ONLINE ENGINEERING BOOKS MANAGEMENT

A Technical mini project Report submitted in partial fulfillment of the requirements of the award of degree of

BACHELOR OF TECHNOLOGY

COMPUTER SCIENCE & ENGINEERING

Submitted by

PAVIRALA NAGA SATYA SAI	**(18A21A0564)**
KUMMARAPURUGU RAMESH	**(18A21A05A1)**
POLISETTY PAVAN SITA RAM	**(18A21A0580)**
TUMU RAM SAI JAGAN	**(18A21A05A5)**
SALADI RAMANJANEYULU	**(19A25A0511)**

2021-2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SWARNANDHRA COLLEGE OF ENGINEERING & TECHNOLOGY

(AUTONOMOUS)

Accredited by National Board of Accreditation, AICTE, New Delhi

(Approved by AICTE & Affiliated to JNTUK Kakinada) An ISO 9001:2000 certified Institution

Seetharampuram, Narasapur-534280, West Godavari (Dt.), A.P

SWARNANDHRA COLLEGE OF ENGINEERING AND TECHNOLOGY

(AUTONOMOUS)

(Accredited by NBA and NAAC with 'A' Grade(CGPA3.32/4))

Approved by AICTE, Autonomous with JNTU Kakinada)

Seetharampuram, Narsapur-534 280

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the mini project Report entitled " **ONLINE ENGINEERING BOOKS MANAGEMENT**", is being submitted by **PAVIRALA NAGA SATYA SAI (18A21A0564), KUMMARAPURUGU RAMESH (18A21A05A1), POLISETTY PAVAN SITA RAM (18A21A0580), TUMU RAM SAI JAGAN (18A21A05A5), SALADI RAMANJANEYULU (19A25A0511)** in partial fulfillment for the award of the degree of "**Bachelor of Technology** in Computer Science & Engineering" during the academic year 2021-2022 and it has been found worthy of acceptance according to the requirements of the autonomous.

Mini Project Co-Ordinator

DR.B.RAMA KRISHNA M.Tech.,Ph.D.,

Assistant Professor/ CSE

Head of Department

Dr. P. Srinivasulu M.Tech Ph.D.,

Professor & Head / CSE

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of every task during my dissertation would be complete without the mention of the people who made it possible. I consider it my privilege to my gratitude and respect to all who guided, inspired and helped me in completion of my mini project.

I extend my heartfelt gratitude to the almighty for giving me strength in Proceeding with this mini project report on " **ONLINE ENGINEERING BOOKS MANAGEMENT**"

I owe my special thanks to the **Dr.S.Ramesh Babu, M.Tech.,Secretary and Correspondent**, Swarnandhra College of Engineering and Technology, Seetharampuram for providing necessary arrangements to carry out this seminar

I would like to express my sincere thanks to **Dr.P.Srinivasulu, M.Tech., Ph.D.,** Professor & Head, Department of Computer Science and Engineering for valuable suggestions at the time of need.

I would like to express my profound sense of gratitude to Mini Project Coordinator **DR.B.RAMA KRISHNA M.Tech.,Ph.D.,** Assistant Professor, Department of Computer Science and Engineering for his consistent encouragement and earnest support to complete it successfully

PAVIRALA NAGA SATYA SAI	**(18A21A0564)**
KUMMARAPURUGU RAMESH	**(18A21A05A1)**
POLISETTY PAVAN SITA RAM	**(18A21A0580)**
TUMU RAM SAI JAGAN	**(18A21A05A5)**
SALADI RAMANJANEYULU	**(19A25A0511)**

Abstract

"Online engineering books (learn to earn)" is an online based web portal which provides free access to all 8 semesters engineering books of all the main engineering streams. The students or the faculties can access this portal and get free access to the books for improving their knowledge.

Our main target is to avoid unemployment of engineering students if they have good knowledge and no arrears so that they can be placed in a good company and can contribute to the development of nation

Owing to the advancement of technology, organization of an Online Engineering Books becomes much simple. The Online Engineering Books Management has been designed to computerize and automate the operations performed over the information about the members, book issues and returns and all other operations. This computerization of books helps in many instances of its maintenances. It reduces the workload of management as most of the manual work done is reduced.

TABLE OF CONTENTS

ABSTRACT

1. INTRODUCTION **1**

1.1 PROJECT AIMS AND OBJECTIVES 1

1.2 BACKGROUND OF PROJECT 2

2. SYSTEM ANALYSIS **3**

2.1 SOFTWARE REQUIREMENT SPECIFICATION 3

2.2 EXISTING VS PROPOSED 7

2.3 SOFTWARE TOOL USED 8

3. SYSTEM DESIGN **10**

3.1 TABLE DESIGN 10

3.2 DATA FLOW DIAGRAM'S 12

4. SYSTEM IMPLEMENTATION **16**

4.1 SCREEN SHOTS 16

5. SYSTEM TESTING **25**

5.1 UNIT TESTING 25

5.2 INTEGRATION TESTING 26

6. CONCLUSION & FUTURE SCOPE **27**

7. REFERENCES **28**

8. APPENDIX **29**

CHAPTER 1

INTRODUCTION

This chapter gives an overview about the aim , objectives ,background and
operation environment of the system.

1.1 PROJECT AIMS AND OBJECTIVES

The project aims and objectives that will be achieved after completion of this project are discussed in this subchapter. The aims and objectives are as follows:

- Online books reading.
- A search column to search availability of books.
- Facility to download required books.
- Once user is successfully logged in he can access to all books online or if he wants a hard copy of a book he can also purchase them.
  - An Admin login page where admin can add books, videos or page sources
  - Open link for Learning Websites
  - The students can access calculator, aptitude books for placement and we embedded an IDE for coding of computer science students for hands on practice.

1.2 BACKGROUND OF PROJECT

Online engineering books is an application which refers to management of books which are generally small or medium in size. It is used by user to manage the online books using a online website for books where he/she can add new books, videos and Page sources.

Books and student maintenance modules are also included in this system which would keep track of the students using the Online engineering books and also a detailed description about the books contains in this website. With this computerized system there will be no loss of book record or member record which generally happens when a non computerized system is used.

All these modules are able to help user to manage the online books with more convenience and in a more efficient way as compared to management of books which are not computerized.

PROCESSOR	PENTIUM-IV PROCESSOR OR BETTER PROCESSOR
OPERAT ING SYSTEM	WINDOWS VISTA ,WINDOWS 7, UBUNTU
MEMORY	1GB RAM OR MORE
HARD DISK SPACE	MINIMUM 3 GB FOR DATABASE USAGE FOR FUTURE
DATABASE	ORACLE 10G
APPLICATION SERVER	APACHE TOMCAT 5.0/6.X
FRONT-END DESIGN	HTML,CSS
BACK-END DESIGN	SERVLETS
DATABASE CONNECTIVITY	JDBC

CHAPTER 2

SYSTEM ANALYSIS

In this chapter, we will discuss and analyze about the developing process of Online engineering books Management including software requirement specification (SRS) and comparison between existing and proposed system . The functional and non functional requirements are included in SRS part to provide complete description and overview of system requirement before the developing process is carried out. Besides that, existing vs proposed provides a view of how the proposed system will be more efficient than the existing one.

2.1 SOFTWARE REQUIREMENT SPECIFICATION

2.1.1GENERAL DESCRIPTION

PRODUCT DESCRIPTION:

Online engineering books is a online website which helps user to manage the website daily activity in electronic format. It reduces the risk of paper work such as file lost, file damaged and time consuming.
It can help user to manage the transaction or record more effectively and time-saving.

PROBLEM STATEMENT:

The problem occurred before having computerized system includes:

- File lost when computerized system is not implemented file is always lost because of human environment.Some times due to some human error there may be a loss of records.

- File damaged when a computerized system is not there file is always lost due to some accident like spilling of water by some member on file accidentally. Besides some natural disaster like floods or fires may also damage the files.

- Difficult to search record

  When there is no computerized system there is always a difficulty in searching of records if the records are large in number .

- Space consuming

  After the number of records become large the space for physical storage of file and records also increases if no computerized system is implemented.

- Cost consuming

  As there is no computerized system the to add each record paper will be needed which will increase the cost for the management of library.

2.1.2 SYSTEM OBJECTIVES

- Improvement in control and performance

  The system is developed to cope up with the current issues and problems of library. The system can add user, validate user and is also bug free.

- Save cost

  After computerized system is implemented less human force will be required to maintain the online books thus reducing the overall cost.

- Save time

  User is able to search record by using few clicks of mouse and few search keywords thus saving his valuable time.

- Option of online Notice board

  User will be able to provide a detailed description of workshops going in the college as well as in nearby colleges

- Lecture Notes

  Faculty have a facility to upload lectures notes in a pdf file having size not more than 10mb.

2.1.3 SYSTEM REQUIREMENTS

2.1.3.1 NON FUNCTIONAL REQUIREMENTS

- Product Requirements_

EFFICIENCY REQUIREMENT

When online enginnering books will be implemented user and admin will easily access online books as searching and book transaction will be very faster.

RELIABILITY REQUIREMENT

The system should accurately performs member registration ,member validation , report generation, book transaction and search

USABILITY REQUIREMENT

The system is designed for a user friendly environment so that student and staff of library can perform the various tasks easily and in an effective way.

ORGANIZATIONAL REQUIREMENT

IMPLEMENTATION REQUIREMNTS

In implementing whole system it uses html and css scripting language in front end and servlets which will be used for database connectivity and the backend i.e the database part is developed using oracle 10g.

DELIVERY REQUIREMENT

The whole system is expected to be delivered in six months of time with a weekly evaluation by the project guide.

2.1.3.2 FUNCTIONAL REQUIREMENTS

1. NORMAL USER

1.1 USER LOGIN

This feature used by the user to login into system. They are required to enter user id and password before they are allowed to enter the system .The user id and password will be verified and if invalid id is there user is allowed to not enter the system.

Functional requirements

-user id is provided when they register

-The system must only allow user with valid id and password to enter the system

-The system performs authorization process which decides what user level can acess to.

-The user must be able to logout after they finished using system.

1.2 REGISTER NEW USER

This feature can be performed by all users to register new user to create account.

Functional requirements

-System must be able to verify information.

-System must be able to delete information if information is wrong.

1.3 REGISTER NEW BOOK

This feature allows to add new books to the online books website

Functional requirements

-System must be able to verify information

-System must be able to enter number of copies into table.

- System must be able to not allow two books having same book id.

1.4 SEARCH BOOK

DESCRIPTION OF FEATURE

This feature is found in book maintenance part . we can search book based on book id , book name , publication or by author name.

Functional requirements

- System must be able to search the database based on select search type.
- System must be able to filter book based on keyword entered.
- System must be able to show the filtered book in table view.

2.1.4 SOFTWARE AND HARDWARE REQUIREMENTS

This section describes the software and hardware requirements of the system

2.1.4.1 SOFTWARE REQUIREMENTS

- Operating system- Windows,Ubuntu is used as the operating system as it is stable and supports more features and is more user friendly
- Database ORACLE 10G-ORACLE 10G is used as database as it easy to maintain and retrieve records by simple queries which are in English language which are easy to understand and easy to write.
- Development tools and Programming language- HTML is used to write the whole code and develop webpages with CSS, JDBC for database connectivity and servlets for back end design.

