**VISVESVARAYA TECHNOLOGICAL UNIVERSITY**

"Jnana Sangama", Belagavi-590 018



A Database Management System Mini Project Report on

**CSE Book Store**

Submitted in partial fulfilment of the requirement for the award of the Degree of

**Bachelor of Engineering**

In

**computer Science & Engineering**

by

**Preetham Gouda S Patil 1AY19CS079**

**Pradeep N 1AY19CS072**

Under the guidance of

**Prof. Ancy Thomas**

**Prof. Rashmi Kulkarni,**

**Prof. Chidananda T**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**ACHARYA INSTITUTE OF TECHNOLOGY**

**(AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI)**

**2021-22**

ACHARYA INSTITUTE OF TECHNOLOGY

**(AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI)**

Soldevanahalli, Bengaluru-560107



**Certificate**

Certified that the Database Management System mini project work entitled **“CSE Book Store”** is a bonafide work carried out by **Preetham Gouda S Patil(1AY19CS079) and Pradeep N(1AY19CS072)**  in partial fulfilment for the award of the degree of **Bachelor of Engineering** in **Computer Science and Engineering** of the **Visvesvaraya Technological University,** **Belagavi**, during the year 2021-22. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The Project has been approved as it satisfies the academic requirements in respect of Project work prescribed for the **Bachelor of Engineering Degree.**

**Signature of Guides**  **Signature of H.O.D**

**Name of the Examiners Signature with date**



**ACKNOWLEDGEMENT**

The satisfaction that accompanies the successful completion of this mini-project would be incomplete without the mention of the people who made it possible through constant guidance and encouragement.

We would take this opportunity to express our heart-felt gratitude to **Sri. B. Premnath Reddy**, Chairman, Acharya Institutes and **Dr. Prakash M R**, Principal, Acharya Institute of Technology for providing the necessary infrastructure to complete this mini-project.

We wish to express our deepest gratitude and thanks to **Dr**. **Ajith Patadiya**, Head of the Department, Computer Science and Engineering.

We wish to express sincere thanks to my guide **Prof. Ancy Thomas**, and Assistant Professors, Department of Computer Science and Engineering for helping us throughout and guiding us from time to time.

A warm thanks to all the faculty of Department of Computer Science and Engineering, who have helped us with their views and encouraging ideas.

i

# ABSTRACT

The main objective of the Online CSE Book Store is to create an online book store that allows users to purchase a book online based on title and publisher. The selected books are displayed in a tabular format and the user can order their books online. Using this website, the user can purchase a book online instead of going out to a book store and wasting time.

Online CSE Book Store: Can lead to error free, secure, reliable and fast management system. The user can select the required books. They should give the details of their name, contact number and shipping address. The user can also request for the required book by writing message. The books are divided into many categories based on publishers available in market.

**TABLE OF CONTENTS**

Acknowledgement (i)

Abstract (ii)

Table of contents (iii)

List of figures (vi)

## 1 Introduction ...................................................................................... .......1

1.1 Introduction to database management system .............................................. .....1

1.2 History of DBMS ............................................................................................... 2

1.3 Applications of DBMS ...................................................................................... 3

1.4 Overview of the project…………………………....................……..…………3

1.5 Theory and Concepts ......................................................................................... 4

1.6 xampp server ...................................................................................................... 4

### 2 System Requirements Specification .................................................................... 5

2.1 Functional Requirements ................................................................................... 5

2.2 Non Functional Requirements ........................................................................... 5

2.2.1 Hardware Requirements ..................................................................... 5

2.2.2 Software Requirements ...................................................................... 6

2.3 About Technologies Used .................................................................................. 6

### 3 System Design ........................................................................................................ 8

3.1 Input Design ....................................................................................................... 8

3.1.1 Admin ................................................................................................. 8

3.1.2 User ..................................................................................................... 8

3.2 Database design ................................................................................................. 8

3.3 Relational schema .............................................................................................. 9

3.4 ER Diagram ..................................................................................................... 10

3.5 Output design ................................................................................................... 12

### 4 System Implementation ...................................................................................... 13

4.1 Implementation ................................................................................................ 13

4.2 Create connection to database .......................................................................... 13

4.3 Closing a connection ........................................................................................ 13

4.4 Triggers and stored Procedures ........................................................................ 13

4.4.1 Stored Procedure ............................................................................... 14

4.4.2 Trigger............................................................................................... 14

4.5 Pseudocode ...................................................................................................... 15

4.15 Result ............................................................................................................. 23

### 5 Snapshots .............................................................................................................. 24

5.1 Home page..................................................................................................24

5.2 Admin Login Page.....................................................................................24

5.3 Admin’s Home Page..................................................................................25

5.4 Schedules details........................................................................................25

5.5 Schedules details........................................................................................26

5.6 Rankings.....................................................................................................26

5.7 Stadium Details..........................................................................................27

5.8 Cricket boards information.........................................................................27

5.9 Player search..............................................................................................28

5.10 Add player..................................................................................................28

5.11 Add schedule..............................................................................................29

5.12 Add Stadium...............................................................................................29

5.13 Delete player..............................................................................................30

5.14 Delete schedule..........................................................................................30

5.15 Delete Stadium...........................................................................................31

5.16 User login page...........................................................................................31

5.17 Create an account page...............................................................................32

5.18 Home page for users...................................................................................32

5.19 Schedules....................................................................................................33

5.20 Schedules details........................................................................................33

5.21 Rankings.....................................................................................................34

5.22 Stadium details...........................................................................................34

5.23 Cricket boards............................................................................................35

5.24 Player search for users and admin..............................................................35 5.25 About the IPL for users and admin............................................................36 **6 Conclusion and Future Enhancements............................................................37**

6.1 Conclusion..................................................................................................37

6.2 Future Enhancements.................................................................................37

**...Bibiliography......................................................................................................38**

## LIST OF FIGURES

Figure 3.1: Relational schema ……………………………………………………….......……..09

Figure 3.2: ER-Diagram. …………………………………………………………………........11

**CHAPTER 1**

# INTRODUCTION

## 1.1 INTRODUCTION TO DATABASE MANAGEMENT SYSTEM

Databases and database technology have a major impact on the growing use of computers. It is fair to say that databases play a critical role in almost all areas where computers are used, including business, electronic commerce, engineering, medicine, genetics, law, education, and library science. The word database is so commonly used that User must begin by defining what a database is. Our initial definition is quite general. A database is a collection of related data.1 By data, User mean known facts that can be recorded and that have implicit meaning. For example, consider the names, telephone numbers, and addresses of the people you know. You may have recorded this data in an indexed address book or you may have stored it on a hard drive, using a personal computer and software such as Microsoft Access or Excel. This collection of related data with an implicit meaning is a database. The preceding definition of database is quite general; for example, User may consider the collection of words that make up this page of text to be related data and hence to constitute a database. However, the common use of the term database is usually more restricted.

A database has the following implicit properties:

* A database represents some aspect of the real world, sometimes called the mini world or the universe of discourse. Changes to the in world are reflected in the database.
* A database is a logically coherent collection of data with some inherent meaning. A random assortment of data cannot correctly be referred to as a database
* A database is designed, built, and populated with data for a specific purpose. It has an intended. group of users and some preconceived applications in which these users are interested.

A database management system (DBMS) is a collection of programs that enables users to create and maintain a database. The DBMS is a general-purpose software system that facilitates the processes of defining, constructing, manipulating, and sharing databases among various users and applications. Defining a database involves specifying the data

types, structures, and constraints of the data to be stored in the database. The database definition or descriptive information is also stored by the DBMS in the form of a database catalogue or dictionary; it is called meta-data. Constructing the database is the process of storing the data on some storage medium that is controlled by the DBMS. Manipulating a database includes functions such as querying the database to retrieve specific data, updating the database to reflect changes in the in world, and generating reports from the data. Sharing database allows multiple users and programs to access the database simultaneously.

## 1.2 APPLICATIONS OF DBMS

Applications where we use Database Management Systems are:

**Telecom**: There is a database to keeps track of the information regarding calls made, network usage, customer details etc. Without the database systems it is hard to maintain that huge amount of data that keeps updating every millisecond.

**Industry**: Where it is a manufacturing unit, warehouse or distribution Centre, each one needs a database to keep the records of ins and outs. For example, distribution Centre should keep a track of the product units that supplied into the Centre as well as the products that got delivered out from the distribution Centre on each day; this is where DBMS comes into picture.

**Banking System**: For storing customer info, tracking day to day credit and debit transactions, generating bank statements etc. All this work has been done with the help of Database management systems.

**Education sector**: Database systems are frequently used in schools and colleges to store and retrieve the data regarding student details, staff details, course details, exam details, payroll data, attendance details, fees detail etc. There is a hell lot amount of inter-related data that needs to be stored and retrieved in an efficient manner.

**Online shopping**: You must be aware of the online shopping websites such as Amazon, Flipkart etc. These sites store the product information, your addresses and preferences, credit details and provide you the relevant list of products based on your query. All this involves a Database management system.

## 1.3 Overview of the project

Online CSE Book Store is a user-friendly Application which is based on HTML and CSS which helps students to find the various types of books related to CSE department and also allows to manage the records of various books purchased. The application uses HTML and CSS as a front end for interacting with the user and PHP for connection. At backend we used MySQL for database.

