

FACULTY OF COMPUTER SYSTEMS & SOFTWARE ENGINEERING

UNIVERSITI MALAYSIA PAHANG



SOFTWARE TESTING AND MAINTENANCE

BCS 3343

DOCUMENTATION

AFZAN BINTI AHMAD

CB10097

TO:

MADAM ROZLINA BINTI MOHAMMED

Due Date:

6 MAY 2012

SEMESTER 2

ACADEMIC SESSION 2011/2012

STUDENT'S DECLARATION

I declare that this thesis entitled "SCHOOL FEES PAYMENT and ACCOUNTING MANAGEMENT SYSTEM" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature :

Name : AFZAN BINTI AHMAD

Date :

SUPERVISOR'S DECLARATION

“I hereby declare that I have checked this project report and in my opinion this project is satisfactory in terms of scope and quality for the award of the degree of Bachelor of Computer Science (Software Engineering)”

Signature :

Name of Supervisor : ROZLINA BINTI MOHAMMED

Date :

DEDICATION

Thankful to Allah S.W.T

Special Dedication of This Great Moment to My Family...

My Mother, Noor Aini Bt Lani

My Father, Ahmad Bin Hj Salam

All My Siblings

To My Supervisor...

Mdm Rozlina Bt Mohammed

Thanks a lot for your support, encouragement and guidance

To all my lecturers FSKKP UMP...

Thanks a lot for your support, encouragement and guidance

To all my friends out there....

And all 3BCS students.

Thanks for yours support and cooperation.

ACKNOWLEDGEMENT

Alhamdulillah, firstly I would like to thank Allah S.W.T. because without HIS blessing, I cannot finish my final year project successfully. I would also like to thank my supervisor, Miss Rozlina Bt Mohammed for her concerns, encouragement, critics, and ideas for the whole phase in order to finish this project.

Besides that, we would like to thank to my parents because give spirit and support. Even though they are not around, but I can feel that they always prayed for my success. I would also very thankful to all lecturers in the faculty.

Lastly, my fellow friends of 3BCS students that support me from the beginning till the end of the project. Thank you again and may Allah bless all of you.

ABSTRACT

School Fees Payment and Accounting Management System was developed to manage the fees payment that included PIBG fees, School fees and Hostel fees and managing school accounting record. Current practice use manual fees form to manage fees payment taken by teacher and using book accounting record to record all debit and credit for every month. This methods burden staff and teacher who want to manage fees payment and accounting record in manual way. SFPAMS was the solution to this problem. One of the upgrades is easing the process of by auto calculation fees payment. This system can help teacher to manage the fees payment because one of the objective to archive. Therefore, this system will be very helpful and can make the management of school fees payment and accounting record run smoothly. SFPAMS is develop by using Adobe Dreamweaver CS4 and Xamp Server

ABSTRAK

School Fees Payment and Accounting Management System dibangun untuk menguruskan pembayaran yuran seperti yuran PIBG, yuran sekolah dan yuran asrama dan juga menguruskan akaun sekolah.. Cara yang digunakan sebelum ini ialah menggunakan borang bayaran yuran yang diuruskan oleh guru dan menggunakan buku log akaun untuk merekod segala pembelanjaan sekolah terimaan dan bayaran untuk setiap bulan. Cara ini menyusahkan staf dan guru dalam pengendalian pembayaran yuran sekolah dan menguruskan akaun sekolah. SFPAMS merupakan penyelesaian bagi masalah ini. Salah satu daripada penaikkan taraf tersebut adalah memudahkan pengiraan pembayaran yuran dalam automatik pengiraan ini kerana salah satu objektif untuk dicapai. Oleh itu, system ini sangat berguna dan membolehkan system ini berjalan dengan sempurna. SFPAMS menggunakan Adobe Dreamweaver CS4 dan Xamp server.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
1	INTRODUCTION	1
	1.1 Background	1 - 2
	1.2 Problem Statement	3
	1.3 Objectives	3
	1.4 Scopes	4
	1.5 Thesis Organization	4
2	LITERATURE REVIEW	5
	2.1 Overview of the manual system	5 - 6
	2.1.1 The process of manual system	6 - 7
	2.1.2 Existing on manual school fees payment	7
	2.1.2.1 PIBG fees payment form	7 – 8
	2.1.2.2 PIBG fees form for Form 1	8
	2.1.2.3 Receipt	9
	2.1.2.4 Hostel fees form	9
	2.1.2.5 Hostel monthly fees form	10
	2.1.2.6 Daily record	10
	2.1.3 Existing system	11
	2.1.3.1 School Accounting Management	11
	2.1.3.1.1 Login	11 - 12
	2.1.3.1.2 Change Password	12
	2.1.3.1.3 Update Profile	12 - 13
	2.1.3.1.4 Balance PCG	13

2.1.3.1.5 Balance BMA	13 - 14
2.1.3.1.6 Debit PCG	14
2.1.3.1.7 Sheet Balance PCG	14 - 15
2.1.3.1.8 PCG payment	15
2.1.3.1.9 PCG payment balance	15 - 16
2.1.3.1.10 Debit BMA	16
2.1.3.1.11 BMA receipt balance	16 - 17
2.1.3.1.12 Balance of BMA	17
2.1.3.1.13 Balance info of BMA	17 - 18
2.2 Propose System	18
2.3 Comparison between existing manual and purpose system	18 - 19
2.4 Data Flow	19 - 20
2.4.1 Defining data flow diagram	20 - 21
2.4.1.1 Data Flow Diagrams	21
2.4.2 Defining DFD components	21
2.4.2.1 Entity	22
2.4.2.2 Process	22
2.4.2.3 Data Store	22
2.4.2.4 Data Flow	22 - 23
2.4.3 Process for developing DFD	23 - 24
2.4.3.1 A simple DFD example	24
2.4.3.2 Level 1 DFD	24 - 25
2.5 Business Process Modelling	25
2.5.1 Cross Functional Flow Charts	25 - 26
2.5.1.1 Standard Cross Functional	26 - 28
Flowchart symbols and usage	
2.5.1.2 Cross Functional Flowchart	28 - 29
Process	
2.6 Unified Modelling Language	30

2.6.1 Use Case	30
2.6.1.1 Use Case Diagram	30
2.6.1.2 Definitions and symbols	30 - 31
2.6.2 Sequence Diagram	31 - 32
2.6.2.1 Symbols and definitions	32 - 33
2.6.3 Activity Diagram	33
2.6.3.1 Symbols and description	33
2.6.4 Class Diagram	34
2.6.4.1 Basic class diagram symbols and notation	34 - 37
2.7 Comparison between data flow diagram, business process modelling and unified modelling language	37 - 39
2.8 Conclusion	39 - 40
3 METHODOLOGY	41
3.1 Iterative and Incremental Development	41 - 42
3.1.1 Definition	42
3.1.2 Incremental Development	43 - 45
3.2 Phase	45
3.2.1 Planning	46
3.2.1.1 Development Tools	46 - 48
3.2.1.2 Risk Management	48 - 50
3.2.2 Requirement	50 - 56
3.2.2.1 Documentation	57
3.2.3 Analysis and Design	57
3.2.3.1 Database Design	57 - 58
3.2.3.3 Design	58 - 62
3.2.4 Implementation	63 - 64

	3.2.5 Testing	65
	3.2.5.1 Document	65
	3.2.6 Deployment	67
	3.3 Conclusion	67
4	IMPLEMENTATION	68
	4.1 Interface	68
	4.1.1 Login Interface	68 - 69
	4.1.2 Setup Fees Form Interface	69 - 70
	4.1.3 List of Fees Payment Interface	70 - 71
	4.1.4 Receipt Fees Payment Interface	71 - 72
	4.1.5 Record of BMA debit	72
	4.1.6 Record of BMA debit balance	73
	4.1.7 Record of BMA debit	73
	4.2 Coding	74
	4.2.1 Insert Coding	74
	4.2.2 View Coding	74 - 75
	4.2.3 Delete Coding	75
	4.2.4 Search Coding	76
5	RESULT AND DISCUSSION	77
	5.1 Testing	77
	5.1.1 Objectives	77 - 78
	5.1.2 Background	78
	5.1.3 Scope	78
	5.2 Test Case	78
	5.2.1 Test case – Validation Login	79 - 80

	5.2.1.1 Result	80 - 82
	5.2.2 Test case – Test Calculation of fees payment	82 - 83
	5.2.2.2 Result	83 - 84
	5.3 Expected Result	84
	5.4 Advantage and disadvantage School Fees Payment Accounting Management System	85
	5.4.1 Advantage	85
	5.4.2 Disadvantage	85
	5.5 Constraint	85
	5.5.1 System Constraint	85
	5.5.2 Development Constraint	85 - 86
6	CONCLUSION	87
	REFERENCES	88
	APPENDICES	89
	Appendix A	90
	Appendix B	91
	Appendix C	92
	Appendix D	93
	Appendix E	94
	Appendix F	95

LIST OF TABLES

TABLES NO	TITLE	PAGE
2.1	Comparison between existing school payment in manual and purpose system	4
2.2	Standard cross functional flowchart symbols	26 - 28
2.3	Symbols and definitions	31
2.4	Symbols and definitions for sequence diagram	32- 33
2.5	Symbols and descriptions for activity diagram	33
2.6	Symbols and description for class diagram	34 - 37
2.7	Comparison between data flow diagram, business process modelling and unified modelling language	37 - 39
3.1	Hardware requirements	46 - 47
3.2	Software facilities and purpose	47 - 48
3.3	Risk management	49 - 50

LIST OF FIGURES

FIGURE NO	TITLE	PAGE
2.1	List PIBG fees form by each form	8
2.2	PIBG fee form for form 1	8
2.3	Receipt school fees payment	9
2.4	Receipt hostel fees payment	9
2.5	Hostel monthly fees	10
2.6	Record daily	10
2.7	Login interface	11
2.8	Change password interface	12
2.9	Update profile interface	12
2.10	Balance information PCG	13
2.11	Balance information BMA	13
2.12	Debit PCG interface	14
2.13	Sheet balance receipts PCG	14
2.14	PCG payment paperwork interface	15
2.15	PCG payment balance sheet information interface	15
2.16	Paperwork debit BMA interface	16
2.17	BMA receipts balance information interface	16
2.18	Balance of payment information BMA interface	17
2.19	Balance of payment information BMA interface	17
2.20	Data flow diagram	21
2.21	DFD components	23
2.22	Level 0 diagrams	24
2.23	Context diagram	24

2.24	Level 1 DFD	25
2.25	Business process as transformations of inputs to outputs	25
2.26	Flow process	29
2.27	Cross functional map	29
3.1	Iterative and incremental development	42
3.2	Incremental development stage 1	43
3.3	Incremental development stage 2	44
3.4	Incremental development stage 3	44
3.5	Context diagram for SFPAMS	52
3.6	Level 1 DFD - Login	54
3.7	Level 1 DFD – Fees setup	55
3.8	Level 1 DFD – Payment	56
3.9	Database ER-Diagram	58
3.10	Fees Payment interface	59
3.11	Fees interface	60
3.12	Debit BMA record	61
3.13	Balance of debit BMA record	62
3.14	Login interface	64
3.15	PIBG fees payment	66
4.1	Login teacher interface	69
4.2	Login staff interface	69
4.3	Setup fees payment interface	70
4.4	View of fees payment interface	70
4.5	PIBG fees payment interface	71
4.6	Receipt of fees payment interface	72
4.7	Search year record of BMA debit	72
4.8	Display of record of BMA debit by year	72
4.9	Record of BMA interface	73
4.10	Print record of BMA interface	73

4.11	Insert coding	74
4.12	Call data to display	75
4.13	View coding	75
4.14	Delete coding	75
4.15	Search coding	76
5.1	Entering username and password by null	80
5.2	Entering username by null	81
5.3	Entering invalid username and password	81
5.4	Entering username by integer	82
5.5	Not check the checkbox	83
5.6	Check all the checkbox	84
5.7	Uncheck one checkbox	84

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	Gantt Chart	90
B	School Accounting Management System	91
C	Software Development Planning (SDP)	92
D	Software Requirement Specification (SRS)	93
E	Software Design Description	94
F	Software Testing and Maintenance	95

CHAPTER 1

INTRODUCTION

This chapter briefly discuss on the overview of this research. It contains five sections. The first section is introduction; follow by the problem statement. Next are the objectives where the project's goal is determined. After that are the scopes of the system and lastly is the thesis organization which briefly describes the structure of this thesis.

1.1 Background

Nowadays, every school will be receive new student to study at their school at begining of the year. This event calls as the registration for receiving new students. As the teacher, they need to managing transaction fees payment during that day. As the parents of the student, they need to be paid that any types of fees payment such as PIBG fees, school fees, hostel fees and etc by according to the total maount of the fees payment that have been state by the school.

Each of transaction need to be record as a evidence that describe the transaction have been made or to make redo view back as the referencing. For example at school need to record each of the transaction payment in a account book as record such as PIBG fees, school fees , hostel and etc. Each of them need to be record in the systematically way. Accounting is like to use for measure its financial performance by noting and classifying all the transaction like sales, purchase, aset and etc [1]. Other said that it is the art of recording, classifying and summarizing in a significant manner and in terms of money, transactions and events which are, in part at least of a financial character and interpreting the result thereof [1]

To build the system need to store data to keep data in safety places and keep data in integrity. Database can store and elaborate complex information and huge amount of data. The advantage using database is guick and real time access, high security, standards establishment and etc. [2]

Therefore, developing the system and need to be implement at Secondary Religion School need the effective and smooth management system that have the similiry with the system that have been use at all the world. That system call as School Management System. A School Management System is software that are installed all school in the world that are allows users to store almost all their schools information in form electronically [3]. This information can be easily shared with authorized users, records and can be easily searched and report can be easily generated [3].

For this thesis our scope will be focus on developing at Secondary Religion School in all school Pahang State will be implemented by using the similirity system that use School Management System use by school in the world. School Fees Payment and Accounting Management System (SFPAMS) is a part of a module from School Management System that will be focus on manage payment fees and updating school accounting. These system are using web base application and MySql PHP.