2.1.4.2 HARDWARE REQUIREMENTS

Intel core i5 2nd generation is used as a processor because it is fast than other processors an provide reliable and stable and we can run our pc for longtime. By using this processor we can keep on developing our project without any worries.

- Ram 1 gb is used as it will provide fast reading and writing capabilities and will in turn support in processing.

Existing System:

- Early days Libraries are managed manually. It required lot of time to record or to retrieve the details. The employees who have to record the details must perform their job very carefully. Even a small mistake would create a lot of problems. Security of information is very less. Report generations of all the information is very tough task.
- Maintenance of Library catalogue and arrangement of the books to the catalogue is very complex task. In addition to its maintenance of member details, issue dates and return dates etc. manually is a complex task.
- All the operations must be performed in perfect manner for the maintenance of the library with out any degradation which may finally result in the failure of the entire system.

Proposed System:

To solve the inconveniences as mentioned in the existing system, an **Online Engineering Books Management** is proposed. The proposed system contains the following features:

- The students will register them through Online

- Individually each member will have his account through which he can access the information he needs.

- Book details like authors, number of copies totally maintained by the website, present available number of books, reference books, non-reference books etc. all this information can be made handy.

- Regarding the members designation, number of books was issued.

- Issue dates and returns of each member is maintained separately and fine charged if there is any delay in returning the book.

- Administrator can add, update the books.

- Time consuming is low, gives accurate results, reliability can be improved with the help of security.

2.3 SOFTWARE TOOLS USED

The whole Project is divided in two parts the front end and the back end.

2.3.1 Front end

The front end is designed using of HTML and CSS

- HTML- **HTML** or **Hyper Text Markup Language** is the main markup language for creating web pages and other information that can be displayed in a web browser.HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets (like <html>), within the web page content. HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent *empty elements* and so are unpaired, for example <img>. The first tag in a pair is the *start tag*, and the second tag is the *end tag* (they are also called *opening tags* and *closing tags*). In between these tags web designers can add text, further tags, comments and other types of text-based content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page.HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages.

- CSS- **Cascading Style Sheets**(**CSS**) is a style sheet language used fordescribing the look and formatting of a document written in a markup language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification.

- ORACLE **- ORACLE** database is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information. A database server is the key to solving the problems of information management. In general, a server reliably manages a large amount of data in a multiuser environment so that many users can concurrently access the same data. All this is accomplished while delivering high performance. A database server also prevents unauthorized access and provides efficient solutions for failure recovery.Oracle Database is the first database designed for enterprise grid computing, the most flexible and cost effective way to manage information and applications. Enterprise grid computing creates large pools of industry-standard, modular storage and servers. With this architecture, each new system can be rapidly provisioned from the pool of components. There is no need for peak workloads, because capacity can be easily added or reallocated from the resource pools as needed.The database has logical structures and physical structures. Because the physical and logical structures are separate, the physical storage of data can be managed without affecting the access to logical storage structures.

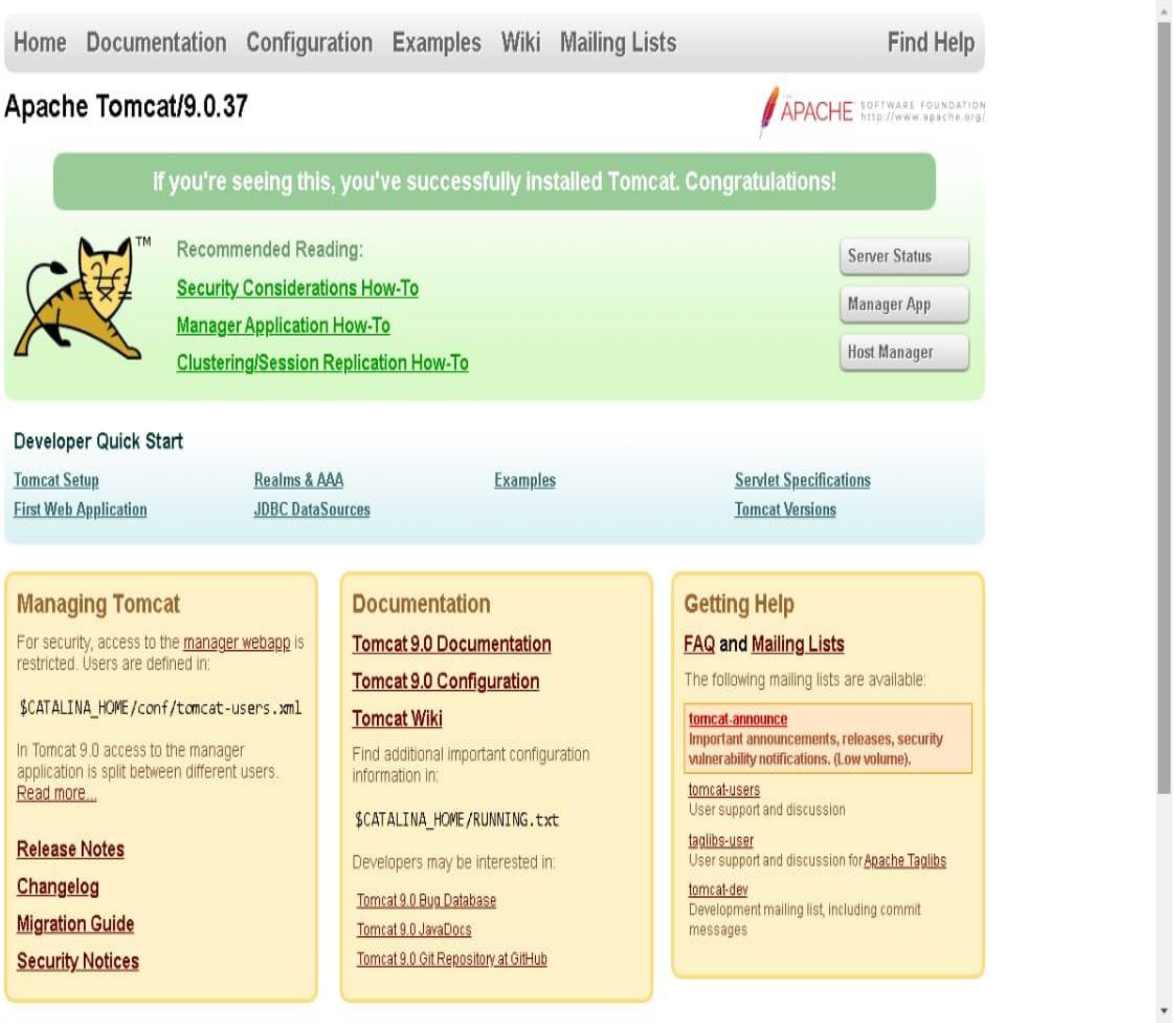
- JDBC -**JDBC** or Java Database Connectivity is a specification from Sun microsystems that provides a standard abstraction (that is API or Protocol) for java applications to communicate with various databases. It provides the language with java database connectivity standard. It is used to write programs required to access databases. JDBC along with the database driver is capable of accessing databases and spreadsheets. The enterprise data stored in a relational database (RDB) can be accessed with the help of JDBC APIs. Enterprise applications that are created using the JAVA EE technology need to interact with databases to store application-specific information. So, interacting with a database requires efficient database connectivity which can be achieved by using the ODBC(Open database connectivity) driver. This driver is used with JDBC to interact or communicate with various kinds of databases such as Oracle, MS Access, My SQL and SQL server database.

- SERVLETS - **SERVLETS** are programs that run on a Web or Application server and act as a middle layer between a requests coming from a Web browser or other HTTP client and databases or applications on the HTTP server.Using Servlets, you can collect input from users through web page forms, present records from a database or another source, and create web pages dynamically.Java Servlets often serve the same purpose as programs implemented using the Common Gateway Interface (CGI). But Servlets offer several advantages in comparison with the CGI. Java Servlets are Java classes run by a web server that has an interpreter that supports the which are a standard part of the Java's enterprise edition, an expanded version of the Java class library that supports large-scale development projects.These classes implement the Java Servlet and JSP specifications. At the time of writing this tutorial, the versions are Java Servlet 2.5 and JSP 2.1.Java servlets have been created and compiled just like any other Java class. After you install the servlet packages and add them to your computer's classpath. you can compile servlets with the JDK's Java compiler or any other current compile

CHAPTER 3

SYSTEM DESIGN

3.1 HOME PAGE OF APACHE TOMCAT



Apache Tomcat/9.0.37

If you're seeing this, you've successfully installed Tomcat. Congratulations!

Recommended Reading:

Security Considerations How-To

Manager Application How-To

Clustering/Session Replication How-To

Developer Quick Start

Tomcat Setup

First Web Application

Realms & AAA

JDBC DataSources

Examples

Servlet Specifications

Tomcat Versions

Managing Tomcat

For security, access to the manager webapp is restricted. Users are defined in:

```
$CATALINA_HOME/conf/tomcat-users.xml
```

In Tomcat 9.0 access to the manager application is split between different users. Read more...

Release Notes

Changelog

Migration Guide

Security Notices

Documentation

Tomcat 9.0 Documentation

Tomcat 9.0 Configuration

Tomcat Wiki

Find additional important configuration information in:

```
$CATALINA_HOME/RUNNING.txt
```

Developers may be interested in:

Tomcat 9.0 Bug Database

Tomcat 9.0 JavaDocs

Tomcat 9.0 Git Repository at GitHub

Getting Help

FAQ and Mailing Lists

The following mailing lists are available:

