

THESIS^ APPROVAL STATUS FORM

JUDUL: e-LECTURER ASSISTANT SYSTEM

SESI PENGAJIAN: 2004/2005

Saya LAW BOON KIAT
(HURUF BESAR)

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Kolej Universiti Teknikal Kebangsaan Malaysia.
2. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. ** Sila tandakan (/)

<u> </u>	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA
	RASMI	1972)
<u> </u>	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
<u> / </u>	TIDAK TERHAD	


(TANDATANGAN PENULIS)

Alamat tetap : 2, JLN BERSATU, TMN BERTAYA,
85010 BULOH KASAP, SEGAMAT, JOHOR

Tarikh : 20/10/2004


(TANDATANGAN PENYELIA)

PN NORASHIKIN AHMAD
Nama Penyelia

Tarikh : 20/10/2004

CATATAN: ** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.
^ Tesis dimaksudkan sebagai Laporan Projek Sarjana Muda (PSM)

e-LECTURER ASSISTANT SYSTEM

LAW BOON KIAT

This report is submitted in partial fulfillment of the requirements for the Bachelor of
Information and Communication Technology (Software Development)

**FACULTY OF TECHNOLOGY AND COMMUNICATION TECHNOLOGY
KOLEJ UNIVERSITI TEKNIKAL KEBANGSAAN MALAYSIA
2004**

DEDICATION

I am as ever, especially indebted to my parents, LAW SENG GUAN and SOH AH
TUAN for their love and support throughout my life...

ACKNOWLEDGEMENT

First and foremost, I would like to thank my thesis supervisor, Puan Norashikin Ahmad who gave me many guidelines, advises, and suggestion in helping me to complete the project. Without helping from the supervisor, the project can not be completed in the scheduled time.

My deepest gratitude and appreciation is to my course mate named Yeow Chen Lee. He has helping me a lot during the development of the system. He has shown me guidance, kindness, patience, care, and friendship throughout my project. Besides, he also provides many ideas that helping me in develop more flexible and efficiency system.

Besides that, I would like to thank others people, such as my family, friends and all the lecturers in Faculty Information Technology and Communication which are directly or indirectly related to my thesis. My hope is for them to be happy and enjoy themselves in their daily life.

Lastly, I want to apologize to everyone if there got any mistake that I made during development of the system. Hopefully enjoy reading this report. Thank you so much.

ABSTRAK

Sistem e-Lecturer Assistant akan dibangunkan pada akhir tesis ini. Tujuan utama sistem ini adalah untuk membantu pensyarah dalam memudahkan kerja harian mereka seperti mengira peratusan kedatangan siswa and siswi, mengira markah, menyediakan kertas peperiksaan, menyemak balik soalan peperiksaan tahun lepas, mengingatkan pensyarah tentang mesyuarat dan perjanjian, menghantar file antara pensyarah dan perbualan antara pensyarah. Sistem ini akan menggabungkan semua fungsi dalam satu aturcara sahaja. Jadi, sistem ini sangat sesuai bagi kegunaan pensyarah terutamanya mereka yang mengendalikan pelbagai kerja harian. Sebelum ini, kebanyakan kerja harian akan dikendalikan secara manual dan membuang masa pensyarah. Selain itu, pensyarah perlu menggunakan pelbagai aplikasi seperti Microsoft Word atau Excel untuk menyiapkan kerja mereka. Oleh itu, mereka perlu membazir banyak masa bagi mempelajari cara menggunakan aplikasi untuk membantu mereka dalam melaksanakan kerja harian. Kajian telah dibuat untuk mengumpul data. Selepas itu, skop projek dan keperluan pengguna boleh diketahui. Penggunaan metodologi adalah untuk menghasilkan kualiti produk model air terjun telah dipilih sebagai satu metodologi untuk membangunkan projek ini. Projek ini akan dilaksanakan berdasarkan kepada fasa-fasa yang terdapat dalam model air terjun. Kesimpulannya, sistem ini bukan sahaja berfungsi dalam dunia nyata, bahkan juga berfungsi dengan cepat dan tepat. Sistem ini boleh menampung banyak pengguna dalam sesuatu masa. Jadi, pensyarah boleh menggunakan sistem dalam pada bila-bila masa.

