online musical instrumental store management system project report.

Kamal Acharya[1]

[1]Affiliation not available

October 31, 2024

Abstract

A musical instrument store is a web based application where users can view various musical instruments along with their description. The project provides user with a flexible and attractive GUI and shows him a list of products and carry out all the shopping activities online. This project is developed for users to have a brief look at the instruments without actually visiting the store. The website displays different kinds of instruments of different brands so that user can easily get their expected instrument. Users can explore instruments available and add it to their cart. Once the user is done with products selection, the system calculates the overall cost of the products bought and generates online bill for the user. User can even make payment online using credit card system. Having paid the bill online the user receives an email notification on his registered email id that provides a bill receipt of the instruments bought. The system also has an admin account that shows when visitors visited the site, which instruments they bought, their bill amount and when they logged out.

ABSTRACT

A musical instrument store is a web based application where users can view various musical instruments along with their description. The project provides user with a flexible and attractive GUI and shows him a list of products and carry out all the shopping activities online.

This project is developed for users to have a brief look at the instruments without actually visiting the store. The website displays different kinds of instruments of different brands so that user can easily get their expected instrument.

Users can explore instruments available and add it to their cart. Once the user is done with products selection, the system calculates the overall cost of the products bought and generates online bill for the user. User can even make payment online using credit card system. Having paid the bill online the user receives an email notification on his registered email id that provides a bill receipt of the instruments bought. The system also has an admin account that shows when visitors visited the site, which instruments they bought, their bill amount and when they logged out.

Chapter 1

Introduction

1.1 Background

The objective of this project is to implement a Musical Instrumental Store web application with user interface. The motivation of this project comes from my desire to learn the increasingly growing field of .NET, SQL server database designing, website designing and their growing popularity by taking up this case study.

An online Musical Instrumental Store that allows users to check for various instruments available at the online store and purchase online. The project consists of list of Musical Instrumental tools displayed in various categories. The user may browse through these items as per categories. If the user likes a product he or she may add it to their shopping carts. Once user wishes to checkout they must register on the site first. Then they can login using same id password next time. Now they can pay through a credit card or cash on delivery. Once the user makes a successful transaction he or she gets a copy of the shopping receipt on his email id. Here we use user friendly interface to make the entire frontend. The middle tier or code behind model is designed for fast processing. And SQL serves as a backend to store tools lists data. Thus, the online Musical Instrumental Store project brings an entire shop online and makes it easy for both buyer and seller. This project is helpful to computerize the sales activities and the payment given to the user.

Shops are generally closed on their specific days but online Musical Instrumental Store will be open all the time for the service of the people. Online Musical Instrumental Store is very fast technique for ordering instruments at home.
People don't need large places for their shops they can run these shops by sitting at their home with the help of internet.

So in generally Online Musical Instrumental Store’s only motive is to provide best quality of instrumental tools in time and without taking any efforts and providing a good service to each and every member of the society.

1.2 Objectives

- Providing with latest instruments of different categories and various geners.
- Maintains proper database and information.
- The system calculates bill instantly and user can pay online.
- Provides after sales services.
- Saves time, efforts and money.

1.3 Purpose, Scope, and Applicability

1.3.1 Purpose

The purpose of this project is to explore the capabilities of the Microsoft .NET Framework and to provide a convenient service of buying a musical instrument to online customers.

1.3.2 Scope

- Establish a well-respected music lesson department and start a rapport early with school band and church music leaders.
- Offer extended hours to serve a larger portion of the buying public than our competitors do.
- Educate the buying public by merchandising our products with informational/tutorial signage and literature, and by backing that up with knowledgeable salespeople.
- Offer the services of a full time repair department to our client base.
- Continually modify the product and service offerings to stay on the leading edge of technology within our market.
- Exploit the many weaknesses of our local and national competitors to differentiate ourselves from them.
- Users can view detailed of the parts without going anywhere.
- Cost is calculated by the system which saves time and efforts.

1.3.3 Applicability

This application called the Musical Instrumental Store is implemented using PHP and MySQL. This project covers the following applications**:**

An online product catalog that can be browsed:

The work starts with adding many new product catalog features which includes displaying categories, products, and product details.

Searching the Catalog:

For the visual part, a text box is used in which the visitor can enter one or more words to search through the product catalog. The words entered by the visitor are searched for in the products names and descriptions.

A Custom Shopping Cart and checkout in PHP:

A custom shopping basket is implemented, which stores its data into the local database. Also a “shopping cart summary control” is created that shows up in every catalog page except the shopping cart page.

Handling Customer Accounts:

Customers can log in via a login page or dialog box to get access to secured areas of the web site. Once logged in, the Web Application remembers the customer until the customer logs out (either manually via Log Out button or automatically, if the session times out or a server error occurs). All secure pages in a Web Application need to check whether a customer is logged in before allowing access.

Catalog Administration:

This administrative interface is implemented for easy management of the web store data.

The catalog administration page allows the administrator to:

- Add or remove instruments, and update the details of existing instruments.
- View and manage the categories that belong to an instruments.
- Manage the list of products in a specific category, and edit product details.

- Remove a product from a category or delete the product from the catalog.
- Manage orders by updating their status.
- Manage the shopping carts by removing those which haven’t been updated by the customer in certain amount of time.
- The administration page also needs to ask for a username and password, so that only the website administrator is allowed to perform administrative tasks.

1.4 Achievements

Online Musical Instrumental Store are widely used over the world with the help of internet. In today's generation people make a large use of internet for each and everything. Online Musical Instrumental Store has achieved a large no of progress in today's world.

People are able to order any type of instrument by just sitting in their homes. Because of this, the time to travel to the actual store is saved and people find it a convenient way to order their needs. You can easily login in this website/application just by filling login form. This steps of ordering online cakes are very easy and user friendly. People are easily able to interact over internet.

1.5 Organization of Report

The project Online Musical Instrumental Store system is a web based application that allows administration to handle all the activities online quickly and safely. Using Interactive GUI anyone can quickly learn to use the complete system.

Using this, the administrator doesn't have to sit and manage the entire activities on paper, and at the same time, the head will feel comfortable to keep check of the whole system. This system will give him power and flexibility to manage the entire system from a single online portal.

The project aims to provide varieties of products to the customer. As we know that due to the busy life people don't have time to visit to a particular shop and buy the things. They find the most convenient way of shopping which is online shopping. So Online Musical Instrumental Store is one of the convenient way of providing varieties of instruments to customer just by sitting at home.

Chapter 2

SURVEY OF TECHNOLOGY

Frontend:

A front-end system is part of an information system that is directly accessed and interacted with by the user to receive or utilize back-end capabilities of the host system. It enables users to access and request the features and services of the underlying information system. The front-end system can be a software application or the combination or hardware, software and network resources.

A front-end system is primarily used to send queries and requests, and receive data from the back-end system or the host information system. It serves or provides users with the ability to interact and use an information system. Typically, front-end systems have very limited computational or business logic processing capabilities and rely on the data and functions from the host system. However, some advanced level front-end systems do maintain copies of data, such as a duplicate of each transaction sent to the back-end system.

A front-end system may include or consist of textual or graphical user interface (GUI) and/or a front-end client application that is connected by the back-end system

In Frontend we have :

HTML5

HTML5 is a W3C specification that defines the fifth major revision of the Hypertext Markup Language (HTML). One of the major changes in HTML5 is in respect to how HTML addresses Web applications. Other new features in HTML5 include specific functions for embedding graphics, audio, video, and interactive documents. New elements also allow you to define sections of your Web page using new tags such as <article> which defines an article,<nav> which defines navigation links, <source> which defines media resources, and many others. For example, the navigation section of your page would be enclosed in the <nav> tags.

CSS3

Cascading Style Sheets Level 3 (CSS3) is the iteration of the CSS standard used in the styling and formatting of Web pages. CSS3 incorporates the CSS2 standard with some changes and improvements.

Some of the major modules of CSS3 are:

- Box model
- Image values and replaced content
- Text effects
- Selectors
- Backgrounds and borders
- Animations
- User interface (UI)
- Multiple column layout
- 2D/3D transformations

Client side

Client-side refers to a specific part of client/server architecture, which is a network structure distinguishing clients or computers ordering information from servers, hardware pieces that deliver that information and process requests

In a traditional client/server structure, clients consist of physical personal computers or desktop computer stations. These use web browsers or other connections to make demands on servers. In this kind of structure, if something is client-side, that means it's run in the workstations or computers that represent clients.

Client side have:

JAVASCRIPT

JavaScript is a programming language commonly used in web development. It was original developed by Netscape as a means to add dynamic and interactive elements to websites. While JavaScript is influenced by Java, the syntax is more similar to C and is based on ECMA Script, a scripting language developed by Sun Microsystems.

JQUERY

JQuery is a concise and fast JavaScript library that can be used to simplify event handling, HTML document traversing, Ajax interactions and animation for speedy website development. JQuery simplifies the HTML's client-side scripting, thus simplifying Web 2.0 applications development.

JQuery is a free, open-source and dual-licensed library under the GNU General Public License. It is considered one of the favorite JavaScript (JS) libraries available today. As of 2012, it is used by more than half of the Web's top sites.

Server side

Occurring on the server side of a client-server system. For example, on the World Wide Web, CGI scripts are server-side applications because they run on the Web server. In contrast, JavaScript scripts are client-side because they are executed by your browser (the client). Java applets can be either server-side or side depending on which computer (the server or the client) executes them.

Server side includes:

AJAX

Ajax (Asynchronous JavaScript and XML) is a method of building interactive applications for the Web that process user requests immediately. Ajax combines several programming tools including JavaScript, dynamic HTML (DHTML), Extensible Markup Language (XML), cascading style sheets (CSS), the Document Object Model (DOM), and the Microsoft object, XML, HttpRequest.

MySQL

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software application which may run either on the same computer or on another computer across a network (including the Internet).

Chapter 3

REQUIREMENTS AND ANALYSIS

3.1 PROBLEM DEFINITION

- **TIME CONSUMPTION:**

As the Test records are manually maintained it consume a lot of time.

- **PAPER WORK:**

Lot of paper work is involved as the records are maintained in the files and registers.

- **STORAGE REQUIREMENT:**

As files and registers are used the storage space requirement is increased.

- **LESS RELIABLE:**

Use of paper for storing valuable data information is not at all reliable.

- **ACCURACY:**

As the system is in manually there are lots many chances of human errors. These causes error in calculating total payment given by Patients etc.

DIFFICULTY IN KEEPING NEW RECORDS:

It is difficult for keeping all new entries of the distributors.

This phase consists of two main tasks:

a. The first is to review the needs that originally initiated the project.

b. The second is to identify at an abstract level the expected capabilities of the new system.

It helps in understanding the system properly so that all the problems are identified correctly. It also involves considering all the alternatives that exist to achieve the objectives with respect to modifying the system, even all the various ways to implement the alternatives.

After we thoroughly understood the existing system, it was concluded that all of the work was done manually. All kinds of calculations and planning were done using the human brain instead of taking advantage of the modern Information Technology.

The following limitations were found out in the existing system:

a. Increased paper work.
b. Extended time to access the data, and to search required data.
c. Duplication of data.
d. Absence of integration data.
e. Increased Chances of Information Leakage or Loss of Information.
f. Error-prone due to manual work.
g. Lack of tools for manipulation of data.

The proposed system will overcome all problems mentioned above:

a. Storing data into database.
b. Less time required to access the data.
c. No redundancy of data.
d. Reducing manual work by doing tasks automatically.
e. Data Integration is provided.
f. Data is secured from leakage as it is accessible to authenticated user.

3.2 REQUIREMENTS SPECIFICATION

The Software Requirement Specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, an indication of performance requirement and design constraints, appropriate validation criteria, and other data pertinent to requirements.

The proposed system has the following requirements:

- System needs store information about new entry of instruments.
- System needs to keep information of Item Category and find them as per various queries.
- System need to maintain quality record.
- System need to keep the record of Cart.
- System need to update and delete the record.
- System also needs a search area.

3.3 PLANNING AND SCHEDULING

Project Planning:

Software project plan can be viewed as the following:

1. **Within the organization:** How the project is to be implemented? What are various constraints (time, cost, and staff)? What is market strategy?
2. **With respect to the customer:** Weekly or timely meetings with the customer with presentation on status report. Customer's feedback is also taken and further modification and development are done. Project milestones and deliverables are also presented to the customer.

For a successful software project, the following steps can be followed:

- Select a project
  - Identifying projects aims and objectives
  - Understanding requirements and specification
  - Method of analysis, design and implementation
  - Testing techniques
  - Documentation
- Project milestones and deliverables
- Budget allocation
  - Exceeding limits within control
- Project Estimate
  - Cost
  - Time
  - Size of code
  - Duration
- Resource Allocation
  - Hardware
  - Software
  - Previous relevant project information
  - Digital Library
- Risk Management
  - Risk avoidance
  - Risk detection

3.4 SOFTWARE AND HARDWARE REQUIREMENTS

Hardware Components:

System should be tested on different configurations & Platforms. For Optimum Performance of our Project, the requirement is as shown below

Processor (CPU Type)	Pentium III , IV or higher
RAM	4GB or higher
Disk Space	Minimum 5GB

Software Requirements:

Operating System	Windows Xp / 07 or Higher
Screen	Sublime Text3 (FRONT END)
Database Management Software	MySQL Server. (BACK END)

3.5 PRELIMINARY PRODUCT DESCRIPTION

The first step in the system development life cycle is the preliminary investigation to determine the feasibility of the system. The purpose of the preliminary investigation is to evaluate project requests. It is not a design study nor does it include the collection of details to describe the business system in all respect. Rather, it is the collecting of information that helps committee members to evaluate the merits of the project request and make an informed judgment about the feasibility of the proposed project.

Analysts working on the preliminary investigation should accomplish the following objectives:

- Clarify and understand the project request
- Determine the size of the project.
- Assess costs and benefits of alternative approaches.
- Determine the technical and operational feasibility of alternative approaches.
- Report the findings to management, with recommendations outlining the acceptance or rejection of the proposal.

Functions of the system:

- **Admin login:** The system is under admin's supervision where he can add or modify and update products information.
- **User login:** User has to first create an account to log into the system. User can access all the available instruments and shop from this registered account.
- **Products categories:** The instruments are categorized according to types like drums, guitars, flutes and so on.
- **Bill calculation:** As the user is done with product selection, a bill is generated stating the total cost incurred of the products bought.
- **Credit card payment:** The software has online payment facility where they can make payment via credit card.
- **Email notification:** The products thus bought and the respective bill receipt is emailed to the user.

3.6 CONCEPTUAL MODELS

Data Flow Diagram

Data Flow Diagram after having design the database for our project. We design the dataflow model which represents the process as a set of activities each of which carries out some data transformation. It shows how the input to the process such as specification is transformed to an output such as design. The activities here may be lower than in a workflow model. They may represent transformations carries out by people or computers.

