### PROJECT TITLE

**A PROJECT REPORT**

***Submitted by***

### ARUN H 727821TUCS021

***in partial fulfillment for the award of the degree of***

### BACHELOR OF TECHNOLOGY

### IN

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

### June 2023

**BONAFIDE CERTIFICATE**

Certified that this project report **“PROJECT TITLE”** is the bonafide work of

### “ARUN H” who carried out the project work under my supervision.

#### SIGNATURE SIGNATURE

**Dr.R.VIDHYA**

#### (Industry Mentor Name) HEAD OF THE DEPARTMENT

Designation Associate Professor,

Department of Computer Science

and Engineering

Sri Krishna College of Technology,

Coimbatore-641042.

Certified that the candidates were examined by us in the Project Work Viva Voce examination held on at Sri Krishna College of Technology, Kovaipudur, Coimbatore -641042

### INTERNAL EXAMINER EXTERNAL EXAMINER

***ACKNOWLEDGEMENT***

### ACKNOWLEDGEMENT

First and foremost, we thank the **Almighty** for being our light and for showering his gracious blessings throughout the course of this project.

We are grateful to our beloved Principal **Dr.M.G. Sumithra M.E., Ph.D**. for her tireless and relentless support.

We extend our sincere thanks to our Head of the Department **Dr.R.Vidhya M.Tech., Ph.D.** for her encouragement and inspiration.

We are greatly indebted to our Industry Mentor **Mr.Manoj Kumar**  for his valuable guidance and suggestions in all aspects that aided us to ameliorate our skills.

We are thankful to all those who have directly and indirectly extended their help to us in completing this project work successfully.

# ABSTRACT



iv

## ABSTRACT

Nowadays most people are using computers and laptops in their day-to-day life. Some people are using laptops and some people use computers but the customer needs some specific specification for their laptops and computers to carry out customer needs. In laptops the customers invest an extra amount to modify or upgrade their laptops but in computers they can buy the components as they wish, they can select their own specific components for their needs but the customers can’t buy all their needed components in the one shop or platform. So the main objective of this project is to help customers to buy their needed hardware and components at a single place with affordable cost. The product produces customized PCs and sells them.

# TABLE OF CONTENTS



**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
| **CHAPTER.NO** | **TITLE** | **PAGE NO** |
|  | **ABSTRACT** | **iv** |
|  | **LIST OF FIGURES** | **vi** |
|  | **LIST OF ABBREVIATIONS** | **vii** |
| 1 | 1.2 OVERVIEW | 2 |
|  | 1.3 OBJECTIVE | 2 |
| 2 | LITERATURE SURVEY | 3 |
| 3 | SYSTEM SPECIFICATIONS | 5 |
| 4 | PROPOSED SYSTEM | 8 |
|  | 4.1 PROPOSED SYSTEM | 8 |
|  | 4.2 ADVANTAGES | 9 |
| 5 | METHODOLOGIES | 10 |
|  | 5.1 LOGIN PAGE | 11 |
|  | 5.2 SIGN UP PAGE | 12 |
|  | 5.3 FORGET PASSWORD | 13 |
|  | 5.4 PRODUCT PAGE | 13 |
|  | 5.5 CART PAGE | 14 |
| 6 | IMPLEMENTATION AND RESULT | 15 |
| 7 | CONCLUSION AND FUTURE SCOPE | 36 |
|  | 7.1 CONCLUSION | 36 |
|  | 7.2 FUTURE SCOPE | 37 |
| 8 | REFERENCES | 38 |

# LIST OF FIGURES



vi

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No** | **TITLE** | **Page No** |
| 5.1 | Process flow diagram | 10 |
| 5.2 | Login page flowchart | 11 |
| 5.3 | Signup page flowchart | 12 |
| 5.4 | Products page flowchart | 13 |
| 5.5 | Cartpage flowchart | 14 |
| 6.1 | Login page | 15 |
| 6.2 | Signup page | 16 |
| 6.3 | Recovery page | 17 |
| 6.4 | Otp verification | 17 |
| 6.5 | Home page | 18 |
| 6.6 | Product page | 19 |

