

ABSTRACT

This is a small scale project for Online shopping system. The basic idea is that customers can buy products online. The user can enter the name and password and can create an account and then place order for a particular product.

The Online Shopping system enables customers to browse through the lists of products. Also on the agenda is designing an online shopping site to manage the items in the shop and also help customers purchase them online without having to visit the shop physically. Our online shopping mall will use the internet as the sole method for selling goods to its consumers. Shopping will be highly personalized and the mall will provide lower prices than most competitors.

There are three main different categories included so that the customer could buy the products according to their requirements.

It includes product and customer management modules. This website will be useful for someone who wants to purchase desired items using the internet.

The website provides a reliable, easy and secure gateway to make payment for the product they ordered by Internet Banking facility.

CONTENTS

ACKNOWLEDGEMENT	i
ABSTRACT	ii
CHAPTERS	iii
LIST OF FIGURES	iv

CHAPTERS	Page No.
1. INTRODUCTION	1
1.1 INTRODUCTION TO HTML	
1.2 INTRODUCTION TO CSS	
1.3 INTRODUCTION TO BOOTSTRAP 4	
1.4 INTRODUCTION TO JAVASCRIPT	
1.5 INTRODUCTION TO PHP	
2. OBJECTIVES	8
3. SPECIFICATION AND SYSTEM REQUIREMENTS	9
3.1 OVERVIEW	
3.2 GOALS	
3.3 SPECIFICATIONS	
3.4 SYSTEM REQUIREMENTS	
4. SOURCECODE	10
4.1 SOURCE CODE FOR HTML	
4.2 SOURCE CODE FOR CSS	
4.3 SOURCE CODE FOR JAVASCRIPT	
4.4 SOURCE CODE FOR PHP	
5. SNAPSHOTS	16
6. CONCLUSION AND FUTURE SCOPE	21

LIST OF FIGURES

Figure No.	Title of Figures	Page No.
5.1	SIGNUP	16
5.2	LOGIN	16
5.3	HOMEPAGE	17
5.4	ALL CATEGORIES	17
5.5	PRODUCTS	18
5.6	PRODUCT DESCRIPTION	18
5.7	CHECKOUT	19
5.8	SUCCESSFUL TRANSACTION	19
5.9	FAILED TRANSACTION	20

Chapter 1

INTRODUCTION

1.1 INTRODUCTION TO HTML

HTML (HyperText Markup Language) is a Universal language which allows an individual using special code to create web pages to be viewed on the Internet. In HTML, a tag tells the browser what to do. When you write an HTML page, you enter tags for many reasons; to change the appearance of text, to show a graphic, or to make a link to another page.

Sample HTML Code:

```
<HTML>
<HEAD>
<TITLE>This is my page title! </TITLE>
</HEAD>
<BODY> This is my message to the world!
</BODY>
</HTML>
```

Browser Display:

This is my message to the world!

Frames allow an author to divide a browser window into multiple (rectangular) regions. Multiple documents can be displayed in a single window, each within its own frame. Graphical browsers allow these frames to be scrolled independently of each other, and links can update the document displayed in one frame without affecting the others. You can't just "add frames" to an existing document. Rather, you must create a frameset document that defines a particular combination of frames, and then display your content documents inside those frames. The frameset document should also include alternative non-framed content in a NOFRAMES element. An HTML comment begins with "<!--", ends with "--> ", and does not contain "--" or ">" anywhere in the comment. A hypertext link is a special tag that links one page to another page or resource. If you click the link, the browser jumps to the link's destination.

According to HTML standards, each HTML document begins with a DOCTYPE declaration that specifies which version of HTML the document uses. Many browsers use the document's DOCTYPE declaration to determine whether to use a stricter, more

standards-oriented layout mode, or to use a “quirks” layout mode that attempts to emulate older, buggy browsers.

A table can be embedded inside a cell in another table. The main caveat about nested tables is that older versions of Netscape Navigator have problems with them if you don't explicitly close your TR, TD, and TH elements. To avoid problems, include closing tags for your TR, TD, and TH even though the HTML specifications don't require them.

You use the `<table align="right">` property to float a table to the right. Put left in place of right to float left.

XHTML (Extensible Hyper text markup language)

Late 1990's XHTML 1.0 started by W3C which was a version of HTML that used stricter XML syntax rules. XHTML has strict rules to make page rendering more predictable by forcing authors to create web pages without syntax errors.

HTML5

A group of developers at Opera and Mozilla formed the WHATWG (web hypertext application technology working group) within the W3C WHATWG, HTML5 adopted instead of XHTML 2.0 by 2009 which has three aims:

- i. Specify unambiguously how browsers should deal with invalid markup.
- ii. Provide an open non-proprietary programming framework.
- iii. Be backward compatible with the existing web.

1.2 INTRODUCTION TO CSS

Cascading Style Sheets (CSS) is a list of statements (or rules) that can assign various rendering properties to HTML elements. Style rules can be specified for a single element occurrence, multiple elements, and entire document, or even multiple documents at once. A class is a group of instances of the same element to which a unique style can be attached. Grouping means gathering into a comma-separated list two or more selectors that share the same style or into a semicolon-separated list two or more declarations that are attached to the same selector.

ID selector is an individually identified (named) selector to which a specific style is declared. Using the ID attribute, the declared style can then be associated with one and only one HTML element per document as to differentiate it from all the other elements. They use the # character followed by a name. A contextual selector addresses a specific occurrence of

an element. It is a string of individual selectors separated by white space, a search pattern, where only the last element in the pattern is addressed providing it matches the specified context. Example: “ td li {color: red} “

CSS allows Unicode characters to be entered by number. For example, if a class value in some Russian document contains Cyrillic letters EL PE (Unicode numbers 041B and 041F) and you want to write a style rule for that class, you can put that letter into the style sheet by writing: {font-style: italic;} This works on all keyboards, so you don't need a Cyrillic keyboard to write class names in Russian or another language that uses that script.

Property is a stylistic parameter (attribute) that can be influenced through CSS, e.g. font or width. There must always be a corresponding value or values set to each property. The CSS clear property specifies which sides of an element where other floating elements are not allowed. This method cannot control styles for multiple documents at once. Inline Style Advantages Useful for small quantities of style definitions. Can override other style specification methods at the local level so only exceptions need to be listed in conjunction with other style methods.