3.6.1 Data Flow Diagram

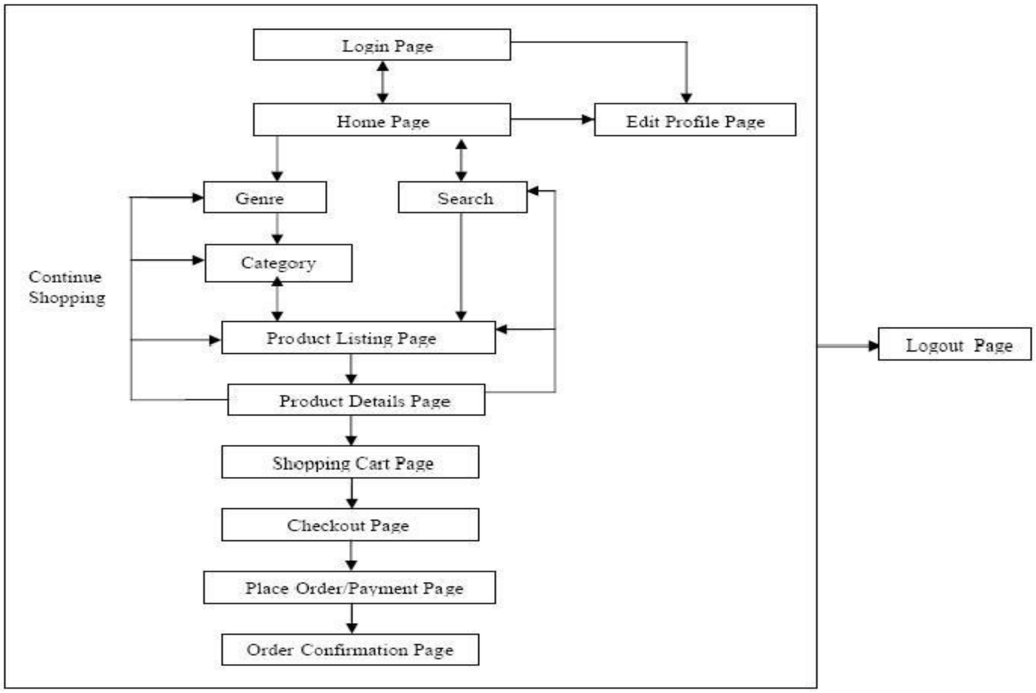


Fig 3.6.1.1 Data Flow Diagram

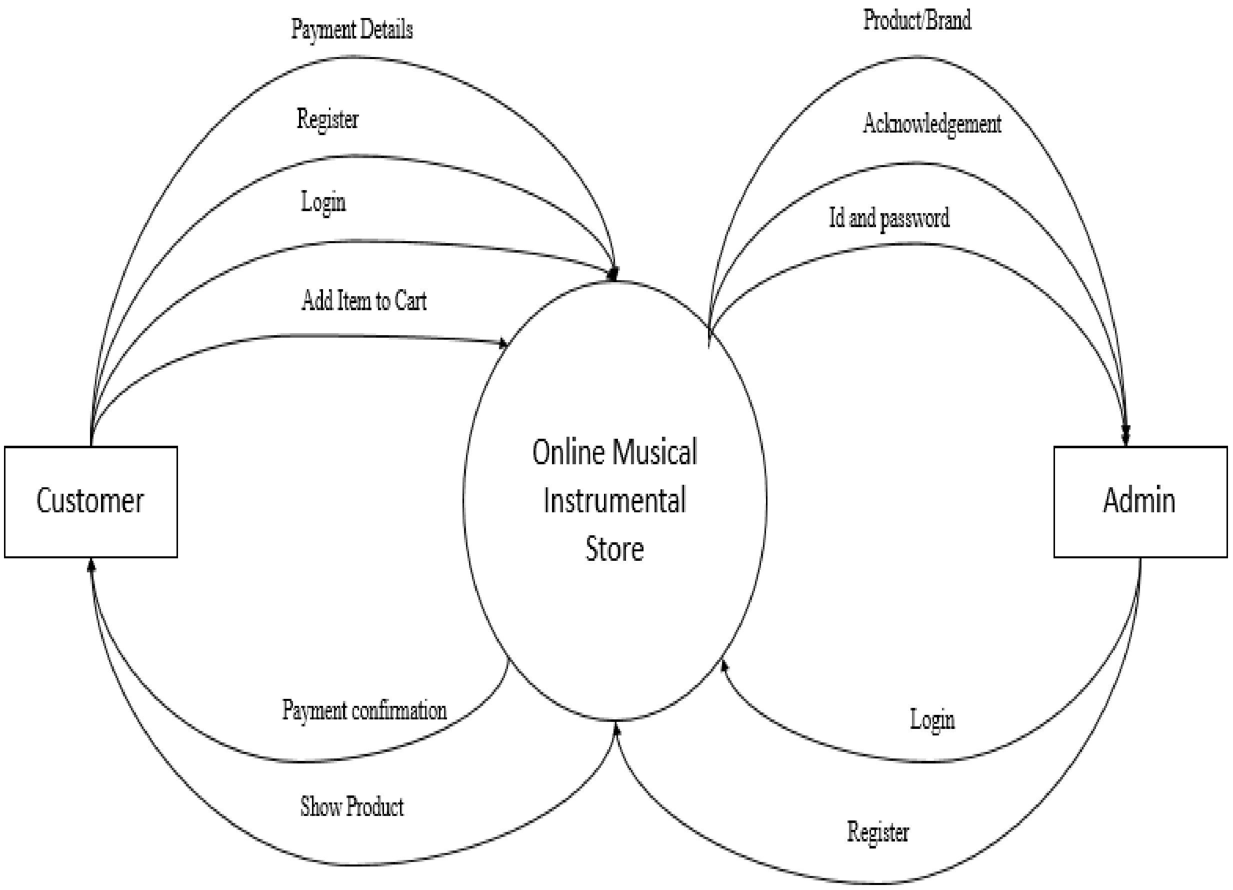
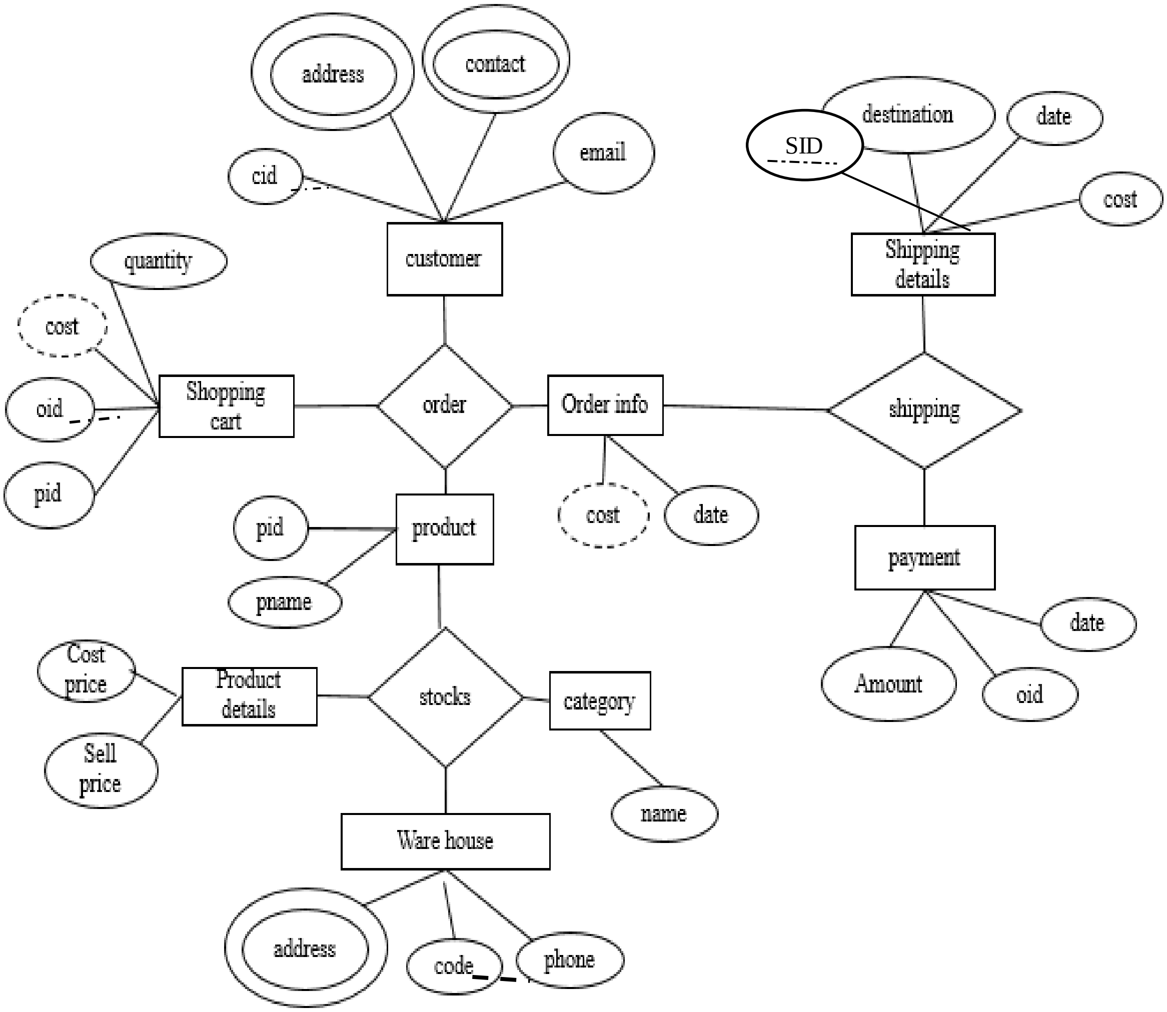


Fig 3.6.1.2 Data Flow Diagram between Admin and Customer

3.6.2 ER Diagram for Database Design

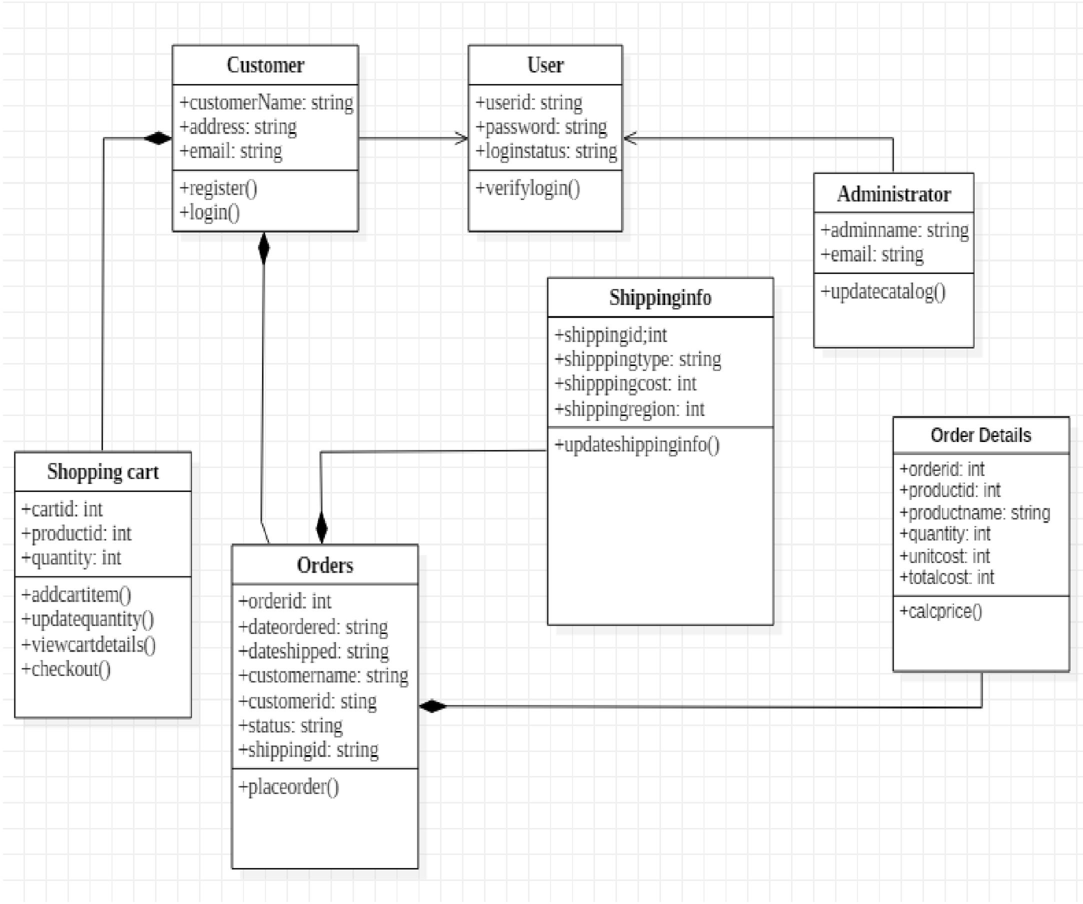
ER Diagram for Database Design After having drawn the structure diagram for our project, it is clear what kind of data should be stored in the database. Since SQL is a relational database, the EER modeling approach is very useful to design the database schema since it maps well to the relational model and the constructs used in the ER model can easily be transformed into relational tables. Here is the ER Diagram for the database of the system. Fig 3.6.3.1: E-R diagram for online Musical Instrumental Store.



3.6.2.1 E-R diagram for online Musical Instrumental Store

3.6.3 UML Class Diagram

The next step of the design phase is to draw an UML Class Diagram of the system. Since the programming language of the system is an object oriented one, an UML Class Diagram is particularly adapted to show the classes of the system, their inter- relationships, and the operations and attributes of the classes. Here is the class diagram of the project. Fig 3.3.1: UML class diagram for online Musical Instrumental Store.



3.6.3.1 Class diagram for online Musical Instrumental Store

Chapter 4

SYSTEM DESIGN

4.1 BASIC MODULES

The goal of the design phase is to transform the requirements specified in the SRS document into a structure that is suitable for implementation in some programming language. A system is simply a set of components that interact to accomplish some purpose. Systems are of two types:

- **Open Systems**
- **Closed Systems**

Systems that interact with their environments are open systems. They receive input and produce output. In contrast; systems that do not interact with their surroundings are closed systems all on going systems are open. Closed systems exist only as a concepts.

System development can generally be thought of as having two major components

- System Analysis.
- System Design.

Effective analysts emphasize investigation and questioning to learn how the system currently operates and to identify the requirements users have for a new or modified one. Only after analysts fully understand the system are they able to analyze it and assemble recommendations for system design.

Input Design

Input design is the process of converting user-originated inputs to a computer-based format. Input design is one of the most expensive phases of the operation of computerized system and is often the major problem of a system. In the project, the input design is made in various window forms with various methods

Output Design

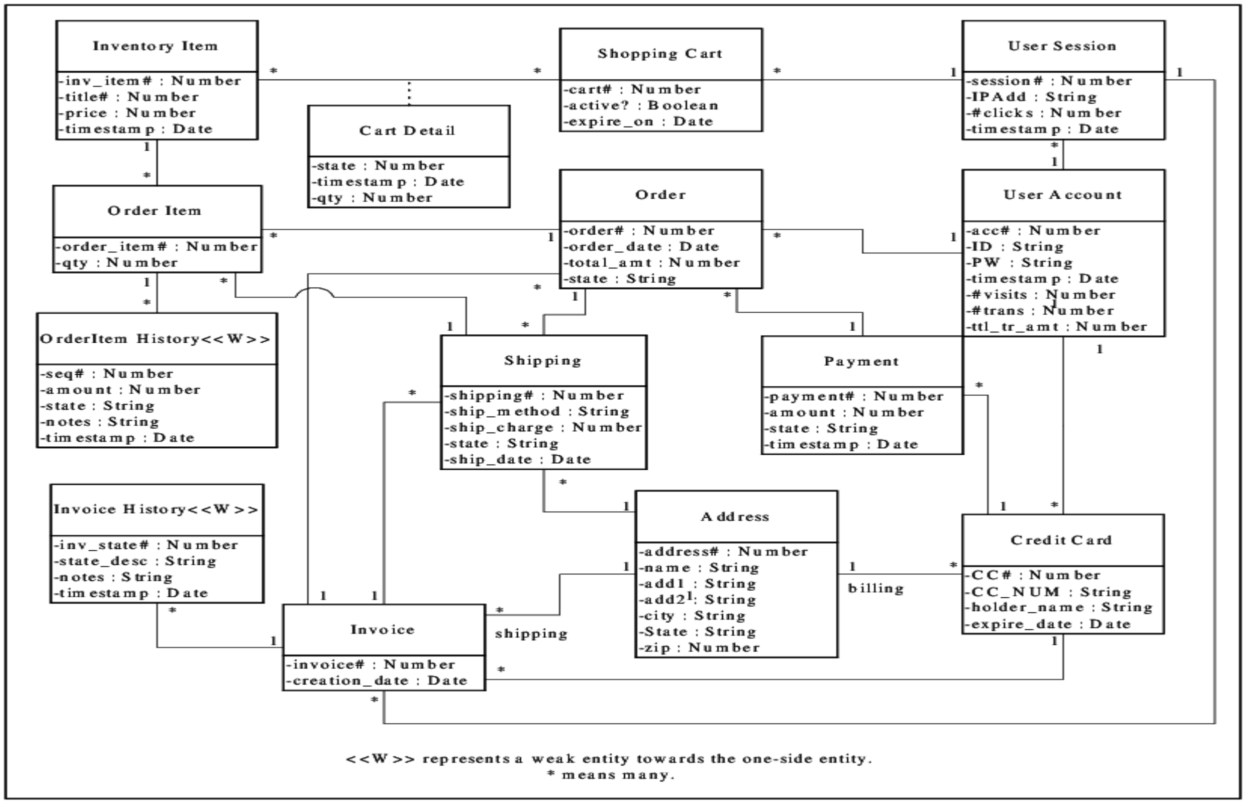
Output design generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application.