# LIST OF ABBREVIATIONS



vii

### LIST OF ABBREVIATIONS

#### ABBREVIATIONS

**ACRONYMS**

**HTML** HYPERTEXT MARKUP LANGUAGE

**CSS** CASCADING STYLESHEET

**JS** JAVASCRIPT

**PC** PERSONAL COMPUTER

# INTRODUCTION



1

## CHAPTER 1 INTRODUCTION

This project is a development of a customized PC selling website. The website sells the customized and most powerful PC’s in the website. We offer a wide variety of customization services to help you make your PC your own. From adding custom cooling solutions to upgrading your graphics card, our experienced technicians have the knowledge and tools to help you get the most out of your machine. In addition, we also offer a range of custom parts and accessories to help you take your PC to the next level so if you're looking to make your PC stand out from the crowd, you've come to the right place. Check out our services today and start customizing your PC.

PC modification and gadgets have become increasingly popular in recent years. There is an ever-growing selection of components and devices available to customize and upgrade your computer. From the latest gaming graphics cards to an impressive array of cases, fans, and cooling systems, the options are almost limitless. Modifying your PC can push its performance to the limit and make it a powerhouse for gaming, video editing, and more. Gadgets are also a great way to enhance your computing experience. From specialized mice and keyboards to a variety of headsets and speakers, there are plenty of options to choose from.

Additionally, there are a number of gaming controllers, streaming devices, and other accessories that can boost your experience. No matter what your goals are, PC modification and gadgets can help you get the most out of your computing experience. Whether you’re looking for top-of-the-line performance or just want to add some cool features, there are plenty of options available to fit any budget. So, if you’re ready to take your PC to the next level, get ready to explore the world of PC modification and gadgets.

2

### PROBLEM STATEMENT

How can we create a website that allows customers to customize their own PCs based on their individual needs, while also providing them with the best possible customer service?

### OVERVIEW

This report discusses the result of the work done in development for the pc customization and e-commerce on react platform. It is a react project that mainly aims in providing a common platform for facilitating an online service of buying customized PCs at the affordable price.

### OBJECTIVE

The objective of the PC customisation ecommerce website is to provide customers with a platform to customize their PCs according to their individual needs and preferences. The website should be able to offer a wide range of components and peripherals, such as graphics cards, processors, RAM, hard drives, cooling systems, and cases, as well as allow customers to assemble and configure their PCs according to their specifications. Furthermore, the website should provide customers with access to reviews and advice from experts in the field, and provide services such as installation, maintenance, and troubleshooting.

# LITERATURE SURVEY



## CHAPTER 2 LITERATURE SURVEY

### 2.1 RELATED WORKS

1. An Analysis of E-Commerce Customization: A Literature Review by M.L. Moreno,

M. Ramos, and P.A. García .

A comprehensive overview of the literature concerning e-commerce customization, discussing its challenges, benefits, trends, and issues. The authors review the theoretical and empirical evidence that describes the motivations and usage of customization, the challenges of its implementation, and the effects of customization on customer satisfaction, loyalty, and profitability.

1. A Review of the Literature on E-Commerce Customization by A.B. de Oliveira, G.P. de Oliveira, and S.Q.M. de Oliveira.

The paper gives a comprehensive on e-commerce customization, discussing its challenges, benefits, trends, and issues. The authors analyze the theoretical and empirical evidence that describes the motivations and usage of customization, the challenges of its implementation, and the effects of customization on customer satisfaction, loyalty, and profitability.

1. E-Commerce Customization: A Literature Review by P.J.H.M. van den Besselaar and

M.P. van Leeuwen .

The comprehensive overview of the literature concerning e-commerce customization, discussing its challenges, benefits, trends, and issues. The authors analyze the theoretical and empirical evidence that describes the motivations and usage of customization, the challenges of its implementation, and the effects of customization on customer satisfaction, loyalty, and profitability.

1. E-Commerce Customization: A Literature Review by S.G. Nandu and A.K. Nandu .

This paper provides a comprehensive overview of the literature concerning e-commerce customization, discussing its challenges, benefits, trends, and issues. The authors review the theoretical and empirical evidence that describes the motivations and usage of customization, the challenges of its implementation, and the effects of customization on customer satisfaction, loyalty, and profitability.

