VICTORIA UNIVERSITY
MELBOURNE AUSTRALIA

ENGINEERING AND SCIENCE

NIT6150

Advanced Project

Project Proposal

Design and Implementation of a Library Management System

Team Leader: Choongyeol Kim (4519560)
Team Members: Choongyeol Kim (4519560)
                    Brijender Partap Rana (4511875)
                    Madhaban Bhowmik (4340084)
                    Priyanka Sharma (4488341)
                    Manoj Adhikari (4511811)

Client: VU Sydney

Supervisor: Dr. Rachid Hamadi
                     College of Engineering and Science

Coordinator: Dr. Quan Vu
                     College of Engineering and Science

vu.edu.au
CRICOS Provider No. 00124K

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

We have been building our reputation from the years for producing, creating and implementing a variety of software solutions for different business sectors of the market. We have extensive experience in many diverse areas of both software and hardware development. Our experienced staff of professionals has worked with a wide array of platforms, languages, and tools in the course of our various projects. We provide the customized solutions for Websites, CMS and database application.

II.   CLIENT PROFILE

Victoria University is one of the leading universities not only for domestic university but also for international students from last 100 years with delving high quality of education. The Sydney Campus is really famous for offering international students different courses of under graduation, post-graduation and diplomas in diverse streams along with vocational training. The infrastructure of the university is really familiar to the students and fulfils all aspect of their needs.

III.   PURPOSE

Victoria University has been maintaining a library for students and its professors and teaching staff, which is now growing more as the number of students is increasing in Sydney Campus. As per now they use Microsoft Excel for the library management such as lending books to students and teaching staff. This includes how many books and how many copies are of a book in the library, and which are issued to student.

Now the number of students and quantity of books considerably increased, so it became really hard to manage the library with Excel effectively. They need an automotive system which is easy to manage the library and cost effective to fulfils all their requirements.

Moreover, to reduce the workload from the front desk, the future library system shall be accessed by the student in the campus area online, so student can check which books are available online and come with books to get issued, which will help students and staff to manage efficiently.

IV. OBJECTIVE

We will give an intranet based complete solution for the library management, which will help library staff to reduce the paper work and it will be easy to use in term of finding and tracking books, as well as accessing student records. Moreover, it will be helpful to calculate fines and send alert to the borrowers on due date.

This customized solution will work on intranet so students and teaching staff also can access the system in the campus.

V. TECHNOLOGY STACK

As we are using the latest technologies for customized solutions, while selecting the technology, we also think about the cost factor with the advancement.

   I.   PHP
  II.   MySQL
 III.   JQuery / AJAX

IV.   Apache web server

V.   HTML / WEB 2.0

VI. SCOPE

Software Reverse Engineering (SRE) is the practice of analysing a software system, either in whole or in part, to extract design and implementation information.

I.   Prepare Software Requirement Specification (SRS)

II.   Analyse the application

III.   Module based development approach

We will follow the agile methodology to develop the application, which will help us to analyse the project requirement and involve with Victoria University in the development process. This will also help us to make sure the development closely reviews the client's needs and the deliverables deliver on time. To initiate this complex process we define the following scope of the project.

I.   We will provide a single screen for the login and on the bases of roles a user will be identified and show a screen which he/she has permission, like if the user has permission for super user it will be redirected to Super User dashboard.

II.   Three types of user are

1.   Super Admin/Manager

2.   Front Desk Officer

3.   Student/Teaching Staff

III.   Super user will have access to dashboard where admin will be able to manage different things on one screen such as creating a user account for the system according to specified roles. Furthermore, the super user will be able

to generate different reports for books, enter books into the system and remove books from system.

IV. Super user will be able to add different categories, assign different book assign to category, so all CRUD operations to all entities.

V. The second users will be Front Desk Officers, who have to view books with customized search functionalities by author's name or publisher name etc., and issue books, list students who have specific books with their due dates so can hold a book, renew books and automatic fine calculator for student and also can send reminder to the students.

VI. The third users will be students or teaching staff, who can access their accounts. They can check how many books they have, when the due dates for the books are and what is their fine due to pay to library.

VII. Super user/front desk user will be able to generate different reports like books due with time period, so it would be like due more than 5 days or 30 days etc. and how many books are available in the library according to subjects/categories.

NOT IN SCOPE

I. Payment method integration is not a part of this project, as if fine is due student will use ECA website to pay fine with their student ID.

II. We will give a book id to every book, so search can be done with book id and student id but right now we are not integrated the barcode into the system.

VII. DELIVERABLES

This section states deliverables of this project, which are as follows:

1. Project Allocation Form

2. Project Proposal

3. System Analysis and Design Documentation

4. Project Evaluation Report including Software Testing

5. User Manual

6. E-Poster

7. Software Package

VIII. MILESTONES

Below table shows the milestones modules for the prototype to be build.

Table 1 Brief Milestones

	Description of Work
Milestone Deliverables	- Make Database design. - Creating HTML for once new design has been approved. - Implementation of Data layer and business logic layer. - Create interface for admin to set up library. - Make User interface for Front Desk/Desk manager. - Make Student Interface.

IX. RESOURCES

Followings are the details of human resources that will take care of this prestigious project.

Table 2 Human Resources

Resource/Developers	Skill Set
Choongyeol Kim	Strong hand on PHP, MySQL
Brijender Parta Rana	Strong hand on PHP, MySQL
Manoj Adhikari	Strong hand in Photoshop, HTML, HTML 5, Illustrator, JavaScript and JQuery
Priyanka Sharma	Strong hand in Photoshop, HTML, HTML 5, Illustrator, JavaScript and JQuery
Madhaban Bhowmik	System Analyst and QA team member

X. FUNCTIONAL REQUIREMENTS

We have three different type of users one is Admin, Librarian and Student and Staff. Admin is the super user which is not able to insert into the system. We will insert manually in the database.

Once a user login, we will check which role he/she belongs to first, and then check the different permission of menu and page according to the role.

Admin Scope Functionalities are:

1. Admin will able to manage different user, like student /staff and Librarian, More over he will delete edit and assign.
2. Admin will able to insert different item like Publisher, Author with their list and delete functionality.
3. Admin will fix the fine for the delay of books.
4. Fine will calculate automatically on date return and date due.
5. Admin will fix the duration for how much long a person can borrow a book.
6. Add different book add their different copies of the books.
7. Admin user can book/reserve a book for student.
8. Admin user can generate fine for different students.
9. Admin user can generate different report with different period. The reports will generated on term get selected by the admin user.

Librarian Scope Functionalities are:

1. Librarian user will able to insert different item like Publisher, Author with their list and delete functionality.
2. Librarian user will able to insert categories and delete and edit.
3. Librarian user will able to insert Section and delete and edit.
4. Fine will calculate automatically on date return and date due.
5. Add different book add their different copies of the books.
6. Librarian user can book/reserve a book for student.

7. Librarian user can generate fine for different students.

8. Librarian user can generate different report with different period. The reports will generated on term get selected by the admin user.

Student/Staff Scope Functionalities are:

1. Student /staff will able to access his/her account detail, change password

2. Student/Staff will able access how many and which books are along with him/her.

3. Student/Staff will able to reserve a book online for him, but only if book is not with someone who is review it or which will comes in first. Add different book add their different copies of the books.

4. Student/Staff will able access their fine details.

ENTITY RELATIONSHIP (ER) DIAGRAM

Figure 1 shows Entity Relationship Diagram of the library management system.
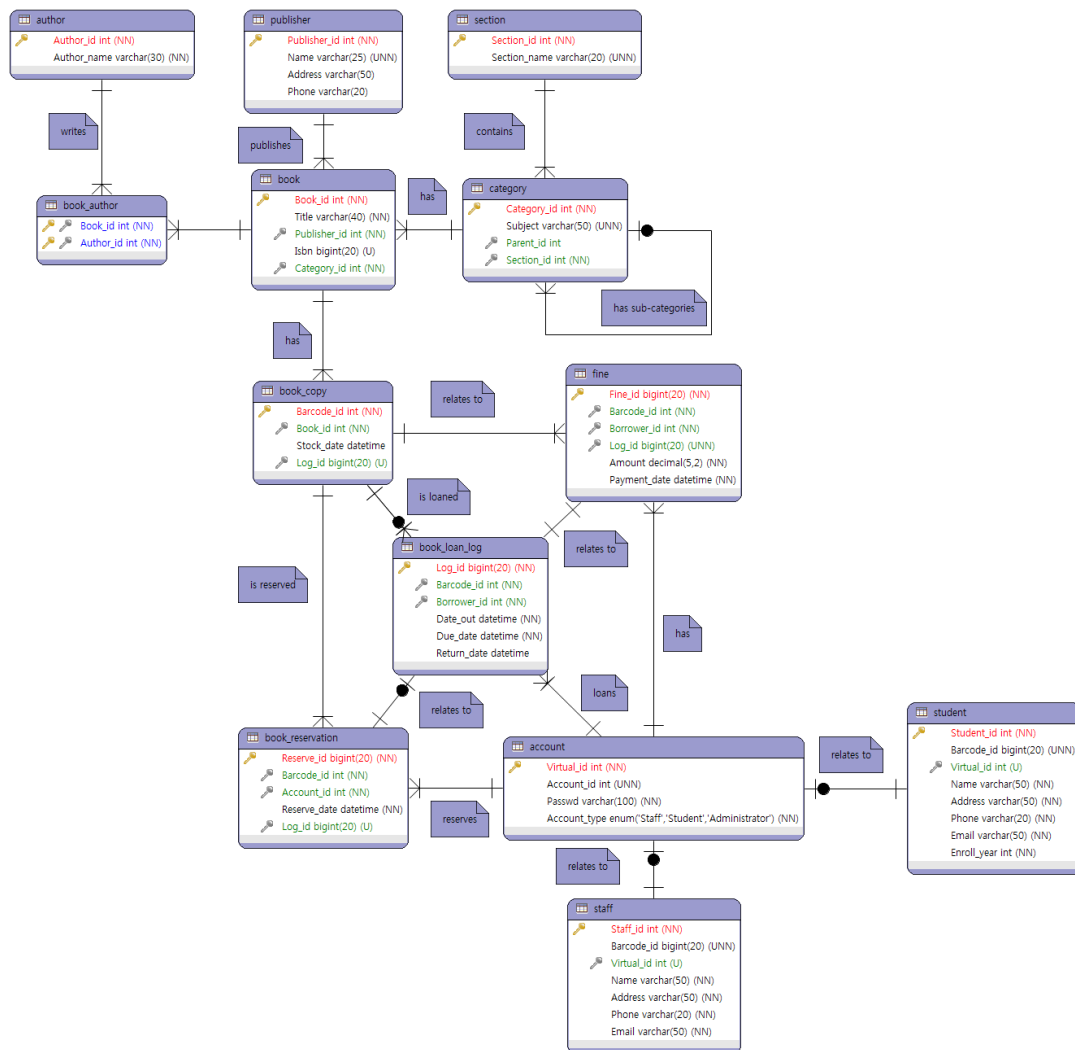
Figure 1 Entity Relationship Diagram

USE CASE DIAGRAM

This section presents Use Case Diagrams in this project. The Use Case Diagram for Admin is shown in Figure 2.
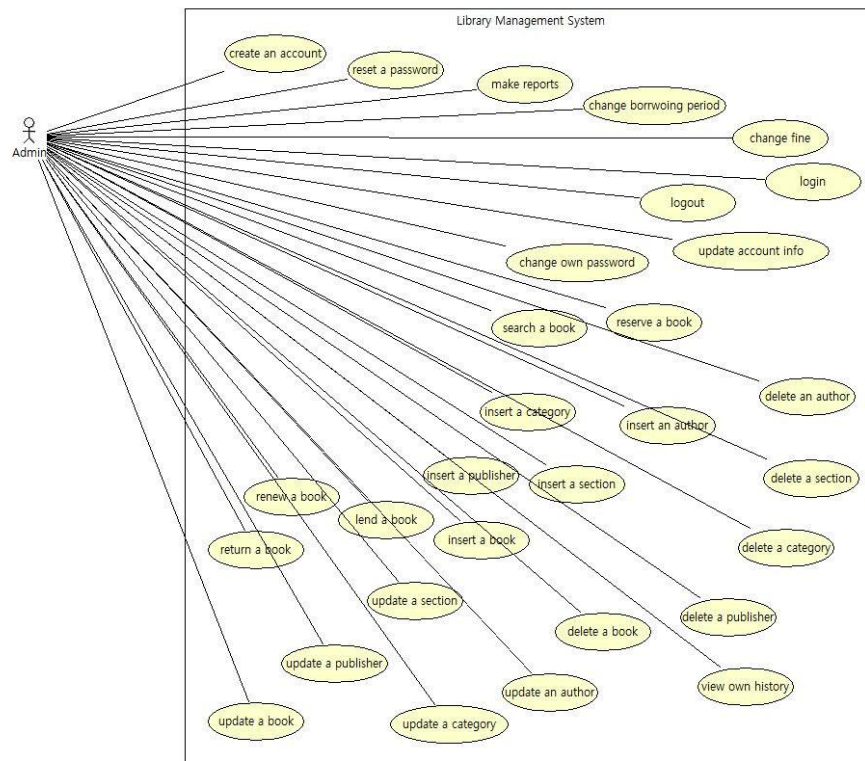
Figure 2 Use Case Diagram for Admin