1.2 Problem Statement

In early of the year, student will start school session and they have to pay for their school fee such as hostel, PIBG fee, annual fee and etc. Teacher will be collecting fee by using form and do manual calculating by using calculator. However, the record for student who do the payment may be missing or forget to updated and it will be causing a lot of time to search the record back or the student will be identified as debtor. For the special case need one family only to pay their PIBG fee and it will cause other member family who don't know paid twice for PIBG fee. Teacher need managing account record by using manual way such as calculating using calculator devices, write down the transaction debit and credit on the paper, need be tally with other account file and also take a time if have do some mistaken in account report. Besides that, many schools are still using manual way and not using current technology nowadays.

1.3 Objectives

There are several objectives of this research:

- i. To develop a School Fess Payment system to record payment for several entities & receipt printing as a proof for the payment transaction.
- ii. To develop an Accounting Management System (SFPAMS) for assisting SMA in managing their basic accounting needs
- iii. Comparative study between Unified Modeling Language, Business Process Modeling and Data Flow Diagram for assisting system developer in developing a small Information System based project.

1.4 Scopes

The scopes of this project are:

- i. School Fees Payment and Accounting Management System are developing for SMA Pahang.
- ii. This system will be used by two users which are staff and teacher.
- iii. School Fees Payment and Accounting Management System will develop based on modeling flow process.
- iv. School Fess Payment and Accounting Management System contain 5 modules which are login module, fees setup module, fees payment module, generate record module and accounting record module.

1.5 Thesis Organization

This thesis consists of five chapters. Chapter 1: Introduction briefly describes and introduces the system. This system preliminary shows the basic concept of the system, problem statements of the system, objectives, scopes, and how the report is organized. Chapter 2: Literature Review depicts the manual systems and the existing systems as the case studies of the project. This chapter also reviews the technique, method, equipment, and technology that had been used in the case studies. Chapter 3: Methodology discusses about the overall workflow in the development of the project. This chapter also discusses the method, technique or approach that has been used while designing and implementing the project. Chapter 4: Testing discusses about the expected result by testing system using insert data, functionality, and flow of the process system. Chapter 4: Conclusion briefly summarizes the project.

CHAPTER 2

LITERATURE REVIEW

This chapter elaborates on the current system that uses in all religious secondary school in all Pahang state. In this chapter we select Sekolah Menengah Agama Tengku Ampuan Fatimah located at Pekan Pahang as our case study. This chapter also explains the importance developing of School Fees Payment and Accounting Management System that must be develop for religious secondary school in all Pahang state. This chapter also describes the explanations between the current system practice by the organization/present review system and process modeling that represent flow process system by using Business Process Modeling, Data Flow Diagram and UML.

2.1 Overview of the manual system

Sekolah Menengah Agama Tengku Ampuan Fatimah 26600 Pekan, Pahang is still use manual system to manage the payment such as PIBG fee form, School fee form, Hostel fee form, other fee and managing school accounting record. This work is handled by teacher or staff at school to manage it. Teachers need be aleart and know which one of students are paid for half or not paid or paid all. To

overcome this matter teacher will take note by writing some sign such as “HALF PAID” or “NOT PAID” or “PAID” and etc.

For PIBG fee payment, some students have their siblings that study in different form at same school. Student will tell their teacher about existing his/her siblings that will be pay their PIBG fee payment. To resolve this matter, teacher or staff will check it first whether the student statement is true or not by do some referencing from student registration record. If true, teacher will be take note by put some sign to state that his/her sibling have paid for PIBG fee.

At the end of month, staffs will be record all their collecting school fees payment and do updated manage account such as create new account file, updated debit or credit account and updated current total amount left for school and etc.

2.1.1 The process of manual system

In these section will be explain about the existing manual system that has been used nowadays at Sekolah Menengah Agama Tengku Ampuan Fatimah 26600 Pekan, Pahang. Firstly, teacher or staff will search student name first. After they found student name then they will ask whether that is their name or not. If the student said that is their name, teacher/staff will ask whether they want to pay full, half paid or not want to paid fees payments.

If the student said that they want to delay for fees payment and the teacher will take note by giving some sign not paid. If the student want to paid, they will tell to teacher/staff they want to paid half or full fees payment and they give the money to the teacher or staff. Teacher or staff will take the money and start to calculate total amount of fees payments.

After done calculating fees, teacher or staff will start to write down receipt fees payment then they will give back balance money and copy receipt that contains several different type of fees such as PIBG fees, school fees and hostel fees. Student

takes balance money and copy receipts. All activity for fees payment will be repeated until no more student want to pay fees.

Teacher or staff will start to write record and updated student status. Teacher or staffs will calculating all collecting fees total amount that they get today. For the accounting record, staff will be manage it by identifying which one transaction should be debit or credit. It happen during at last week in each month. As the staff that have assigned to manage school account, they need to create new account record and they will be start calculate to get know how much many left, how much money that have been spend for this month, how much money should be spend for next month and etc. This work need to be tally very well and accurate for avoiding school become bankrupt and etc. After get the result for total amount left, they should make planning to use money in valuable way not waste it. But, they still using the calculator devices to calculate it and create new account record using the specific book for accounting book. They will be repeating same method for every month at last week.

2.1.2 Existing on manual School Fees Payment.

Sekolah Menengah Agama Ampuan Fatimah, 26600 Pekan Pahang is still using the manual form to managing school fees payment. This manual form is including all information lists of form PIBG fees payment, school fees payment, hostel fees payment, school record and accounting record.

2.1.2.1 PIBG fees payment form

Figure 2.1 show list of PIBG fee form by each form. It contain list of information PIBG fees payment and included for payment.

2.1.2.2 PIBG fees form for Form 1

REPUBLIC OF SOUTH AFRICA SOUTH AFRICAN REPUBLICAN AIR FORCE TECHNICAL COLLEGE OF AIRCRAFT ENGINEERING		 NABU-PTC-1001			
REGISTRATION NUMBER	NO.	EXAMINATION DATE YEAR MONTH DAY	EXAMINER NAME Surname	SIGNATURE OF EXAMINER	SIGNATURE OF CANDIDATE
AIRCRAFT ENGINEERING	1	1999	1999		
	2	1999	1999		
	3	1999	1999		
	4	1999	1999		
	5	1999	1999		
	6	1999	1999		
	7	1999	1999		
	8	1999	1999		
	9	1999	1999		
	10	1999	1999		
	11	1999	1999		
	12	1999	1999		
	13	1999	1999		
	14	1999	1999		
	15	1999	1999		
P.T.C. 1001	1	1999	1999		
	2	1999	1999		
	3	1999	1999		
	4	1999	1999		
	5	1999	1999		
	6	1999	1999		
	7	1999	1999		
	8	1999	1999		
	9	1999	1999		
	10	1999	1999		
	11	1999	1999		
	12	1999	1999		
	13	1999	1999		
	14	1999	1999		
	AIRCRAFT ENGINEERING	1	1999	1999	
2		1999	1999		
3		1999	1999		
4		1999	1999		
5		1999	1999		
6		1999	1999		
7		1999	1999		
8		1999	1999		
9		1999	1999		
10		1999	1999		
11		1999	1999		
12		1999	1999		
13		1999	1999		
14		1999	1999		
15		1999	1999		
SIGNATURE OF EXAMINER: _____ SIGNATURE OF CANDIDATE: _____ PRINT NAME: _____ PRINT NAME: _____ NO. OF MARKS: _____ NO. OF MARKS: _____					
SIGNATURE OF EXAMINER: _____ SIGNATURE OF CANDIDATE: _____ PRINT NAME: _____ PRINT NAME: _____ NO. OF MARKS: _____ NO. OF MARKS: _____					

Figure 2.2: PIBG fee form for form 1

2.1.2.3 Receipt

Figure 2.3 shows receipt school fees payment. It contain list of information for school fees receipt.

RECEIPT SCHOOL FEES PAYMENT

Ministry of Education and Higher Education
Republic of Indonesia

No. 08/750

School Name: _____

Student Name: _____

No	Description	Amount
1	Registration Fee	
2	Library Fee	
3	Student Activity Fee	
4	Student Health Fee	
5	Student Insurance Fee	
6	Student Uniform Fee	
7	Student Transportation Fee	
8	Student Other Fee	
9	Student Other Fee	
10	Student Other Fee	

Total: _____

Signature: _____

Date: _____

Figure 2.3: Receipt school fees payment

2.1.2.4 Hostel fees form

Figure 2.4 shows receipt hostel fees payment. It contain list of information for hostel fees payment.

RECEIPT HOSTEL FEES PAYMENT

Ministry of Education and Higher Education
Republic of Indonesia

No. 00214

School Name: _____

Student Name: _____

No	Description	Amount
1	Registration Fee	
2	Library Fee	
3	Student Activity Fee	
4	Student Health Fee	
5	Student Insurance Fee	
6	Student Uniform Fee	
7	Student Transportation Fee	
8	Student Other Fee	
9	Student Other Fee	
10	Student Other Fee	

Total: _____

Signature: _____

Date: _____

Figure 2.4: Receipt Hostel fees payment

2.1.3 Existing System

In these section will be describe about the existing system that have been implement and use at other school.

2.1.3.1 School Accounting Management System

School Accounting Management System is a system that have been develop according to the school accounting that have been manage by department of internal audit, Minister of Education Malaysian. By developing these system can help KPM management get the information, observation, monitor and evaluate financial performance at each school. Data or information of expenditure and receipts for each month need to be updated to these system or before 5th each month after document and financial record have been updated, balanced and closed according to the rule that have been state. In this section will be show several description about these system will be explain in this section. Refer to **APPENDIX B** to get detail more information about School Accounting Management System.

2.1.3.1.1 Login

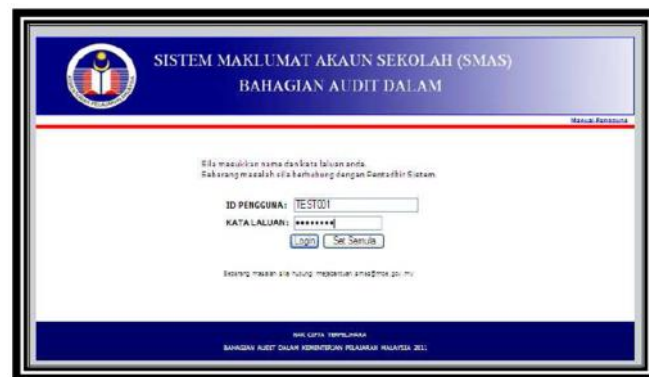


Figure 2.7: Login Interface

This figure 2.7 shows that first homepage for login these system. For the user these system need to enter their username and password. Then click button 'LOGIN'.

2.1.3.1.2 Change Password



Figure 2.8: Change Password Interface

This figure 2.8 show that change user password interface. For the user that want to change their password or forget their current password and want to make new password.

2.1.3.1.3 Update profile



Figure 2.9: Update Profile Interface

This figure 2.9 show that update profile user information. For the user that want to change their profile information.

2.1.3.1.4 The balance information PCG

MAKLUMAT BAKI BANTUAN DERIAH PER KAPITA / PER CAPITA DERIAH (PCG)	
Silalahitkan yang hendak diinput.	
Tahun : 2011 <input type="text" value="01"/>	
MAKLUMAT BAKI BANTUAN DERIAH PER KAPITA / PER CAPITA DERIAH (PCG)	
BAKID00-01-0011	RM 11,300.00
BAKIDUKUTUALAJOR BULAN	
JANUARY 2011 (RM 00.00)	RM 72,864.00
FEBRUARY 2011 (RM 00.00)	RM
MARCH 2011 (RM 00.00)	RM
APRIL 2011 (RM 00.00)	RM
MAY 2011 (RM 00.00)	RM
JUNE 2011 (RM 00.00)	RM
JULY 2011 (RM 00.00)	RM
AUGUST 2011 (RM 00.00)	RM
SEPTEMBER 2011 (RM 00.00)	RM
OCTOBER 2011 (RM 00.00)	RM
NOVEMBER 2011 (RM 00.00)	RM
DECEMBER 2011 (RM 00.00)	RM

Figure 2.10: Balance information PCG

This figure 2.10 shows that PCG balance information that display cash balance for each month after PCG have been filled. This homepage balance information about PCG will be display just like the above.

2.1.3.1.5 The balance information BMA

MAKLUMAT BAKI BANTUAN DERIAH PER KAPITA / PER CAPITA DERIAH (BMA)	
Silalahitkan yang hendak diinput.	
Tahun : 2011 <input type="text" value="01"/>	
MAKLUMAT BAKI BANTUAN DERIAH PER KAPITA / PER CAPITA DERIAH (BMA)	
BAKID00-01-0011	RM 11,300.00
BAKIDUKUTUALAJOR BULAN	
JANUARY 2011 (RM 00.00)	RM 72,864.00
FEBRUARY 2011 (RM 00.00)	RM
MARCH 2011 (RM 00.00)	RM
APRIL 2011 (RM 00.00)	RM
MAY 2011 (RM 00.00)	RM
JUNE 2011 (RM 00.00)	RM
JULY 2011 (RM 00.00)	RM
AUGUST 2011 (RM 00.00)	RM
SEPTEMBER 2011 (RM 00.00)	RM
OCTOBER 2011 (RM 00.00)	RM
NOVEMBER 2011 (RM 00.00)	RM
DECEMBER 2011 (RM 00.00)	RM

Figure 2.11: Balance information BMA

This figure 2.11 shows that BMA balance information that display cash balance for each month after BMA have been filled. This homepage balance information about BMA will be display just like the above.

