PROJECT REPORT ON

# E-commerce

SUBMITTED BY:

Pavan Gupta (201500467)

Kshitij Varshney (201500356)

Prabhu Awasthi (201500479)



## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSITUTE OF ENGINNERING & TECHNOLOGY

### GLA UNIVERSITY, MATHURA



DECLARATION

I would like to express my special thanks of gratitude to my project guide Ms. Ruchi Talwar mam who gave us the golden opportunity to do this wonderful project on the topic Positivae(E-commerce website), which also helped us in doing a lot of research and we came to know about so many new things we are really thankful to them.

Secondly, we would also like to thank our parents and friends who helped us a lot in finalizing this project within the limited time frame.

Candidate’s Names:

Pavan Gupta 201500467

Kshitij Varshney 201500359

Prabhu Awasthi 201500479

CERTIFICATE

This is to certify that the above statements made by the candidates are correct to the best of my/our knowledge and belief.

Project Supervisor

Ms. Ruchi Talwar

Assistant Professor

Date: 23-Nov-2022

Table of Content

### 1. Introduction

1.1 Overview …………………………….

1.2 Objective.…………………..…………

### 2. Technology Used

2.1 HTML ……………………….………….

2.2 CSS………………………………………

2.3 MongoDB……………………………

2.4 JavaScript……………………………………

### 3. System Requirements

3.1 Software Required………………….………

3.2 Hardware Required……….……………………...

### 4. Implementation

4.1 Explanation of Source Code……………………......

4.2 Final Code …………………….................................

4.3 Output……………………………………………....

5. Conclusion

INTRODUCTION

In today’s world, every retail shop owner faces the problem of constant fall of customers, reason being the digital platforms taking over the business world. In order to survive the entrepreneur competition going on, the local shop owners now have to take their business strategies to a new level, by including the digitalization and e-marketing for their business. They need to provide the clients/customers door-step service, where they can provide the customer the most reliable and an easy service.

We have taken over this project of developing a website for clothing products dealer to help him make profits in the earning and provide the customers the features that can ease their problems regarding what to buy.

Web:

Web development refers to the process of creating websites on the Internet.

The term “web development” is relatively broad in its application. You could create a single website page from a wix template, or you could painstakingly develop a massive website with thousands of original pages — and technically, both of those would count as web development.

Now a day’s web development is the thing which is in boost.

### TECHNOLOGY USED

HTML:

HTML is an acronym which stands for Hyper Text Markup

Language which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

Hyper Text: Hyper Text simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. Hyper Text is a way to link two or more web pages (HTML documents) with each other.

Markup language: A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

Web Page: A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. With the help of HTML only, we can create static web pages.

CSS:

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

**JavaScript**

JavaScript often abbreviated as JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. As of 2022, 98% of websites use JavaScript on the client side for webpage behavior, often incorporating third-party libraries. All major web browsers have a dedicated JavaScript engine to execute the code on users' devices. JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard.[10] It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). The ECMAScript standard does not include any input/output (I/O), such as networking, storage, or graphics facilities. In practice, the web browser or other runtime system provides JavaScript APIs for I/O. JavaScript engines were originally used only in web browsers, but are now core components of some servers and a variety of applications. The most popular runtime system for this usage is Node.js. Although Java and JavaScript are similar in name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.

**MongoDB**

MongoDB is an open-source document-oriented database that is designed to store a large scale of data and also allows you to work with that data very efficiently. It is categorized under the NoSQL (Not only SQL) database because the storage and retrieval of data in the MongoDB are not in the form of tables.

The MongoDB database is developed and managed by MongoDB.Inc under SSPL(Server Side Public License) and initially released in February 2009. It also provides official driver support for all the popular languages like C, C++, C#, and .Net, Go, Java, Node.js, Perl, PHP, Python, Motor, Ruby, Scala, Swift, Mongoid. So, that you can create an application using any of these languages. Nowadays there are so many companies that used MongoDB like Facebook, Nokia, eBay, Adobe, Google, etc. to store their large amount of data.

### SYSTEM REQUIREMENTS

Software Requirement-

#### To build application –

* 64-bit Windows 8/10/11
* Libraries
* Visual Studio code (latest version).