ABSTRACT

The system that will be developed during the thesis is called e-Lecturer Assistant System. The main purpose of the system is to help the lecturers in solving their daily tasks. The lecturers daily tasks are calculate students' attendance percentage, student marks, prepare exam papers, store pass year exam questions, set reminder, file transfer and chatting. The system will combine all the functions in a single program. Therefore, it is suitable for the lecturers use especially in handling multiple tasks. Before the system had been developed, most of the tasks have been carried out manually and cause them wasting valuable times. Besides that, they also need to use several kind of application like Microsoft Word, Excel or other to complete their tasks. Due to this point, they have to spending a lot of time in learning how to use the application. Literature review is to collect data. It enables the developer to determine the scope of project and user requirements. Besides that, methodology can help to produce a better quality product, acceptability to the user, maintainability and consistency of software. The Waterfall Model has been chosen to be the methodology for this project. The project will be implemented based on the phases in the model to ensure the user requirement and objective of the project have been fulfilled. As a conclusion, the system developed not only expected to be workable in the real environment, but also efficient in term of speed and correctness. It can support many concurrent users to interact with the system in a time. Therefore, the users can use the system in anytime they like.

TABLE OF CONTENTS

CONTENT	PAGE
THESIS^ APPROVAL STATUS FORM	
e-LECTURER ASSISTANT SYSTEM	i
ADMISSION	ii
DEDICATION	iii
ACKNOWLEDGEMENTS	iv
ABSTRAK	v
ABSTRACT	vi
TABLE OF CONTENTS	vii
LIST OF TABLES	xi
LIST OF FIGURES	xiii
LIST OF ACRONYMS	xv
LIST OF APPENDICES	xvi
CHAPTER I INTRODUCTION	1
1.1 Overview	1
1.2 Problem Statement	3
1.3 Objective	5
1.4 Scope	6
1.5 Contributions	9
1.6 Expected Output	9
1.7 Summary	11

CHAPTER II LITERATURE REVIEW	12
2.1 Introduction	12
2.2 Fact and finding	13
2.2.1 Research of Methodology	13
2.2.2 Research on Concept	15
2.2.3 Research on Technology	17
2.2.4 Research on Protocol	18
2.2.5 Research on Related System	18
2.2.6 Research on Programming Language	21
2.3 Conclusion	21
CHAPTER III PROJECT PLANNING AND METHODOLOGY	23
3.1 Introduction	23
3.2 High-Level Project Requirements	24
3.2.1 Project Facilities Requirement	24
3.2.2 Software Requirement	24
3.2.3 Hardware Requirement	25
3.3 System Development Approach	26
3.4 Project Schedule and Milestones	30
3.4.1 PSM I Activities	31
3.4.2 PSM II Activities	32
3.5 Conclusion	32
CHAPTER IV ANALYSIS	34
4.1 Introduction	34
4.2 Analysis of Current System	35
4.2.1 Business Process	35
4.2.2 Problem Analysis	36
4.2.3 Problem Statements	38
4.3 Analysis of To Be System	39
4.3.1 Functional Requirement	39
4.3.2 Technical Requirement	42

4.3.2.1	Software Requirement	42
4.3.2.2	Hardware Requirement	43
4.3.2.3	Implementation/Deployment Requirement	45
4.4	Conclusion	45
CHAPTER V DESIGN		46
5.1	Introduction	46
5.2	Preliminary/High-Level Design	47
5.2.1	Raw input/Data	47
5.2.2	System Architecture	48
5.2.3	User Interface Design	49
5.2.3.1	Navigation Design	49
5.2.3.2	Input Design	49
5.2.3.3	Output Design	51
5.2.4	Database Design	53
5.2.4.1	Logical Database Design	54
5.3	Detailed Design	56
5.3.1	Software Specification	56
5.3.1.1	Sequence Diagram	59
5.3.1.2	Collaboration Diagram	66
5.3.1.3	Activity Diagram	66
5.3.1.4	Class Diagram	69
5.3.1.5	Pseudo code	70
5.3.2	Physical Database Design	72
5.4	Conclusion	77
CHAPTER VI IMPLEMENTATION		78
6.1	Introduction	78
6.2	Software Development Environment Setup	79
6.2.1	Programming Language	80
6.2.2	Development Environment	80
6.2.3	Operating System	80
6.2.4	Database	81
6.3	Software Configuration Management (optional)	81

