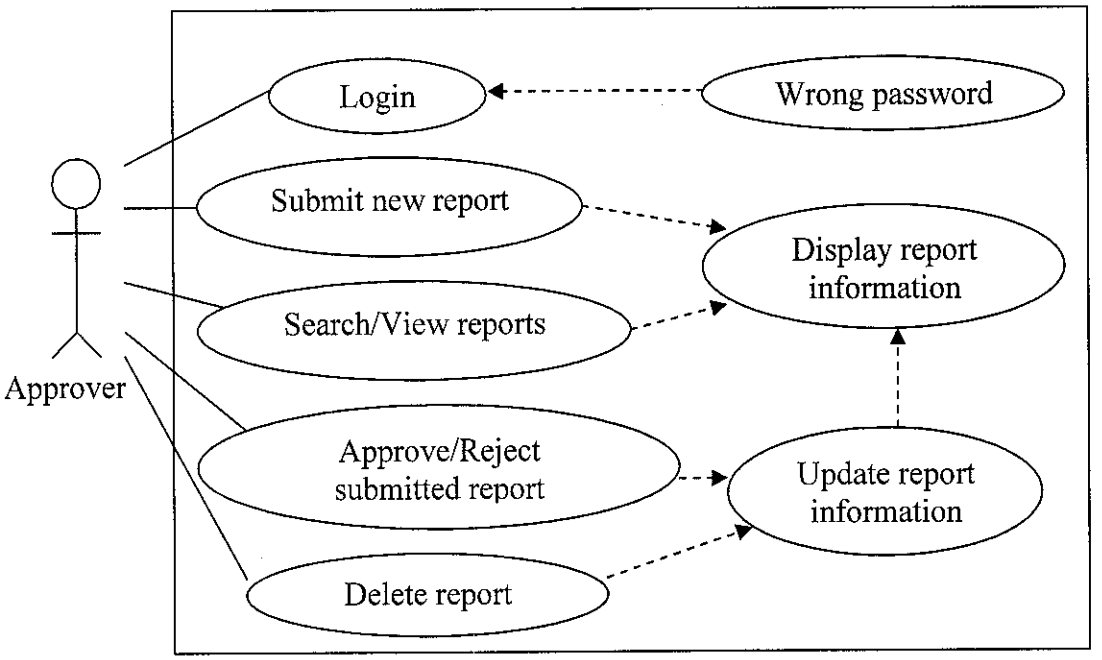
APPENDIX IV: DESIGN PHASE



Use Case Diagram for Submitter



Use Case Diagram for Administrator



Use Case Diagram for Approver

APPENDIX V: QUESTIONNAIRE

Project Title: Implementation of Dynamic Graphics in Problem Reporting System for ICT/BIS Academic Buildings

Questionnaire produced by Diyana Rasid, Information Technology (IT) Final Year Student of Universiti Teknologi PETRONAS, Tronoh, Perak regarding the Final Year Project.

Abstract:

This is a survey intended for research purposes. This questionnaire is used to gather information, opinion and feedback for the research work. It is hoped that with the help of the result gathered from this survey, the author will come out with an effective online PMMD Problem Reporting System.

Kindly please answer ALL questions.

1. How do you usually report a property problem at your area?
   - ☐ Go to liable department for reporting the problem
   - ☐ Fix the problem without reporting
   - ☐ Others: _____
   - ☐ I do not care to report and fix the problem

2. Have you ever use any online based system for property problem reporting?
   - ☐ Yes
   - ☐ No

3. How do you find an online based system would help for the property problem reporting process?
   - ☐ Unsatisfactory
   - ☐ Below Average
   - ☐ Average
   - ☐ Good
   - ☐ Excellent