Code Design

The code design should be such that with less amount of coding we can achieve more results. The speed of the system will be more if the coding is less. Whether the data in the system is usable and readable by the system is depending on the coding. In the project, the coding is being done such that proper validations are made to get the perfect input. No error inputs are accepted. In addition care is taken such that the data integrity and referential integrity is not violated in the database. In addition, coding is designed such that concurrency avoidance of accessing the database, limited user access to the table is made perfect.

4.2 DATA DESIGN

4.2.1 Schema Design



<<W>> represents a weak entity towards the one-side entity.
* means many.

4.2.2 Data Integrity and Constraints

1. Customer:

Sr No.	Name	Type	Description
1	User ID	Varchar	Primary key for Customer identification
2	First Name	Varchar	
3	Last Name	Varchar	
4	Password	Varchar	Security for Customer
5	City	Varchar	
6	Contact no	Integer	
7	Postal code	Integer	
8	Email address	Varchar	

9	State	Varchar	
10	Address	Varchar	

2. **Product:**

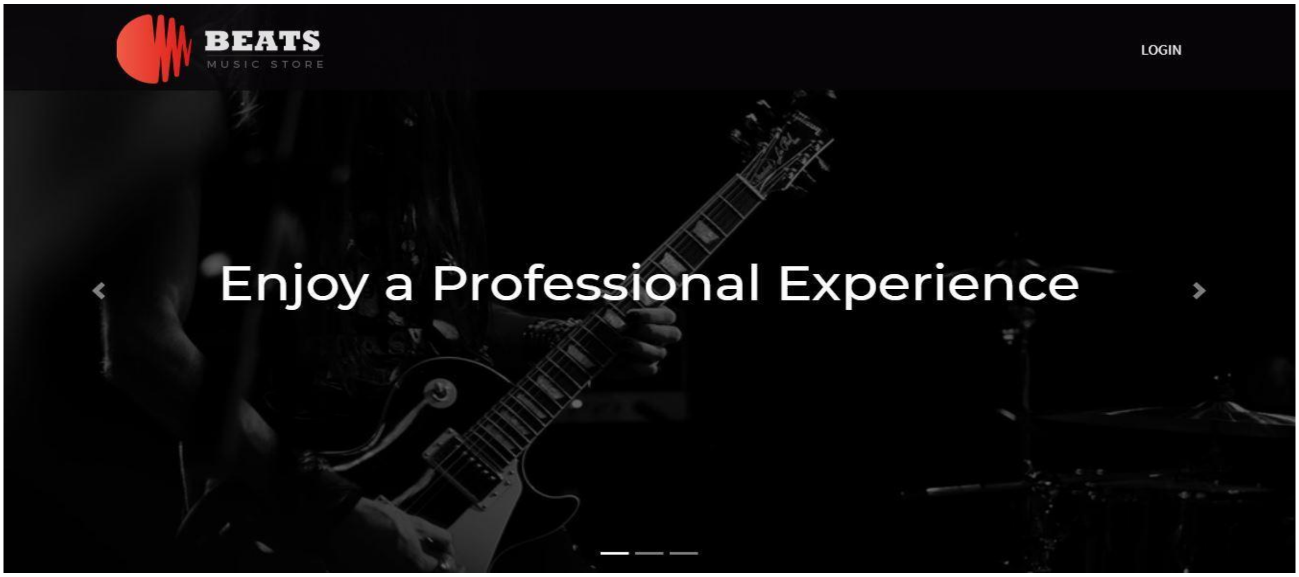
Sr No.	Name	Type	Description
1	Product ID	Integer	Primary key for Identification
2	Product Name	Varchar	
3	Price	Integer	
4	Category Id	Integer	

3. **Category**

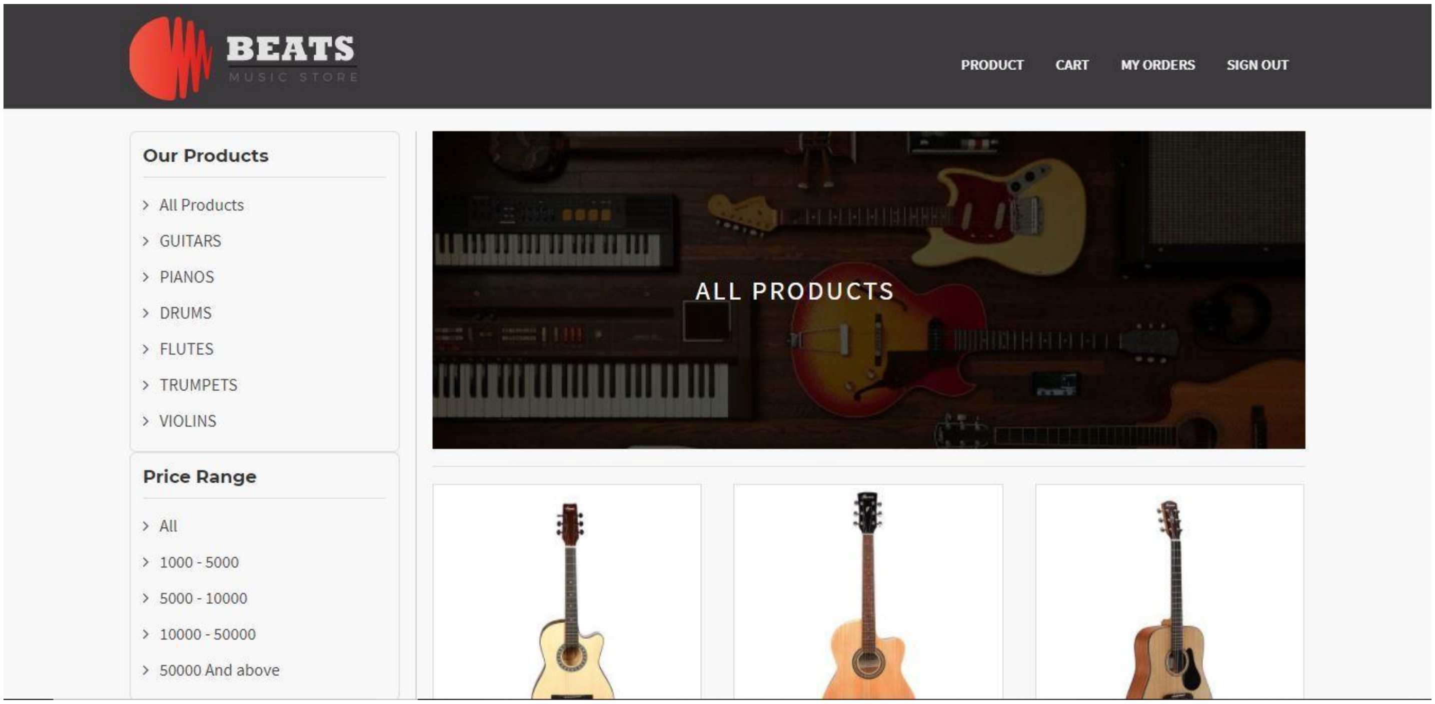
Sr No.	Name	Type	Description
1	Category ID	Integer	Primary key for Identification
2	Category Name	Varchar	
3	Description	Varchar	

4.3 USER INTERFACE DESIGN

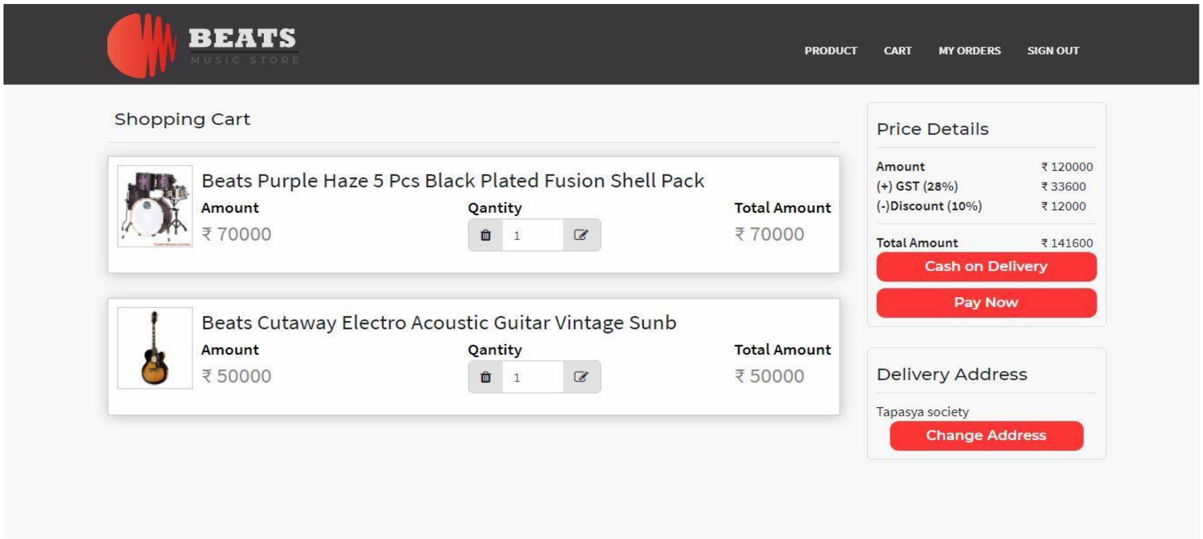
Index page



Home page



Shopping cart page



Shopping Cart

Beats Purple Haze 5 Pcs Black Plated Fusion Shell Pack

Amount	Qantity	Total Amount
₹ 70000	1	₹ 70000

Beats Cutaway Electro Acoustic Guitar Vintage Sunb

Amount	Qantity	Total Amount
₹ 50000	1	₹ 50000

Price Details

Amount	₹ 120000
(+) GST (28%)	₹ 33600
(-)Discount (10%)	₹ 12000
Total Amount	₹ 141600

Delivery Address

Tapasya society

4.4 SECURITY ISSUES

4.4.1 Security mechanisms

This system is provided with authentication, without this user can pass. So only the legitimate users are allowed to use the application. If the legitimate users share the authentication information then the system is open to outsiders.

4.4.2 Limitations

Since it is an online project, customers need internet connection to buy products.

People who are not familiar with computers can’t use this software.

Customer must have debit card or credit card to purchase products.

4.4.3 Future scope and further enhancement

This web application involves almost all the features of the online shopping. The future implementation will be online help for the customers and chatting with website administrator.

4.5 TEST CASES DESIGN

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test.

The essence of testing is to:

□ Catch as many errors as possible.

□ Correct the errors.

□ Track the errors to understand their causes and any patterns that may exist.

□ Revalidate the stability of the solutions, including ensuring that the correction of one error does not lead to introduction of another error somewhere else.

A primary purpose for this testing is to detect software failures so that defects may be uncovered and corrected. The scope of this software testing often includes examination of code as well as execution of that code in various environments and conditions as well as examining the aspects of code: does it do what it is supposed to do and do what it needs to do.

Testing is done on the following levels:

Regression Testing: Regression testing focuses on finding defects after a major code change has occurred. Specifically, it seeks to uncover software regression, or old bugs that have come back. Such regressions occur whenever software functionality that was previously working correctly stops working as intended.

Stability Testing:

Stability testing checks to see if the software can continuously function well in or above an acceptable period. This activity of non-functional software.

Usability Testing:

Usability testing is needed to check if the user interface is easy to use and understand. It approaches towards the use of the application.

Chapter 5

IMPLEMENTATION AND TESTING

5.1 IMPLEMENTATION APPROACHES

Start Planning the Site Design

Many people think this is where you jump into your web editor and start building, but the best sites start with a plan and start that plan even before the first wireframe is built.

Site Design plan includes:

- Details about the information architecture.
- The planned structure of the site.
- A site map of the pages to be designed and built.
- And technical details like if scripts or Ajax will be used, whether there will be a server-side language like PHP in use, shopping cart and so on.

Design Starts after Planning

This is where most of us start to have fun with the design phase of the project. We can jump right into editor now. But it is recommended still remain outside of it and make your design in a graphics program or even on paper first.

Must consider the following points while designing:

- Wireframes and the layout of the designs.
- Color including creating a color scheme for the site and how those colors work together in harmony.
- Also plan on a theme for site including decorative images and icons, including a site favicon.

Gather or Create the Site Content

Content is what people come to your site for. This can include text, images, and multimedia. By getting at least some of the content ready ahead of time, it will become more easy to start building the site.

Must look for:

- **Text**: This can be articles, blog posts, lists, reviews, or anything that you want to write about on my site.
- **Graphics**: There are lots of places to find images for web pages including free images. Be sure you're using the right format for your images.
- **Multimedia**: Remember that multimedia can have a negative impact on your site. Make sure that you're adding sound and video to your sites appropriately. Multimedia isn't appropriate for all target audiences.

Start Building the Site

After completion of planning and designing of the website, then building the HTML and CSS will be easier. It is the best part to work on.

Use lots of different technologies to build the site:

- HTML: This is the basis of the website.
- CSS: Once done with HTML, CSS helps to create the design planned. And CSS is easy to learn.
- JavaScript
- PHP
- Databases

Testing the Site

Testing the website is critical both throughout the building phase and after its is built. While building it, preview of your pages must be done periodically to make sure the HTML and CSS are working correctly.

Then make sure:

- The site meets the goals set out in step one.
- The technical features (HTML, CSS, scripts, and so on) work correctly. Troubleshoot the problems efficiently, and remember to validate.
- The design works in significant browsers.

5.2 CODING DETAILS AND CODE EFFICIENCY

Coding details

Efficiency is the amount of computing resources and code required by a program to perform its functions. Efficient codes are required for the better performance of the system. Efficient coding makes a system robust. A method is robust if it does not fail even if it receives improper parameters. Robustness against internal bugs may be trade off against efficiency.

The need for efficiency arises due to the cost of consideration. If some resources are scarce and expensive, it is desirable that those resources should be used efficiently. In the computer system the resources the most often considered for efficiency are processor time and less memory. This coding for the system has been started after completing the design phase, because all software methodologies emphasize the importance of first designing then coding. The codes that construct this system are structured, modular, efficient, and require less amount of computing resources. All functions of this system are understandable since the functions are small and coherent. Functions and variables are meaningful variable names to increase the readability and avoided abbreviations that may confuse the users.