2.1.3.1.6 Homepage Debit PCG

The screenshot shows the 'Maklumat Terimaan PCG Tahun 2011' interface. At the top, it says 'Maklumat Terimaan PCG Dan Lain Terimaan (PCG: Masa Pelajaran / Baki Masa Pelajaran / LPT: Pra-Debit / Voucher Khas)'. Below this, it asks 'Silalah pilih tahun yang hendak dilihat' with a dropdown menu set to '2011'. A box labeled 'n Maklumat tentang Pilih.' points to the year selection. The main table shows the following data:

	PCG	SBT	RMT	KWAPM	Pakaian Seragam	Lain Terimaan	Jumlah
Baki 1.1.2011	10,000.00	200.00	300.00	400.00	400.00	200.00	11,500.00
Januari	1,234.00	0.00	100.00	0.00	0.00	0.00	1,334.00
Februari	200.00	0.00	0.00	0.00	0.00	0.00	200.00
Mac							
April							
Mei							
Jun							
Julai							
Ogos							
September							
Oktober							
November							
Disember							
Jumlah	11,434.00	200.00	400.00	400.00	400.00	200.00	13,234.00

A red box contains the text: 'Maklumat di ruangan ini akan dipamerkan secara automatik setelah maklumat Terimaan PCG dikunci masuk dan disimpan.'

Figure 2.12: Debit PCG interface

This figure 2.12 shows that debit PCG information will be lock by user start on date 1.1.2011 balance.

2.1.3.1.7 Information Sheet Balance Receipts PCG 1.1.2011

The screenshot shows the 'Maklumat Terimaan PCG Baki 1.1.2011' interface. It contains input fields for the following items:

- PCG: RM 10000.00
- SBT: RM 200.00
- RMT: RM 300.00
- KWAPM: RM 400.00
- Pakaian Seragam: RM 400.00
- Lain Terimaan: RM 200.00

A red box labeled 'Terimaan hujuk kepada Vang' points to the input fields. At the bottom, there are three buttons: 'Simpan', 'Set Semula', and 'Batal'. A red box labeled 'Selesai proses kunci masuk data / maklumat, Klik butang Simpan.' points to the 'Simpan' button.

Figure 2.13: Sheet Balance Receipts PCG 1.1.2011

This figure 2.13 shows that balance money that have been insert and all the information will be save in the database.

2.1.3.1.8 PCG payment paperwork

[illegible]

Figure 2.14: PCG payment paperwork interface

This figure 2.14 shows that these information will be display in automatic way after all PCG payment information have been lock in and stored.

2.1.3.1.9 PCG payment balance sheet information

[illegible]

Figure 2.15: PCG payment balance sheet information interface

This figure 2.15 shows that user need to key in in this page as the payment for PCG balance 1.1.2011

2.1.3.1.10 Paperwork Debit BMA

The screenshot displays a web interface titled "Maklumat Terimaan BMA Dan Lain Terimaan". It includes a dropdown menu for the year, currently set to "2011". Below this is a table titled "Maklumat Terimaan BMA Tahun 2011". The table has columns for "LPBT", "Bantuan Makanan", "Yuran Makanan", "Yuran Pendaftaran", "Yuran Dobi", "Pelbagai", "Lain Terimaan", and "Jumlah". The rows represent the months from January to December, followed by a "Jumlah" row. A red box highlights the "Jumlah" column header with the text "Maklumat di ruangan ini akan dipamerkan secara automatik setelah maklumat Terimaan BMA dikunci masuk dan disimpan." Another red box points to the "Jumlah" column header with the text "Maklumat di ruangan ini akan dipamerkan secara automatik setelah maklumat Terimaan BMA dikunci masuk dan disimpan."

Figure 2.16: Paperwork Debit BMA interface

This figure 2.16 shows that this data information will be display on automatic after BMA debit information have been lock in and stored.

2.1.3.1.11 BMA receipts balance information

The screenshot displays a web interface titled "Maklumat Terimaan BMA Baki 1.1.2011". It features a list of receipt types on the left: "LPBT:", "Bantuan Makanan:", "Yuran Makanan:", "Yuran Pendaftaran:", "Yuran Dobi:", "Pelbagai:", and "Lain Terimaan (Tidak Perlu Isi):". To the right of each type is an input field labeled "RM". A red box highlights the input fields with the text "masukkan Jumlah Terimaan a ruangan yang telah :diakan dengan merujuk ada Buku Tunai Kumpulan ng Asrama". At the bottom, there are three buttons: "Simpan", "Set Semula", and "Batal". A red box points to the "Simpan" button with the text "Selesai proses kunci masuk data / maklumat. klik butane Simpan."

Figure 2.17: BMA receipts balance information interface

This figure shows that user need to insert data that have been use during past month.

2.2 Propose System

School Fees Payment and Accounting Management System is the solution to solve this matter by transform form manual way into current technology. School Fees Payment and Accounting Management System can generate form such as PIBG fees, school fees, hostel fees form and etc. All the description system flow processes in details are refers to **APPENDIX D**, Software Requirement Specification (SRS).

2.3 Comparison between existing school payment in manual and purpose system

In this section will describe comparison between existing manual system and purpose system below.

Table 2.1: Comparison between existing school payment in manual and purpose system

No	Features	Existing manual system	Purpose system
1.	Setup fees form	Using Microsoft Office by typing one by one list of fees and state fees payment for each different type of fees.	This system provides generate fees form by insert list of fees and state fees payment. Then, system will be generating fees form after click button setup.
2.	Fill form student information	Teacher need to write student information for each different type of fees.	This system will be display student information in different

			type of fees.
3.	Calculate fees	Teacher need calculator to help them to calculate total amount.	This system will provide auto calculation and generate automatic get total amount
4.	Record of current student done make fees payment	Sometimes teacher forget for updated current status fees payment and it will causes student are suspected not paid.	This system will save all record in database. If teacher want to view current status fees payment, teacher just need click view then system will be generate current status fees payment.
5.	Receipt	Copy of receipt is not clear and sometimes the word on the copy of receipt is blurring.	System will be print and show clear each words on copy of fees receipt.
6.	Account record	Teacher need to manage accounting record by calculate debit, credit and etc.	System will be generated automatic and display after teacher done key in for debit, credit and etc.

2.4 Data Flow

Definition of Data flow diagrams (DFDs) is reveal relationships between the various components in a system [4]. DFDs are an important technique for modeling a

system's by showing input data is transformed and get output results through a sequence of functional transformations.

DFDs consist of four major components which are entities, processes, data stores, and data flows. The symbols used is to show that how these components interact in a system are simple and easy to understand. DFD have their own syntax which is remain constant by using simple verb and noun constructs. In this section will be describe more the concept of Data Flow Diagrams in more detail.

2.4.1 Defining data flow diagrams (DFDs)

Data flow diagrams (DFDs) contain three principal reasons which are:-

- i. Technical and non-technical audiences are easier to understand using DFD model.
- ii. Provide a high level system overview, complete with boundaries and connections to other systems.
- iii. Provide detailed representations of system components.

By using DFDs help system designers and others understanding of the boundary between existing system and postulated systems during initial analysis stages.

DFDs represent the following:-

- i. External devices sending and receiving data.
- ii. Processes that change that data.
- iii. Data flow themselves.
- iv. Data storage locations.

The hierarchical DFD typically consists of :-

- i. A top-level diagram (Level 0) - underlain by cascading lower

- ii. Level diagrams (Level 1, Level 2...) - represent different parts of the system.

2.4.1.1 Data Flow Diagrams

DFD illustrates those functions that must be performed in a program as well as the data that the functions will need [4]. A DFD is illustrated in Figure 2.20 shows example data flow diagrams.

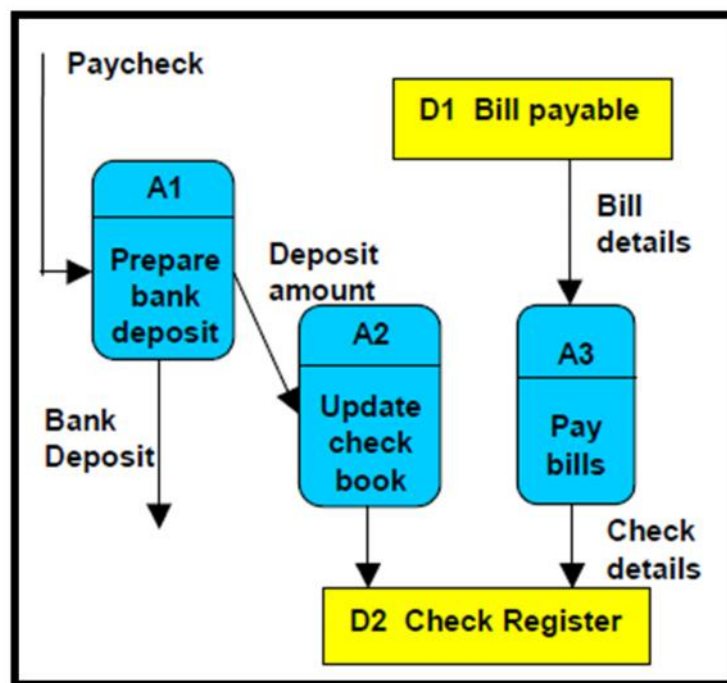


Figure 2.20: Data flow diagram

2.4.2 Defining DFD components

DFDs consist of four basic components which are entity, process, data store, and data flow.

2.4.2.1 Entity

An entity is the source or destination of data. Provide data to the system (referred to as a source) or receive data from it (referred to as a sink). Entities are represented as rectangles (a diagonal line across the right-hand corner means that this entity is represented somewhere else in the DFD) [4].

2.4.2.2 Process

The process is the manipulation or work that transforms data, performing computations, making decisions (logic flow), or directing data flows based on business rules. For example, a process receives input and generates some output. Processes can be drawn as circles or a segmented rectangle on a DFD, and include a process name and process number [4].

2.4.2.3 Data Store

A data store is where a process stores data between processes for later retrieval by that same process or another one. Data stores are usually drawn as a rectangle with the right hand side missing and labeled by the name of the data storage area it represents, though different notations do exist [4].

2.4.2.4 Data Flow

Data flow is the movement of data between the entity, the process, and the data store. Data flow portrays the interface between the components of the DFD. The flow of data in a DFD is named to reflect the nature of the data used (these names should also be unique within a specific DFD). Data flow is represented by an arrow, where the

arrow is annotated with the data name [4]. These DFD components are illustrated in Figure 2.21 shows DFD components.

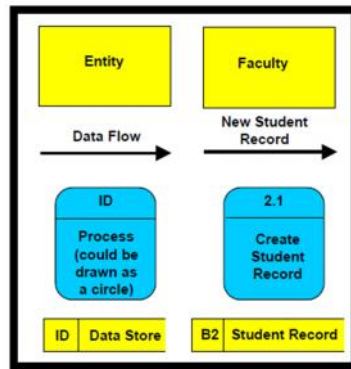


Figure 2.21: DFD components

2.4.3 Process for Developing DFDs

First, start with begin by making a list of business activities to determine the DFD elements (external entities, data flows, processes, and data stores). Next, start to draw a context diagram which is shows only a single process (representing the entire system), and associated external entities. The Diagram-0, or Level 0 diagram, is next, which reveals general processes and data stores. Following the drawing of Level 0 diagrams, child diagrams will be drawn (Level 1 diagrams) for each process illustrated by Level 0 diagrams.

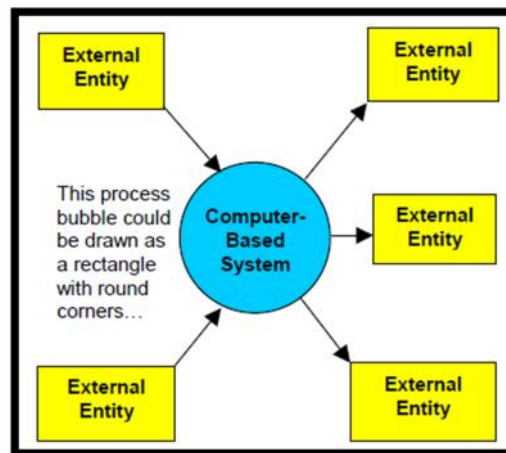


Figure 2.22: Level 0 diagrams

2.4.3.1 A simple DFD example

Figure 2.23 show that the 'Context Diagram' is an overall, simplified, view of the target system, which contains only one process box and the primary inputs and outputs.

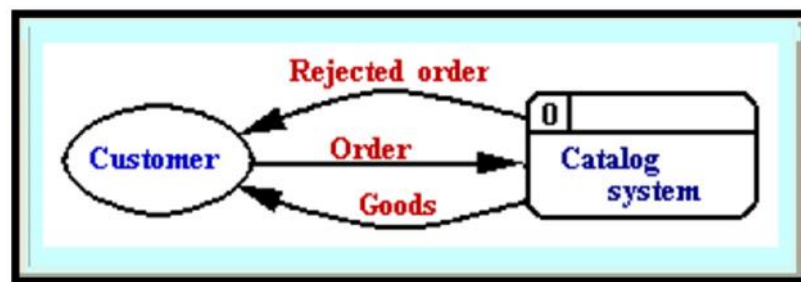


Figure 2.23: Context Diagram

2.4.3.2 Level 1 DFD

Figure 2.24 shows that level 1 DFD is describing the whole of the target system.

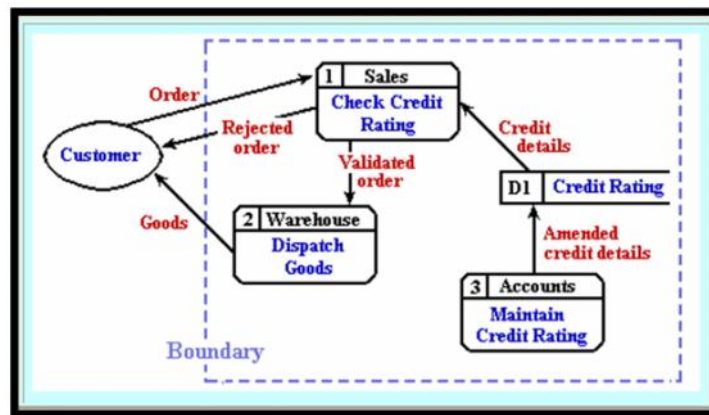


Figure 2.24: Level 1 DFD

2.5 Business Process Modelling

According to Sandy Kemsley said that Business process modeling can be defined as a management practical for managing business process [5]. Another definition is a set of logically related tasks performed to achieve a defined business outcome. A process is a structure, measured set of activities designed to produce a specific output for a particular customer or market [6].



