E-BOOK MAKER SYSTEM

NUR SYAZILA BINTI HAMZAH

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

E-BOOK MAKER SYSTEM

NUR SYAZILA BINTI HAMZAH

report is submitted in partial fulfillment of the requirements for the

Bachelor of Computer Science (Software Development)

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2018

DEDICATION

To my beloved parent, Hamzah Bin Aziz and A'dah Binti Samsudin who have supported me to get through everything from diploma until degree. To my Supervisor, Encik Muhammad Suhaizan Bin Sulong who has helped and generously providing guidance to me from the very beginning of this final year project until the end. Lastly, to my housemates who have never failed to help me and support me to get through finishing my final year project.

ACKNOWLEDGEMENTS

First of all, I would like to express my deepest appreciation to my Supervisor Encik Muhammad Suhaizan Bin Sulong for being very helpful in order for me to finish this Final Year Project. He has the attitude and the substance of a genius; he continually helped me during my project. Without his guidance and persistent help, this project would not be this successful.

Also, I would like to thank beloved parents for being supportive throughout my studies in Universiti Teknikal Malaysia Melaka (UTeM). They have spent a lot on me and I wish nothing but to manage finishing my degree successfully and get a scroll.

In addition, a thank you to all my housemates and friends, especially the ones that have been staying with me since the day one (1) in Universiti Teknikal Malaysia Melaka (UTeM) until today. Without them, my life in UTeM would be dull.

ABSTRACT

The E-Book Maker System is a web based system that is only focusing on a novel that have various types of genre such as romance, thriller and many more. This system is created for authors, publishers and readers. It was developed with the aim of helping authors to write the book online and helping publisher to publish the book online. In addition, it is also facilitating readers to read the book online without having to buy books in the bookstore. The system contains six (6) modules which is, register-login-recovery-profile, content e-book, generate e-book, readers comment and report. Methodology use to develop E-Book Maker System is Agile Methodology. By having the E-Book Maker System, it will make it easier for authors to write books online and publishers publish books online. Also, facilitates readers to read books online. In a conclusion, the E-Book Maker System has been successfully completed, although it does not fulfill some of the requirements of this system.

ABSTRAK

E-Book Maker System *adalah sistem berasaskan web yang hanya memberi tumpuan kepada novel yang mempunyai pelbagai jenis genre seperti roman, thriller dan banyak lagi. Sistem ini dicipta untuk penulis, penerbit dan pembaca. Ia telah dibangunkan dengan tujuan membantu penulis menulis buku dalam talian dan membantu penerbit untuk menerbitkan buku dalam talian. Di samping itu, ia juga memudahkan pembaca membaca buku dalam talian tanpa perlu membeli buku di kedai buku. Sistem ini mengandungi enam (6) modul iaitu mendaftar-login-recovery-profile, e-book kandungan, menghasilkan e-book, ulasan pembaca dan laporan. Kaedah yang digunakan untuk membangunkan* E-Book Maker System *adalah* Agile Methodology. *Dengan memiliki* E-Book Maker System, *ia akan memudahkan penulis menulis buku dalam talian dan penerbit menyiarkan buku dalam talian. Juga, memudahkan pembaca membaca buku dalam talian. Dalam kesimpulannya,* E-Book Maker System *telah berjaya diselesaikan, walaupun ia tidak memenuhi beberapa keperluan sistem ini.*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Introduction

The idea of proposed system named as E-Book Maker System. This system is only focusing on novel that have various type of genre. This system will allow authors to create E-Book for free. In today's competitive world, it becomes imperative for the authors or content marketers to create an E-Book that appeals to the readers. The E-Book is available to read right away keeping it fresh in reader minds rather than waiting for the physical edition to be released. It even saves the substantial time of the reader to go to the bookstore, buy it and then read. With this system authors can save their prepared E-Book to continue writing later or save their completed book.

E-Book Maker System is a web based system that allows the authors to write their book without any problem and any charge. Author will have to register into the system and existing author can write new book or resume with the old-book. After author have

finish write the book, author need to send the writing to the publisher to publish the book. This system also allows readers to give comment and rating about the book.

1.2 Problem Statement

The main problem found in the current system is that the author writes a book in a computer using word processing software. This does not guarantee that the book kept in the computer is safe. In case if there is probability that will occur to the computer such as computer error or damage, it will cause the loss of writing.

Thus, by using the online system the authors can store and backup their writing in the system. Besides, the authors have to spend money on printing books and readers have to carry a book. Therefore, with using this system the readers can read book online. Lastly, paper wastage occurs. It will be a waste of paper to print all the pages in the books.

1.3 Objective

The objectives of this proposed system are:

i) To create paperless environment substituting paper based system.
ii) To develop a computerized system of publishing book.
iii) To make it easier for authors to write a book
iv) To generate reports tracking progress writing and publishing books.

1.4 Scope

The scope of this project is focus on how to help author and publisher to manage the E-Book. The result of this project is to make sure that the E-Book Maker System can bring more efficient and benefits for author and publisher.

Table 1.1: Scope of User System

User System	Description
1. Author	Author can register and log in into the system. Author can recover their data such as password and also can view their profile. Besides, author can manage the content of E-Book and generate the E-Book. Author also can view the readers feedback and view the report.
2. Publisher	Publisher can publish the book. Publisher also can register the book and publish it to E-Book Maker System.

Table 1.2: Scope of System Modules

System Modules	Description
1. Log In, Register, Recovery, Profile	Allows author and publisher to log in into the system, register the information, recovery data such as password in case of user forgot their password and view profile. User also can update their profile and changed password.

2. Content E-Book	Allows author to write E-Book into the system and database. It is also allowing author to insert image if there have image to put. In this module also allows author to resume the writing.
3. Generate E-Book	In this module, allows author to generate the E-Book to pdf file.
4. Publish E-Book	Allows publisher to publish and register the book into the E-Book Maker System.
5. Readers Comment	Allows readers to give a feedback to the book. Author can view the feedback. Readers also can give rating to the book.
6. Report/Dashboard	Allows author and publisher to view the report based on the readers reading the book.

1.5 Project Significance

The project significance of this proposed system are:

i) Efficiency

- Data is stored in the database and it also can be managed easily.

ii) Save Time

- Author can directly save the writing into the system without having any problem.

- Publisher can read or view the book directly into the system when author send the book to publisher.

iii) Centralize Tasks

- When the task is being centralized, it makes it easier for the user and data replication also can be avoided.

1.6 Expected Output

The expected output of this project is to develop system E-Book Maker System successfully that it will be able to perform functionalities correctly and therefore achieve the objectives of the project.

1.7 Conclusion

In conclusion, the system will provide solutions for the user based on the current problem faced. The system is built with all the functions stated and it will definitely help the author and publisher. Therefore, with the existence of this system, it will handle the data efficiently and boost up the book and publish service.

CHAPTER II

LITERATURE REVIEW AND PROJECT METHODOLOGY

2.1 Introduction

This chapter will discuss about the review of the literature that pertains to the domain of the E-Book Maker project as well as the project methodology that is used to conduct the project. The literature will cover the existing system that are in the domain that the proposed system and system will support. Through the analysis of the current system, the requirements and design of the proposed system can be gathered to ensure that the system be more fitting to be used. The project methodology used will help make the entire development of the application be more organized and run smoothly by following the try and tested project methodologies.

2.2 Facts and Findings

Research has been made on various type of references such as documentation, internet sources and observation about the existing systems in the related domain.

2.2.1 Domain

Writing an e-book of type Novel is the domain for this project. It provides any genre to chooses from such as Romance, Thriller, Adventure and many more. This Novel domain is chosen because it is one of the popular types of e-book which is written by.

2.2.2 Existing System

The studies yielded completely different results as to the issue of awareness of e-books among members of the academic community, however otherwise the speed of agreement between the studies was high. Most of them found that academic users generally search e-books for discrete bits of information, a behavior summed up by the formula "use rather than read." They also show that such use of e-books is typical across disciplines, but that members of the humanities and social-sciences were on the entire less satisfied with e-books than their counterparts within the hard-sciences and business (Staiger, Jeff, 2016).

E-Books is great for reading on the go. A significant benefit of e-book is their small form factor combines with the ability to store them while not the need for bookshelves. Whether you are traveler or minimalist, e-books are the ultimate portable way to enjoy reading. There is no need to fill your suitcase with a dozen heavy hardcovers. Retaining

information can be easier for a few with physical books. Those who read the paper book fared higher on the quiz. There's a sort of feeling that comes with reading books that may actually fire a different set of synapses in our brains. Also, we're so accustomed to skim-reading digital media that it's hard to read with the same focus. Another very important factor in the e-book vs traditional paper debate concerns publishing, it is rudimentary and typically unengaged to publish information in digital formats. This sidesteps the need to find an agent to secure a book deal with a publisher. (David Richardson, 2018).

2.2.3 Technique

In this section will tell the other techniques that used to make this system. Among the technique is plugin text editor in JavaScript. This text editor is a lightweight WYSIWYG HTML Editor written in JavaScript that enables rich text editing capabilities for applications. The function of this text editor is the same as Microsoft Word which can make the text bold, italic and underline. In addition, it also can insert picture, videos, links and table. This text editor is used for writing the book in E-Book Maker System. It is used to make it easier for users to write books more efficiently. It also makes it easy for the user to insert the picture while write the book.

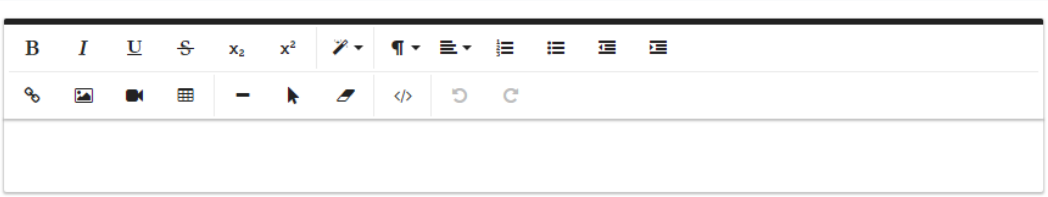


Figure 2.1: Text Editor used in E-Book Maker System

The next technique used in this system is generate the book into pdf file. This technique is used to facilitate the author to generate the book into pdf file and facilitate the readers to read the book through pdf file. In addition, the book that has been generated into pdf file is easy to print and saved into the user's computer.

The last technique used in this system is making the pie chart graph. This pie chart graph is used to facilitate the author to know about the status of the book that has been sent to the publisher. The status of book is divided into three (3) which is approved, pending and reject. So with this pie chart it can make it easier for authors and publishers to know about book status.

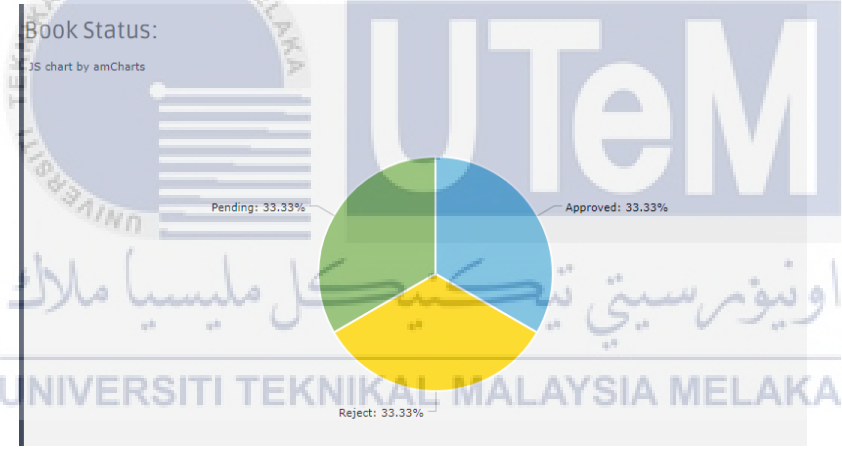


Figure 2.2: Pie Chart for Book Status in author user interface

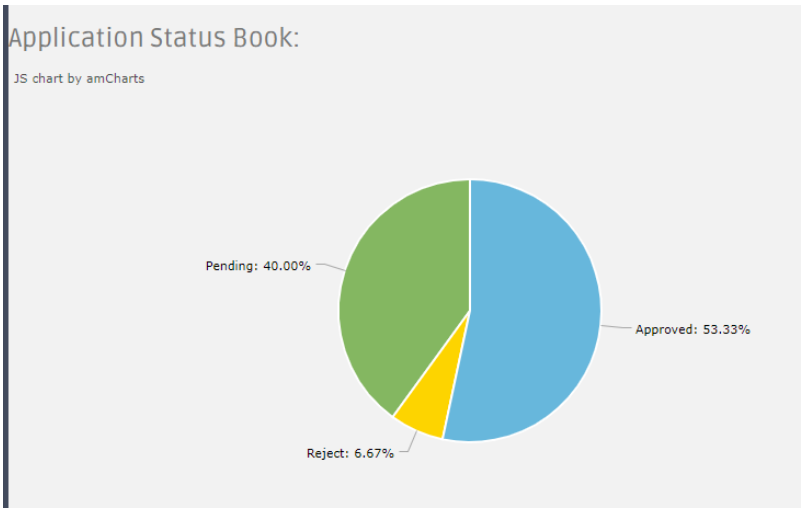


Figure 2.3: Pie Chart for Application Status Book in publisher user interface

2.3 Project Methodology

The E-Book Maker System will be developed according to the agile methodology approach. This method consists of five (5) main phases. There are requirement gathering and planning, analysis and design, implementation, testing and feedback. The goal of agile method is to adapt change and deliver working system as quickly and possible. The fundamental of agile methods is incorporate iteration and the continuous feedback that it provides to successively refine and deliver a system that user need. They all involve continuous evolution, continuous integration. What is more important about agile methods is that they focus on empowering people to collaborate and make decisions together quickly and effectively.

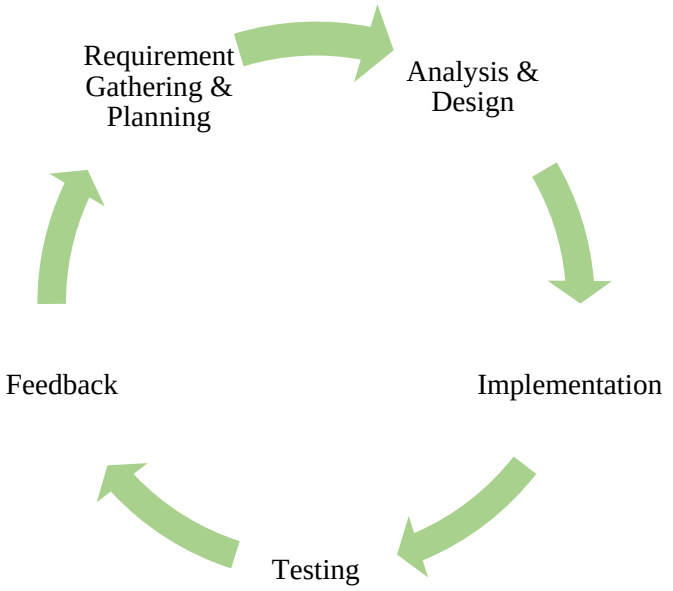


Figure 2.4: Agile Methodology

The first phase is ensuring the project that will be chosen can be completed by the time that has been given. In the requirement gathering and planning phase, the requirements needs by each of the user are gathered by conducting an interview for each user according to their responsibilities. There are two (2) user which are author and publisher. Each user holds different responsibilities and function on the system, so the requirement gathered by each of the user are also different. It is important to ensure the requirement is not switched with another user during development.

The analysis and design phase will first identify the lack and weaknesses of the current system. Based on the weakness that has been found in the current system, the important design model such as the Use Case Diagram and the Entity Relational Diagram (ERD) will be developed to ensure the next phase can be proceed without any ambiguity. For the design phase, the system design is prepared from the requirements identified in the previous phase. The input interface and output interface of the system will be designed quickly. The coding of user interface is done by using html5 and styling the interface by using css3. For further explanation regarding ERD or user interface, will be discussing in the Chapter IV.

The implementation phase is all about creating, testing features and scheduling iterations for deployment. The main function and modules needed to develop a system will be defined. The function coding will be using PHP language and JavaScript is used to make validation on every module to be developed. Once the code has been developed, it is tested against the requirements to make sure the system is actually solving user needs. The testing phase is where the users and the developer meet to deliver the system that has been developed to ensure the function works. The users will use the system and will come out with their own comments and feedback. For further explanation, will be discussing in Chapter V.

The last phase in agile methodology is feedback. Accepting user's feedback and work into the requirement stated during the testing phase. The modification and enhancement will be made to the system. Multiple iteration will take place during agile software development lifecycle and each follows its own workflow. During the iteration, it is important that users provide feedback to ensure that the system is developed according to the user's needs.