Figure 3 depicts the Use Case Diagram for Staff.
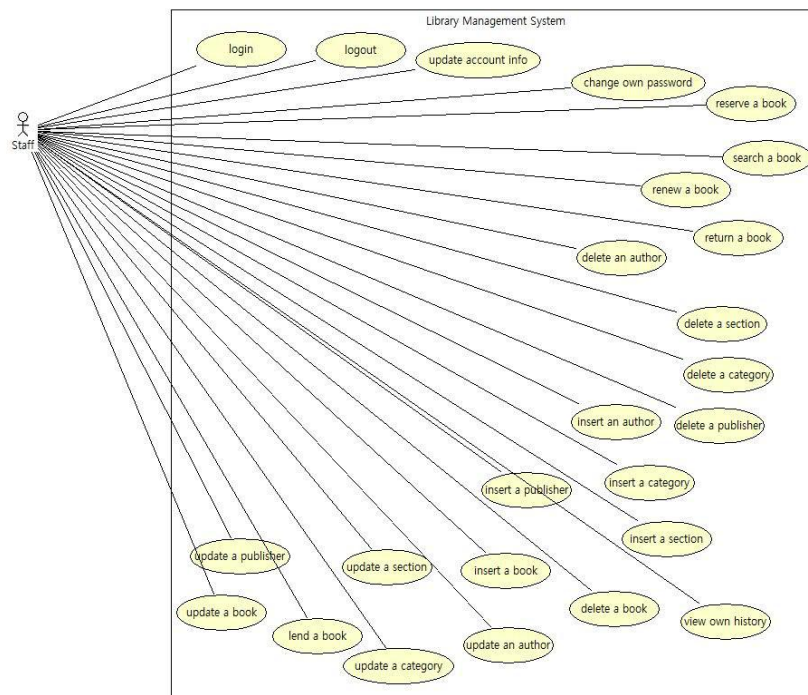
Figure 3 Use Case Diagram for Staff

Figure 4 illustrates the Use Case Diagram for Student.
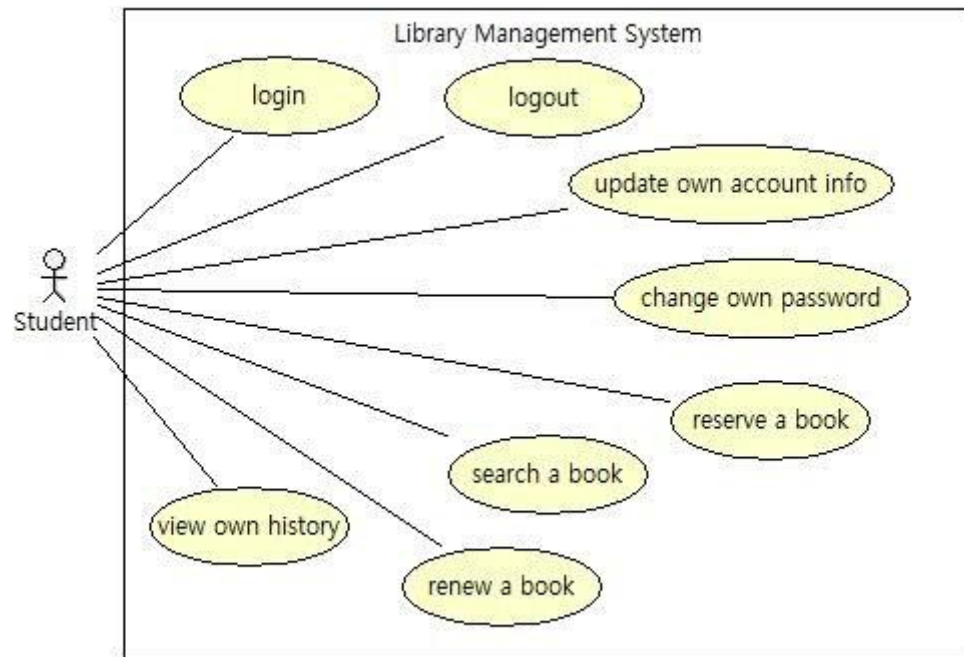
Figure 4 Use Case Diagram for Student

Table 3 enumerates the aforementioned Use Cases and their corresponding Actors.

Table 3 Use Case List and Corresponding Actors

Use Case	Actor		
	Admin	Staff	Student
Create an account	O	X	X
Reset a password	O	X	X
Make reports	O	X	X
Change borrowing period	O	X	X
Change fine	O	X	X
Update account info	O	O	X

Insert/Update/Delete a section	O	O	X
Insert/Update/Delete a category	O	O	X
Insert/Update/Delete a publisher	O	O	X
Insert/Update/Delete an author	O	O	X
Insert/Update/Delete a book	O	O	X
Lend/Return a book	O	O	X
Update own account info	O	O	O
Change own password	O	O	O
Login/Logout	O	O	O
Renew a book	O	O	O
Search a book	O	O	O
Reserve a book	O	O	O
View own history	O	O	O

Note that the Use Case of "Lend/Return a book" is triggered by borrowers, which means that Student cannot be an Actor, but an Initiator in this use case.

Table 4 describes each use case.

Table 4 Description of Use Cases

Use Case	Description

Create an account	This use case begins when the administrator enters an account name, a type, a residence address, a phone number and an email address into the Library Management System.
	Pre-condition: The administrator has completed login process.
Reset a password	This use case begins when the administrator enters an account id and new password into the Library Management System.
	Pre-condition: The administrator has completed login process.
Make reports	This use case begins when the administrator selects a report type, searching period and other required data into the Library Management System.
	Pre-condition: The administrator has completed login process.
	Post-condition: The report is showed up in web pages.
Change borrowing period	This use case begins when the administrator enters a new borrowing period according to the account type into the Library Management System.
	Pre-condition: The administrator has completed login process.
Change fine	This use case begins when the administrator enters a new amount of fine according to the account type into the Library Management System.
	Pre-condition: The administrator has completed login process.
Update account info	This use case begins when the administrator or a staff enter new information regarding an account into the Library Management System.
	Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the account.
Insert a section	This use case begins when the administrator or a staff enter new section name into the Library Management System.
	Pre-condition: The administrator or the staff have completed login process.

Update a section	This use case begins when the administrator or a staff enter new section name to replace the old one and its id into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the section.
Delete a section	This use case begins when the administrator or a staff enter a section id to delete into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the section.
Insert a category	This use case begins when the administrator or a staff enter new category name, corresponding section id and parent category id into the Library Management System. Pre-condition: The administrator or the staff have completed login process.
Update a category	This use case begins when the administrator or a staff enter new category information to replace the old one and its id into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the category.
Delete a category	This use case begins when the administrator or a staff enter a category id to delete into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the category.
Insert a publisher	This use case begins when the administrator or a staff enter new publisher name, an address and phone number into the Library Management System. Pre-condition: The administrator or the staff have completed login process.

Update a publisher	This use case begins when the administrator or a staff enter new publisher information to replace the old one and its id into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the publisher.
Delete a publisher	This use case begins when the administrator or a staff enter a publisher id to delete into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the publisher.
Insert an author	This use case begins when the administrator or a staff enter new author name into the Library Management System. Pre-condition: The administrator or the staff have completed login process.
Update an author	This use case begins when the administrator or a staff enter new author name to replace the old one and its id into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the author.
Delete an author	This use case begins when the administrator or a staff enter an author id to delete into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the author.
Insert a book	This use case begins when the administrator or a staff enter new book title, a publisher id, ISBN and its category id into the Library Management System. Pre-condition: The administrator or the staff have completed login process.

Update a book	This use case begins when the administrator or a staff enter new book information to replace the old one and its id into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the book.
Delete a book	This use case begins when the administrator or a staff enter a book id to delete into the Library Management System. Pre-condition: The administrator or the staff have completed login process and already loaded the old information of the book.
Lend a book	This use case is triggered by borrowers, and begins by the administrator or a staff. When starting, the borrower's id and the book's barcode id are entered into the Library Management System. Pre-condition: The administrator or the staff have completed login process, and the borrower do not have unpaid fine.
Return a book	This use case is triggered by borrowers, and begins by the administrator or a staff. When starting, the borrower's id and the book's barcode id are entered into the Library Management System. Pre-condition: The administrator or the staff have completed login process. Post-condition: When the return is over due date, fine is imposed to the borrower.
Update own account info	This use case begins when users want to change their account information such as address, phone number and email address. Pre-condition: The users have completed login process.
Change own password	This use case begins when users want to change their current password. To do this, they should enter old password for verification as well. Pre-condition: The users have completed login process.

Login	This use case begins when a user wants to use the Library Management System. They are required to enter their account id and password. If wrong password is tried several times, the account is locked, and the user should inquire to unlock the account to the administrator. Pre-condition: The users have not yet logged in the Library Management System. Post-condition: The session has the expiration time.
Logout	This use case begins when a user wants to logout from the Library Management System. Pre-condition: The user has completed login process.
Renew a book	This use case begins when a borrower wants to renew his/her already borrowing books. Pre-condition: The user has completed login process.
Search a book	This use case begins when a borrower wants to search books with author name, subject and title. Pre-condition: The user has completed login process.
Reserve a book	This use case begins when a borrower wants to reserve books. Pre-condition: The user have completed login process, and searched books. Post-condition: The reserved books are not renewed, nor borrowed by others.
View own history	This use case begins when a user wants to see his/her history such as his/her reserved books or borrowing books. Pre-condition: The users have completed login process.

UML CLASS DIAGRAM
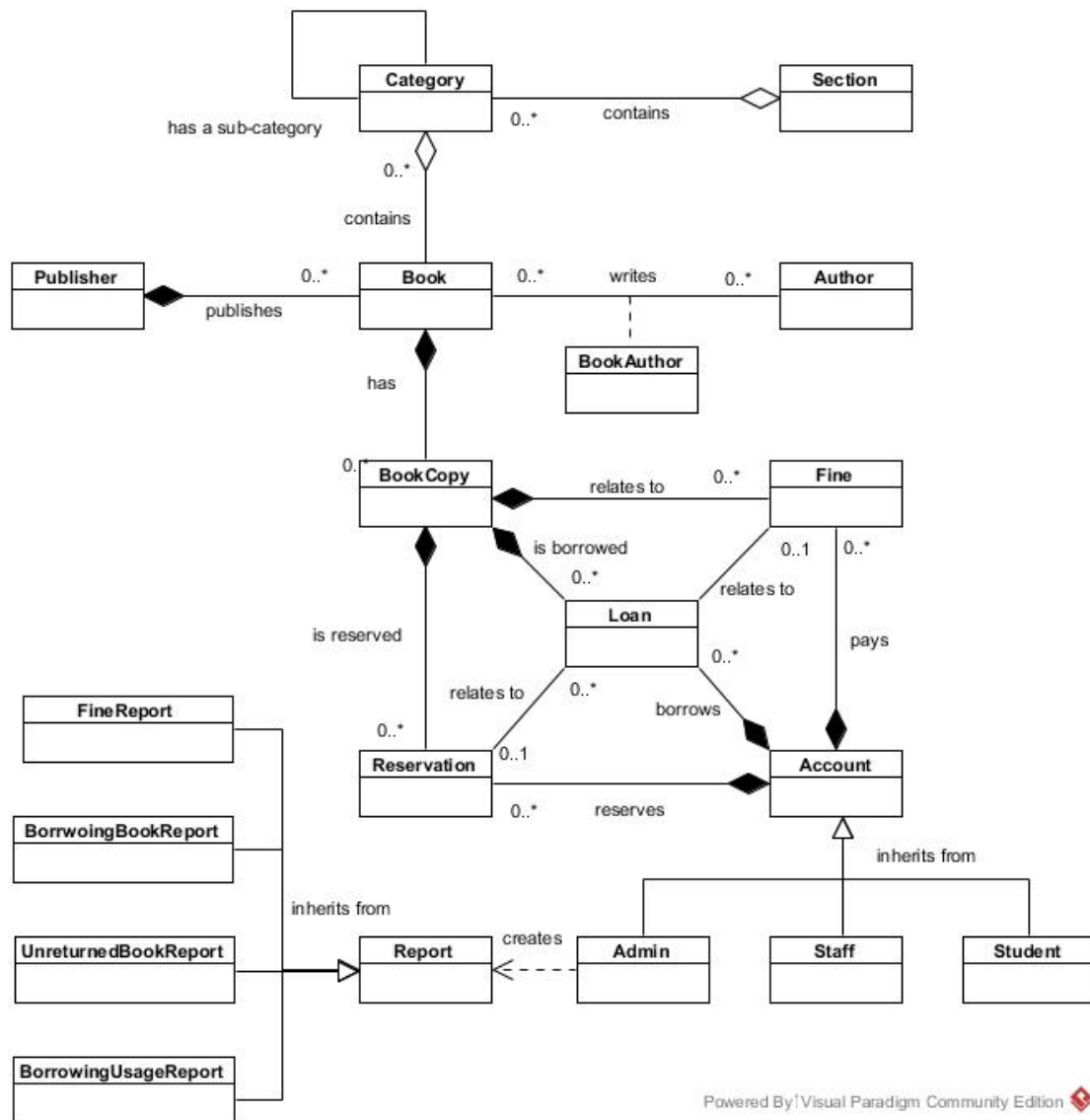
Figure 5 delivers the UML Class Diagram.

Figure 5 Class Diagram

XI. NON-FUNCTIONAL REQUIREMENTS

This section explains non-functional requirements as the following sequence.

- Cultural and political requirements

- Legal and licencing requirements

- Look and feel and portability

- Maintainability and support requirements

- Operational requirements

- Performance requirements

- Security issues

- Usability and humanity requirements

CULTURAL AND POLITICAL REQUIREMENTS

This section describes cultural and political requirements, which are based on criteria written in the materials by Chung (2015) and Robertson & Robertson (2015).

As cultural and political requirements, the followings are assumed previously.

- This project will support only English.

- This library management system will not store any information regarding users' gender and religion, and this policy will applied to the user login web page as well.

- Throughout web pages, the system will not show any expressions and symbols regarding religious perspectives.

Political requirements are assumed as follows.

- The system will not display any expressions and symbols regarding gender and political perspectives.

- The web site administrator can execute all functionalities which staff can do in the library management system.

- The system does not distinguish staff and librarians functionally.

LEGAL AND LICENSING REQUIREMENTS

To comply with regulations about data security, the system will maintain only essential personal information such as name, residence address, phone number, email address and optionally enrol year if the user is a student.