#### To Run Website –

* Web Browsers (chrome, Mozilla)

Hardware Requirement –

* x86\_64 CPU architecture; 2nd generation Intel Core or newer
* 8 GB RAM or more
* 8 GB of available disk space minimum

### IMPLEMENTATION

Final Code:

 page 1:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Login</title>

<link rel="stylesheet" href="loginstyle.css">

</head>

<body>

<section class="user bcimage">

<div class="user\_options-container">

<div class="user\_options-text">

<div class="user\_options-unregistered">

<h2 class="user\_unregistered-title">Don't have an account?</h2>

<p class="user\_unregistered-text">Create an account faster to make yourself "Prepration less" to "Prepared"</p>

<button class="user\_unregistered-signup" id="signup-button">Sign up</button>

</div>

<div class="user\_options-registered">

<h2 class="user\_registered-title">Have an account?</h2>

<p class="user\_registered-text">Banjo tote bag bicycle rights, High Life sartorial cray craft beer whatever street art fap.</p>

<button class="user\_registered-login" id="login-button">Login</button>

</div>

</div>

<div class="user\_options-forms" id="user\_options-forms">

<div class="user\_forms-login">

<h2 class="forms\_title">Login</h2>

<form class="forms\_form" id="form\_">

<div class="forms\_field">

<input type="text" name="email" id="email" placeholder="Email / Id" class="forms\_field-input" required autofocus />

</div>

<div class="forms\_field">

<input type="password" id="password" name="password" placeholder="Password" class="forms\_field-input" required />

</div>

<div class="forms\_buttons">

<button type="button" class="forms\_buttons-forgot">Forgot password?</button>

<input type="submit" value="Log In" class="forms\_buttons-action">

</div>

</form>

</div>

<div class="user\_forms-signup">

<h2 class="forms\_title" style="margin-top: -5px;">Sign Up</h2>

<form class="forms\_form" id="form\_r" style="margin-top: -25px;">

<div class="forms\_field">

<input type="text" placeholder="unique id" id="id" class="forms\_field-input" required />

</div>

<div class="forms\_field">

<input type="text" placeholder="Full Name" id="name" class="forms\_field-input" required />

</div>

<div class="forms\_field">

<input type="email" placeholder="Email " id="email" class="forms\_field-input" required />

</div>

<div class="forms\_field">

<input type="password" placeholder="Password" id="pass" class="forms\_field-input" required />

</div>

<div class="forms\_buttons">

<input type="submit" value="Sign up" class="forms\_buttons-action">

</div>

</form>

</div>

</div>

</div>

</section>

<script>

window.onload = () => {

fetch('http://localhost:4000/login/if', {

credentials: "include"

}).then(s => s.text()).then(

res => {

console.log(res)

if (res == "logged") {

window.location.replace('http://localhost:4000/main.html')

}

}

)

}

document.getElementById("form\_").addEventListener('submit', (e) => {

e.preventDefault();

login()

})

function login() {

const formm = document.getElementById("form\_")

const data = {

id: formm.elements['email'].value,

pass: formm.elements['password'].value

}

console.log(data)

fetch('http://localhost:4000/login', {

"method": "POST",

"body": JSON.stringify(data),

credentials: "include",

headers: {

'Content-Type': 'application/json',

}

}).then(d => d.text()).then(s => {

console.log(s)

localStorage.setItem('userInfo', {

name: data.name

})

if (s == "sucess") {

window.location.replace("http://localhost:4000/main.html");

}

})

}

document.getElementById("form\_r").addEventListener('submit', (e) => {

e.preventDefault();

register()

})

function register() {

const formm = document.getElementById("form\_r")

const data = {

id: formm.elements['id'].value,

pass: formm.elements['pass'].value,

email: formm.elements['email'].value,

name: formm.elements['name'].value

}

console.log(data)

fetch('http://localhost:4000/register', {

"method": "POST",

"body": JSON.stringify(data),

credentials: "include",

headers: {

'Content-Type': 'application/json',

}

}).then(d => d.text()).then(s => {

alert('sucess now verify mail and login again')

localStorage.setItem('userInfo', {

name: data.name,

email: data.email

})

window.location.reload(false);

})

}

const signupButton = document.getElementById('signup-button'),

loginButton = document.getElementById('login-button'),

userForms = document.getElementById('user\_options-forms')

signupButton.addEventListener('click', () => {

userForms.classList.remove('bounceRight')

userForms.classList.add('bounceLeft')

}, false)

loginButton.addEventListener('click', () => {

userForms.classList.remove('bounceLeft')

userForms.classList.add('bounceRight')