2.4 Project Requirements

2.4.1 Software Requirement

Table 2.1: Software Requirement

Type of Software	Description
1. Sublime Text 3	Sublime Text 3 is an open source software for PHP code editor.
2. XAMPP Control Panel	XAMPP Control Panel is the server to connect database.
3. MySQL Database	MySQL is an available open source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL).
4. Microsoft Windows 10	Windows 10 is the Operating System (OS) used in the laptop that is used to develop the system.
5. Microsoft Word	Microsoft Word is word editor for report documentation.
6. Draw.io	Draw.io is a free online diagram software that is used to create the diagrams that is used to visualize the system such as the entity relationship diagram, flowchart and others.

2.4.2 Hardware Requirement

i) Lenovo Laptop (64-bit)

ii) HP Deskjet Ink Advantage 2545 printer

2.5 Project Schedule and Milestones

Table 2.2: Project Milestone

Milestone	Task
08/02/2018	Provide project title
15/02/2018	Planning output: – Requirement gathered. – Existing problem in existing system are defined. – Objective project are clearly defining. – Understand business process. Submit proposal.
22/02/2018	Analysis output: – Methodology to use in development are choose. – Comparison on business process in existing system and system to be has produce. – List the business entity that include in the existing system. Submit progress report I, II and III
01/03/2018	Design output: – Logical database – Physical database – System architecture design – Interface design
08/03/2018	Demonstration interface design to supervisor
15/03/2018	Implementation output: – Functioning the modules o Register, Log In, Recovery and Profile o Content E-Book
22/03/2018	Demonstration modules 1 and 2 to supervisor. Submit progress report IV.
29/03/2018	Implementation output: – Continue functioning the modules

	○ Generate E-Book ○ Publish E-Book
05/04/2018	Submit progress report V.
12/04/2018	Demonstration modules 3 to supervisor
19/04/2018	Implementation output: – Continue functioning the modules ○ Readers Comment ○ Generate Report Demonstration modules 4 to supervisor
26/04/2018	Demonstration modules 5 to supervisor
03/05/2018	Demonstration modules 6 to supervisor Submit progress report VI and VII.
10/05/2018	Testing and feedback output.
17/05/2018	Presentation & Submit Full Report.

Table 2.3: Project Schedule

TASKS	WEEKS														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PLANNING															
Identify project requirement															
Develop project plan															
ANALYSIS															
Identify the current system problem															
Identify the methodology															
DESIGN															
Design ERD															
Sketch storyboard															
Design input and output interface															
IMPLEMENTATION															
Install software environment															
Code user interface															
Code functional requirement															
MAINTENANCE															
Test system function															
Test system validation															
Identify error occur															
Fix system error															
PRESENTATION															
Presentation															

2.6 Conclusion

In a nutshell, to complete this project on time, the planned schedule and milestone are being followed strictly. Any problems or issues occurred during the development of this system that interrupt the progress of the milestone has been resolved as soon as possible because it is a matter of time to produce a system that meets all the stated objectives. Next chapter will discuss on system analysis including analysis of current system and proposed system.

CHAPTER III

ANALYSIS

3.1 Introduction

The analysis of the current problems as well as how the proposed system should be developed to address said problem will be covered in this chapter. The analysis phase is very important to the software development process as it follows the requirement of the software product to be gathered by the developers. Through comprehensive problem analysis and requirements gathering, the developers will be able to create a software that fulfils its requirements and will be of high quality and therefore be useful for the intended users.

3.2 Problem Analysis

The current system faced a lot of problems daily in handling tasks and data. The main problem found in the current system is that the author writes a book in a computer using word processing software. This does not guarantee that the book kept in the computer is safe. In case if there is a probability that will occur to the computer such as computer error or damage, it will cause the loss of writing. Thus, by using the online system the authors can store and backup their writing in the system.

Besides, the publisher has to publish and print the books manually. It will be a waste of paper to print all the pages in the books and it will be using a lot of cost expenses for printing the books. Readers also need to spend money on buying books. With this system, readers can read books for free.

Other than that, readers will have a problem storing their books if there are too many books are owned by the readers. Readers also need space to store the books. So with the system, readers can read the books online also can store books in the computer via pdf file. Last but not least, readers also have to bring the book everywhere. So with the E-Book Maker System, readers can open the system and read it online.

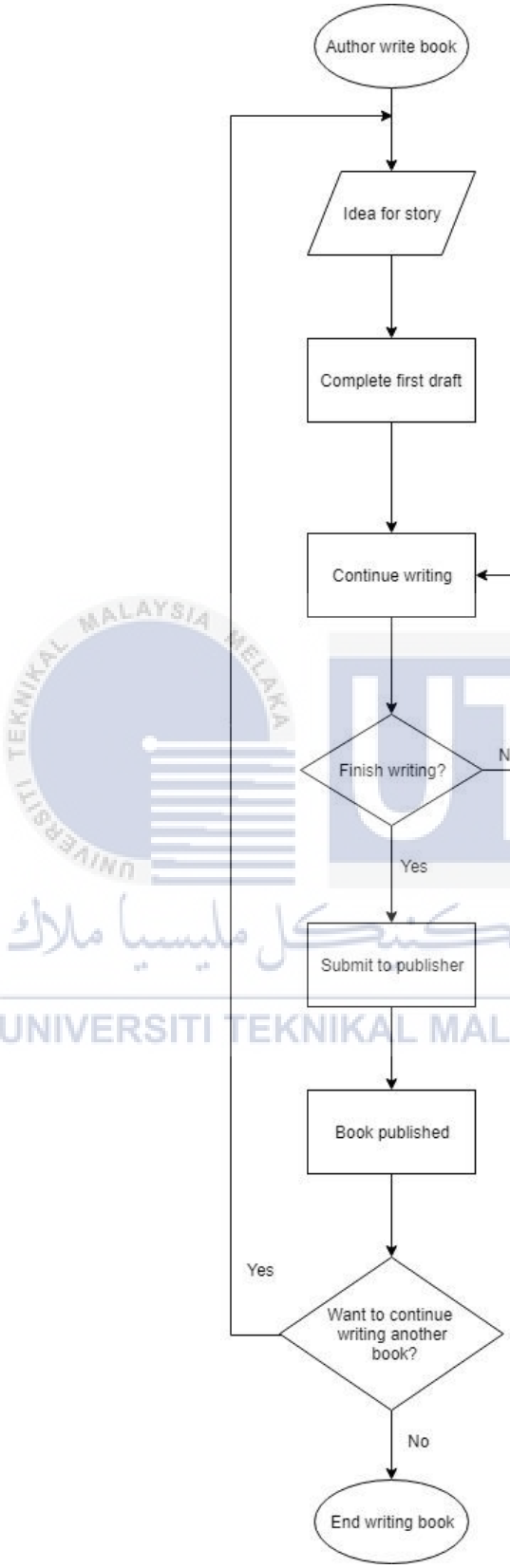


Figure 3.1: Flowchart Current System

3.3 Requirement Analysis

3.3.1 Data Requirement

In this section will be explained in the Chapter 5. For further explanation regarding Data Dictionary can refer in chapter 5.

3.3.2 Functional Requirement

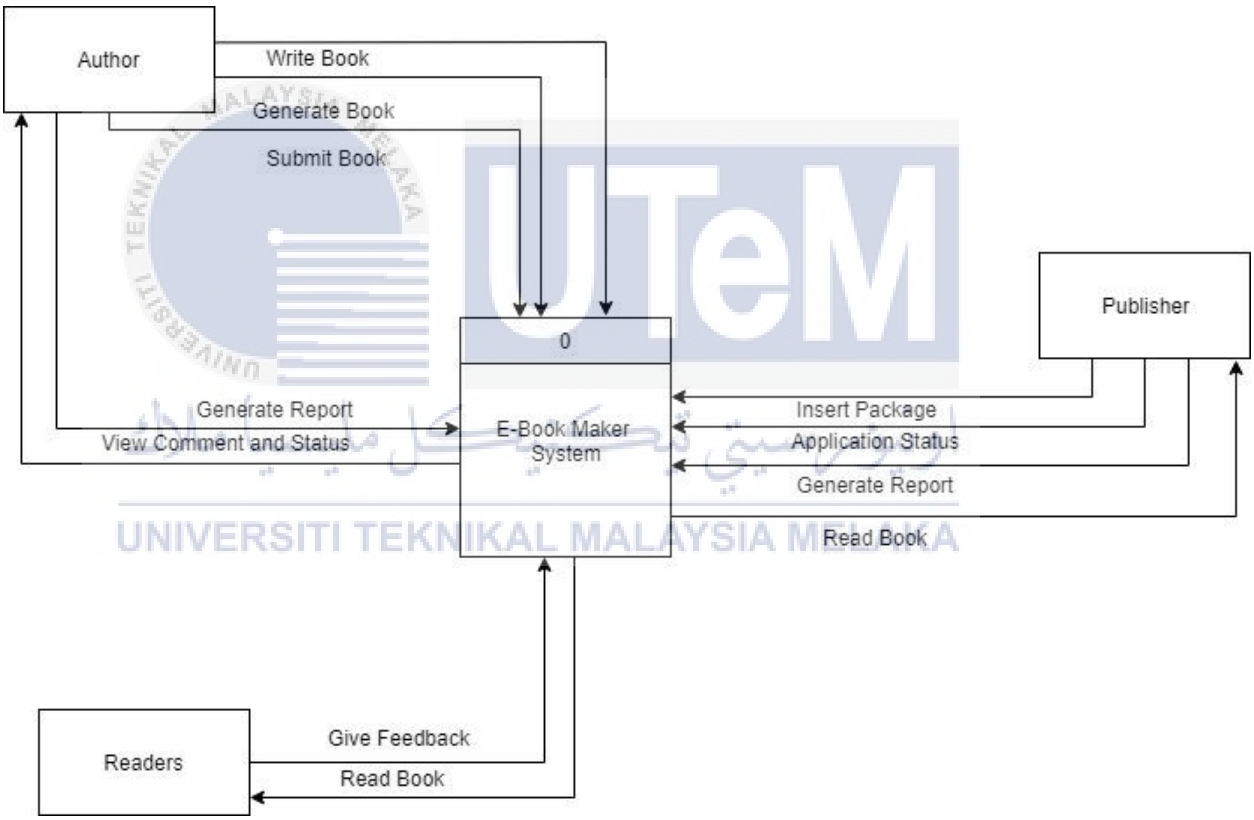


Figure 3.2: Context Diagram

Figure 3.2 shows the context diagram that is drawn for E-Book Maker System. It contains a process that represent the system to model. It also shows the participants who interact with the system, called the external entities. In this system, author, publisher and readers are the entities who interact with the system. In between the process and the

external entities, there are data flow that indicate the existence of information exchange between the entities and the system.

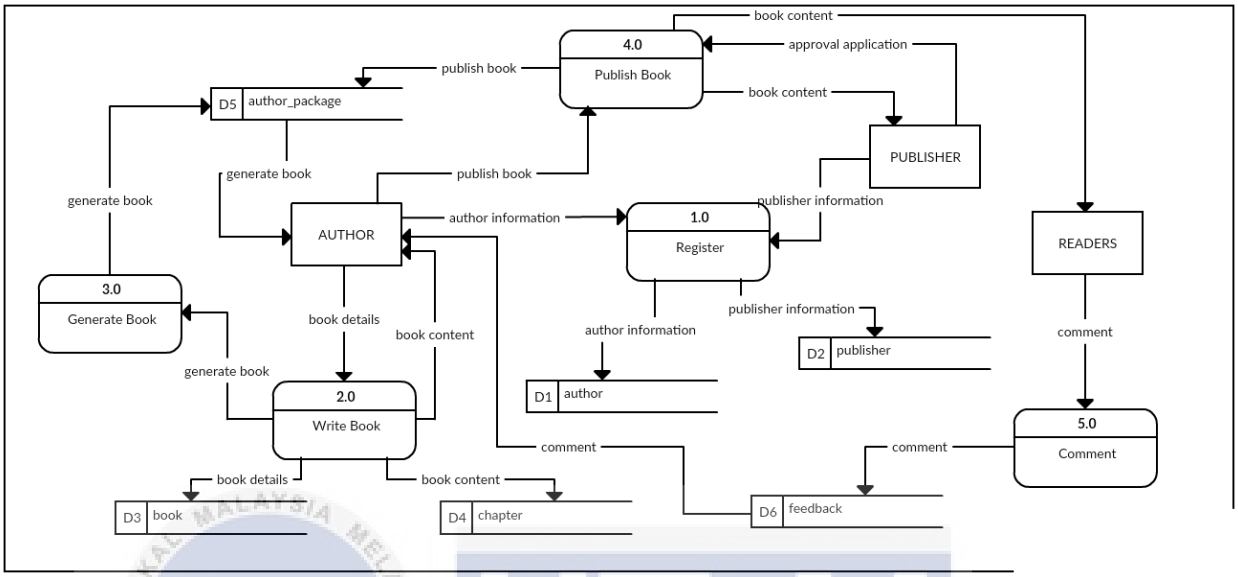


Figure 3.3: Data Flow Diagram Level 1

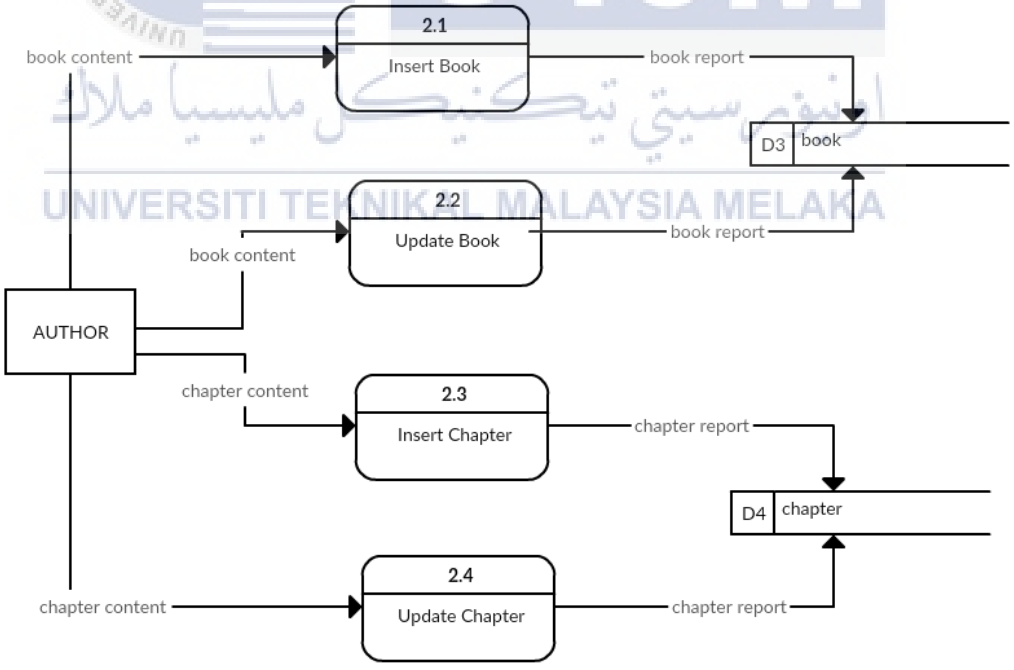


Figure 3.4: Data Flow Diagram Level 2

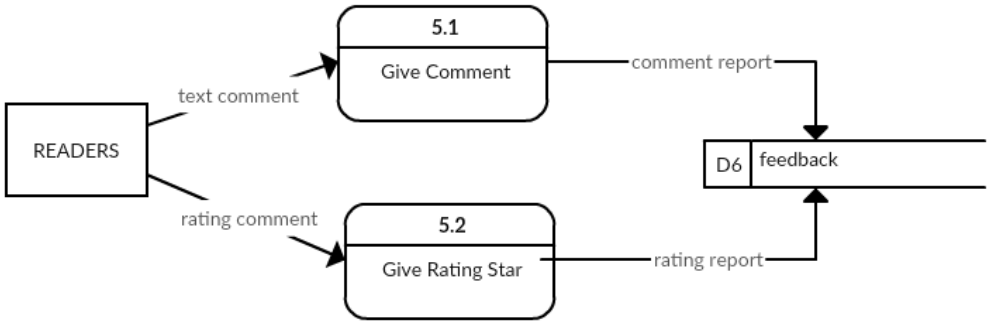


Figure 3.5: Data Flow Diagram Level 2

Figure 3.3 shows the data flow diagram level 1 which is the decomposition of the E-Book Maker System process. It contains five (5) process, three (3) external entities and six (6) data stores. Figure 3.4 shows the data flow diagram level 2 for write book while Figure 3.5 shows the daya flow diagram level 2 for give comment.