As mentioned above, this project was decided to use open source products, there is not much licencing issues including patents. However, as we provisionally have decided to use Microsoft Windows as an Operating System (OS) for the library management system, Victoria University should obtain a licence from Microsoft Corporation once the university approves to use the final product of this project. Alternatively, Linux rather than Microsoft Windows can replace it for reducing the Total Cost of Ownership (TCO) because APM (Apache/PHP/MySQL) stack is one open source stack so it is compatible in both Microsoft Windows and Linux.

LOOK AND FEEL AND PORTABILITY

To give uniformity, all web pages shall deploy CSS (Cascading Style Sheets) by which same fonts and colour will be used. Particularly, the main colour is blue because the Website of Victoria University primarily uses the colour for its identity.

Regarding to page construction, all pages should include both header and footer PHP files which easily give coherence throughout the website as well as make the development of the site quicker.

Along with the usage of CSS and HTML5, the web site can easily be displayed by any web browsers equipped in mobile devices. The CSS will convert the layout of normal web pages to pages which are best fit to mobile display.

MAINTAINABILITY AND SUPPORT REQUIREMENTS

For simplifying this project, we will implement it on a single server machine, which is generally well-known as having issues as follows (Anicas, 2014).

- When the workload of either web server or database server increases, the system will not be able to satisfy the clients' performance requirements.
- The system will not be scalable.
- In terms of security, the single server is less secure than the constitution of standalone web servers and database servers.

However, because the project is assumed to manage to deal with the daily usage of data traffic throughout the system, we only discuss maintainability issues which can occur under the single server implementation in this section. The system requires at least one maintenance staff for general server management and database per each.

- The database maintenance staff will handle troubles occurring in the database as well as monitor workload and disk usage for satisfying the service requirements. In addition, although the database backup will be performed automatically by the system, the maintenance staff will observe the process. For the backup period, once a week is recommended for full backup, and a daily incremental backup is also recommended.
- Using network monitoring software such as Multi Router Traffic Grapher (MRTG), the general maintenance staff will monitor the network traffic and the workload of the web server so as to resolve any outage affecting services.
- If both maintenance staff agree to separate the database and the web server due to increasing network traffic, the system shall be re-organised.

OPERATIONAL REQUIREMENTS

Software constraints are summarised below. We use 64 bit WampServer version 5.5 as APM stack. For more information, refer to the web site of WampServer (http://www.wampserver.com/).

- Microsoft Windows (>= Windows 7 for x64)

- Apache webserver (>= 2.4.9 for x64)

- PHP (>= 5.5.12 for x64)

- MySQL (>= 5.6.17)

Hardware constraints are also another important factor for the operation of any software system. However, before confirming the specification, monitoring the current network traffic and estimating the future needs should precisely be investigated. This work should be done by experts. With this reason, we do not mention it in this document except the capacity of hard disk for the database server, which should have enough large capacity of hard disk drive(s) for logging and backup.

PERFORMANCE REQUIREMENTS

According to NielsenJakob (1993), he advices that 1 second of response time is the threshold that users do not feel that they are interruped. Thus, we use this guideline for the performance requirement of the library management system. This time includes network propagation time, so our system should complete any tasks requested by users and show the result within 1 second except the case that the result contains tremendous data, for example, when searching books.

SECURITY ISSUES

Security issues are the most sensitive and delicate issues in software systems. As we use APM stack for implementation of this project, we consider three aspects of security issues.

Apache web server –

- We will use https rather than http because https use Secure Socket Layer (SSL) which provides encryption method during data transmission. Consequently, all inbound web page request to http (port 80) should be redirected to https (port 443). Also, any information to be submitted to the web server and to be protected against the packet sniffing attack should be done by POST method rather than GET method.

- The directory search function of Apache web server should be disabled not to give directory information to users.

- Users' password is recommended to change every three months.

PHP –

- To protect SQL injection, we will apply mysql_real_escape_string() function to all input data to be entered to SQL statements.

- PHP interpreter engine should not display any errors occurred from the database. The errors are only tracked in the database error logs. Especially, any PHP web pages should not display any information regarding database schema.

- For users' password, currently, the most favoured encrypted hashing technique in PHP is "bcrypt", so we will use this technique which is applied in both PHP functions, password_hash() and password_verify(). The "bcrypt" can be used only PHP version 5.5.0 and over.

- The process for verifying users' password only check true or false through SQL query, namely, it cannot get password strings from the database.

- Once a user successfully logged in the system, the session will last for 3 hours for students and for 8 hours for staff.

- Once a user fails to logged in the system with an incorrect password consecutively three times, the account will be locked, so he/she should inquire the unlock of the account.

MySQL –

- The database access should be limited only to legitimate database administrators through firewall based on IP filtering.
- The database SQL queries are only accepted by either stored procedures or stored functions. Plain SQL queries should not allowed.
- The users' password will be stored in encrypt hashed strings, so the column for password should be case-sensitive.
- The privileges for accessing the database depends on the type of user.

USABILITY AND HUMANITY REQUIREMENT

Ease of use is the most important factor for determining to apply any software system. Thus, leaning time for understanding the functionalities of the system and adapting to the system should be reasonably short, and the functionalities should be intuitively clear in any web pages. Besides, from the perspectives of web site engineering, if the navigation depth of any web pages are long, the system should show full path of the web page to the user.

XII. DATA TYPES

This section describes data types in the database. Note that student id starts from 3000000, and staff id starts from 5000000.

Data	Data Type
Account ID (Student ID/Staff ID)	Integer
Barcode ID for Account ID	Bigint
Account Type	Enum ('Staff', 'Student', 'Administrator')
Account Name	Varchar (50)
Account Address	Varchar (50)
Account Phone Number	Varchar (20)
Account Email Address	Varchar (50)
Account Password	Varchar (100)
Enrol Year for Student	Integer
Book ID	Integer
Book Title	Varchar (40)
Book ISBN	Bigint
Publisher ID	Integer
Publisher Name	Varchar (25)
Publisher Address	Varchar (50)
Publisher Phone	Varchar (20)
Book Author ID	Integer
Book Author Name	Varchar (50)
Book Category ID	Integer
Book Subject Name	Varchar (50)
Subject ID for Category	Integer

Section Name	Varchar (20)
Barcode ID for Books	Bigint
Log ID for Borrowing	Bigint
Issuing, Due, Return and Reserve Date	Datetime
Fine	Decimal (5, 2)

XIII. SYSTEM NAVIGATION AND USER INTERFACE DESIGN

SYSTEM NAVIGATION

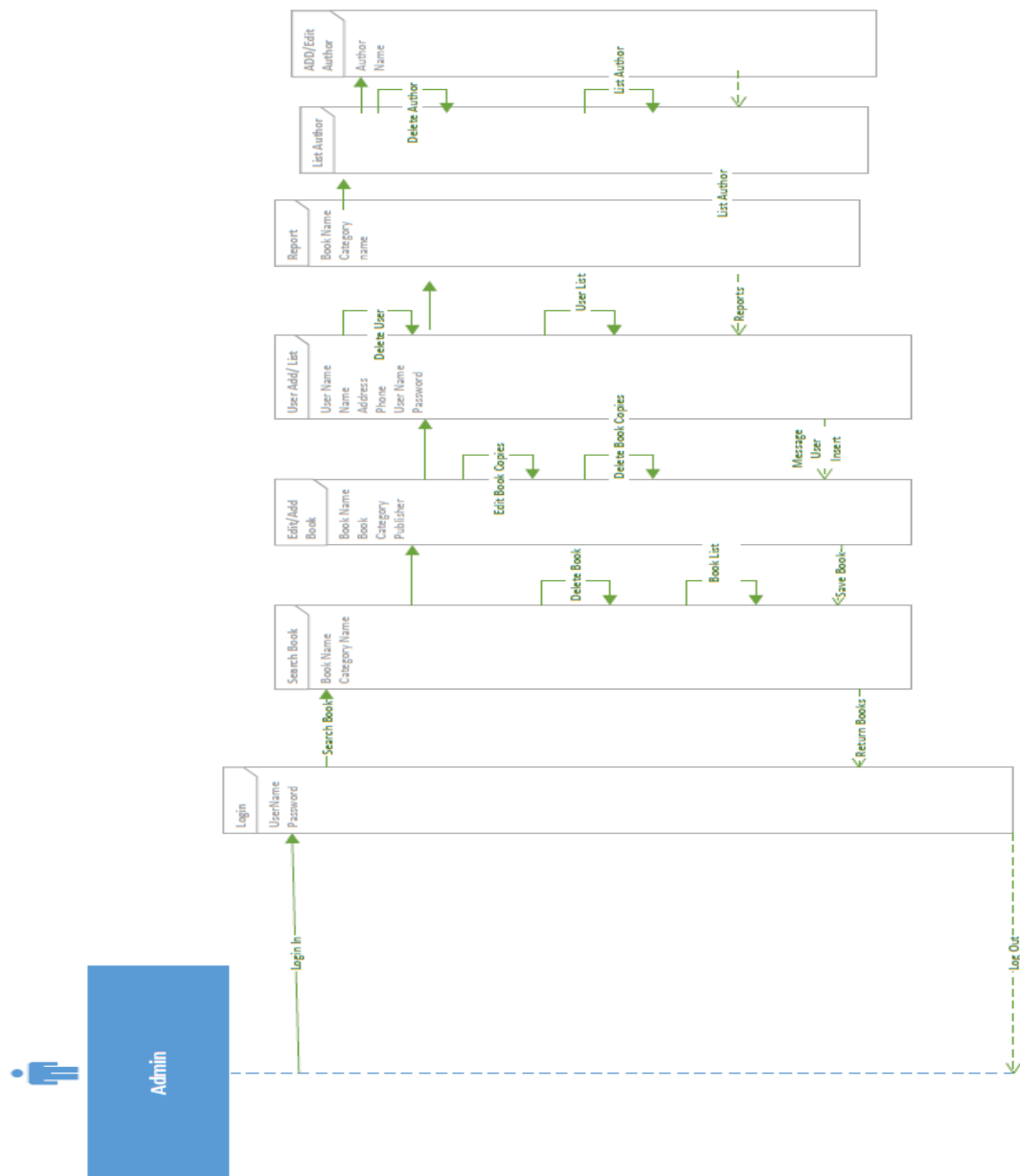Figure 6 shows the site navigation for Admin.

Figure 6 Site Navigation for Admin User

Figure 7 displays the site navigation for Librarian User.
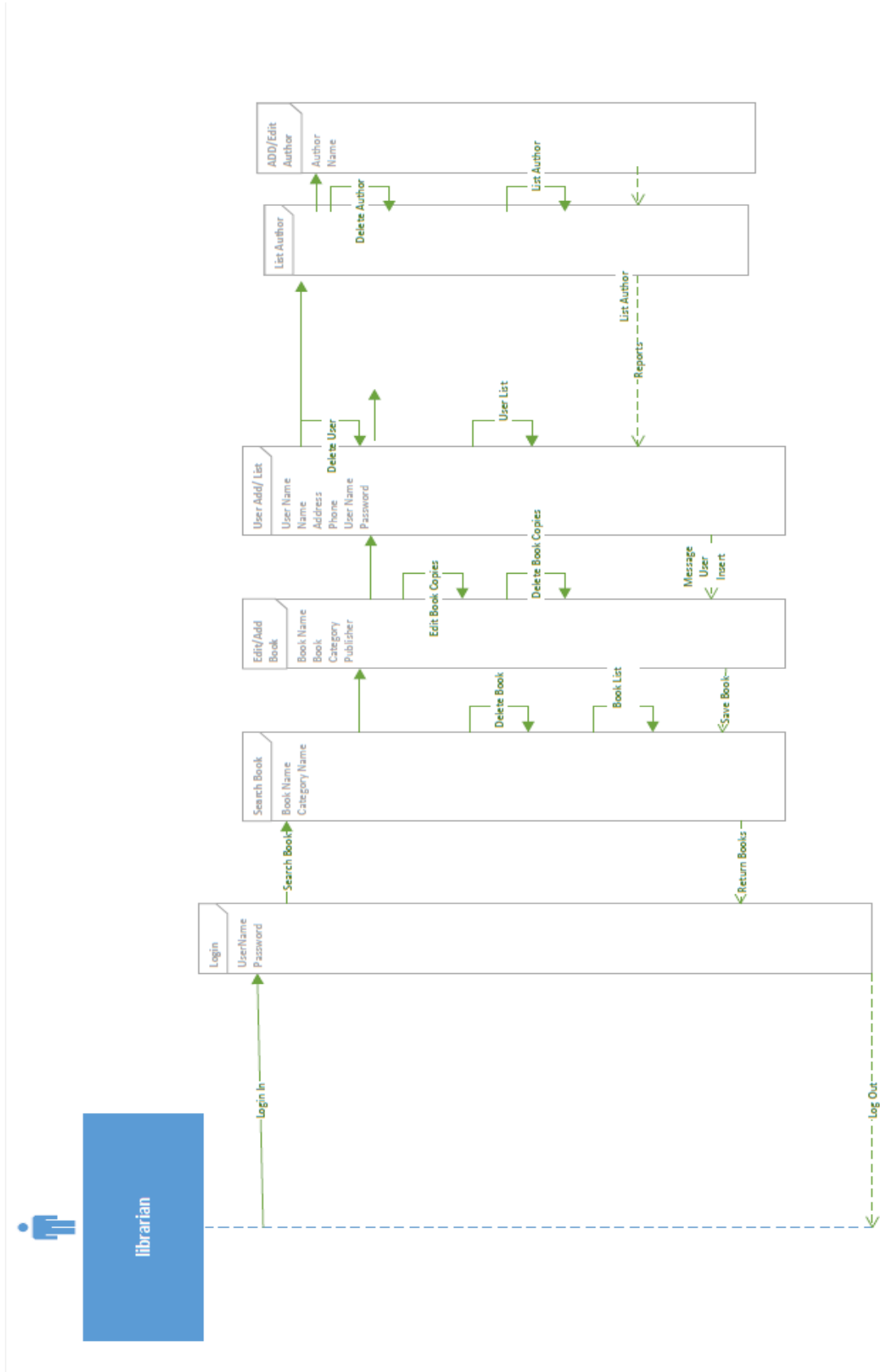
Figure 7 Site Navigation for the Librarian User