}, false)

</script>

</body>

</html>

**Page2:**

/\*\*

\* General variables

\*/

/\*\*

\* General configs

\*/

\* {

box-sizing: border-box;

}

body {

font-family: "Montserrat", sans-serif;

font-size: 12px;

line-height: 1em;

}

button {

background-color: transparent;

padding: 0;

border: 0;

outline: 0;

cursor: pointer;

}

input {

background-color: transparent;

padding: 0;

border: 0;

outline: 0;

}

input[type="submit"] {

cursor: pointer;

}

input::placeholder {

font-size: .85rem;

font-family: "Montserrat", sans-serif;

font-weight: 300;

letter-spacing: .1rem;

color: #ccc;

}

/\*\*

\* Bounce to the left side

\*/

@keyframes bounceLeft {

0% {

transform: translate3d(100%, -50%, 0);

}

50% {

transform: translate3d(-30px, -50%, 0);

}

100% {

transform: translate3d(0, -50%, 0);

}

}

/\*\*

\* Bounce to the left side

\*/

@keyframes bounceRight {

0% {

transform: translate3d(0, -50%, 0);

}

50% {

transform: translate3d(calc(100% + 30px), -50%, 0);

}

100% {

transform: translate3d(100%, -50%, 0);

}

}

/\*\*

\* Show Sign Up form

\*/

@keyframes showSignUp {

100% {

opacity: 1;

visibility: visible;

transform: translate3d(0, 0, 0);

}

}

/\*\*

\* Page background

\*/

.user {

display: flex;

justify-content: center;

align-items: center;

width: 100%;

height: 100vh;

background: #ccc;

background-size: cover;

}

.user\_options-container {

position: relative;

width: 80%;

}

.user\_options-text {

display: flex;

justify-content: space-between;

width: 100%;

background-color: rgba(34, 34, 34, 0.85);

border-radius: 3px;

}

/\*\*

\* Registered and Unregistered user box and text

\*/

.user\_options-registered,

.user\_options-unregistered {

width: 50%;

padding: 75px 45px;

color: #fff;

font-weight: 300;

}

.user\_registered-title,

.user\_unregistered-title {

margin-bottom: 15px;

font-size: 1.66rem;

line-height: 1em;

}

.user\_unregistered-text,

.user\_registered-text {

font-size: .83rem;

line-height: 1.4em;

}

.user\_registered-login,

.user\_unregistered-signup {

margin-top: 30px;

border: 1px solid #ccc;

border-radius: 3px;

padding: 10px 30px;

color: #fff;

text-transform: uppercase;

line-height: 1em;

letter-spacing: .2rem;

transition: background-color .2s ease-in-out, color .2s ease-in-out;

}

.user\_registered-login:hover,

.user\_unregistered-signup:hover {

color: rgba(34, 34, 34, 0.85);

background-color: #ccc;

}

/\*\*

\* Login and signup forms

\*/

.user\_options-forms {

position: absolute;

top: 50%;

left: 30px;

width: calc(50% - 30px);

min-height: 420px;

background-color: #fff;

border-radius: 3px;

box-shadow: 2px 0 15px rgba(0, 0, 0, 0.25);

overflow: hidden;

transform: translate3d(100%, -50%, 0);

transition: transform .4s ease-in-out;

}

.user\_options-forms .user\_forms-login {

transition: opacity .4s ease-in-out, visibility .4s ease-in-out;

}

.user\_options-forms .forms\_title {

margin-bottom: 45px;

font-size: 1.5rem;

font-weight: 500;

line-height: 1em;

text-transform: uppercase;

color: #e8716d;

letter-spacing: .1rem;

}

.user\_options-forms .forms\_field:not(:last-of-type) {

margin-bottom: 20px;

}

.user\_options-forms .forms\_field-input {

width: 100%;

border-bottom: 1px solid #ccc;

padding: 6px 20px 6px 6px;

font-family: "Montserrat", sans-serif;

font-size: 1rem;

font-weight: 300;

color: gray;

letter-spacing: .1rem;

transition: border-color .2s ease-in-out;

}

.user\_options-forms .forms\_field-input:focus {

border-color: gray;

}

.user\_options-forms .forms\_buttons {

display: flex;

justify-content: space-between;

align-items: center;

margin-top: 35px;