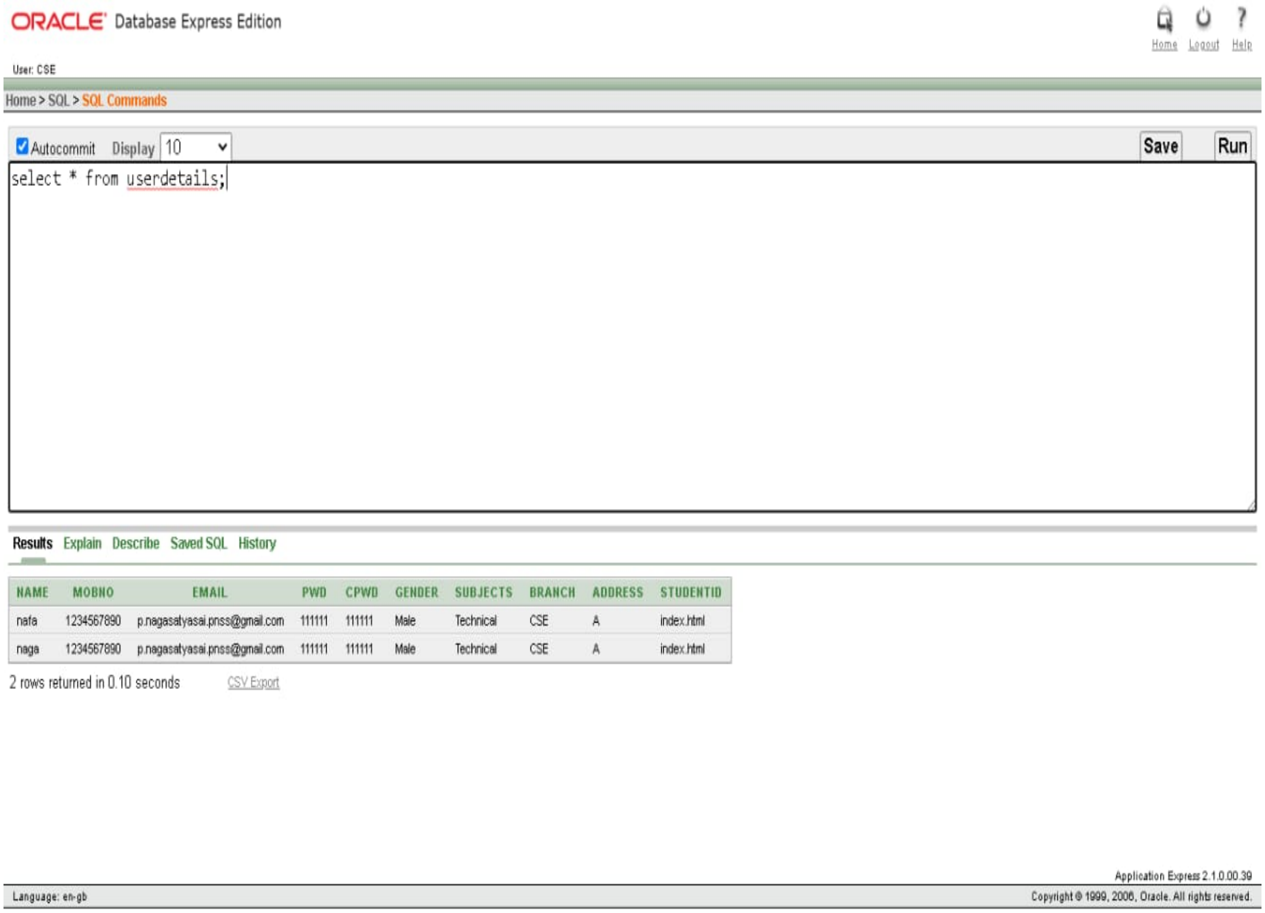
tomcat-announce
Important announcements, releases, security vulnerability notifications. (Low volume).

tomcat-users
User support and discussion

taglibs-user
User support and discussion for Apache Taglibs

tomcat-dev
Development mailing list, including commit messages

3.2 DATABASE TABLE



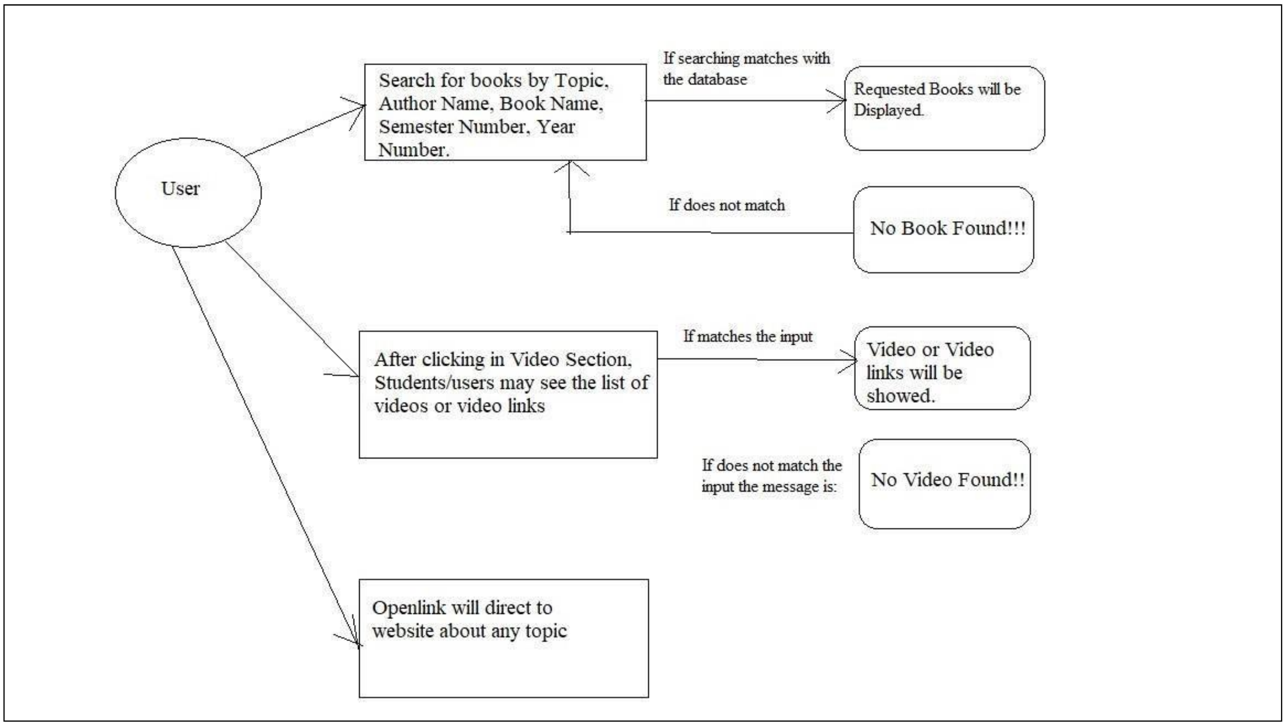
ORACLE Database Express Edition
Home
Logout
Help
User: CSE
Home > SQL > SQL Commands
☑ Autocommit
Display 10
Save
Run
Results
Explain
Describe
Saved SQL
History
CSV Export
Language: en-gb
Application Express 2.1.0.00.39
Copyright © 1999, 2006, Oracle. All rights reserved.

```
select * from userdetails;
```

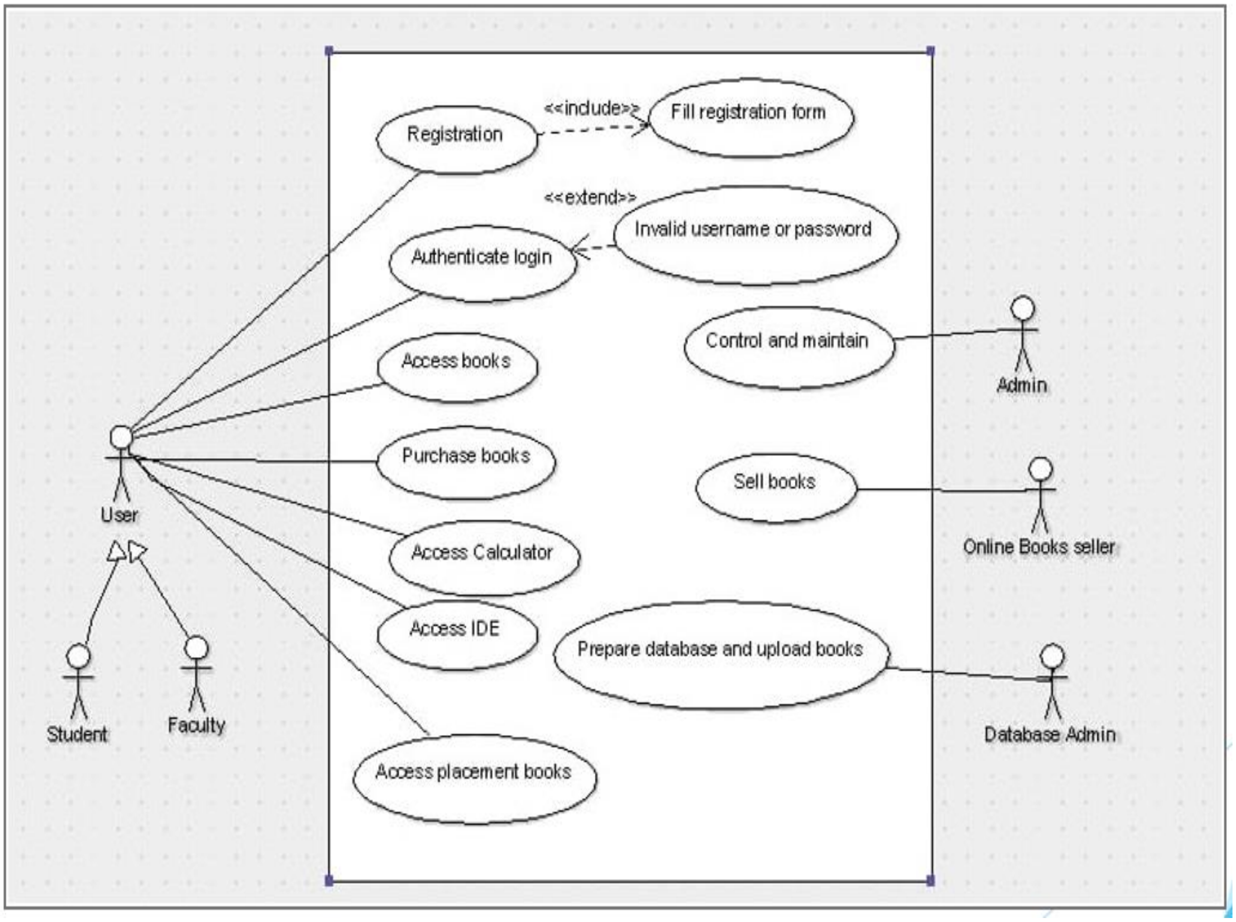
NAME	MOBNO	EMAIL	PWD	CPWD	GENDER	SUBJECTS	BRANCH	ADDRESS	STUDENTID
nafa	1234567890	p.nagasatyasai.pnss@gmail.com	111111	111111	Male	Technical	CSE	A	index.html
naga	1234567890	p.nagasatyasai.pnss@gmail.com	111111	111111	Male	Technical	CSE	A	index.html