Figure 2.25: Business processes as transformations of inputs to outputs

2.5.1 Cross Functional Flow Charts

For developing Business Process Modeling it will use cross functional flowcharts as a drawer to describe about the system process. Cross functional process show, as their name indicates, the relationship between a business process and the


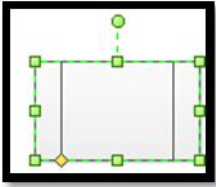
organizational and functional units, such as departments, that are responsible for steps in that process.


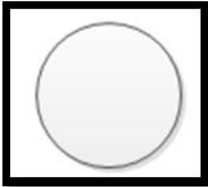


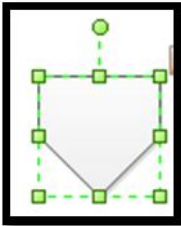
Cross Functional Flowcharts focus on showing "who does which activity", the actual process flow and identifies the people or groups involved at each step [5]. It is displaying process flows across organizational boundaries and identifying delays, repetitive steps, excessive control points, specialized tasks, and potential points of process failure.


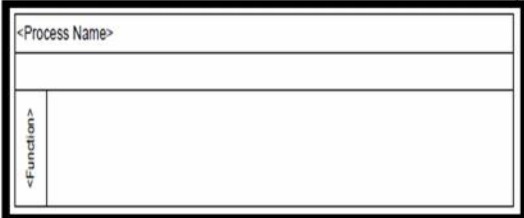
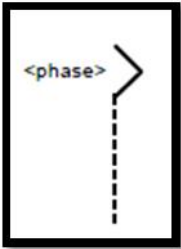
2.5.1.1 Standard Cross Functional Flowchart symbols and usage

Flowcharts use special shapes to represent different types of actions or steps in a process. Lines and arrows show the sequence of these steps, and the relationships between them. Table 2.2 below shows a description about Standard Functional Flowcharts symbol that use in for Business Process Modeling process.

Table 2.2: Standard Cross Functional Flowchart symbols

No.	Cross Functional Flowchart symbols	Figure
1.	Process represents a step in your process.	
2.	Predefined process indicates a set of steps that combine to create a sub-process that is defined elsewhere, often on another page of the same drawing.	

3.	Decision indicates a point where the outcome of a decision dictates the next step. There can be multiple outcomes, but often there are just two - yes and no.	
4.	Terminal points indicate the starting and ending points of a process.	
5.	Arrows and connecting lines diagram the logical progression through the course, subject to the choices made at decision or action points within the process.	
6.	The input/action symbol represents a user response that directs the course flow from that point onwards, i.e., an online test, or questionnaire form.	
7.	Off-page reference Use the set of hyperlinks between two pages of a flowchart or between a sub-process shape and a separate flowchart page that shows the steps in that sub-process.	
8.	Document represents a step that	

	results in a document.	
9.	Functional band represents the functional units. Shapes representing steps in the process are placed in bands that correspond to the functional units responsible for those steps.	
10.	Separators, which indicate phases within a particular process.	

2.5.1.2 Cross Functional Flowchart Process

A cross functional process map displays the tasks and decisions that are carried out during a process by all the contributing workers. Process maps can be inspected visually, they can be analyzed with the value added criteria, and they can be modeled with software. The complexity of process analysis tools varies greatly. Process drawing tools automate the documentation of the flow of task steps in a single process; process modeler tools capture cost and time data to provide simulations of 'as is' and 'to be' process models for the enterprise.

Figure 2.26 show that a cross functional process map displays all the workers that contribute to a process, even though they may work in different departments and apply different types and levels of skills.

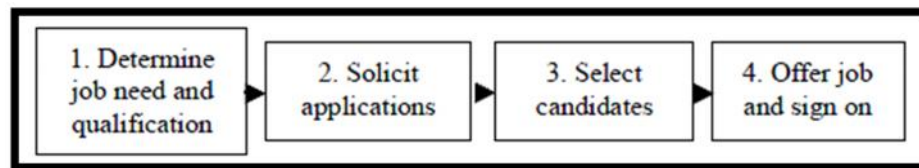


Figure 2.26: Flow process

Each of the high level tasks is then expanded, as shown in the cross functional map in Figure 2.27.

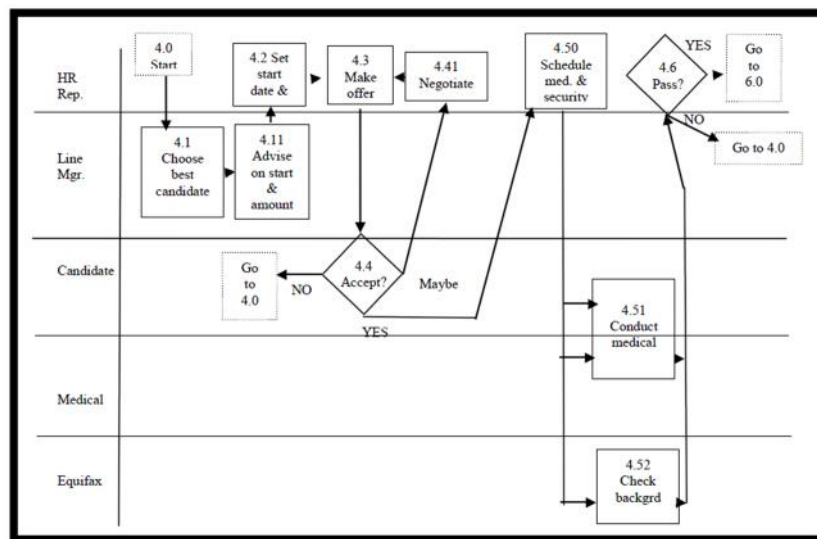


Figure 2.27: Cross functional map

2.6 Unified Modelling Language

Unified Modeling Language (UML) is a standardized general-purpose modeling language in the field of object-oriented software engineering. UML includes a set of graphic notation techniques to create visual models of object-oriented software-intensive systems. UML combines techniques from data modeling (entity relationship diagrams), business modeling (work flows), object modeling, and component modeling. It can be used with all processes, throughout the software development life cycle, and across different implementation technologies [7]

2.6.1 Use Cases

A use case shows the behavior or functionality of a system [8]. It consists of a set of possible sequences of interactions between a system and a user in particular environments that are related to a particular goal. A use case model consists of actors and use case.

2.6.1.1 Use Case Diagram

A use case diagram is a picture showing system behavior along with the key actors that interact with the system [8].

2.6.1.2 Definitions and symbols

Use case diagramming is relatively simple in that it involves only a few symbols. These few symbols can be used to represent quite complex situations. The key symbols in a use case diagram are explained in table 2.3 below.

Table 2.3: Symbols and definitions

No.	Symbols	Definitions
1.	Actor	An actor is involved with the functioning of a system at some basic level. Actors are represented by stick figures.
2.	Use case	Each use case is represented as an ellipse. Each use case represents a single system function. The name of the use case can be listed inside the ellipse or just below it.
3.	System boundary	The system boundary is represented as a box that includes all of the relevant use case.
4.	Connections	The actors are connected to use case with lines, and that use case is connected to each other with arrows.
5.	Extended relationship	An association between two use cases where one adds new behaviors or actions to the other.
6.	Included relationship	An association between two use cases where one use cases the functionality contained in the other.

2.6.2 Sequence Diagram

A sequence diagram depicts the interactions among objects during certain period of time [8]. Because the pattern of interactions varies from one use case to another, each sequence diagram shows only the interactions pertinent to a specific use case. It

shows the participating objects by their lifelines and the interactions among those objects arranged in time sequence by the messages they exchange with one another.

2.6.2.1 Symbols and definitions

The key symbols in a sequence diagram are explained in table 2.4 below.

Table 2.4: Symbols and definition for sequence diagram

No.	Symbols	Definitions
1.	Vertical dashed line	Lifeline represents the object's existence over a certain period of time.
2.	Object symbol	A box with the object's name underlined is placed at the head of each lifeline.
3.	An activation	Activation shows the time period during which an object performs an operation, either directly or through a call to some subordinate operation.
4.	Synchronous message	Represents as a full, solid arrowhead, is one where the caller has to wait for the receiving object to complete execution of the called operation before it can resume execution.
5.	Simple message	Represent a simply transfer control from the sender to the recipient without describing the details of the communication.
6.	Asynchronous	Represent as a half arrowhead in a

	message	sequence diagram, is one where the sender does not have to wait for the recipient to handle the message.
--	---------	--

2.6.3 Activity Diagram

An activity diagram shows the conditional logic for the sequence of system activities needed to accomplish a business process [8]. An individual activity may be manual or automated and often represent the actions needed to move an object between states. Further activity is the responsibility of a particular organizational unit.

2.6.3.1 Symbols and description

The key symbols in a use case diagram are explained in table 2.5 below.

Table 2.5: Symbols and description for Activity Diagram

No.	Symbols	Description
1.	swim lane	Represents the organizational unit responsible for certain activities.
2.	Start	The process starts at the larger dot.
3.	Fork	Represent several parallel, independent sequences of activities are initiated.
4.	Activity	Represent by rectangles with rounded corners.
5.	Join	Represent independents streams of activities now must all reach completion to move on to the next step.
6.	Branch	Represents indicates conditional logic.

2.6.4 Class Diagram

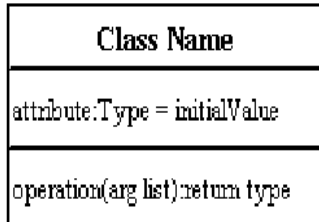
Class diagram is a diagram that shows the static structure of an object-oriented model: the object classes, their internal structure, and the relationships in which they participate [8].

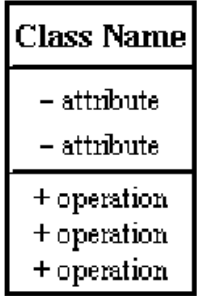
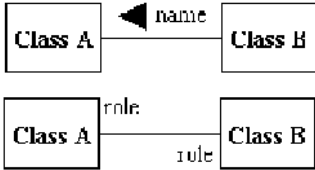
2.6.4.1 Basic class diagram symbols and notation

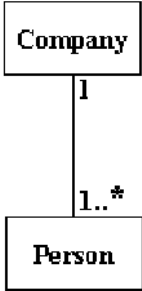
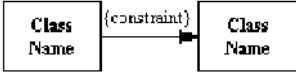
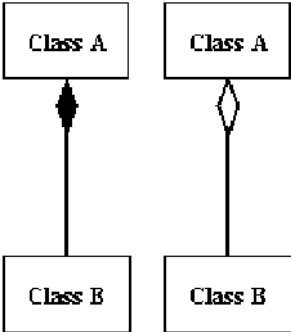
Classes represent an abstraction of entities with common characteristics.

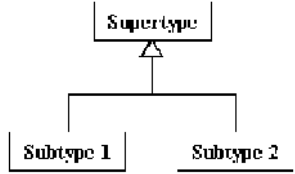
Associations' represent the relationships between classes that will explain in table 2.6 below.

Table 2.6: Symbols and description for class diagram

No.	Symbols	Description	Figure
1.	Classes	Illustrate classes with rectangles divided into compartments. Place the name of the class in the first partition (centered, bolded, and capitalized), list the attributes in the second partition, and write operations into the third.	
2.	Visibility	Use visibility markers to signify who can access the information contained within a class. <ul style="list-style-type: none"> • Private visibility 	

		<p>hides information from anything outside the class partition.</p> <ul style="list-style-type: none"> • Public visibility allows all other classes to view the marked information. • Protected visibility allows child classes to access information they inherited from a parent class. 	
3,	Associations	<p>Associations represent static relationships between classes. Place association names above, on, or below the association line.</p> <p>Use a filled arrow to indicate the direction of the relationship. Place roles near the end of an association. Roles represent the way the two classes see each other.</p>	
4.	Multiplicity	Place multiplicity	

	(Cardinality)	notations near the ends of an association. These symbols indicate the number of instances of one class linked to one instance of the other class.	 <pre> classDiagram class Company class Person Company "1" -- "1..*" Person </pre>
5.	Constraint	Place constraints inside curly braces {}.	 <pre> classDiagram class "Class Name" class "Class Name" "Class Name" -- "Class Name" : {constraint} </pre>
6.	Composition and aggregation	<p>Composition is a special type of aggregation that denotes a strong ownership between Class A, the whole, and Class B, its part. Illustrate composition with a filled diamond.</p> <p>Use a hollow diamond to represent a simple aggregation relationship, in which the "whole" class plays a more important role than the "part" class, but the two classes are not dependent on each</p>	 <pre> classDiagram class "Class A" class "Class B" "Class B" -- "Class A" : composition (filled diamond) "Class B" -- "Class A" : aggregation (hollow diamond) </pre>

		<p>other.</p> <p>The diamond end in both a composition and aggregation relationship points toward the "whole" class or the aggregate.</p>	
7.	Generalization	<p>Generalization is another name for Inheritance or an "is a" relationship. It refers to a relationship between two classes where one class is a specialized version of another.</p>	 <pre> classDiagram class Supertype class Subtype1["Subtype 1"] class Subtype2["Subtype 2"] Supertype < -- Subtype1 Supertype < -- Subtype2 </pre>

2.7 Comparison between Data Flow Diagram, Business Process Modeling and Unified Modeling Language

In this section will be describe comparison modeling between Data Flow Diagram, Business Process Modeling and Unified Modeling Language that will shown on table 2.7 below.

