

Book Bank Automation System

Boomija .M.D^[1], Sahaya Michel Rosetta .K^[2], Priya Dharshini .P^[3],
Sree Swarna Laximi .S^[4]

^[1]Assistant Professor - 1, Department of Information Technology, Prathyusha Engineering College
(Thiruvallur, Tamil Nadu, India), Anna University, India

^[2] ^[3] ^[4]Department of Information Technology, Prathyusha Engineering College (Thiruvallur, Tamil Nadu,
India), Anna University, India.

Abstract: In this project, the process of ordering books in a book bank by the student is automated. At present, this project is designed for the engineering colleges following Anna University syllabus, only for a few common branches of engineering such as IT, CSE, Civil, and Mech. The student has to first register into the system with the following details: Name, Password, DOB, College, Dept, Semester, Email, Phone number. On registration, the student will be provided with a unique student registration ID which is auto generated by the system. Using the name and password, the student can log into the system. In the student profile page, all the details of the students are displayed. From this page, the student can navigate to the request books page, where he can select his department the semester, and the books that he needs. Once he chooses the books and submits the request, the system will reserve the book and send him an automated email regarding the book chosen, and the date of issue. The administrator can log into the system, add new book details, edit the existing book details, and also edit the student details. When the student comes personally to receive the books, the admin updates the student profile that the student has received the books. This application was coded using Java Servlets, JSP, for the server and client side scripting, MySQL 2005 as the rear end, HTML as the front end.

I. Introduction

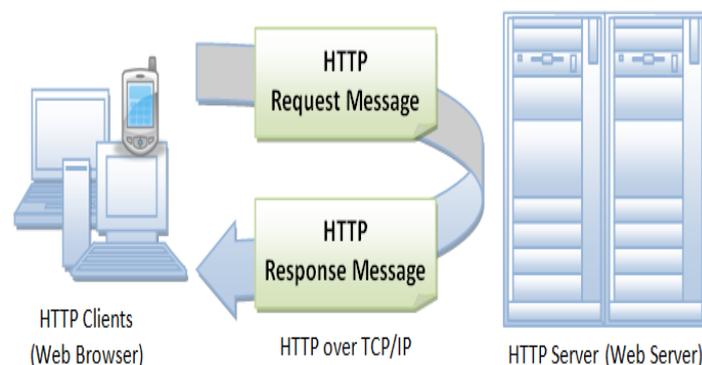
What is Web Application?

Web application or web app is an application program that is stored on a remote server and delivered over the internet. Web applications are accessed over a network connection using HTTP and run on a compatible browser. In some web applications, a small part of the program is stored in the user's desktop, but processing is done over the internet on external server. The main concepts involved are:

- HTTP
- Java Server Pages

HTTP:

The Hypertext Transfer Protocol (**HTTP**) is an application protocol for distributed, collaborative, and hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web. Hypertext is structured text that uses logical links (hyperlinks) between nodes containing text. HTTP functions as a request-response protocol in the client-server computing model. A web browser, for example, may be the *client* and an application running on a computer hosting a website may be the *server*. The client submits an HTTP *request* message to the server. The server which provides *resources* such as HTML files and other content, or performs other functions on behalf of the client, returns a *response* message to the client. The response contains completion status information about the request and may also contain requested content in its message body.

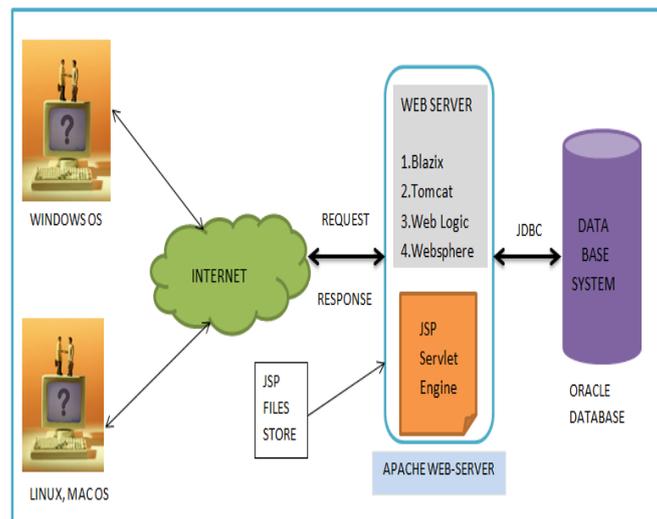


Java Server Pages:

JavaServer Pages (JSP) is a technology that helps software developers create dynamically generated web pages based on HTML, XML, or other document types. JSP is similar to PHP, ASP and React's JSX but it uses the Java programming language. To deploy and run JavaServer Pages, a compatible web server with a servlet container, such as Apache Tomcat is required.

JSP is a high-level abstraction of Java servlets. JSPs are translated into servlets at runtime, therefore JSP is a Servlet; each JSP servlet is cached and re-used until the original JSP is modified. JSP can be used independently or as the view component of a server-side model–view–controller design, with JavaBeans as the model and Java servlets as the controller.

JSP allows Java code and certain pre-defined actions to be interleaved with static web markup content, such as HTML, with the resulting page being compiled and executed on the server to deliver a document. The compiled pages, as well as any dependent Java libraries, contain Java bytecode rather than machine code. Like any other Java program, they must be executed within a Java virtual machine (JVM) that interacts with the server's host operating system to provide an abstract, platform-neutral environment.



JSPs are usually used to deliver HTML and XML documents, but through the use of OutputStream, they can deliver other types of data as well. The Web container creates JSP implicit objects like request, response, session, application, config, page, pageContext, out and exception. JSP Engine creates these objects during translation phase.

Objective And Scope

The main objective of this project is to automate the process of registering and ordering books in a book bank by the student. At present, this project is designed for a few common branches (IT, CSE, Mech, and Civil) of the engineering colleges following Anna University syllabus. At present, all the tasks in a book bank are done manually, involving a lot of paper work. The student has to visit book bank for the process of registration and also request for books. Also, it is also difficult for the staff to find information of a particular student since the data is stored in an Excel spreadsheet. In this Book Bank Automation System the process of registration and ordering books is automated. The students can register online and also view their profile online. The main purpose of developing this book bank automation system is to design an application, which could store student and book data and provide an interface for registering, ordering books online, retrieving student and book related details with 100% accuracy This book bank automation system also allows admin to add and modify the details of a new book, delete account. Using this system admin can also search any individual profile in few seconds. The system will check the user's existence in the database and provide the set of services.

II. System Analysis

Existing System:

The present system is only some book banks are partially automated, that is by storing details in MS Excel Spreadsheets, whereas many other book banks are manual. A student has to visit the book bank personally at least thrice to get a set of the books that are needed. Also only the student has to give the list of books needed along with the exact details of the ordered books, such as author's name, edition, name of the publication, etc. Only the list of students registered and the books available are stored in the database. Due to this the manual

process and work involved in the existing system is more. A student has to visit the book bank personally to get registered, to request and also to receive books, and this is time consuming. The book bank works only for a specific time within which the student should visit and complete all the procedures. The staff will have to manually search for the details of any particular student and the list of books ordered by the student.

Proposed System:

In our proposed system the student can register online in the system, enter his/her personal details, and can fill the registration form with unique Student Identifier. The student can complete all the procedures such as ordering books and confirming the booklist online and go to the book bank only for receiving the books. The system confirms the book list by means of automated email service. The books that would be needed by the student are predicted by the system according to the department and semester chosen in the request books page. Hence the student is given a set of choices for selecting a book.

Due to this all the processes such as registering, requesting books, etc can be done online and automated. It saves a lot of time for the students and the book bank staff as well. There is no specific time for the student to register into the system and download the registration form. The student need not give the list of books needed, and can send only a confirmation for the same.