1.3 Introduction to Bootstrap 4

Mark Otto announced Bootstrap 4 on October 29, 2014.[9] The first alpha version of Bootstrap 4 was released on August 19, 2015.[10] The first beta version was released on 10 August 2017. Mark suspended work on Bootstrap 3 on September 6, 2016, to free up time to work on Bootstrap 4. Bootstrap 4 was finalized on January 18, 2018.

Significant changes include:

- Major rewrite of the code
- Replacing Less with Sass
- Addition of Reboot, a collection of element-specific CSS changes in a single file, based on Normalize
- Dropping support for IE8, IE9, and iOS 6
- CSS Flexible Box support
- Adding navigation customization options
- Adding responsive spacing and sizing utilities
- Switching from the pixels unit in CSS to root ems
- Increasing global font size from 14px to 16px
- Dropping the panel, thumbnail, pager, and well components
- Dropping the Glyphicons icon font

- Huge number[quantify] of utility classes
- Improved form styling, buttons, drop-down menus, media objects and image classes

Bootstrap 4 supports the latest versions of Google Chrome, Firefox, Internet Explorer, Opera, and Safari (except on Windows). It additionally supports back to IE9 and the latest Firefox Extended Support Release (ESR).

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.

Bootstrap also comes with several JavaScript components in the form of jQuery plugins. They provide additional user interface elements such as dialog boxes, tooltips, and carousels. Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields.

The most prominent components of Bootstrap are its layout components, as they affect an entire web page. The basic layout component is called "Container", as every other element in the page is placed in it. Developers can choose between a fixed-width container and a fluid-width container. While the latter always fills the width of the web page, the former uses one of the four predefined fixed widths, depending on the size of the screen showing the page:

- Smaller than 576 pixels
- 576–768 pixels
- 768–992 pixels
- 992–1200 pixels
- Larger than 1200 pixels

Once a container is in place, other Bootstrap layout components implement a CSS grid layout through defining rows and columns.

A precompiled version of Bootstrap is available in the form of one CSS file and three JavaScript files that can be readily added to any project. The raw form of Bootstrap, however,

enables developers to implement further customization and size optimizations. This raw form is modular, meaning that the developer can remove unneeded components, apply a theme and modify the uncompiled Sass files.

1.4 Introduction to JavaScript

JavaScript often abbreviated as JS, is a high-level, interpreted scripting language that conforms to the ECMAScript specification. JavaScript has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions. Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it, and major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has APIs for working with text, arrays, dates, regular expressions, and the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities. It relies upon the host environment in which it is embedded to provide these features.

JavaScript supports much of the structured programming syntax from C (e.g., if statements, while loops, switch statements, do while loops, etc.). One partial exception is scoping: JavaScript originally had only function scoping with `var`. ECMAScript 2015 added keywords `let` and `const` for block scoping, meaning JavaScript now has both function and block scoping. Like C, JavaScript makes a distinction between expressions and statements. One syntactic difference from C is automatic semicolon insertion, which allows the semicolons that would normally terminate statements to be omitted.

JavaScript is dynamically typed like most other scripting languages. A type is associated with a value rather than an expression. For example, a variable initially bound to a number may be reassigned to a string. JavaScript supports various ways to test the type of objects, including duck typing. JavaScript includes an `eval` function that can execute statements provided as strings at run-time.

JavaScript is almost entirely object-based. In JavaScript, an object is an associative array, augmented with a prototype (see below); each string key provides the name for an object property, and there are two syntactical ways to specify such a name: dot notation (`obj.x = 10`) and bracket notation (`obj['x'] = 10`). A property may be added, rebound, or deleted at run-time. Most properties of an object (and any property that belongs to an object's prototype inheritance chain) can be enumerated using a `for...in` loop.

JavaScript uses prototypes where many other object-oriented languages use classes for inheritance. It is possible to simulate many class-based features with prototypes in JavaScript.

Like many scripting languages, arrays and objects (associative arrays in other languages) can each be created with a succinct shortcut syntax. In fact, these literals form the basis of the JSON data format. JavaScript also supports regular expressions in a manner similar to Perl, which provide a concise and powerful syntax for text manipulation that is more sophisticated than the built-in string functions.

As of May 2017 94.5% of 10 million most popular web pages used JavaScript.[10] The most common use of JavaScript is to add client-side behavior to HTML pages, also known as Dynamic HTML (DHTML). Scripts are embedded in or included from HTML pages and interact with the Document Object Model (DOM) of the page. JavaScript code can run locally in a user's browser (rather than on a remote server), increasing the application's overall responsiveness to user actions. JavaScript code can also detect user actions that HTML alone cannot, such as individual keystrokes.

1.5 Introduction to PHP

The term PHP is an acronym for PHP: Hypertext Preprocessor. PHP is a server-side scripting language designed specifically for web development.

- Websites like www.facebook.com, www.yahoo.com are also built on PHP.
- One of the main reason behind this is that PHP can be easily embedded in HTML files and HTML codes can also be written in a PHP file.
- The thing that differentiates PHP with client-side language like HTML is, PHP codes are executed on server whereas HTML codes are directly rendered on the browser. PHP codes are first executed on the server and then the result is returned to the browser.
- The only information that the client or browser knows is the result returned after executing the PHP script on the server and not the actual PHP codes present in the PHP file. Also, PHP files can support other client-side scripting languages like CSS and JavaScript.

PHP can actually do anything related to server-side scripting or more popularly known as the backend of a website. For example, PHP can receive data from forms, generate dynamic page content, can work with databases, create sessions, send and receive cookies,

send emails etc. There are also many hash functions available in PHP to encrypt user's data that makes PHP secure and reliable to be used as a server-side scripting language. So these are some of the abilities of PHP that makes it suitable to be used as server-side scripting language. You will get to know more of these abilities in further tutorials.