## 1.4 Theory and Concepts

**Inheritance**: In object-oriented programming, inheritance is when an object or class is based on another object or class, using the same implementation (inheriting from an object or class) or specifying a new implementation to maintain the same behavior (realizing an interface). Such an inherited class is called a subclass of its parent class or super class.

**Encapsulation**: In object-oriented programming, encapsulation is a mechanism of binding the data, and the functions together in a class and use them by creating an object of that class.

**Data Abstraction**: Data abstraction refers to, providing only essential information to the outside world and hiding their background details, i.e., to represent the needed information in program without presenting the implementation details. Data abstraction is a programming (and design) technique that relies on the separation of interface and implementation.

## 1.5 Xampp server

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, Maria DB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible. It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well. XAMPP's ease of deployment means a WAMP or LAMP stack can be installed quickly and simply on an operating system by a developer, with the advantage that common add-in applications such as Word Press and Joomla! can also be installed with similar ease using Bitnami. Though it is a heavy app for most of the operating systems even

when owing to its less size it takes a load on the processor speed. MySQL was replaced with MariaDB on 19 October 2015 and beginning with XAMPP versions 5.5.30 and 5.6.14,effectively altering the meaning of the acronym. While both letters P are de facto interchangeable, convention used at the Apache Friends website indicates that the first letter P is short for PHP and the latter letter P is short for Perl.

**Chapter 2**

# System Requirements Specification

## 2.1 Functional Requirements

The specific functional requirements of the Online CSE book store are stated as follows: **1.User**

* User can be able to view description about books.
* User can be able to search books and view there information
* Users can be able to search books published by particular publisher and access its information
* User can be able to access and view the information regarding price and publisher

**2. Admin**

* Admin can be able to login to a system
* Admin can be able to to add books to database
* Admin can be able to to delete books from the database
* Admin can be able to to add books under different publishers
* Admin can be able to to edit the information related to books
* Admin can be able to search books and retrive their information

## 2.2 Non Functional Requirements

### 2.2.1 Hardware Requirement

The section of hardware configuration is an important task related to the software development insufficient random-access memory may affect adversely on the speed and efficiency of the entire system. The process should be powerful to handle the entire operations. The hard disk should have sufficient capacity to store the file and application

|  |  |
| --- | --- |
| Processor | : Intel PentiumT4200/ Intel Core Duo 2.0 GHz / more |
| RAM | : Minimum 1GB RAM capacity |
| Hard disk | : Minimum 40 GB ROM capacity |
| Cache Memory | : L2-1 MB |
| GPU | : Intel HD Graphics |

### 2.2.2 Software Requirement

A major element in building a system is the section of compatible software since the software in the market is experiencing in geometric progression. Selected software should be acceptable by the firm and one user as well as it should be feasible for the system.

This document gives a detailed description of the software requirement specification. The study of requirement specification is focused specially on the functioning of the system. It allows the developer or analyst to understand the system, function to be carried out the performance level to be obtained and corresponding interfaces to be established.

Front End : PHP(Hypertext preprocessor)

Back End : XAMPP server, My SQL

Operation System : Windows 7 Or Windows 8.1 Or Windows 10

Client side : CSS (cascading Style sheet)

## 2.3 About Technologies used

* **HTML** is integrated in **PHP**. It provides a means to structure text-based information in a document. It allows users to produce web pages that include text, graphics and hyperlinks. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets.
* **CSS (**Cascading Style Sheets) is a style sheet language used for describing the presentation of a document written in a mark-up language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document.
* **MYSQL** is the language used to manipulate relational databases. It is tied closely with the relational model. It is issued for the purpose of data definition and data manipulation. Program runs as a server providing multi-user access to a number of databases. MySQL is a multithreaded, multi-user SQL database management system (DBMS). It includes facilities to add, modify or delete data from the database, ask questions (or queries) about the data stored in the database and produce reports summarizing selected contents.
* **PHP** is a scripting language originally designed for producing dynamic web pages. It has evolved to include a command line interface capability and can be used in standalone graphical applications. PHP is a general-purpose scripting language that is especially suited for web development. PHP generally runs on a web server, taking PHP code as its input and creating web pages as output. It can also be used for command-line scripting and client-side GUI applications. PHP can be deployed on most web servers, many operating systems and platforms, and can be used with many relational database management systems. It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use. PHP stores whole numbers in a platform-dependent range. It requires MySQL connection between the front end and back end components to write to the database and fetch required data. PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. On a web server, the result of the interpreted and executed PHP code which may be any type of data, such as generated HTML or binary image data would form the whole or part of an HTTP response. Various web template systems, web content management systems, and web frameworks exist which can be employed to orchestrate or facilitate the generation of that response. Additionally, PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their

**Chapter 3**

# SYSTEM DESIGN .

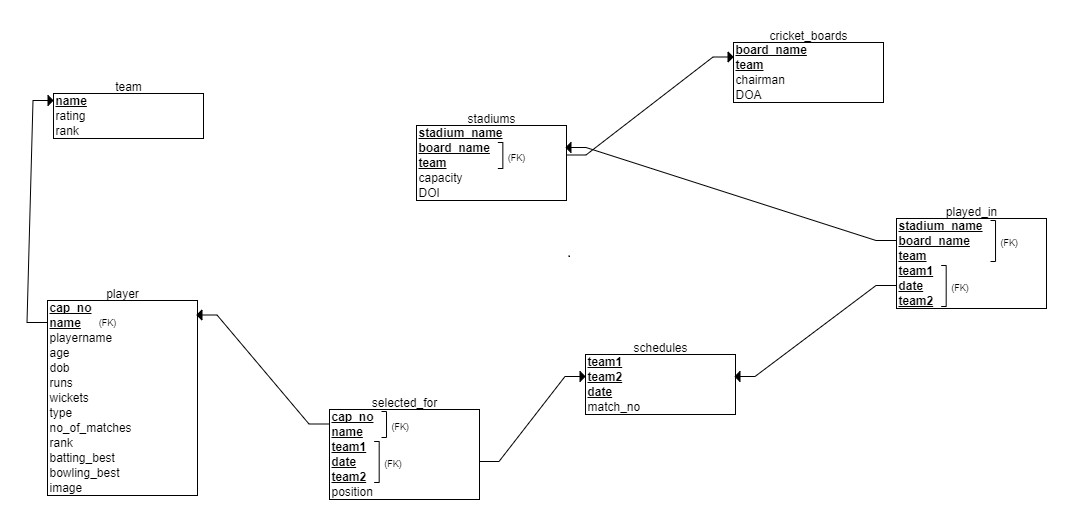
## 3.1 Database Design

The data in the system has to be stored and retrieved from database. Designing the database is part of system design.

Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MS Access database has been chosen for developing the relevant databases.

## 3.2 Relational Schema

The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database. A relational schema shows references among fields in the database. When a primary key is referenced in another table in the database, it is called a foreign key. This is denoted by an arrow with the head pointing at the referenced key attribute. A schema diagram helps organize values in the database. It also gives an idea of what order the tables should be created in. The following diagram shows the schema diagram for the database.

team1 team2 date match\_no venue

### *Figure 3.1: Relational schema*

In Fig 3.1 the schema diagram of Online CSE book store, where each table has attributes and some attributes are underlined ,it is called primary key and this primary key is referred in another table as foreign key and this representation is done with the schema diagram as shown in figure.

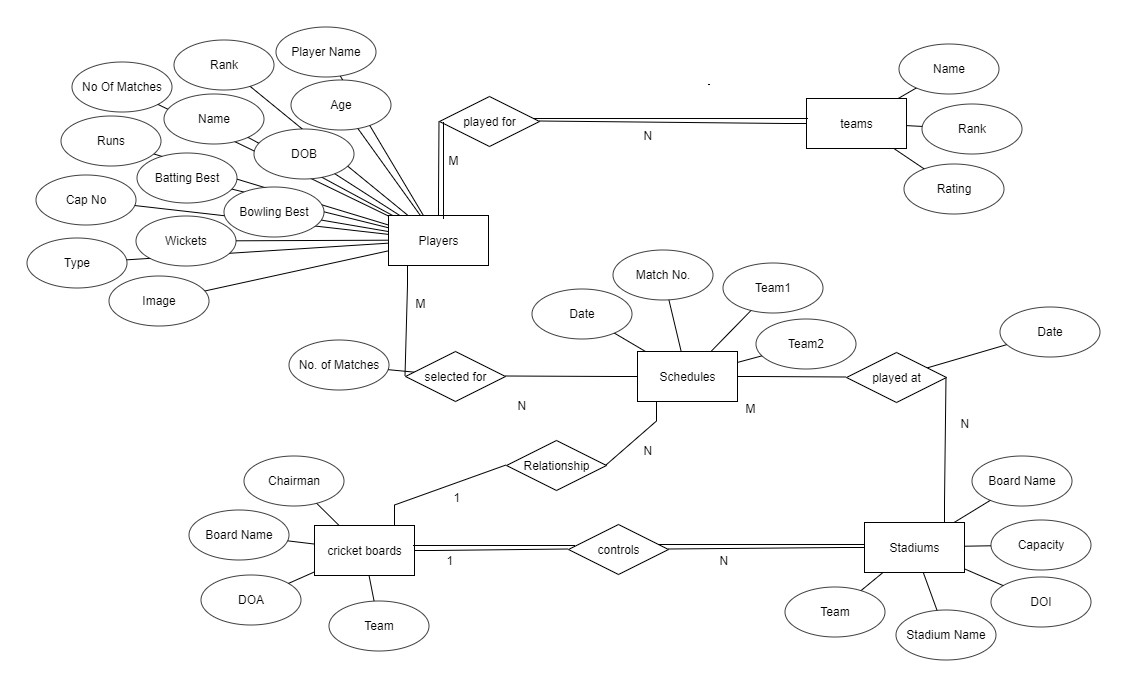
## 3.3 ER Diagram

An entity–relationship model is usually the result of systematic analysis to define and describe what is important to processes in an area of a business. An E-R model does not define the business processes; it only presents a business data schema in graphical form. It is usually drawn in a graphical form as boxes (entities) that are connected by lines (relationships) which express the associations and dependencies between entities. An ER model can also be expressed in a verbal form, for example: one building may be divided into zero or more apartments, but one apartment can only be located in one building. Entities may be characterized not only by relationships, but also by additional properties (attributes), which include identifiers called "primary keys". Diagrams created to represent attributes as well as entities and relationships may be called entity-attributerelationship diagrams, rather than entity-relationship models.