}

.user\_options-forms .forms\_buttons-forgot {

font-family: "Montserrat", sans-serif;

letter-spacing: .1rem;

color: #ccc;

text-decoration: underline;

transition: color .2s ease-in-out;

}

.user\_options-forms .forms\_buttons-forgot:hover {

color: #b3b3b3;

}

.user\_options-forms .forms\_buttons-action {

background-color: #e8716d;

border-radius: 3px;

padding: 10px 35px;

font-size: 1rem;

font-family: "Montserrat", sans-serif;

font-weight: 300;

color: #fff;

text-transform: uppercase;

letter-spacing: .1rem;

transition: background-color .2s ease-in-out;

}

.user\_options-forms .forms\_buttons-action:hover {

background-color: #e14641;

}

.user\_options-forms .user\_forms-signup,

.user\_options-forms .user\_forms-login {

position: absolute;

top: 70px;

left: 40px;

width: calc(100% - 80px);

opacity: 0;

visibility: hidden;

transition: opacity .4s ease-in-out, visibility .4s ease-in-out, transform .5s ease-in-out;

}

.user\_options-forms .user\_forms-signup {

transform: translate3d(120px, 0, 0);

}

.user\_options-forms .user\_forms-signup .forms\_buttons {

justify-content: flex-end;

}

.user\_options-forms .user\_forms-login {

transform: translate3d(0, 0, 0);

opacity: 1;

visibility: visible;

}

/\*\*

\* Triggers

\*/

.user\_options-forms.bounceLeft {

animation: bounceLeft 1s forwards;

}

.user\_options-forms.bounceLeft .user\_forms-signup {

animation: showSignUp 1s forwards;

}

.user\_options-forms.bounceLeft .user\_forms-login {

opacity: 0;

visibility: hidden;

transform: translate3d(-120px, 0, 0);

}

.user\_options-forms.bounceRight {

animation: bounceRight 1s forwards;

}

/\*\*

\* Responsive 990px

\*/

@media screen and (max-width: 990px) {

.user\_options-forms {

min-height: 350px;

}

.user\_options-forms .forms\_buttons {

flex-direction: column;

}

.user\_options-forms .user\_forms-login .forms\_buttons-action {

margin-top: 30px;

}

.user\_options-forms .user\_forms-signup,

.user\_options-forms .user\_forms-login {

top: 40px;

}

.user\_options-registered,

.user\_options-unregistered {

padding: 50px 45px;

}

}

.bcimg {

background-image: url('https://images.unsplash.com/photo-1579546929518-9e396f3cc809?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1170&q=80');

background-repeat: no-repeat;

}

### WORKING

Anyone can view Online Shopping portal and

available products, but every user must login by his/her Username

and password in order to purchase or order products. Unregistered

members can register by navigating to registration page. Only Admin

will have access to modify roles, by default developer can only be an

‘Admin’. Once user register site, his default role will be ‘User’.

HOMEPAGE:

The Home Screen will consist of screen were one can browse

through the products which we have on our website

Arrival:

This page consists of new product details. This page appears same

for both visitors and users.

Featured:

This page consists of featured products which can be manged by

admin of the website.

Contact Us Page:

Visitors and Registered users can contact website owners or

administrators from here.

ABOUT US PAGE:

This page describes about website and owners

REGISTER PAGE:

New users can register here.

LOGIN PAGE:

Login page for both users and administrators.

Admin Page:

Only difference you see in this page is Role: Admin. User and Admin

role will be checked once the page was login and Session [“role”]

will be either Admin or User. If credentials belong to Admin then

role will be Admin and if credentials belong to User then role will be

User.

ORDER VIEW FOR USER: Once users order item they are able to see

ordered products and grand total.

PAYMENT: Once users orders products they are redirected to

payment pag

### CONCLUSION AND FUTURE WORK

Conclusion:

As per the goal of this project an attempt is made to show how we can take a business online and also enable users to Order clothes online following B2C business model.

Positivae website acts as a platform between user (student or faculties) and sellers of Shopping complex of GLA university to provide Clothes online. Also, it provides support to sellers to take their business online and grow it.

#### Future work: -

The scope of clothes ordering in India bloomed during the pandemic. This success can also be the result of increased internet use among our generation.

Many different adaptations, tests, and experiments have been left for the future due to lack of time. The present system is just an interactive UI with user database but Future work concerns with developing an Admin panel and a data base to store products detail with user information.

Github Link:

https://github.com/pavangupta352/e-commerce-mini-project