Even if you are not convinced by the above abilities of PHP, there are some more features of PHP. PHP can run on all major operating systems like Windows, Linux, Unix, Mac OS X etc. Almost all of the major servers available today like Apache supports PHP. PHP allows using wide range of databases. And the most important factor is that it is free to use and download and anyone can download PHP from its official source.

Chapter 2**OBJECTIVES**

The aim of this project is on the online shopping application it is developed using HTML5, JavaScript, CSS, PHP, Bootstrap 4. The application is very useful where the buyer can directly buy the products from home via internet on mobile or system. The application reduces lot of work load for customer as well as owner. The transaction of money is completed in real time system.

Some of the online shops are EBAY Amazon. By this online shopping the product is directly delivered to customer home. Online shopping is the process consumers go through to purchase products or Services over the Internet. An online shop, e-shopping, e-store, internet shop, webshop , web store , online store, or virtual store evokes the physical analogy of buying products or services at a bricks-and-mortar retailer or in a mall. The metaphor of an online catalog is also used, by analogy with mail order catalogs.

All types of stores have retail web sites, including those that do and do not also have physical storefronts and paper catalogs. Online shopping is a type of electronic commerce used for business-to-business(B2B) and business-to-consumer (B2C) transactions. The term Web shop also refers to a place of business where web development, web hosting and other types of web related activities take place (Web refers to the World Wide Web and "shop" has a colloquial meaning used to describe the place where one's occupation is carried out).

Chapter 3

SPECIFICATION AND SYSTEM REQUIREMENTS

3.1 OVERVIEW

The E-commerce website is based on:

- To signup and login as a new user.
- To browse and buy a chosen listed products.

3.2 GOALS

The goal of this work is to provide user with a interface that can make user easy to use and place a product of the merchant by using online payment gateway for reliable and secure online transactions.

3.3 SPECIFICATIONS

Using this website a user can signup and login with personal email address and respective password, on successful login user can browse and lookup products listed on the website, and also they can place order by providing the details of fund transfer on successful transaction.

3.4 SYSTEM REQUIREMENTS

Software Requirements:

Operating System	: Windows, Linux, MacOS
Workspace	: Sublime Text Editor, Browser
Server	: Bitnami WAMP
Language	: HTML, CSS, JavaScript, PHP

Software Requirements:

Processor	: x86 compatible processor with 1.7GHz or more
Memory	: 2GB or more
Hard Disk Space	: 20GB or more

Chapter 4

SOURCE CODE

4.1 SOURCE CODE FOR HTML

```
<!DOCTYPE html><html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>signin/signup</title>
  <link rel="stylesheet" href="./style.css">
</head>
<body>
<html>
<head>
  <title>Login or Register Account </title>
  <meta charset="utf-8" />
  <meta name="viewport"
content="width=device-width,initial-scale=1,minimum-scale=1.0,max
imum-scale=1.0,user-scalable=no" />
</head>
<body>
  <div class="container" id="div1">
    <div class="tab">
      <div class="tablinks" id="logintab"
data-form-id="login">Login</div>
      <div class="tablinks" id="signuptab"
data-form-id="signup">Sign up</div>
    </div>
    <div class="tabcontent" id="login">
      <form name="formlogin" action = "login.php" onsubmit =
"return validation()" method = "POST">
        <label for="loginEmail">Email:</label>
        <input required type="email" id="loginemail" name
="loginemail"placeholder="you@domain.com" />
        <label for="loginPassword">Password:</label>
        <input required type="password" id="loginpassword"
name="loginpassword" placeholder="*****" />
        <button type="submit">Login</button>
      </form>
    </div>
    <div class="tabcontent" id="signup">
      <form name="formsignup" action = "signup.php" onsubmit =
"return validationSignup()" method = "POST">
        <label for="username">Name:</label>
        <input required type="text" id="username"
name="username" placeholder="username" />
        <label for="signupemail">Email:</label>
        <input required type="email" id="signupemail"
name="signupemail" placeholder="you@domain.com" />
        <label for="signuppasword">Password:</label>
        <input required type="password" id="signuppasword"
name="signuppasword" placeholder="*****" />
        <button type="submit">Sign up</button>
      </form>
    </div>
  </div></body></html>
  <script src="./script.js"></script>
</body>
```

4.2 SOURCE CODE FOR CSS

```
@import
url('https://fonts.googleapis.com/css2?family=Poiret+One&display=
swap');
body {
    margin: 0 auto;
    font-family: 'Poiret One', cursive;
    font-weight: bolder;
    background: url("image/loginbg.png");
}
div.container {
    margin: 4em auto 2em auto;
    padding: 1em;
    overflow: hidden;
    margin-top: 25vh;
}
div.container label,div.container input,div.container button {
    display: block;
    width: 80%;
    font-size: 1em;
}
div.container label {
    margin: 0.5em auto 0.5em auto;
}
div.container input {
    padding: 0.8em 0 0.8em 0.8em;
    outline: none;
    border: none;
    border-bottom: 2px solid black;
    margin: 0.5em auto 0.5em auto;
}
div.container button {
    outline: none;
    margin: 2em auto 1em auto;
    padding: 1.2em;
    cursor: pointer;
    letter-spacing: 0.2em;
    background-color: rgba(0,0,0,0.04);
    color: black;
    border: none;
    border-radius: 25px;
}
div.container button:hover {
    background-color: black;
    color: white;
    transition: 0.09s ;
}
.tab {
    display: flex;
    width: 100%;
}
.tablinks {
    padding: 1em 0 1em 0;
    width: 50%;
    text-align: center;
    cursor: pointer;
    border-radius: 25px;
    transition: all 0.2s linear;
    background-color: rgba(0,0,0,.04);
}
```

```
.active {
  animation-name: activetab;
  animation-duration: 0.5s;
  animation-iteration-count: 1;
  background-color: black;
  color: white;
}
@keyframes activetab {
  from {
    transform: rotateY(180deg);
  }
  to {
    transform: rotateY(0deg);
  }
}
.tabcontent {
  width: 100%;
  margin: 0 auto 0 auto;
  display: none;
  padding: 2em 0 2em 0;
  overflow: hidden;
}
@media only screen and (min-width: 1224px) {
  div.container {
    overflow: hidden;
    width: 400px;
  }
}
@media only screen and (min-width: 1824px) {
  div.container {
    width: 400px;
  }
}
@media only screen and (min-device-width: 320px) and
(max-device-width: 480px) {
  div.container {
    width: 80%;
  }
}
```