In order to execute the database queries faster used the following methods. During the retrieval of data from the database using SELECT statement all the known conditions are specified in the WHERE clause.

If we want to find the maximum, minimum, sum and average value or the count of a database column, use a Select list with aggregate functions instead of computing the aggregates within the program. The RDBMS is responsible for aggregated computations instead of transferring large amount of data to the application. Overall network, application server and database load is also considerably less.

Index page

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Musical instrumental store</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<link href="../assets/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">
<link href="../assets/css/style.css" rel="stylesheet" id="bootstrap-css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/jqueryui/1.12.1/jquery-ui.css"
type="text/css" media="all" />
<script src="https://cdnjs.cloudflare.com/ajax/libs/jqueryui/1.12.1/jquery-ui.min.js"
type="text/javascript"></script>
<link href="//maxcdn.bootstrapcdn.com/font-awesome/4.2.0/css/font-awesome.min.css"
rel="stylesheet">
</head>
<body>
<!-- Top navigation -->
<nav class="navbar navbar-expand-md fixed-top top-nav  headerbg" style="border: none;">
<div class="container">
<a class="navbar-brand" href="index.php">
<img  class="img-responsive" src="../assets/images/logoremake.png"></a>
```

```
<a class="hm-login-link "  data-toggle="modal" href="#myModal">Login</a></div></nav>

<!-- Intro Banner -->

<section class="intro carousel slide bg-overlay-light h-auto" id="carouselExampleCaptions">

<ol class="carousel-indicators">

<li data-target="#carouselExampleCaptions" data-slide-to="0" class="active"></li>

<li data-target="#carouselExampleCaptions" data-slide-to="1" class=""></li>

<li data-target="#carouselExampleCaptions" data-slide-to="2" class=""></li>

</ol>

<div class="carousel-inner" role="listbox">

<div class="carousel-item active">

<img style="width: 100%;" class="d-block img-fluid" alt="First slide"
src="../assets/images/bg_7.jpg">

<div class="carousel-caption ">

<h2 class="display-4 text-white mb-2 mt-4">Enjoy a Professional Experience</h2></div></div>

<div class="carousel-item">

<img style="width: 100%;height: 100%;" class="d-block img-fluid" alt="Second slide"
src="../assets/images/bg_9.jpg">

<div class="carousel-caption ">

<h2 class="display-4 text-white mb-2 mt-4">Simple &amp; Elegant Design</h2></div></div>

<div class="carousel-item">

<img  style="width: 100%;" class="d-block img-fluid" alt="Third slide"
src="../assets/images/bg_4.jpg">

<div class="carousel-caption ">

<h2 class="display-4 text-white mb-2 mt-4">Quality &amp; Value
Assurance</h2></div></div></div>

<a class="carousel-control-prev" href="#carouselExampleCaptions" role="button" data-
slide="prev">

<span class="carousel-control-prev-icon" aria-hidden="true"></span>

<span class="sr-only">Previous</span></a>

<a class="carousel-control-next" href="#carouselExampleCaptions" role="button" data-
slide="next">
```

```html
<span class="carousel-control-next-icon" aria-hidden="true"></span>

<span class="sr-only">Next</span></a></section>

<!--End Intro Banner -->

<!-- Middle -->

<section class="info-section">

<div class="container">

<div class="head-box text-center mb-5">

<h2>Explore Our Products</h2>

<h6 class="text-underline-primary">Beats brings you a wide selection of musical instruments and equipment in India online.</h6></div>

<div class="three-panel-block mt-5">

<div class="row">

<div class="col-lg-3 col-md-6 col-sm-6" >

<a href="product.html">

<div class="service-block-overlay text-center mb-5 p-lg-3 hm-box1" >

<h3>Guitars</h3></div></a></div>

<div class="col-lg-3 col-md-6 col-sm-6">

<a href="#">

<div class="service-block-overlay text-center mb-5 p-lg-3 hm-box2">

<h3>Pianos</h3></div></a></div>

<div class="col-lg-3 col-md-6 col-sm-6">

<a href="#">

<div class="service-block-overlay text-center mb-5 p-lg-3 hm-box3">

<h3>Violins</h3></div></a></div>

<div class="col-lg-3 col-md-6 col-sm-6">

<a href="#">

<div class="service-block-overlay text-center mb-5 p-lg-3 hm-box4">

<h3>Drums</h3></div></a></div></div></div></div></section>

<!--- Footer  --->
```

```
<footer>
<div class="container">
<div class="row ">
<div class="col-md-12 pull-right ">
<ul class="social-network social-circle ">
<li><a href="#" class="icoRss" title="Rss"><i class="fa fa-rss"></i></a></li>
<li><a href="#" class="icoFacebook" title="Facebook"><i class="fa fa-facebook"></i></a></li>
<li><a href="#" class="icoTwitter" title="Twitter"><i class="fa fa-twitter"></i></a></li>
<li><a href="#" class="icoGoogle" title="Google +"><i class="fa fa-google-plus"></i></a></li>
<li><a href="#" class="icoLinkedin" title="Linkedin"><i class="fa fa-linkedin"></i></a></li>
</ul></div>
<div class=" col-md-12 text-center ">
Copyright &copy; 2019 reserved by Mayuresh.
</div></div></div></footer>
</body></html>
<!-- Middle -->
<!-- footer start -->
<!-- footer ends -->
<!--- Model Popup Box --->
<script src="../assets/js/jquery.min.js"></script>
<script src="../assets/js/bootstrap.min.js"></script>
<script src="../assets/js/main.js"></script>
<script src="../assets/js/index.js"></script>
</body></html>
```

Registration page

```
<form id="register-form" name="register-form" method="post" role="form" style="display: none;">
<div class="modal-dialog">
```

```
<div class="modal-content ">

<!-- Modal Header -->

<div class="modal-header">

<h4 class="modal-title">Sign Up</h4>

<button type="button" class="close" data-dismiss="modal">&times;</button></div>

<!-- Modal body -->

<div class="modal-body">

<div class="panel">

<div class="panel-body">

<div class="row">

<div class="col-lg-12">

<div class="form-group">

<input type="text" name="name" id="name" tabindex="1" class="form-control"
placeholder="Full Name" value=""></div>

<div class="form-group">

<input type="email" name="email" id="email" tabindex="2" class="form-control"
placeholder="Email Address" value=""></div>

<div class="form-group">

<input type="text" name="phoneno" id="phoneno" tabindex="3" class="form-control"
placeholder="Phone Number" value=""></div>

<div class="form-group">

<input type="text" name="address" id="address" tabindex="4" class="form-control"
placeholder="Address" value=""></div>

<div class="form-group">

<input type="text" name="username" id="username" tabindex="5" class="form-control"
placeholder="Username" value=""></div>

<div class="form-group">

<input type="password" name="password" id="password" tabindex="6" class="form-control"
placeholder="Password"></div>

<div class="form-group">
```

```
<input type="password" name="cpassword" id="cpassword" tabindex="7" class="form-control" placeholder="Confirm Password"></div>

<div class="form-group text-center">

<input type="button" name="btnregister" id="btnregister" tabindex="4" class="form-control btn btn-success" value="Register Now">

<div class="text-danger" id="err2"></div></div>

<div class="form-group text-center">

<a href="#" class="model-link" id="login-form-link">Login</a></div>

</div></div></div></div></div></div>

</form></div>

<!--- Model Popup Box --->
```

Login page

```
<div class="modal" id="myModal">

<form id="login-form" name="login-form" method="post" role="form" style="display: block;">

<div class="modal-dialog">

<div class="modal-content ">

<!-- Modal Header -->

<div class="modal-header">

<h4 class="modal-title">Login</h4>

<button type="button" class="close" data-dismiss="modal">&times;</button></div>

<!-- Modal body -->

<div class="modal-body">

<div class="panel">

<div class="panel-body">

<div class="row">

<div class="col-lg-12">

<div class="form-group">
```

```
<input type="text" name="username" id="username" tabindex="1" class="form-control" placeholder="Username" value=""></div>

<div class="form-group">

<input type="password" name="password" id="password" tabindex="2" class="form-control" placeholder="Password"></div>

<div class="form-group  text-center">

<button type="button" name="btnlogin" id="btnlogin" tabindex="3" class="form-control btn btn-success">Log in</button></div>

<div class="form-group text-danger text-center" id="err1"></div>

<div class="form-group text-center">

<a href="#" class="model-link" id="register-form-link">Register</a></div></div></div></div>

</div></div></div></div></form>
```

Product page

```
<?php

require_once "header.php";

?>

<section  class="middle-container" >

<div class="container">

<div class="row">

<div class="col-md-3">

<div class="cat-list">

<h3>Our Products</h3>

<ul>

<li><a href="product.php">All Products</a></li>

<?php

$rows = $objproject->select("id, category", "categorymaster", "1=1");

// pre($rows); //for viewing the data in rows

$rows = json_decode($rows, true);
```

```
// pre($rows); //for viewing the data in rows

foreach ($rows as $row)  { ?>

<li><a href="product.php?catid=<?php echo $row['id']; ?>&catname=<?php echo
$row['category'];?>"><?php echo $row["category"]; ?></a></li>

<?php } ?>

</ul></div>

<div class="cat-list">

<h3>Price Range</h3>

<ul><li><a href="product.php?price=0 and 10000000">All</a></li>

<li><a href="product.php?price=1000 and 5000">1000 - 5000</a></li>

<li><a href="product.php?price=5000 and 10000">5000 - 10000</a></li>

<li><a href="product.php?price=10000 and 50000">10000 - 50000</a></li>

<li><a href="product.php?price=50000 and 10000000">50000 And above</a></li>

</ul></div></div>

<div class="col-md-9 cat-products " style="overflow: hidden;">

<?php

$catname = "All Products";

if(isset($_GET['catname']))

$catname = $_GET['catname'];?>

<div class="block-30 block-30-sm item  " style="background-image: url('../../assets/images/<?php
echo $catname; ?>.jpg');" data-stellar-background-ratio="0.5">

<div class="container">

<div class="row align-items-center ">

<div class="col-md-10 text-center  " >

<span class="subheading-sm  ">

<?php

echo "$catname";?>

 </span></div></div></div></div></h3><hr/>

<div class="row">
```

```
<?php

$condition = "1=1";

if(isset($_GET['catid']))

$condition = "caregoryid = " . $_GET['catid'];

if(isset($_GET['price']))

$condition = "price between " . $_GET['price'];

$rows = $objproject -> select("*", "productmaster", $condition);

$rows = json_decode($rows, true);

// pre($rows);

if($rows == 0)

{

echo "<h2>No Products Found</h2>";

}

else

foreach ($rows as $row)

{ ? >

<!-- start -->

<div class="col-md-4 col-sm-6" >

<div class="product-grid " >

<!-- <div class="spinner-border text-danger"></div> -->

<div class="product-image">

<a><img class="pic-1"  src="<?php echo $row['img']; ?>"> </a>

<ul class="social">

<li><a href="addqty.php?prodid=<?php echo $row['id'];?>" data-tip="Add to Cart"><i class="fa
fa-shopping-cart text-center" style="font-size: 28px;"></i></a></li>

</ul></div>

<div class="product-content">

<h3 class="title"><a href="addqty.php?prodid=<?php echo $row['id'];?>"><?php echo
$row['name']; ?></a></h3>
<h6 class="title"> Type: <?php echo $row['type'];?> </h6>
<div class="price"> <i class="fa fa-inr"></i> <?php echo $row['price'] - ($row['price'] * 0.1); ?>
<span> <i class="fa fa-inr"></i> <?php echo $row['price']; ?></span></div>
<a class="add-to-cart" href="addqty.php?prodid=<?php echo $row['id'];?>">+ Add To Cart</a>
</div></div></div>
<!-- end -->
<?php } ?>
</div></div></div></div></section>
<!-- Middle -->
<?phpre("footer"); ?>
```

5.2.1 Code Efficiency

- The code repeating again was made as a module which is common for all
- Proper indentation is given to understand the code
- The forms are designed in a way that all the data and buttons are properly viewed and spaced. User can easily see and understand.
- Any person can use it easily as it is user friendly
- This code also gives a flicker free experience with the use of Ajax technology.
- All the validations on the input fields are performed through Ajax which prevents the page from refreshing to check validations on every click on submit button.

Code Optimization

1. Modulo and division operation take a lot of time and they should be replaced by something else.
2. Try to analyze the problem and obtain an alternate representation of the problem.
3. Try to eliminate the IF statements from your code in the case that their only purpose is to set some values based on a condition.

Helper module is created to for the code repeated which is common for all.

Helper.php

```
<?php
```

```
require_once 'connect.php';

class helper extends connect

{

   function update($table, $collist, $condition)

   {

      $sql= "update $table set $collist where $condition";

      return $this->conn->query($sql);

   }

   function delete($table,$condition)

   {

      $sql= "delete from $table where $condition";

      return $this->conn->query($sql);

   }

   function insert($table,$collist,$valuelist)

   {

         $sql= "insert into $table ($collist) values ($valuelist)";

         return $this->conn->query($sql);

   }

   function select($columns,$table,$condition)

   {

         $sql="Select $columns from $table where $condition";

         $answer=$this->conn->query($sql);

         if($answer->num_rows==0)

         {
```

```
                    return 0;

        }

        else

        {

                    while($ans=$answer->fetch_array(1))

                    {

                                $rows[]=$ans;

                    }

        }

        return json_encode($rows);

}

function dropdownlist($id, $collist, $table, $condition, $class)

{

        echo "<select name='$id' id='$id' class='$class'>";

        echo "<option value='-1'>Select</option>";

      $rows = $this->select($collist, $table, $condition);

    //pre($rows);

    $rows = json_decode($rows, true);

    //pre($rows);

    foreach ($rows as $row) {

      //pre($row);

        $row = array_values($row);