Figure 8 shows the site navigation for both Student and Staff.
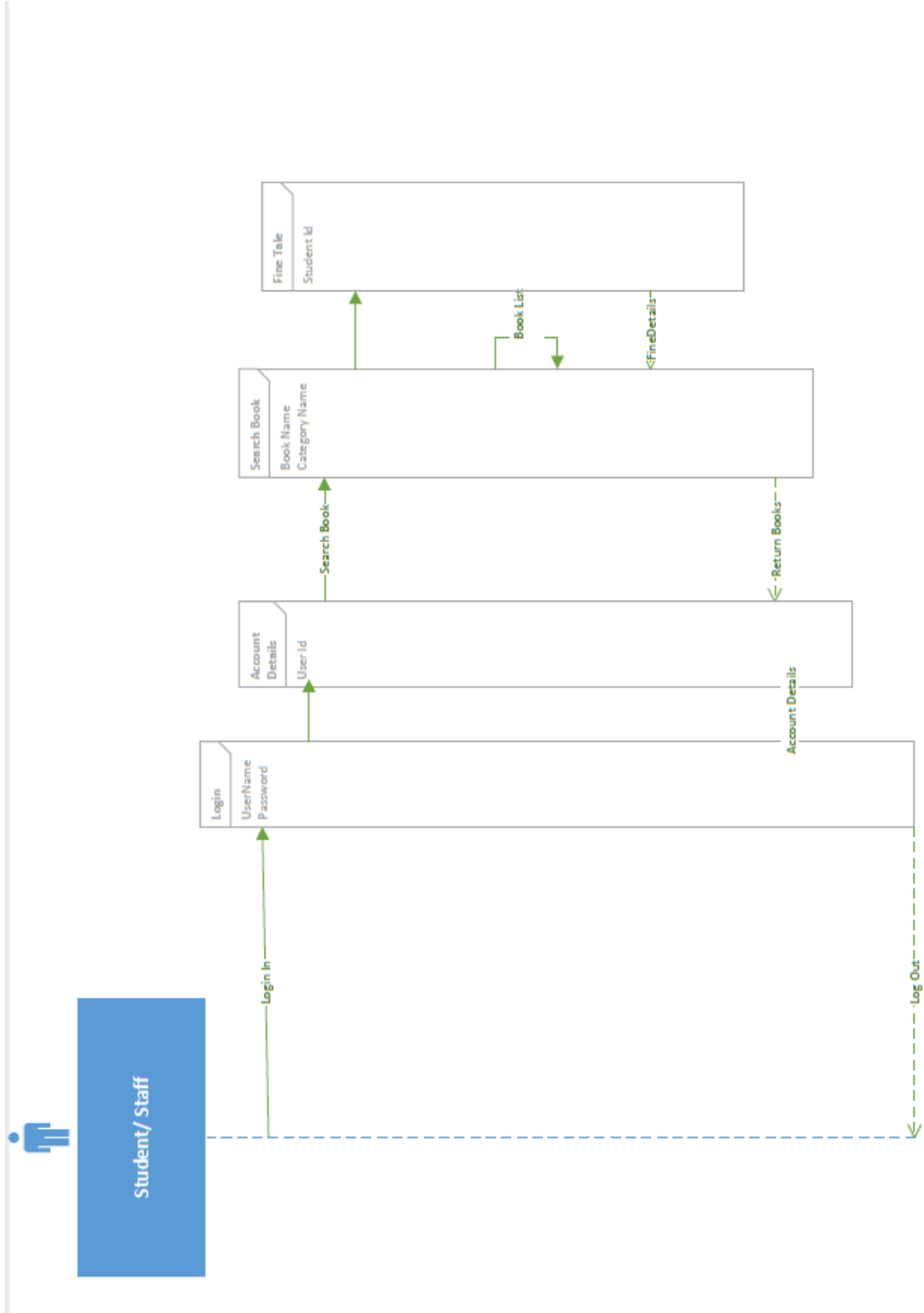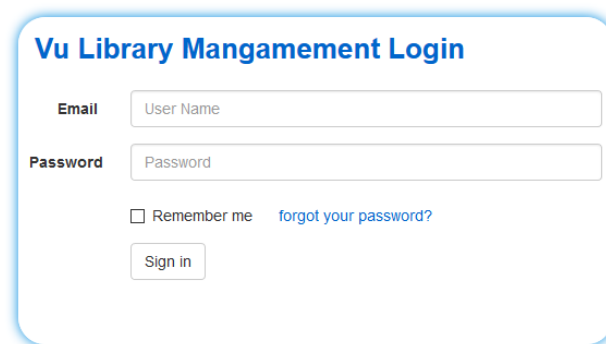
Figure 8 Site Navigation for Staff and Student

USER INTERFACE DESIGN

1. Login Page

   We are create single login page for all different type of users like admin, librarian and student/staff, The user will get redirected on the bases of role he log-in.

Figure 9 Login Page

2. Dashboard

   The dashboard is a page where user will get land after login. There are three different dashboard.

2.1 Add Admin Dashboard
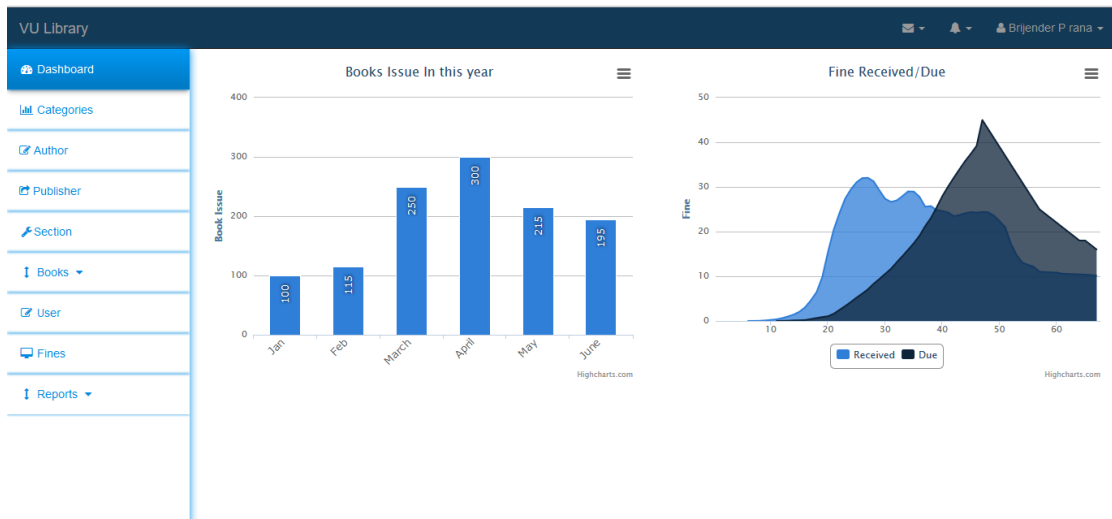
Figure 10 Admin Dashboard

2.2 Librarian Dashboard

Figure 11 Librarian Dashboard

2.3 Student/Staff Dashboard
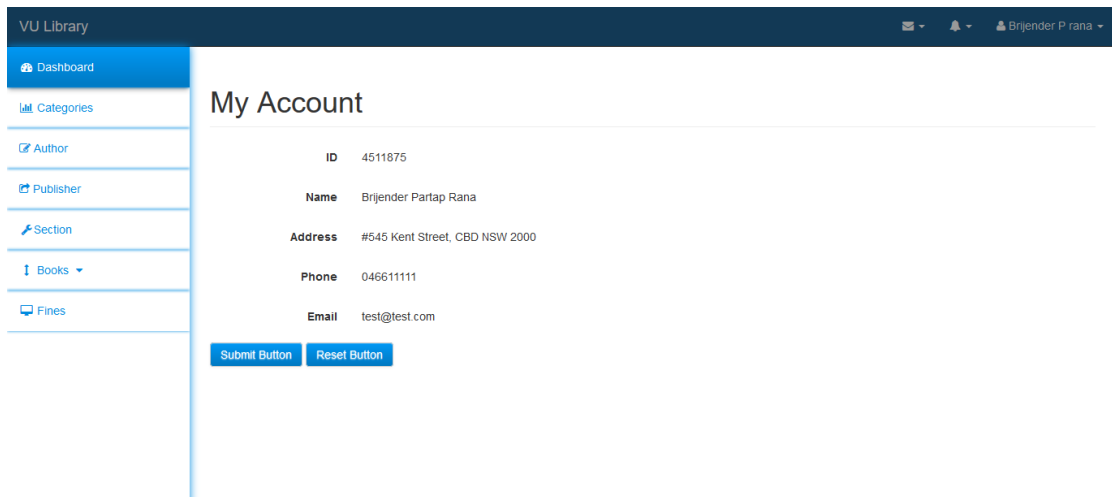
Figure 12 Student/Staff Dashboard

3. Section

Admin can add different section in the library to trace out the book physically in library.

3.1 Add Section
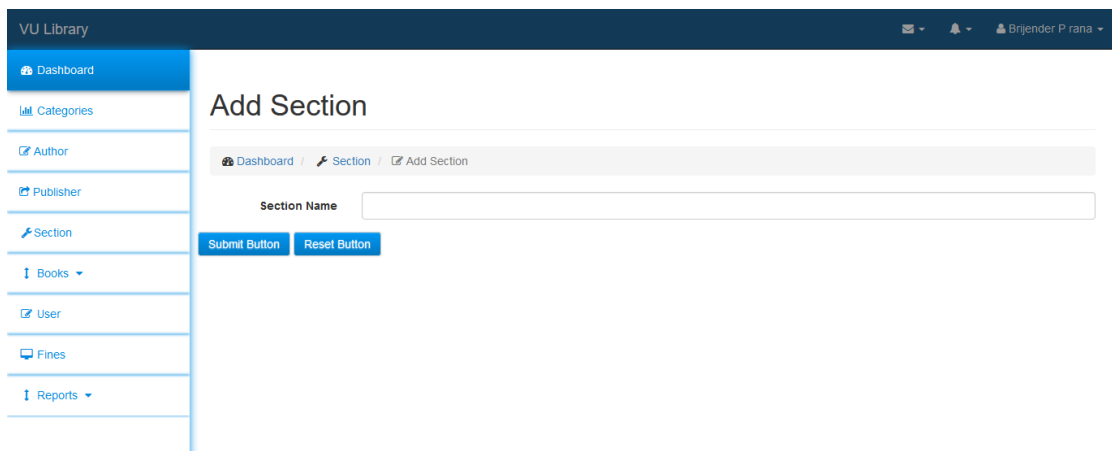
Figure 13 Add Section

3.2 Section List
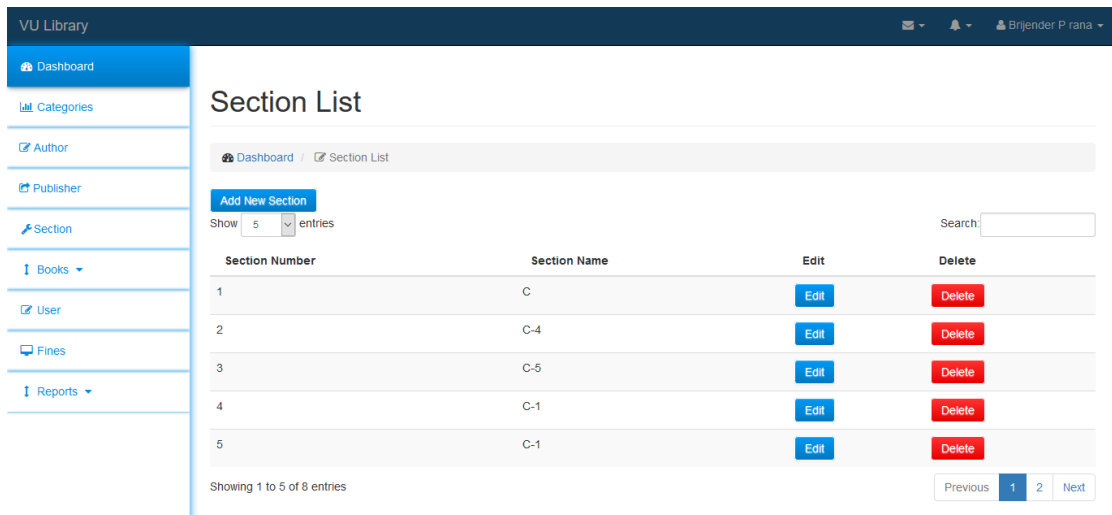
Figure 14 Section List

4. Author Section

Admin and staff can add different author and view admit and delete author from following screen.