6.3.1 Configuration Environment Setup	81
6.3.1.1 Installation of Microsoft Visual Studio 6.0	81
6.3.1.2 Installation / Configuration MySQL	82
6.3.2 Version Control Procedure	82
6.4 Implementation Status	83
6.5 Conclusion	84
CHAPTER VII TESTING	85
7.1 Introduction	85
7.2 Test Plan	86
7.2.1 Test Organization	86
7.2.2 Test Environment	87
7.2.3 Test Schedule	87
7.3 Test Strategy	88
7.3.1 Classes of Tests	90
7.4 Test Design	90
7.4.1 Test Description	90
7.4.2 Test Data	98
7.5 Test Case Results	103
7.6 Summary	105
CHAPTER VIII PROJECT CONCLUSION	106
8.1 Observation on Weaknesses and Strengths	106
8.2 Propositions for Improvement	107
8.3 Conclusion	107
BIBLIOGRAPHY	111
APPENDICES	112

LIST OF TABLES

TABLE NO	TITLE	PAGE
2.1:	Comparison between Waterfall Model and Spiral Model	15
3.1:	Software Requirement	24
3.2:	Hardware Requirement	25
3.3:	Tasks and Activities in PSM I	31
3.4:	Tasks and Activities in PSM II	32
4.1:	Software Requirement for Developer Side	42
4.2:	Software Requirement for User Side	43
4.3:	Hardware Requirement for Developer Side	43
4.4:	Hardware Requirement for User Side	44
5.1:	Sample of Students Data	47
5.2:	Table Input Specification	50
5.3:	Table Output Specification	51
5.4:	Description of Actor	57
5.5:	Description of Use Case	57
5.6:	Table studentlist	72
5.7:	Table member	73
5.8:	Table attendance	73
5.9:	Table book	74
5.10:	Table reminder	75
5.11:	Table question_bank	75
5.12:	Table evaluation_schema	76
5.13:	Table student_mark	77

6.1:	Implementation Status of e-Lecturer Assistant System for Each Module	83
7.1:	Test schedule according tasks, activities and duration to carry out testing activities	88
7.2:	e-Lecturer Assistant System Test Script	91
7.3:	Test Summary Report	98
7.4:	Login Test Data	98
7.5:	Manage Member Test Data	99
7.6:	Set Reminder Test Data	99
7.7:	Set Academic Test Data	100
7.8:	Calculate Students Mark Test Data	100
7.9:	Manage Books Data Test Data	101
7.10:	Manage Students Data Test Data	102
7.11:	Chatting Test Data	102
7.12:	Test case record according test case ID, tester, test objective, test data, result	103

LIST OF FIGURES

FIGURE NO	TITLE	PAGE
2.1:	Spiral Model	14
5.1:	Actor and Use Case Notation	56
6.1:	Software Development Environment Setup Architecture	79

LIST OF ACRONYMS

ACRONYM	DESCRIPTION
DBMS	Database Management System
ERD	Entity Relationship Diagram
ELS	Electronic Lecture System
FTP	File Transfer Protocol
GUI	Graphic User Interface
KUTKM	Kolej Universiti Teknikal Kebangsaan Malaysia
OOP	Object-Oriented Programming
ODBC	Open Database Connectivity
PSM I	<i>Projek Sarjana Muda I</i>
PSM II	<i>Projek Sarjana Muda II</i>
PDA	Personal Digital Assistant
SQL	Simplified Query Language
UML	Unified Modeling Language
UTP	Unshielded twisted-pair

LIST OF APPENDICES

APPENDIX NO	TITLE	PAGE
APPENDIX A:	Gantt Chart	112
APPENDIX B:	e-Lecturer Assistant System Literature Review	117
(1)	Waterfall Model	118
(2)	Client Server Architecture	119
(3)	2-tier Architecture	119
(4)	Alarm / Expiration Function	120
(5)	Attachment Files Function	121
(6)	Problem Statement	122
(7)	Three Layer Architectures	122
(8)	Navigation Design for Desktop System	123
(9)	Entity Relationship Diagram	123
APPENDIX C:	Class Diagram	124
(1)	UML Diagram about System Functions	125
(2)	Notation of Class	125
(3)	Class Diagram for e-Lecturer Assistant System	126
APPENDIX D:	Sequence Diagram	127
(1)	Basic Flow for Lecturer Login	128
(2)	Alternate Flow for Lecturer Login	128
(3)	Exception Flow for Lecturer Login	129
(4)	Basic Flow for System Administrator Login	129
(5)	Alternate Flow for System Administrator Login	130