        // pre($row);

     echo "<option value='{$row[0]}'>{$row[1]}</option>";
```

```
        }
      echo "</select>";
   }
}?>
```

5.3 TESTING APPROACH

Software testing is a process which is used to measure the quality of software developed. It is also a process of uncovering errors in a program and makes it a feasible task. It is useful process of executing program with the intent of finding bugs.

In order to prove that a piece of software works, the software must be tested to determine if the requirements of the application are met. There are several different types of tests used throughout the development process.

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation.

5.3.1 Unit Testing

Unit Testing is defined as a type of software testing where individual units/ components of a software are tested. Unit testing of software applications is done during the development (coding) of an application. The objective of Unit Testing is to isolate a section of code and verify its correctness. In procedural programming, a unit may be an individual function or procedure. Unit testing is usually performed by the developer.

Sometimes software developers attempt to save time by doing minimal unit testing. This is a myth because skipping on unit testing leads to higher Defect fixing costs during System Testing, Integration Testing and even Beta Testing after the application is completed. Proper unit testing done during the development stage saves both time and money in the end. Here, are key reasons to perform unit testing.

STUB	LEVEL	ACTUAL VALUE	EXPECTED VALUE	DEFECT	COMMENT
$("#btnlogin").click(function(**{** **data = $("#login-form").serialize(),** **$("#err1").html("Login is successful.... ")**	Login.php	Login is successful	Login is successful	0	Data is submited
<?php session_start(); $username = $_SESSION["username"]; if($username == "admin") header("location: admin/category.php");?>	Redirect.php	Open a admin/ category.php page	Open a admin/ category.php page	0	Give a direction one page to another

public $conn="";
function __construct()
{$this->conn = new mysqli(self::HOST1, self::USER1, self::PWD1,self::DB1);}
	Connect.php	Successful Connection to Data base	Successful Connection to Data base	0	Connected to DB

function __destruct()
{$this->conn->close();
echo "DB Disconnected";
}
	Connect.php	Close the data base connection	Close the data base connection	0	Connection is Closed

function insert($table,$collist,$valuelist)	Helper.php	Could not able to insert a value	Data inserted Successfully	1	0 rows effected

Boundary value testing

Boundary Value Testing is the process of testing between extreme ends or boundaries between partitions of the input values.

- So these extreme ends like Start- End, Lower- Upper, Maximum-Minimum, Just Inside-Just Outside values are called boundary values and the testing is called "boundary testing".
- The basic idea in boundary value testing is to select input variable values at their:

1. Just below the minimum
2. Minimum
3. Just above the minimum
4. A nominal value
5. Just above the Maximum
6. Maximum
7. Just below the maximum

Boundary value test for shopping cart:

MIN+	2
Min	1
Min-	0
Nominal	5

Max+	11
Max	10
Max-	9

Cart value	Expected output	Actual output	Defects	comments	recommendation
2	Product added	Product added	None	Product added	None
1	Product added	Product added	None	Product added	None
0	Product not added	Product not added	None	Minimum quantity is 1	Minimum 1 product must be added
5	Product added	Product added	None	Product added	None
11	Product not added	Product not added	None	Limit of the quantity exceeds	Maximum 10 products at a time
10	Product added	Product added	None	Product added	None
9	Product added	Product added	None	Product added	None

5.3.2 Integrated Testing

Integration Testing is the phase in software testing in which individual software modules are combined and tested as a group.

A typical software project consists of multiple software modules, coded by different programmers. Integration Testing focuses on checking data communication amongst these modules.

Hence it is also termed as **'I & T'** (Integration and Testing), **'String Testing'** and sometimes 'Thread Testing'.

Login

STUB	LEVEL	ACTUAL VALUE	EXPECTED VALUE	DEFECT	COMMENT

setTimeout(function()
{
if(result.match('1'))
window.location.href = "redirect.php";
else
$("#err1").html(result)
}, 1000)
	Login.php	Redirect to product.php page.	Redirect to product.php page
			OR
			Show the Invalid username and password message
				0	----------

5.3.3 Beta Testing

Beta Testing is one of the Acceptance Testing types, which adds value to the product as the end-user (intended real user) validates the product for functionality, usability, reliability, and compatibility.

Inputs provided by the end-users helps in enhancing the quality of the product further and leads to its success. This also helps in decision making to invest further in the future products or the same product for improvisation.

No	User Acceptance	ACTUAL VALUE	EXPECTED VALUE	DEFECT	Comments
1	Register user data in data base	Data Registered Successfully....	Data Registered Successfully....	0	Registration is Done successfully
2	Access a user data using login form	Invalid Username and Password	Login successfully	1	User login using data
3	Project run successfully	No error message, open a Home page	Opening of Home page of project	0	Project is open successfully

5.4 MODIFICATIONS AND IMPROVEMENTS

- Zooming of the product image can be done by hovering a mouse over the image before adding it to the cart.
- User friendly interface to add or remove the product from the shopping cart.
- Sorting can be done with the categories and the price range of the products.
- Payment module is added for online payment.
- OTP is sent to the user’s mobile for verifying the mobile number for COD.
- Attractive color combinations of the GUI makes user to shop effortlessly without straining the eyes and searching for the modules.

5.5 TEST CASES

SR No	Form Name	Test Condition	Step or Procedure	Input Test Data	Expected Result	Actual Output	Pass/ Fail
1	Login	Check login with valid input	Username with Wrong password	User name: admin Password: Admin	Display Message: “Invalid Username or Password”	Display Message “Invalid Username or Password”	Pass
2	Login	Check login with valid input	Wrong Username with correct password	User name: ADMIN Password: Admin	Display Message: “Invalid Username or Password”	Display Message “Invalid Username or Password”	Pass
3	Registration	Check Alphabetic Values	If Numbers Are Inserted	Name: maya123	Display Message: “only Characters are allowed”	Display Message: “Only Characters are allowed”	Pass
4	Registration	Check Numeric Value	If Alphabets Are Inserted	Salary	Display Message: “only digits are allowed”	Display Message: “only digits are allowed”	Pass
5	Registration	Check Phone Number	If Phone Number Is More Than 10 Digit	9773566021	Display Message: “Enter 10 digit number only”.	Display Message: “Enter 10 digit number only”.”	Pass