4.1   Add Author
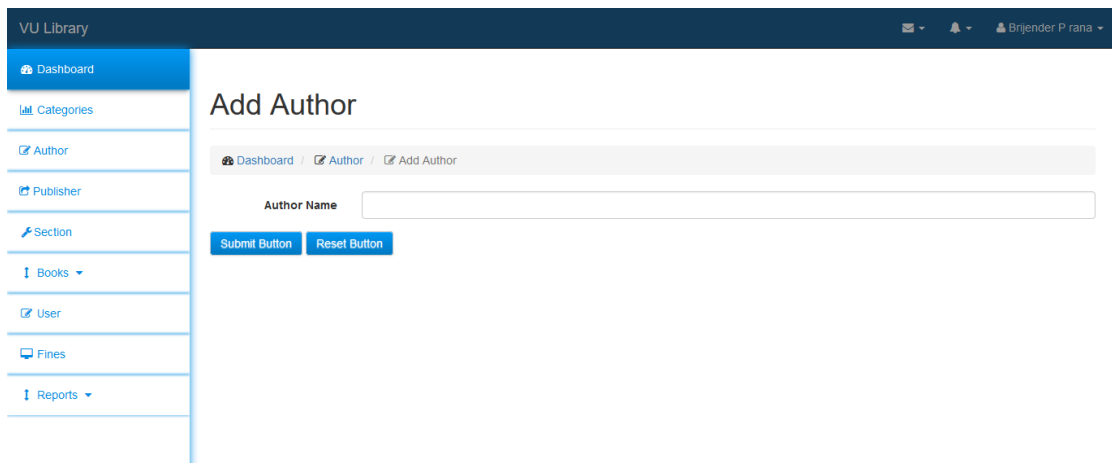
Figure 15 Author Section Add Author

4.2   List Author
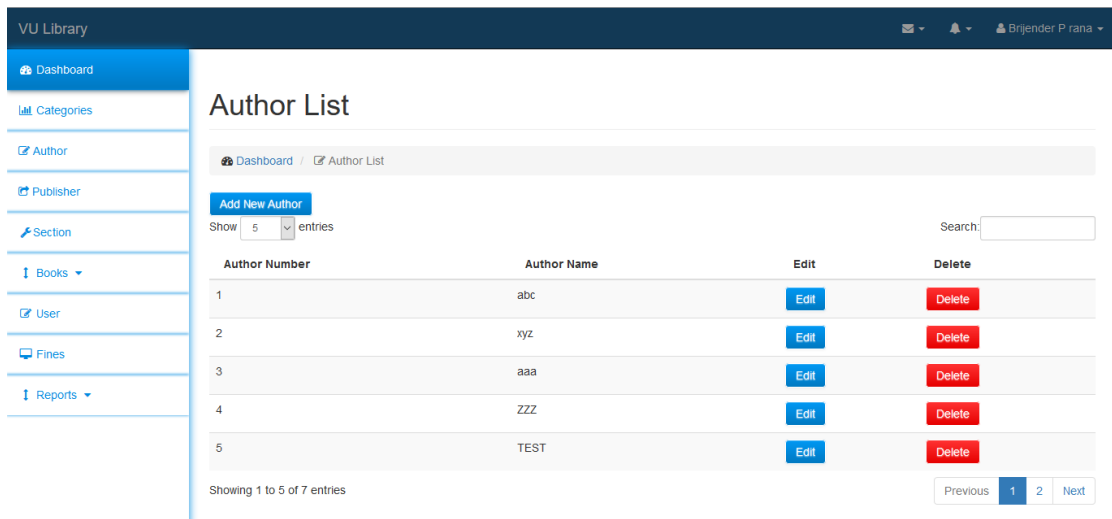
Figure 16 Author Section List Author

5. Category Section

Admin/Staff will able to different category so it will be easy to manage book under different category.

5.1 Add Category
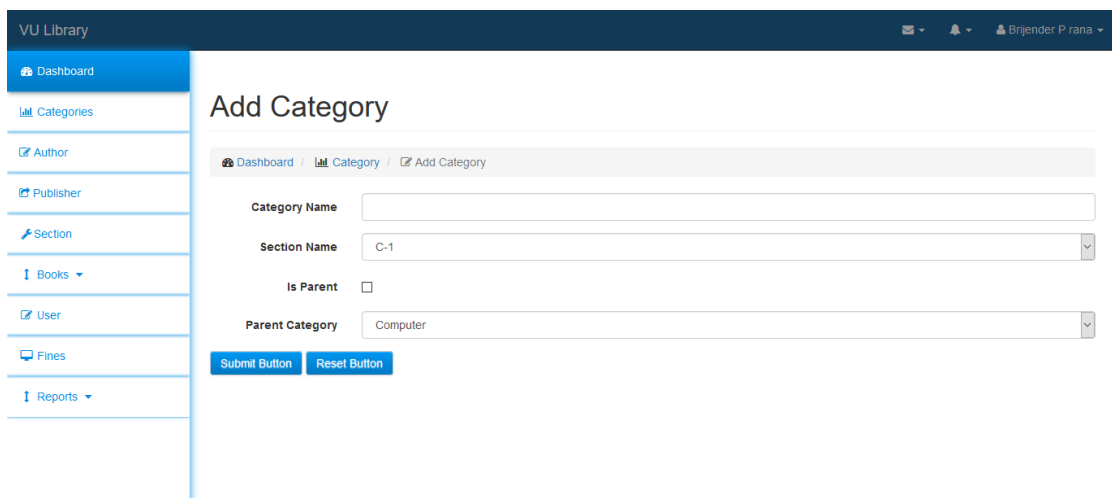
Figure 17 Category Section Add Category

5.2 List Category
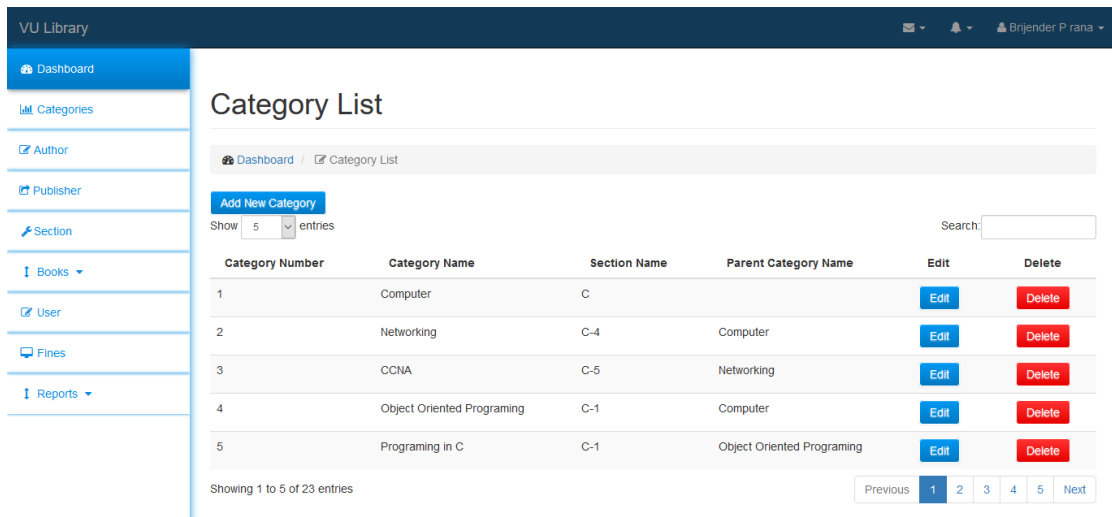
Figure 18 Category Section Category List

6. Publisher Section

Admin/Staff can make a relation with publisher different book.

6.1   Publisher Add
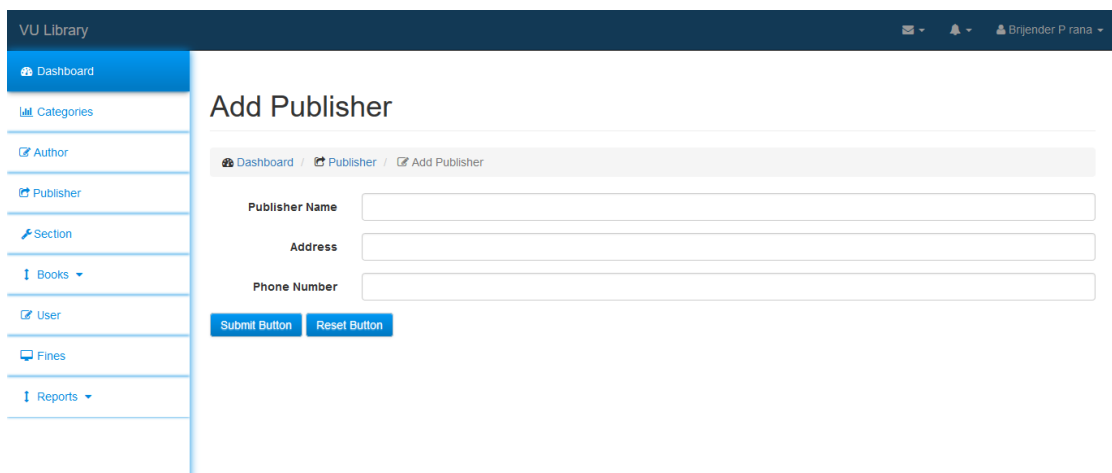
Figure 19 Publisher Section Add Publisher

6.2   Publisher List
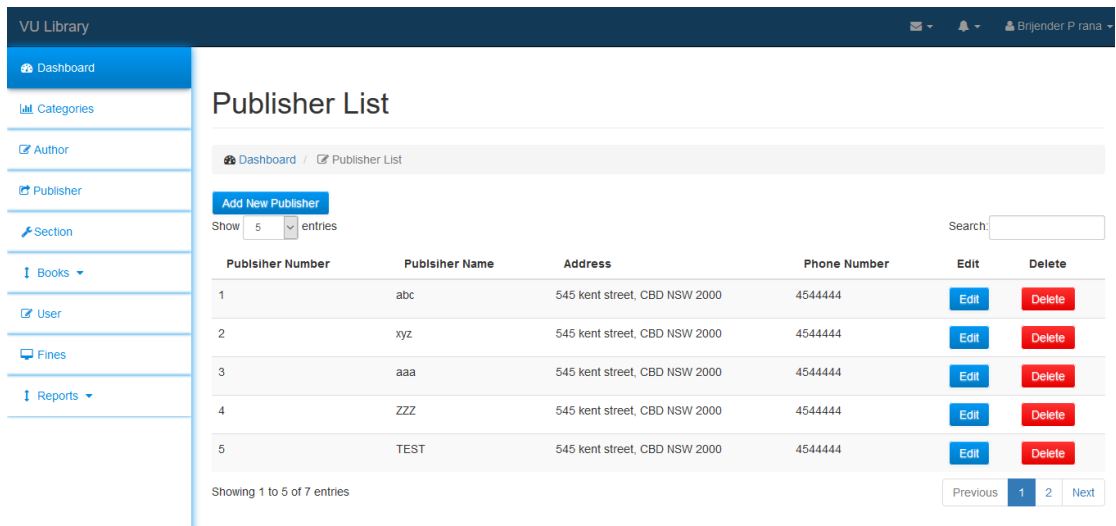
Figure 20 Publisher Section Publisher List

7. Books

Admin/Staff can manage books we make drop down menu, for adding, issuing and reserving books.

`   7.1 Books Search
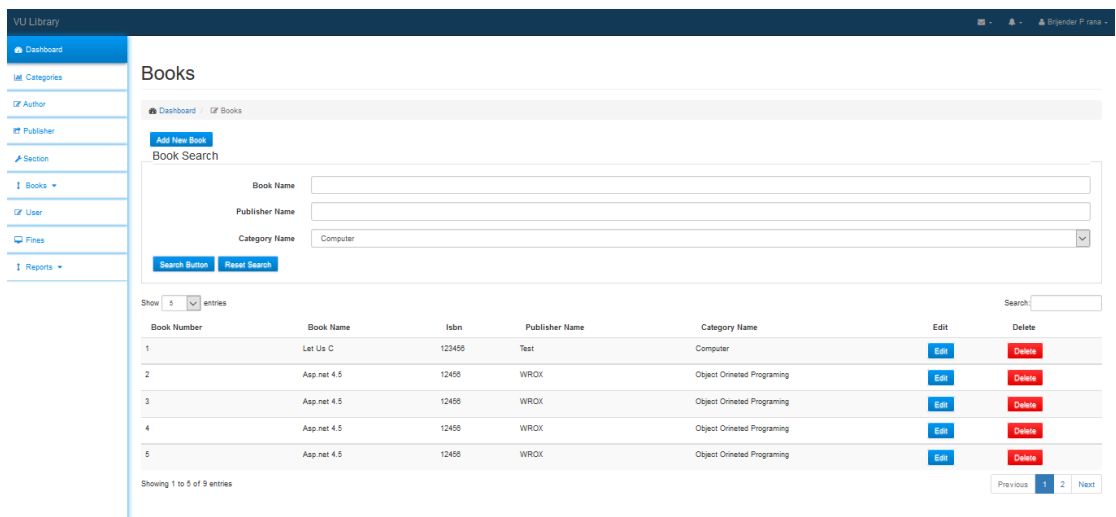
Figure 21 Search Book and List Book

7.2 Edit Books

Figure 22 Book section Add/Edit Book

8. User Section

Admin can add different user for manage portal. Student/staff also can manage their account and search book online, so they also have their account.

8.1 Create User

Figure 23 User Section Add User

8.2 User List

Figure 24 User Section User List

9. Book Report

Figure 25 Book Report

10. Student/Staff interface

Along with admin and librarian the student and staff also can search books and view their accounts.