4.3 SOURCE CODE FOR JAVASCRIPT

```
var tablinks = document.querySelectorAll("div.tablinks");
for (var i = 0; i < tablinks.length; i++) {
    tablinks[i].onclick = function(e) {
        openForm(e.target.getAttribute('data-form-id'),
e.target.getAttribute('id'));
    }
}

function openForm(formid, tabid) {
    var tabcontent = document.querySelectorAll("div.tabcontent");
    var links = document.querySelectorAll("div.tablinks");
    for (var i = 0; i < links.length; i++) {
        links[i].classList.remove("active");
    }
    for (i = 0; i < tabcontent.length; i++) {
        tabcontent[i].style.display = "none";
    }
    document.getElementById(formid).style.display = "block";
    document.getElementById(tabid).classList.add("active");
}
document.getElementById("logintab").click();

function validationSignup()
{
    var sun=document.div1.signup.formsignup.username.value;
    var sid=document.div1.signup.formsignup.signuemail.value;
    var
sps=document.div1.aignup.formsignup.signuppassword.value;
    if(sid.length==" " && sps.length==" ") {
        alert("User Name and Password fields are empty");
        return false;
    }
    else
    {
        if(sid.length==" ") {
            alert("User Name is empty");
            return false;
        }
        if (sps.length==" ") {
            alert("Password field is empty");
            return false;
        }
        if (sun.length==" ") {
            alert("username field is empty");
            return false;
        }
    }
}

function validation()
{
    var id=document.div1.login.formlogin.loginemail.value;
    var ps=document.div1.login.formlogin.loginpassword.value;
```



```
if(id.length==" " && ps.length==" ") {
    alert("User Name and Password fields are empty");
    return false;
}
else
{
    if(id.length==" ") {
        alert("User Name is empty");
        return false;
    }
    if (ps.length==" ") {
        alert("Password field is empty");
        return false;
    }
}
}
```

4.4 SOURCECODE FOR PHP

connection.php

```
<?php

$host = "localhost";
$user = "root";
$password = 'shashank';
$db_name = "webproject";
$con = mysqli_connect($host, $user, $password, $db_name);

if(mysqli_connect_errno()) {
    die("Failed to connect with MySQL: ".mysqli_connect_error());
}

?>
```

signup.php

```
<?php
include('connection.php');
$username = $_POST['username'];
$email = $_POST['signuemail'];
$password = $_POST['signuppasword'];
$username = stripslashes($username);
$email = stripslashes($email);
$password = stripslashes($password);
$email = mysqli_real_escape_string($con, $email);
$password = mysqli_real_escape_string($con, $password);
$sql = "INSERT INTO signup (sname,semail, spassword) VALUES
('$username','$semail', '$spassword')";
if ($con->query($sql) === TRUE) {
    header('Location: index.html');
    exit;
}
else {
    ?>
```

```
        <<script type='text/javascript'>
        alert('Email address already exists.')
        window.location.href = "index.html";
        </script>
        <?php
        }
    ?>
```

login.php

```
<?php
include('connection.php');
$username = $_POST['loginemail'];
$password = $_POST['loginpassword'];

//to prevent from mysqli injection
$username = stripslashes($username);
$password = stripslashes($password);
$username = mysqli_real_escape_string($con, $username);
$password = mysqli_real_escape_string($con, $password);
$sql = "select *from signup where semail = '$username' and
spassword = '$password'";
$result = mysqli_query($con, $sql);
$row = mysqli_fetch_array($result, MYSQLI_ASSOC);
$count = mysqli_num_rows($result);
if($count == 1){
    header('Location: homepage.html');
    exit;
}
else{
    ?>
    <<script type='text/javascript'>
    alert('Email and password do not match.')
    window.location.href = "index.html";
    </script>
    <?php
    }
?>
```