1. The Impact of Customization on E-commerce: A Literature Review by M.W. Rahman and F. Rahman .

This paper reviews the literature on e-commerce customization, discussing its challenges, benefits, trends, and issues. The authors analyze the theoretical and empirical evidence that describes the motivations and usage of customization, the challenges of its implementation, and the effects of customization on customer satisfaction, loyalty, and profitability.

# SYSTEM SPECIFICATION



## CHAPTER 3 SYSTEM SPECIFICATION

In this chapter, we are gonna see the softwares that we have used to build the website. This chapter gives you a small description about the softwares used in the project.

### 3.1 VS CODE

Visual Studio Code is a source code editor developed by Microsoft for Windows, Linux, and macOS. It includes support for debugging, embedded Git control, syntax highlighting, intelligent code completion, snippets, and code refactoring. It is also customizable, so users can change the editor's theme, keyboard shortcuts, and preferences.

VS Code is an excellent code editor for React projects. It is lightweight, customizable, and has a wide range of features that make it ideal for React development. It has built-in support for JavaScript, JSX, and TypeScript, and enables developers to quickly move between files and view detailed type definitions. It also has a built-in terminal for running tasks. Additionally, VS Code has an extensive library of extensions that allow developers to quickly add features like code snippets, debugging tools, and linting support to their projects.

### 3.2.REACT

React is a JavaScript library created by Facebook for building user interfaces. It is a component-based, declarative, and highly efficient library that is used to develop interactive UIs (user interfaces) for single page web applications. React uses a virtual DOM (Document Object Model) that makes it faster and easier to manipulate the DOM

elements. It also provides declarative components that allow developers to write code that is easy to read and maintain. React also offers an extensive library of tools and components that make it easier to develop complex user interfaces.

### ROUTERS IN REACT

Routers are important components in React applications. They provide the ability to navigate between different views or components of the application. React Router is the most popular library to handle routing in React applications. It provides the ability to define routes, set up links, and render components based on the current route. It also provides features like data fetching, code-splitting, and server-side rendering.

### LOCAL STORAGE

Local storage is a type of web storage for storing data on the client side of a web browser. It allows websites to store data on a user’s computer, which can then be accessed by the website again when the user returns. Local storage is a more secure alternative to cookies because it allows websites to store data without having to send it back and forth with each request. Local storage is a key-value pair storage mechanism, meaning it stores data in the form of a key and corresponding value. It is similar to a database table in that it stores data in columns and rows, except that local storage stores the data in the browser rather than in a database. Local storage is often used to store user information such as preferences and settings, or to store data that is not meant to be shared with other websites. It is also used to cache data to improve the performance of a website. Local storage is supported by all modern web browsers, including Chrome,

Firefox, Safari, and Edge. It is accessible through the browser’s JavaScript API. Local storage is a powerful tool for websites to store data on the client side. It is secure, efficient, and can be used to store data that does not need to be shared with other websites.

Local Storage is a great way to improve the performance of a website by caching data. Local storage in web browsers allows website data to be stored locally on the user’s computer. It is a way of persistently storing data on the client side, which is not sent to the server with each request. This allows users to store data such as preferences, login information, and form data without needing to send it to a server. It is typically stored in a browser’s cookie file, but it can also be stored in other locations such as HTML5 Local Storage and IndexedDB. The data stored in local storage is persistent and can be accessed by the website even if the user closes the browser or navigates to another page. It is a great way for websites to store user-specific data, as it is secure, reliable, and fast. It is also a great way for developers to store data that does not need to be sent to the server with each request.

One of the key benefits of using local storage is its reliability. Unlike server-side storage, which can be affected by network outages or other server issues, local storage is stored locally on the user’s machine, and so is not affected by these issues. Another advantage of local storage is its speed. Because the data is stored locally, it is accessed quickly, as there is no need to send requests to a server. This makes it ideal for storing data that needs to be accessed quickly, such as user preferences or session data. Local storage is also secure, as the data is stored on the user’s machine and not on a server. This means that the data is not accessible by anyone other than the user, making it a good choice for storing sensitive information.