6	Registration	Check Phone Number	If Phone Number Is Less Than 10 Digit	8200776609	Display Message: “Phone number cannot be less than 10 digit”.	Display Message: “Phone number cannot be less than 10 digit”.	Pass

Chapter 6

RESULTS AND DISCUSSION

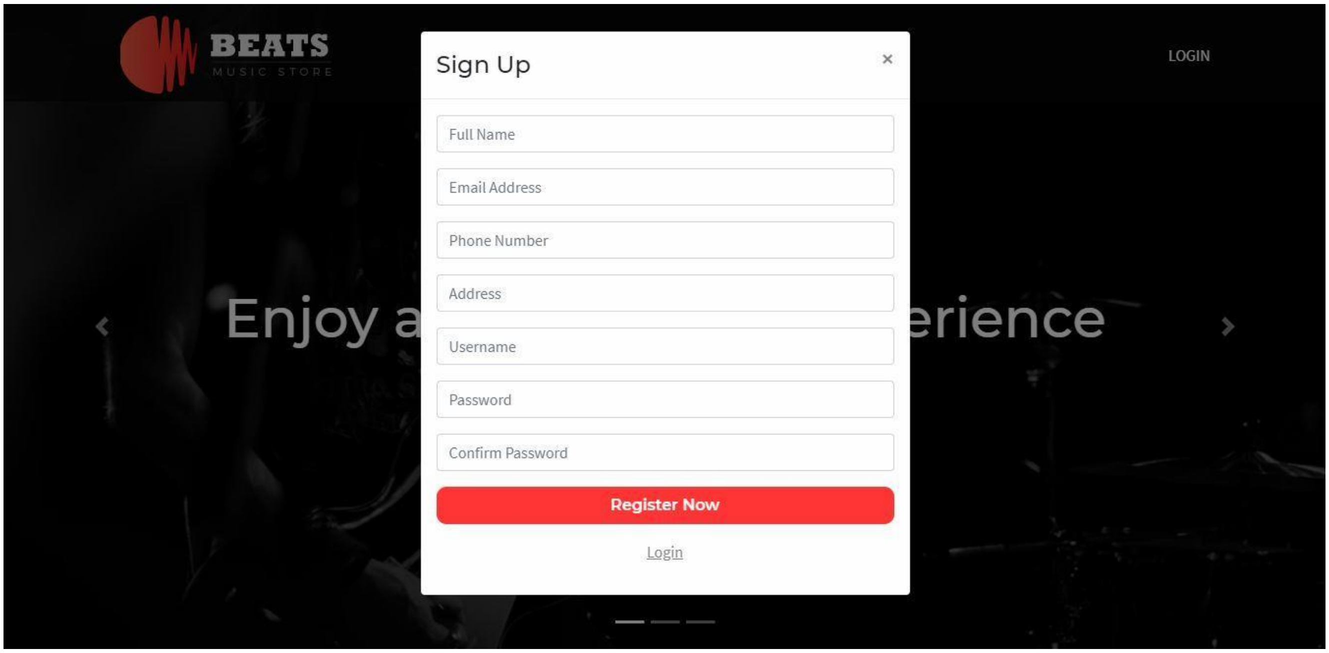
6.1 TEST REPORTS

Test case	Description	Actual output	Expected output	Result	Comment
Login page	Validate users details and redirect	Valid details, redirected successfully	Valid details, redirected successfully	0	-----
Registration page	Validate inputs and store it in database	Valid inputs, redirected successfully	Valid inputs, redirected successfully	0	-----
Add to cart	Add the products in the shopping cart	Specified quantity added successfully	Specified quantity added successfully	0	-----
Remove from cart	Delete the product from the cart	Specified quantity removed successfully	Specified quantity removed successfully	0	-----
Payment page	Accept payment from the users debit card	Money debited from users account and credited to company's account	Money debited from users account and credited to company's account	0	-----

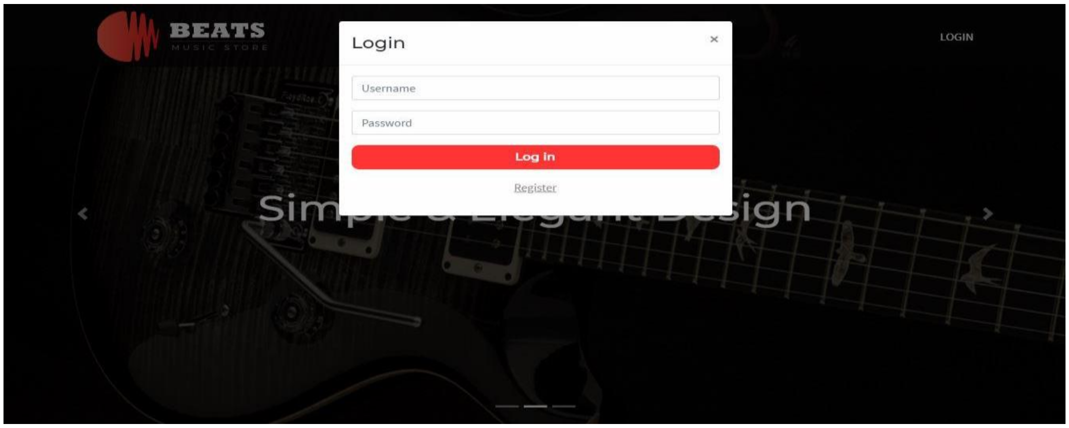
6.2 USER DOCUMENTATION

The project Online Musical Instrumental Store system is a web based application that allows administration to handle all the activities online quickly and safely. Using Interactive GUI anyone can quickly learn to use the complete system.

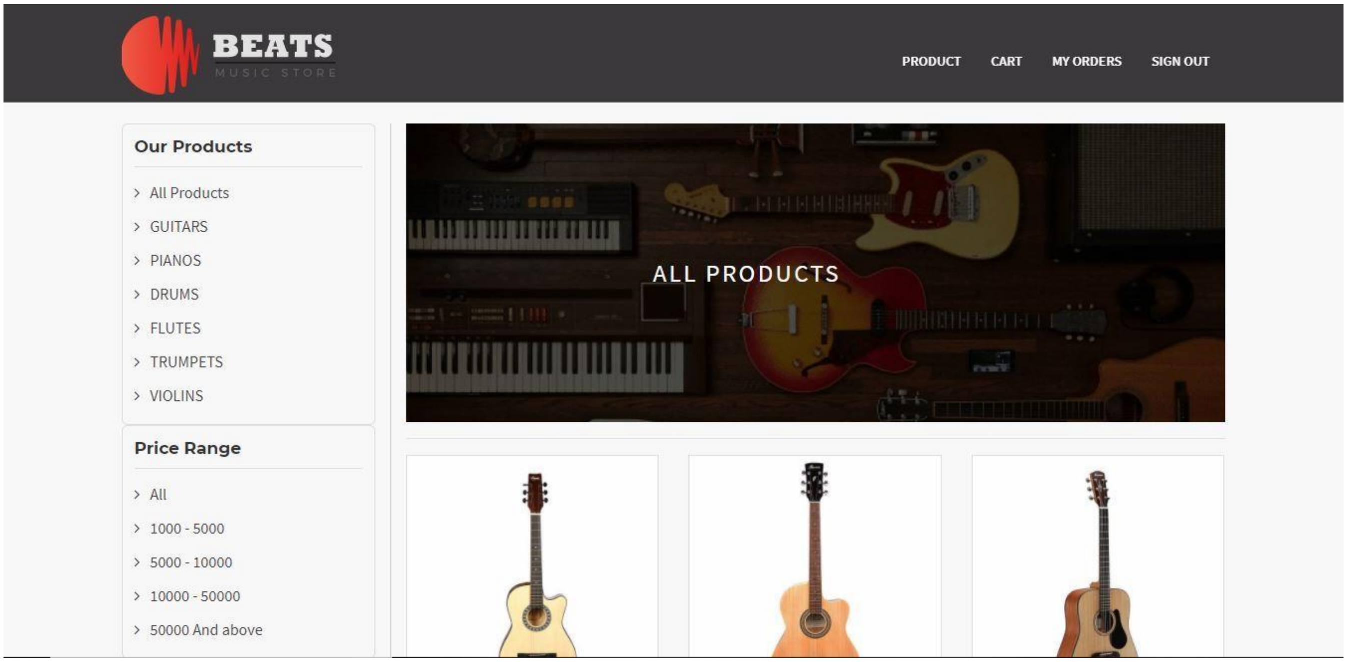
Registeration page:



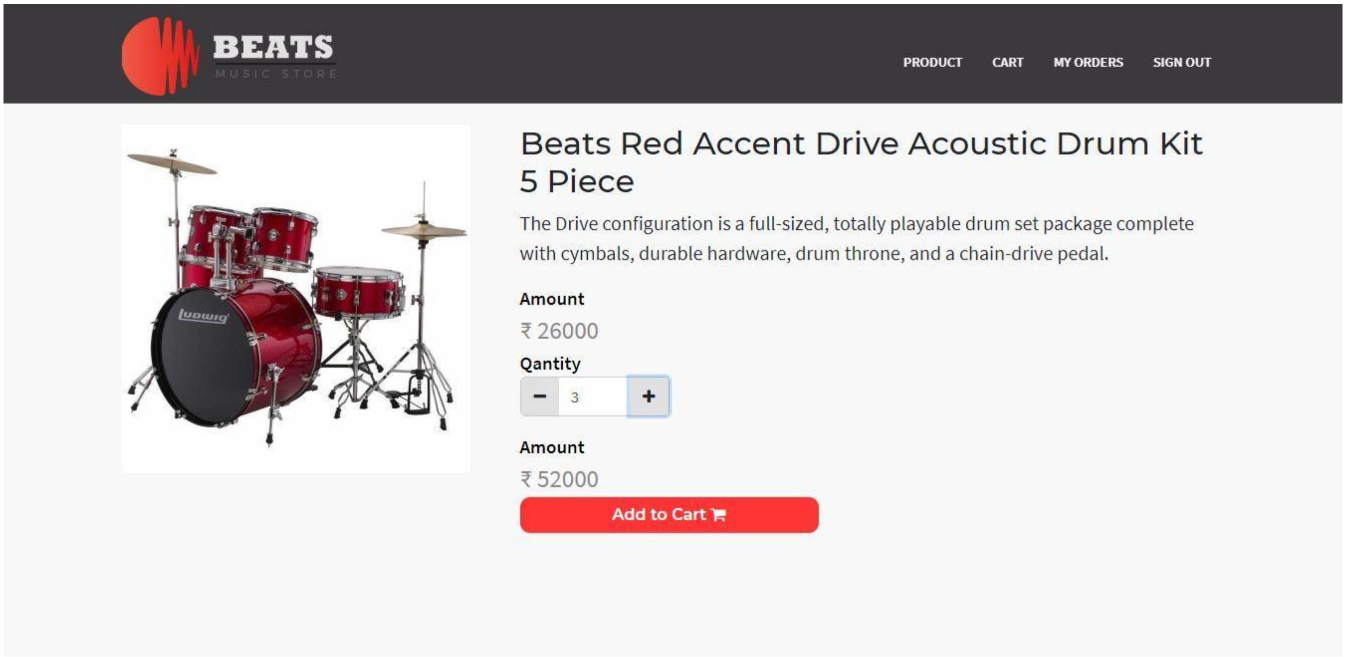
Login page:



Product page:



Add to cart page:



Beats Red Accent Drive Acoustic Drum Kit 5 Piece

The Drive configuration is a full-sized, totally playable drum set package complete with cymbals, durable hardware, drum throne, and a chain-drive pedal.

Amount

₹ 26000

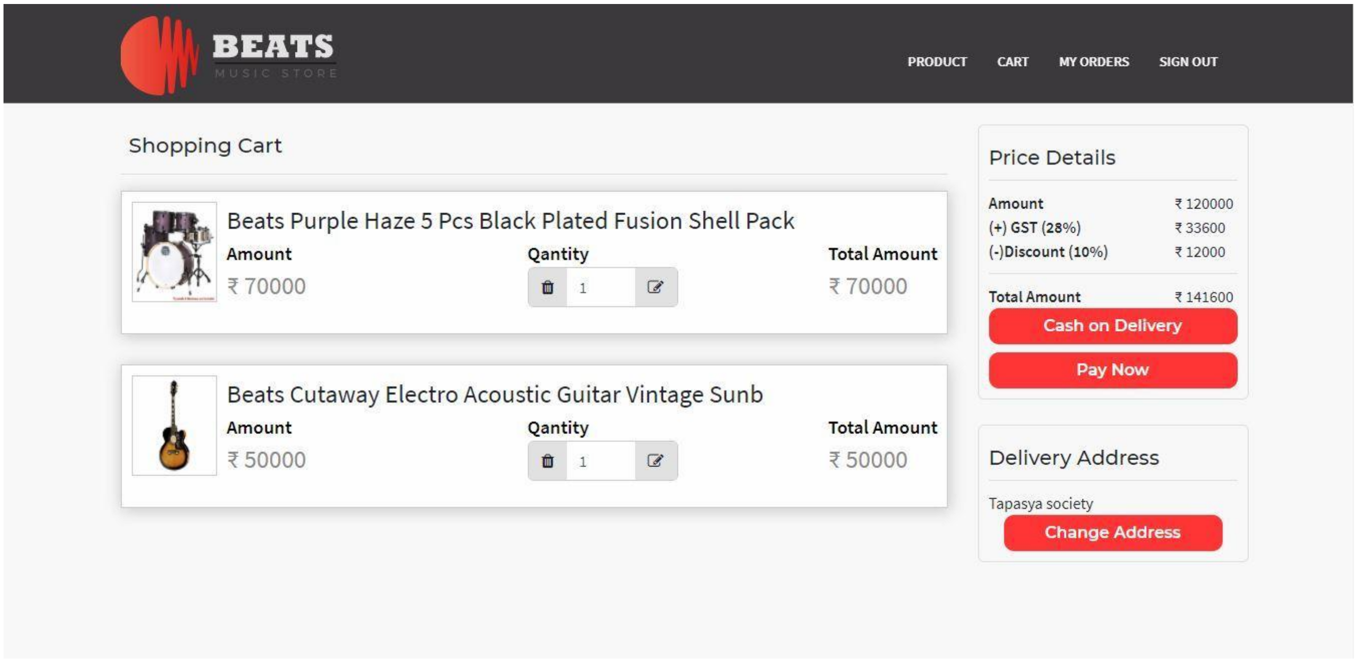
Qantity

− 3 +

Amount

₹ 52000

Cart page:



Shopping Cart

Beats Purple Haze 5 Pcs Black Plated Fusion Shell Pack

Amount	Qantity	Total Amount
₹ 70000	1	₹ 70000

Beats Cutaway Electro Acoustic Guitar Vintage Sunb

Amount	Qantity	Total Amount
₹ 50000	1	₹ 50000

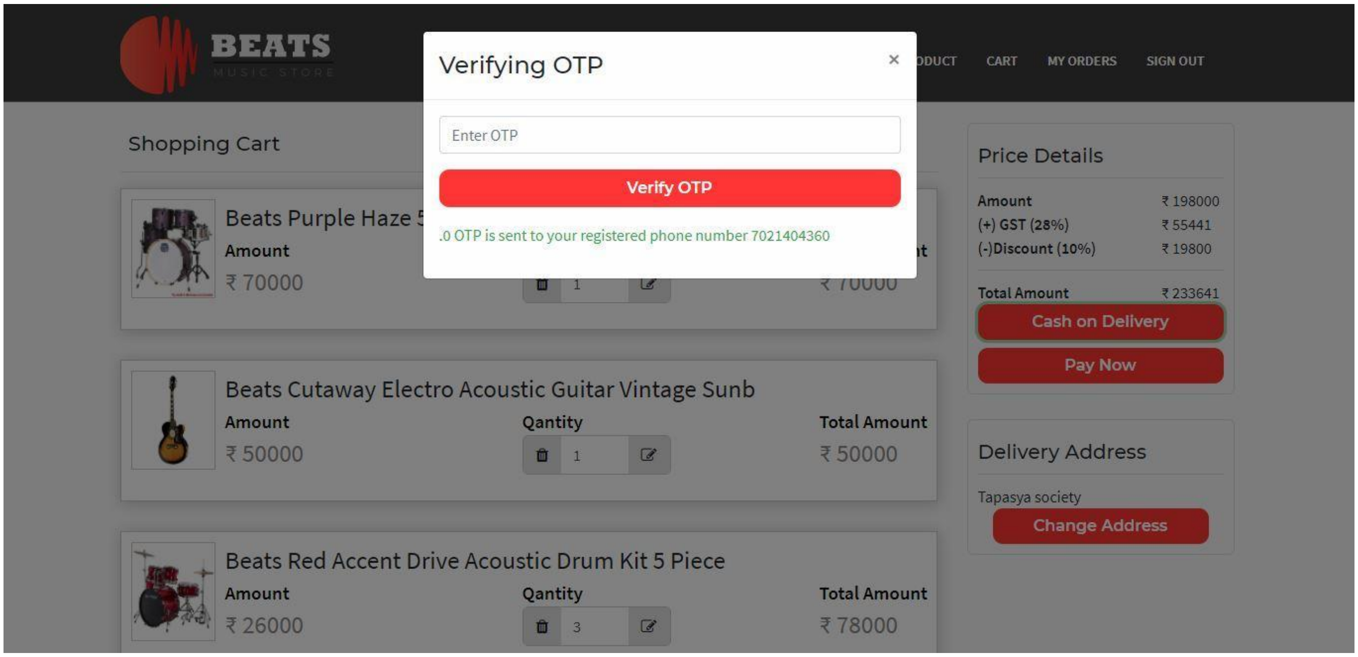
Price Details

Amount	₹ 120000
(+) GST (28%)	₹ 33600
(-)Discount (10%)	₹ 12000
Total Amount	₹ 141600

Delivery Address

Tapasya society

Cash on Delivery page:



Verifying OTP

Enter OTP

.0 OTP is sent to your registered phone number 7021404360

Shopping Cart

Beats Purple Haze 5

Amount

₹ 70000

1

₹ 70000

Beats Cutaway Electro Acoustic Guitar Vintage Sunb

Amount	Qantity	Total Amount
₹ 50000	1	₹ 50000

Beats Red Accent Drive Acoustic Drum Kit 5 Piece

Amount	Qantity	Total Amount
₹ 26000	3	₹ 78000

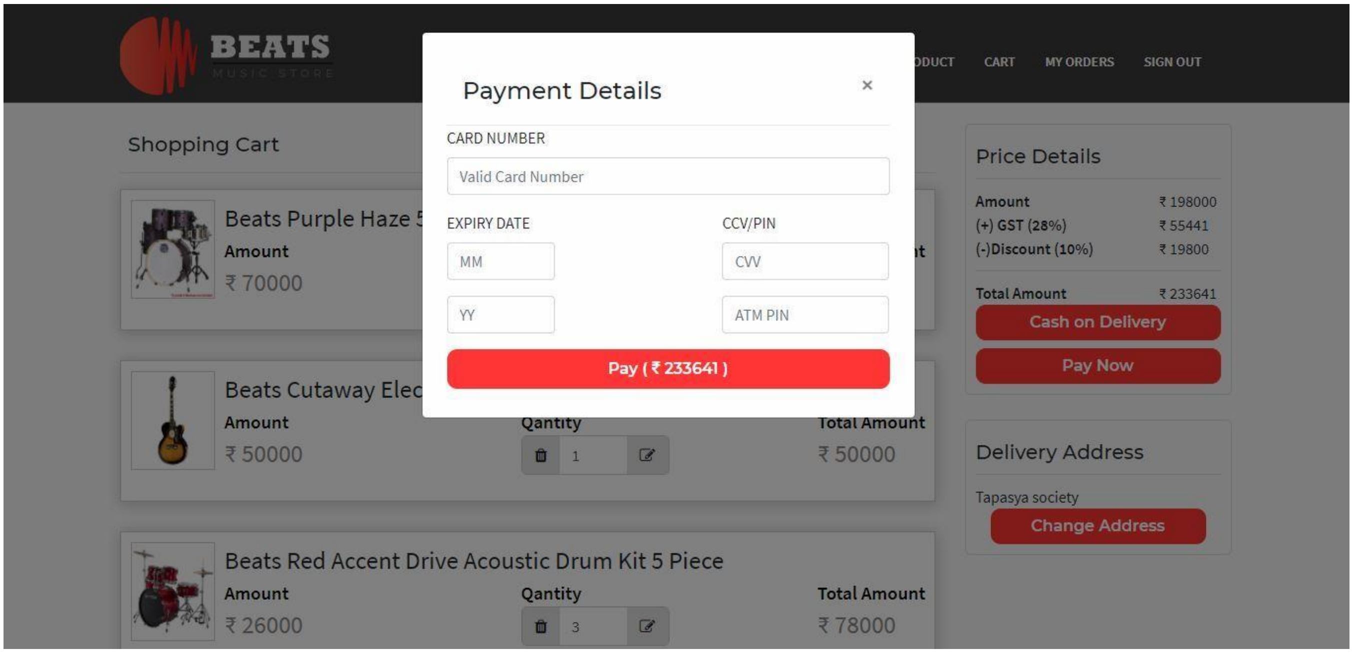
Price Details

Amount	₹ 198000
(+) GST (28%)	₹ 55441
(-)Discount (10%)	₹ 19800
Total Amount	₹ 233641

Delivery Address

Tapasya society

Online payment page:



Payment Details

CARD NUMBER

Valid Card Number

EXPIRY DATE

MM

YY

CCV/PIN

CVV

ATM PIN

Shopping Cart

Beats Purple Haze 5

Amount

₹ 70000

Beats Cutaway Elec

Amount	Qantity	Total Amount
₹ 50000	1	₹ 50000

Beats Red Accent Drive Acoustic Drum Kit 5 Piece

Amount	Qantity	Total Amount
₹ 26000	3	₹ 78000

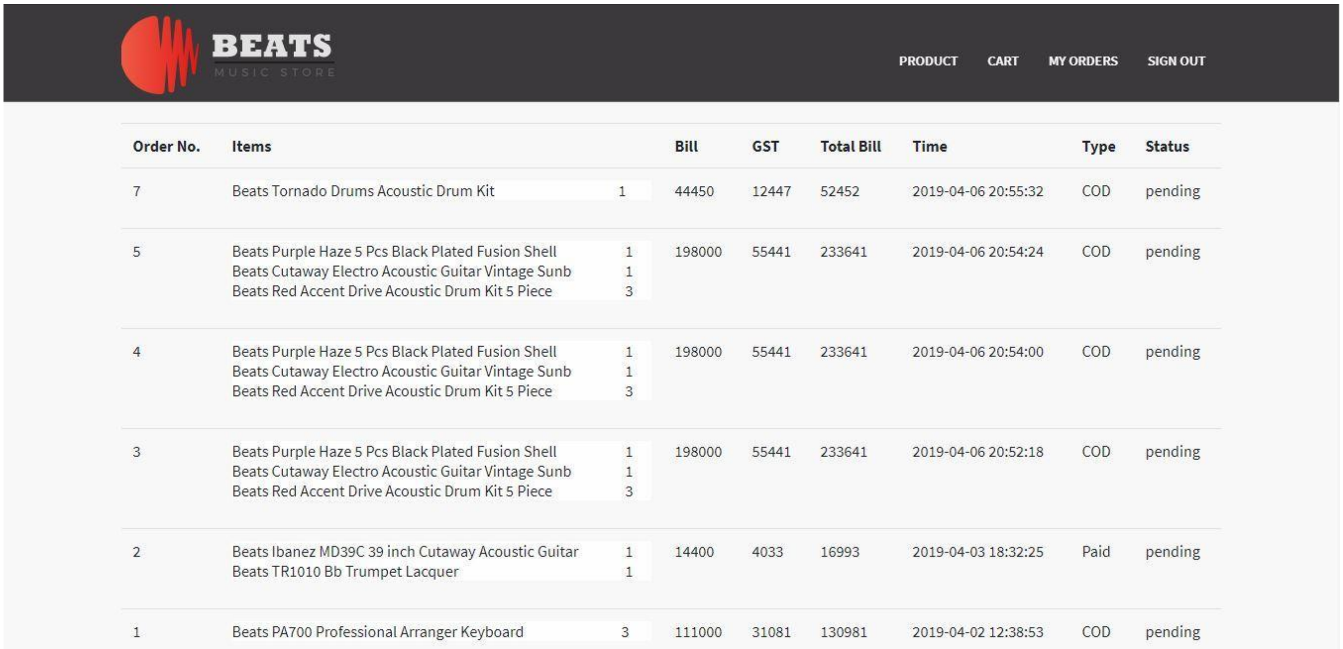
Price Details

Amount	₹ 198000
(+) GST (28%)	₹ 55441
(-)Discount (10%)	₹ 19800
Total Amount	₹ 233641

Delivery Address

Tapasya society

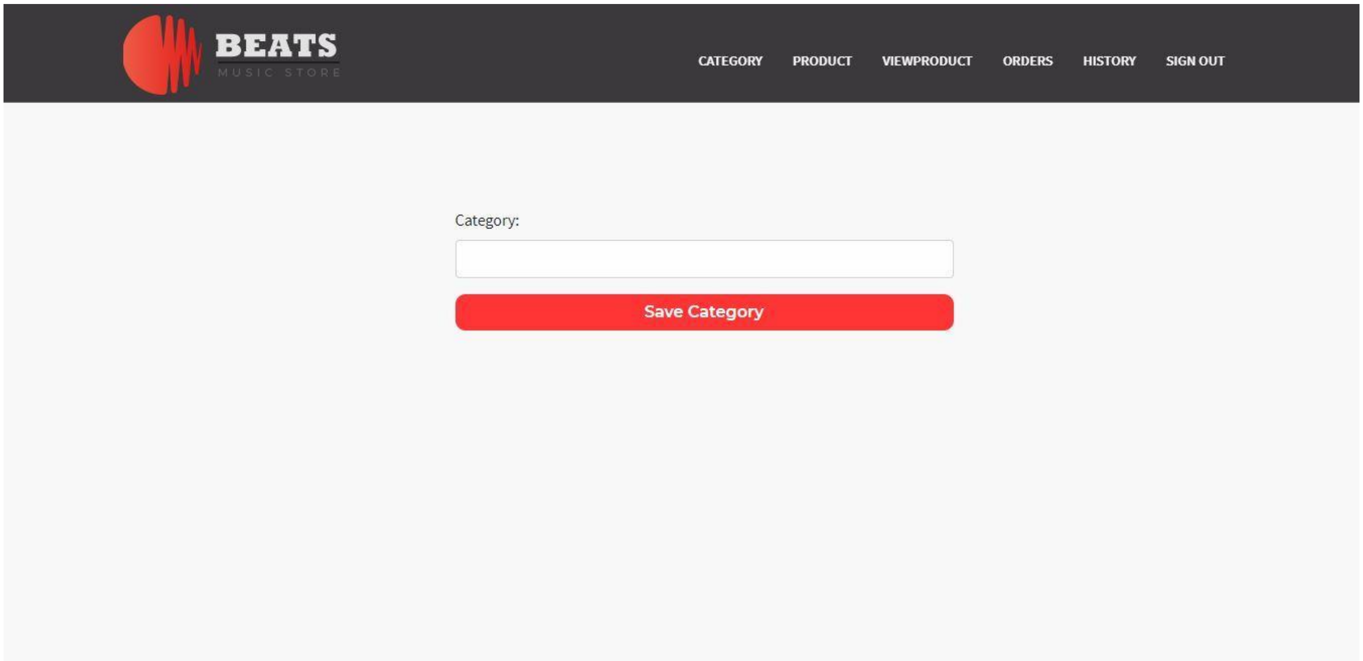
Orders page:



Order No.	Items		Bill	GST	Total Bill	Time	Type	Status
7	Beats Tornado Drums Acoustic Drum Kit	1	44450	12447	52452	2019-04-06 20:55:32	COD	pending
5	Beats Purple Haze 5 Pcs Black Plated Fusion Shell Beats Cutaway Electro Acoustic Guitar Vintage Sunb Beats Red Accent Drive Acoustic Drum Kit 5 Piece	1 1 3	198000	55441	233641	2019-04-06 20:54:24	COD	pending
4	Beats Purple Haze 5 Pcs Black Plated Fusion Shell Beats Cutaway Electro Acoustic Guitar Vintage Sunb Beats Red Accent Drive Acoustic Drum Kit 5 Piece	1 1 3	198000	55441	233641	2019-04-06 20:54:00	COD	pending
3	Beats Purple Haze 5 Pcs Black Plated Fusion Shell Beats Cutaway Electro Acoustic Guitar Vintage Sunb Beats Red Accent Drive Acoustic Drum Kit 5 Piece	1 1 3	198000	55441	233641	2019-04-06 20:52:18	COD	pending
2	Beats Ibanez MD39C 39 inch Cutaway Acoustic Guitar Beats TR1010 Bb Trumpet Lacquer	1 1	14400	4033	16993	2019-04-03 18:32:25	Paid	pending
1	Beats PA700 Professional Arranger Keyboard	3	111000	31081	130981	2019-04-02 12:38:53	COD	pending

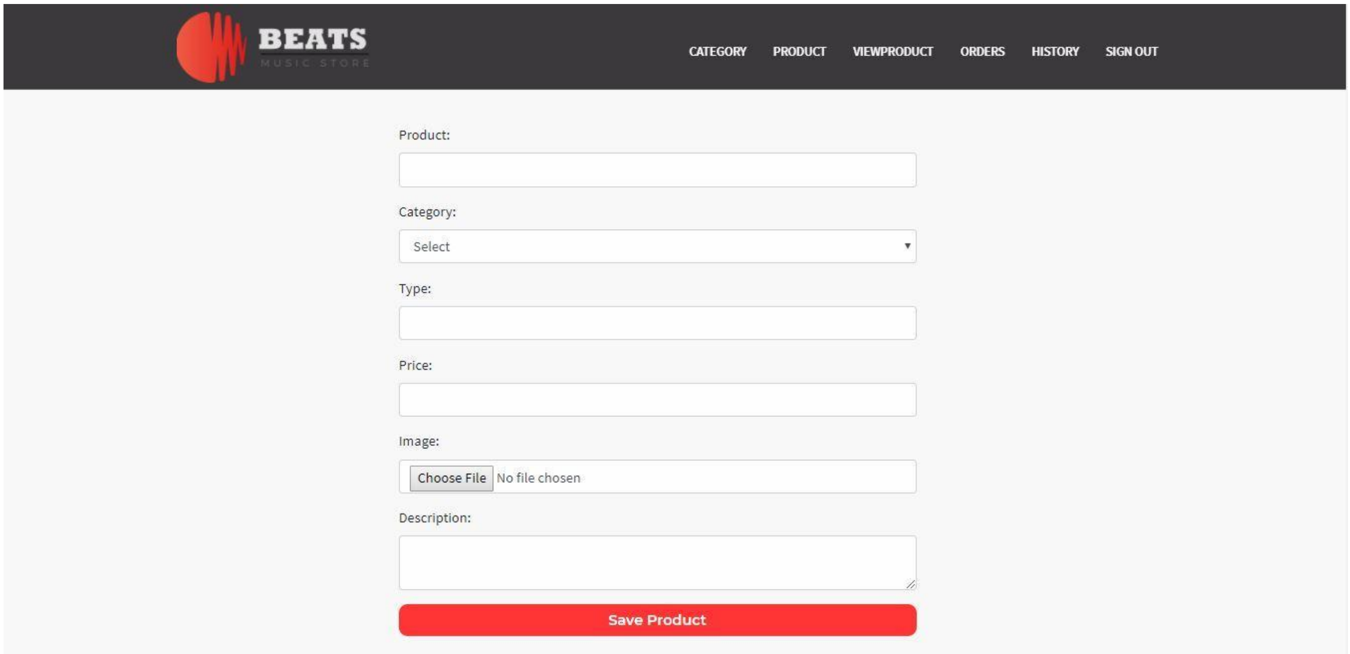
Admin

Add category page:



Category:

Add product page:



Product:

Category:

Select

Type:

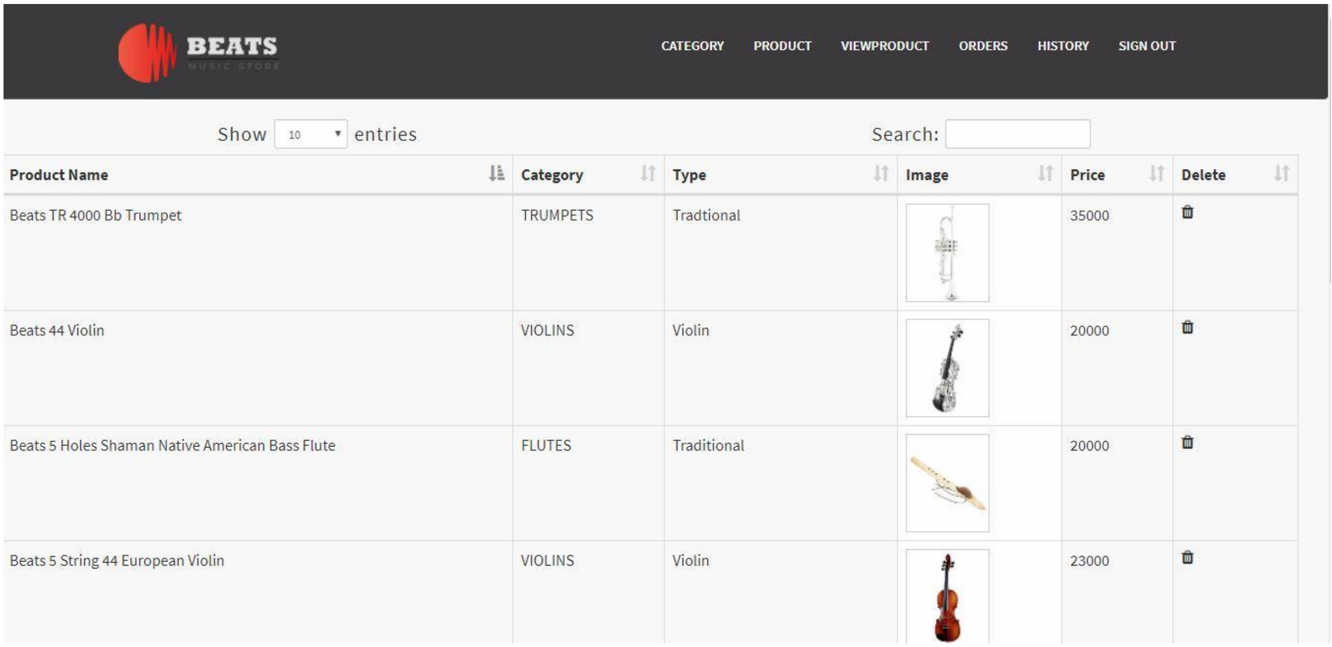
Price:

Image:

Choose File No file chosen

Description:

View product page:

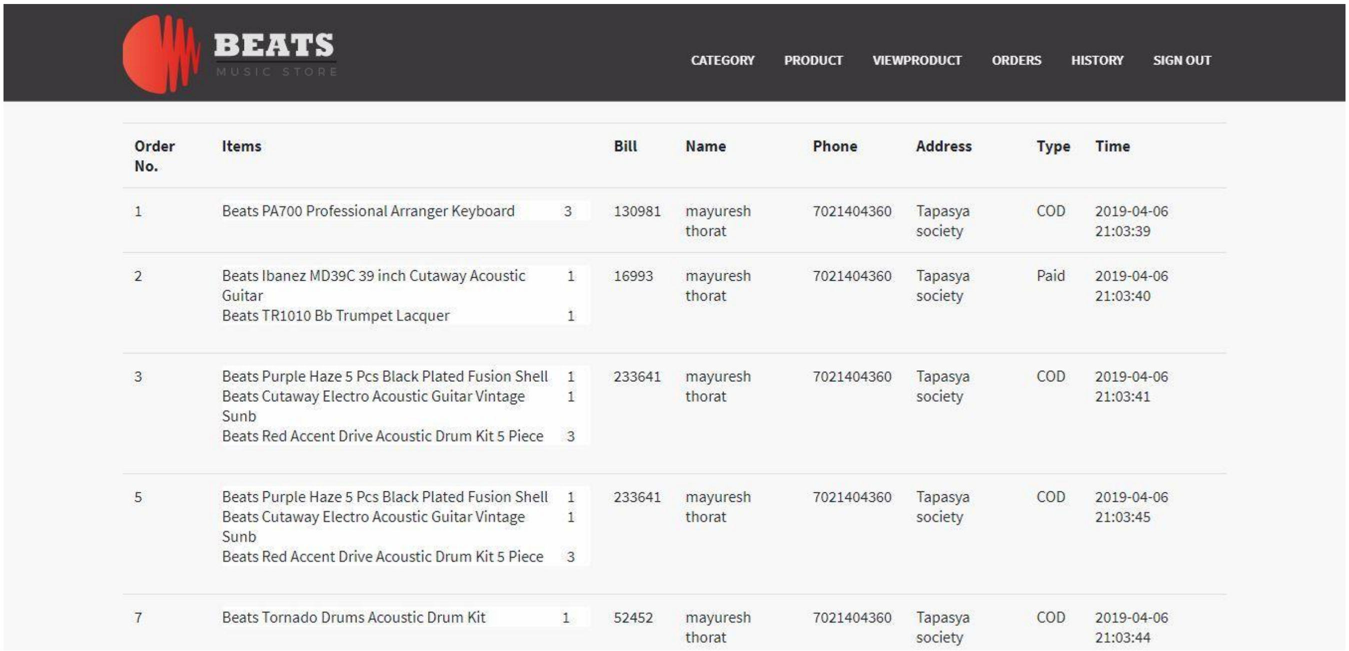


Show 10 entries

Search:

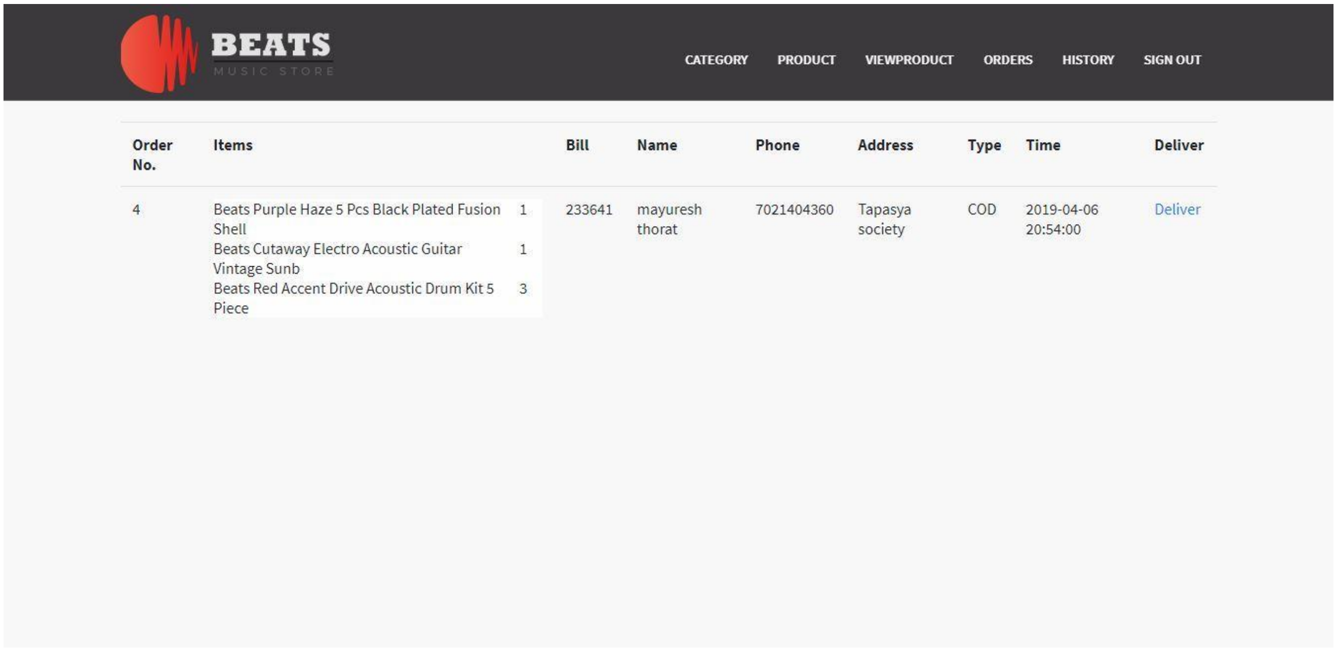
Product Name	Category	Type	Image	Price	Delete
Beats TR 4000 Bb Trumpet	TRUMPETS	Tradtional		35000	🗑
Beats 44 Violin	VIOLINS	Violin		20000	🗑
Beats 5 Holes Shaman Native American Bass Flute	FLUTES	Traditional		20000	🗑
Beats 5 String 44 European Violin	VIOLINS	Violin		23000	🗑

Order history page:



Order No.	Items		Bill	Name	Phone	Address	Type	Time
1	Beats PA700 Professional Arranger Keyboard	3	130981	mayuresh thorat	7021404360	Tapasya society	COD	2019-04-06 21:03:39
2	Beats Ibanez MD39C 39 inch Cutaway Acoustic Guitar Beats TR1010 Bb Trumpet Lacquer	1 1	16993	mayuresh thorat	7021404360	Tapasya society	Paid	2019-04-06 21:03:40
3	Beats Purple Haze 5 Pcs Black Plated Fusion Shell Beats Cutaway Electro Acoustic Guitar Vintage Sunb Beats Red Accent Drive Acoustic Drum Kit 5 Piece	1 1 3	233641	mayuresh thorat	7021404360	Tapasya society	COD	2019-04-06 21:03:41
5	Beats Purple Haze 5 Pcs Black Plated Fusion Shell Beats Cutaway Electro Acoustic Guitar Vintage Sunb Beats Red Accent Drive Acoustic Drum Kit 5 Piece	1 1 3	233641	mayuresh thorat	7021404360	Tapasya society	COD	2019-04-06 21:03:45
7	Beats Tornado Drums Acoustic Drum Kit	1	52452	mayuresh thorat	7021404360	Tapasya society	COD	2019-04-06 21:03:44

Pending order page:



Order No.	Items		Bill	Name	Phone	Address	Type	Time	Deliver
4	Beats Purple Haze 5 Pcs Black Plated Fusion Shell Beats Cutaway Electro Acoustic Guitar Vintage Sunb Beats Red Accent Drive Acoustic Drum Kit 5 Piece	1 1 3	233641	mayuresh thorat	7021404360	Tapasya society	COD	2019-04-06 20:54:00	Deliver

Chapter 7

CONCLUSIONS

7.1 CONCLUSION

We have successfully implemented the site ‘Online Musical Instrumental Store’. With the help of various links and tools, we have been able to provide a site which can now be live and run on the web. We have been successful in our attempt to take care of the needs of both the customers as well as the administrator.

Finally we hope that this will go a long way in popularizing the organization and making its work of enrollment, keeping track of Musical instruments, customer’s orders, etc much more efficient.

7.1.1 Significance of the System