An ER model is typically implemented as a database. In a simple relational database implementation, each row of a table represents one instance of an entity type, and each field in a table represents an attribute type. In a relational database a relationship between entities is implemented by storing the primary key of one entity as a pointer or "foreign key" in the table of another entity. There is a tradition for ER/data models to be built at two or three levels of abstraction. Note that the conceptual-logical-physical hierarchy below is used in other kinds of specification, and is different from the three schema approach to software engineering.

The four main cardinal relationships are:

* One-to-one (1:1) - For example, each customer in a database is associated with one mailing address.
* One-to-many (1: N) - For example, a single customer might place an order for multiple products. The customer is associated with multiple entities, but all those entities have a single connection back to the same customer.
* Many-to-one (N: 1) – For example, many employees will have only one manager above them but one manager can have many employees below him.
* Many-to-many (M: N)- For example, at a company where all call center agents work with multiple customers, each agent is associated with multiple customers, and multiple customers might also be associated with multiple agents.



Venue

***Figure 3.2: ER-Diagram***.

## 3.4 Output Design

Designing computer output should proceed in an organized, well throughout manner; the right output element is designed so that people will find the system whether or executed. When we design an output, we must identify the specific output that is needed to meet the system. The usefulness of the new system is evaluated on the basis of their output. Once the output requirements are determined, the system designer can decide what to include in the system and how to structure it so that require output can be produced. For the proposed software, it is necessary that the output reports be compatible in format with the existing reports. The output must be concerned to the overall performance and the system’s working, as it should. Proper messages and appropriate directions can control errors committed by users. The output design is the key to the success of any system. Output is the key between the user and the sensor. The output must be concerned to the system’s working, as it should. Output design consists of displaying specifications and procedures as data presentation. User never left with the confusion as to what is happening without appropriate error and acknowledges message being received. Even an unknown person can operate the system without knowing anything about the system.

**Chapter 4**

# System Implementation

## 4.1 Implementation

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, an evaluation of change over methods. Implementation is the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. Any system developed should be secured and protected against possible hazards. Security measures are provided to prevent unauthorized access of the database at various levels.

## 4.2 Create Tables

**4.2.1 Admin**

|  |  |  |  |
| --- | --- | --- | --- |
| **SNO** | **COLUMN\_NAME** | **DATA\_TYPE** | **DESCRIPTION** |
| 1 | Name | Varchar | Primary Key |
| 2 | Passward | Varchar | Primary Key |

CREATE TABLE IF NOT EXISTS `admin` (

`name` varchar(20) COLLATE latin1\_general\_ci NOT NULL,

`pass` varchar(40) COLLATE latin1\_general\_ci NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1\_general\_ci;

INSERT INTO `admin` (`name`, `pass`) VALUES

('admin', 'd033e22ae348aeb5660fc2140aec35850c4da997');

**4.2.2 Books**

|  |  |  |  |
| --- | --- | --- | --- |
| **SNO** | **COLUMN\_NAME** | **DATA\_TYPE** | **DESCRIPTION** |
| 1 | Book\_isbn | Varchar | Primary Key |
| 2 | Book\_title | Varchar |  |
| 3 | Book\_author | Varchar |  |
| 4 | Book\_image | Varchar |  |
| 5 | Book\_descr | Text |  |
| 6 | Book\_price | Decimal |  |
| 7 | Publisherid | Int |  |

CREATE TABLE IF NOT EXISTS `books` (

`book\_isbn` varchar(20) COLLATE latin1\_general\_ci NOT NULL,

`book\_title` varchar(60) COLLATE latin1\_general\_ci DEFAULT NULL,

`book\_author` varchar(60) COLLATE latin1\_general\_ci DEFAULT NULL,

`book\_image` varchar(40) COLLATE latin1\_general\_ci DEFAULT NULL,

`book\_descr` text COLLATE latin1\_general\_ci,

`book\_price` decimal(6,2) NOT NULL,

`publisherid` int(10) unsigned NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1\_general\_ci;

INSERT INTO `books` (`book\_isbn`, `book\_title`, `book\_author`, `book\_image`, `book\_descr`, `book\_price`, `publisherid`) VALUES

('978-0-321-94786-4', 'Learning Mobile App Development', 'Jakob Iversen, Michael Eierman', 'mobile\_app.jpg', 'Now, one book can help you master mobile app development with both market-leading platforms: Apple''s iOS and Google''s Android. Perfect for both students and professionals, Learning Mobile App Development is the only tutorial with complete parallel coverage of both iOS and Android. With this guide, you can master either platform, or both - and gain a deeper understanding of the issues associated with developing mobile apps.\r\n\r\nYou''ll develop an actual working app on both iOS and Android, mastering the entire mobile app development lifecycle, from planning through licensing and distribution.\r\n\r\nEach tutorial in this book has been carefully designed to support readers with widely varying backgrounds and has been extensively tested in live developer training courses. If you''re new to iOS, you''ll also find an easy, practical introduction to Objective-C, Apple''s native language.', '250.00', 6),

('978-0-7303-1484-4', 'Doing Good By Doing Good', 'Peter Baines', 'doing\_good.jpg', 'Doing Good by Doing Good shows companies how to improve the bottom line by implementing an engaging, authentic, and business-enhancing program that helps staff and business thrive. International CSR consultant Peter Baines draws upon lessons learnt from the challenges faced in his career as a police officer, forensic investigator, and founder of Hands Across the Water to describe the Australian CSR landscape, and the factors that make up a program that benefits everyone involved. Case studies illustrate the real effect of CSR on both business and society, with clear guidance toward maximizing involvement, engaging all employees, and improving the bottom line. The case studies draw out the companies that are focusing on creating shared value in meeting the challenges of society whilst at the same time bringing strong economic returns.\r\n\r\nConsumers are now expecting that big businesses with ever-increasing profits give back to the community from which those profits arise. At the same time, shareholders are demanding their share and are happy to see dividends soar. Getting this right is a balancing act, and Doing Good by Doing Good helps companies delineate a plan of action for getting it done.', '200.00', 2),

('978-1-118-94924-5', 'Programmable Logic Controllers', 'Dag H. Hanssen', 'logic\_program.jpg', 'Widely used across industrial and manufacturing automation, Programmable Logic Controllers (PLCs) perform a broad range of electromechanical tasks with multiple input and output arrangements, designed specifically to cope in severe environmental conditions such as automotive and chemical plants.Programmable Logic Controllers: A Practical Approach using CoDeSys is a hands-on guide to rapidly gain proficiency in the development and operation of PLCs based on the IEC 61131-3 standard. Using the freely-available\* software tool CoDeSys, which is widely used in industrial design automation projects, the author takes a highly practical approach to PLC design using real-world examples. The design tool, CoDeSys, also features a built in simulator / soft PLC enabling the reader to undertake exercises and test the examples.', '20.00', 2),

('978-1-1180-2669-4', 'Professional JavaScript for Web Developers, 3rd Edition', 'Nicholas C. Zakas', 'pro\_js.jpg', 'If you want to achieve JavaScript''s full potential, it is critical to understand its nature, history, and limitations. To that end, this updated version of the bestseller by veteran author and JavaScript guru Nicholas C. Zakas covers JavaScript from its very beginning to the present-day incarnations including the DOM, Ajax, and HTML5. Zakas shows you how to extend this powerful language to meet specific needs and create dynamic user interfaces for the web that blur the line between desktop and internet. By the end of the book, you''ll have a strong understanding of the significant advances in web development as they relate to JavaScript so that you can apply them to your next website.', '300.00', 1),

('978-1-44937-019-0', 'Learning Web App Development', 'Semmy Purewal', 'web\_app\_dev.jpg', 'Grasp the fundamentals of web application development by building a simple database-backed app from scratch, using HTML, JavaScript, and other open source tools. Through hands-on tutorials, this practical guide shows inexperienced web app developers how to create a user interface, write a server, build client-server communication, and use a cloud-based service to deploy the application.\r\n\r\nEach chapter includes practice problems, full examples, and mental models of the development workflow. Ideal for a college-level course, this book helps you get started with web app development by providing you with a solid grounding in the process.', '250.00', 3),