3.3.3 Non-Functional Requirement

Non-functional requirements of E-Book Maker System are listed as follows:

1) Response Time

   The system will refresh the page when author insert the chapter in the Table of Content page. It is easier for author to write the book based on chapters.

2) Security

   The system should not reveal other user's information for the other users beside the administrators.

3) Integrity

Consistency – data shall be 100% consistent at all sites and at all times between all the interfacing component.

3.3.4 Others Requirement

Table 3.1: Software Requirement

Type of Software	Description
1. Sublime Text 3	Sublime Text 3 is an open source software for PHP code editor.
2. XAMPP Control Panel	XAMPP Control Panel is the server to connect database.
3. MySQL Database	MySQL is an available open source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL).
4. Microsoft Windows 10	Windows 10 is the Operating System (OS) used in the laptop that is used to develop the system.
5. Microsoft Word	Microsoft Word is word editor for report documentation.
6. Draw.io	Draw.io is a free online diagram software that is used to create the diagrams that is used to visualize the system such as the entity relationship diagram, flowchart and others.

Table 3.2: Hardware Requirement

Component	Description
Laptop	Lenovo 64-bit G40
Printer	HP Deskjet Ink Advantage 2545 printer

3.4 Conclusion

As a conclusion, this chapter discussed about the analysis of the SDLC whereby the current problem was analyzed and the requirement of the proposed system was elicited based on the analysis of the current problem. As a result, the requirements analysis will allow for the development to proceed to the next step which is the design phase which will be discussed in the next chapter.

CHAPTER IV

SYSTEM DESIGN

4.1 Introduction

In this chapter will discuss about the design of the system whereby the high-level design of the system will be formulated so that the system can proceed to the development phase. This phase is very important to ensure that the knowledge and requirement that have been made from the previous phases will be used to design the system that will be developed. The system that is developed will be able to fulfill all the requirement based on the good design that has been used.

4.2 High-Level Design

4.2.1 System Architecture

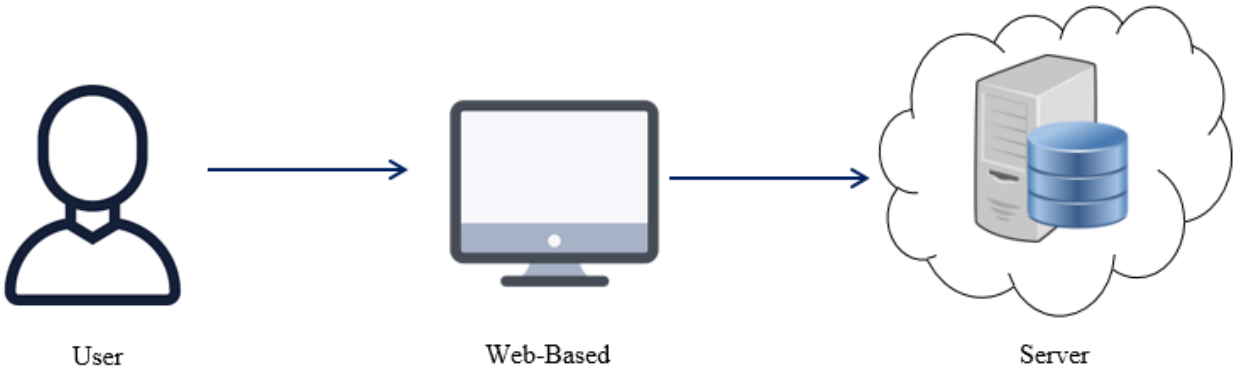


Figure 4.1: System Architecture

Figure above shows the system architecture diagram for E-Book Maker System. The system architecture consists of the user of E-Book Maker System that obtain specific information. The information can be obtained either from web-based. All files and database storage are located on the server which are the web server and database server. This system is a web-based system.

4.2.2 User Interface Design

4.2.2.1 Navigation Design

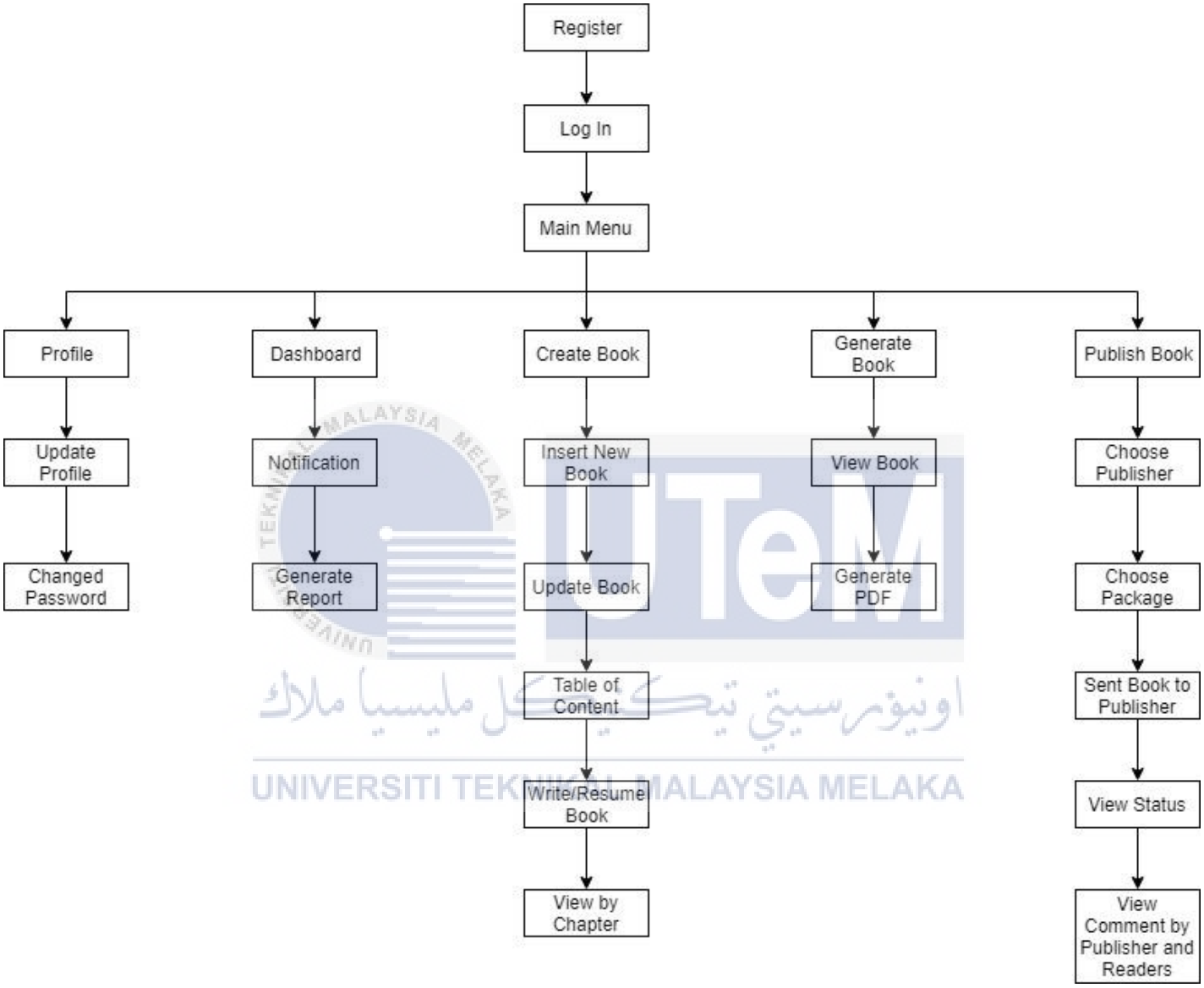


Figure 4.2: Navigation Design for Author

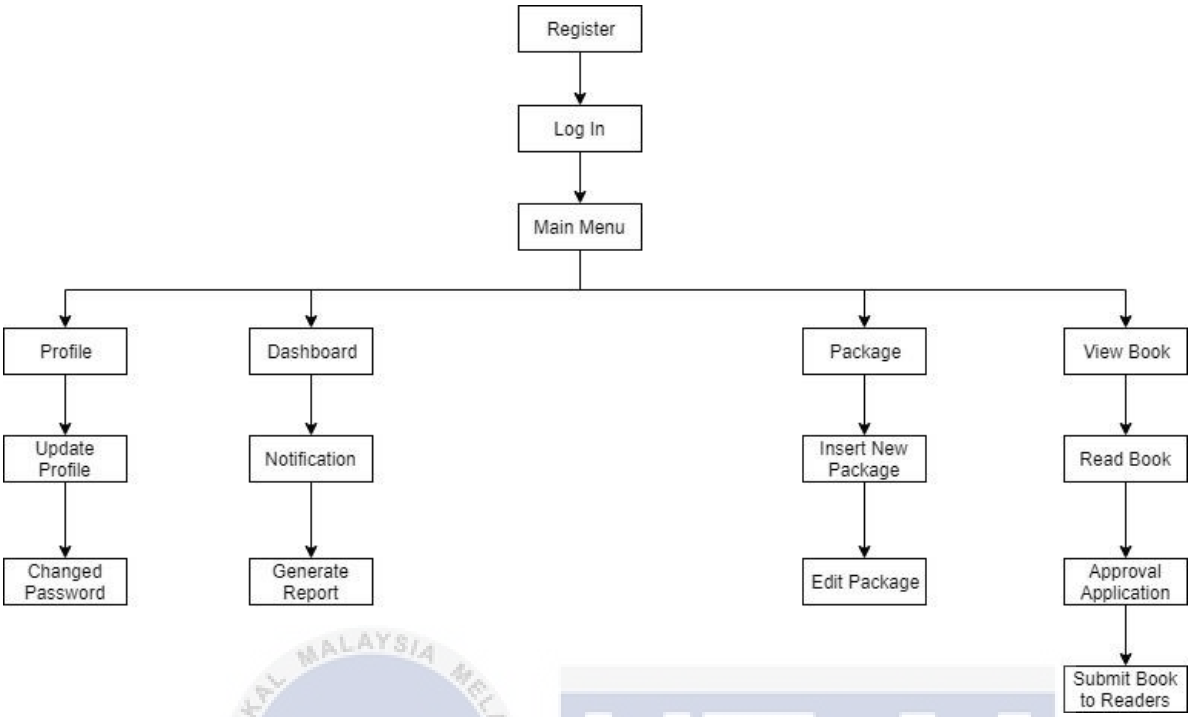


Figure 4.3: Navigation Design for Publisher

The navigation design is to show the relationship between user interface after clicking the button. This is intended to make it easier for users to use the system.

4.2.2.2 Input Design

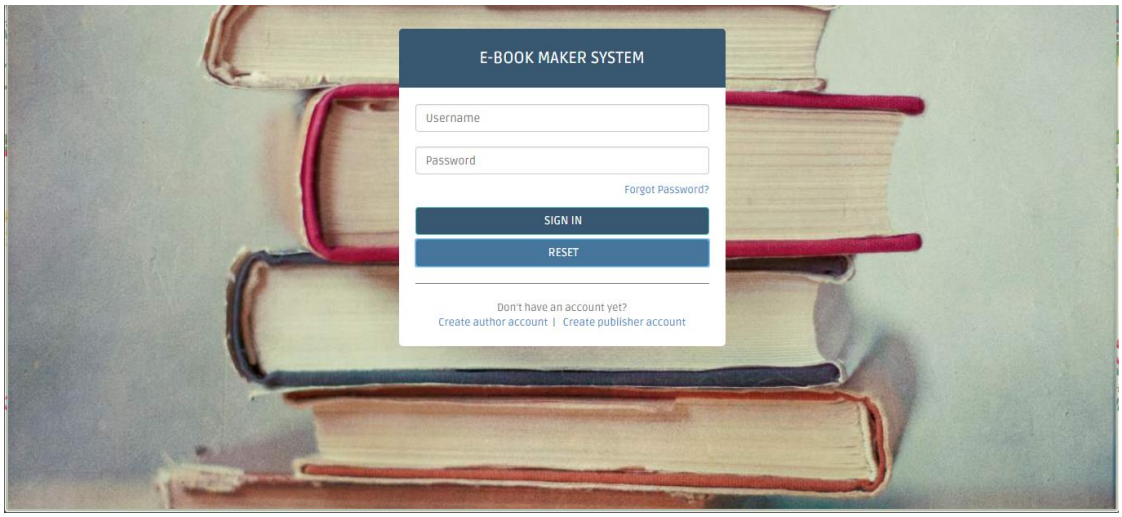


Figure 4.4: Log In form

This is the user interface for user which is author and publisher to log in. User need to enter username and password to access into the system.

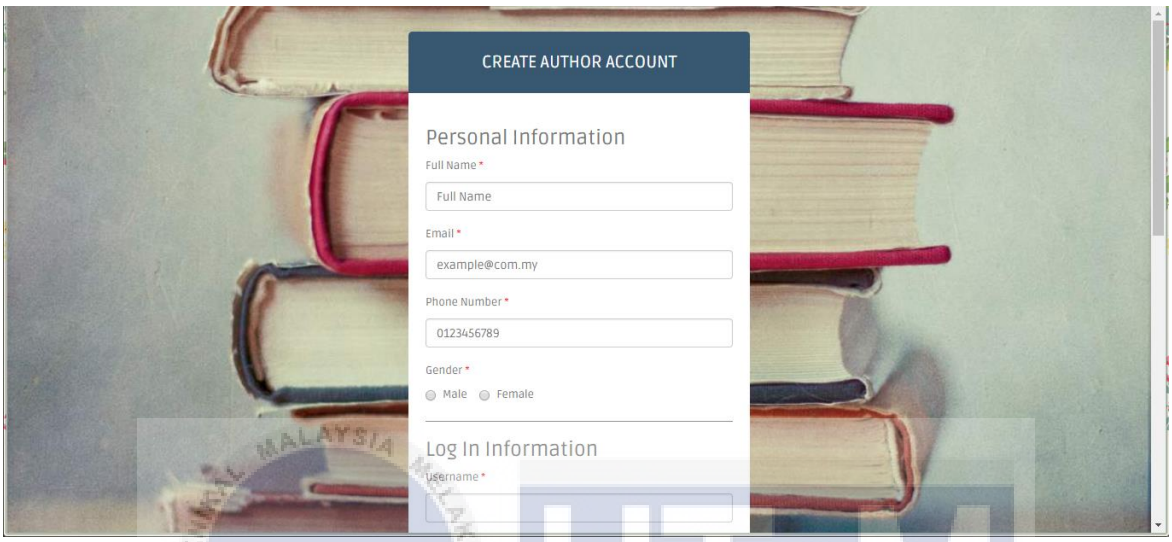


Figure 4.5: Registration form for Author

Figure 4.6: Registration form for Publisher

Figures above show the registration form for author and publisher. Author have to enter the general information such as full name, email, phone number, gender, username, password, confirm password and two (2) answers for security question. Besides, publisher have to enter the publisher information such as publisher name, address, phone number, fax number, email, username, password, confirm password and two (2) answers for security question.

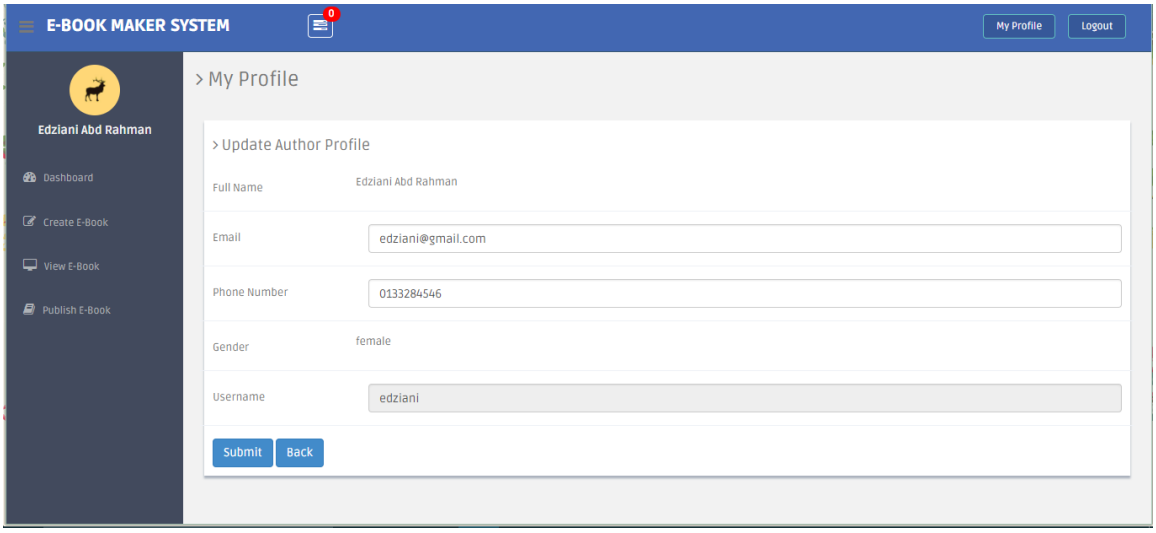


Figure 4.7: Update Author Profile form

This form allows author to update their profile. The information that can be updated are email and phone number.