With the help of Online Musical Instrumental Store web design we get an opportunity to have products and services available to customers 24 hours. It gives a good exposure to our business and help us to reach out to potential customers. Since most of the people prefer to shop online due to lack of time, we can easily make more revenue. With the help of an e-commerce website people can select and buy desired products anytime. They can pay easily through credit cards or other payment options available in website.

When it comes to buying gifts for your family and friends, an ecommerce shopping cart is what people prefer these days. Almost everything is available over internet. You just need to visit the website, select a product, and add it in your shopping cart and pay. The gift will be delivered to the mentioned destination on time. Is not that simple and hassle free?

Another feature of an Online Musical Instrumental Store website is that one can save an item in your '**cart**' and buy it later. Thus, they do not have to go through the procedure of searching the product again as it is already saved in their list. This is why e-commerce websites are so user-friendly. For those wondering about the delivery of products and issues related to it, here is how it works.

Since the whole process of purchasing things takes place online, people sometimes doubt about the timely delivery of the products. In case their requested product does not reach or reaches late, they can make a complaint at the 'goods return' section. The required measures will be taken then to ensure that they get what they had paid for.

Thus, ecommerce development is highly important in today's competitive environment.

7.2 LIMITATIONS OF THE SYSTEM

- Limitation of Online Musical Instrumental Store is, it does not have a real banking system for online payment system and need to work on it on further updates.
- Customers can’t give the reviews and ratings on the product buyed.
- Tracking of the delivery is not made by the system.
- Returning of product modules are not been developed.

7.3 FUTURE SCOPE OF THE PROJECT

This web application involves almost all the features of the online shopping. The future implementation will be :

- Online help for the customers and chatting with website administrator.
- Online tracking system for the customers to track their products before delivery.
- Customers will able to provide comments and ratings on the product the purchased.
- Online customer support system will be created for answering any queries of the customers.

References

1. *Kamal Acharya. School management system project report. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254873.34023165/v1*
2. *Kamal Acharya. A CASE STUDY OF CINEMA MANAGEMENT SYSTEM PROJECT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254873.30191075/v1*
3. *Kamal Acharya. A CASE STUDY ON ONLINE TICKET BOOKING SYSTEM PROJECT. Authorea. August 01, 2024 DOI: https://doi.org/10.22541/au.172254872.26972790/v1*
4. *Kamal Acharya. Web chatting application project report management system. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254871.18588592/v1*
5. *Kamal Acharya. RETAIL STORE MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254871.14590154/v1*
6. *Kamal Acharya. SUPERMARKET MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172252491.19145062/v1*
7. *Kamal Acharya. SOCIAL MEDIA MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172252491.11210579/v1*
8. *Kamal Acharya. Online music portal management system project report. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172252488.89734698/v1*
9. *Kamal Acharya. COLLEGE BUS MANAGEMENT SYSTEM PROJECT REPORT. Authorea. July 31, 2024. DOI: https://doi.org/10.22541/au.172245277.70798942/v1*
10. *Kamal Acharya. AUTOMOBILE MANAGEMENT SYSTEM PROJECT REPORT. Authorea. July 31, 2024. DOI: https://doi.org/10.22541/au.172245276.67982593/v1*
11. *Kamal Acharya. Ludo management system project report. Authorea. July 31, 2024. DOI: https://doi.org/10.22541/au.172243999.98091616/v1*
12. *Kamal Acharya. Literature online quiz system project report. Authorea. July 31, 2024 DOI: https://doi.org/10.22541/au.172243825.53562953/v1*
13. *Kamal Acharya. Avoid waste management system project. Authorea. July 29, 2024. DOI: https://doi.org/10.22541/au.172228528.85022205/v1*
14. *Kamal Acharya. CHAT APPLICATION THROUGH CLIENT SERVER MANAGEMENT SYSTEM PROJECT. Authorea. July 29, 2024. DOI: https://doi.org/10.22541/au.172228527.74316529/v1*
15. *Acharya, K. (2024). online taxi booking management system project report. DOI: https://doi.org/10.5281/zenodo.13642990*
16. *Acharya, K. (2024). Chat application through client server management system project report. DOI: https://doi.org/10.5281/zenodo.13642971*
17. *Acharya, K. (2024). online blood donation management system project report. DOI: https://doi.org/10.5281/zenodo.13642922*
18. *Acharya, K. (2024). online course register system project report.*

DOI: https://doi.org/10.5281/zenodo.13642791

19. *Acharya, K. (2024). Fruit shop management system project report. DOI: https://doi.org/10.5281/zenodo.13642698*
20. *Acharya, K. (2024). Toll tax management system project report. DOI: https://doi.org/10.5281/zenodo.13642635*
21. *Acharya, K. (2024). Health insurance claim management system project report. DOI: https://doi.org/10.5281/zenodo.13642551*
22. *Acharya, K. (2024). A case study of cinema management system project report. DOI: https://doi.org/10.5281/zenodo.13642433*
23. *Acharya, K. (2024). Lundry management system project report. DOI: https://doi.org/10.5281/zenodo.13642310*
24. *Acharya, K. (2024). Student information management system project report ii. DOI: https://doi.org/10.5281/zenodo.13642142*
25. *Acharya, K. (2024). A case study of online ticket booking system project report.DOI: https://doi.org/10.5281/zenodo.13641983*
26. *Acharya, K. (2024). Training and placement cell management system. DOI: https://doi.org/10.5281/zenodo.13625502*
27. *Acharya, K. (2024). Graphical password management system project report. DOI: https://doi.org/10.5281/zenodo.13625484*
28. *Acharya, K. (2024). Airplane game management system project report. Zenodo.DOI: https://doi.org/10.5281/zenodo.13372675*
29. *Acharya, K. (2023). Room finder management system project report. DOI: https://doi.org/10.5281/zenodo.13372710*
30. *Acharya, K. (2024). Bouncing ball content management system project report. DOI: https://doi.org/10.5281/zenodo.13374350*
31. *Acharya, K. (2024). Clock report management system project report. DOI: https://doi.org/10.5281/zenodo.13374306*
32. *Acharya, K. (2024). Tictactoe game management system project report. DOI: https://doi.org/10.5281/zenodo.13374244*
33. *Acharya, K. (2024). Company visitor management system projec report. DOI: https://doi.org/10.5281/zenodo.13372931*
34. *Acharya, K. (2024). A file system for mobile computing project report. DOI: https://doi.org/10.5281/zenodo.13372919*
35. *Acharya, K. (2024). Clinic management system project. DOI: https://doi.org/10.5281/zenodo.13372899*
36. *Acharya, K. (2024). Clothes store management system project report. DOI: https://doi.org/10.5281/zenodo.13372884*