Table 2.7: Comparison between Data Flow Diagram, Business Process Modeling And Unified Modeling Language

	Data Flow Diagram	Business Process Modeling	Unified Modeling Language
Definition	<ul style="list-style-type: none"> Conceptual 	<ul style="list-style-type: none"> Diagrammatic 	<ul style="list-style-type: none"> UML is a

	representation of data objects and associations among data objects in a database.	representation of a sequence of activities showing events, actions and connection points in the sequence	standardized general-purpose modeling language in the field of object-oriented software engineering
Main type of process	<ul style="list-style-type: none"> Entity-Relationship diagram (widely known as ER diagram), which is a pictorial representation of data objects and interactions between them. This is valuable because it could be easily learned and could be used to communicate with end users. Data document that describes data objects, 	<ul style="list-style-type: none"> First one is the 'as is' or baseline model that displays the current situation. This model could be used to identify weak points and bottlenecks, which could be useful for future improvements. The other model is the 'to be' model, which represents the 	<ul style="list-style-type: none"> The Unified Modeling Language (UML) is used to specify, visualize, modify, construct and document the artifacts of an object-oriented software-intensive system under development. (Chapter 11: Data Flow Diagrams , n.d)

	relationships among data objects, and rules required by the database. This is used by the database developer to develop the database.	intended new situation. This incorporates identified potential improvements from the base line model and could be used to demonstrate and test the new process before actually implementing it.	<ul style="list-style-type: none"> UML combines techniques from data modeling (entity relationship diagrams), business modeling (work flows), object modeling, and component modeling.
Methods	<ul style="list-style-type: none"> Using element of object oriented analysis. 	<ul style="list-style-type: none"> Detail about system process flow work. 	<ul style="list-style-type: none"> Using element of object oriented analysis and design.
Syntax	<ul style="list-style-type: none"> Using employing English noun or noun-adjective-verb constructions 	<ul style="list-style-type: none"> Using employing English verb constructions. 	<ul style="list-style-type: none"> Using employing English verb construction.

2.8 Conclusion

From the beginning, there are discussing about the existing system and proposed system that will be developed to replace the existing manual system. Then, it also

contain a description about the modeling between Data Flow Diagram, Unified Modeling Language and Business Process Modeling in detail by giving description each modeling and including their flow process. After that, comparison between Data Flow Diagram, Unified Modeling Language and Business Process Modeling to choose for the best process modeling system that will be use together with proposed system. From the comparison between modeling. Unified Modeling Language is the good choice to make as our system process design architecture in more detail, understanding the flow process and the important thing is familiar to use.

CHAPTER 3

METHODOLOGY

This chapter briefly discusses about the methodology and the procedures of School Fees Payment and Accounting Management System by implement methods and techniques used, and hardware and software specification of the system. The software development process is done by using iterative and incremental development. The justification of the chosen methodology will be discussed in detail in the next section while the stages of the iterative and incremental development methodology will be discussed in detail in Section 3.2. Besides that, this chapter depicts all the requirements of School Fees Payment and Accounting Management System.

3.1 Iterative and Incremental Development

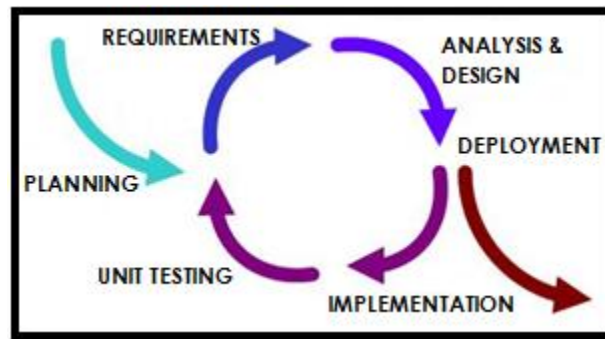


Figure 3.1: Iterative and incremental development

Iterative and incremental software development is a method of software development that is modeled around a gradual increase in feature additions and a cyclical release and upgrade pattern [9].

Iterative and incremental software development begins with planning and continues through iterative development cycles involving continuous user feedback and the incremental addition of features concluding with the deployment of completed software at the end of each cycle.

3.1.1 Definition

Incremental development is a staging and scheduling strategy in which various parts of the system are developed at different times or rates and integrated as they are completed. The alternative strategy to incremental development is to develop the entire system with big-bang integration at the end [10].

Iterative development is a rework scheduling strategy in which time is set aside to revise and improve parts of the system. The alternative strategy to iterative development is to plan to get everything right the first time [10].

3.1.2 Incremental Development

In this section will be discuss on more about incremental development by using the simulation figure how the incremental development methods is work. Start with by break up the work into smaller pieces and schedule them to be developed over time and integrated as they are completed. Figures 3.2 - 3.4 shows the illustrate of this procedure by imagine that the top sheet of blocks represents various user interface components, the middle sheet represents middleware, and the bottom sheet represents back end or database components.

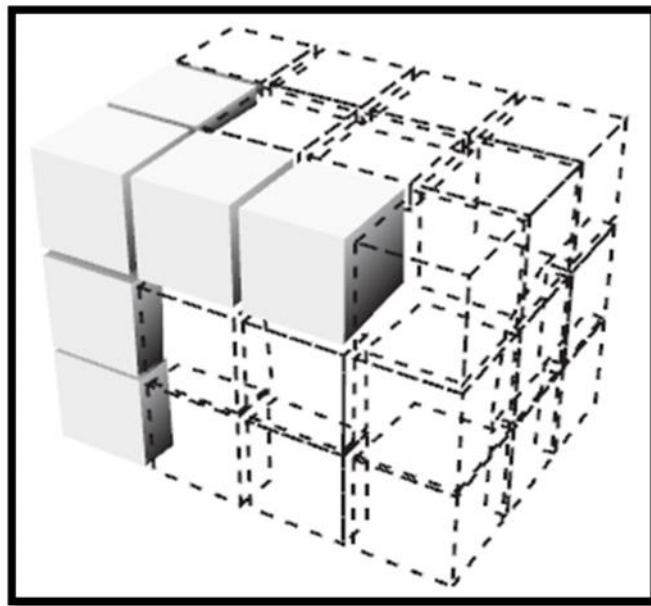


Figure 3.2: Incremental development stage 1

Figure 3.2 shows that in the first increment, a full piece of functionality is built from the user interface (UI) through to back end.

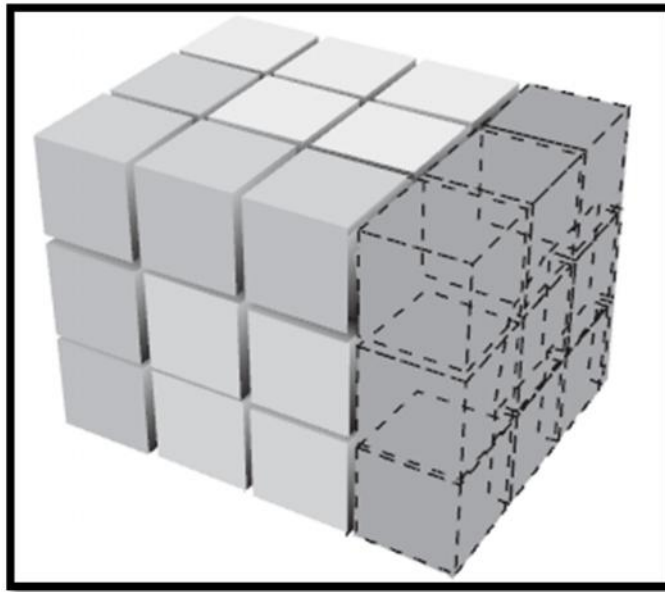


Figure 3.3: Incremental development stage 2

Figure 3.3 shows that the additional functionality is added across all layers of the system. This may be a sufficient point to deploy the system as it is so far to real users and start accruing business benefit.

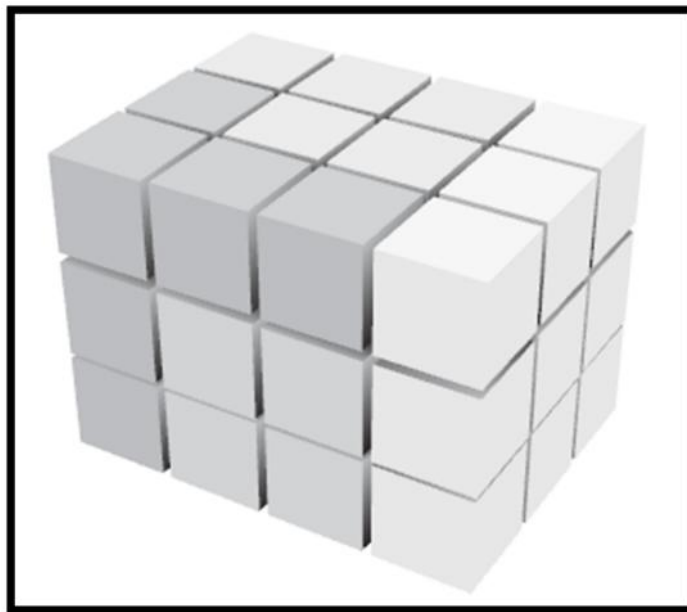


Figure 3.4: Incremental development stage 3

Figure 3.4 shows that the rest of the system is completed and incremented. The pattern just described is the one used by most modern projects whether it is Agile or not.

3.2 Phase

Incremental and Iterative development (IID) slices the system functionality into increments. It contains 6 phases which are planning, requirement, analysis and design, implementation, unit testing and deployment that will be described more in detail for each phase and activities that happen during that phase.

- **Planning** - Planning phase begins with developing a Software Development Plan (SDP) and other planning documents. Provide the basis for acquiring the resources needed to achieve a solution including Gantt Chart.
- **Requirement** - Requirement phase begins when project team along with the customer makes a detailed list of user requirements. The document which contains all this information is called Software Requirement Specification (SRS) documentation.
- **Analysis and Design** – Analysis and Design phase begins using SRS documentation as input, system design is done. The design process translates requirements into representation of the software that can be assessed for quality before generation of code begins.
- **Implementation** - Implementation phase begins with start to implement system design, code generation begins.
- **Unit Testing** – Unit Testing phase starts to integrate a system as it is built. So a complete software is at hand which is tested to check if it meets the functional and performance requirements of the customer.
- **Deployment** – Deployment phase begins by product release is created and distributed to users.

3.2.1 Planning

The planning phase for this project is to define the project that will developed including the scope, objective and planning in School Fees Payment and Accounting Management System. The Gantt chart is use to schedule or plan the activity that will be involved during project development. This project will take semesters to have done. A schedule has been made to guide along the system development to make sure it can be finished at the time given. All the project planning is being done in Gantt chart using Microsoft Project; refer to **Appendix A** for Gantt chart. After this phase is completed, the project will be going into another phase.

3.2.1.1 Development Tools

Any project needs a project environment including project School Fees Payment and Managment System. Whether it will use some of the skill programming techniques or more traditional techniques need to determine the physical resources, software tools and procedures to be follow in developing the system. For developing this system, the software and hardware tools that have been define that will use for developing this system. The specification of the software and hardware for develop School Fees Payment and Managment System showing in table 3.1 and table 3.2.

i. Hardware Requirement

Table 3.1: Hardware Requirements

Item	Quantity	Minimum Requirement	Purpose
Computer	1	AMD Turion X2 Hard Disk 140 RAM 2 GB	To develop the system and present the system

Printer	1	HP Deskjet D2560	Print the proposal and report of project
---------	---	------------------	--

ii. Software Requirement

The software and application specification and purposes of the chosen software for the development have been listed below:-

Table 3.2: Software Facilities and Purposes

SOFTWARE	PURPOSE
Microsoft SQL Server 2005	<ul style="list-style-type: none"> Database for the system; database platform, generate database, and database management
Antivirus Software <ul style="list-style-type: none"> AVG Anti-Virus version 8 free edition 2007 Kaspersky edition 2008 	<ul style="list-style-type: none"> Protection against virus and spy wares Protection against malicious program like viruses, Trojan horses, worms and spy wares
Rational Rose Enterprise Edition	<ul style="list-style-type: none"> Modeling and Designing, UML
Microsoft Windows Operating System <ul style="list-style-type: none"> Windows XP Professional 	<ul style="list-style-type: none"> As a platform for a system to run Operating system which will be used to develop the system

<ul style="list-style-type: none"> Windows 7 Professional 	
<p>Microsoft Office</p> <ul style="list-style-type: none"> Microsoft Word 2007 Microsoft PowerPoint 2007 Microsoft Project 2007 Microsoft Visio 2007 	<ul style="list-style-type: none"> Prepare proposal and documentation Prepare slide for presentation Create scheduling, planning and prepare Gantt Chart Design and draw chart and Diagram
<p>Microsoft Visual Studio 2008</p> <ul style="list-style-type: none"> Visual Basic.net 2008 Macromedia Dreamweaver 8 	<ul style="list-style-type: none"> Design interface and generate coding
WinRAR 8.1 SR-1	<ul style="list-style-type: none"> Compress project files
<p>Web Browser Software:</p> <ul style="list-style-type: none"> Mozilla Firefox Microsoft Internet Explorer 8 Google Chrome 	<ul style="list-style-type: none"> To access internet and search related information on internet

3.2.1.2 Risk Management

Every project on development some kind of system has their risk on managing the system. The major risk has been identified during developing the system. The following are detailed descriptions of the risk that might encounter during the project.

Table 3.3: Risk Management

RISK	POSSIBILITY	MONITORING PROCEDURE	RESPONSIBILITY	PREVENTION METHOD
Power failure	Medium	Provide an Uninterrupted Power Supply (UPS) backup system.	Configuration Manager	Save all necessary file in other portable medium drive, such as external hard-disk, thumb drive or CD/DVD before the backup supply ends.
Lack of support	Low	Manage the schedule more tough	Project Leader and Quality Manager	Give an order to group members straightly.
Damage Documents or Files	Medium	Always be prepared by having backups in other mediums such as other computers or external storage device (softcopy and hardcopy).	Configuration Manager and Developer	Load and print all the saved documents from the backup file and destroy the damaged documents
Hardware Failure	Medium	Always updating the hardware's components such as its anti-virus	Configuration Manager and Developer	Backup all system related directories in external storage devices daily.

Server	High	Get a good server with a good specification and service.	Configuration Manager	Tasks will save into a backup storage.
Corrupted Software	High	Always do the installation of software that need to be used in developing the system correctly	Project Leader and team members	Backup or move the necessary file into other portable media change the development machine.
Choose project model	Medium	Choose the right model for the project	Configuration manager	Identify rightly about what model want to use in the system.
Virus Attack	High	Get a good virus update, scan and protection with the latest version of virus information	Developer and Configuration Manager	Try to repair the files that are infected and do backup of files.