# PROPOSED SYSTEM



## CHAPTER 4 PROPOSED SYSTEM

This chapter gives a small description about the proposed idea behind the development of our website

### PROPOSED SYSTEM

The website has a modern, user-friendly design that is optimized for desktop and mobile devices. It features an easy-to-navigate structure with clear categories for parts, upgrades, installation services, and product reviews. The website includes user registration and login for purchasing products. The website offers customizable gaming Rigs and components for PC machines like Mouse, Keyboard and Monitor. It has a secure shopping cart system that allows customers to purchase items directly from the website. Customers should also be able to save items to their cart for future purchases. The website accepts major credit cards and other forms of payment such as PayPal and Apple Pay. Customers will be able to leave reviews and ratings for products and services that they have purchased or experienced. It should also have an integrated payment processing system to securely process payments. It also offers installation services for customers who need help installing their parts or upgrades. All the systems and components sold come under proper warranty. In case of defect or any kind of malfunctioning, the customers can contact the website through the About Us page. The products will be either repaired or replaced under warranty if it is not physically damaged. The website also has an email and chat-based customer support team available to answer any questions or concerns customers may have. The website should have a secure system in place to protect customers' personal information and payment details.

9

We also provide the cancellation facility to the customers to cancel the ordered item. The customer will be refunded the amount of the order within a week.

### ADVANTAGES

**·** Customization options: Customers can select specific components and configurations to build their ideal computer.

* Price transparency: Customers can see the cost of each component and the final price of their build, allowing them to make informed decisions.
* Compatibility assurance: The website will typically ensure that the components selected are compatible with each other, reducing the risk of compatibility issues.
* Time-saving: Customers can quickly and easily build their computer online, rather than having to visit multiple stores or websites to find the components they need.
* Quality assurance: The website can offer warranty and quality assurance on all the products and components provided, this can help customers to trust the website.
* Cost savings: Building a computer from individual components can often be less expensive than purchasing a pre-built system.

# METHODOLOGIES



## CHAPTER 5 METHODOLOGIES

This chapter gives a small description about how our system works.