2 rows returned in 0.10 seconds

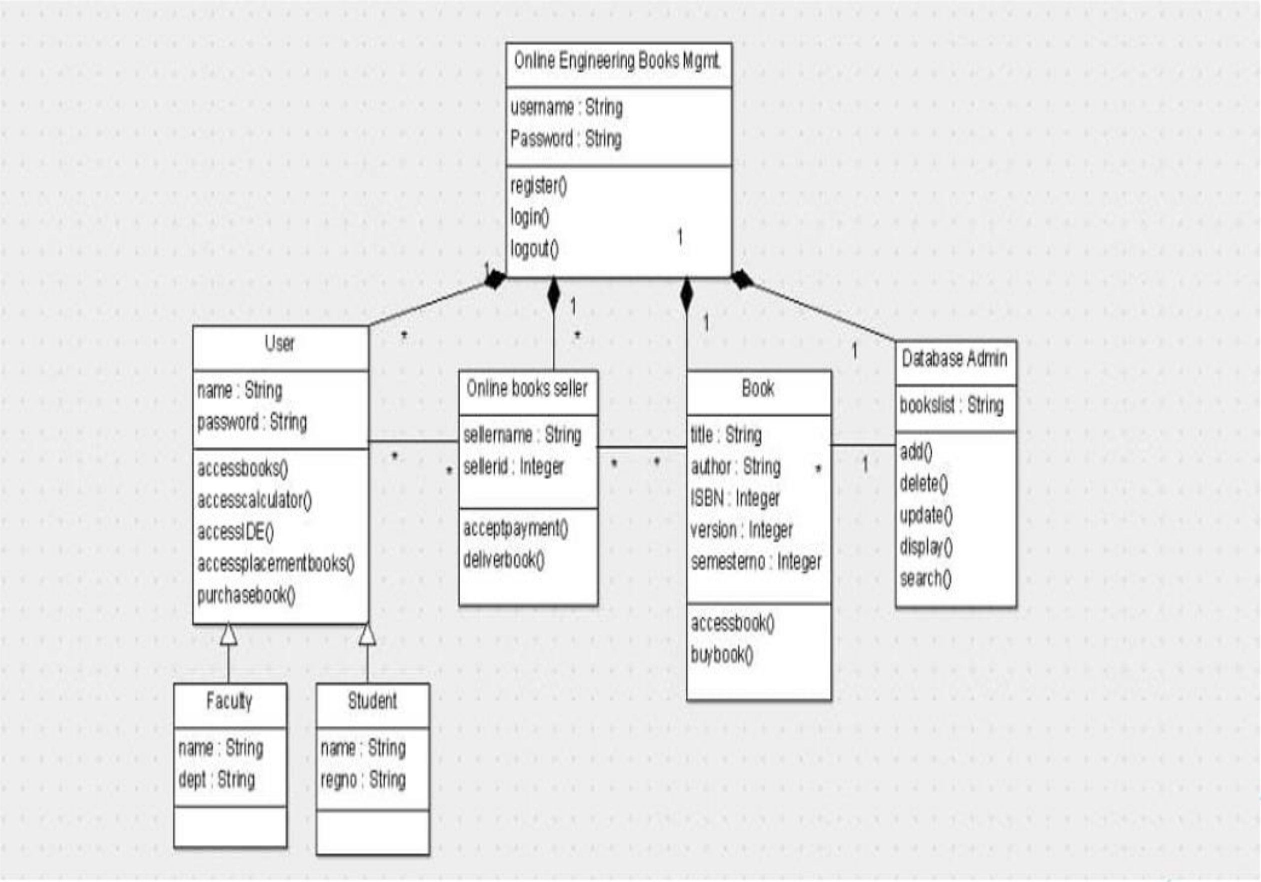
DATA FLOW DIAGRAM FOR USER


User
Search for books by Topic, Author Name, Book Name, Semester Number, Year Number.
If searching matches with the database
Requested Books will be Displayed.
If does not match
No Book Found!!!
After clicking in Video Section, Students/users may see the list of videos or video links
If matches the input
Video or Video links will be showed.
If does not match the input the message is:
No Video Found!!
Openlink will direct to website about any topic