10.1 Account Details

Figure 26 User Account Details

10.2 My Books

Figure 27 Books along with user

XIV. PROJECT MANAGEMENTS

WORK BREAKDOWN STRUCTURE (WBS)

1. Initiation
   - 1.1. Project Setup/Initiation
   - 1.2. Develop Project Charter
   - 1.3. Project Plan Documentation
   - 1.4. Project Schedule Development
   - 1.5. Weekly PMIS Reporting
   - 1.6. Risk Management
   - 1.7. Subcontract Management
   - 1.8. Project Meetings
   - 1.9. Project Charter Signed/Approved

2. Planning
   - 2.1. Business Process/Use Case Meetings
   - 2.2. Business Process/Use Case Documentation
   - 2.3. Requirements Meetings
   - 2.4. Requirements Documentation
   - 2.5. Create Preliminary Scope Statement
   - 2.6. Determine Project Team
   - 2.7. Project Team Kickoff Meeting
   - 2.8. Develop Project Plan
   - 2.9. Submit Project Plan
   - 2.10. Milestone: Project Plan Approval

3. Design
   - 3.1. Database Design Meetings
   - 3.2. Database Design Documentation
   - 3.3. Application Design Meetings
   - 3.4. Application Design Documentation
   - 3.5. System Architecture Design Meetings
   - 3.6. System Architecture Design Documentation
   - 3.7. System Integration Design Meetings
   - 3.8. System Integration Design Documentation

4. Execution
   - 4.1. Project Kickoff Meeting
   - 4.2. Procure Hardware/Software
   - 4.3. Programming
      - 4.3.1. Implementation of Design
      - 4.3.2. Programming For Admin Section
      - 4.3.3. Programming For Staff Section
      - 4.3.4. Programming For Student/Brower section
   - 4.4. Code Reviews

4.5.   Application Testing
    4.5.1. Factory Testing
    4.5.2. Issue/Bug Resolution
    4.5.3. Site Acceptance Testing

5. Deployment
    5.1.   Database Development
    5.2.   Data Conversion/Migration
    5.3.   Data QA/QC
    5.4.   Release Planning & Management
    5.5.   Issue/Bug Management
    5.6.   Server Management
    5.7.   Source Code Management
    5.8.   Database Management
    5.9.   Install Documentation
    5.10.   Installation
    5.11.   Help & Training Material Documentation
    5.12.   User Training

6. Closeout
    6.1.   Audit Procurement
    6.2.   Document Lessons Learned
    6.3.   Update Files/Records
    6.4.   Gain Formal Acceptance
    6.5.   Archive Files/Documents

SCHEDULES

Figure 28 Project Schedule -1

42

ID	(i)	Task Name	Duration	Start	Finish	Predecessors	Resource Names
21		2.10 Milestone: Project Plan Approval	1 day	Thu 9/3/15	Thu 9/3/15	20	Project Manager
22		3 Design	12 days	Fri 9/4/15	Mon 9/21/15		
23		3.1 Database Design Meetings	1 day	Fri 9/4/15	Fri 9/4/15		Project Manager[50%]
24		3.2 Database Design Documentation	1 day	Mon 9/7/15	Mon 9/7/15	23	Developer 1[50%],Developer2[5
25		3.3 Application Design Meetings	1 day	Tue 9/8/15	Tue 9/8/15	24	Project Manager[50%
26		3.4 Application Design Documentation	2 days	Wed 9/9/15	Thu 9/10/15	25	Developer 1[50%],Developer2[5
27		3.5 System Architecture Design Meetings	2 days	Fri 9/11/15	Mon 9/14/15	26	Project Manager[50%],Team
28		3.6 System Architecture Design Documentation	2 days	Tue 9/15/15	Wed 9/16/15	27	Developer 1[50%],Team
29		3.7 System Integration Design Meetings	1 day	Thu 9/17/15	Thu 9/17/15	28	Team Leader[50%],Project
30		3.8 System Integration Design Documentation	2 days	Fri 9/18/15	Mon 9/21/15	29	Developer 1[50%],Developer2[5
31		4 Execution	37 days	Tue 9/22/15	Wed 11/11/15		
32		4.1 Project Kickoff Meeting	1 day	Tue 9/22/15	Tue 9/22/15		Project Manager
33		4.2 Procure Hardware/Software	4 days	Wed 9/23/15	Mon 9/28/15	32	Developer 1[50%],Team
34		4.3 Programming	34 days	Tue 9/22/15	Fri 11/6/15		
35		4.3.1 Implementation of Design	7 days	Tue 9/29/15	Wed 10/7/15		Developer 1[50%],Te
36		4.3.2 Programming For Admin Section	7 days	Thu 10/8/15	Fri 10/16/15	35	Developer 1[50%],Developer2[5
37		4.3.3 Programming For Staff Section	7 days	Mon 10/19/15	Tue 10/27/15	36	Developer 1[50%],Developer2[5
38		4.3.4 Programming For Student/Browser section	7 days	Wed 10/28/15	Thu 11/5/15	37	Developer 1[50%],Developer2[5
39		4.4 Code Reviews	1 day	Fri 11/6/15	Fri 11/6/15		Team Leader[50%],P
40		4.5 Application Testing	3 days	Mon 11/9/15	Wed 11/11/15		
41		4.5.1 Factory Testing	1 day	Mon 11/9/15	Mon 11/9/15		Tester
42		4.5.2 Issue/Bug Fixing	1 day	Tue 11/10/15	Tue 11/10/15	41	Developer 1[50%],De

Project: Gantt_chart_123.mpp
Date: Fri 9/11/15

Task
Split
Milestone
Summary
Project Summary
External Tasks
External Milestone

Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only

Finish-only
External Tasks
External Milestone
Progress
Split

Page 2

Figure 29 Project Schedule -2

ID	i	Task Name	Duration	Start	Finish	Predecessors	Resource Names
43		4.5.3 Site Acceptance Testing	1 day	Wed 11/11/15	Wed 11/11/15	42	Tester[50%],Quality A...
44		5 Deployment	22 days	Thu 11/12/15	Fri 12/11/15		
45		5.1 Database Development	1 day	Thu 11/12/15	Thu 11/12/15		Developer 1[50%],De...
46		5.2 Data Conversion/Migration	1 day	Fri 11/13/15	Fri 11/13/15	45	Developer 1[50%],Developer2[50...
47		5.3 Data QA/QC	1 day	Mon 11/16/15	Mon 11/16/15	46	Quality Assurance M...
48		5.4 Release Planning & Management	1 day	Tue 11/17/15	Tue 11/17/15	47	Project Manager[50%],Team...
49		5.5 Issue/Bug Management	2 days	Wed 11/18/15	Thu 11/19/15	48	Developer 1[50%],De...
50		5.6 Server Management	1 day	Fri 11/20/15	Fri 11/20/15	49	Developer 1
51		5.7 Source Code Management	3 days	Mon 11/23/15	Wed 11/25/15	50	Developer2
52		5.8 Database Management	2 days	Thu 11/26/15	Fri 11/27/15	51	Developer 1[50%],De...
53		5.9 Documentation	2 days	Mon 11/30/15	Tue 12/1/15	52	Developer 1
54		5.10 Installation	1 day	Wed 12/2/15	Wed 12/2/15	53	Developer2[50%],De...
55		5.11 Help & Training Material Documentation	2 days	Thu 12/3/15	Fri 12/4/15	54	Developer 1
56		5.12 User Training	5 days	Mon 12/7/15	Fri 12/11/15	55	Trainer
57		6 Closeout	5 days	Mon 12/14/15	Fri 12/18/15		
58		6.1 Audit Procurement	1 day	Mon 12/14/15	Mon 12/14/15		Team Leader[50%],P...
59		6.2 Document Lessons Learned	1 day	Tue 12/15/15	Tue 12/15/15	58	Project Manager
60		6.3 Update Files/Records	1 day	Wed 12/16/15	Wed 12/16/15	59	Developer 1[50%],Te...
61		6.4 Gain Formal Acceptance	1 day	Thu 12/17/15	Thu 12/17/15	60	Project Manager
62		6.5 Archive Files/Documents	1 day	Fri 12/18/15	Fri 12/18/15	61	Developer 1

Legend:
Task, Split, Milestone, Summary, Project Summary, External Tasks, External MileTask
Inactive Milestone, Inactive Summary, Manual Task, Duration-only, Manual Summary Rollup, Manual Summary, Start-only
Finish-only, External Tasks, External Milestone, Progress, Split

Project: Gantt_chart_123.mpp
Date: Fri 9/11/15

Page 3

Figure 30 Project Schedule -3

44

Oct '15 | 27 | Nov '15 | 1 | 8 | 15 | 22 | Dec '15 | 29 | 6 | 13 | 20 | 27 | Jan '16 | 3 | 10 | 17 | 24 | Feb '16 | 31 | 7 | 14 | 21 | 28 | Mar '16 | 6 | 13 | 20 | 27 | Apr '16 | 3 | 10 | 17 | 24 | May '16 | 1 | 8 | 15

m Manager[10%]

%],Project Manager[50%]

hnical Team Manager[50%]

ger[50%],Project Manager[50%]

],Team Leader[50%]
Team Leader[50%]
50%],Team Leader[50%]