Chapter 5

SNAPSHOTS

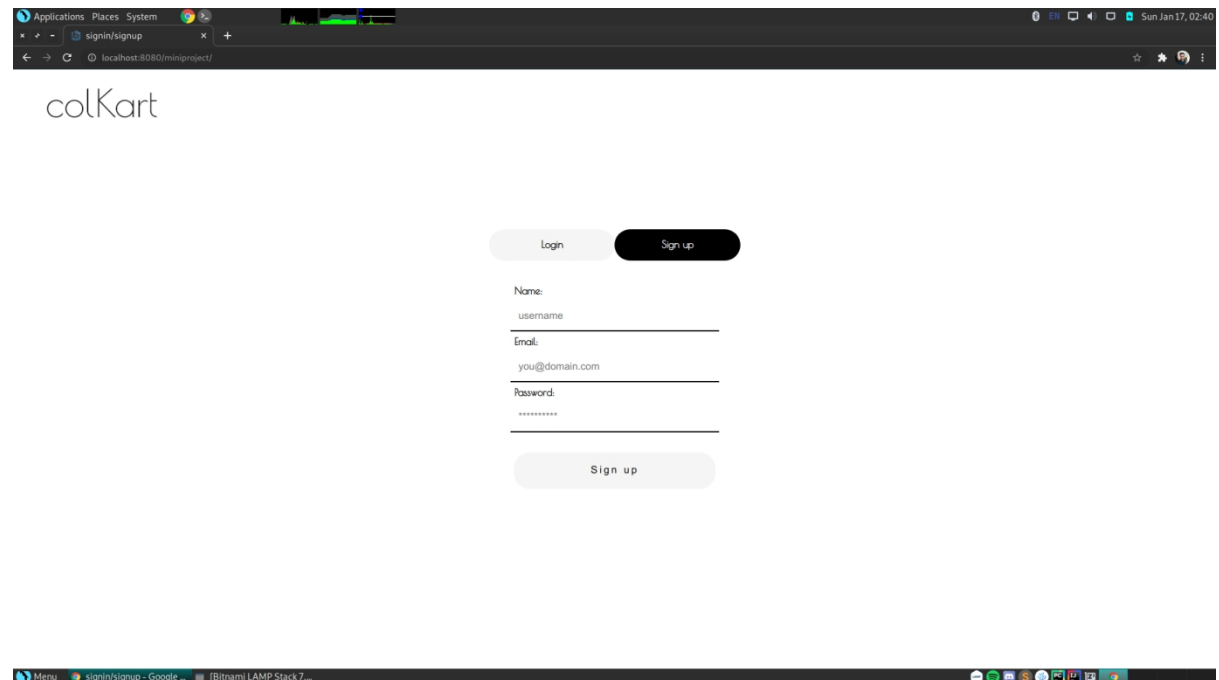


Fig 5.1 Signup

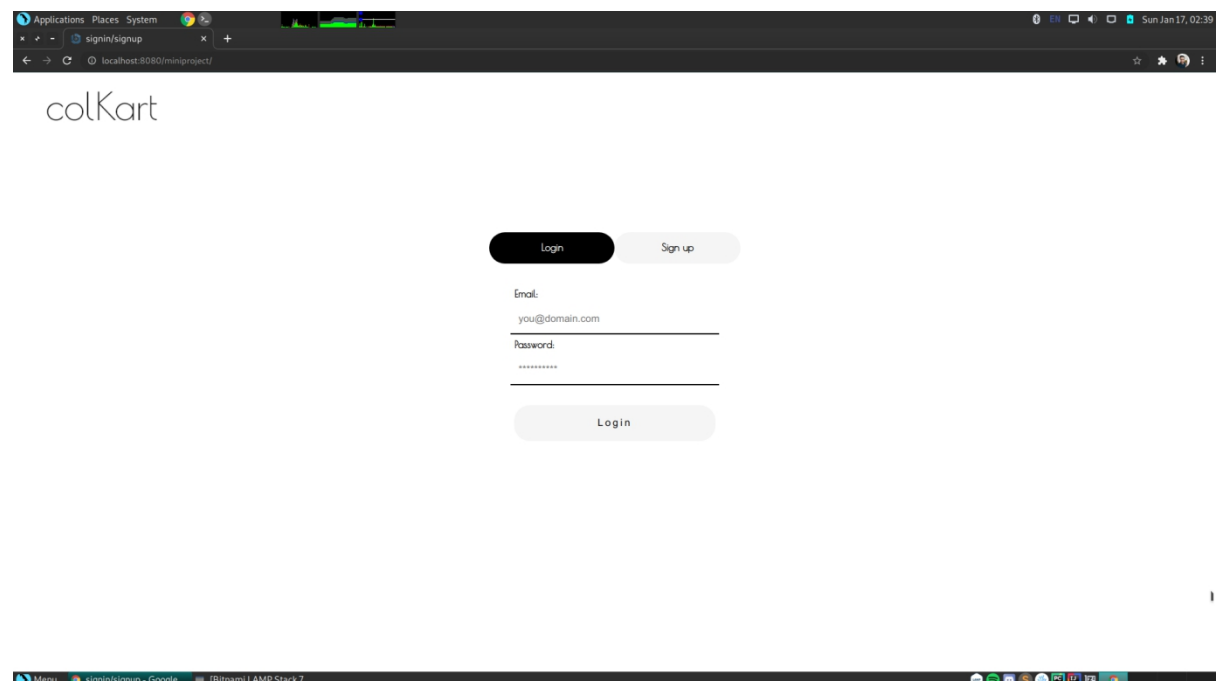


Fig 5.2 Login

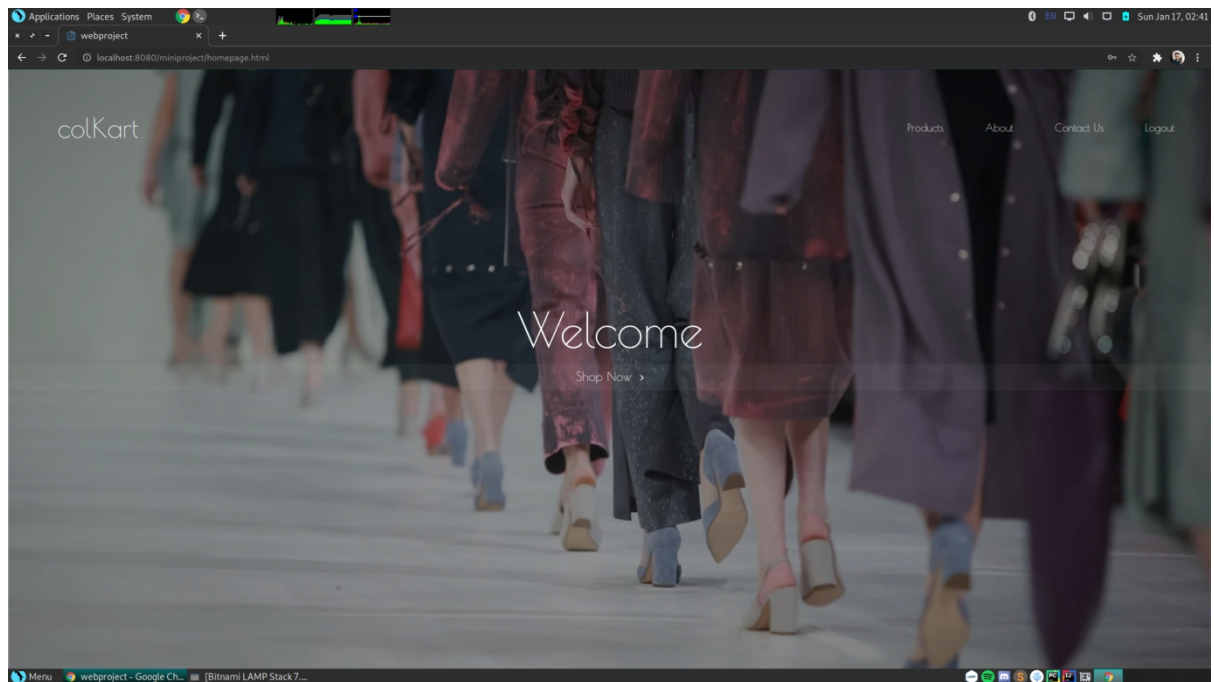


Fig 5.3 Homepage

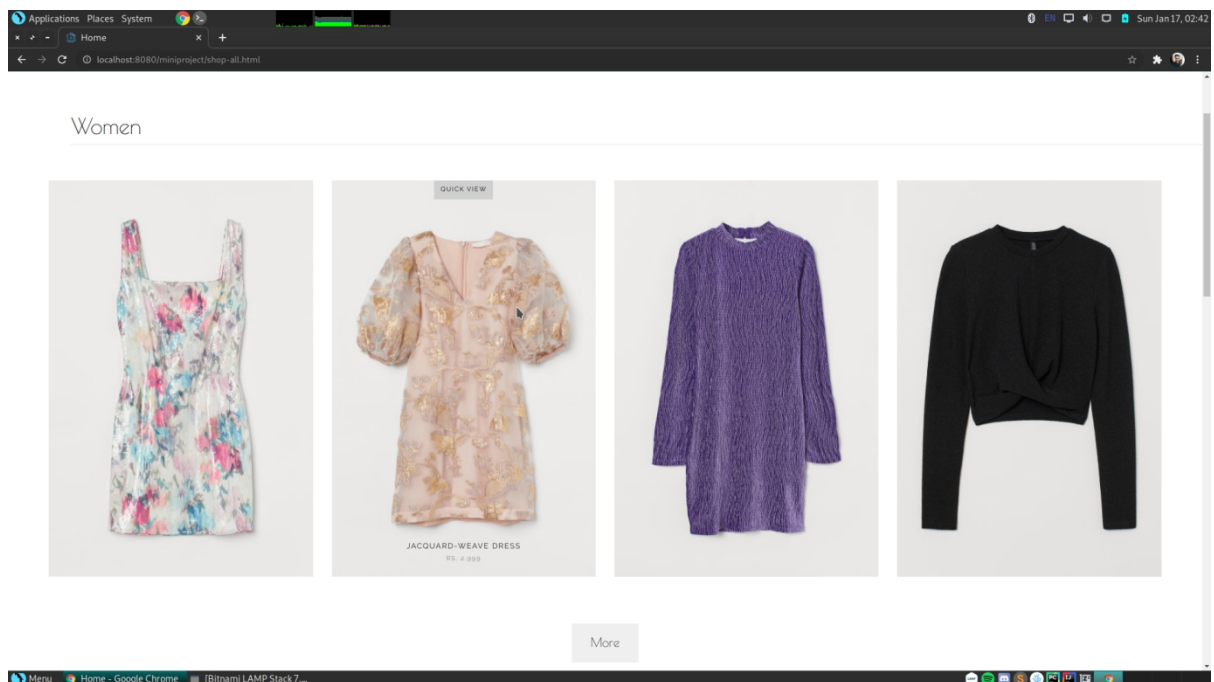


Fig 5.4 All categories

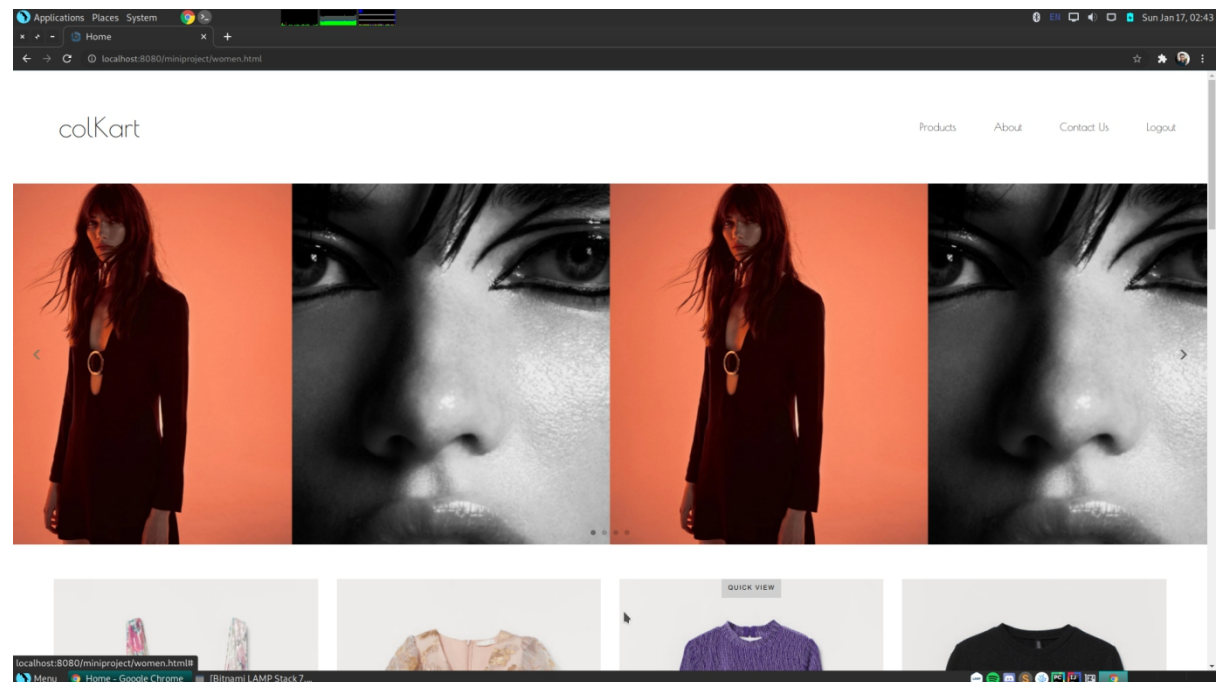


Fig 5.5 Products

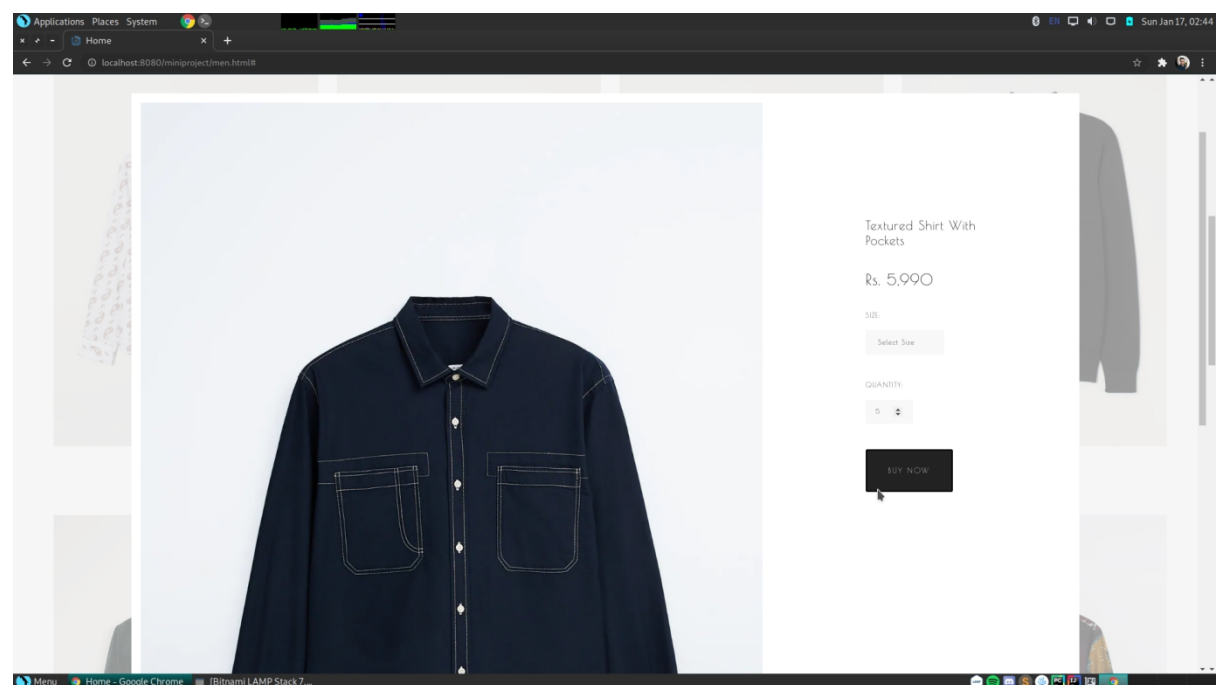


Fig 5.6 Product description

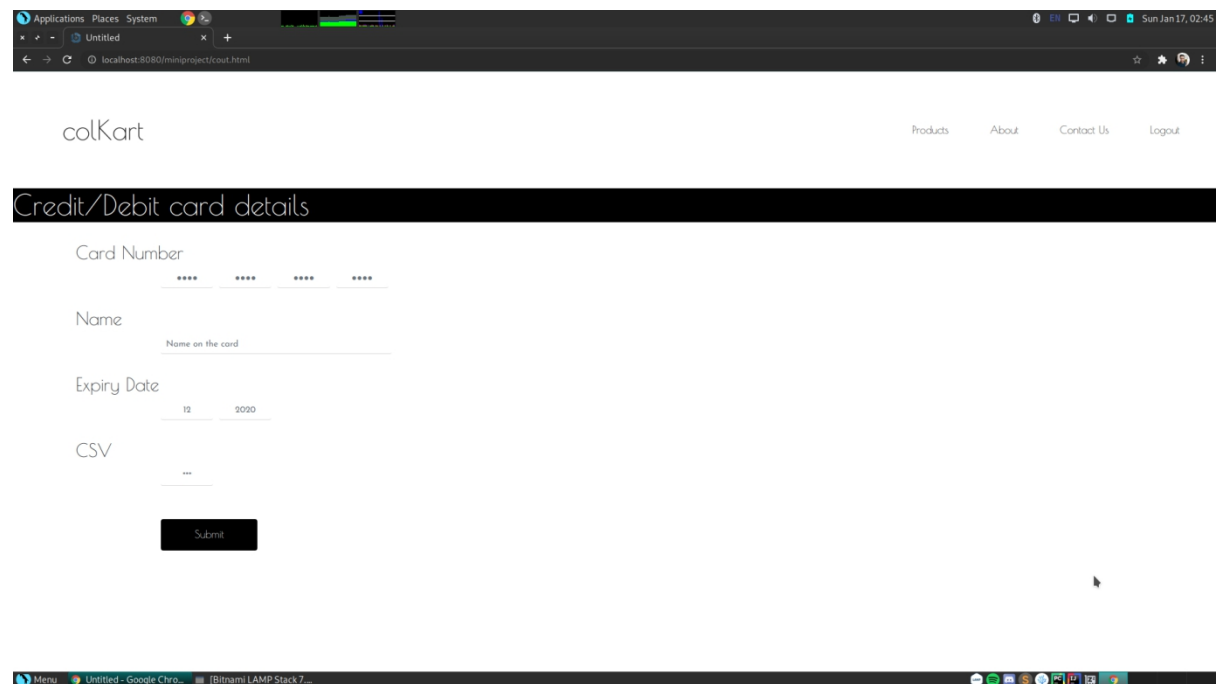


Fig 5.7 Checkout

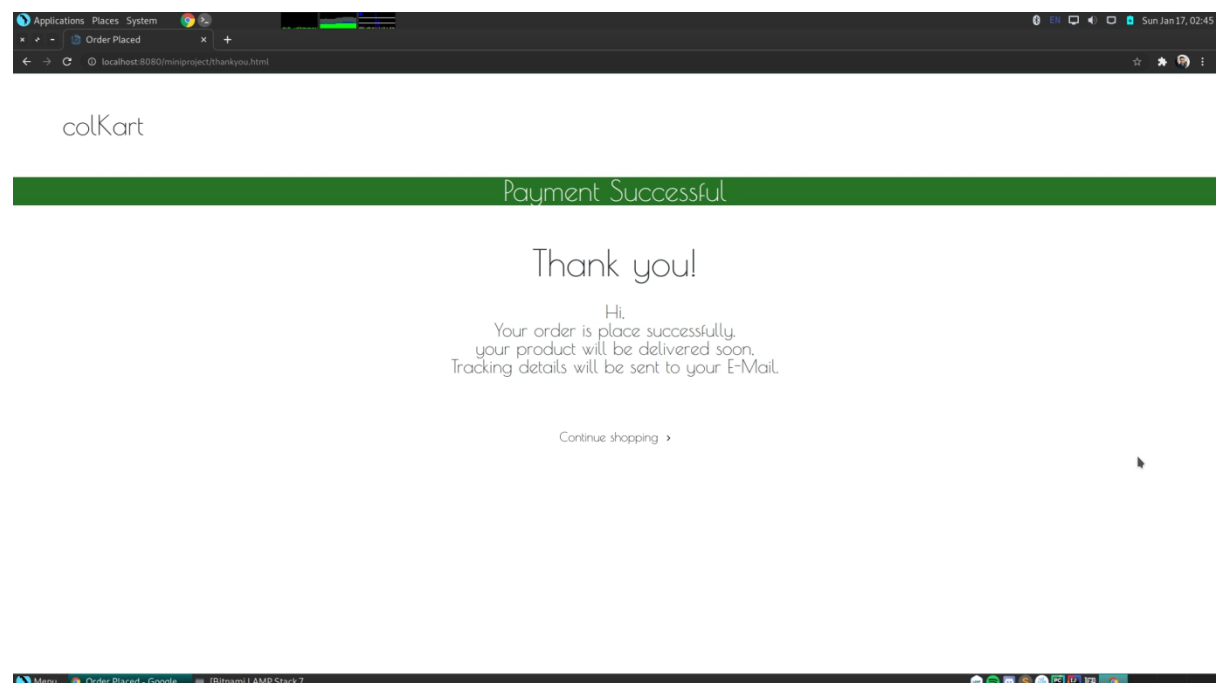


Fig 5.8 Successful transaction

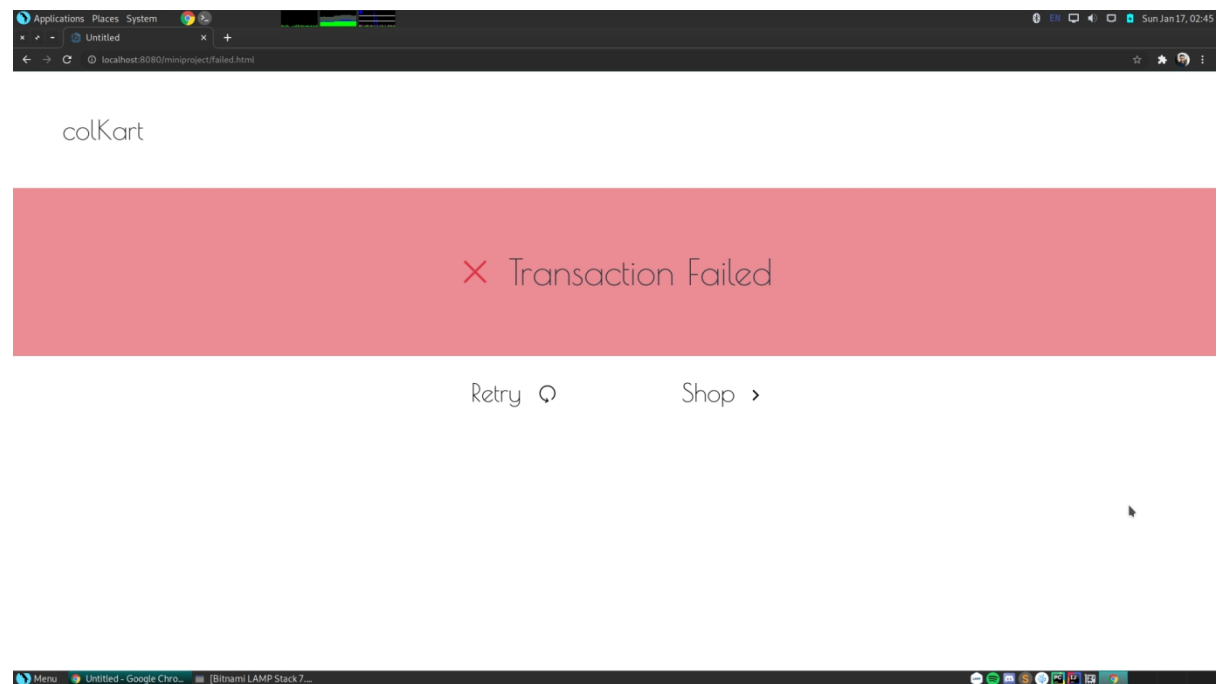


Fig 5.9 Failed transaction

Chapter 6

CONCLUSION AND FUTURE SCOPE