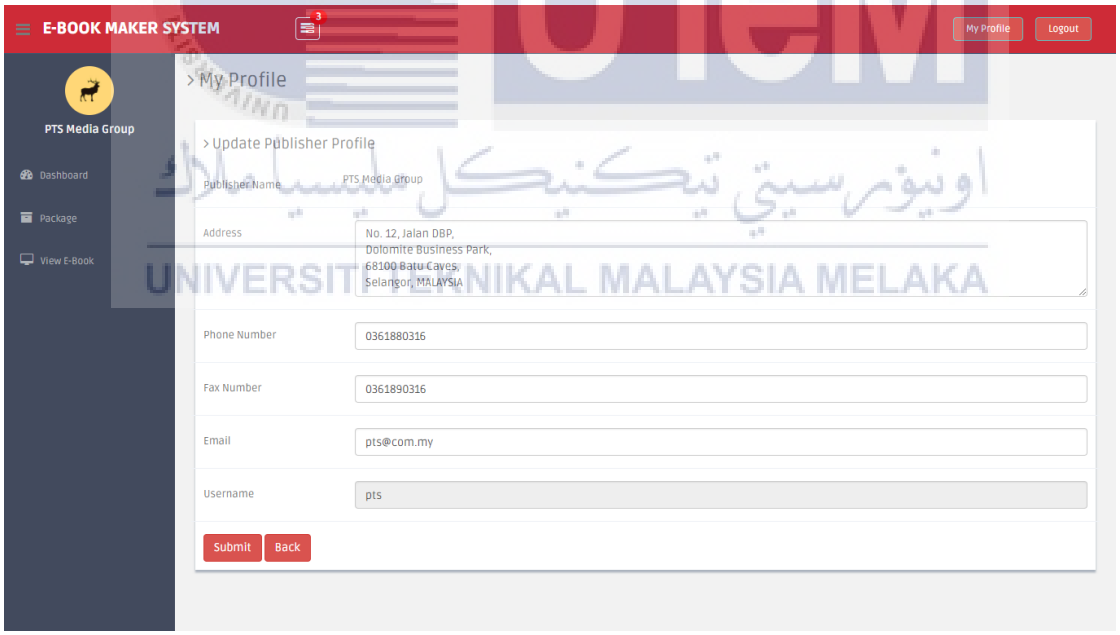


Figure 4.8: Update Publisher Profile form

This form allows publisher to update their profile. The information that can be updated are address, phone number, fax number and email address.

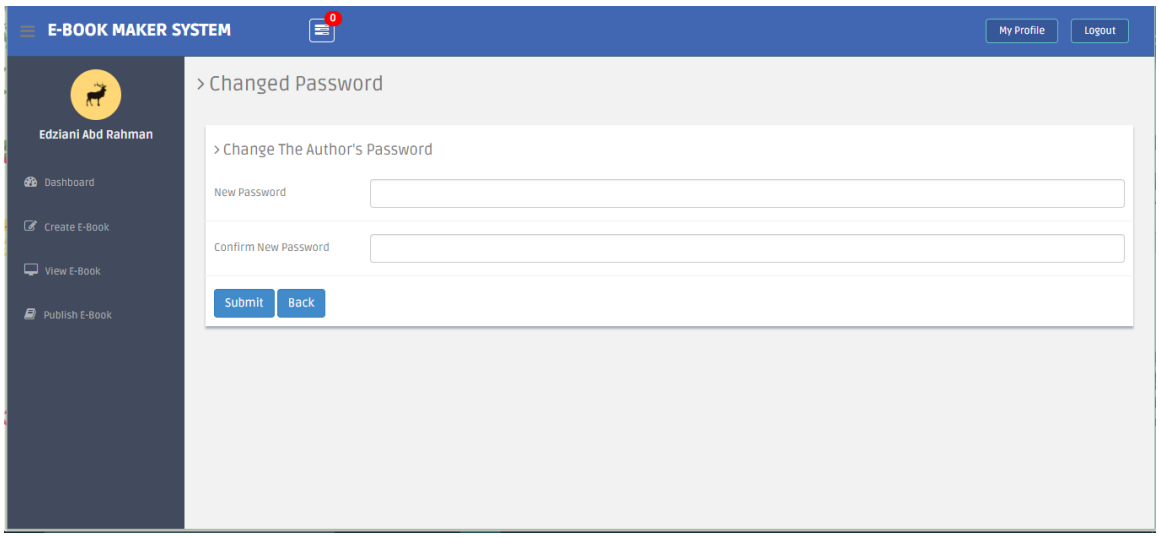


Figure 4.9: Changed Password form

Figure above shows the user interface that are allows author to change the password. Author need to enter the new password and confirm new password to change the password.

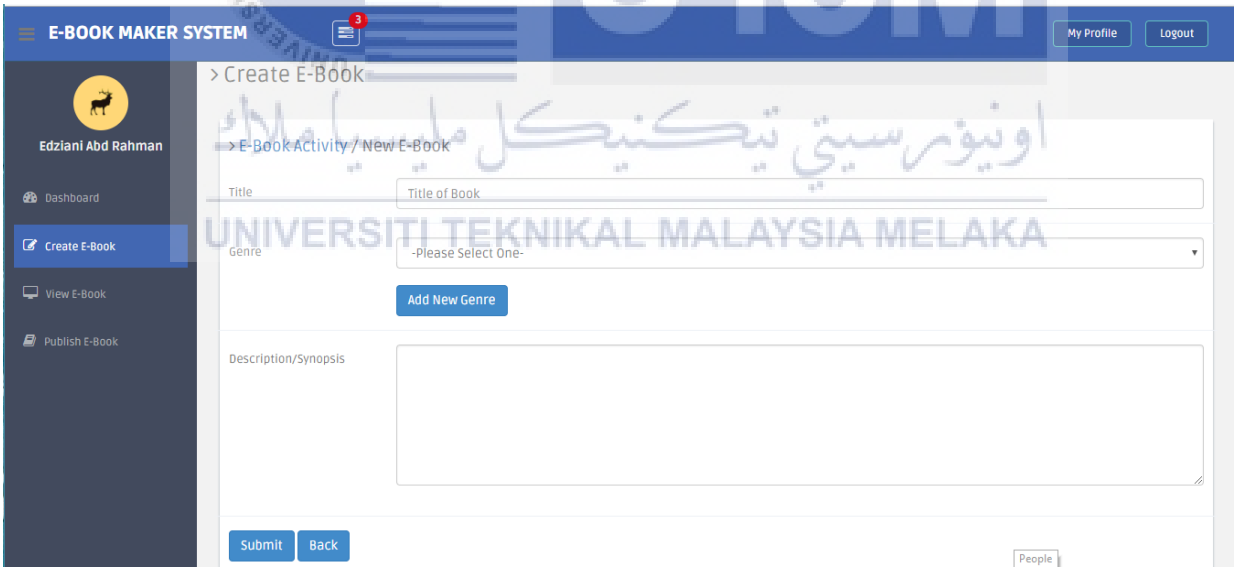


Figure 4.10: Create New Book form

This user interface allows author to create new book. Author need to enter the title of book, genre of book, year and description/synopsis. Author also can insert a new genre if the type of genre is not available on the genre section.

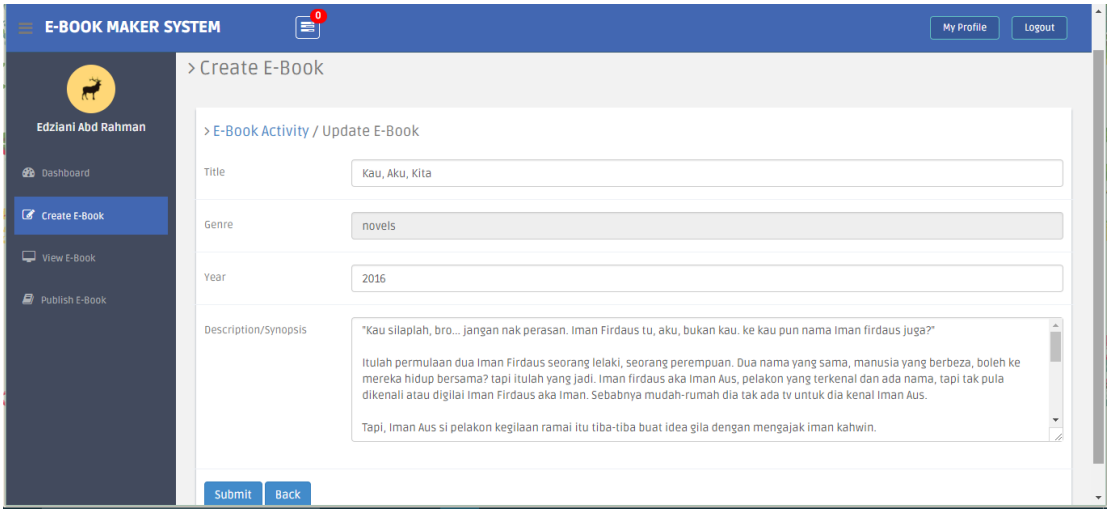


Figure 4.11: Update Book form

This form allows author to update the book. The information that can be updated are title of book, year and description/synopsis.

Figure 4.12: Table of Content

This user interface allows author to select the type of chapters and insert the title of new content. After author enter, author click the button Add and the data will be displayed in this user interface back. Author also can add new chapter if the type of chapter is not available in the chapter section.

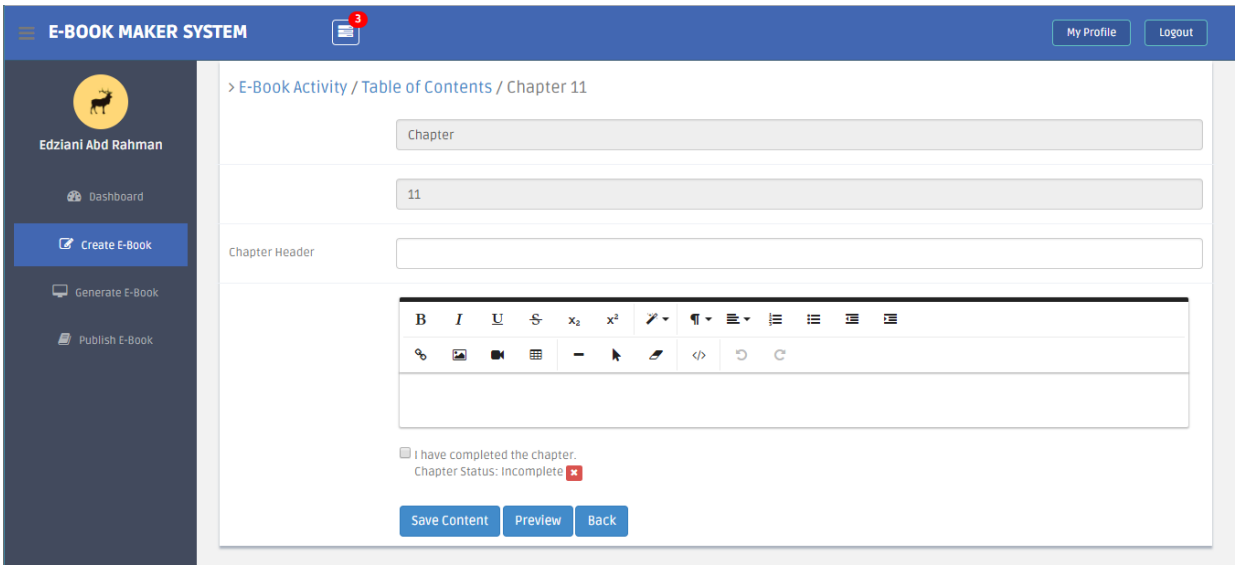


Figure 4.13: Write E-Book form

This is the user interface for author write the book. Author have to write the book in the text editor. Author also need to tick on the checkbox to confirm that the chapter is complete. Then, click the button Save Content to save the writing into the database.

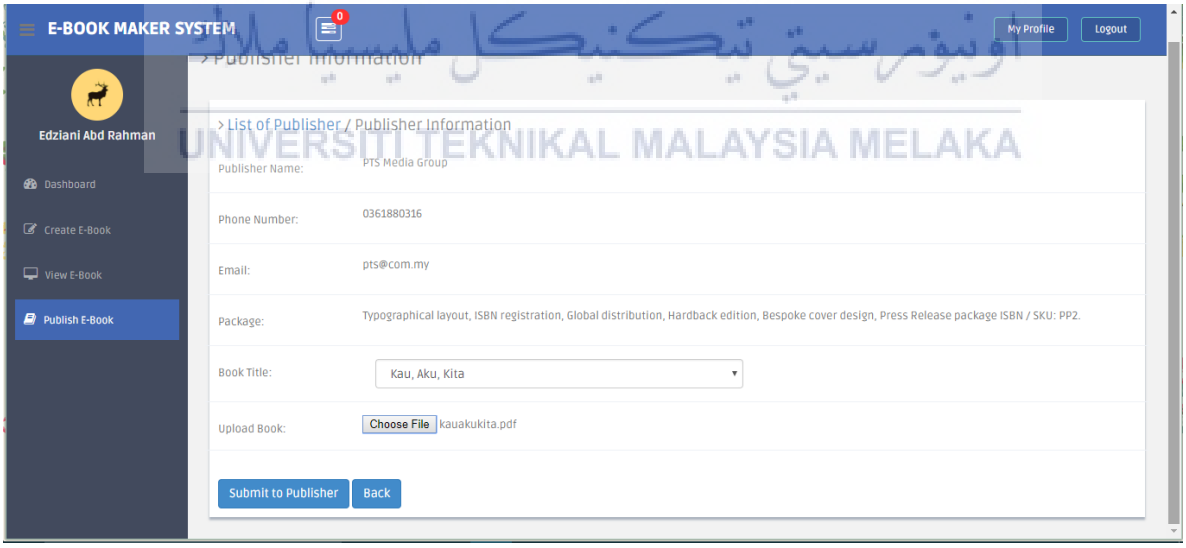


Figure 4.14: Sent PDF file to publisher form

Figure above shows the user interface for author to send the PDF file of book to submit to publisher. Author need to choose book title and choose the file to submit to publisher.

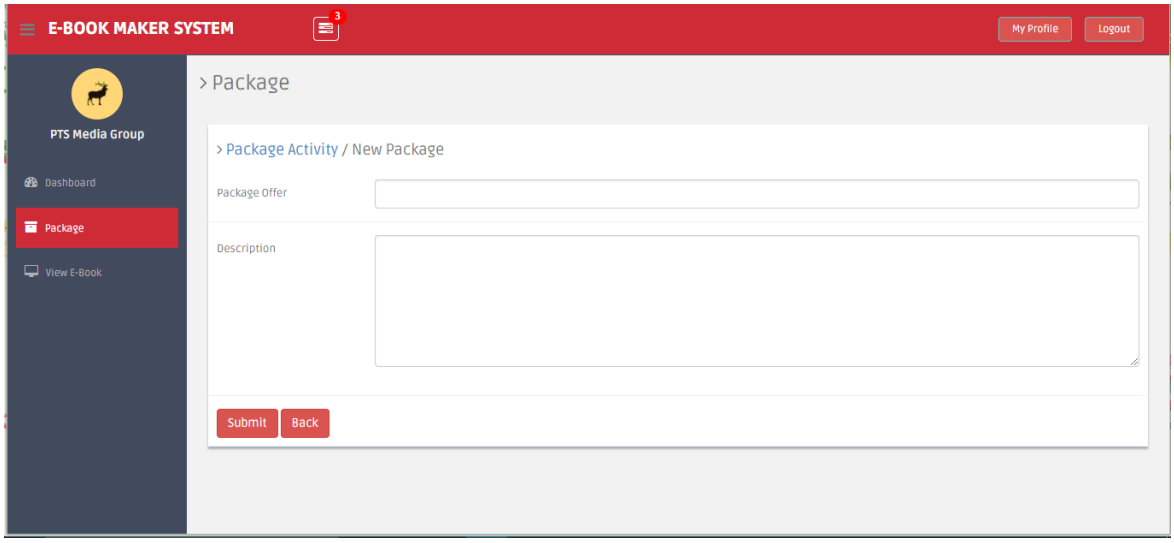


Figure 4.15: Insert New Package form

This user interface allows publisher to insert new package offer and description for author to choose in order to publish the book.