(6) Exception Flow for System Administrator Login	130
(7) Basic Flow for Lecturer Set Reminder	131
(8) Alternate Flow for Lecturer Set Reminder	131
(9) Exception Flow for Lecturer Set Reminder	132
(10) Basic Flow for Lecturer Update Reminder	132
(11) Alternate Flow for Lecturer Update Reminder	133
(12) Exception Flow for Lecturer Update Reminder	133
(13) Basic Flow for Lecturer Delete Reminder	134
(14) Alternate Flow for Lecturer Delete Reminder	134
(15) Basic Flow for Lecturer Generate Exam Paper	135
(16) Alternate Flow for Lecturer Generate Exam Paper	135
(17) Basic Flow for System Administrator Update Member Data	136
(18) Alternate Flow for System Administrator Update Member Data	136
(19) Exception Flow for System Administrator Update Member Data	137
(20) Basic Flow for System Administrator Delete Member Data	137
(21) Alternate Flow for System Administrator Delete Member Data	138
(22) Basic Flow for Lecturer Calculate Student Marks	138
(23) Alternate Flow for Lecturer Calculate Student Marks	139
(24) Exception Flow for Lecturer Calculate Student Marks	139
(25) Basic Flow for Lecturer Update Student Marks	140
(26) Alternate Flow for Lecturer Update Student Marks	140
(27) Exception Flow for Lecturer Update Student Marks	141
(28) Basic Flow for Lecturer Delete Student Marks	141
(29) Alternate Flow for Lecturer Delete Student Marks	142
(30) Basic Flow for Student Insert Book Data	142
(31) Alternate Flow for Student Insert Book Data	143
(32) Exception Flow for Student Insert Book Data	143
(33) Basic Flow for Student Delete Book Data	144

(34) Alternate Flow for Student Delete Book Data	144
(35) Basic Flow for Lecturer Update Student Data	144
(36) Alternate Flow for Lecturer Update Student Data	145
(37) Basic Flow for Lecturer Delete Student Data	145
(38) Alternate Flow for Lecturer Delete Student Data	145
(39) Basic Flow for Lecturer Take Student Attendance	146
(40) Alternate Flow for Lecturer Take Student Attendance	146
(41) Basic Flow for Lecturer Calculate Student Attendance	147
APPENDIX E: Collaboration Diagram	148
(1) Collaboration Diagram for Login Module (Lecturer)	149
(2) Collaboration Diagram for Login Module (Administrator)	149
(3) Collaboration Diagram for Reminder Module - Set Reminder	150
(4) Collaboration Diagram for Reminder Module - Update Reminder	150
(5) Collaboration Diagram for Reminder Module - Delete Reminder	151
(6) Collaboration Diagram for Generate Exam Paper Module	151
(7) Collaboration Diagram for Manage Member Data Module – Update Member	152
(8) Collaboration Diagram for Manage Member Data Module – Delete Member	152
(9) Collaboration Diagram for Students Mark Module – Calculate Student Mark	153
(10) Collaboration Diagram for Students Mark Module – Update Student Mark	153
(11) Collaboration Diagram for Students Mark Module – Delete Student Mark	154
(12) Collaboration Diagram for Manage Books Data Module - Insert Book Data	154

(13) Collaboration Diagram for Manage Books Data Module - Delete Book Data	155
(14) Collaboration Diagram for Manage Students Data Module - Update Student Data	155
(15) Collaboration Diagram for Manage Students Data Module - Delete Student Data	156
(16) Collaboration Diagram for Student Attendance Module - Take Student Attendance	156
(17) Collaboration Diagram for Manage Students Data Module - Calculate Student Attendance	157
APPENDIX F: Activity Diagram	158
(1) Activity Diagram for Login Module	159
(2) Activity Diagram for Manage Member Data Module	160
(3) Activity Diagram for Student Attendance Module	161
(4) Activity Diagram for Students Mark Module	162
(5) Activity Diagram for Manage Books Data Module	163
(6) Activity Diagram for Reminder Module	164
(7) Activity Diagram for Generate Exam Paper Module	164
(8) Activity Diagram for Manage Students Data Module	165
APPENDIX G: User Interface Design	166
(1) Main Page	167
(2) Administration Page	168
(3) Set Academic Page	168
(4) Attendance Page	169
(5) Books Page	170
(6) Chatting Page	170
(7) File Transfer Page	171
(8) Forget Password Page	171
(9) Generate Exam Paper Page	172
(10) Login Page	172
(11) Maintenance Page	173

(12) Reminder Page	174
(13) Member Registration Page	175
(14) Evaluation Schema Page	176
(15) Student List Page	177
APPENDIX H: User Manual	178

CHAPTER I

INTRODUCTION

1.1 Overview

Interest topic of this project is developing an “**e-Lecturer Assistant System**”. This system is designed especially for lecturers in Kolej Universiti Teknikal Kebangsaan Malaysia. As we know, lecturers have spending a lot of time and effort to teach students from the state that they know nothing about programming to the state that they can develop a system using their own programming knowledge.