Technology has made significant progress over the years to provide consumers a better online shopping experience and will continue to do so for years to come. With the rapid growth of products and brands, people have speculated that online shopping will overtake in-store shopping. While this has been the case in some areas, there is still demand for brick and mortar stores in market areas where the consumer feels more comfortable seeing and touching the product being bought. However, the availability of online shopping has produced a more educated consumer that can shop around with relative ease without having to spend a large amount of time. In exchange, online shopping has opened up doors to many small retailers that would never be in business if they had to incur the high cost of owning a brick and mortar store. At the end, it has been a win-win situation for both consumer and sellers.

People are now using the E-Commerce for fulfilling their shopping desires. Most of them are completely aware of all the pros and cons of online shopping. Graduates are the majority users of E-retail. Cash on delivery remains the best choice for payment followed by Debit and credit cards. Users are marginally more interested in shopping through the internet due to its convenience. Most users are shopping once or twice a week through the web with spending ranging from rupees 100 to 2500 monthly. With most of the E-retailers selling branded goods and having flexible return policies they are being well trusted by the users. Apparel, footwear and Accessories lead as the most demanded goods online.

Though there are a number of challenges, still the e-commerce industry is expected to reach heights. Let us find out what experts are expecting. The e-commerce industry is growing at an unexpected rate. So, the scope is limitless. This is why experts say that a lot of new eCommerce trends will be soon seen emerging in the industry.

Here are some future scope of this work

- Social Media Influence
- Mobile Apps
- Google Buttons
- Drone
- Artificial Intelligence
- App-only Approach