3.2.2 Requirement

The requirement phase is where the project starts to analysis and architecture of the project will be design. In developing a system, it is compulsory to identify the software, hardware or any other resource that will be implementing in the application. In this phase, it more on the research about of the system and identify that how the user can use including hardware design, interface layout, flow of the

process and other documentation. It will be implemented by using Data Flow Diagram that will be shown in figure 3.5. Refer to **APPENDIX D** Software Requirement Specification.

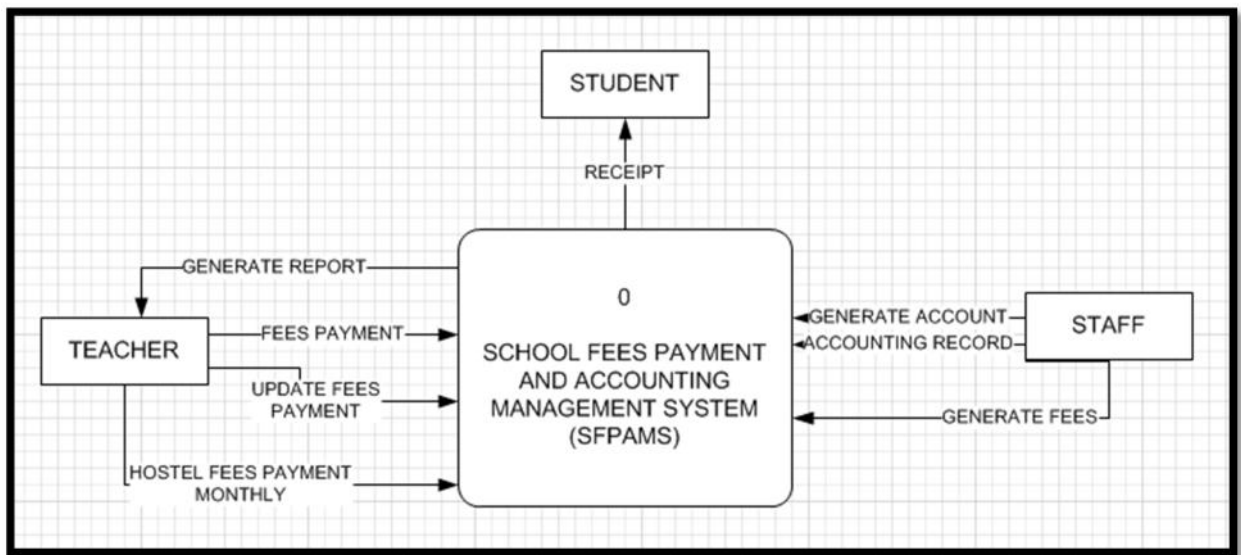


Figure 3.5: Context Diagram for School Fees Payment and Accounting Management System

Refer to the figure 3.5 that contain use case diagram for School Fees Payment and Accounting Management System is mainly developing for staff and teacher that enable to generate report, setup fees, payment, registration, setup account and account record. In use case diagram there are two users which are staff and teacher.

- Setup Fees Form - For generate fees form, staff can setup school fees by adding new list fees state and payment. Besides, they can view after finish setup feed form.
- Payment - For payment form, teacher can view each school fees. Besides, they are enabling to make auto calculation for school fees payment form. Then, they are enabling to print school fees.
- Update Payment - For update payment, teacher can managing fees payment for those student did not paid yet.
- Generate Report - For generate report form, teacher can view report to know who is student did not paid for school fees. Besides, they can update data for those did not pay school fees.
- Login - For login form, staff and teacher will be enter username and password for access the system.

- Setup Account - For generate new account, staff can setup account for create new account.
- Account Record - For account record form, teacher enable to generate account, view and update account record.

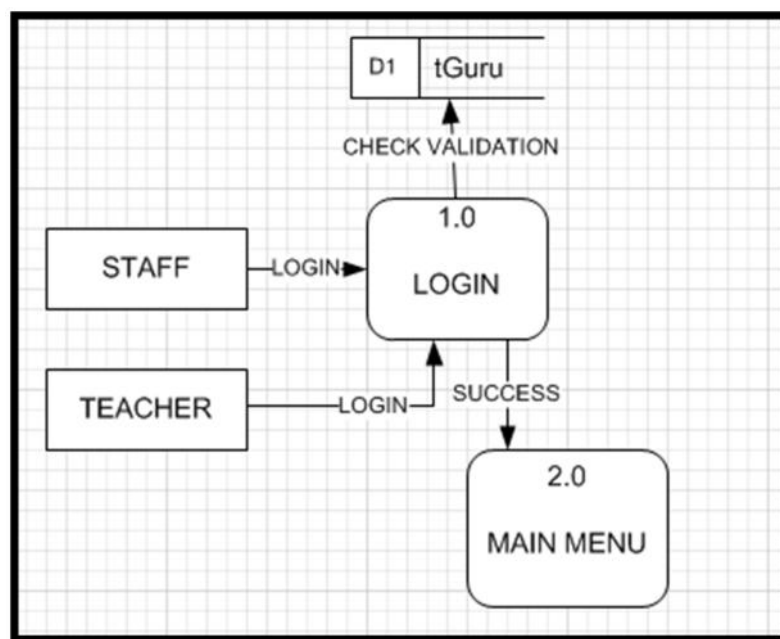


Figure 3.6: Level 1 DFD - Login

Refer to figure 3.6, shows level 1 DFD for login for both users which are staff and teacher. The user need to enter user ID and Password and click button Login. System will check whether user ID and password is valid or not. If valid system will be display Main Menu for different user interface and if not, system will pop up message to tell them invalid user ID and password. They need to re-enter again user ID and password.

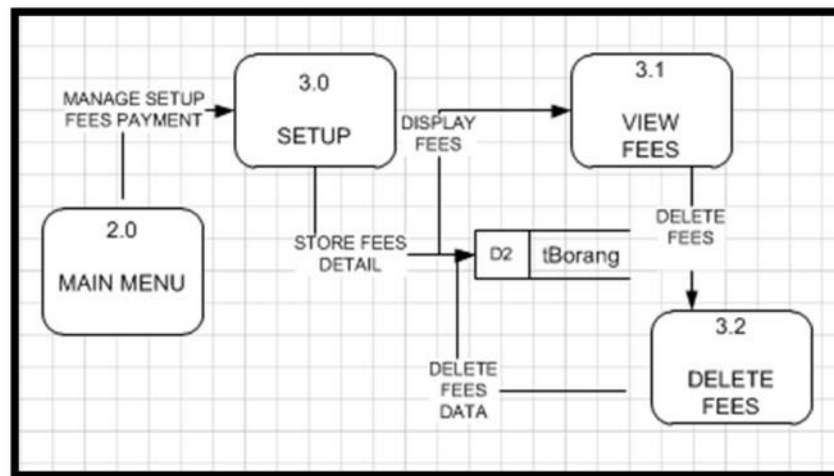


Figure 3.7: Level 1 DFD - Fees Setup

From figure 3.7 shows level 1 DFD for fees setup. This module is use by staff. After passing form login interface, staff click menu generate button and system will be display fees payment interface. Staff only insert list of fees detail with payment and then click generate. System will be save data and will be display back fees list.

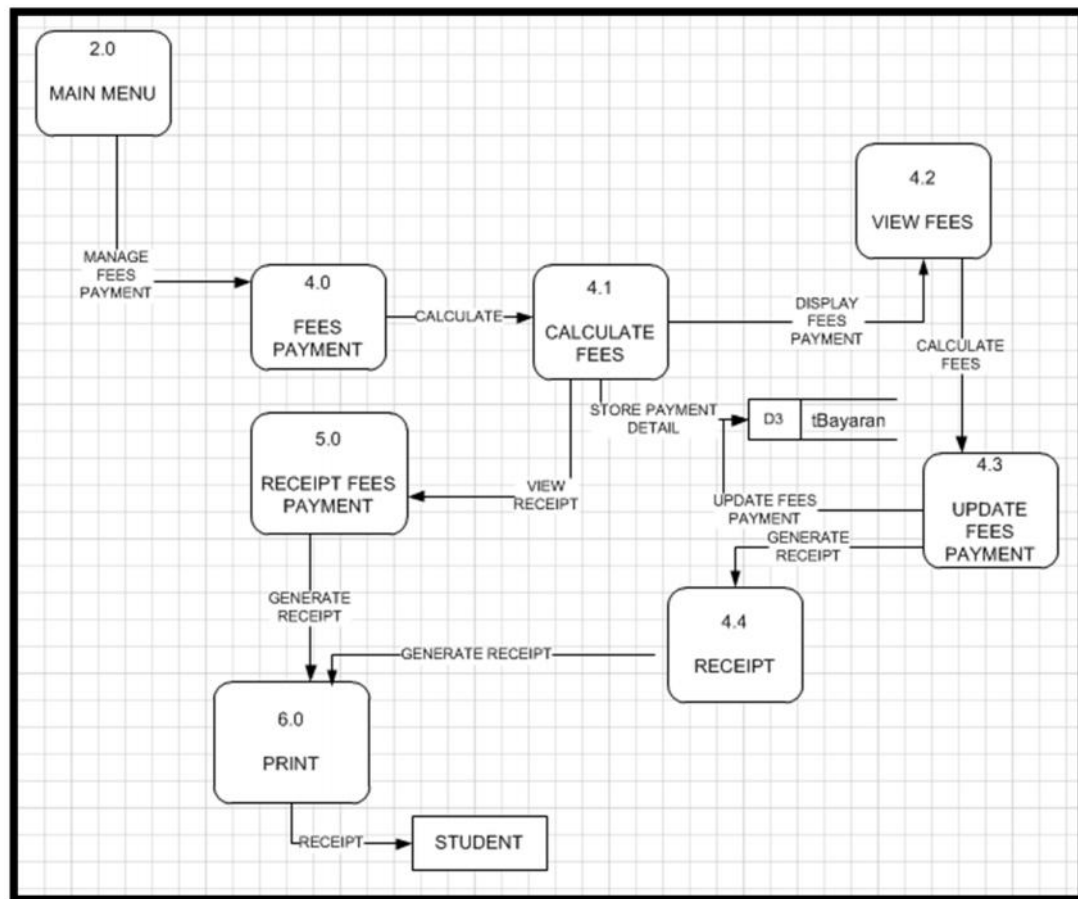


Figure 3.8: Level 1 DFD - Payment

Figure 3.8 show level 1 DFD for payment for school fees payment form. This module is use by teacher. After passing login interface, system will be display list of student according class. Each list of student name have link that contain school fees. Teacher just need click one from link school fees and system will be appearing fee form interface. In the fees form interface will display student information with list of fees that student need paid. Then teacher just need click checkbox for calculate. System will be auto total calculation. All information for paying school fees need to save in the database. Then, teacher will be print fees form interface as receipt to give student along with balance money.

3.2.2.1 Documentation

Refer to **Appendix C** for Software Development Planning (SDP) and **Appendix D** for Software Requirement Specification (SRS).

3.2.3 Analysis and Design

In this phase, the main focus to the development of School Fees Payment and Accounting Management System. It involves several important that are design interfaces, create database and implement coding process. Database that had produced are base on the interface. The interface created need to consider to the user requirement. Before started to develop the coding for the system, a few programs need to be installed in order to run and compile XAMP for this system. For this phase focus on analysis and design.

3.2.3.1 Database Design

A database is an integrated collection of logically related records or files which consolidates records into a common pool of data records that provides data for many applications. For School Fees Payment and Accounting Management System database is very important as a collection of information that is organized so that it can easily be accessed, managed, and updated especially for admin. Refer to figure 3.11 that will describe the database in School Fees Payment and Accounting Management System.

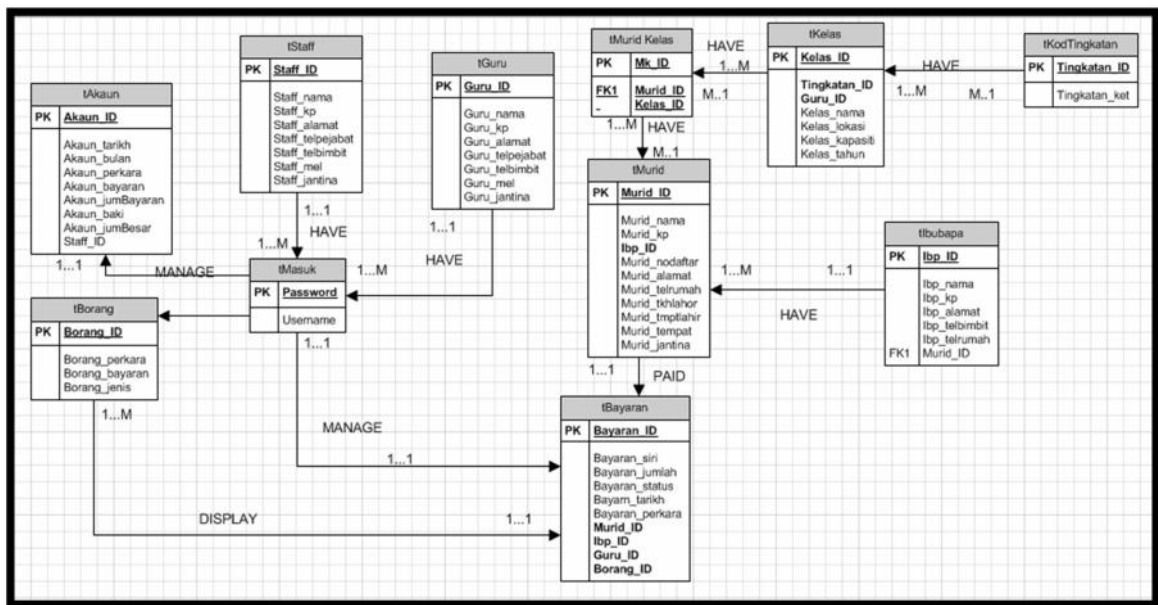




Figure 3.9 : Database ER-Diagram

3.2.3.2 Design

Design is the planning that lays the basis for the making of every object or system and also important for developing this system. It will show several the interface systems in School Fees Payment and Accounting Managment System and it will describe on figure 3.10 till 3.13.