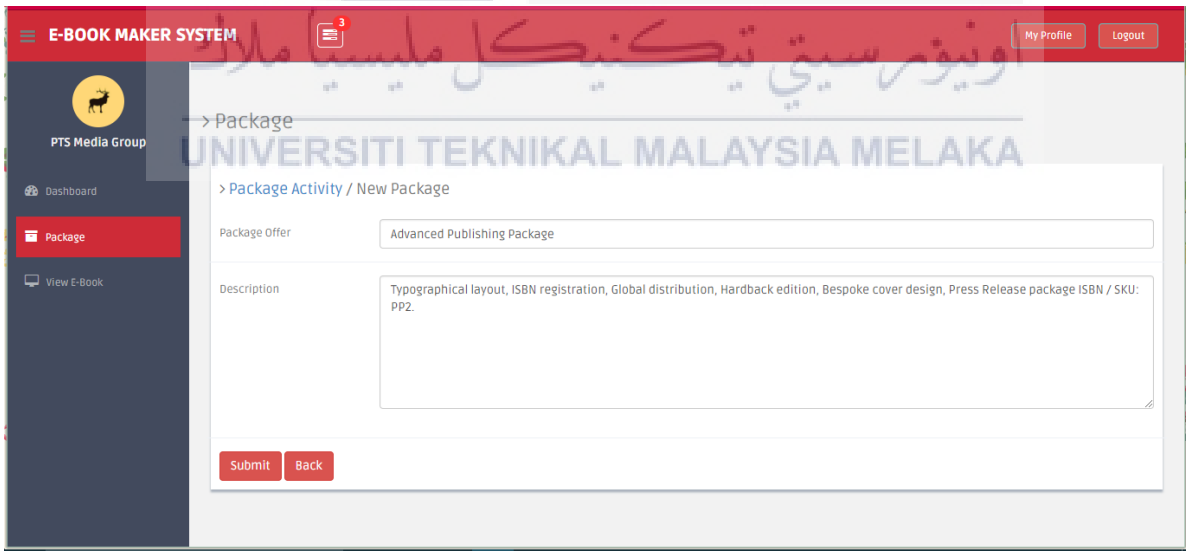


Figure 4.16: Update Package

Figure above shows the user interface for publisher to update the package.

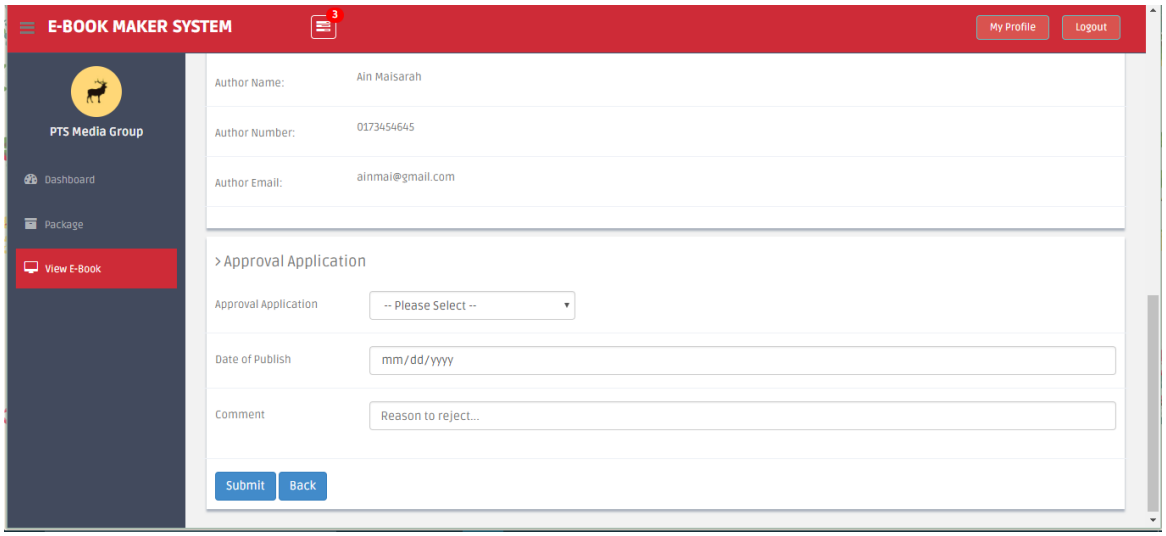


Figure 4.17: Approval Application form

In this user interface, publisher need select the approval application whether publisher approved or reject the book that have been submit by author. Publisher also need to enter the date of publish. If publisher reject the application, publisher need to give any comment about the book.

4.2.2.3 Output Design

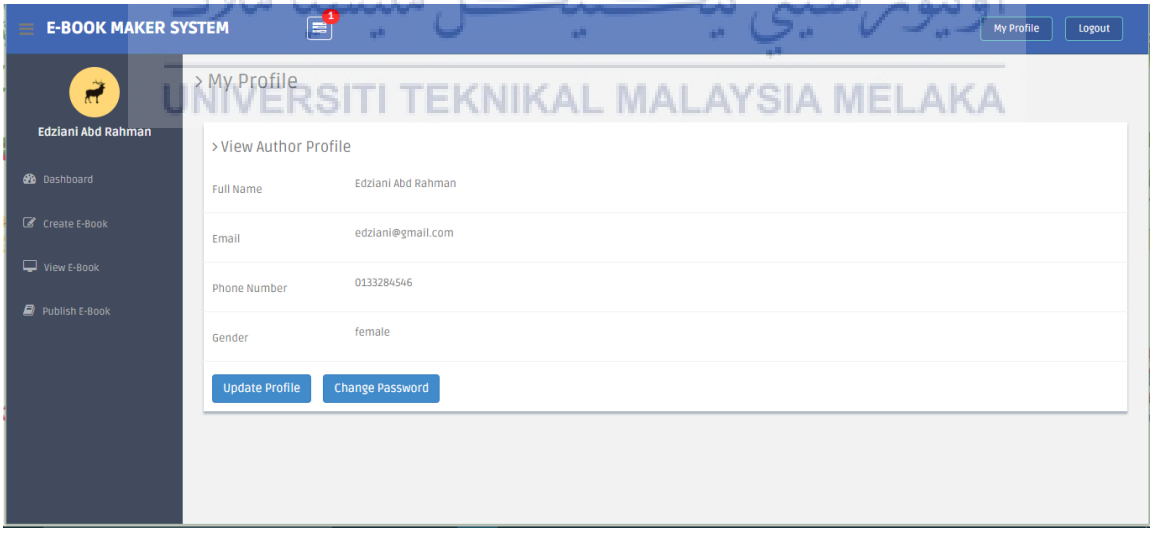


Figure 4.18: My Profile screen

In this user interface display the general information about author such as full name, email address, phone number and gender.

Figure 4.19: View Book by Chapter screen

This user interface will display the content of book that have been write by author. This screen will display the content of book by chapter.

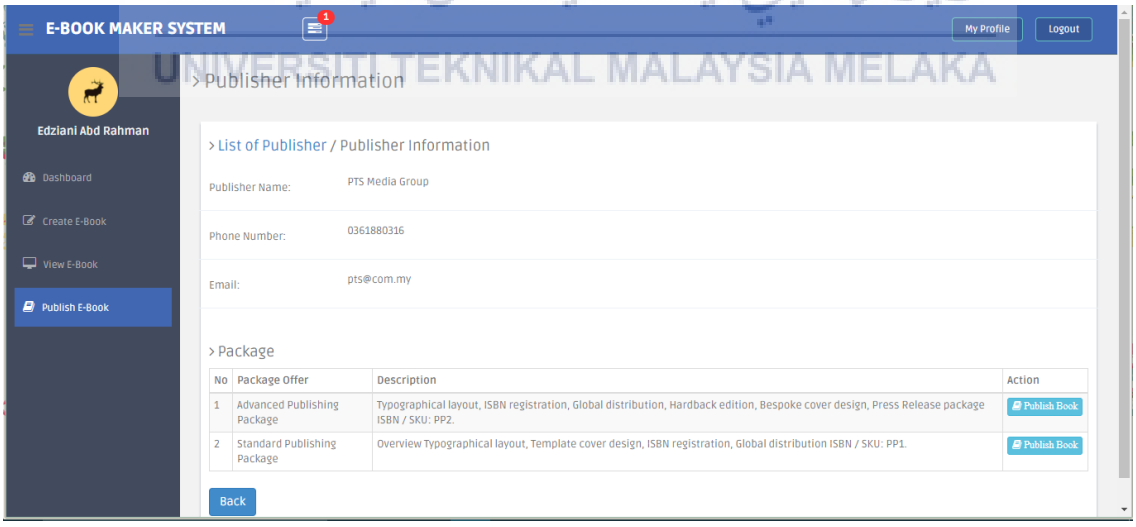


Figure 4.20: Publisher Information screen

This user interface will display the publisher information and package that publisher have offered.

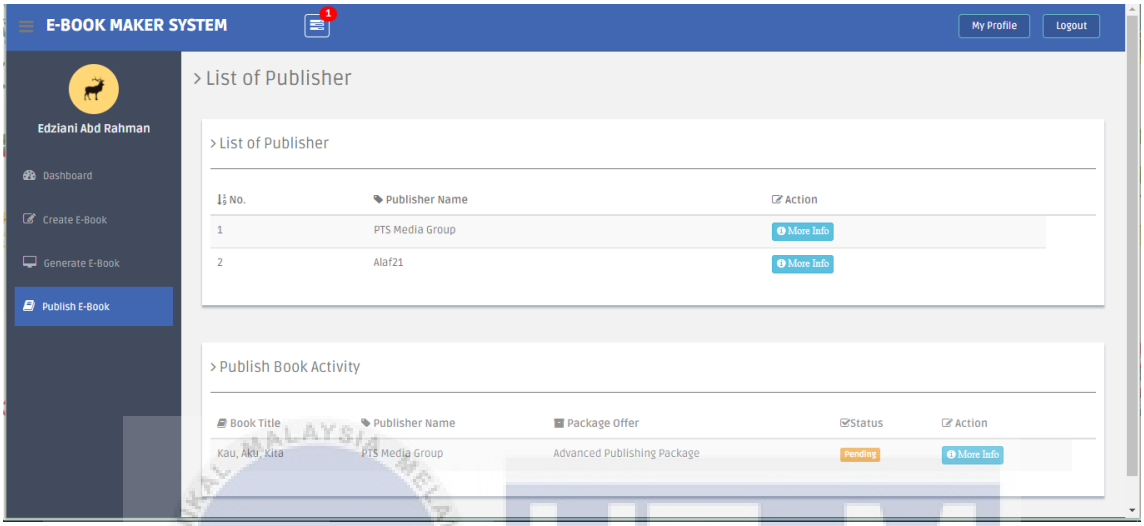


Figure 4.21: Publish Book Activity screen

This user interface will display the publish book activity. After author submit the book to publisher, author can check the status in this user interface whether the book is approved, reject or still pending by publisher.

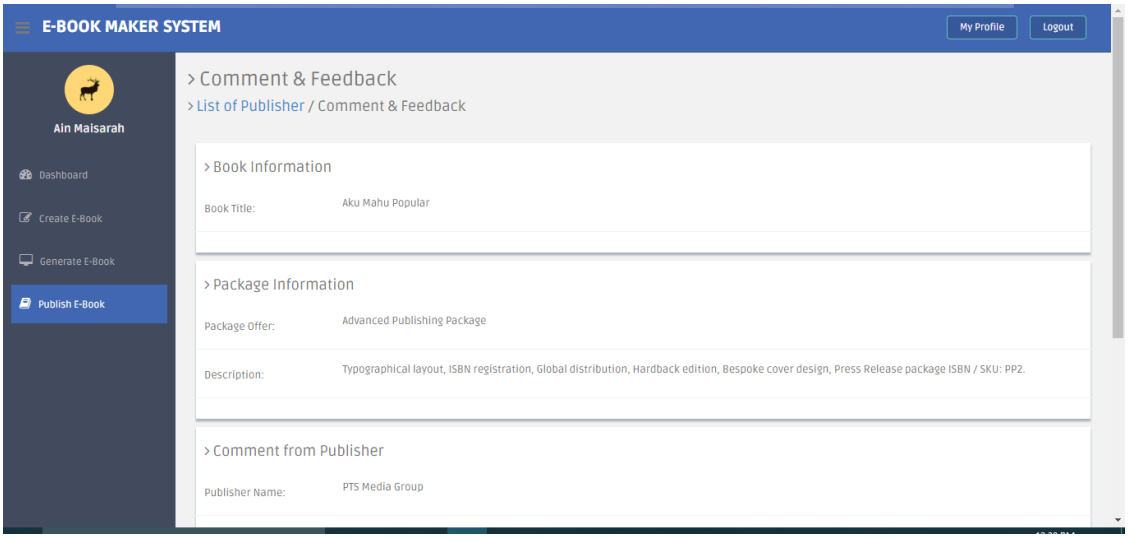


Figure 4.22: Comment & Feedback screen

This user interface will display the book information, package information, comment from publisher if publisher reject the book and the comment from readers when book is published.

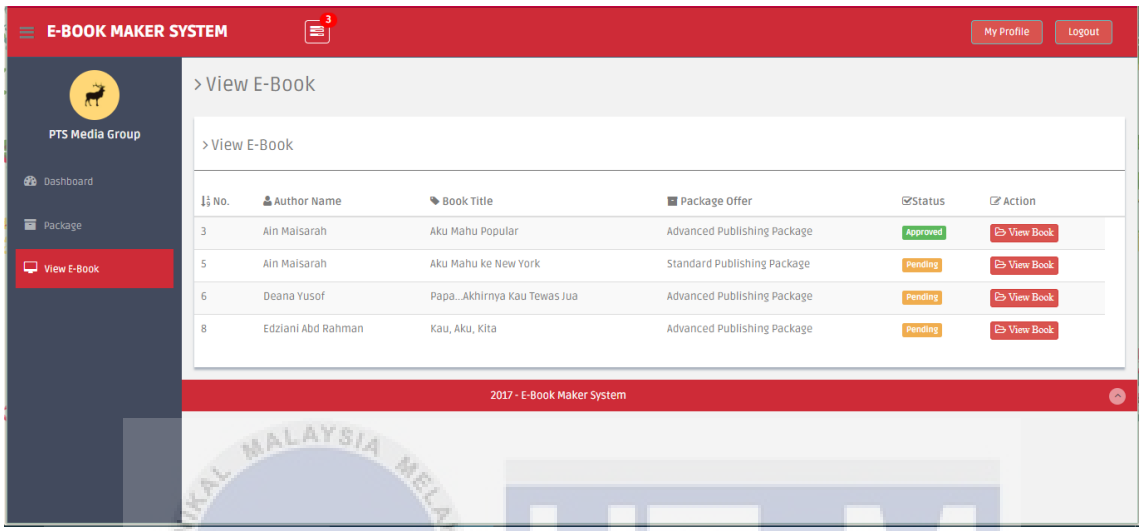


Figure 4.23: View E-Book screen

This user interface will display the book that have been submitted by author. Also it will display the status of the application.

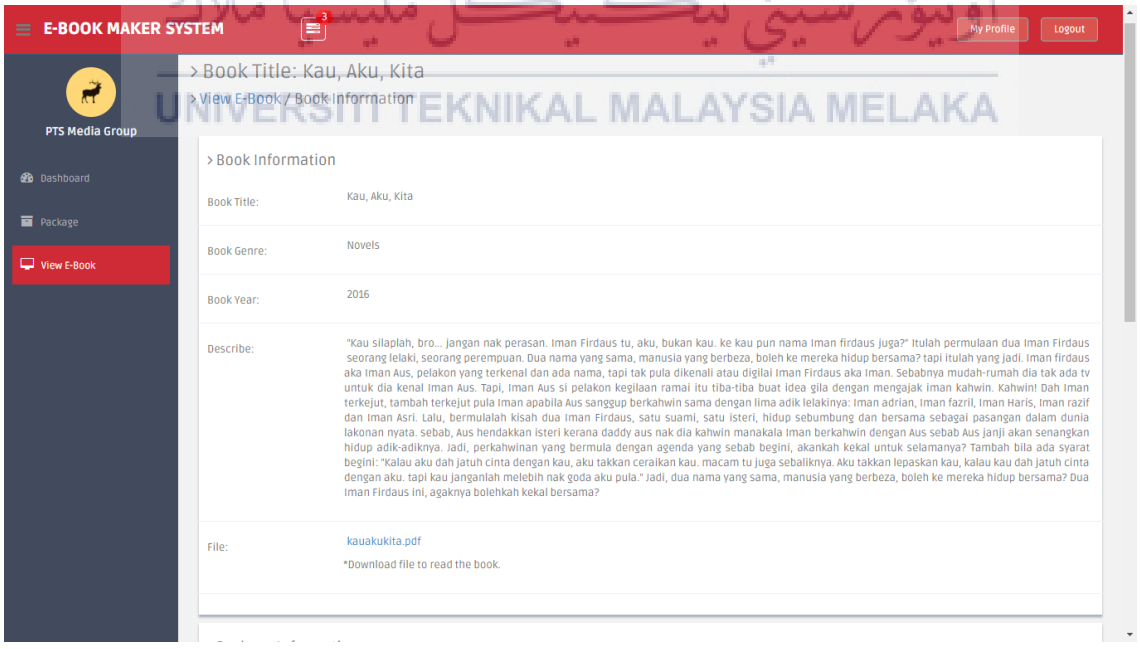


Figure 4.24: Book Information screen

This user interface will display the book information, author information and package that have been choose by author. Publisher can read the book by download the PDF file that have been submit by author.