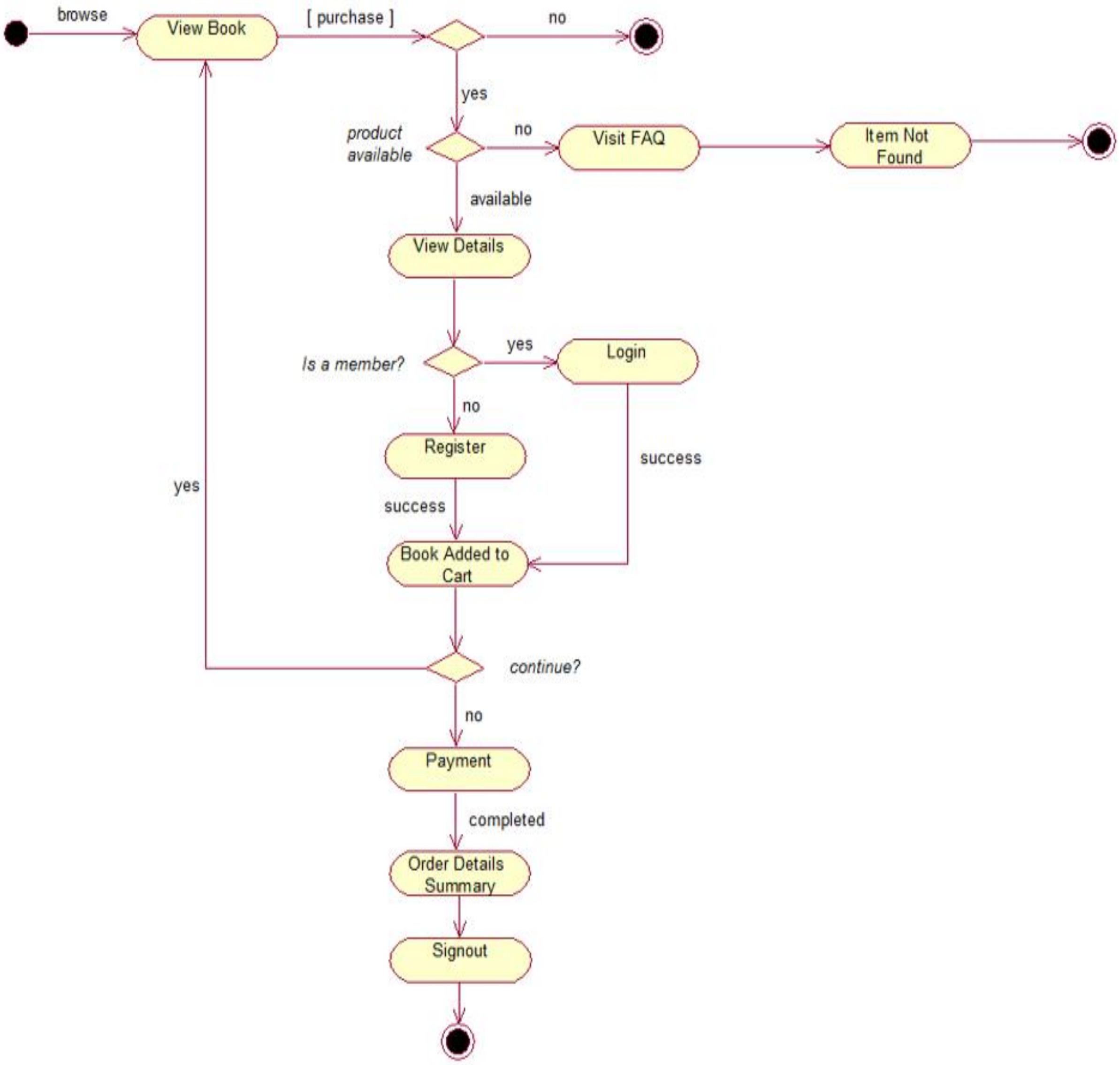
USE CASE DIAGRAM



CLASS DIAGRAM



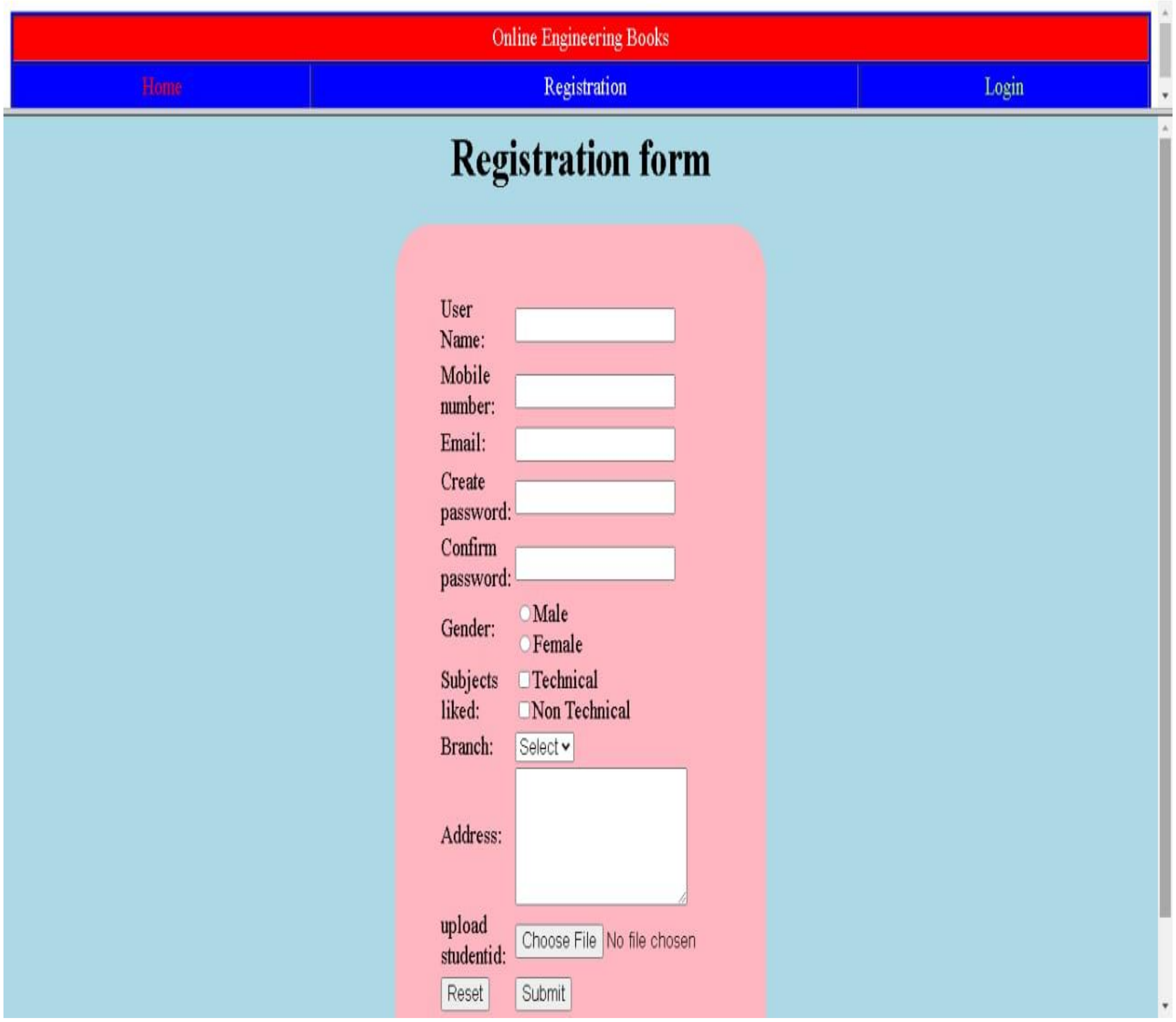
ACTIVITY DIAGRAM



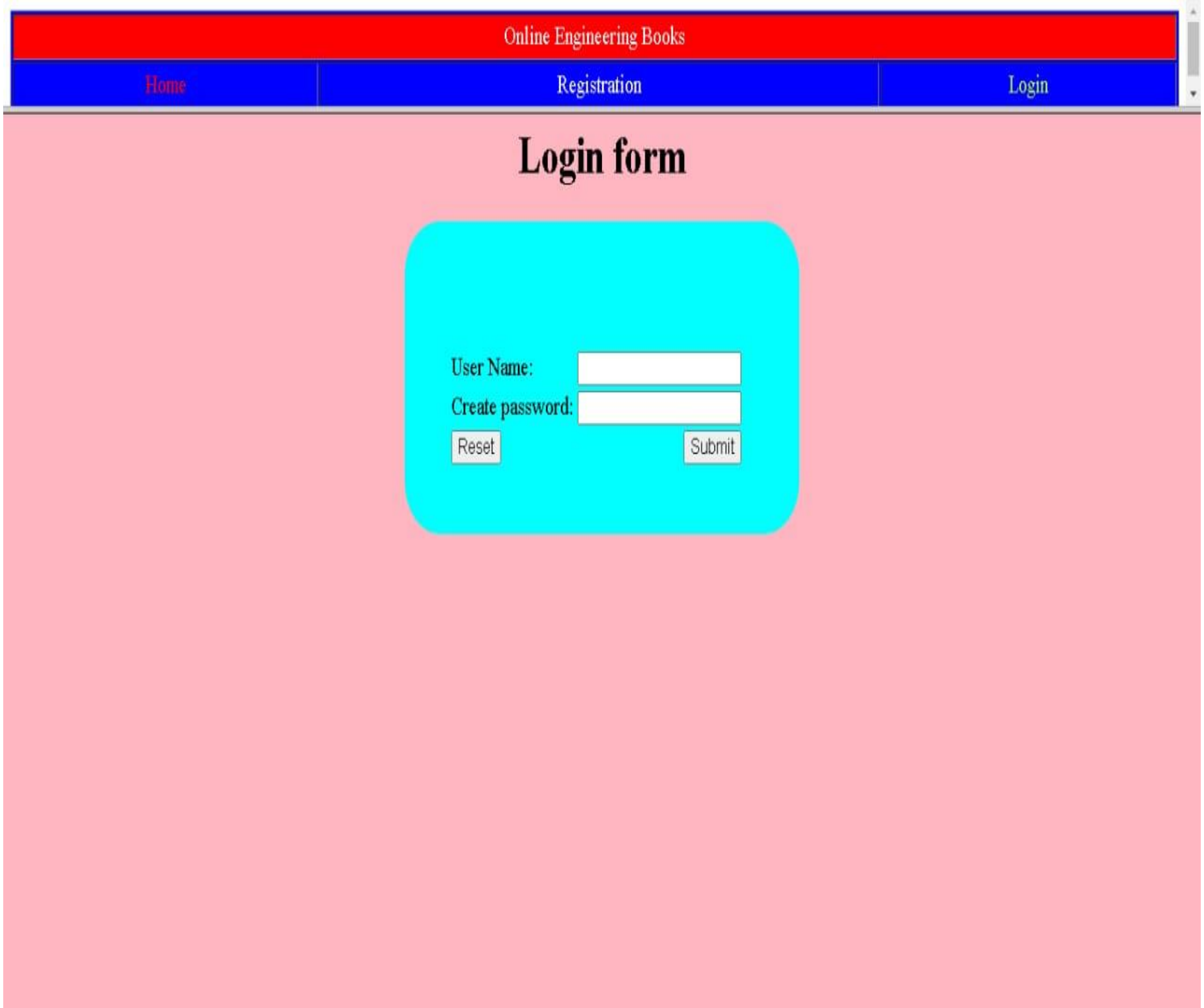
CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 Screenshot for registration form



4.2 Screenshot of login form



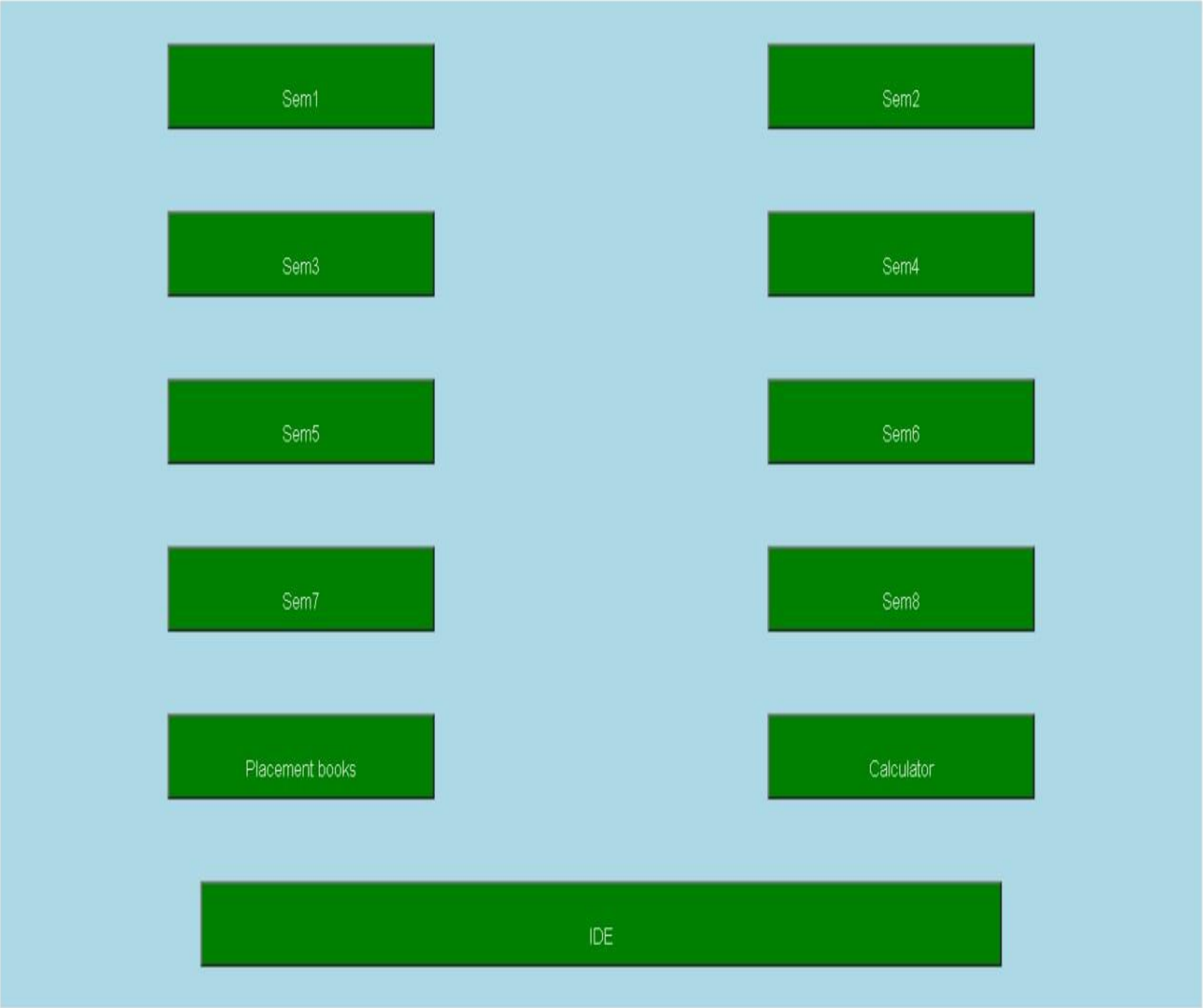
Online Engineering Books

Login form

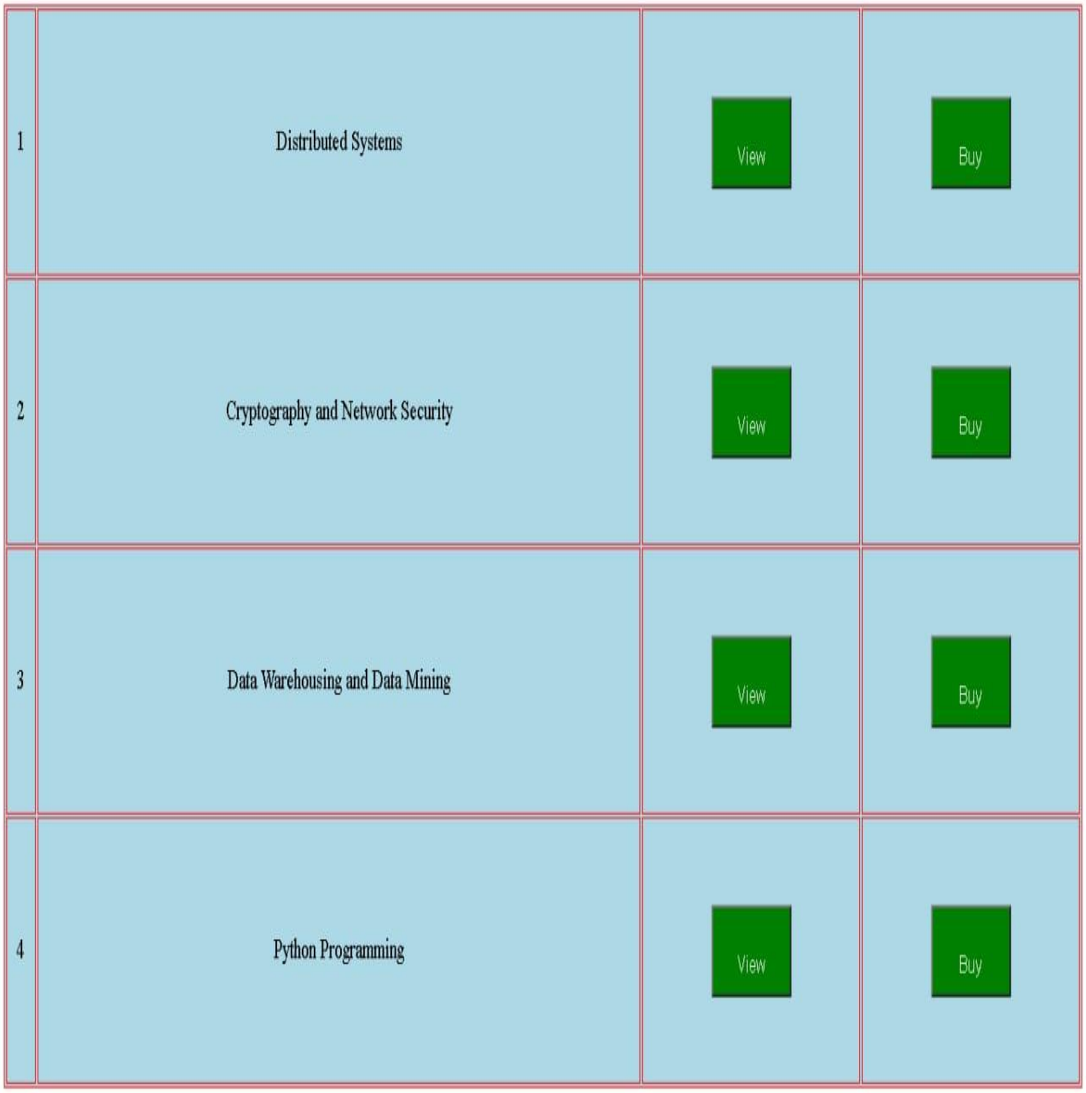
User Name:

Create password:

4.3 Screenshot of buttons for semwise, placement books,calculator and IDE

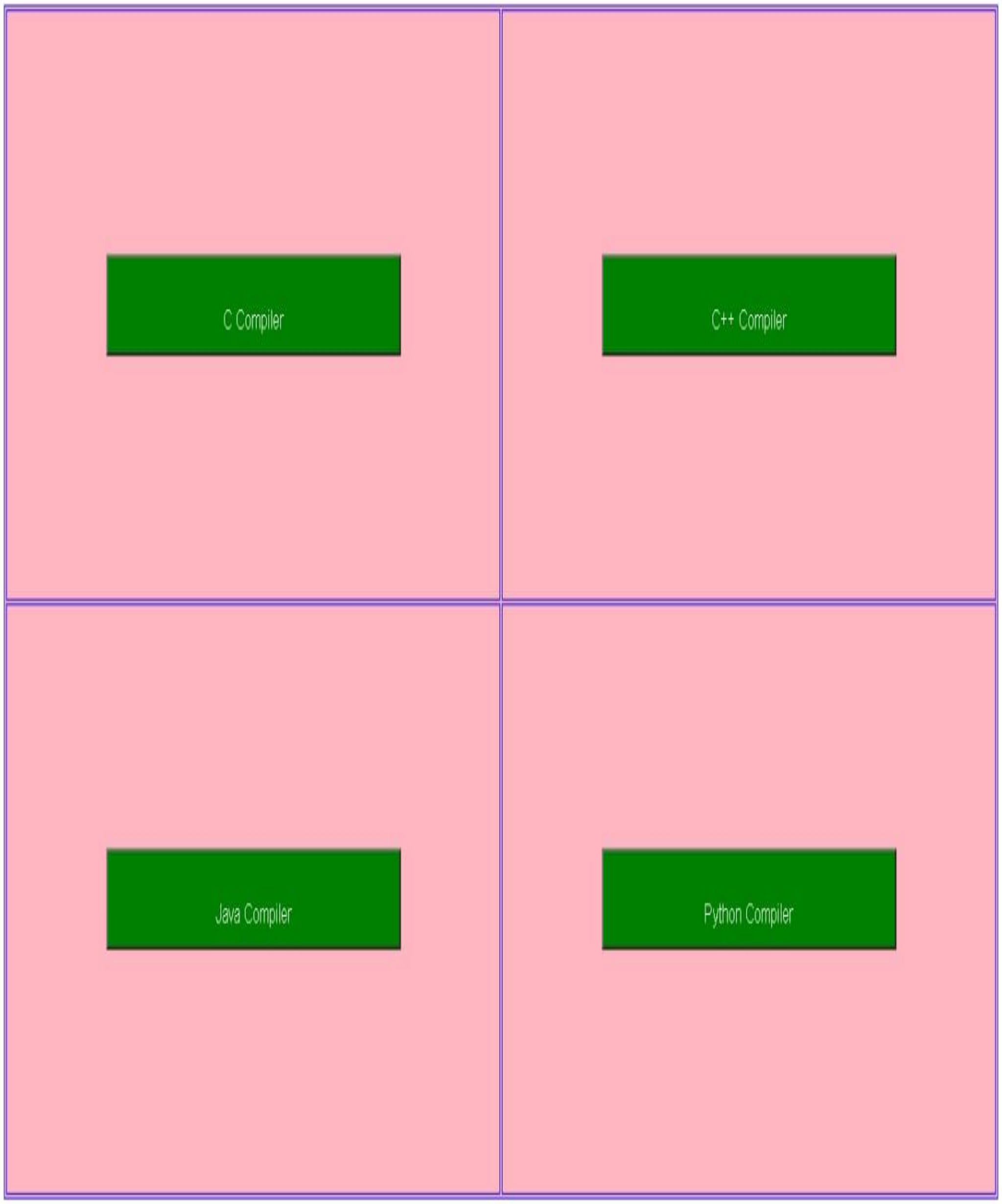


4.4 Screenshot of view and buy the text books

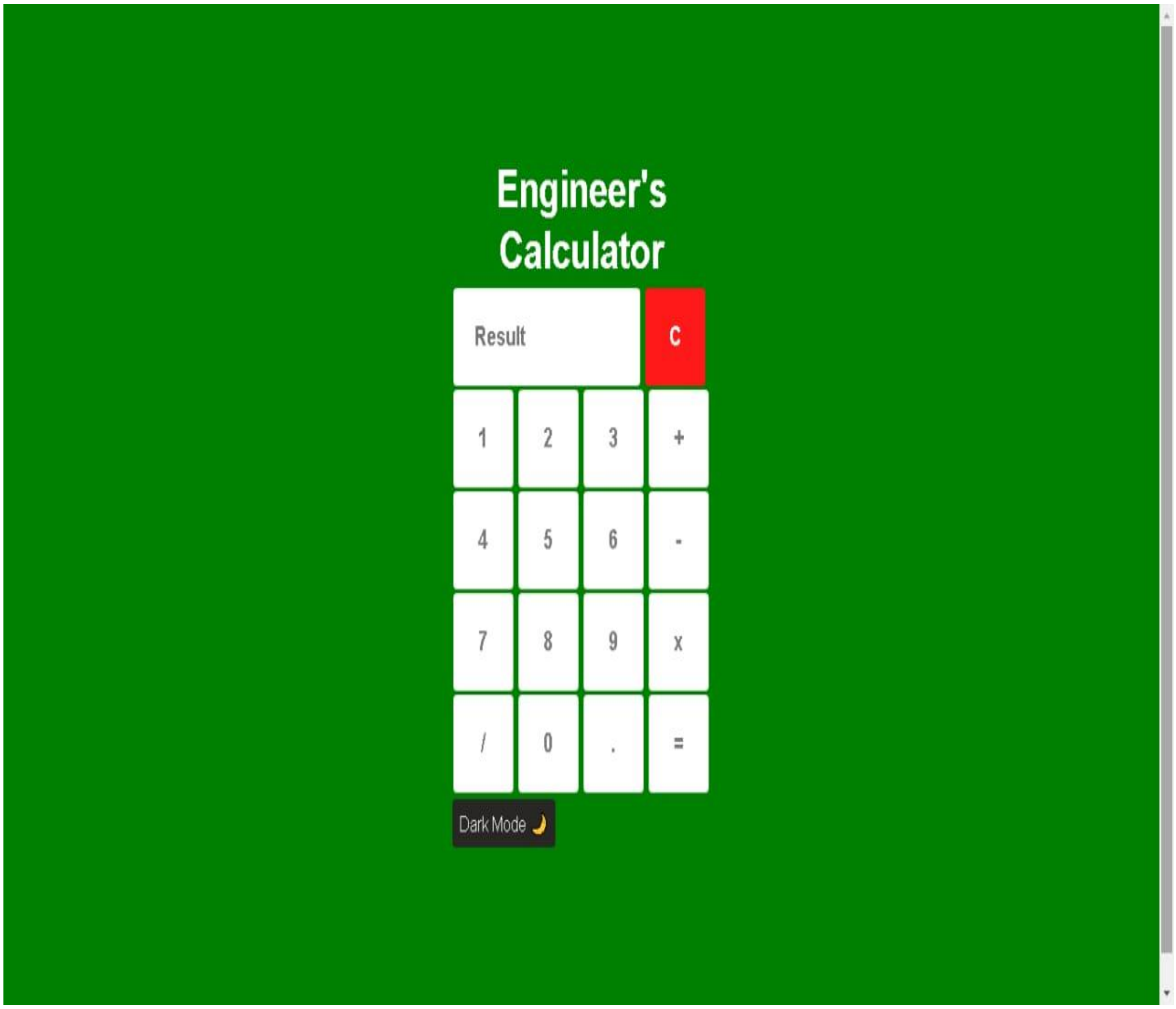


1	Distributed Systems	View	Buy
2	Cryptography and Network Security	View	Buy
3	Data Warehousing and Data Mining	View	Buy
4	Python Programming	View	Buy

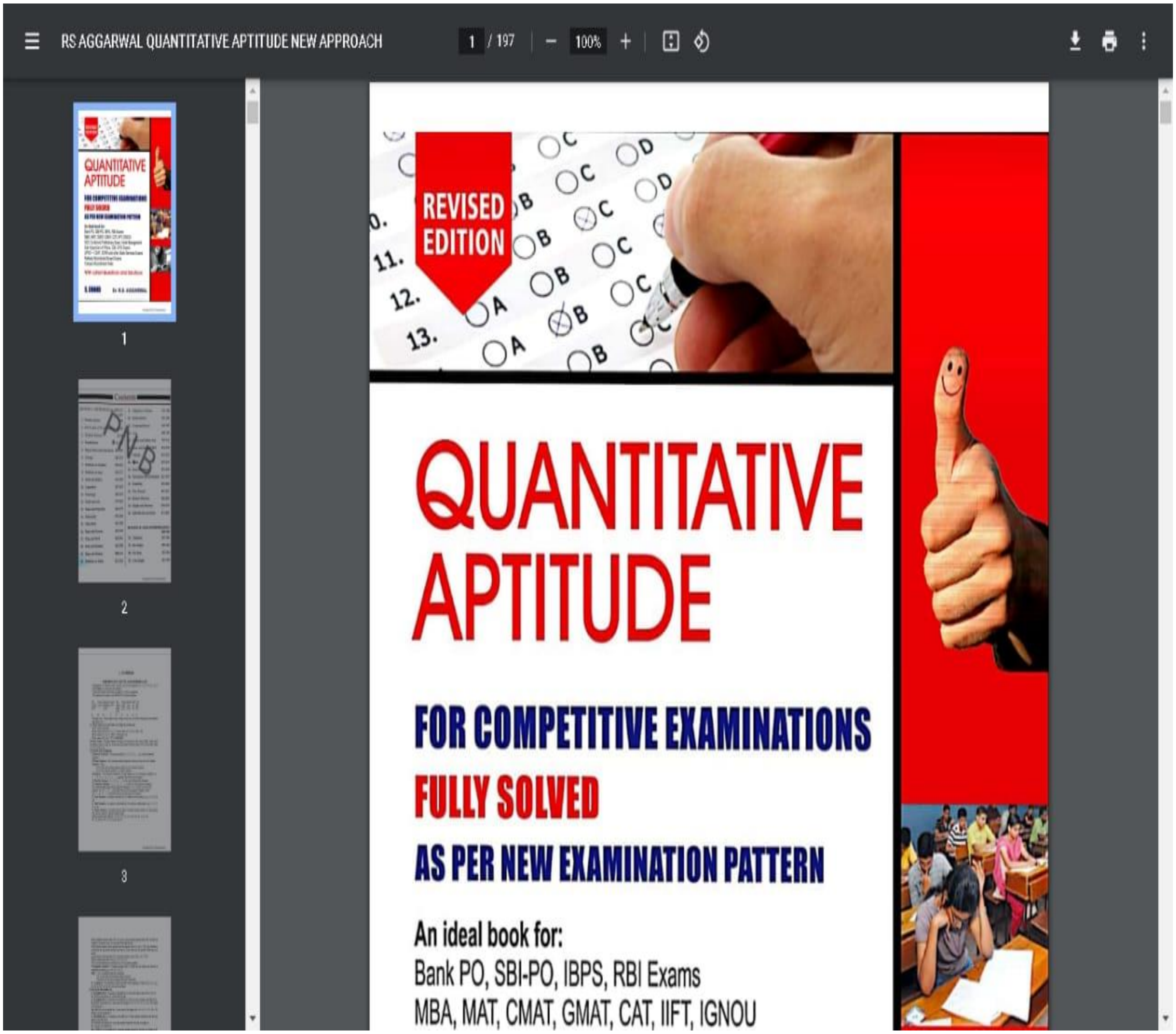
4.5 Screenshot of C,C++,Java and Python compilers