('978-1-44937-075-6', 'Beautiful JavaScript', 'Anton Kovalyov', 'beauty\_js.jpg', 'JavaScript is arguably the most polarizing and misunderstood programming language in the world. Many have attempted to replace it as the language of the Web, but JavaScript has survived, evolved, and thrived. Why did a language created in such hurry succeed where others failed?\r\n\r\nThis guide gives you a rare glimpse into JavaScript from people intimately familiar with it. Chapters contributed by domain experts such as Jacob Thornton, Ariya Hidayat, and Sara Chipps show what they love about their favorite language - whether it''s turning the most feared features into useful tools, or how JavaScript can be used for self-expression.', '350.00', 3),

('978-1-4571-0402-2', 'Professional ASP.NET 4 in C# and VB', 'Scott Hanselman', 'pro\_asp4.jpg', 'ASP.NET is about making you as productive as possible when building fast and secure web applications. Each release of ASP.NET gets better and removes a lot of the tedious code that you previously needed to put in place, making common ASP.NET tasks easier. With this book, an unparalleled team of authors walks you through the full breadth of ASP.NET and the new and exciting capabilities of ASP.NET 4. The authors also show you how to maximize the abundance of features that ASP.NET offers to make your development process smoother and more efficient.', '300.00', 1),

('978-1-484216-40-8', 'Android Studio New Media Fundamentals', 'Wallace Jackson', 'android\_studio.jpg', 'Android Studio New Media Fundamentals is a new media primer covering concepts central to multimedia production for Android including digital imagery, digital audio, digital video, digital illustration and 3D, using open source software packages such as GIMP, Audacity, Blender, and Inkscape. These professional software packages are used for this book because they are free for commercial use. The book builds on the foundational concepts of raster, vector, and waveform (audio), and gets more advanced as chapters progress, covering what new media assets are best for use with Android Studio as well as key factors regarding the data footprint optimization work process and why new media content and new media data optimization is so important.', '250.00', 4),

('978-1-484217-26-9', 'C++ 14 Quick Syntax Reference, 2nd Edition', ' Mikael Olsson', 'c\_14\_quick.jpg', 'This updated handy quick C++ 14 guide is a condensed code and syntax reference based on the newly updated C++ 14 release of the popular programming language. It presents the essential C++ syntax in a well-organized format that can be used as a handy reference.\r\n\r\nYou won''t find any technical jargon, bloated samples, drawn out history lessons, or witty stories in this book. What you will find is a language reference that is concise, to the point and highly accessible. The book is packed with useful information and is a must-have for any C++ programmer.\r\n\r\nIn the C++ 14 Quick Syntax Reference, Second Edition, you will find a concise reference to the C++ 14 language syntax. It has short, simple, and focused code examples. This book includes a well laid out table of contents and a comprehensive index allowing for easy review.', '400.00', 4),

('978-1-49192-706-9', 'C# 6.0 in a Nutshell, 6th Edition', 'Joseph Albahari, Ben Albahari', 'c\_sharp\_6.jpg', 'When you have questions about C# 6.0 or the .NET CLR and its core Framework assemblies, this bestselling guide has the answers you need. C# has become a language of unusual flexibility and breadth since its premiere in 2000, but this continual growth means there''s still much more to learn.\r\n\r\nOrganized around concepts and use cases, this thoroughly updated sixth edition provides intermediate and advanced programmers with a concise map of C# and .NET knowledge. Dive in and discover why this Nutshell guide is considered the definitive reference on C#.', '500.00', 3);

**4.2.3 Customers**

|  |  |  |  |
| --- | --- | --- | --- |
| **SNO** | **COLUMN\_NAME** | **DATA\_TYPE** | **DESCRIPTION** |
| 1 | customerid | Int | Primary Key |
| 2 | name | Varchar |  |
| 3 | address | Varchar |  |
| 4 | city | Varchar |  |
| 5 | zip\_code | Varchar |  |
| 6 | country | Varchar |  |

CREATE TABLE IF NOT EXISTS `customers` (

`customerid` int(10) unsigned NOT NULL,

`name` varchar(60) COLLATE latin1\_general\_ci NOT NULL,

`address` varchar(80) COLLATE latin1\_general\_ci NOT NULL,

`city` varchar(30) COLLATE latin1\_general\_ci NOT NULL,

`zip\_code` varchar(10) COLLATE latin1\_general\_ci NOT NULL,

`country` varchar(60) COLLATE latin1\_general\_ci NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=latin1 COLLATE=latin1\_general\_ci;

INSERT INTO `customers` (`customerid`, `name`, `address`, `city`, `zip\_code`, `country`) VALUES

(1, 'a', 'a', 'a', 'a', 'a'),

(2, 'b', 'b', 'b', 'b', 'b'),

(3, 'test', '123 test', '12121', 'test', 'test');

**4.2.4 Orders**

|  |  |  |  |
| --- | --- | --- | --- |
| **SNO** | **COLUMN\_NAME** | **DATA\_TYPE** | **DESCRIPTION** |
| 1 | Orderid | Int | Primary Key |
| 2 | Customerid | Int |  |
| 3 | Amount | Decimal |  |
| 4 | Date | Timestamp |  |
| 5 | Ship\_name | Char |  |
| 6 | Ship\_address | Char |  |
| 7 | Ship\_city | Char |  |
| 8 | Ship\_zip\_code | Char |  |
| 9 | Ship\_country | Char |  |

CREATE TABLE IF NOT EXISTS `orders` (

`orderid` int(10) unsigned NOT NULL,

`customerid` int(10) unsigned NOT NULL,

`amount` decimal(6,2) DEFAULT NULL,

`date` timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`ship\_name` char(60) COLLATE latin1\_general\_ci NOT NULL,

`ship\_address` char(80) COLLATE latin1\_general\_ci NOT NULL,

`ship\_city` char(30) COLLATE latin1\_general\_ci NOT NULL,

`ship\_zip\_code` char(10) COLLATE latin1\_general\_ci NOT NULL,

`ship\_country` char(20) COLLATE latin1\_general\_ci NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=latin1 COLLATE=latin1\_general\_ci;

INSERT INTO `orders` (`orderid`, `customerid`, `amount`, `date`, `ship\_name`, `ship\_address`, `ship\_city`, `ship\_zip\_code`, `ship\_country`) VALUES

(1, 1, '60.00', '2015-12-03 13:30:12', 'a', 'a', 'a', 'a', 'a'),

(2, 2, '60.00', '2015-12-03 13:31:12', 'b', 'b', 'b', 'b', 'b'),

(3, 3, '20.00', '2015-12-03 19:34:21', 'test', '123 test', '12121', 'test', 'test'),

(4, 1, '20.00', '2015-12-04 10:19:14', 'a', 'a', 'a', 'a', 'a');

**4.2.4 Order\_items**

|  |  |  |  |
| --- | --- | --- | --- |
| **SNO** | **COLUMN\_NAME** | **DATA\_TYPE** | **DESCRIPTION** |
| 1 | Orderid | Int |  |
| 2 | Book\_isbn | Varchar |  |
| 3 | Item\_price | Decimal |  |
| 4 | Quantity | tinyint |  |

CREATE TABLE IF NOT EXISTS `order\_items` (

`orderid` int(10) unsigned NOT NULL,

`book\_isbn` varchar(20) COLLATE latin1\_general\_ci NOT NULL,

`item\_price` decimal(6,2) NOT NULL,

`quantity` tinyint(3) unsigned NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1\_general\_ci;

INSERT INTO `order\_items` (`orderid`, `book\_isbn`, `item\_price`, `quantity`) VALUES

(1, '978-1-118-94924-5', '20.00', 1),

(1, '978-1-44937-019-0', '20.00', 1),

(1, '978-1-49192-706-9', '20.00', 1),

(2, '978-1-118-94924-5', '20.00', 1),

(2, '978-1-44937-019-0', '20.00', 1),

(2, '978-1-49192-706-9', '20.00', 1),

(3, '978-0-321-94786-4', '20.00', 1),

(1, '978-1-49192-706-9', '20.00', 1);

**4.2.5 Publishers**

|  |  |  |  |
| --- | --- | --- | --- |
| **SNO** | **COLUMN\_NAME** | **DATA\_TYPE** | **DESCRIPTION** |
| 1 | Publisherid | Int | Primary Key |
| 2 | Publisher\_name | Varchar |  |

CREATE TABLE IF NOT EXISTS `publisher` (

`publisherid` int(10) unsigned NOT NULL,

`publisher\_name` varchar(60) COLLATE latin1\_general\_ci NOT NULL

) ENGINE=InnoDB AUTO\_INCREMENT=7 DEFAULT CHARSET=latin1 COLLATE=latin1\_general\_ci;

INSERT INTO `publisher` (`publisherid`, `publisher\_name`) VALUES

(1, 'Wrox'),

(2, 'Wiley'),

(3, 'O''Reilly Media'),

(4, 'Apress'),

(5, 'Packt Publishing'),

(6, 'Addison-Wesley');

## 4.3 Triggers and Stored procedures

* Triggers are the user defied constraints which will be checked before insertion

or updating.