The e-Lecturer Assistant System will become an essential assistant for the lecturers in handling their daily task. This system tries to computerize most of the daily tasks conducted by the lecturers, which includes preparing questions and answers for test and exam papers, calculating students attendance percentage, check whether particular student can sit for final exam or not base on attendance, recording and calculating students' marks, set reminder for a special event or appointment, storing and managing students' information and books, chatting and sending files between lecturers and others.

Before this, lecturers do not have a system that helps them handling their daily tasks. Therefore, they need to complete their tasks manually. For example, lecturers need to open Microsoft Word application for preparing test and exam papers. Besides that, they need to open Microsoft Excel to record students' marks. Therefore, lecturers will use two programs to complete their tasks. This will waste the lecturers' time

because they need to spend a lot of time and effort to complete their tasks manually or semi-computerized.

Actually, preparing test or exam papers and recording students' marks can be done by using developed system and can save lecturers time. One of the system purposes is to help lecturers get done of their daily tasks in less time. To put in briefly, the e-Lecturer Assistant System help lecturers work even smarter.

The risk of losing data problem will occur when lecturers come to take students' attendance. Normally, most lecturers will take students' attendance manually by passing around an attendance sheet in lecture hall for students to fill up their name and sign on the sheet. After that, lecturers have to transfer the attendance into spreadsheet in Microsoft Excel or just keep it in a pocket file. Sometimes, lecturers forget where they have put the file.

Due to this point, attendance taken by lecturers will be saved into MySQL database through developed system. Hence, they no need to waste any time in finding attendance sheets. Besides that, this system can help them during calculating percentage of attendance for each student to determine whether that particular student can sit for final exam or not.

Before this, most lecturers have to do a complex calculation marks and percentages for each test, lab test, quiz, midterm exam, final exam, assignment, attendance and others manually or using Microsoft Excel application. This will cause them headache.

Besides, lecturers also lack of proper way to jot down time and date for appointments, meeting or other special occasions. Normally, they will jot down in a piece of paper, but this method does not effective enough to remind them while they are very busy with other tasks.

With e-Lecturer Assistant System, lecturers can insert the date and time of the appointments, meetings or special occasions in the system. Then the system will send reminding message to them a day before appointments, meetings or occasions take place. So, they will not miss any important event.

Nowadays, lecturers do not have a proper way to store students' information such as email address, phone number, and address. Most of them will jot down in a book or Microsoft Word. Due to this point, they have to spend money to buy books. Besides, lecturers always busy with their tasks until forget where they have kept the book. This will waste their time in finding the book when it is needed.

By using this system, lecturers can save their money and time because they can store the information in database. By the way, database has large space to store the information. So, lecturers not require a physical space in office to keep the information.

It is beneficial to use standard steps to develop a system; these steps are called system development methodology. Methodology that used during development of this system is called Waterfall Model. This methodology consists of six main phases such as requirements, design, implementation, integration, evaluation and maintenance.

1.2 Problem Statement

Before the e-Lecturer Assistant System has been developed, there are some problems faced by the KUTKM lecturers in handling their daily tasks. These problems had been identified during the research phase and listed below:

i) Lack of Systematic System

Lecturers normally do their daily tasks manually by using various applications such as Microsoft Word, Excel or other. There do not have systematic systems to handle all the tasks.

ii) Wasting Valuable Time

Valuable time will be wasted in finding files (attendance sheet) when lecturers forget the place that they had kept the files.

iii) Losing Data or File

Sometime lecturers do not aware that they have deleted the important files in the desktop computer. This will cause them losing the data or file.

iv) Handling Complicated Calculation

Lecturers need to handle complicated calculation for the students' marks in each semester.

v) Forget Certain Appointments, Meetings or Special Occasions

Lecturers sometime very busy until forget to attend certain appointments, meetings or special occasions because there do not have an effective reminder to remind them.

vi) Communication between Lecturers Become Less

There does not have a system to enable them to communicate with each other.