SISTEM PENGURUSAN SEKOLAH MENENGAH AGAMA KERAJAAN NEGERI PAHANG

MODUL PENGURUSAN MAKLUMAT SEKOLAH

[Jumaat, 30 Dis 2011]
ID: 2
NAMA: ABD MUBIN BIN MOHAMED

KEMASKINI
SENARAI NAMA
BORANG
▼ BUKU TUNAI

Tingkatan:1
Kelas:1 CERDAS











No.	Nama Pelajar	Jenis Bayaran		
		PIBG	YURAN	ASRAMA
1	AHMAD YAMINHANIF BIN SIRUN			
2	NADIRAH ADIBA BINTI AFANDI			
3	NIK NUR HUSNA NADIA BINTI NIK SHUHAIRAZI			

Figure 3.10: Fees Payment Interface


SISTEM PENDIDIKAN ISLAM MELAYU MURNI
MODUL PENGURUSAN MAKLUMAT SEKOLAH

NG KE PORTAL MODUL PENGURUSAN MAKLUMAT SEKOLAH
 [Jumaat, 30 Dis 2011] ID: 2 NAMA: ABD MUBIN BIN MOHAMED

KEMAHKINI SENARAI NAMA BORANG • BUKU TUNAI

Maklumat Pelajar

No Siri: 002

Nama: AHMAD YAMINHANIF BIN SIRUN
 Kad Pengenalan: 950120065423
 Kelas: 1 CERDAS
 Tingkatan: 1

Senarai Pembayaran Yuran PIBG

No	Perkara	Bayaran	Tanda
1	YURAN PIBG	RM 30.00	<input type="checkbox"/>
2	YURAN SEKOLAH	RM 30.00	<input type="checkbox"/>
3	YURAN SEKOLAH	RM 30.00	<input type="checkbox"/>
4	YURAN PIBG	RM 30.00	<input type="checkbox"/>

Jumlah Besar: RM 120.00

Ringgit Malaysia:

Maklumat Keluarga Murid

Nama Bapa: ABD MUBIN BIN MOHAMED
 Kad Pengenalan: 950120065423

Nama Ahli Keluarga:-

No.	Nama
1	ABD MUBIN BIN MOHAMED

Maklumat Penerima

Nama Penerima: ABD MUBIN BIN MOHAMED
 Tarikh:

Figure 3.11: Fees Interface





MAKLUMAT TERIMAAN BMA DAN LAIN TERIMAAN									
Sila pilih tahun yang hendak dilihat:									
Tahun: <input type="text" value="-- Sila Pilih --"/> <input type="button" value="CARI"/>									
Maklumat Terimaan BMA Tahun 2012									
	LPBT	BANTUAN MAKANAN	YURAN MAKANAN	YURAN PENDAFTARAN	YURAN DOBI	LAIN TERIMAAN	Jumlah	ISI TERIMAAN BMA	LIHAT
	RM	RM	RM	RM	RM	RM	RM		
JANUARI	2	2	2	2	2	2	12		
FEBRUARI	5	5	5	5	5	5	30		

Figure 3.12: Debit BMA Record

**SEKOLAH MENENGAH AGAMA PULAU TAWAR, 27050 Pulau Tawar, Jerantut,
Pahang**

Tel Sek: 092669342 **Faks: 092671270**

Maklumat Terimaan BMA Baki

Bulan:

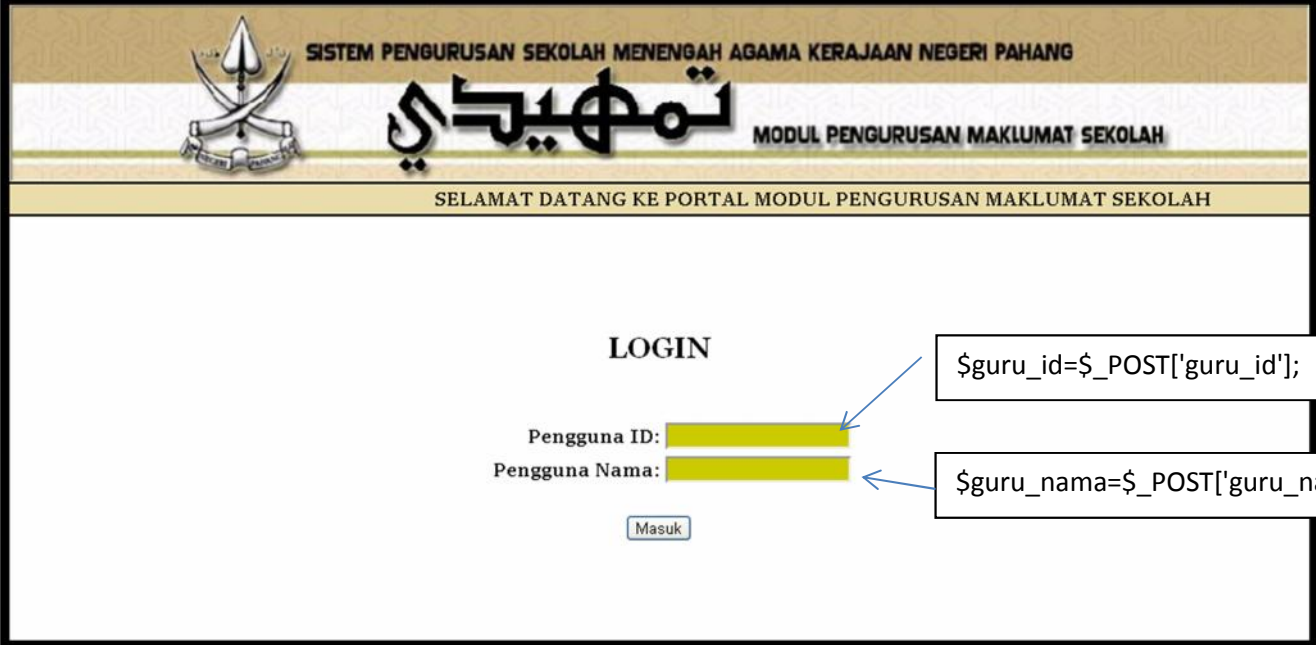
Tahun:

LPBT:	RM:	<input type="text"/>
Bantuan Makanan:	RM:	<input type="text"/>
Yuran Makanan:	RM:	<input type="text"/>
Yuran Dobi:	RM:	<input type="text"/>
Yuran Pendaftaran:	RM:	<input type="text"/>
Lain Terimaan:	RM:	<input type="text"/>
Jumlah:	RM:	<input type="text"/>

Figure 3.13: Balance of Debit BMA Record

3.2.4 Implementation

School Fees Payment and Accounting Management System development process involves several important that are design interface, create database and implement coding process. Database that had produced are based on the interface. The interface created need to consider to the user requirement and refer to figure 3.14 below.



The image shows a login interface for a school management system. At the top, there is a header with a logo on the left and text in Malay and Arabic. The text reads: "SISTEM PENGURUSAN SEKOLAH MENENGAH AGAMA KERAJAAN NEGERI PAHANG" and "MODUL PENGURUSAN MAKLUMAT SEKOLAH". Below this, a banner says "SELAMAT DATANG KE PORTAL MODUL PENGURUSAN MAKLUMAT SEKOLAH". The main content area is titled "LOGIN" and contains two input fields: "Pegguna ID:" and "Pegguna Nama:". A "Masuk" button is located below these fields. Two callout boxes with arrows point to the input fields, containing PHP code snippets: "\$guru_id=\$_POST['guru_id'];" for the ID field and "\$guru_nama=\$_POST['guru_nama'];" for the name field.

SISTEM PENGURUSAN SEKOLAH MENENGAH AGAMA KERAJAAN NEGERI PAHANG

MODUL PENGURUSAN MAKLUMAT SEKOLAH

SELAMAT DATANG KE PORTAL MODUL PENGURUSAN MAKLUMAT SEKOLAH

LOGIN

Pegguna ID:

Pegguna Nama:

Masuk

`$guru_id=$_POST['guru_id'];`

`$guru_nama=$_POST['guru_nama'];`

Figure 3.14: Login Interface

3.2.5 Testing

From the testing, its need to review the some comment that can help to repair and improved the system. It happened because a few function in the system is not finish. Although the system can run overall, a few function like undefined variable in it appear on view catalog page site, if enter the number, the system still can run and also can correct calculate. Refer to figure 3.15

3.2.5.1 Document

Refer to **Appendix E** for detail design description SDD documentation and **Appendix F** for STM documentation

Maklumat Keluarga Murid	
1	
Nama Bapa:	SIRUN BIN HAJI MAMAT
Kad Pengenalan:	890208045388
Nama Ahli Keluarga:-	
No.	Nama
1	AHMAD YAMINHANIF BIN SIRUN

Supposedly display another sibling name but only can display owner PIBG fees.

Figure 3.15: PIBG Fees Payment

3.2.6 Deployment

Deployment starts after the code is appropriately tested and approved for release. This may involve installation, customization by setting parameters to the user's values, testing, and possibly an extended period of evaluation. In this phase product release is created and distributed to users. In this case School Fees Payment and Accounting Management System is distributed to religion secondary school to be used by teacher and staff after the testing process done.

3.3 Conclusion

There are different causes that make software development to fail. One of these factors is the poor project management. In this for develop system must try to minimize the factor offailure related to project management by the elaboration of a guide. For this reason, the iterative and incremental development phases were selected as a software development approach. The iterative and incremental development approach is iterative, architecture-centric, and use case driven. The iterative and incremental development has iterations in every phase of its development. Each iterations builds on the work of the previous iterations to produce an executable that is one closer to the final product.

As a result, managerial activity support iterative and incremental development with changing documents, risk identification and addressing on early phases of the project, defect detections and correction over several iterations, and going integration during the development and not at the end. Identification of risks is emphasized on the guide, as well as commitment of the participants. Identification of risks allows us to detect possible causes of deviations from the established plans and adjust planning according to the situation that is evaluated in every iteration and phase.

CHAPTER 4

IMPLEMENTATION

This chapter briefly discusses about the implementation and testing of School Fees Payment and Accounting Management System. In these section will be explain more about interfaces and explanantion of coding that implement to the system.

4.1 Interface

This system contains two user which are staff and teacher. Each user have their own role in these system which are for staff responsibility to setup fees module, setup account module and managing account module menawhile for teacher responsibility in managae student payment module. Each module will be display about it interfaces in detail.

4.1.1 Login Interface

Figure 4.1 and 4.2 shows that login interface for diffrent user (staff and teacher).



The image shows a login interface for a teacher. It has a brown background. In the top right corner, there is a small icon of a person with the text 'ADMIN' below it. In the center, there is a white box with the title 'LOGIN'. Inside this box, there are two input fields: 'Pegguna ID:' and 'Pegguna Nama:'. The 'Pegguna ID' field contains the text '12' and the 'Pegguna Nama' field contains the text 'AFZAN AHMAD'. Below these fields is a button labeled 'Masuk'.

Figure 4.1: Login teacher interface



The image shows a login interface for a staff member. It has a brown background. In the center, there is a white box with the title 'LOGIN'. Inside this box, there are two input fields: 'Pegguna ID:' and 'Pegguna Nama:'. The 'Pegguna ID' field contains the text '890208' and the 'Pegguna Nama' field contains the text 'AFZAN AHMAD'. Below these fields is a button labeled 'Masuk'.

Figure 4.2: Login staff interface

4.1.2 Setup Fees Form Interface

Figure 4.3 and 4.4 shows that Setup Fees Form interface that allowed staff to insert new list fees form and at the same time in same interface staff is allowed to view and delete list if fees payment.

Figure 4.3: Setup Fees Payment interface

SENARAI PERKARA BAGI YURAN ASRAMA				
No.	Perkara	Bayaran	Jenis Borang	Padam
1	SELIMUT	RM 10.00	ASRAMA	
2	TILAM	RM 10.00	ASRAMA	
3	BANTAL	RM 20.00	ASRAMA	
4	SUKAN	RM 15.00	YURAN	
5	AHLI PIBG	RM 15.00	PIBG	
6	KASUT	RM 30.00	ASRAMA	
7	INSURAN	RM 10.00	PIBG	
8	TABUNG	RM 10.00	YURAN	
9	PEPERIKSAAN	RM 15.00	YURAN	

[KEMBALI](#)

Figure 4.4: View of Fees Payment interface

4.1.3 List of Fees Payment Interface

Figure 4.5 shows that interface for fees payment which are PIBG fees, School fees and Hostel fees.

MAKLUMAT PELAJAR			
No Siri:	2576		
Nama Pelajar:	AHMAD YAMINHANIF BIN SIRUN	Kad Pengenalan:	950120065423
Kelas:	1 CERDAS	Tingkatan:	1
Senarai Pembayaran Yuran PIBG			
No.	Perkara	Bayaran	Tanda
1	AHLI PIBG	RM 15.00	<input type="checkbox"/>
2	INSURAN	RM 10.00	<input type="checkbox"/>
Jumlah:		RM	
Ringgit Malaysia: <input type="text"/> Sahaja			
Maklumat Keluarga Murid			
Nama:	SIRUN BIN HAJI MAMAT	Kad Pengenalan:	890208045388
Nama ahli keluarga:-			
No	Nama		
1	NADIRAH ADIBA BINTI AFANDI		
MAKLUMAT PENERIMA			
Nama Penerima:	ABD MUBIN BIN MOHAMED	Tarikh:	<input type="text"/>
<input type="button" value="SUBMIT"/>			

Figure 4.5: PIBG fees payment interface

4.1.4 Receipt Fees Payment Interface

Figure 4.6 shows that interface of receipt fees payment that will be handover to student as a proof to show that they have done paid their fees.

SEKOLAH MENENGAH AGAMA PULAU TAWAR, 27050 Pulau Tawar, Jerantut, Pahang
Tel Sekolah: 092669342 Faks: 092671270

Resit Pembayaran Yuran Asrama

Nama: AHMAD YAMINHANIF BIN SIRUN Kad Pengenalan: 950120065423
Kelas: 1 CERDAS Tingkatan: 1

Senarai Pembayaran Yuran Asrama

No. Siri: 778

No.	Senarai Yuran Asrama	Bayaran	Tarikh Bayar
1	BANTAL	RM 20.00	2012-05-19
2	TILAM	RM 10.00	2012-05-19
3	SELIMUT	RM 10.00	2012-05-19

Maklumat Penerima

Nama: ABD MUBIN BIN MOHAMED

[PRINT NOW](#)

Figure 4.6: Receipt of fees payment interface

4.1.5 Record of BMA debit and other information Interface

Figure 4.7 shows that interface of record of BMA debit by search using year.