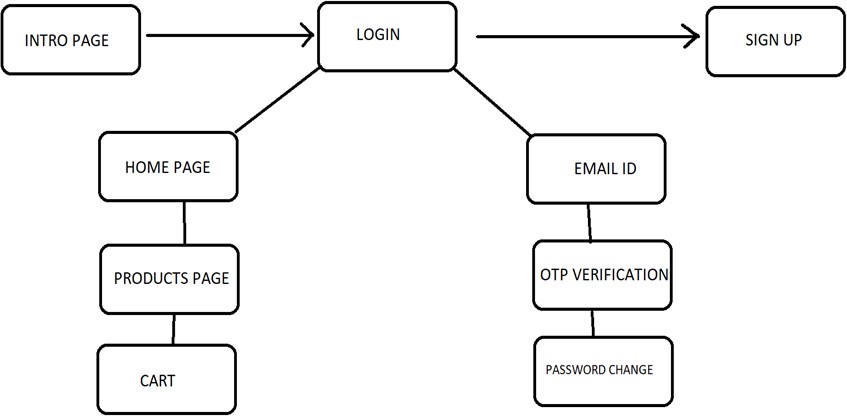


Fig 5.1 PROCESS FLOW DIAGRAM

### 5.1.LOGIN

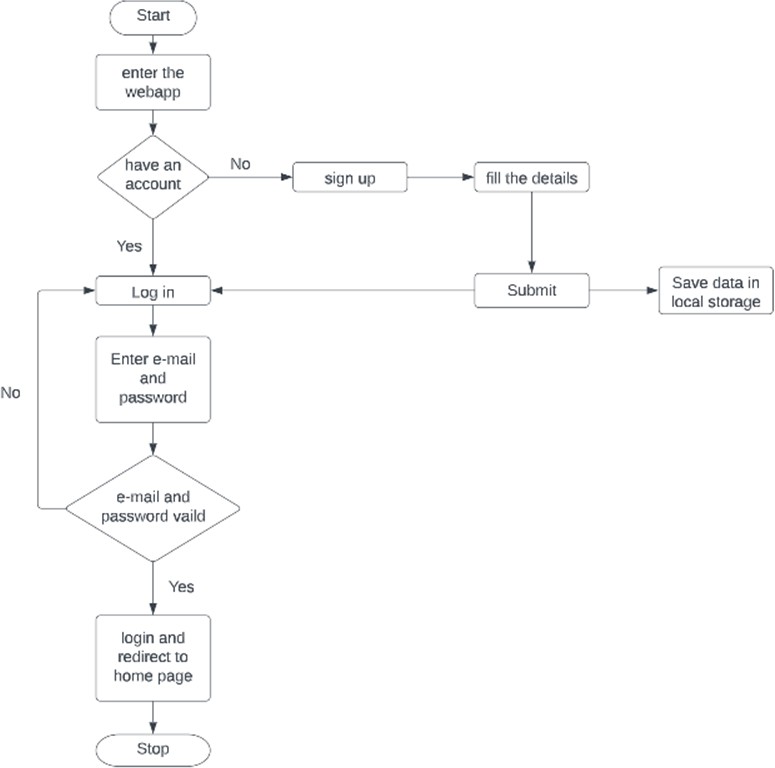


Fig 5.2. LOGIN PAGE FLOWCHART

In this page we will be asking about the username and password of the user. Firstly the website validates the user inputs. It verifies the username and password by checking it with the usernames and passwords stored in the local storage when the user creates an account in the website.

### 5.2 SIGNUP PAGE

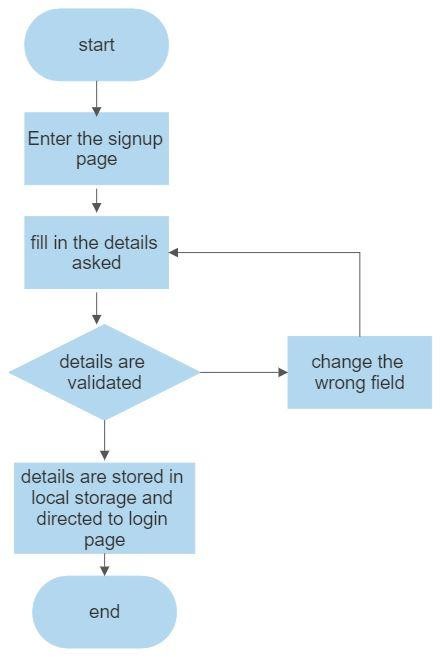


Fig 5.3 SIGNUP PAGE FLOWCHART

This page asks users about the basic details of the user to create an account. This page asks for details like username, password,email id, phone number. After the user enters the details , these details are then validated by our code . If all details are correct then the user is then directed to the login page

### FORGOT PASSWORD

If an existing user forgets the password he can access a component named forgot password where he will be directed for a page that’s requesting their email id, after entering the existing email id the user will be redirected to the email OTP verification, then the user can change their password.

### PRODUCTS PAGE

This page displays the list of products available for sale. These details have been displayed from the list of products available in the javascript file named data.js. So that if we want to add some products to a particular product we can easily do it by adding the details to the products js file.

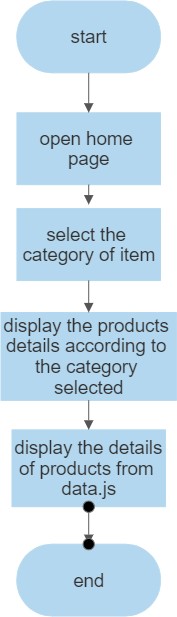


Fig 5.4 PRODUCTS PAGE FLOWCHART

14

### 5.5 CART PAGE

This page displays the set of products that the user selected by clicking the button “Add to cart” available on the products page. It also displays the products with their names and its price with quantity. This also helps you to increase the quantity of the products you have added using a plus button which is available on the page after every product's name. You can also remove or reduce the quantity by using the minus symbol button which is also available below every product's name. If the cart doesn’t contain any item it will display that the cart is empty.