Working Environment

Hardware Requirements

All the hard physical materials used in any system are called hardware. These are the devices, which are made to perform specific functions and can do only that works as instructed. They can be both electronic devices and mechanical systems. The hardware requirements for this web application are as follows:

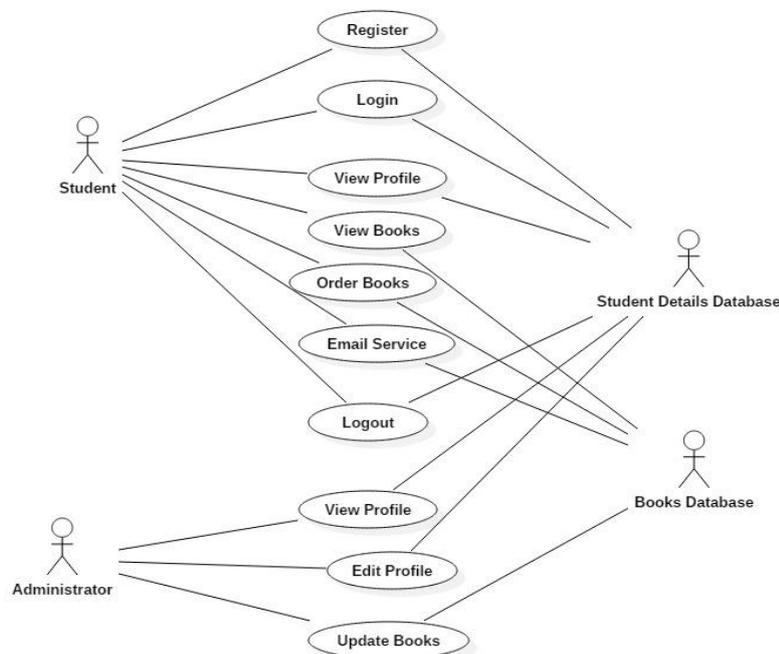
- Processor : Pentium Dual Core 2.3 GHz
- Hard Disk : 250 GB or Higher
- Ram : 1 GB (Min)

Software Requirements

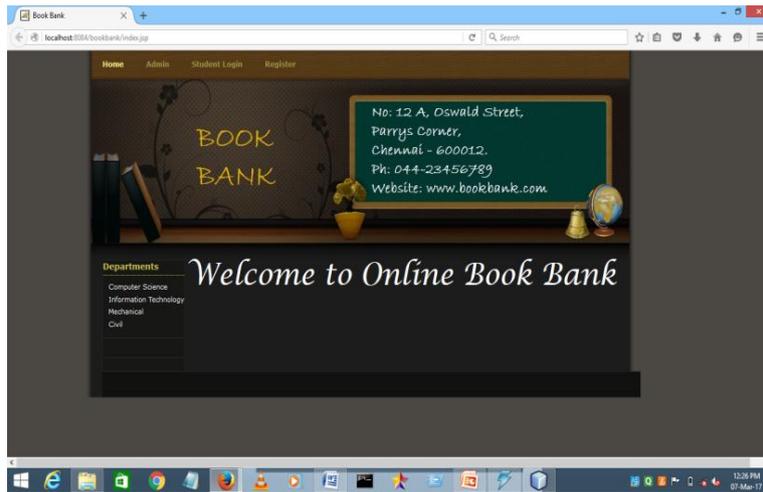
Software is a set of instructions that are used to command any system to perform any operation. Software has advantage to make decisions to deliver sensible results and in handling complex situations. The software requirements for this web application are as follows:

- Operating System : Windows XP or Higher
- Languages used : Java (JSP, Servlet), HTML
- Tools : JDK 1.7, Net Beans 7.3.1
- Backend : MySQL 5.5.2