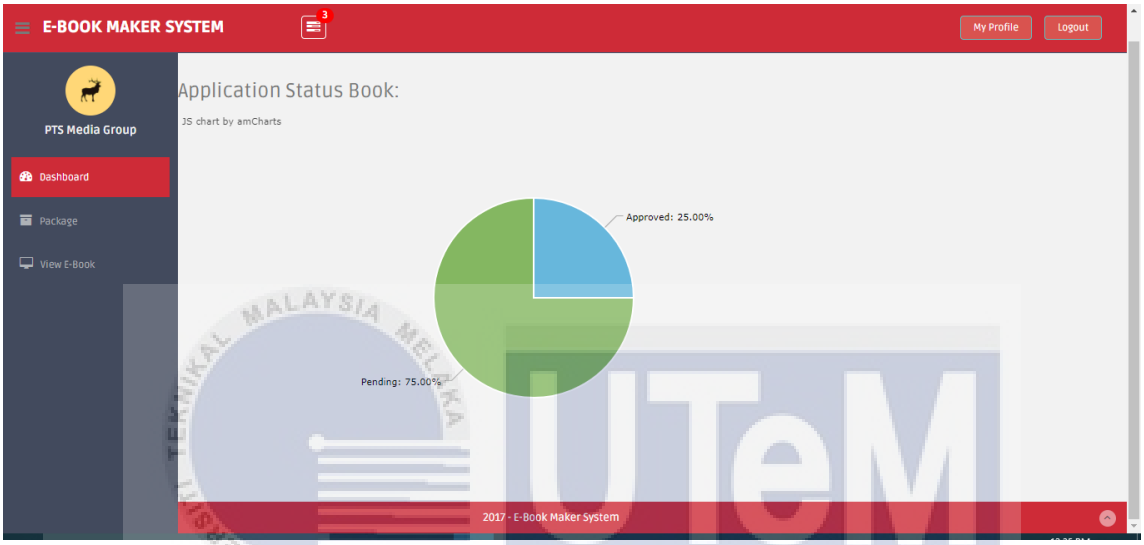


Figure 4.25: Report

This user interface will display the pie chart of approval application. The approval application is based on author submit the book to publisher.

4.2.3 Database Design

4.2.3.1 Conceptual and Logical Database Design

4.2.3.1.1 Entity Relationship Diagram

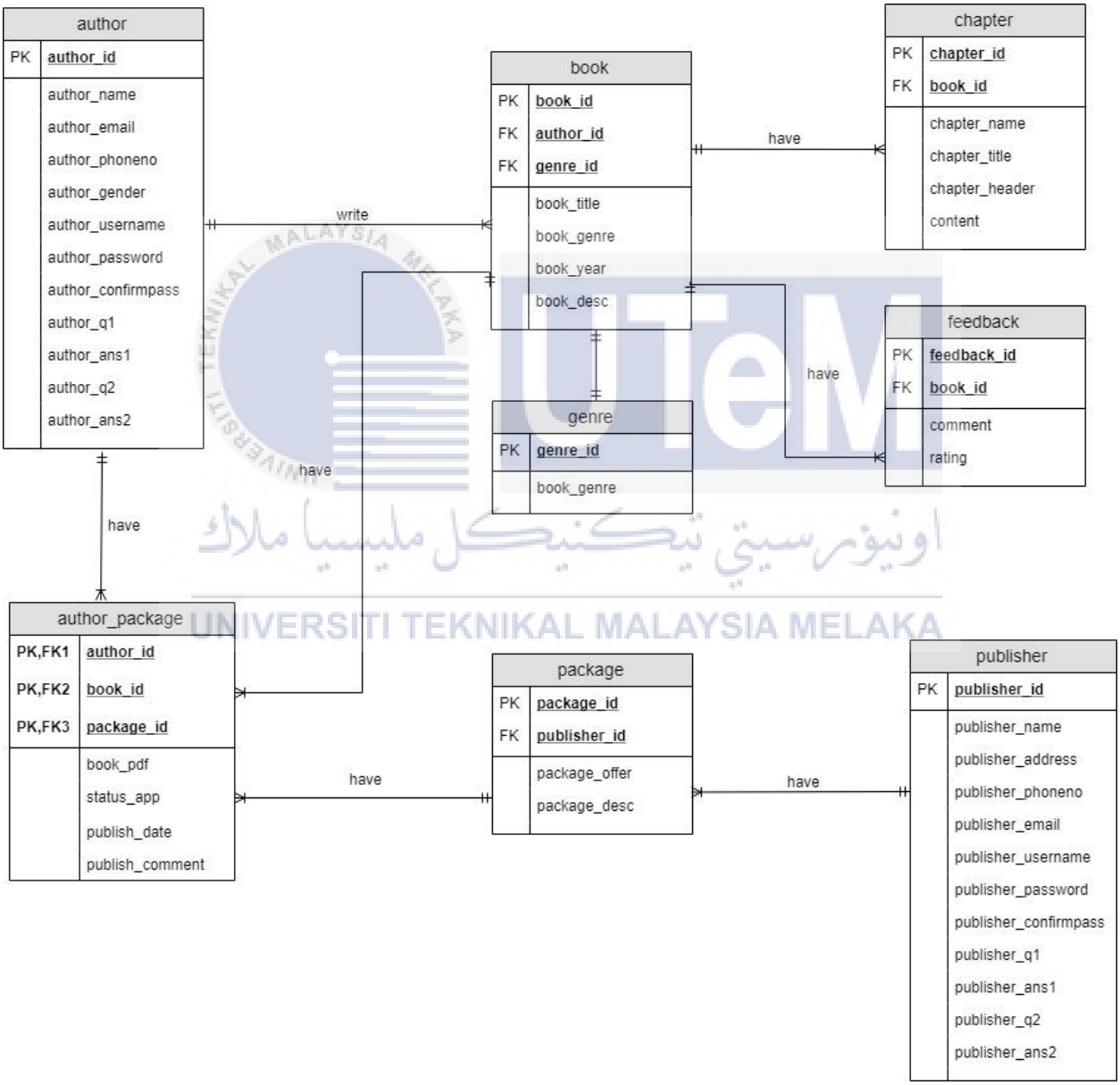


Figure 4.26: Entity Relationship Diagram

The figure above shows the Entity Relationship Diagram (ERD) for E-Book Maker System. It shows the relationships of every table and the process of how the system works.

4.2.3.1.2 Business Rules

1. An author can write one or more books.
2. A book must be written by one or more authors.
3. A publisher can have one or more packages.
4. A package must have one or more publisher.
5. A book can have one or more chapters.
6. A chapters must have by one or more books.
7. An author can have one or more author_package.
8. An author_package must have by one or more author.
9. A book can have one or more author_package.
10. An author_package must have by one or more books.
11. A package can have one or more author_package.
12. An author_package must have by one or more package.
13. A book can have one or more feedback.
14. A feedback must have by one or more book.
15. A book has only one genre.
16. A genre only has one book.

4.2.3.1.3 Data Dictionary

Table 4.1: Data Dictionary for all table

Attribute Name	Content Name	Data Type	Length	PK/FK
author_id	Author Id	INTEGER	100	PK
author_name	Author Name	VARCHAR	255	
author_email	Author Email	VARCHAR	255	
author_phoneno	Author Phone Number	VARCHAR	255	
author_gender	Author Gender	VARCHAR	100	
author_username	Author Username	VARCHAR	255	
author_password	Author Password	VARCHAR	255	
author_confirmpass	Author Confirm Password	VARCHAR	255	
publisher_id	Publisher Id	INTEGER	100	PK
publisher _name	Publisher Name	VARCHAR	255	
publisher_address	Publisher Address	VARCHAR	255	
publisher _phoneno	Publisher Phone Number	VARCHAR	255	
publisher _faxno	Publisher Fax Number	VARCHAR	255	
publisher _email	Publisher Email	VARCHAR	255	
publisher _username	Publisher Username	VARCHAR	255	
publisher _password	Publisher Password	VARCHAR	255	
publisher _confirmpass	Publisher Confirm Password	VARCHAR	255	
book_id	Book Id	INTEGER	100	PK
book_title	Book Title	VARCHAR	255	
book_genre	Book Genre	VARCHAR	255	
book_year	Book Year	VARCHAR	10	
book_desc	Book Description or Synopsis	TEXT	-	
genre_id	Genre Id	INTEGER	100	FK
author_id	Author Id	INTEGER	100	FK

chapter_id	Chapter Id	INTEGER	100	PK
chapter_name	Chapter Name	TEXT	-	
chapter_title	Chapter Title	TEXT	-	
content	Content of the Book	TEXT	-	
book_id	Book Id	INTEGER	100	FK
package_id	Package Id	INTEGER	100	PK
package_offer	Package Offer	VARCHAR	255	
package_desc	Package Description	TEXT	-	
publisher_id	Publisher Id	INTEGER	100	FK
author_id	Author Id	INTEGER	100	PK/FK
book_id	Book Id	INTEGER	100	PK/FK
package_id	Package Id	INTEGER	100	PK/FK
book_pdf	File PDF	TEXT	-	
status_app	Application status	VARCAR	255	
feedback_id	Feedback Id	INTEGER	100	PK
comment	Comment	TEXT	-	
rating	Rating	VARCHAR	255	
book_id	Book Id	INTEGER	100	FK
genre_id	Genre Id	INTEGER	100	PK
book_genre	Book Genre	VARCHAR	255	

4.3 Detailed Design

4.3.1 Software Design

4.3.1.1 Login Account

Description:

To log in the user to their account to use the system.

Input:

Username, Password

Output:

None

Pseudo code:

1. Start.
2. Log in page is displayed.
3. User enters the Username and Password in the text field.
4. Username and Password entered is validated.
5. The dashboard screen is displayed.
6. End.

Report Screen:

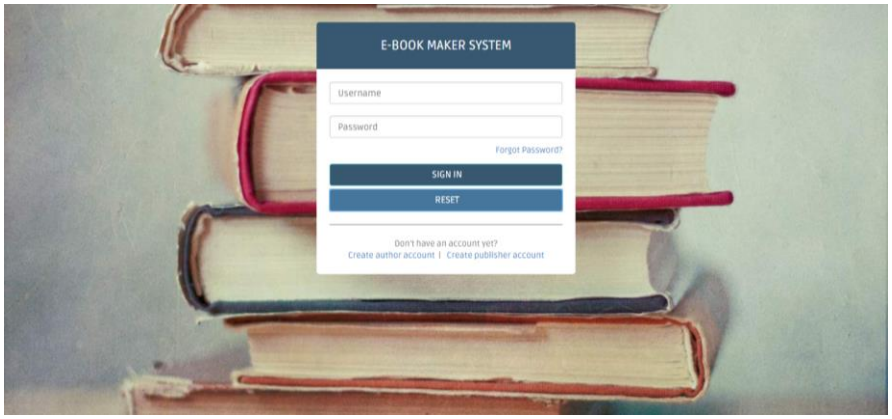


Figure 4.27: Log In Form

4.3.1.2 Register Account

Description:

To register the user account to be used in the system.

Input:

Author – Full Name, Email, Phone Number, Gender, Username, Password and Confirm Password.

Publisher – Publisher Name, Address, Phone Number, Fax Number, Email, Username, Password and Confirm Password.

Output:

User account created.

Pseudo code:

1. Start.
2. Registration form is displayed.
3. User enters the required information.
4. Information entered is validated.
5. Account is created on success.
6. The log in page is displayed.
7. End.

Report Screen:

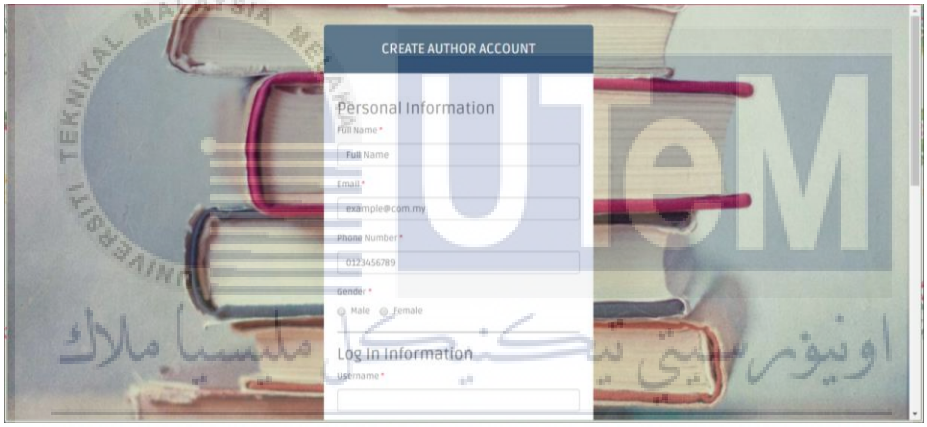


Figure 4.28: Register Form for author

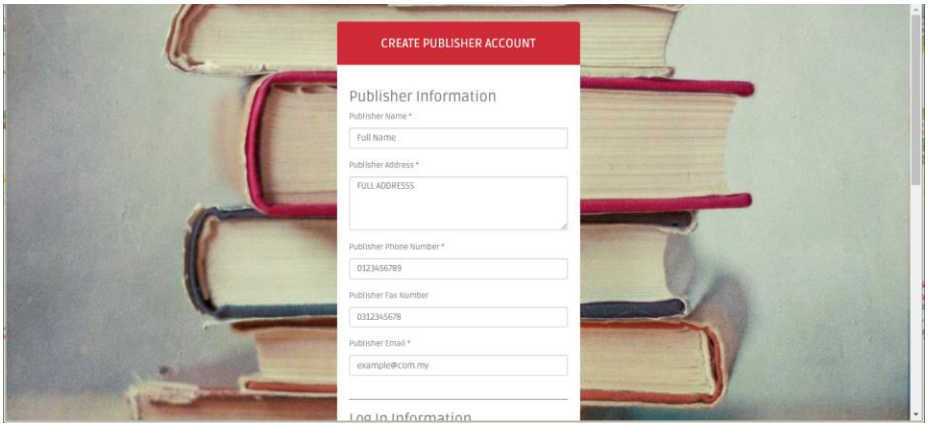


Figure 4.29: Register Form for Publisher

4.3.1.3 Create New Book

Description:

To create the new book that will be write by the author.

Input:

Title of Book, Genre, Year and Description or Synopsis.

Output:

New book created.

Pseudo code:

1. Start.
2. Create New E-Book form is displayed.
3. Author enters the required information such as title, genre, year and description.
4. Information entered is validated.
5. New book is created on success.
6. The new book will be displayed in the list of E-Book interface.
7. End.

Report Screen:

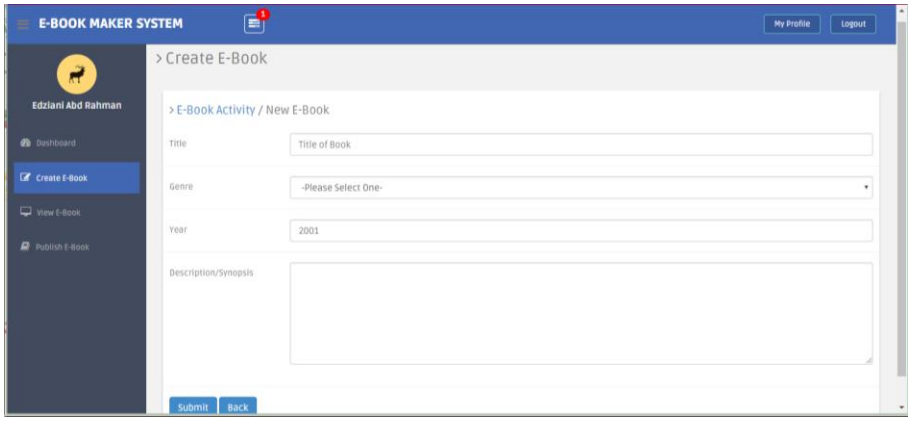


Figure 4.30: Create New E-Book

4.3.1.4 Write Book

Description:

To get the input of author writing the book.