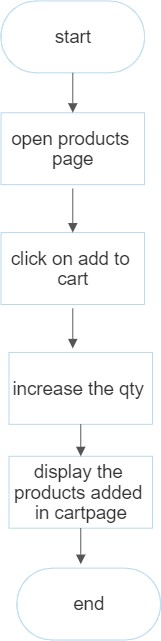


Fig 5.5 CART PAGE FLOWCHART

# IMPLEMENTATION AND RESULT



**CHAPTER 6**

**IMPLEMENTATION AND RESULT**

This chapter gives a description about the output that we produced by developing the website of our idea.

### 6.1 LOGIN

When User enters our website he will be asked about his login details like email id and password. The login details will be verified with the details given while the user creates an account.

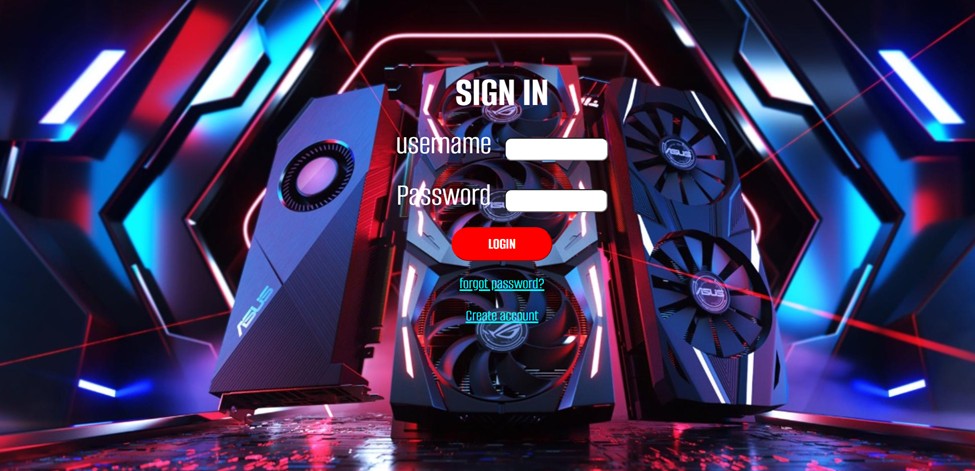


Fig 6.1 LOGIN PAGE

### SIGN UP

If a user doesn’t have an account on the website, User can use a component named create new account available in the login page. When the user clicks on that he will be redirected to the signup page. In sign up he should fill up his email id, password and phone number. These inputs will be validated.



Fg 6.2 SIGNUP PAGE

### FORGOT PASSWORD

If an existing user forgets the password he can access a component named forgot password where he will be directed for a page that’s requesting their email id, after entering the existing email id the user will be redirected to the email OTP verification, then the user can change their password.