MAKLUMAT TERIMAAN BMA DAN LAIN TERIMAAN

Sila pilih tahun yang hendak dilihat:

Tahun:

Figure 4.7: Search year record of BMA debit

Figure 4.8 shows that interface of record of BMA debit after search the record.

Maklumat Terimaan BMA Tahun 2012									
	LPBT	BANTUAN MAKANAN	YURAN MAKANAN	YURAN PENDAFTARAN	YURAN DOBI	LAIN TERIMAAN	Jumlah	ISI TERIMAAN BMA	LIHAT
	RM	RM	RM	RM	RM	RM	RM		
JANUARI	2	2	2	2	2	2	12		
FEBRUARI	5	5	5	5	5	5	30		
MAC									

Figure 4.8: Display of record of BMA debit by year

4.1.6 Record of BMA debit and other information Interface

Figure 4.9 shows that interface of record of BMA by insert how much school receive money at that month.

Figure 4.9 : Record of BMA interface

4.1.7 Record of BMA debit and other information Interface

Figure 4.10 shows that interface print record of BMA debit to print out as the references.

Figure 4.10: Print Record of BMA interface

4.2 Coding

This section will be explain about coding that have been use and impelement during develop for these system.

4.2.1 Insert Coding

Insert coding is using when to insert new data into the database. Figure 4.11 show the example of insert data coding into the database.

```
$sql="INSERT INTO tborang(borang_perkara,
borang_bayaran,kborang_id,borang_keluaran,borang_asrama)
VALUES
('$borang_perkara','$borang_bayaran','$kborang_perkara','$borang_keluaran','$borang_asrama')";
;
```

Figure 4.11: Insert coding

4.2.2 View Coding

View coding is use to display data from the database. Figure 4.12 show the example to call data from selected database to display and figure 4.13 show the coding to execute data to display.

```
<?php
$query2 =
"SELECT * FROM tborang, kod_borang WHERE kod_borang.kborang_id = tborang.kborang_id";
$result2 = mysql_query($query2) or die(mysql_error());
$count = 1;
while($row = mysql_fetch_array($result2))
{?>
```

Figure 4.12: Call data to display

SENARAI PERKARA BAGI YURAN ASRAMA				
No.	Perkara	Bayaran	Jenis Borang	Padam
   		RM 		

<?php echo \$row['borang_perkara'];?>

Figure 4.13: View coding

4.2.3 Delete Coding

Delete coding is use to delete data from the database. Figure 4.14 show the example to delete data to the database.

```

<?php
mysql_connect("localhost", "root123", "080289") or die("no connection");
mysql_select_db("spsma") or die("no database");
$ref = $_REQUEST['ref1'];
$query = "DELETE FROM tborang WHERE borang_id='$ref'";
mysql_query($query) or die('<script>alert("Delete Complete");</script>');
echo ('<script> alert("Data anda telah dipadam dari pengkalan
data.");location.href=\'Kemaskini_dokumen_borang_percubaan.php\'</script>');
?>

```

Figure 4.14: Delete coding

4.2.4 Search Coding

Search coding is use to search data from the database. Figure 4.15 show the example to search data to the database.

```
<?php
if(isset($_GET['search']))
{
    $sterimaan_tahun = $_POST['terimaan_tahun'];
    trim ($sterimaan_tahun);
    $query2 = "SELECT DISTINCT terimaan_tahun FROM terimaan_bma
    WHERE terimaan_tahun LIKE '%" . $sterimaan_tahun . "%' ";
    $result2 = mysql_query($query2);
    while ($row = mysql_fetch_array ($result2))
    {
        $sterimaan_tahun = $row[terimaan_tahun];
    }
}
```

Figure 4.15: Search coding

CHAPTER 5

TESTING, RESULT AND DISCUSSION

This chapter briefly discusses about the testing, result and discussion of School Fees Payment and Accounting Management System. In these section will be explain more about testing system that use the test case, result and discussion during developing these system.

5.1 Testing

The purpose of this testing is to set the scope, approach, resources, schedule of testing activities for the School Fees Payment Accounting Management System (SFPAMS). The goal is to identify the features to be tested, recognize the items to be tested, the testing task to be performed, the personnel responsible for each task and the risks related.

5.1.1 Objective

The objectives that will be archive of this test plan are:

- i. To define all the resources needed for the testing.
- ii. To list all responsible staffs who involve needed in the test and which tasks that they should do for testing system.
- iii. To follow manual testing to conduct system testing.

5.1.2 Background

This test plan is created to provide a manual testing that approach for the SFPAMS system, so that the testing can be done and completed in the designated environment of the web base application. All the test should be follow according manual testing procedure.

5.1.3 Scope

The test for this SFPAMS system is to determine that all the modules had been made properly. Besides that, the GUI and functionality must be met with the requirements.

This test plan however only attempt to present manual testing system which that all the functional requirements are being tested. It includes the input data test, form navigations; pop up message box, button's functioned and etc.

5.2 Test Case

The purpose of doing the test case is to provide a chronological record of relevant details about the execution on School Fees Payment Accounting Management System (SFPAMS). The test case have been tested in module by module according to test case condition that have been design.

5.2.1 Test Case – Validation Login

Test case:	1.1	Test case name:	Test Validation Login
System:	SFPAMS	Subsystem:	Login
Designed by:	Afzan Bt Ahmad	Design date:	1/5/2012
Executed by:		Execution date:	
Short description:	Test Validation Login (staff and teacher)		

Pre – conditions

1. The user (staff and teacher) is a valid user – contain username and password data in database.
2. The system will be display main menu.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Enter Username “AFZAN AHMAD”	System will not pop up message box for validation textbox form.	Pass	
2.	Enter username by null	System will be display pop up message box to tell that please insert your username.	Pass	
3.	Enter username by integer “1234”	System will be display pop up message to tell that please insert your username.	Pass	
4.	Enter password “890208*****”	System will not pop up message box for validation textbox form.	Pass	

5.	Enter password by null	System will be display pop up message box to tell that please insert your password.	Pass	
6.	Enter password alphabet “abcd”	System will be display pop up message to tell that please insert your password.	Pass	
7.	Click button “Login”	System will be display main menu for different user homepage according to username and password.	Pass	

Post – conditions

None

5.2.1.1 Result



Figure 5.1: Entering username and password by null

Figure 5.1 shows that user login without entering password and username. System will check data if data empty it will top up message box to alert user that they need entering username and password.

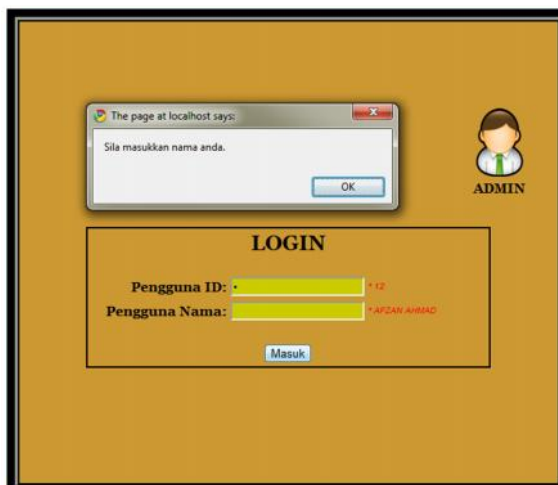


Figure 5.2: Entering username by null

Figure 5.2 shows that user login without entering username. System will check data if data and top up message box to alert user that they need entering username.



Figure 5.3: Entering invalid username and password

Figure 5.3 shows that user login invalid entering username. System will check data if data is invalid and top up message box to alert user that their username and password are invalid data.



Figure 5.4: Entering username by integer

Figure 5.4 shows that user login enter username by integer. System will check data if data is invalid and top up message box to alert user that their username and password are invalid data.

5.2.2 Test Case – Test Calculation of fees payment

Test case:	1.2	Test case name:	Test Calculation of fees payment
System:	SFPAMS	Subsystem:	Fees payment (PIBG, school, hostel)
Designed by:	Afzan Bt Ahmad	Design date:	1/5/2012
Executed by:		Execution date:	
Short description:	Test calculation of fees payment.		

Pre – conditions

1. The teacher will be calculate by total up fees payment that student need to be paid.

Step	Action	Expected System Response	Pass/Fail	Comment
1.	Click check and uncheck at checkbox.	System will be auto calculates depends on checkbox that have been check and display total amount of fees payment.	Pass	

Post – conditions

5.2.2.2 Result

No.	Perkara	Bayaran	Tanda
1	54 AHLI PIBG	RM 15.00	<input type="checkbox"/>
2	57 INSURAN	RM 10.00	<input type="checkbox"/>

Jumlah: RM

Ringgit Malaysia: Sahaja

Figure 5.5: Not check the checkbox

Figure 5.5 shows that system did not calculate for getting amount fees payment

Senarai Pembayaran Yuran PIBG			
No.	Perkara	Bayaran	Tanda
1	54 AHLI PIBG	RM 15.00	<input checked="" type="checkbox"/>
2	57 INSURAN	RM 10.00	<input checked="" type="checkbox"/>
Jumlah:		RM 25	
Ringgit Malaysia: dua puluh-lima ringgit dan sifar sen Sahaja			

Figure 5.6: Check all the checkbox

Figure 5.6 shows that system will be calculate total amount depends on checkbox that have been check.

Senarai Pembayaran Yuran PIBG			
No.	Perkara	Bayaran	Tanda
1	54 AHLI PIBG	RM 15.00	<input checked="" type="checkbox"/>
2	57 INSURAN	RM 10.00	<input type="checkbox"/>
Jumlah:		RM 15	
Ringgit Malaysia: lima belas ringgit dan sifar sen Sahaja			

Figure 5.7: Uncheck one checkbox

Figure 5.7 shows that system will be deduct from total amount depends on checkbox that have been uncheck.

5.3 Expected Result

System has expected to run successful without error and capability to give the right output to the user. User can use the system with systematic and continues. These system will be tested by expert tester to test these system that have meet the user requirement before start to deploy these system to the user.

5.4 Advantage and disadvantage School Fees Payment Accounting Management System (SFPAMS)

5.4.1 Advantage

The School Fees Payment Accounting Management System (SFPAMS) have are several advantages will be get. For teacher, they can manage student fees payment by viewing list of student name in the system according to teacher classroom during registration day. Besides, for the staff can manage setup fees payment using these system without need the paper to copy.

5.4.2 Disadvantage

None

5.5 Constraint

5.5.1 System Constraint

The constraints of the system when build the system is want to get the total amount of each fees such as PIBG fees, School fees and hostel fees. Need to total up by search using date of pay to get the total amount for managing school accounting record for that month.

5.5.2 Development Constraint

The development usually faces the development constraint during the development of the system either at the beginning of the development, in the middle of development or during the end of the system development. At the beginning of the development, the developer may have a problem to interact or communicate with the client about the requirement that they need. They need an agreement to make sure they understand what the requirement that must have in their system. During the middle of development, the changes maybe happen because of client requirements for the system or the developer

cannot follow the date that have been fixed for them because of many changes. To overcome the constraints, all the document and system progress have to do a backup and must be record so that it is easier to refer when the problem occur.

CHAPTER 6

CONCLUSION

The developed applications, School Fees Payment and Accounting Management System have met all the objectives of this project. All of objectives were successfully. This system is provide the process of School Fees Payment and Accounting Management System that record about registration new hired teacher and setup fees school form for staff module meanwhile for module teacher is calculating school fees payment, generate record and accounting school record

.

This system is web based application. This system will have two types of user which are teacher and staffs which are this system are connected to a network. Although this means it can share resources with any other computers.

The staff needs to login by entering ID as username IC number as password for security authorized and authentication and then it will go to the staff homepage. They also responsible to manage setup school fees list payment, setup account and school account record in debit and credit.

For the teacher they also needs to login to enter ID as username and IC number as password for security authorized and authentication and then it will go to teacher homepage. Teacher will calculating school fees payment to get total amount fees, view and updated student current information for paying school fees payment.

REFERENCES

- [1] K.Charles (n.d). "Accounting Concept." Available:
<http://www.accsoft.ch/download/accountingconcepts.pdf>
- [2] M. Brumbulli, B. Topciu, A." Dalaci. A Web Based School Management Infromation System." Vol. (2), pg. 564. Available:
<http://csconf.org/Volume2/page564.pdf> [September, 5th 2009].
- [3] K. Demir. "School Management Infromation System in Primary Schools." Available: <http://www.eduswift.com/resources/EDUR1001EB-EduSwift-School-Management-System.pdf> [June, 5th 2008].
- [4] P. Sally. (n.d). "Chapter 11:Data Flow Diagram". Available:
http://www.ibm.com/developerworks/rational/library/content/03July/2500/2772/2772_data_flow_diagram.pdf
- [5] S. Kemsley.(2006). "BPMN and Business Process Management." Available:
http://www.omg.org/bpmn/Documents/6AD5D16960.BPMN_and_BPM.pdf
- [6] M.James (n.d). "ICT Business Function Analysis." Available:
http://www.moct.gov.sy/ICTStandards/en_pdf/13.pdf
- [7] M. Tillmans and M. Lin. (2008). "Business Process Modelling." Available:
http://www.tibco.com/multimedia/business-process-modelling_tcm8-2404.pdf
- [8] J. A. Hoffer, J. F. George, J. S. Valacich (n.d). "Software Analysis and Design".
- [9] "Wikipwedia". Available:
http://en.wikipedia.org/wiki/Iterative_Incremental_Development
- [10] Dr. A.Cockurn, J. F. George, J. S. Valaciah (2008). "Iterative and Incremental Development". Available: <http://www.snyders.us/qa-iterative.htm>

APPENDICES

APPENDIX A

APPENDIX B

APPENDIX C

APPENDIX D

APPENDIX E

APPENDIX F