Input:

Type of chapters, title of content, title header and content

Output:

None

Pseudo code:

1. Start.
2. Table of Content will be displayed.
3. Author enters the type of chapter and title of content.
4. Then it will display at the Table of Content.

5. Click one of the chapter that author wants to write
6. Write book form will be displayed.
7. Author enters the title header and content.
8. Information entered is validated.
9. End.

Report Screen:

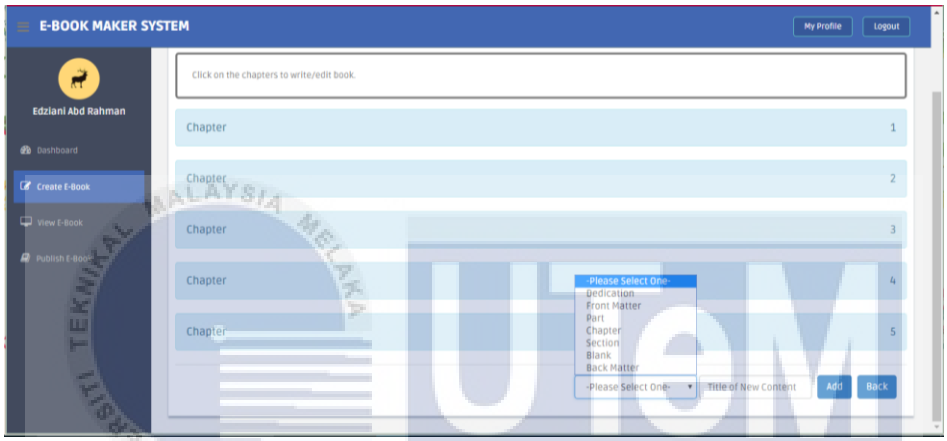


Figure 4.31: Table of Content Form

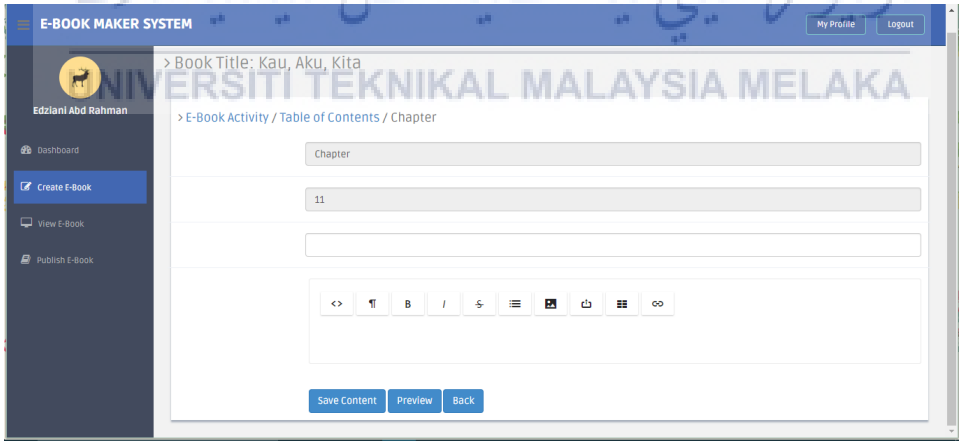


Figure 4.32: Write Book Form

4.3.1.5 Generate Book

Description:

To generate the book convert to PDF file.

Input:

None

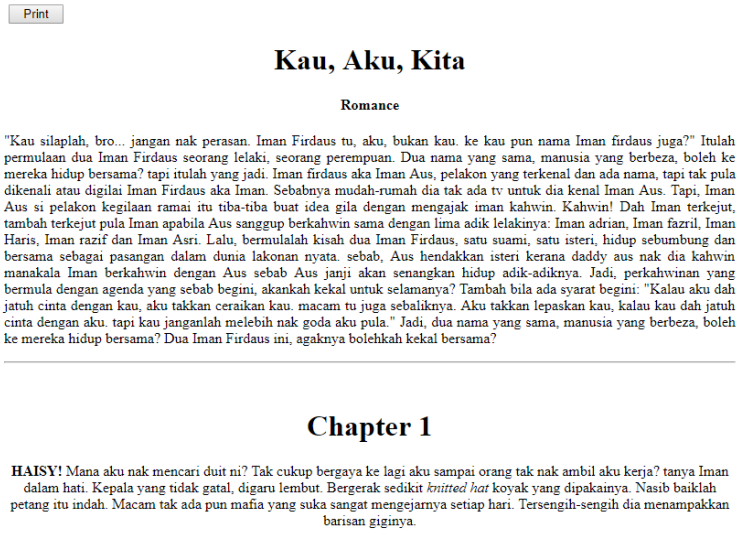
Output:

PDF file

Pseudo code:

1. Start.
2. Author finish writing the book.
3. Author click the button generate PDF
4. PDF file will be displayed
5. Author download the PDF file to submit to publisher
6. End.

Report Screen:


Print

Kau, Aku, Kita

Romance

"Kau silaplah, bro... jangan nak perasan. Iman Firdaus tu, aku, bukan kau. ke kau pun nama Iman firdaus juga?" Itulah permulaan dua Iman Firdaus seorang lelaki, seorang perempuan. Dua nama yang sama, manusia yang berbeza, boleh ke mereka hidup bersama? tapi itulah yang jadi. Iman firdaus aka Iman Aus, pelakon yang terkenal dan ada nama, tapi tak pula dikenali atau digilai Iman Firdaus aka Iman. Sebabnya mudah-rumah dia tak ada tv untuk dia kenal Iman Aus. Tapi, Iman Aus si pelakon kegilaan ramai itu tiba-tiba buat idea gila dengan mengajak iman kahwin. Kahwin! Dah Iman terkejut, tambah terkejut pula Iman apabila Aus sanggup berkahwin sama dengan lima adik lelakinya: Iman adrian, Iman fazril, Iman Haris, Iman razif dan Iman Asri. Lalu, bermulalah kisah dua Iman Firdaus, satu suami, satu isteri, hidup sebumbung dan bersama sebagai pasangan dalam dunia lakonan nyata. sebab, Aus hendakkan isteri kerana daddy aus nak dia kahwin manakala Iman berkahwin dengan Aus sebab Aus janji akan senangkan hidup adik-adiknya. Jadi, perkahwinan yang bermula dengan agenda yang sebab begini, akankah kekal untuk selamanya? Tambah bila ada syarat begini: "Kalau aku dah jatuh cinta dengan kau, aku takkan ceraikan kau. macam tu juga sebaliknya. Aku takkan lepaskan kau, kalau kau dah jatuh cinta dengan aku. tapi kau janganlah melebih nak goda aku pula." Jadi, dua nama yang sama, manusia yang berbeza, boleh ke mereka hidup bersama? Dua Iman Firdaus ini, agaknya bolehkah kekal bersama?

Chapter 1

HAISY! Mana aku nak mencari duit ni? Tak cukup bergaya ke lagi aku sampai orang tak nak ambil aku kerja? tanya Iman dalam hati. Kepala yang tidak gatal, digaru lembut. Bergerak sedikit *knitted hat* koyak yang dipakainya. Nasib baiklah petang itu indah. Macam tak ada pun mafia yang suka sangat mengejarnya setiap hari. Tersengih-sengih dia menampakkan barisan giginya.

Figure 4.33: Generate PDF

4.3.1.6 Publish Book

Description:

To publish the book to the readers.

Input:

Application approval, date of publish, comment

Output:

Status application approval

Pseudo code:

1. Start.

2. Author submit book to publisher.
3. Publisher read the book.
4. Publisher give status for application approval.
5. Publisher publish the book
6. End.

Report Screen:

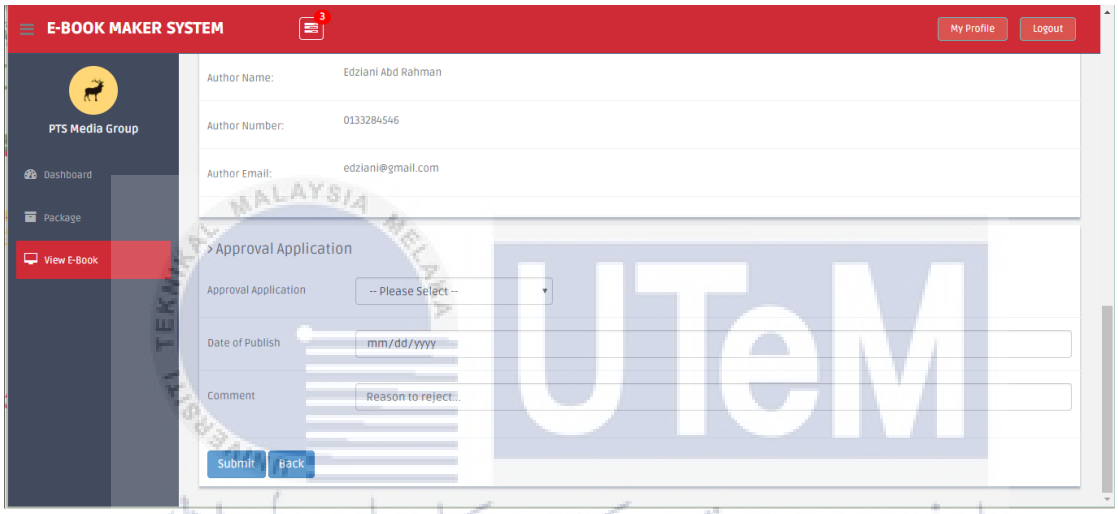


Figure 4.34: Publish Book

4.3.1.7 Readers Comment

Description:

To give comment about the book.

Input:

Comment

Output:

None

Pseudo code:

1. Start.
2. Readers read the book.
3. Readers give comment about the book.
4. End.

Report Screen:

Figure 4.35: Readers Comment

4.3.2 Physical Database Design

Table: author

```
CREATE TABLE author (
author_id INT (100) NOT NULL,
author_name VARCHAR (255) NOT NULL,
author_email VARCHAR (255) NOT NULL,
author_phoneno VARCHAR (255) NOT NULL,
author_gender VARCHAR (255) NOT NULL,
author_username VARCHAR (255) NOT NULL,
author_password VARCHAR (255) NOT NULL,
author_confirmpass VARCHAR (255) NOT NULL,
PRIMARY KEY (author_id)
);
```


Table: publisher

```
CREATE TABLE publisher (
publisher_id INT (100) NOT NULL,
publisher_name VARCHAR (255) NOT NULL,
publisher_address VARCHAR (255) NOT NULL,
publisher_phoneno VARCHAR (255) NOT NULL,
publisher_faxno VARCHAR (255) NOT NULL,
publisher_email VARCHAR (255) NOT NULL,
```

publisher_username VARCHAR (255) NOT NULL,

publisher_password VARCHAR (255) NOT NULL,

publisher_confirmpass VARCHAR (255) NOT NULL,

PRIMARY KEY (publisher_id)

);

Table: book

CREATE TABLE book (

book_id INT (100) NOT NULL,

book_title VARCHAR (255) NOT NULL,

book_genre VARCHAR (255) NOT NULL,

book_year VARCHAR (10) NOT NULL,

book_desc TEXT NOT NULL,

author_id INT (100) NOT NULL,

genre_id INT (100) NOT NULL,

PRIMARY KEY (book_id),

FOREIGN KEY (author_id) REFERENCES author (author_id),

FOREIGN KEY (genre_id) REFERENCES genre (genre_id)

);

Table: chapter

CREATE TABLE chapter (

chapter_id INT (100) NOT NULL,

```
chapter_name TEXT NOT NULL,
chapter_title TEXT NOT NULL,
chapter_header VARCHAR (255) NOT NULL,
content TEXT NOT NULL,
book_id INT (100) NOT NULL,
PRIMARY KEY (chapter_id),
FOREIGN KEY (book_id) REFERENCES book (book_id)
);
```

Table: package

```
CREATE TABLE package (
package_id INT (100) NOT NULL,
package_offer VARCHAR (255) NOT NULL,
package_desc TEXT NOT NULL,
publisher_id INT (100) NOT NULL,
PRIMARY KEY (package_id),
FOREIGN KEY (publisher_id) REFERENCES publisher (publisher_id)
);
```


Table: author_package

```
CREATE TABLE author_package (
author_id INT (100) NOT NULL,
book_id INT (100) NOT NULL,
```

```
package_id INT (100) NOT NULL,
book_pdf TEXT NOT NULL,
status_app VARCHAR (255) NULL,
PRIMARY KEY (author_id, book_id, package_id),
FOREIGN KEY (author_id) REFERENCES author (author_id),
FOREIGN KEY (book_id) REFERENCES book (book_id),
FOREIGN KEY (package_id) REFERENCES package (package_id)
);
```

Table: feedback

```
CREATE TABLE feedback (
feedback_id INT (100) NOT NULL,
comment TEXT NOT NULL,
rating VARCHAR (50) NOT NULL,
book_id INT (100) NOT NULL,
PRIMARY KEY (feedback_id),
FOREIGN KEY (book_id) REFERENCES book (book_id)
);
```


Table: genre

```
CREATE TABLE genre (
genre_id INT (100) NOT NULL,
book_genre VARCHAR (255) NOT NULL,
```

```
PRIMARY KEY (genre_id)
);
```

4.4 Conclusion

As a conclusion, the design of the system had been discussed in this chapter which includes the high-level design whereby the elements such as the system architecture, user interface and database design have been created that will guide the development of the system. In the next chapter, the implementation phase will be discussed

CHAPTER V

IMPLEMENTATION

5.1 Introduction

The system implementation process is construction and delivery phases of the life cycle. Systems implementation is the construction of the new system and the delivery of that system into production. In this chapter, will discussing about how the system is implemented including the system development environment, system configuration management and the security characteristics of E-Book Maker System.

5.2 Software Development Environment Setup

Table 5.1: System Development Environment Setup

Development Environment	Description
Web-Based	-Windows 10 platform -XAMPP
Server-Side Development	-PhpMyAdmin -PHP

5.2.1 Software Architecture Setup

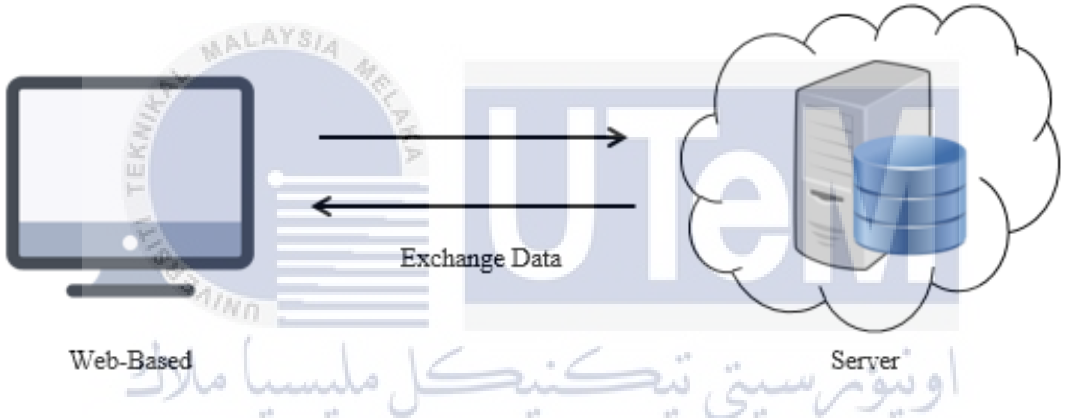


Figure 5.1: Software Architecture Setup

System development environment is programming tools that used to create the program or software product. In this project, develop one web-based application called E-Book Maker System. The system is developed for the author and publisher that want to write book and publish book online.

As for web-based application, Sublime Text 3 is used as text editor in order to write the coding. The text editor works well for writing and editing PHP codes as well as include some configuration on CSS. The system provides dynamic interface.

In addition, MySQL is used as database management system (DBMS). The user of this DBMS platform along with XAMPP server to run E-Book Maker System. XAMPP Server is an open source cross platform web server solution stack package developed by Apache.

5.2.2 Hardware Architecture Setup

Table 5.2: Hardware Architecture Setup

Machine	Hardware	Description
Desktop	Processor	Intel® Core™ i5-4200U CPU @ 1.60GHz 2.30GHz
	RAM	4.00 GB
	Hard Disk Drive	500GB

5.3 Software Configuration Management

System configuration management refers to configuration that must to be set up in order to make sure that the system can be developed successfully without missing of functionalities. It is essential to make sure that the system performs as intended. Configuration management has facilitated the management of system information which can bring some benefits such as performance improvement, reduce risk or correct defects. During the implementation of E-Book Maker System, there are some configuration that need to be done in order to get the system works properly.