4. How do you prefer information of reported problem being displayed in online based system?

   □ Text only

   □ Graphic only

   □ Combination of both graphic and text, but minimal text

   □ Combination of both graphic and text, but minimal graphic

5. Do you have computer with internet facilities?

   □ Yes

   □ No

6. What operating system do you generally use?

   □ Windows

   □ Mac OS

   □ Unix

7. Which web browser do you use?

   □ Netscape Navigator/Communicator

   □ Microsoft Internet Explorer

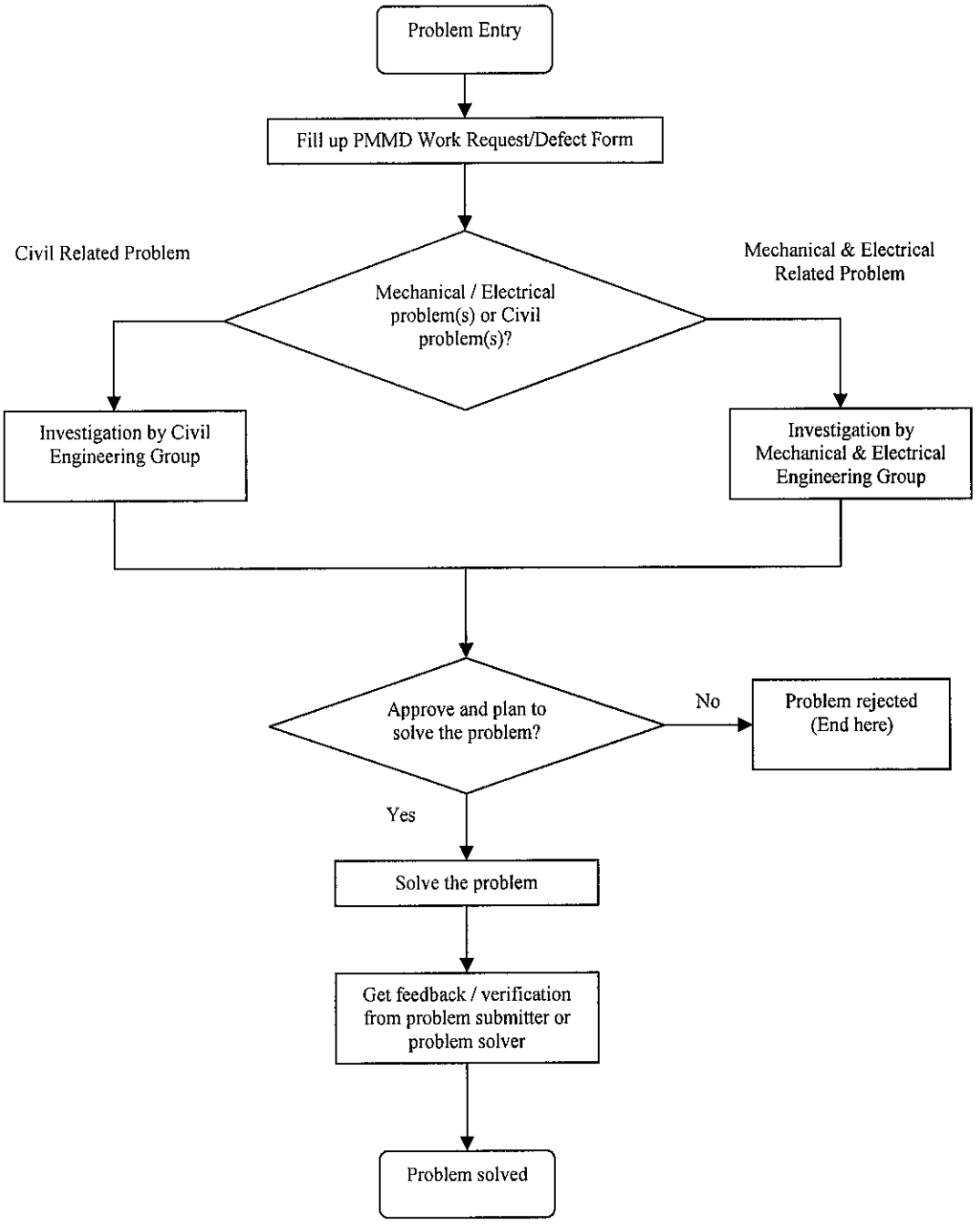
   □ Other

Thank you for your cooperation.