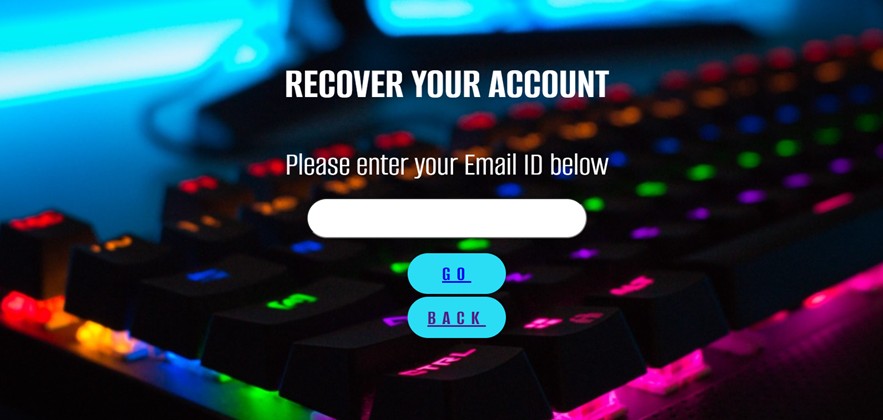


Fig 6.3 RECOVERY PAGE

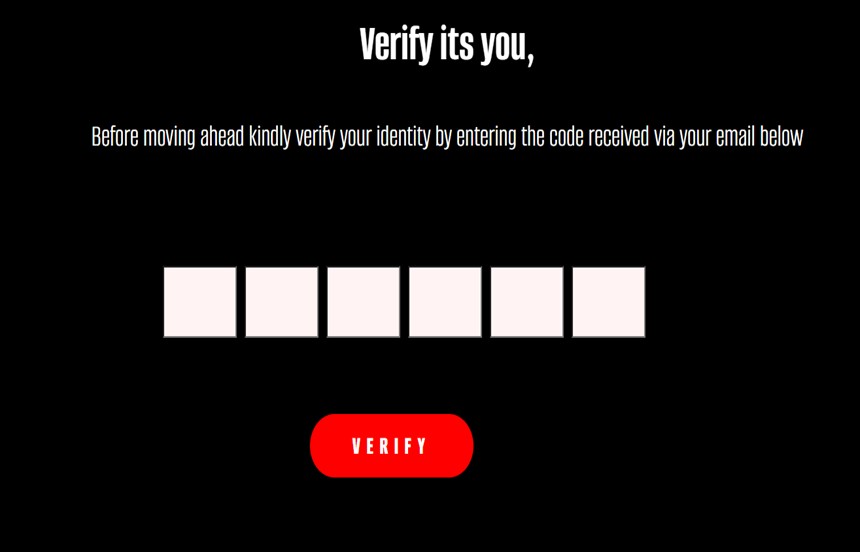


Fig 6.4 OTP VERIFICATION

### CATEGORIES OF PRODUCTS

The webpage has different categories of products available for sale and users can redirect to the products description page by selecting the respective products. This page also contains the main taglines of the whole website idea.



Fig 6.5 HOME PAGE

### LIST OF PRODUCTS

This page displays the list of products available for sale. These details have been displayed from the list of products available in the javascript file named data.js[1]. So that if we want to add some products to the particular product we can easily do it by adding the details to the products js file.

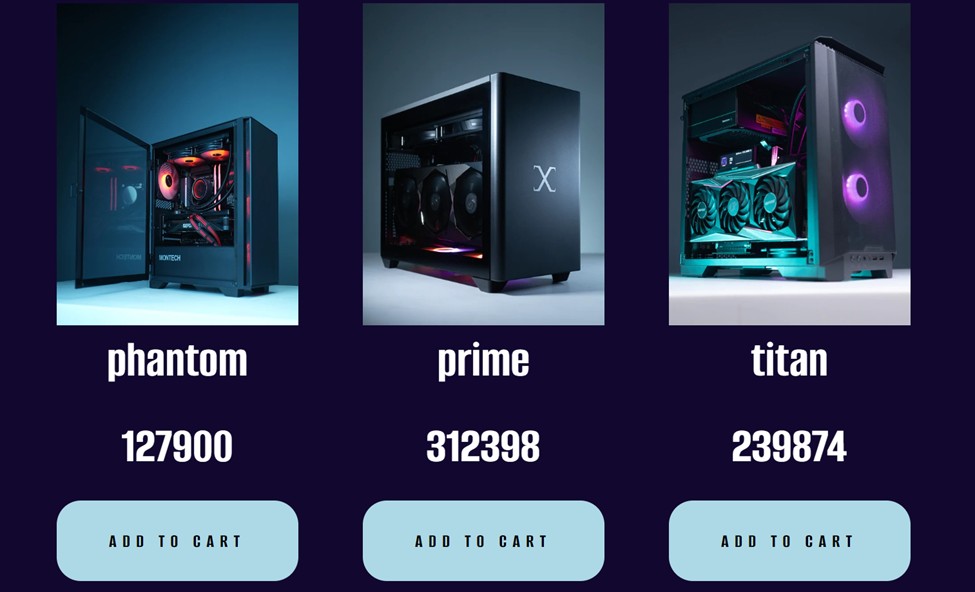


Fig 6.6 PRODUCT PAGE

### CART PAGE

This page displays the set of products that the user selected by clicking the button “Add to cart” available on the products page. It also displays the products with their names and its price with quantity. This also helps you to increase the quantity of the products you have added using a plus button which is available on the page after every product's name[1]. You can also remove or reduce the quantity by using the minus symbol button which is also available below every product's name. If the cart doesn’t contain any item it will display that the cart is empty.