5.3.1 Configuration Environment Setup

5.3.1.1 Install Text Editor

1) Download the Sublime Text 3 from https://www.sublimetext.com/3
2) Run the software installer and wait for the installation to finish.

5.3.1.2 Setup XAMPP Server

1) Download XAMPP Server from https://www.apachefriends.org/download.html
2) Run the software installer and wait for the installation to finish.
3) When the installation finish, run XAMPP Control Panel and click Start on both Apache and MySQL module and wait for the server to be served.

5.4 Implementation Status

In this part, it will explain about the status of development for each component or module contains in the E-Book Maker System. It will describe about the name of module, the description, the duration of completion and the status. The table below shows the implementation status of the system.

Table 5.3: Implementation Status

Name	Description	Duration	Status
Register, Log In, Recovery, Profile	For registration, user must enter the personal information, log in information and two (2) security questions. This to ensure that user enter all the required information before log in into the system.	2 days	Functional
	For log in, user must enter username and password in the log in form. This to ensure that only the authorized user can access the system.	2 days	Functional
	For recovery, user must enter the security question and answer that have been answered during registration. This to ensure that user can recover their account if user forgot their password.	3 days	Functional
	For profile, user can view their profile and can update their general information such as email and phone number. Also, user can changed their password.	2 days	Functional
Content E-Book	Author must enter the title of book, genre, year and description/synopsis. Author also can update the title of book, year and description/synopsis. After that, author need to insert the table of content of book. Then author can write the book content. Author also can insert the image.	5 days	Functional
Generate E-Book	Author can generate the book in PDF. Author also can create a table of content of book.	5 days	Functional

Publish E-Book	Author can choose which publisher and package that their want and submit their book transcript to publisher. Publisher then give application status to author and submit the book to readers page.	5 days	Functional
Readers Comment	Readers can read the book online and can give comment about the book. Readers also can give rating to the book.	2 days	Functional
Generate Report	Author and publisher can generate the report	2 days	Functional

5.5 Conclusion

In conclusion, the implementation phase was discussed in this chapter whereby all of the software is coming to fruition after all the requirement gathering, conceptualizing and designing activities that have been done. This chapter outlined the development environment and configuration management activities that are important to ensure that the development activities will proceed smoothly.

CHAPTER VI

TESTING

6.1 Introduction

This chapter will discuss about the testing activities that are carried out throughout the development of E-Book Maker System. Testing is very important part because it is a tedious process. Through an effective and extensive testing, the system that id produced will be more complete and free from avoidable flaws and bugs.

6.2 Test Plan

Software project test plan details about the objectives, scope and the focus of the testing activities that will be carried out during the testing process. The test plan is also used to outline the overall activities that will be done during the testing and server as a guide in which the testers may refer to ensure that their testing activities are following the objectives of the test plan.

6.2.1 Test Organization

Table 6.1: Test Organization

Role	Responsibility
Test Planner	The person in charge of planning the testing activities as well as to prepare the test script and testing strategy.
Programmer	The developer of the application that is in charge of conducting test before application deployment such as unit testing, integration testing and functional testing.

6.2.2 Test Environment

Table 6.2: Test Environment

Component	Requirement
Hardware	Laptop Printer
Software	-

6.2.3 Test Schedule

Table 6.3: Test Schedule

Activity	Description
Unit Testing	Testing that is conducted on each individual module of the system.
Integration Testing	Testing is done when the modules are integrated in a whole system.
Functional Testing	Testing that is done to assess that the application developed satisfies the requirement.

6.3 Test Strategy

Testing strategy is developed to ensure the system and application is tested effectively and efficiently to find as many flaws and bugs as possible. Make it short, the testing strategies that were chosen to conduct the test were the bottom-up approach, white-box testing and black-box testing.

6.3.1 Classes of tests

1. Bottom-Up Approach

   Bottom-up approach of testing of individual component of the system first then work up towards a higher level whereby the interrelated components will be integrated with one another to form a complete system. This approach was taken to ensure that the testing done will be extensive and that individual components will

be tested before integrating them together to avoid potential bugs to be found later that will be more difficult to diagnose and repair.

2. White-Box Testing

White-box testing is the method to test a software product based on the analysis of the internal structure of the code. This type of test is typically done by the developers of the system as it requires knowledge of the system itself in terms of the code. The types of white-box techniques that are used to test the system are:

Table 6.4: White-Box Techniques

Techniques	Description
Statement Coverage	Testing to try to ensure that 100% of the statement in the code are executed at least once.
Data Flow Testing	Testing to select the paths to the control flow of the system in which variables value are defined and used.
Decision Coverage	Testing to try to cover 100% of the decision condition such as if-else, for loops and others at least once.

3. Black-Box Testing

Black-box testing is the method to test a software product without knowing this internal structure of code. The testing is done based on the user point of view only with what the software is supposed to do rather than how it does it. There are several types of black-box testing techniques:

Table 6.5: Black-Box Technique

Techniques	Description
Error Guessing	Testing based on the experience of the tester to guess where an error can occur.
Use Case Testing	Testing to help identify test cases for an entire system coverage.
Equivalence Class Partitioning	Testing of test cases in which the test cases are divided into partitions such as valid and invalid input.

6.4 Test Design

Test design is created to display in detail about the activity running during the testing phase. The test design will explain about what module is being tested, what data is being tested, the expected result when the certain action is running and what is the exact result of the testing. The testing is very important to make sure the system run according to user requirements.

6.4.1 Test Description

Table 6.6: Test Description

Test Number	Action	Expected Result	Actual Result (Pass/Fail)
REGISTRATION_01	1. Enter all the required information. 2. Press the register button.	1. Create an account based on the information entered. 2. Display that the account is created.	Pass
REGISTRATION_02	1. Enter all the valid information and leave certain fields blank. 2. Press the register button.	1. Account is not created. 2. Display the required field and its condition.	Pass
LOGIN_01	1. Enter correct username and password. 2. Press the Login button.	1. Login successful. 2. Display main screen.	Pass
LOGIN_02	1. Enter all the invalid and incorrect username and password. 2. Press the Login button.	1. Login failed. 2. Display that username and password is incorrect	Pass
LOGIN_03	1. Enter valid username and incorrect password.	1. Login failed.	Pass

	2. Press the Login button.	2. Display that the username and password is incorrect.	
LOGIN_04	1. Leave certain field blank. 2. Press the Login button.	1. Login failed. 2. Display that the username and password is incorrect.	Pass
PASSWORD_01	1. Enter the username to check that if the username exists or not. 2. Press the Submit button.	1. Check successful. 2. Go to next page to answer security question.	Pass
PASSWORD_02	1. Enter the incorrect username. 2. Press the Submit button.	1. Check unsuccessful. 2. Go to next page but cannot answer the security question.	Pass
PASSWORD_03	1. Enter the correct answer for security question. 2. Press the Submit button.	1. Answer is correct and successful. 2. Go to next page to update the password.	Pass
PASSWORD_04	1. Enter the incorrect answer for security question. 2. Press the Submit button.	1. Answer is incorrect and unsuccessful. 2. Display that the answer is wrong, please answer again.	Pass
WRITEBOOK_01	1. Enter the required information. 2. Press the Submit button.	1. Book successful created. 2. Display that the book is successfully created.	Pass
WRITEBOOK_02	1. Leave certain fields blank 2. Press the Submit button.	1. Book is not created. 2. Display that fill all the required blank.	Pass
PACKAGE_01	1. Enter all the required information. 2. Press the Submit button.	1. Package successful created. 2. Display that the package is successfully created.	Pass

PACKAGE_02	1. Leave certain fields blank. 2. Press the Submit button.	1. Package is not created. 2. Display that fill all the required blank.	Pass
PUBLISHER_01	1. Select the approval application and enter the date of publish. 2. Press the Submit button.	1. Book is published.	Pass
PUBLISHER_02	1. Select the approval application and did not enter the date of publish. 2. Press the Submit button.	1. Book is not published. 2. Display that fill the date of publish.	Pass

6.5 Test Results and Analysis

Through the test that have been carried out, it is found that the modules that were developed worked as intended. Prior to the formal testing process, smaller tests were conducted continuously during development to find defects and fix them. Through the ongoing and repetitive process, the quality of the system continued to improve.

6.6 Conclusion

As the conclusion, the overall testing process was discussed in this chapter which consists of elements such as the test plan, test strategy, test schedule, test design and other testing. The testing process make a complete software product can be developed and although it is not certain all the bugs and flaws of the system were discovered, the system and application is now more usable and error-free. Good testing also requires a tester's activity, experience and intuition, together with proper techniques. In overall, the test cases are tested and the results are positive. Besides, the real tester for this system are supervisor, Encik Muhammad Suhaizan Bin Sulong and evaluators. During the tester testing the system, there are few bugs and problems occurred.

CHAPTER VII

CONCLUSION

7.1 Observation on Weaknesses and Strengths

This section will explain about the strength and the weakness of this system:

7.1.1 Weakness

- Lack in validation

  This system does not have validation on some form.

- Auto save data

  For auto save feature, it is said that the way of code implementation applying in the E-Book Maker System is not well observed by means of using text editor plugin that are out of control.

- Auto detect spelling error

  This system does not have auto detect spelling error because this system uses bilingual language.

7.1.2 Strengths

- Save Time, Good Interface and Easy to Use.

  E-Book Maker System has a good and user friendly interface. This means that by one glance, user can easily proceed their task using the system. This interface is also designed to make sure that the user understands the system in simpler way. Author can directly send the book to publisher without having to print the books transcript.

- Use less of paper and friendly environment

  The publisher does not need to make a lot of photocopy of books. The readers and publisher only need to read the book online. Also, authors do not need to print and submit the books transcripts on paper to publisher.

- Have a security question

  This system provides two (2) security question for user to answer. The security question will help user to verify their identity if they forgot their password or to recover their account.

7.2 Propositions for Improvement

Improvement that can be made to the E-Book Maker System is the functionality of this system can be improved for future use. This system should be provided with more details or guide so that the users will understand on how to use it. The content of this system must be well-structured in order to facilitate users in any way. Improvement in writing the books to make it easier for author to write the book.

7.3 Project Contribution

The user manual which details how the user, author and publisher can learn how to use the system can be referred in Appendix I.

7.4 Conclusion

In a conclusion, the E-Book Maker System has been completed within 3 months and half although it does not fulfill some of the requirements of this system. There is a lot of knowledge, advantages and disadvantages learned during this final year project.

REFERENCES

Universiti Teknikal Malaysia Melaka (2015). “Panduan Penulisan Projek Sarjana Muda version 3.” Jawatankuasa Projek Sarjana Muda FTMK.

Books Vs E-books: Pros and Cons

David Richardson - https://pickmyreader.com/books-vs-ebooks-pros-cons/

Bootstrap

Mark Otto-Jacob Thornton - https://getbootstrap.com/

Self-publishing in the Cloud

https://www.fastpencil.com/

How E-books Are Used: A Literature Review of the E-book Studies Conducted from 2006 To 2011

Staiger -Jeff - https://www.questia.com/library/journal/1G1-318105160/how-e-books-are-used-a-literature-review-of-the-e-book

APPENDIX

```php
<?php

$dbhost = "localhost";
$username = "root";
$password = "";
$db = "ebook";

$connect = mysqli_connect($dbhost, $username, $password, $db);
?>
```

Figure 1.1: PHP connect to MySQL

```php
    {

        $name = $_POST['author_name'];
        $email = $_POST['author_email'];
        $phoneno = $_POST['author_phoneno'];
        $gender = $_POST['author_gender'];
        $usrname = $_POST['author_username'];
        $pswrd = $_POST['author_password'];
        $conpswrd = $_POST['author_confirmpass'];
        $q1 = $_POST['question1'];
        $ans1 = $_POST['answer1'];
        $q2 = $_POST['question2'];
        $ans2 = $_POST['answer2'];

        $stmt = "INSERT INTO author (author_name, author_email, author_phoneno, author_gender, author_username, author_password,
            author_confirmpass, question1, answer1, question2, answer2)
         VALUES ('$name','$email', '$phoneno', '$gender', '$usrname', '$pswrd', '$conpswrd', '$q1', '$ans1', '$q2', '$ans2')";
        if(mysqli_query($connect, $stmt))
        {
            ?><script type="text/javascript">alert("You have successfully register. Please Log In.");</script>
            <?php
            echo "<meta http-equiv=\"refresh\"content=\"1;URL=login.php\">";
        }
        else
        {
            ?><script type="text/javascript">alert("Register failed.");</script>
            <?php
            echo "<meta http-equiv=\"refresh\"content=\"10;URL=register_author.php\">";
        }
    }
    mysqli_close($connect);
?>
```


Figure 1.2: PHP Insert Data into MySQL

```php
<?php

    session_start();
    include 'connect.php';

    $usrname = $_POST['author_username'];
    $email = $_POST['author_email'];
    $phone = $_POST['author_phoneno'];

    $sql = "UPDATE author SET author_email = '".$email."', author_phoneno = '".$phone."' WHERE author_username = '".$usrname."'";
    $result = mysqli_query($connect,$sql) or die ("Could not insert data into DB: " . mysqli_error($connect));

        echo "<script>alert('The data successfully updated');";
        echo "window.location.href = 'profile_author.php';</script>";
        exit;

    if($result) {
     echo "Record update successfully";
    }else{
        echo "Could not been updated.";
    }

    mysqli_close($connect);
?>

```

Figure 1.3: PHP Update Data in MySQL

```php
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "ebook";

$connect = mysqli_connect($servername, $username, $password, $dbname);

if (!$connect) {
    die("Connection failed:" . mysqli_connect_error());
}
    $id = $_GET["chapterid"];
    $bukuid = $_GET["bukuid"];
    $sql = "DELETE FROM chapter WHERE chapter_id = '$id' ";

    if(mysqli_query($connect, $sql)) {
        echo "Chapter has been deleted successfully";
        header("location:create_newbook.php?bukuid=$bukuid");
    } else {
        echo "Error deleting the chapter: " . mysqli_error($connect);
    }

?>
```


Figure 1.4: PHP Delete Data from MySQL

```php
<?php
    include ('connect.php');
    $bukuid=$_GET['bukuid'];
    $result2 = mysqli_query ($connect,"SELECT * FROM author a JOIN book b JOIN chapter c JOIN genre g WHERE a.author_id=b.
        author_id AND b.book_id=c.book_id AND g.genre_id=b.genre_id AND c.book_id='".$bukuid."'") or die ("Error running MySQL
        query");

    $result22 = mysqli_query ($connect,"SELECT * FROM author a JOIN book b JOIN chapter c JOIN genre g WHERE a.author_id=b.
        author_id AND b.book_id=c.book_id AND g.genre_id=b.genre_id AND c.book_id='".$bukuid."'") or die ("Error running MySQL
        query");
    $row22 = mysqli_fetch_assoc($result22);
?>
<input id="btnCetak" name="btnCetak" value="Print" style="width:60; margin-left: 270px;" type="button" onclick="javascript:this.t
    ype = 'hidden'; window.print();">

<center>
<div class="print" style="width: 800px;">
<h1><?php echo $row22['book_title'] ?></h1>

<h4><?php echo $row22['book_genre'] ?></h4>
<p style="text-align: justify;"><?php echo $row22['book_desc'] ?></p>

<?php
    while($row2 = mysqli_fetch_assoc($result2))
    {
?>

<hr>
<h1 style='page-break-before: always; margin-top:50px;'><?php echo $row2['chapter_name1'] ?> <?php echo $row2['chapter_title'] ?>
</h1>
<h4><?php echo $row2['chapter_header'] ?></h4>
<p style="text-align: justify;"><?php echo $row2['content'] ?></p>
```

Figure 1.5: Generate PDF file

```
            url: "trash/cubby.php", //get response from this file
            data: {
             name: name_val,
             id: id,
             header: chapter_header,
             status: chapter_status
            },
            success: function (response) {
              //$('textarea[name="edit"]').val(response); //send response to textarea
              console.log(response);
              $('#resultajax').html(response);
            }
          });
        });
      });
    </script>
    <script>
        $(function(){
          $('#edit')
            .on('froalaEditor.initialized', function (e, editor) {
              $('#edit').parents('form').on('submit', function () {
                console.log($('#edit').val());
                return false;
              })
            })
            .froalaEditor({enter: $.FroalaEditor.ENTER_P, placeholderText: null})
//            .oninput(alert('event has been triggered'))

        });

    </script>
</main>
</body>
```

Figure 1.6: Text Editor source code