* The trigger used here updates the ordered books automatically in order log.
* Stored procedures are like functions in C or java, it performs specific instructions like calculating, updating and storing automatically into the database.
* In this project the stored procedure retrieves all values related to books.

### 4.3.1 Stored Procedure

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `stadium`() NO SQL select \* from stadiums$$

DELIMITER;

### 4.3.2 Trigger

CREATE TRIGGER `default\_date`

BEFORE INSERT ON `stadiums` FOR EACH ROW setnew.DOI=CURRENT\_DATE();

CREATE TRIGGER `update` BEFORE UPDATE ON `player`

FOR EACH ROW BEGIN

IF (new.runs<old.runs) THEN SET new.runs=old.runs;

IF (new.wickets<old.wickets) THEN SET new.wickets=old.wickets;

END IF;

IF (new.no\_of\_matches<old.no\_of\_matches) THEN SET new.no\_of\_matches=old.no\_of\_matches;

END IF;

END IF;

END

## 4.4 Result

The resulting system is able to:

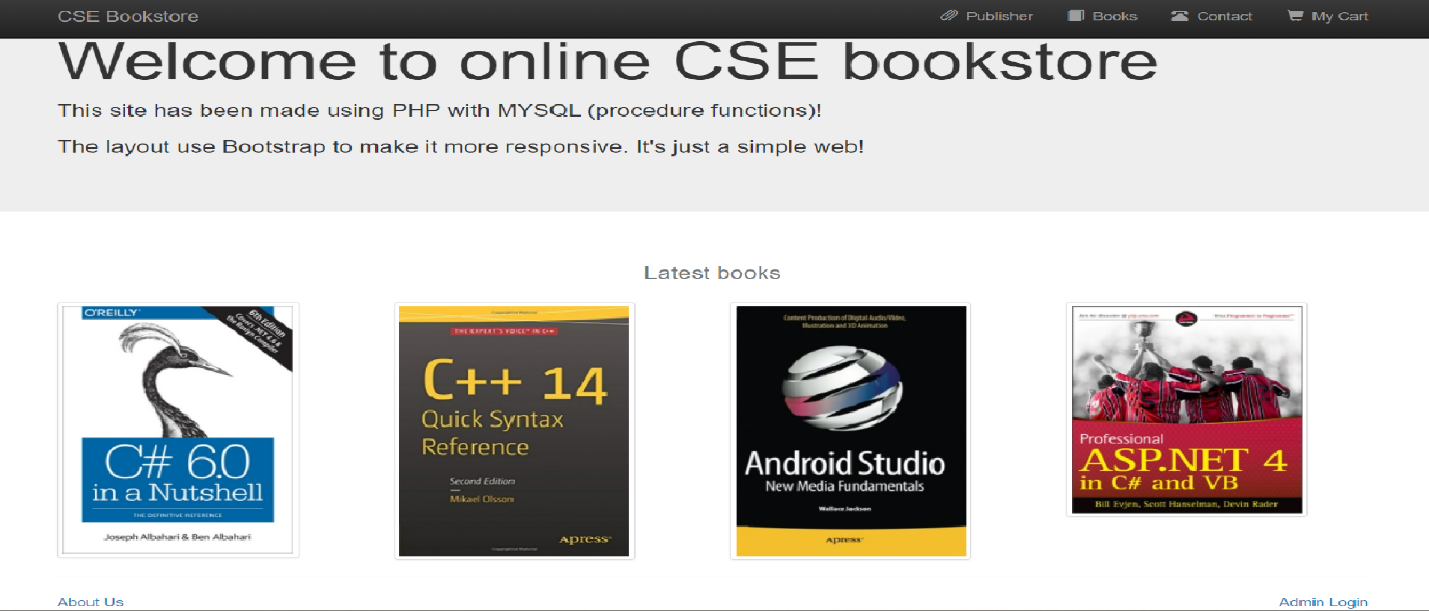
* Allows user to quickly and easily look for details of particular required data.
* The user can see the details required.
* Gives accurate information as updated by the admin.
* Only admin has the rightful access to modify the data in database, users can only view the modified changes in the user view

**Chapter 5**

**Snapshots**

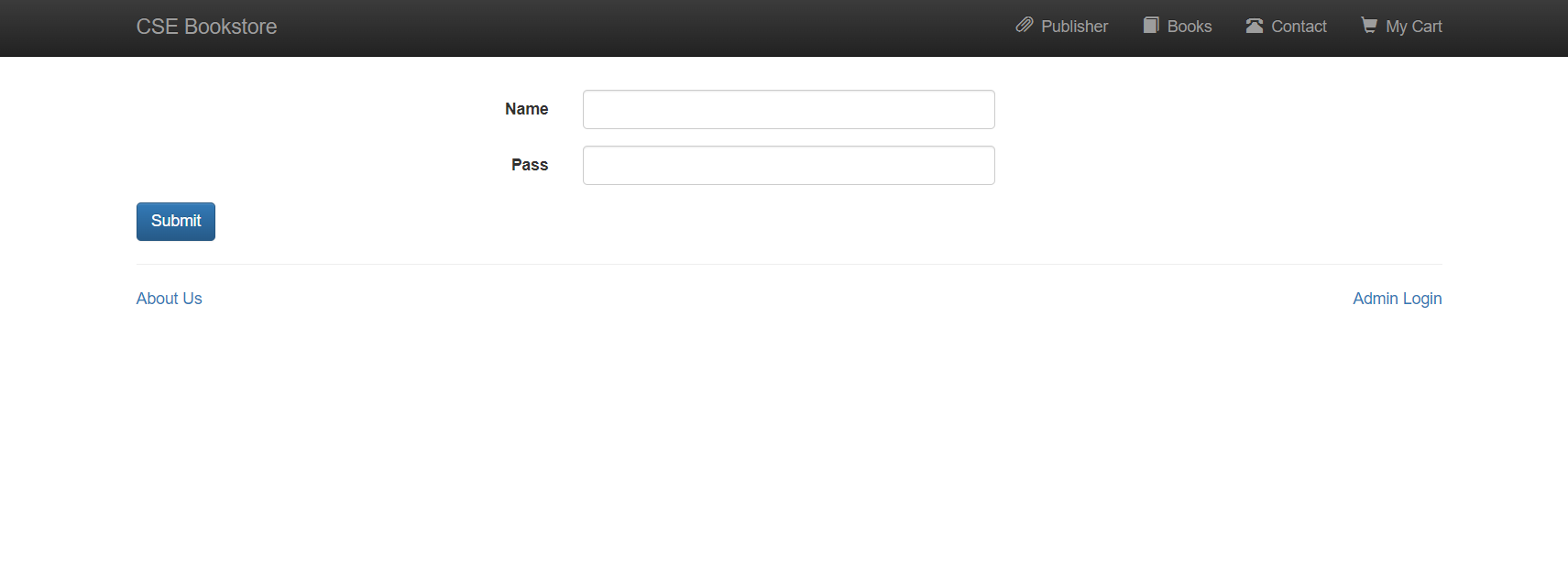
## 5.1 Home Page

**Description:** It shows Home Page which allows to login as admin . admin is the owner of the data base where as users are clients