r

ager[50%],Team Leader[50%]
nager

Task
Split
Milestone
Summary
Project Summary
External Tasks
External Milestone

Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only

Finish-only
External Tasks
External Milestone
Progress
Split
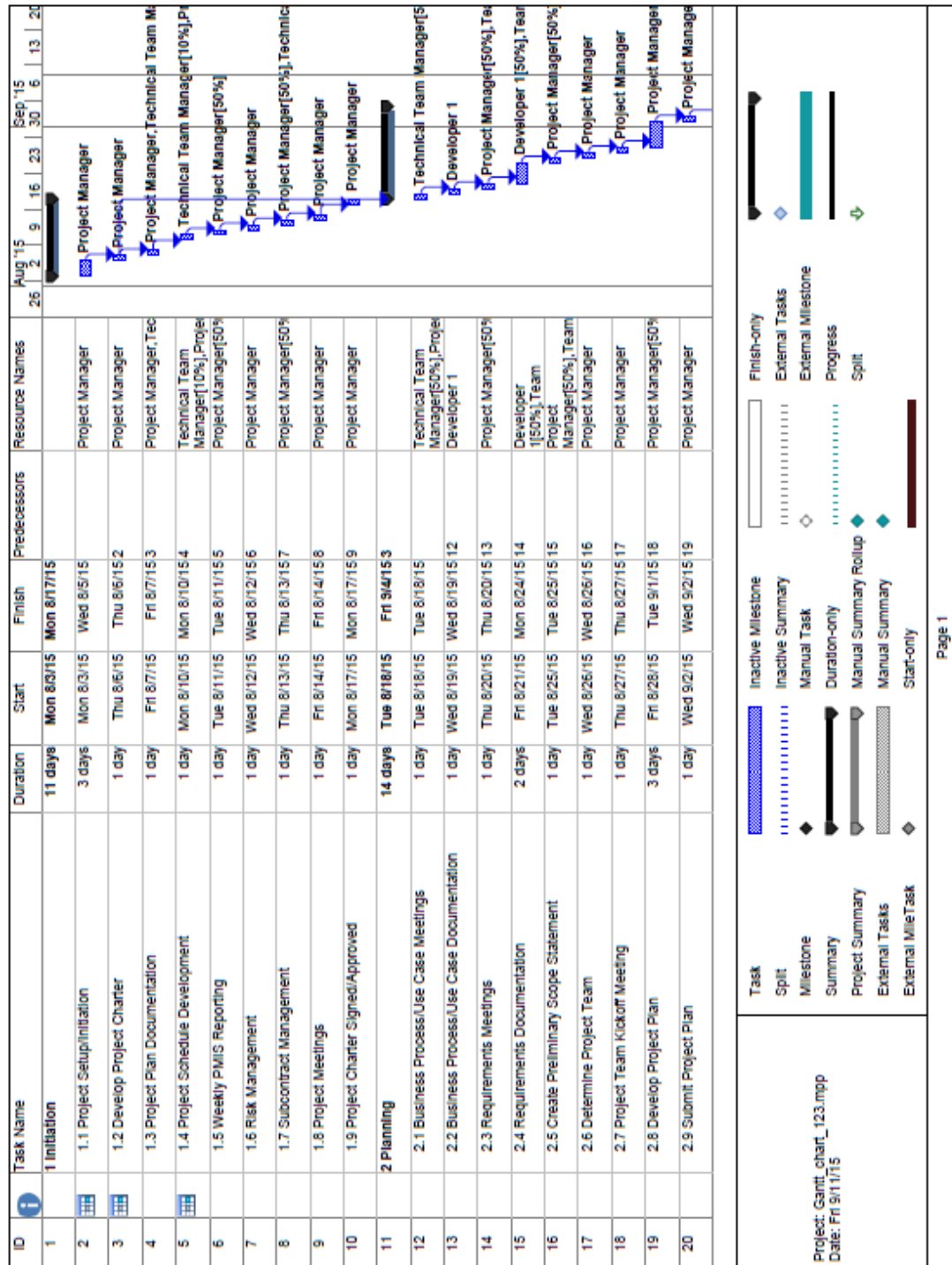
Project: Gantt_chart_123.mpp
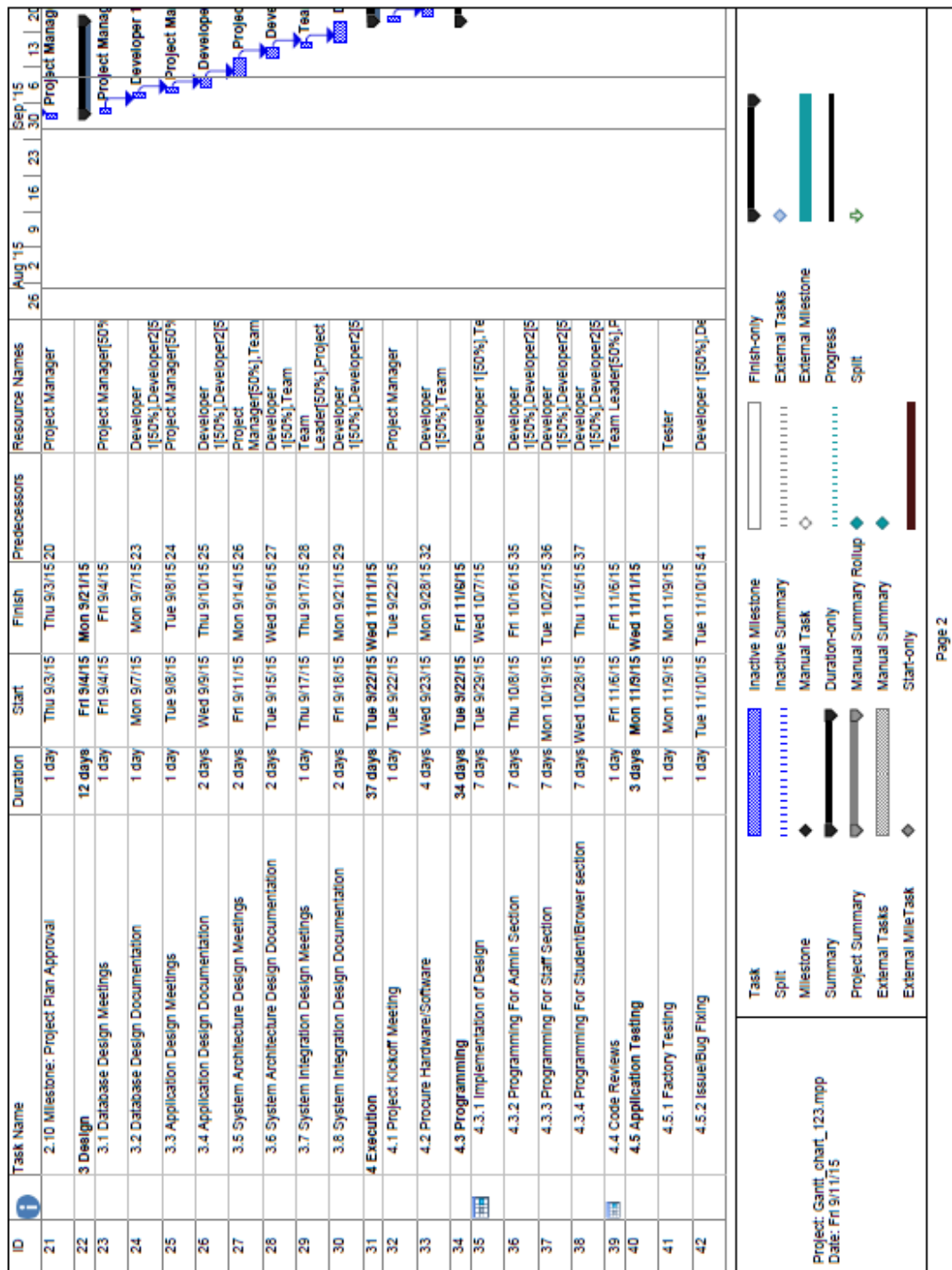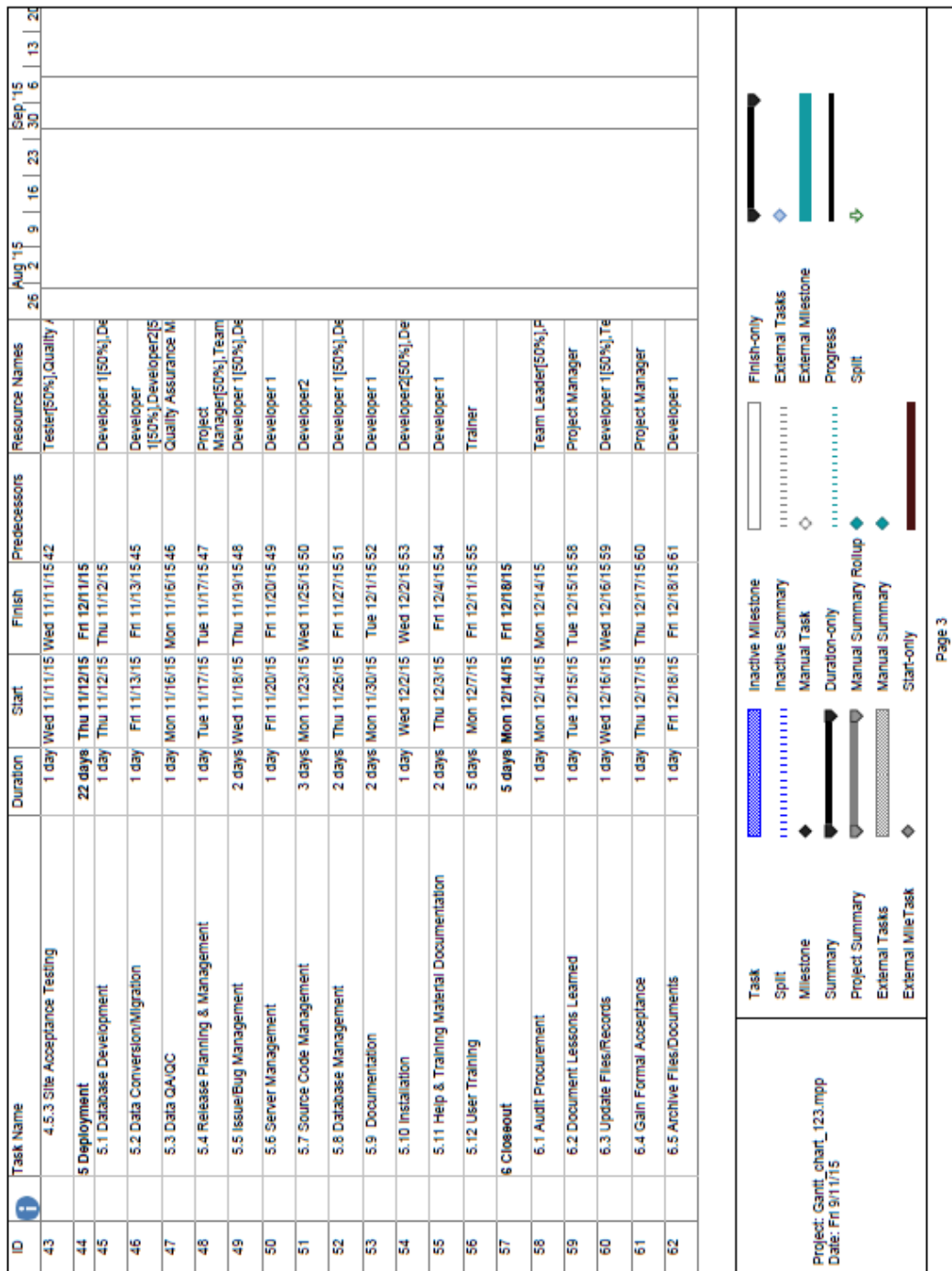Date: Fri 9/11/15

Page 4

Figure 31 Project Schedule -4

45

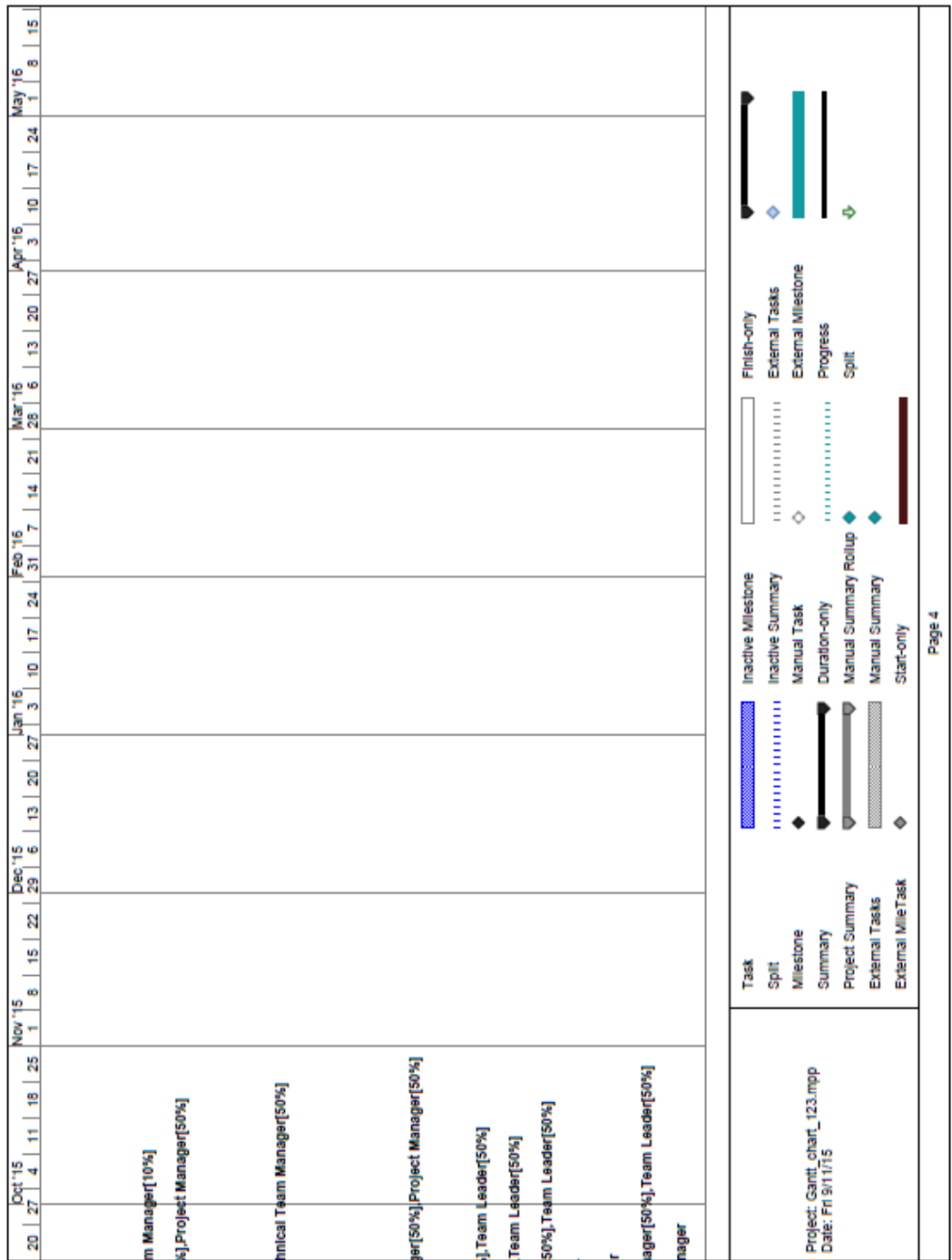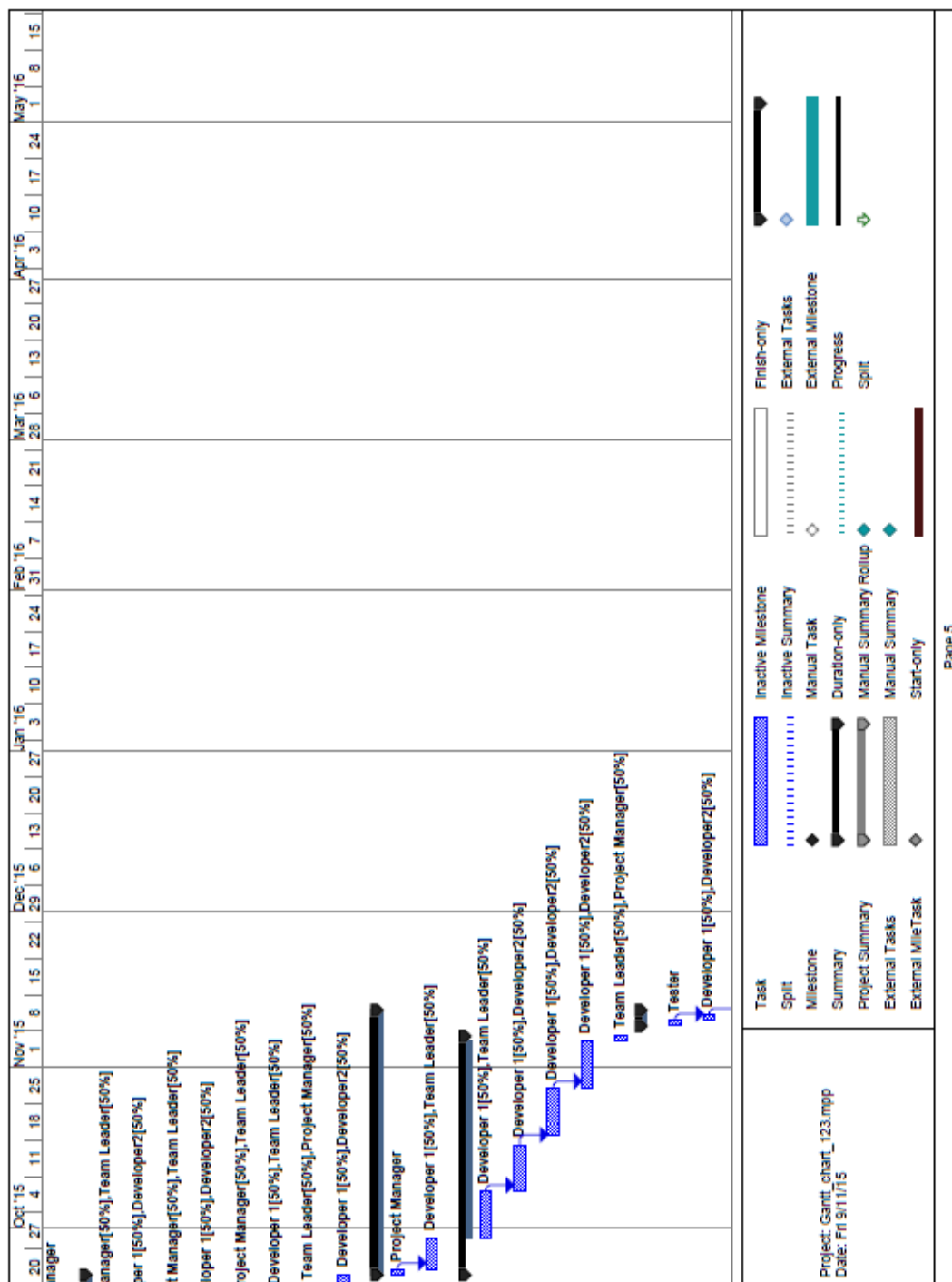Figure 32 Project Schedule -5