37. *Acharya, K. (2024). Student feedback management system project report. DOI: https://doi.org/10.5281/zenodo.13372872*
38. *Acharya, K. (2024). College training and placement system project report. DOI: https://doi.org/10.5281/zenodo.13372847*
39. *Acharya, K. (2024). Telephone billing system project report project. DOI: https://doi.org/10.5281/zenodo.13372841*
40. *Acharya, K. (2023). Soccer management system report project. DOI: https://doi.org/10.5281/zenodo.13372825*
41. *Acharya, K. (2022). Online auditorium booking system project report. DOI: https://doi.org/10.5281/zenodo.13372817*
42. *Acharya, K. (2022). Expense tracker management system project report. DOI: https://doi.org/10.5281/zenodo.13372807*

43. *Acharya, K. (2024). Flower shop billing management system project. DOI: https://doi.org/10.5281/zenodo.13372804*

44. *Acharya, Kamal, Online Job Portal Management System (May 5, 2024). Available at SSRN: https://ssrn.com/abstract=4817534 or http://dx.doi.org/10.2139/ssrn.4817534*

45. *Acharya, Kamal, Employee leave management system. (May 7, 2024). Available at SSRN: https://ssrn.com/abstract=4819626 or http://dx.doi.org/10.2139/ssrn.4819626*

46. *Acharya, Kamal, Online electricity billing project report. (May 7, 2024). Available at SSRN: https://ssrn.com/abstract=4819630 or http://dx.doi.org/10.2139/ssrn.4819630*

47. *Acharya, Kamal, POLICY MANAGEMENT SYSTEM PROJECT REPORT. (December 10, 2023). Available at SSRN: https://ssrn.com/abstract=4831694 or http://dx.doi.org/10.2139/ssrn.4831694*

48. *Acharya, Kamal, Online job placement system project report. (January 10, 2023). Available at SSRN: https://ssrn.com/abstract=4831638 or http://dx.doi.org/10.2139/ssrn.4831638*

49. *Acharya, Kamal, Software testing for project report. (May 16, 2023). Available at SSRN: https://ssrn.com/abstract=4831028 or http://dx.doi.org/10.2139/ssrn.4831028*

50. *Acharya, Kamal, ONLINE CRIME REPORTING SYSTEM PROJECT. (August 10, 2022). Available at SSRN: https://ssrn.com/abstract=4831015 or http://dx.doi.org/10.2139/ssrn.4831015*

51. *Acharya, Kamal, Burger ordering system project report. (October 10, 2022). Available at SSRN: https://ssrn.com/abstract=4832704 or http://dx.doi.org/10.2139/ssrn.4832704*

52. *Acharya, Kamal, Teachers Record Management System Project Report (December 10, 2023). Available at SSRN: https://ssrn.com/abstract=4833821 or http://dx.doi.org/10.2139/ssrn.4833821*

53. *Acharya, Kamal, Dairy Management System Project Report (December 20, 2020). Available at SSRN: https://ssrn.com/abstract=4835231 or http://dx.doi.org/10.2139/ssrn.4835231*

54. *Acharya, Kamal, Electrical Shop Management System Project (December 10, 2019). Available at SSRN: https://ssrn.com/abstract=4835238 or http://dx.doi.org/10.2139/ssrn.4835238*

55. *Acharya, Kamal, Online book store management system project report. (Febuary 10, 2020). Available at SSRN: https://ssrn.com/abstract=4835277 or http://dx.doi.org/10.2139/ssrn.4835277*

56. *Acharya, Kamal, Paint shop management system project report. (January 10, 2019). Available at SSRN: https://ssrn.com/abstract=4835441 or http://dx.doi.org/10.2139/ssrn.4835441*

57. *Acharya, Kamal, Supermarket billing system project report. (August 10, 2021). Available at SSRN: https://ssrn.com/abstract=4835474 or http://dx.doi.org/10.2139/ssrn.4835474*

58. *Acharya, Kamal, Online taxi booking system project report. (March 10, 2022). Available at SSRN: https://ssrn.com/abstract=4837729 or http://dx.doi.org/10.2139/ssrn.4837729*

59. *Acharya, Kamal, Online car servicing system project report. (March 10, 2023). Available at SSRN: https://ssrn.com/abstract=4837832 or http://dx.doi.org/10.2139/ssrn.4837832*

60. *Acharya, Kamal, School management system project report. (July 10, 2021). Available at SSRN: https://ssrn.com/abstract=4837837 or http://dx.doi.org/10.2139/ssrn.4837837*

61. *Acharya, Kamal, Furniture Showroom Management System Project Report (March 21, 2021). Available at SSRN: https://ssrn.com/abstract=4839422 or http://dx.doi.org/10.2139/ssrn.4839422*

62. *Acharya, Kamal, Online Vehicle Rental System Project Report (March 21, 2019). Available at SSRN: https://ssrn.com/abstract=4839429 or http://dx.doi.org/10.2139/ssrn.4839429*

63. *Acharya, Kamal, Fruit Shop Management System Project Report (August 10, 2023). Available at SSRN: https://ssrn.com/abstract=4841048 or http://dx.doi.org/10.2139/ssrn.4841048*

64. *Acharya, Kamal, Hall Booking Management System Project Report (December 21, 2023). Available at SSRN: https://ssrn.com/abstract=4841055 or http://dx.doi.org/10.2139/ssrn.4841055*

65. *Acharya, Kamal, Lundry Management System Project Report (October 21, 2023). Available at SSRN: https://ssrn.com/abstract=4841059 or http://dx.doi.org/10.2139/ssrn.4841059*

66. *Acharya, Kamal, A CASE STUDY OF CINEMA MANAGEMENT SYSTEM PROJECT (September 25, 2023). Available at SSRN: https://ssrn.com/abstract=4841209 or http://dx.doi.org/10.2139/ssrn.4841209*

67. *Acharya, Kamal, A CASE STUDY ON ONLINE TICKET BOOKING SYSTEM PROJECT (May 25, 2024). Available at SSRN: https://ssrn.com/abstract=4841210 or http://dx.doi.org/10.2139/ssrn.4841210*

68. *Acharya, Kamal, ONLINE DATING MANAGEMENT SYSTEM PROJECT REPORT. (April 25, 2023). Available at SSRN: https://ssrn.com/abstract=4842066 or http://dx.doi.org/10.2139/ssrn.4842066*

69. *Acharya, Kamal, ONLINE RESUME BUILDER MANAGEMENT SYSTEM PROJECT REPORT. (April 25, 2021). Available at SSRN: https://ssrn.com/abstract=4842071 or http://dx.doi.org/10.2139/ssrn.4842071*

70. *Acharya, Kamal, TOLL TEX MANAGEMENT SYSTEM PROJECT REPORT (August 21, 2023). Available at SSRN: https://ssrn.com/abstract=4842082 or http://dx.doi.org/10.2139/ssrn.4842082*

71. *Acharya, Kamal, Chat Application Through Client Server Management System Project Report (June 25, 2023). Available at SSRN: https://ssrn.com/abstract=4842761 or http://dx.doi.org/10.2139/ssrn.4842761*

72. *Acharya, Kamal, Web Chatting Application Management System Project Report (April 25, 2022). Available at SSRN: https://ssrn.com/abstract=4842771 or http://dx.doi.org/10.2139/ssrn.4842771*

73. *Acharya, Kamal, Automobile management system project report (May 25, 2022). Available at SSRN: https://ssrn.com/abstract=4846917 or http://dx.doi.org/10.2139/ssrn.4846917*

74. *Acharya, Kamal, College bus management system project report (April 25, 2023). Available at SSRN: https://ssrn.com/abstract=4846920 or http://dx.doi.org/10.2139/ssrn.4846920*

75. *Acharya, Kamal, Courier management system project report (May 25, 2023). Available at SSRN: https://ssrn.com/abstract=4846922 or http://dx.doi.org/10.2139/ssrn.4846922*

76. *Acharya, Kamal, Event management system project report (April 25, 2021). Available at SSRN: https://ssrn.com/abstract=4846927 or http://dx.doi.org/10.2139/ssrn.4846927*

77. *Acharya, Kamal, Library management system project report II (May 25, 2020). Available at SSRN: https://ssrn.com/abstract=4848857 or http://dx.doi.org/10.2139/ssrn.4848857*