4.6 Screenshot of Engineer’s calculator



4.7 Screenshot of RS Aggarwal placement book



REVISED EDITION

QUANTITATIVE APTITUDE

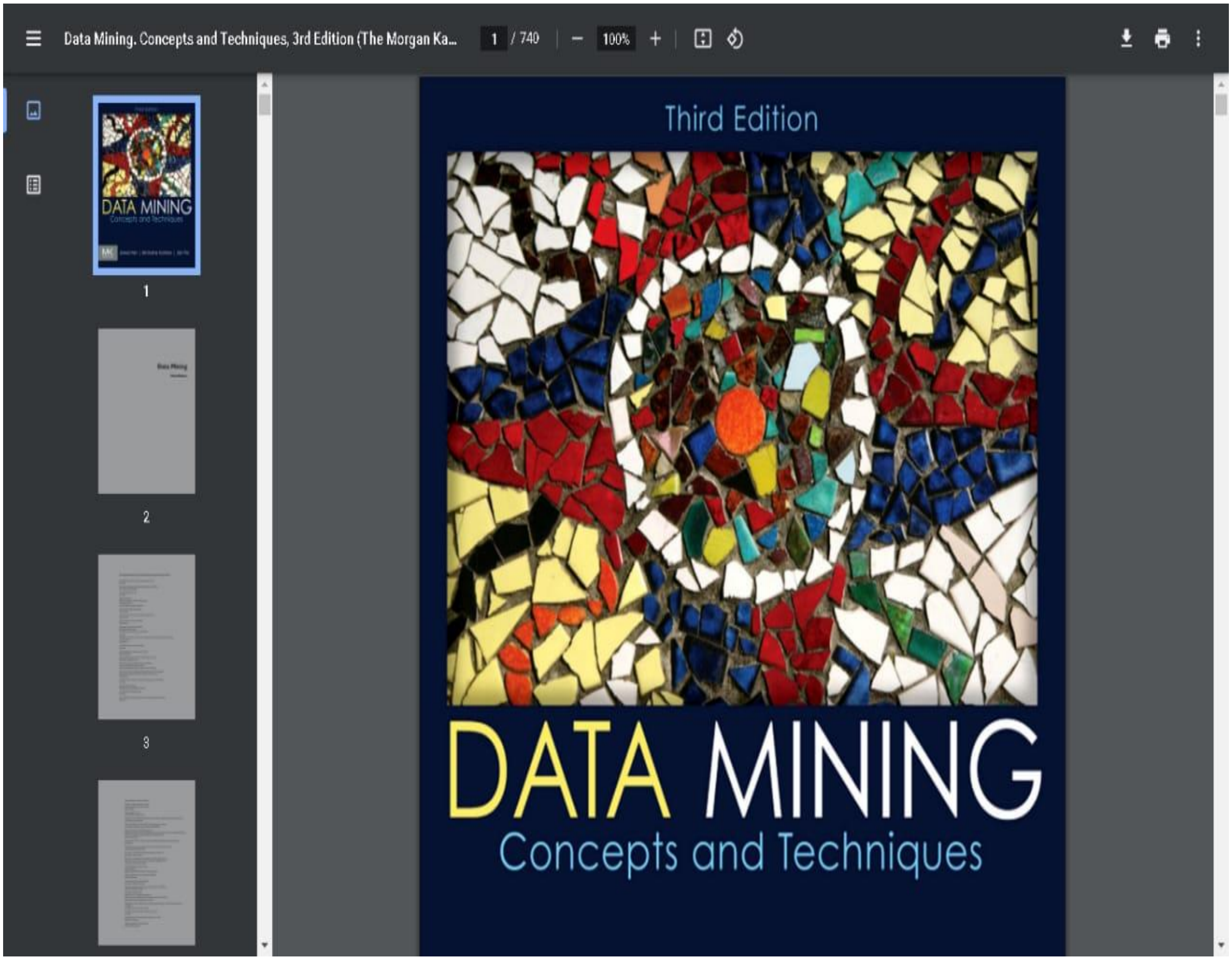
FOR COMPETITIVE EXAMINATIONS

FULLY SOLVED

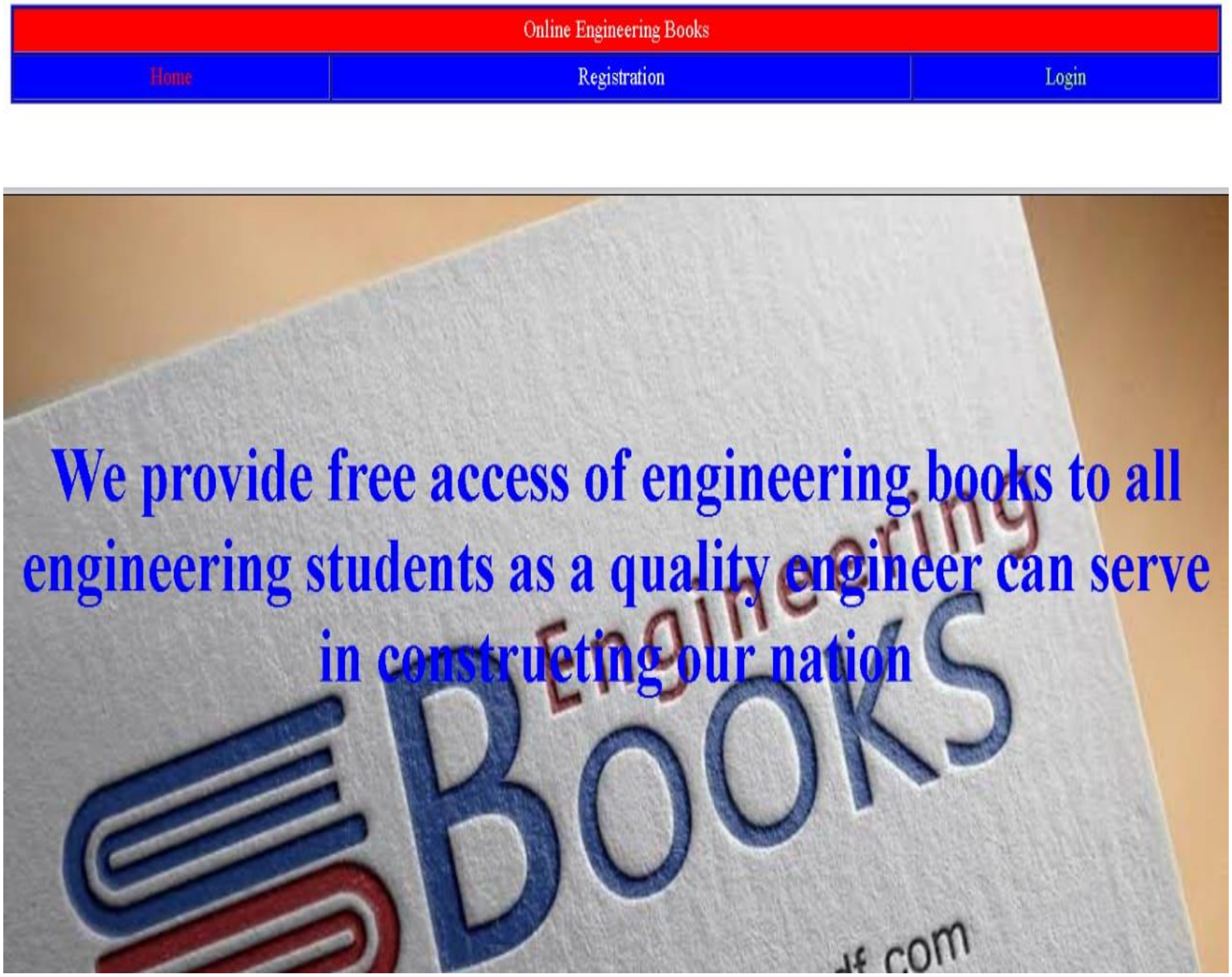
AS PER NEW EXAMINATION PATTERN

An ideal book for:
Bank PO, SBI-PO, IBPS, RBI Exams
MBA, MAT, CMAT, GMAT, CAT, IIFT, IGNOU

4.8 Screenshot of Data Mining Engineering Text Book


Data Mining. Concepts and Techniques, 3rd Edition (The Morgan Ka...
1 / 740
100%
1
2
3
Third Edition
DATA MINING
Concepts and Techniques

4.9 Screenshot of a book with quotation



We provide free access of engineering books to all engineering students as a quality engineer can serve in construeting our nation

CHAPTER 5

SYSTEM TESTING

The aim of the system testing process was to determine all defects in our project .The program was subjected to a set of test inputs and various observations were made and based on these observations it will be decided whether the program behaves as expected or not. Our Project went through two levels of testing

1.Unit testing

2.Integration testing

UNIT TESTING

Unit testing is undertaken when a module has been created and succesfully reviewed .In order to test a single module we need to provide a complete environment i.e besides the module we would require

- The procedures belonging to other modules that the module under test calls
- Non local data structures that module accesses
- A procedure to call the functions of the module under test with appropriate parameters

Unit testing was done on each and every module that is described under module description of chapter 4

1. Test For the admin module

- Testing admin login form-This form is used for log in of administrator of the system.In this we enter the username and password if both are correct administration page will open other wise if any of data is wrong it will get redirected back to the login page and again ask for username and password
- Student account addition- In this section the admin can verify student details from student academic info and then only add student details to main database it contains add and delete buttons if user click add button

data will be added to student database and if he clicks delete button the student data will be deleted.

- Book Addition- Admin can enter details of book and can add the details to the main book table also he can view the books requests .

2. Test for Student login module

- Test for Student login Form-This form is used for log in of Student .In this we enter username and password if all these are correct student login page will open other wise if any of data is wrong it will get redirected back to the login page and again ask for username and password.
- Test for account creation- This form is used for new account creation when student does not fill the form completely it asks again to fill the whole form when he fill the form fully it gets redirected to page which show waiting for conformation message as his data will be only added by administrator after verification.