****

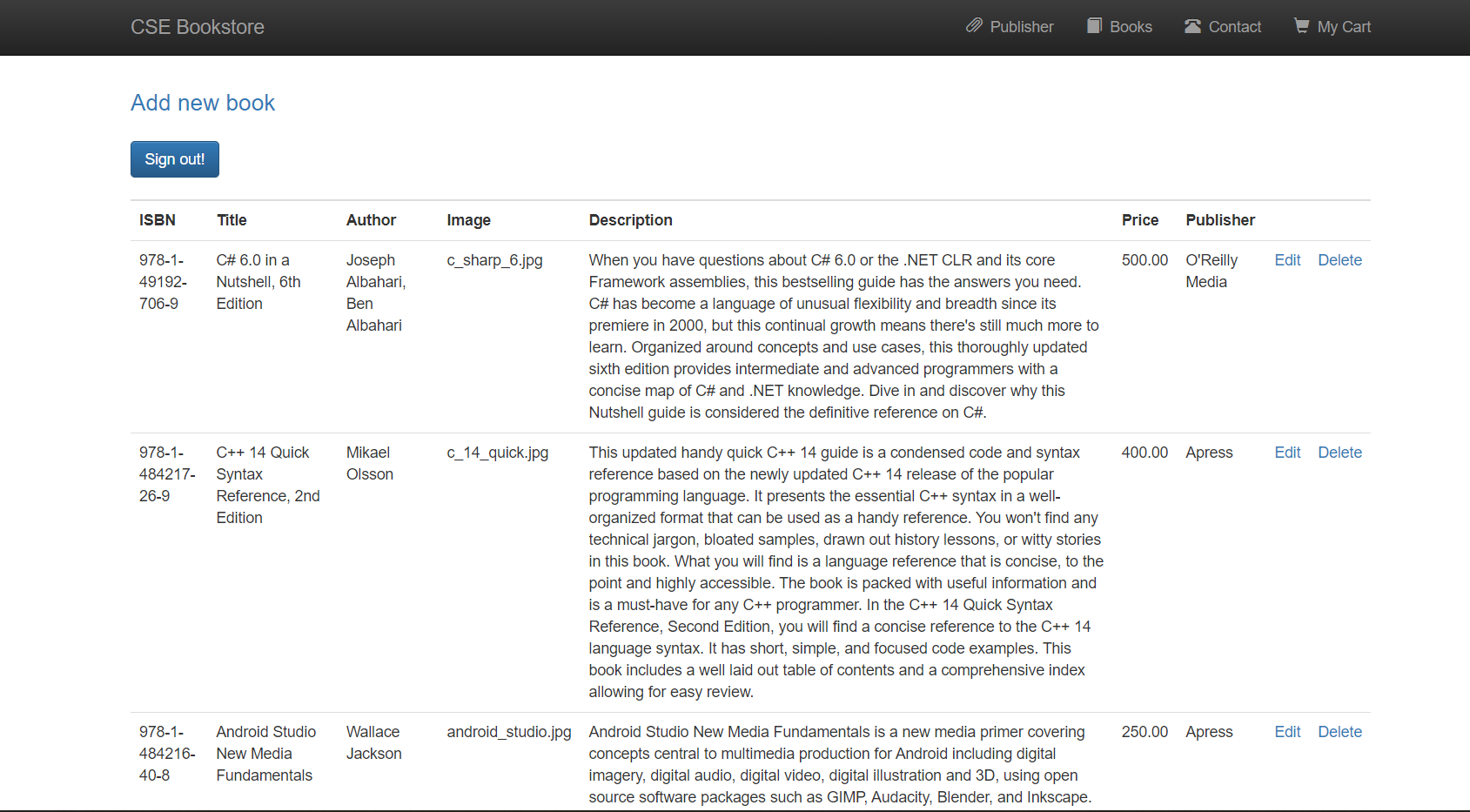
## 5.2 Admin Login Page

**Description:** It shows admin login page which gives authentication to enter into the Admin page. it navigates to the admin home page if username and password are correct, else it popup error.



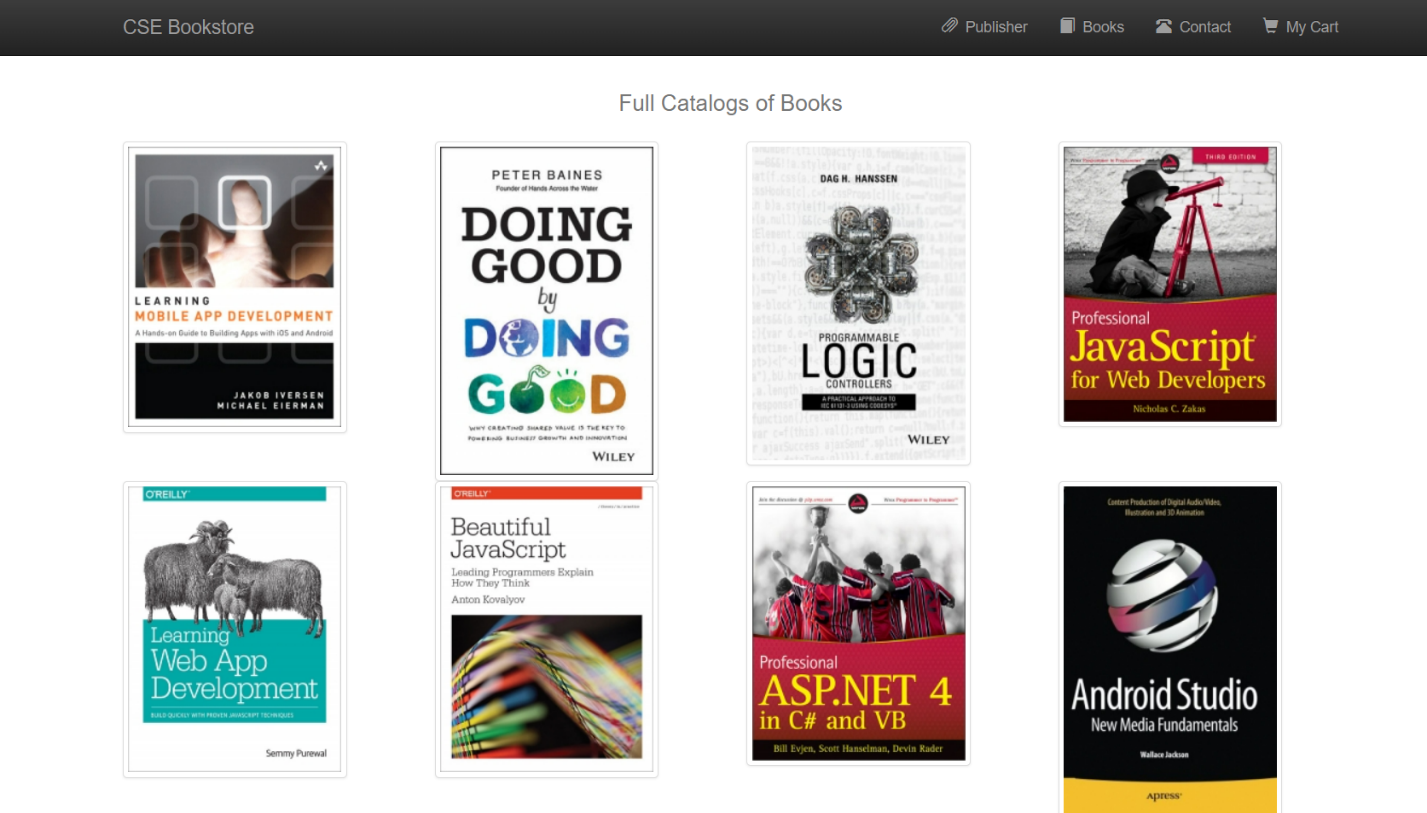
## 5.3 Admin’s Home Page

**Description:** Shows window which allows to add, delete, and edit of Books, title, author, description, price and publishers.