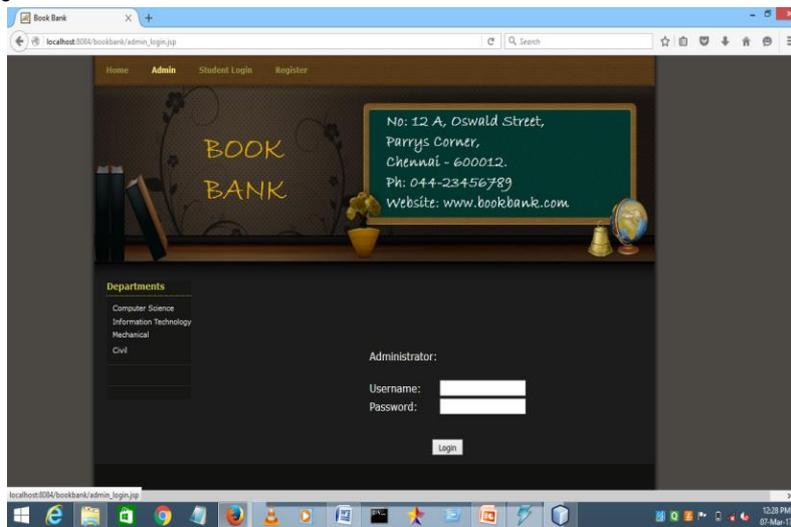
Usecase Diagram



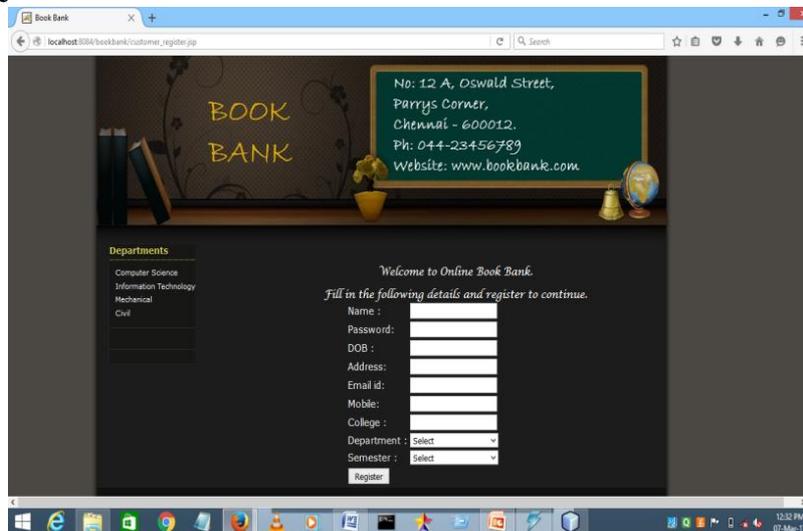
System Implementation Home Page



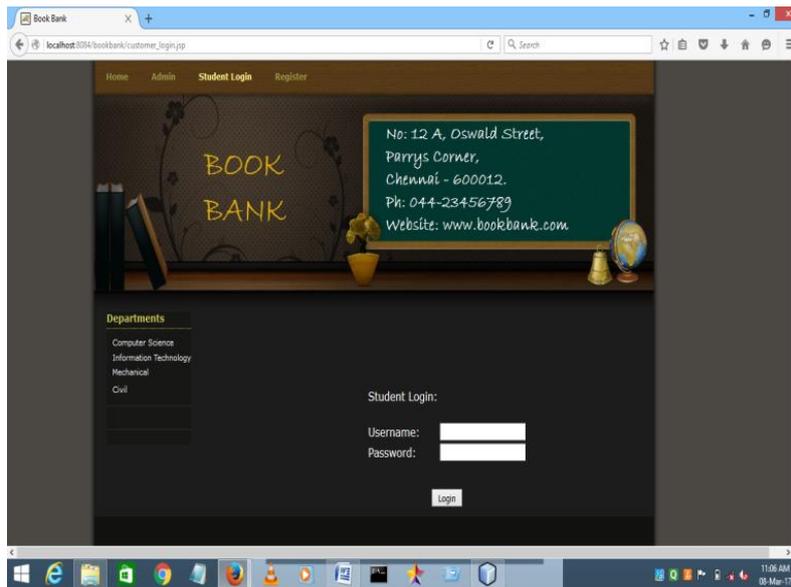
Admin Login page



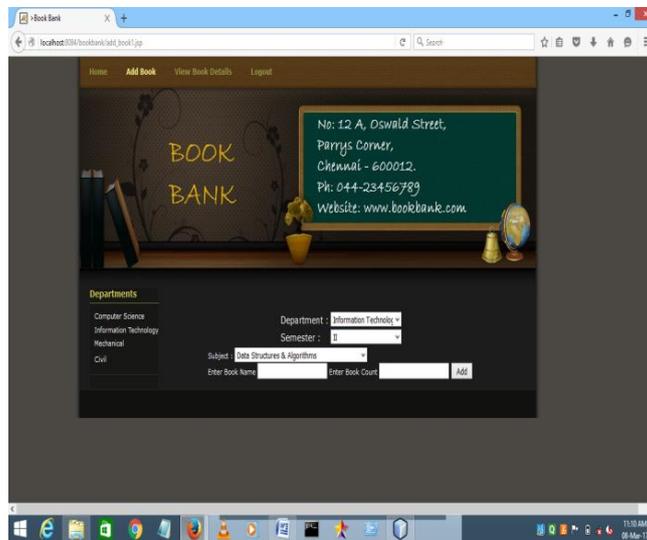
Registration Page



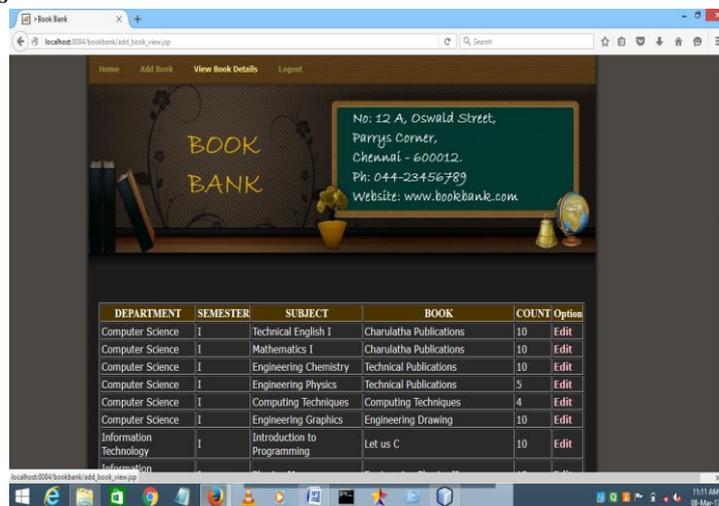
Student Login



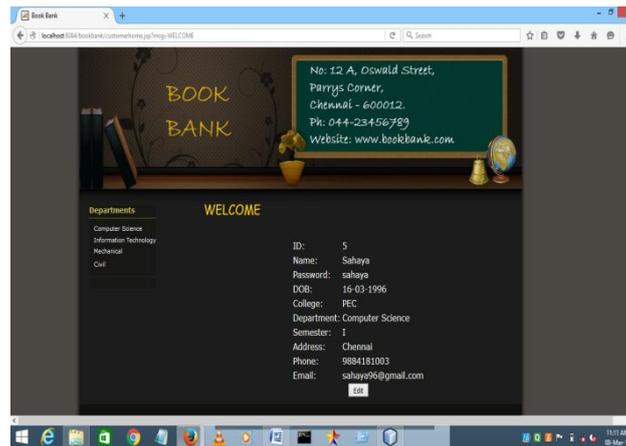
Add Book page



Viewing Book Details



Student Profile



References

- [1]. "Full-scale Online Event Ticketing System The Design and Implementation", a project at University of Macau, March 2003
- [2]. "A Project On Online Bus Booking System", a project at Ganpat University, March 2004
- [3]. "Online Book-Bank Monitoring System" a project at Makerere University, Nov 2005
- [4]. "The design and implementation of an e-commerce site for online book sales", a project at Indiana University South Bend, May 2007
- [5]. "e-VEHICLE BOOKING SYSTEM AT UMP" a project at University Malaysia Pahang , May 2011
- [6]. "Online Library Management System", IOSR Journal of Engineering, Feb 2012
- [7]. "Dependable online appointment booking system for NHIS outpatient in Nigerian teaching hospitals", International Journal of Computer Science & Information Technology (IJCSIT) Vol 6, No 4, August 2014