3. Test for teacher login module-

- Test for teacher login form- This form is used for log in of teacher .In this we enter the username and password if all these are correct teacher login page will open other wise if any of data is wrong it will get redirected back to the login page and again ask for username and password.

INTEGRATION TESTING

In this type of testing we test various integration of the project module by providing the input

.The primary objective is to test the module interfaces in order to ensure that no errors are occurring when one module invokes the other module.

CHAPTER 6

CONCLUSION & FUTURE SCOPE

This website provides a computerized version of online engineering books management system which will benefit the students as well as the staff of the website.

It makes entire process online where student can search books, staff can generate reports and do book transactions. It also has a facility for student login where student can login and can see status of books issued as well request for book or give some suggestions. It has a facility of teacher's login where teachers can add lectures notes and also give necessary suggestion to library and also add info about workshops or events happening in our college or nearby college in the online notice board.

There is a future scope of this facility that many more features such as online lectures video tutorials can be added by teachers as well as online assignments submission facility , a feature Of group chat where students can discuss various issues of engineering can be added to this project thus making it more interactive more user friendly and project which fulfills each users need in the best way possible.

CHAPTER 7

REFERENCES

- http://www.w3schools.com/html/html_intro.asp

http://www.Udemy.com/css/css_background.asp

http://www.w3schools.com/js/js_datatypes.asp

CHAPTER 8

APPENDIX

A. 1.SOURCE CODE

Registration.html

```
<html>
  <head>
    <title>Registration form</title>
    <style>
    .reginform{border:5px ;
      display:flex;
      background-color:lightpink;
      border-radius:40px;
      margin-left:450px;
      margin-right:450px;
      padding:50px;
        padding-top:-10px;

    }
  </style>
    <script
     type="text/javascript" language="javascript">
      function valid() {
       var name = /^[a-zA-Z ]+$/;
        var numbers = /^[0-9]+$/;
        var pwd = /^[0-9a-zA-Z@]+$/;
        var email = /^\w+([\._]?\w+)*@\w+([\._]?\w+)*(\.\w{2,3})+$/;

        if (document.regform.name.value == "") {
          alert("Enter name");
          document.regform.name.focus();
  return false;
} else if (!document.regform.name.value.match(name)) {
  alert("Name should contain only alphabets");
  document.regform.name.focus();
  return false;
} else if (document.regform.mobno.value == "") {
  alert("Enter phone number");
  document.regform.mobno.focus();
  return false;
} else if (!document.regform.mobno.value.match(numbers)) {
  alert("Phone number should contain numbers only");
  document.regform.name.focus();
  return false;
} else if (document.regform.mobno.value.length < 10) {
  alert("Enter a valid Mobile Number");
  document.regform.mobno.focus();
  return false;
} else if (document.regform.email.value == "") {
  alert("Enter email");
  document.regform.email.focus();
  return false;
} else if (!document.regform.email.value.match(email)) {
  alert("mail id is invalid");
  document.regform.email.focus();
  return false;
} else if (document.regform.pwd.value == "") {
  alert("Enter password");
  document.regform.pwd.focus();
  return false;
} else if (document.regform.pwd.value.length < 6) {
  alert("password should be minimum 6 characters");
  document.regform.pwd.focus();
  return false;
} else if (!document.regform.pwd.value.match(pwd)) {
  alert("only numbers,alphabets and @ are allowed in password");
  document.regform.pwd.focus();
  return false;
} else if (document.regform.cpwd.value == "") {
  alert("Enter confirm password");
  document.regform.cpwd.focus();
  return false;
} else if (document.regform.cpwd.value.length < 6) {
  alert("Enter cpwd");
  document.regform.cpwd.focus();
  return false;
} else if (document.regform.pwd.value != document.regform.cpwd.value) {
  alert("Passwords not matched");
  document.regform.pwd.focus();
  return false;
} else if (
  !document.regform.gen[0].checked &&
  !document.regform.gen[1].checked
) {
  alert("Choose Gender");
  document.regform.gen[0].focus();
  return false;
} else if (
  !document.regform.subjects[0].checked &&
  !document.regform.subjects[1].checked
) {
  alert("Choose Subjects");
  document.regform.subjects[0].focus();
  return false;
} else if (document.regform.branch.selectedIndex == "0") {
  alert("Select a branch");
  document.regform.branch.focus();
  return false;
} else if (document.regform.address.value == "") {
```

```
          alert("Enter address");
          document.regform.address.focus();
          return false;
        } else if (document.regform.studentid.value == "") {
          alert("Upload student id");
          document.regform.studentid.focus();
          return false;
        } else {
          return true;
        }
      }
    </script>
  </head>
  <body style="background-color:lightblue">
    <h1 align="center">Registration form</h1>
    <form
      class="reginform" action="./Reg"
      onSubmit="return valid();"
      method="get"
      name="regform"
      align="center"
    >
      <table align=center>
      <tr>
      <td>
     User Name:
       </td>
       <td>
       <input type="text" name="name"></td>
</tr>      <br>
        <tr>
      <td>

      Mobile number:
```

```
  </td>
  <td><input type="number" name="mobno">
</td>
 </tr> <br>
 <tr><td>
Email:</td>
  <td><input type="text" name="email"></td>
  </tr>
<br>
  <tr><td>
Create password:</td><td><input type="password" name="pwd"></td></tr>
<br>
  <tr>
  <td>
Confirm password:</td><td><input type="password" name="cpwd">
  </td>
  </tr>
<br>
  <tr>
  <td>
Gender:</td><td> <input type="radio" name="gen" value="Male">Male
<br>
<input type="radio" name="gen" value="Female">Female</td></tr>
<br>
  <tr>
  <td>
Subjects liked:</td>
  <td>
<input type="checkbox" name="subjects" value="Technical">Technical
<br>
<input type="checkbox" name="subjects" value="Non Technical">Non
Technical</td>
</tr>
<br>
```

```
         <tr>
         <td>
     Branch:</td>
     <td> <select name="branch">
         <option value="0">Select</option>
         <option value="CSE">CSE</option>
         <option value="IT">IT</option>
         <option value="ECE">ECE</option>
         <option value="EEE">EEE</option>
         <option value="ME">ME</option>
         <option value="CE">CE</option>
      </select></td>
         </tr>
      <br>
         <tr>
         <td>
      Address:</td>
         <td><textarea rows="5" cols="20" name="address"> </textarea></td></tr>
      <br>
<tr>
<td>

      upload studentid:</td>
         <td><input type="file" name="studentid"></td>
         </tr>
      <br>
<tr>

     <td> <input type="reset"></td>
      <td><input type="submit" ></td></tr>
    </form>
  </body>
</html>
```

2.Login.html

```
<html >
<head>
    <title>Login form</title>
 <style>
   .loginform{border:5px ;
   background-color: aqua;
   border-radius:40px;
   margin-left:450px;
   margin-right:450px;
   padding:50px;
 }
 </style>
 <script type="text/javascript" language="javascript">
        function valid() {
  var name = /^[a-zA-Z ]+$/;
  var pwd = /^[0-9a-zA-Z@]+$/;
  if (document.logform.name.value == "") {
    alert("Enter name");
    document.logform.name.focus();
    return false;
  } else if (!document.logform.name.value.match(name)) {
    alert("Name should contain only alphabets");
    document.logform.name.focus();
    return false;
  }
   else if (document.logform.pwd.value == "") {
    alert("Enter password");
    document.logform.pwd.focus();
    return false;
  } else if (document.logform.pwd.value.length < 6) {
    alert("password should be minimum 6 characters");
    document.logform.pwd.focus();
    return false;
  } else if (!document.logform.pwd.value.match(pwd)) {
    alert("only numbers,alphabets and @ are allowed in password");
    document.logform.pwd.focus();
    return false;
  }
 else {
    return true;
  }
}
</script>
</head>
<body style="background-color:lightpink">
    <h1 align=center>Login form</h1>
    <form class="loginform" action="Log.class" onSubmit="return valid();"method="get" name="logform"
target="_blank" align=center>
            <table>
            <tr>
            <td>
     User Name:</td>
             <td><input type="text" name="name"></td></tr>
     <br>
             <tr>
             <td>
     Create password:
             </td><td><input type="password" name="pwd"></td></tr>

     <br>
             <tr><td>
     <input type="reset"></td>
             <td align=right>
     <input type="submit"></td></tr>
            </table>
            </form>
    </body>
</html>
```

3.Registration.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
import java.lang.*;
public class Reg extends HttpServlet
 {
    protected void service(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException
    {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Registration</title>");
            out.println("</head>");
            out.println("<body>");
            try
            {
                String name=request.getParameter("name");
                String mobno=request.getParameter("mobno");
                String email=request.getParameter("email");
                String pwd=request.getParameter("pwd");
                String cpwd=request.getParameter("cpwd");
                String gen=request.getParameter("gen");
                String sub="";
                String subjects[]=request.getParameterValues("subjects");
for(int i=0;i<subjects.length;i++)
{sub+=subjects[i]+" ";
}
                String branch=request.getParameter("branch");
                String address=request.getParameter("address");
                String studentid=request.getParameter("studentid");
                String driver="oracle.jdbc.driver.OracleDriver";
                String connurl="jdbc:oracle:thin:@localhost:1521:xe";
                String dbuname="cse";
                String dbpwd="cse";

                String sqltxt="insert into userdetails
values('"+name+"','"+mobno+"','"+email+"','"+pwd+"','"+cpwd+"','"+gen+"','"+sub+"','"+branch+"','"+address+"',
'"+studentid+"')";
                Class.forName(driver);

                Connection con = DriverManager.getConnection(connurl,dbuname,dbpwd);
                Statement stmt=con.createStatement();
                int i=stmt.executeUpdate(sqltxt);

               }
            catch(Exception e)
            {
                out.println(e);
            }
out.println("You are registered successfully");
            out.println("</body>");
            out.println("</html>");
    }
}
```

<u>4.Log.java</u>

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;
public class Log extends HttpServlet {

    protected void service(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

            out.println("<html>");
            out.println("<head>");
            out.println("<title>Login Check</title>");
            out.println("</head>");
            out.println("<body>");
            try
            {
                String name=request.getParameter("name");
                String pwd=request.getParameter("pwd");

                String driver="oracle.jdbc.driver.OracleDriver";
                String connurl="jdbc:oracle:thin:@localhost:1521:xe";
                String dbuname="cse";
                String dbpwd="cse";

                String sqltxt="select * from userdetails where name='"+name+"' and pwd='"+pwd+"'";
```

```
           Class.forName(driver);

           Connection con = DriverManager.getConnection(connurl,dbuname,dbpwd);
           Statement stmt=con.createStatement();
           ResultSet rs;

           rs=stmt.executeQuery(sqltxt);



           if( !(rs.next()) )
        out.println("Incorrect username and password");

           else
{
        out.println("Welcome   "+name);
RequestDispatcher rd=request.getRequestDispatcher("catcse.html");
rd.include(request,response);
}
        }
        catch(Exception e)
        {
           out.println(e);
        }
        out.println("</body>");
        out.println("</html>");
   }
}
```