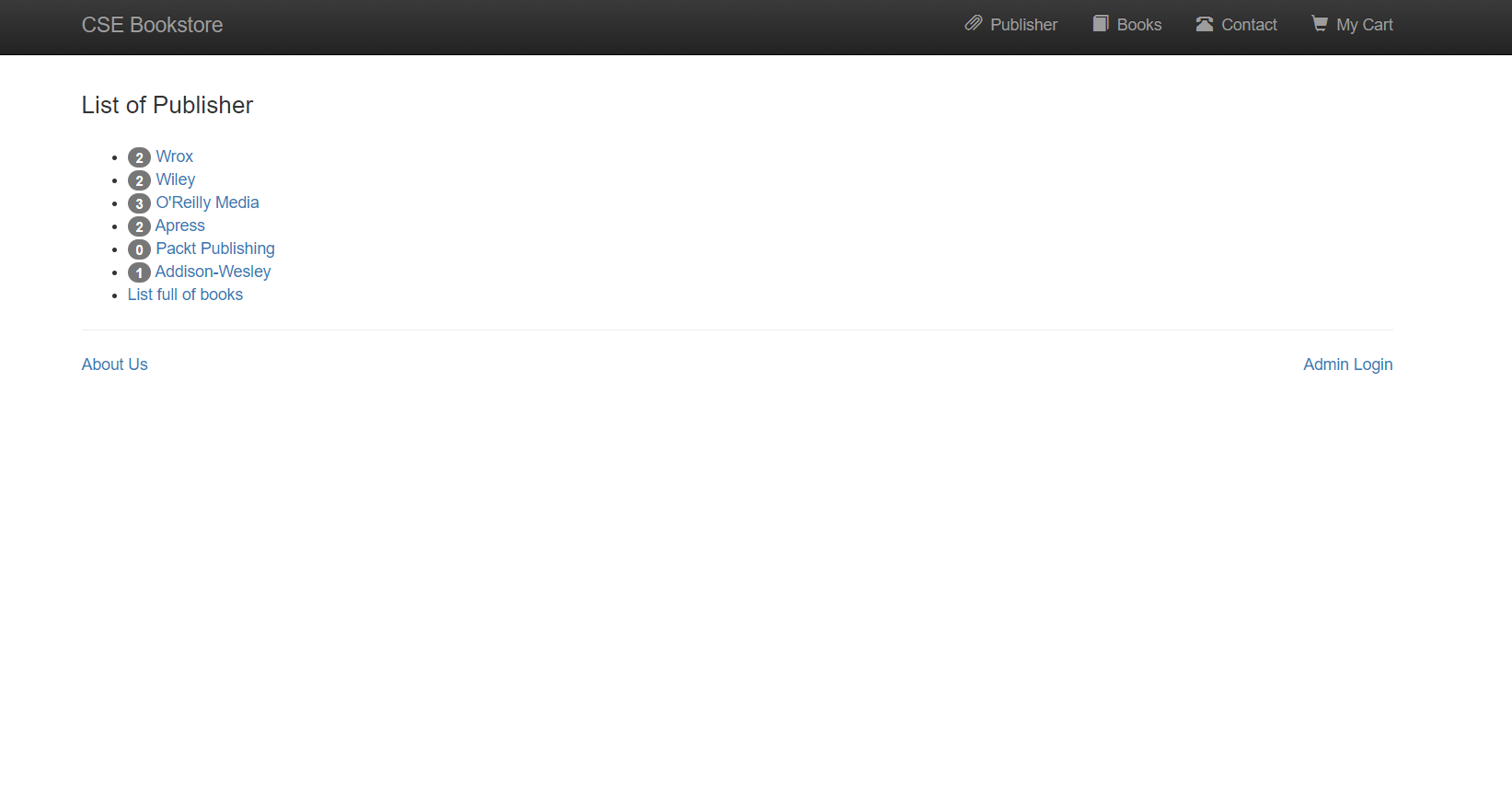
## 5.4 Books page

**Description:** It shows the books available for the customer to purchase.



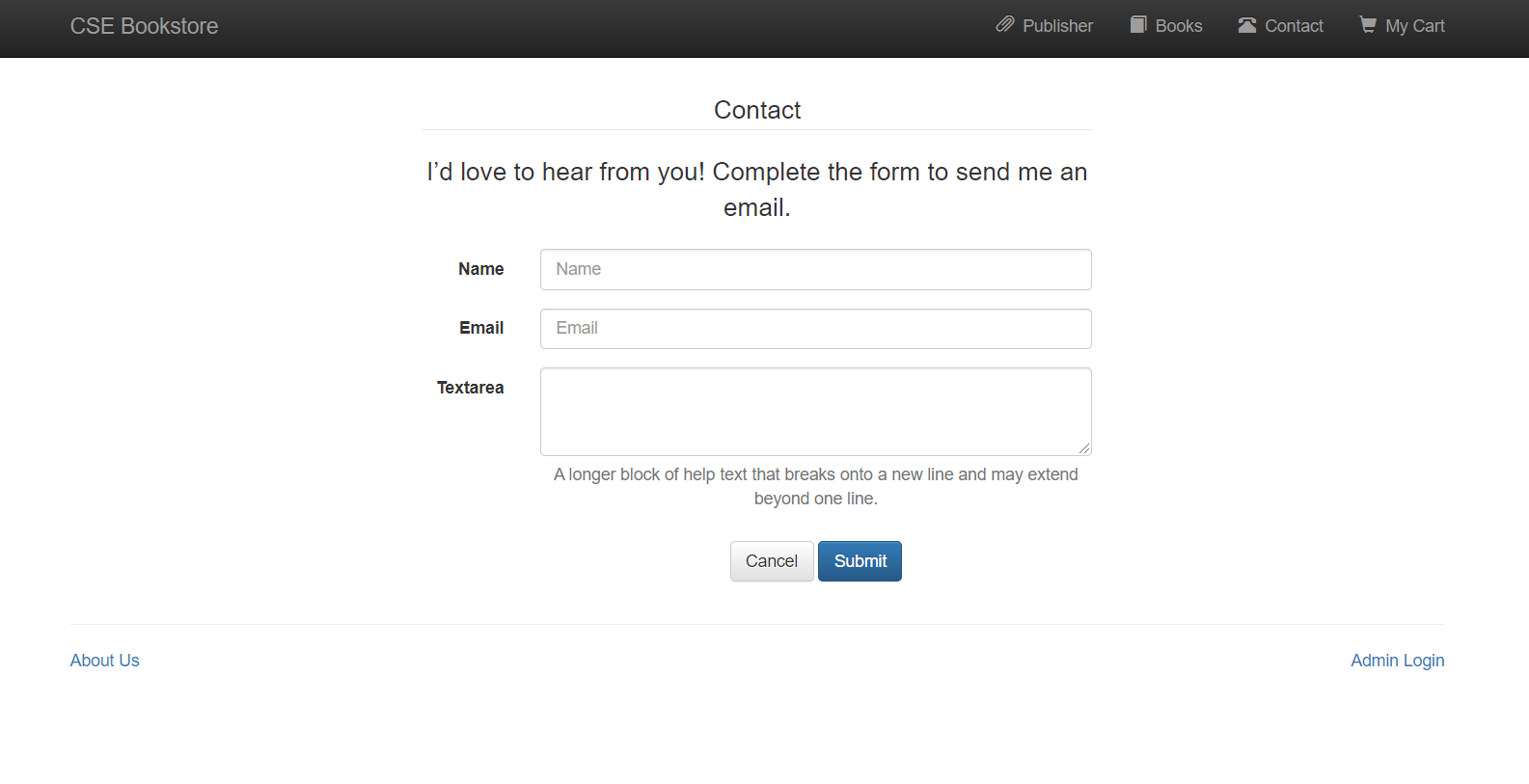
## 5.5 Publisher page

**Description:** It shows publisher name along with the number of books published.



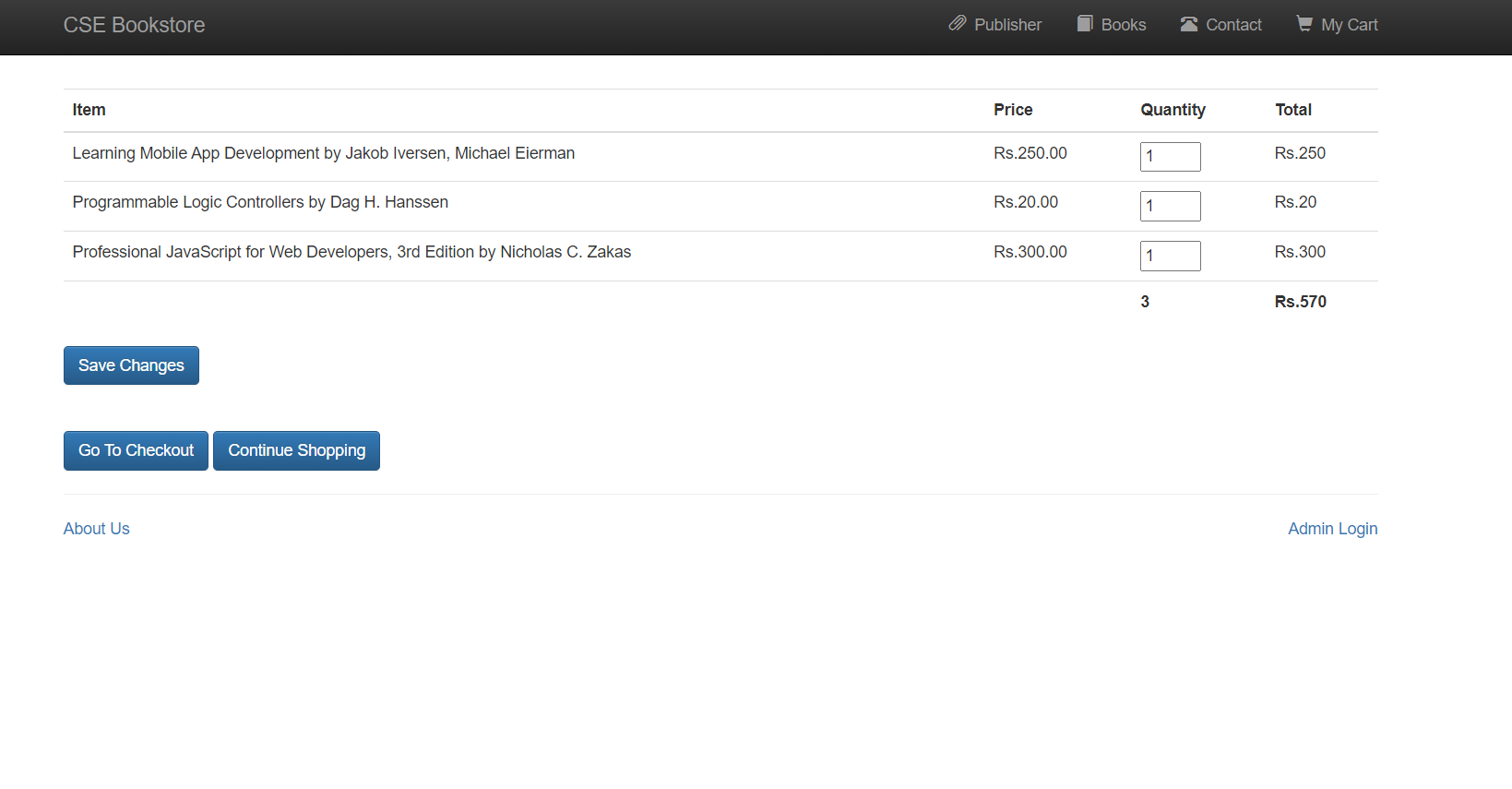
## 5.6 Contact page

**Description:** It contain Customer name, Email address and Text area.



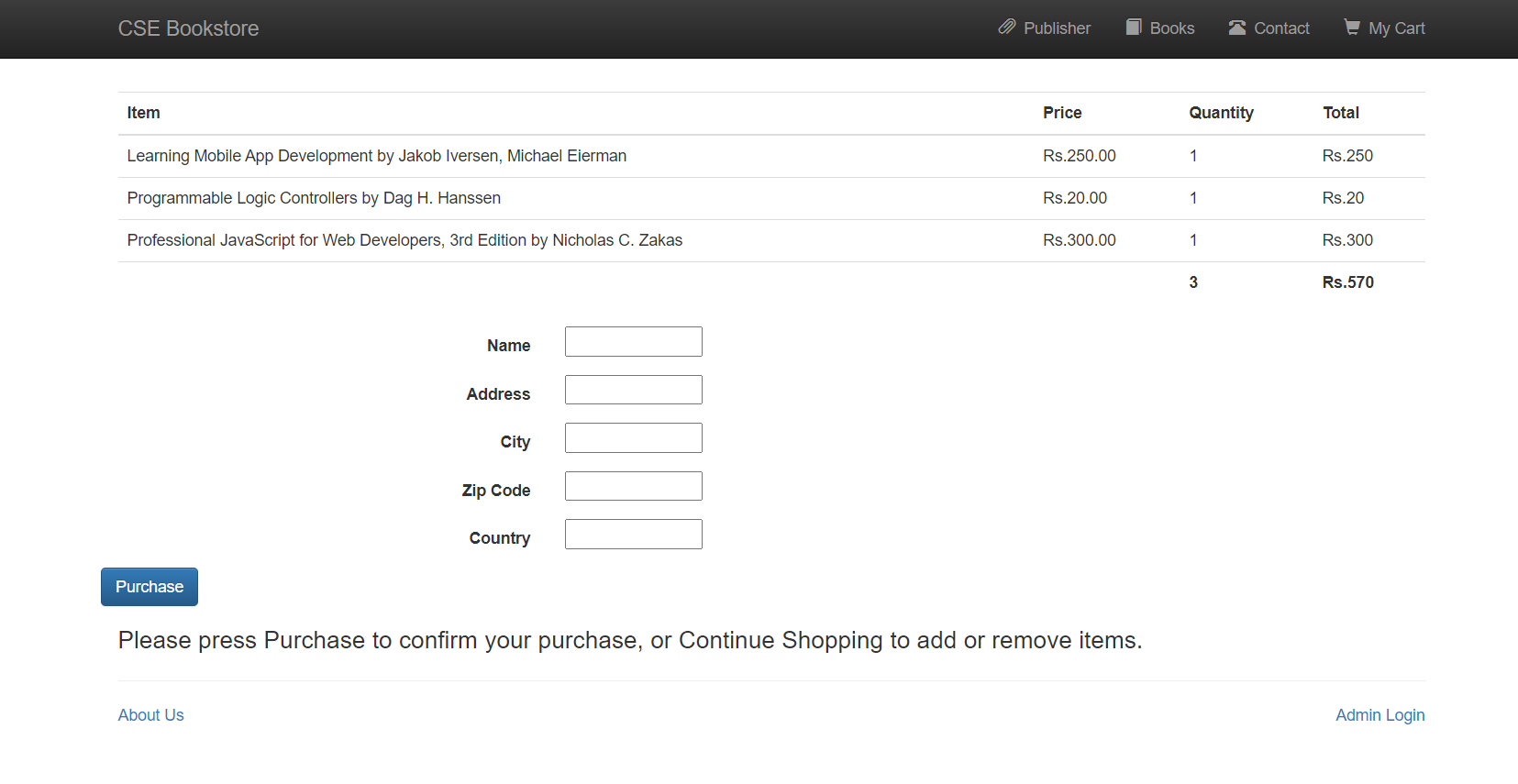
## 5.7 My cart page

**Description:** It shows the items selected for purchase along with price and also quantity.



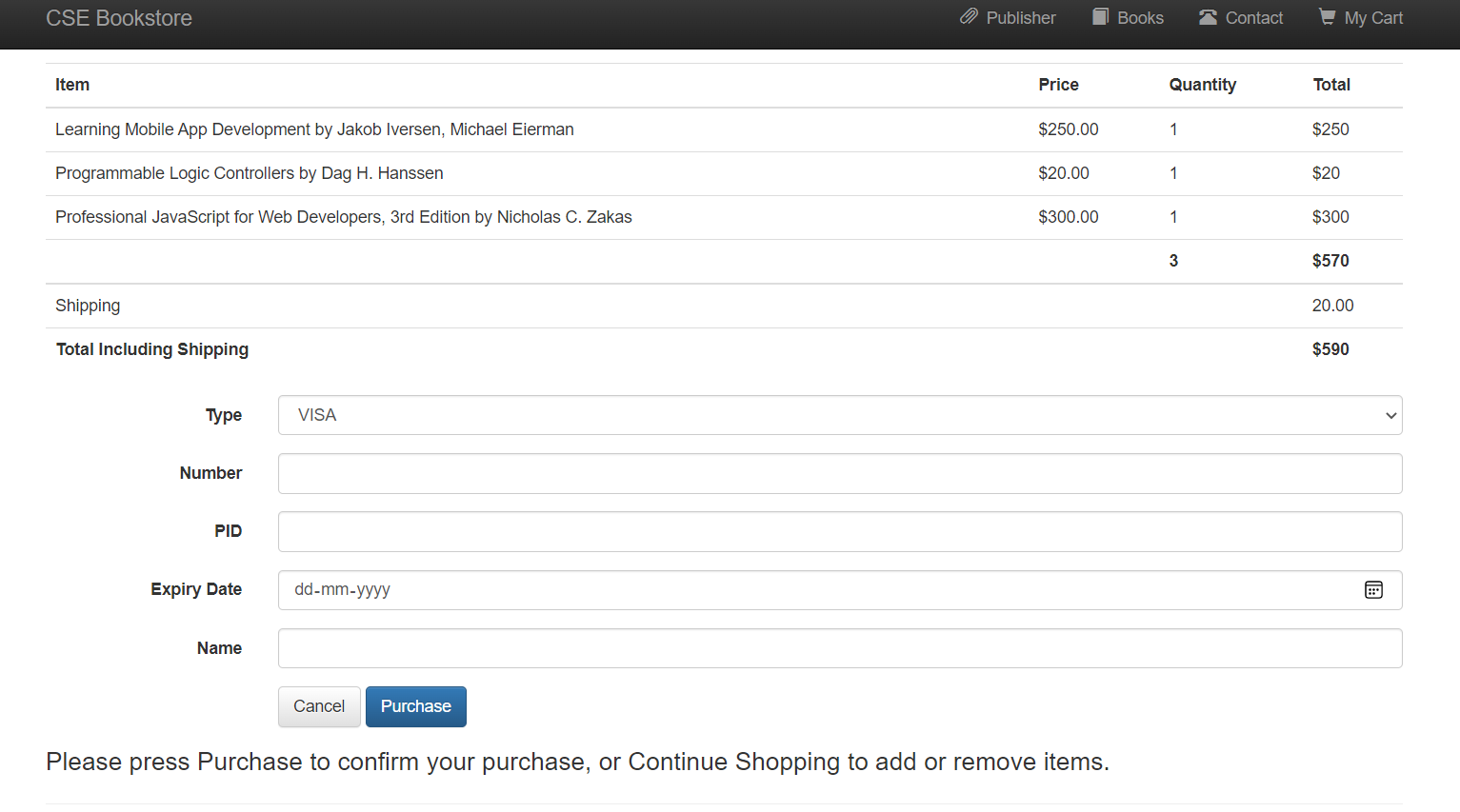
## 5.8 Checkout page

**Description:** It contain information related to shipping address along with customer name.



## 5.9 Payment page

**Description:** It shows the debit card details entered by the customer.



**Chapter 6**

**Conclusion and Future Enhancements**

# 6.1 Conclusion

IPL database management project, developed using PHP and MySQL is based on the requirement specification of the user and the analysis of the existing system, with flexibility for future enhancement. IPL database management software is designed for people who want to manage various particulars can be known by recording them in the database. various records and particulars about match got increased rapidly. Thereby the numbers of matches and there is going to be increased day-by-day. And hence there is a lot of strain on the person who are watching the IPL to know about future matches and also to see the records done by various players and getting details in fingertips. Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented. user can easily see the matches scheduled, player ranking , team ranking, stadiums information, and players profile with their stats in the ipl. It is also easy for admin to make changes to the data in the database in more efficient way.

# Bibiliography

1. RamezElmasri and Shamkant B. Navathe,“Database systems Models, Languages, Design and Application Programming”, 6th Edition, Pearson,2017.
2. Ramakrishnan and Gehrke, “Database management systems”,3rd Edition

McGraw Hill2014

1. SilberschatzKorth and Sudharshan, “Database System Concepts”, 6th Edition Mc-Graw Hill, 2013.
2. Coronel, Morris,and Rob “Database Principles Fundamentals of Design,

Implementation and Management” Cengage Learning 2012

1. Abraham Silberchatz, Henry korth and S.Sudarshan, “Database System Concepts”, McGraw-Hill Education, 16 Jun 2010.
2. Jeffery D.Ullman, “Principles of Data Base System”, Financial Times Prentice Hall 2nd Revised edition(1 December 1982).
3. [http://stackoverflow.com](http://stackoverflow.com/)
4. [https://www.w3schools.com](https://www.w3schools.com/)
5. [http://www.phptpoint.com](http://www.phptpoint.com/)
6. [https://www.bootply.com](https://www.bootply.com/)
7. [https://www.tutorialspoint.com](https://www.tutorialspoint.com/)
8. [http://1000projects.org](http://1000projects.org/)
9. [https://erdplus.com](https://erdplus.com/#/)