Figure 33 Project Schedule -6
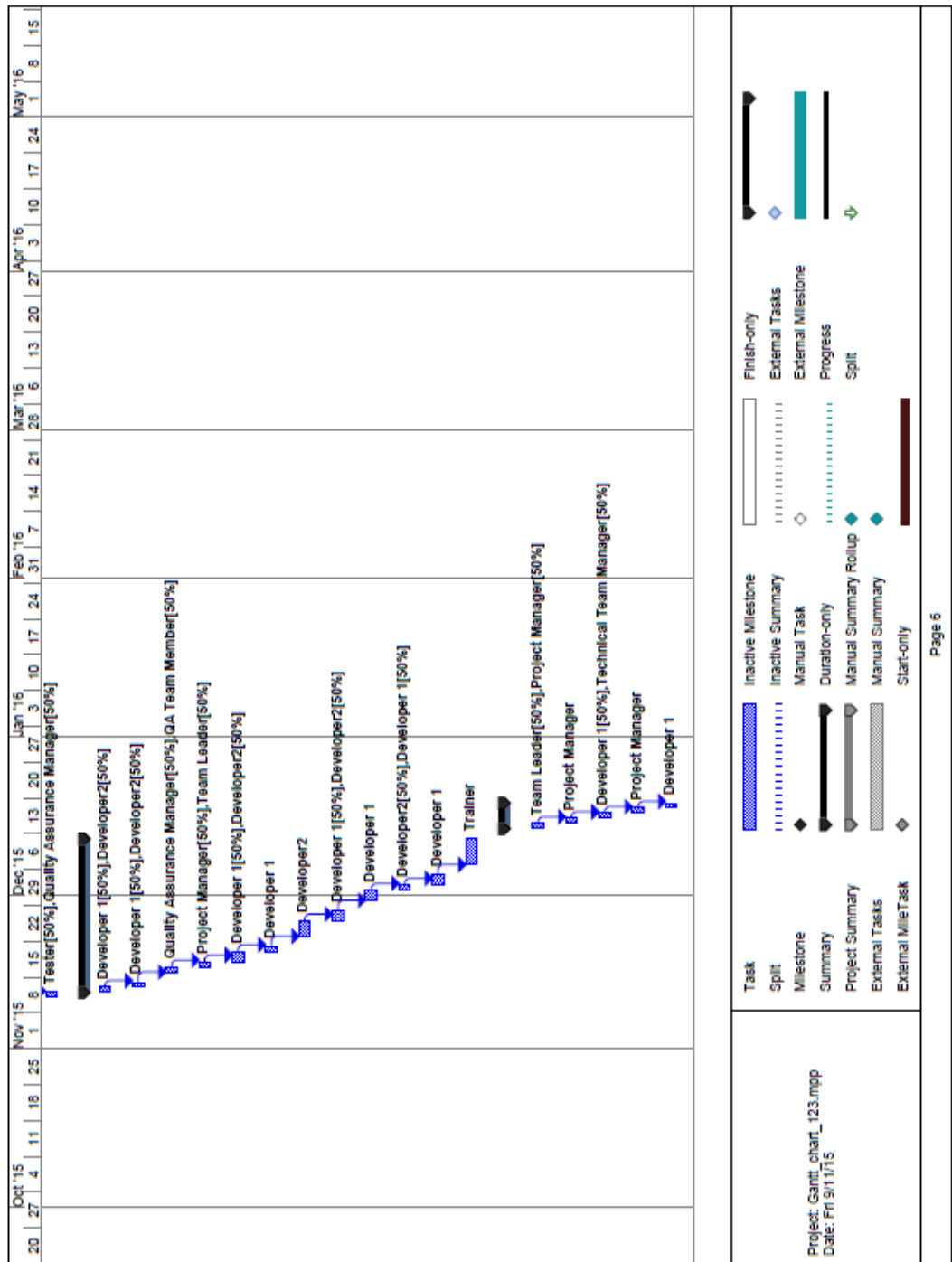
ESTIMATED COSTS

We define the cost element per item and task according to WBC, which will help us to track the progress in term of cost. Later we can evaluate time by time what is actual cost of a task compare to budgeted one.

We are giving below the 3 images for demonstrate the cost of the project.

	Task Name	Duration	Start	Finish	Predecessor	Resource	Cost	Ad
1	⊿ Initiation	11 days	Mon 8/3/15	Mon 8/17/15				$3,996.00
2	Project Setup/Initiation	3 days	Mon 8/3/15	Wed 8/5/15		Project Manager		$1,200.00
3	Develop Project Charter	1 day	Thu 8/6/15	Thu 8/6/15	2	Project Manager		$400.00
4	Project Plan Documentation	1 day	Fri 8/7/15	Fri 8/7/15	3	Project Manager		$428.00
5	Project Schedule Development	1 day	Mon 8/10/15	Mon 8/10/15	4	Technical Team		$228.00
6	Weekly PMIS Reporting	1 day	Tue 8/11/15	Tue 8/11/15	5	Project Manager		$200.00
7	Risk Management	1 day	Wed 8/12/15	Wed 8/12/15	6	Project Manager		$400.00
8	Subcontract Management	1 day	Thu 8/13/15	Thu 8/13/15	7	Project Manager		$340.00
9	Project Meetings	1 day	Fri 8/14/15	Fri 8/14/15	8	Project Manager		$400.00
10	Project Charter Signed/Approved	1 day	Mon 8/17/15	Mon 8/17/15	9	Project Manager		$400.00
11	⊿ Planning	14 days	Tue 8/18/15	Fri 9/4/15	3			$4,540.00
12	Business Process/Use Case Meetings	1 day	Tue 8/18/15	Tue 8/18/15		Technical Team		$340.00
13	Business Process/Use Case Documentatio	1 day	Wed 8/19/15	Wed 8/19/15	12	Developer 1		$240.00
14	Requirements Meetings	1 day	Thu 8/20/15	Thu 8/20/15	13	Project Manager		$360.00
15	Requirements Documentation	2 days	Fri 8/21/15	Mon 8/24/15	14	Developer 1[50%]		$560.00
16	Create Preliminary Scope Statement	1 day	Tue 8/25/15	Tue 8/25/15	15	Project Manager		$360.00
17	Determine Project Team	1 day	Wed 8/26/15	Wed 8/26/15	16	Project Manager		$400.00
18	Project Team Kickoff Meeting	1 day	Thu 8/27/15	Thu 8/27/15	17	Project Manager		$400.00
19	Develop Project Plan	3 days	Fri 8/28/15	Tue 9/1/15	18	Project Manager		$1,080.00
20	Submit Project Plan	1 day	Wed 9/2/15	Wed 9/2/15	19	Project Manager		$400.00
21	Milestone: Project Plan Approval	1 day	Thu 9/3/15	Thu 9/3/15	20	Project Manager		$400.00
22	⊿ Design	12 days	Fri 9/4/15	Mon 9/21/15				$3,560.00
23	Database Design Meetings	1 day	Fri 9/4/15	Fri 9/4/15		Project Manager		$360.00
24	Database Design Documentation	1 day	Mon 9/7/15	Mon 9/7/15	23	Developer 1[50%]		$240.00
25	Application Design Meetings	1 day	Tue 9/8/15	Tue 9/8/15	24	Project Manager		$360.00

Figure 34 Cost Estimation per task-1

49

ID		Task Name	Duration	Start	Finish	Pred	Resource Names	Cost
26		Application Design Documentation	2 days	Wed 9/9/15	Thu 9/10/15	25	Developer 1[50%]	$480.00
27		System Architecture Design Meetings	2 days	Fri 9/11/15	Mon 9/14/15	26	Project Manager	$720.00
28		System Architecture Design Documentatio	2 days	Tue 9/15/15	Wed 9/16/15	27	Developer 1[50%]	$560.00
29		System Integration Design Meetings	1 day	Thu 9/17/15	Thu 9/17/15	28	Team Leader[50%]	$360.00
30		System Integration Design Documentation	2 days	Fri 9/18/15	Mon 9/21/15	29	Developer 1[50%]	$480.00
31		**Execution**	**37 days**	**Tue 9/22/15**	**Wed 11/11/15**			**$9,780.00**
32		Project Kickoff Meeting	1 day	Tue 9/22/15	Tue 9/22/15		Project Manager	$400.00
33		Procure Hardware/Software	4 days	Wed 9/23/15	Mon 9/28/15	32	Developer 1[50%]	$1,120.00
34		**Programming**	**34 days**	**Tue 9/22/15**	**Fri 11/6/15**			**$7,000.00**
35	▦	Implementation of Design	7 days	Tue 9/29/15	Wed 10/7/15		Developer 1[50%]	$1,960.00
36		Programming For Admin Section	7 days	Thu 10/8/15	Fri 10/16/15	35	Developer 1[50%]	$1,680.00
37		Programming For Staff Section	7 days	Mon 10/19/15	Tue 10/27/15	36	Developer 1[50%]	$1,680.00
38		Programming For Student/Brower secti	7 days	Wed 10/28/15	Thu 11/5/15	37	Developer 1[50%]	$1,680.00
39	▦	Code Reviews	1 day	Fri 11/6/15	Fri 11/6/15		Team Leader[50%]	$360.00
40		**Application Testing**	**3 days**	**Mon 11/9/15**	**Wed 11/11/15**			**$900.00**
41		Factory Testing	1 day	Mon 11/9/15	Mon 11/9/15		Tester	$280.00
42		Issue/Bug Fixing	1 day	Tue 11/10/15	Tue 11/10/15	41	Developer 1[50%]	$240.00
43		Site Acceptance Testing	1 day	Wed 11/11/15	Wed 11/11/15	42	Tester[50%],Qua	$380.00
44		**Deployment**	**22 days**	**Thu 11/12/15**	**Fri 12/11/15**			**$5,920.00**
45		Database Development	1 day	Thu 11/12/15	Thu 11/12/15		Developer 1[50%]	$240.00
46		Data Conversion/Migration	1 day	Fri 11/13/15	Fri 11/13/15	45	Developer 1[50%]	$240.00

Figure 35 Cost Estimate per Task -2

#	Task Name	Duration	Start	Finish	ID	Resource Names	Cost
46	Data Conversion/Migration	1 day	Fri 11/13/15	Fri 11/13/15	45	Developer 1[50%]	$240.00
47	Data QA/QC	1 day	Mon 11/16/15	Mon 11/16/15	46	Quality Assuranc	$360.00
48	Release Planning & Management	1 day	Tue 11/17/15	Tue 11/17/15	47	Project Manager	$360.00
49	Issue/Bug Management	2 days	Wed 11/18/15	Thu 11/19/15	48	Developer 1[50%]	$480.00
50	Server Management	1 day	Fri 11/20/15	Fri 11/20/15	49	Developer 1	$240.00
51	Source Code Management	3 days	Mon 11/23/15	Wed 11/25/15	50	Developer2	$720.00
52	Database Management	2 days	Thu 11/26/15	Fri 11/27/15	51	Developer 1[50%]	$480.00
53	Documentation	2 days	Mon 11/30/15	Tue 12/1/15	52	Developer 1	$480.00
54	Installation	1 day	Wed 12/2/15	Wed 12/2/15	53	Developer2[50%	$240.00
55	Help & Training Material Documentation	2 days	Thu 12/3/15	Fri 12/4/15	54	Developer 1	$480.00
56	User Training	5 days	Mon 12/7/15	Fri 12/11/15	55	Trainer	$1,600.00
57	**◢ Closeout**	**5 days**	**Mon 12/14/15**	**Fri 12/18/15**			**$1,660.00**
58	Audit Procurement	1 day	Mon 12/14/15	Mon 12/14/15		Team Leader[50	$360.00
59	Document Lessons Learned	1 day	Tue 12/15/15	Tue 12/15/15	58	Project Manager	$400.00
60	Update Files/Records	1 day	Wed 12/16/15	Wed 12/16/15	59	Developer 1[50%	$260.00
61	Gain Formal Acceptance	1 day	Thu 12/17/15	Thu 12/17/15	60	Project Manager	$400.00
62	Archive Files/Documents	1 day	Fri 12/18/15	Fri 12/18/15	61	Developer 1	$240.00

Figure 36 Cost Estimation Per task -3

51

RISKS

The risk management helps us to aware and identify the possible risk in the project. It is evaluation of both the probability of any adverse incident happen and what will impact of that event. The affect can be financial loss, project delay and loss of performance.

Table 5 Risk Matrix

Risk name	Description/Cause	Owner	Impact	Probability
New technology adoption	The current user/clients staff may not ready to adopt the new technology as they are use to what they are following from ages.	Staff	High	High
Project time extinction	The project may get extended if any of the contractors get sick or if the deliveries get delayed	Contractors	Medium	Low
Requirement redefinition	There is a plausibility if re defining the requirements by the client and if the client wanted something else that is not delivered or not happy with the software	Client, Vendor	Medium	Low
External support	The project depends on many different contractors anything un expected can happened.	Contractors	Very height	Medium
Financial risk	There can be the case if the client cannot afford the estimated budget	client	Low	Low

XV. REFERENCES

Anicas, M. (2014, May 30). *5 Common Server Setups For Your Web Application.*
Retrieved September 10, 2015, from DigitalOcean Inc.:
https://www.digitalocean.com/community/tutorials/5-common-server-
setups-for-your-web-application

Chung, L. (2015). *Non-Functional Requirements.* Retrieved September 10, 2015, from
Department of Computer Science, The University of Texas at Dallas:
https://www.utdallas.edu/~chung/SYSM6309/NFR-18-4-on-1.pdf

Nielsen, J. (1993, January 1). *Response Times: The 3 Important Limits.* Retrieved
September 9, 2015, from http://www.nngroup.com/articles/response-times-
3-important-limits/

Robertson, S., & Robertson, J. C. (2015). *Cultural and Political Requirements: Type 16.*
Retrieved September 10, 2015, from
http://flylib.com/books/en/4.445.1.144/1/