APPENDIX VI:

Current PMMD Problem Report Management Process Flowchart

Current PMMD Problem Report Management Process Flowchart



APPENDIX VII: FUNCTIONAL TESTING RESULTS

Table 1 Test Result of Login Module

Component	Expected Test Result	Actual Test Result
Login Button	▪ To ensure user access identification (Staff ID and Password) text fields are filled. ▪ To verify user access identification. ▪ To permit user access.	▪ Successfully notified unfilled text field through pop-up window. ▪ Successfully verified user access identification. ▪ Successfully permitted user access to the system.
Reset Button	▪ To empty the Staff ID and Password text fields.	▪ Successfully empty the Staff ID and Password text fields.

Table 2 Test Result of Main Module

Component	Expected Test Result	Actual Test Result
Home button	▪ To return to system's main page where Building 1 and Building 2 links resided.	▪ Successfully return to system's main page where Building 1 and Building 2 links resided.
Report button	▪ To add new problem report using the 'New Report Form'. ▪ To ensure the new problem information will be validated and stored once user clicked the 'Submit' button.	▪ Successfully added the new problem report. ▪ Successfully validated and stored new problem information once user clicked the 'Submit' button.

Report Analysis button	▪ To view all records sorted by problem status. ▪ To view all records sorted by problem location. ▪ To view all records sorted by problem type. ▪ To view all records sorted by problem submitter. ▪ To view all cancelled problems.	▪ Successfully viewed all records sorted by problem status. ▪ Successfully viewed all records sorted by problem location. ▪ Successfully viewed all records sorted by problem type. ▪ Successfully viewed all records sorted by problem submitter. ▪ Successfully viewed all cancelled problems.
Advanced Search button	▪ To search for specific problem record(s) according to the conditions selected/specified by users.	▪ Successfully identified and list problem record(s) according to the conditions selected/specified.
Help button	▪ To provide system's User Manual. ▪ To view general information about the online system.	▪ Successfully provided system's User Manual. ▪ To view general information about the online system.
UTP link	▪ To open a new window redirected to the official UTP website. (http://www.utp.edu.my/)	▪ Successfully opened a new window redirected to the official UTP website.
User Tracker link	▪ To open a new window to view the complete system's user history.	▪ Successfully opened a new window to view the complete user history.

Building 1 image link	▪ To display the submitted problems for Building 1 graphically.	▪ Successfully displayed the submitted problems for Building 1 graphically.
Building 2 image link	▪ To display the submitted problems for Building 2 graphically.	▪ Successfully displayed submitted problems for Building 2 graphically.

Table 3 Test Result of Logout Module

Component	Expected Test Result	Actual Test Result
Logout link	▪ To end the user session ▪ To ensure the user is logout from the system	▪ Successfully ended the user login session ▪ Successfully ensuring user is logout from the system

APPENDIX VIII: INTEGRATION TESTING RESULTS

Table 4 Test Result of Integration Testing

Module/Component	Expected Test Result	Actual Test Result
Integration between Subsystems	▪ To ensure the integration between subsystems is successful without any flaw or error. ▪ To ensure each subsystem is well-functioning.	▪ Successfully integrated. ▪ Each subsystem is well-functioning.
Integrated Subsystems as a Whole System	▪ To ensure the system is well-functioning. ▪ To guarantee there is no flaw or error after integrated.	▪ The system is functioning successfully. ▪ Successfully guaranteed there is no flaw or error after the integration.
Linkage 1. Subsystems / Module Links (the main menus of the interface)	▪ To ensure the main menu links (which represent each subsystem/module) is functioning and linking. ▪ To ensure user can go (jump) directly to another subsystem while navigating other subsystem's page.	▪ The main menu is well-functioning and linked together. ▪ The main menu is successfully allowing the user to go to another subsystem, even though they are currently navigating other subsystem's page.

2. Pages Links – link to the following page.	▪ To ensure any links (which have linking to the following page) is successfully linked. ▪ To ensure no lost of variable's value or data after integration.	▪ Each links are successfully linked to their following page. ▪ Successfully ensuring that there is no lost of data after integration.

Table 5 Test Result of Integration between the System and Outside Components

Module/Component	Expected Test Result	Actual Test Result
System and Database	▪ To ensure the connection between the system and database is successful.	▪ The connection is successful.