### 6.7 CODING

#### Login:

const Login = () => {

const history = useNavigate();

const [inpval, setInpval] = useState({ email: "",

password: ""

})

const [data, setData] = useState([]); console.log(inpval);

const getdata = (e) => { console.log(e.target.value);

const { value, name } = e.target; console.log(value,name); setInpval(() => {

return {

...inpval, [name]: value

}

})

}

const addData = (e) => { e.preventDefault();

const getuserArr = localStorage.getItem("user\_Signup"); console.log(getuserArr);

const { email, password } = inpval; if (email === "") { toast.error('email field is requred', {

position: "top-center",

});

} else if (!email.includes("@")) {

toast.error('plz enter valid email addres', { position: "top-center",

});

} else if (password === "") { toast.error('password field is requred', {

position: "top-center",

});

} else if (password.length < 9) { toast.error('password length greater 8', {

position: "top-center",

});

}

else {

if (getuserArr && getuserArr.length) {

const userdata = JSON.parse(getuserArr); const userlogin = userdata.filter((el, k) => {

return el.email === email && el.password === password

});

if (userlogin.length === 0) { alert("invalid details")

} else {

console.log("user login succesfulyy"); localStorage.setItem("user\_login", JSON.stringify(userlogin)) history('/front')

}

}

}

}

return (

<>

<div className='mainlogin'>

<div className="container mt-3">

<section className='d-flex justify-content-between'>

<div className="left\_data mt-3 p-3" style={{ width: "100%" }}>

<div className='s1'>

<h1 className='text-center col-lg-10'>SIGN IN</h1>

<Form >

<Form.Group className="mb-3 col-lg-6" controlId="formBasicEmail">

<Form.Control type="email" name='email' onChange={getdata}

placeholder="Enter email"

style={{width:'300px',height:'40px',fontSize:'25px',borderRadius:'30px'}} />

</Form.Group>

<br/>

<Form.Group className="mb-3 col-lg-6" controlId="formBasicPassword">

<Form.Control type="password" name='password'

onChange={getdata} placeholder="Password" style={{width:'300px',height:'40px',fontSize:'25px',borderRadius:'30px'}}/>

</Form.Group>

<br/>

<Button variant="primary" className='col-lg-6' onClick={addData} type="submit"

style={{fontSize:"18px",fontWeight:"bold",background:"red",color:"white",borderRadiu s:"50px",height:"40px",width:"110px",borderColor:"red"}}>

SUBMIT

</Button>

<br/><br/>

</Form>

<Link to="/recoverpage"style={{color:'aqua',fontSize:'20px'}}>forgot password?</Link><br/><br/>

<Link to="/signup"style={{color:'aqua',fontSize:'20px'}}>Create account</Link>

</div>

</div>

</section>

<ToastContainer />

</div>

</div>

</>

)

}

export default Login;

#### Sign up:

const Register = () => {

const history = useNavigate();

const [inpval, setInpval] = useState({ name: "",

email: "",

phone\_no: "", password: ""

})

const [data,setData] = useState([]); console.log(inpval);

const getdata = (e) => {

// console.log(e.target.value); const { value, name } = e.target;

// console.log(value,name);

setInpval(() => { return {

...inpval, [name]: value

}

})

}

const addData = (e) => { e.preventDefault();

const { name, email, phone\_no, password } = inpval; if (name === "") {

toast.error(' name field is requred!',{ position: "top-center",

});

} else if (email === "") { toast.error('email field is requred',{

position: "top-center",

});

} else if (!email.includes("@")) { toast.error('plz enter valid email addres',{

position: "top-center",

});

} else if (phone\_no === "") { toast.error('phone number field is requred',{

position: "top-center",

});

}else if (phone\_no.length < 10 || phone\_no.length>10) { toast.error('enter a valid phone number',{

position: "top-center",

});

}else if (password === "") { toast.error('password field is requred',{

position: "top-center",

});

} else if (password.length < 9) { toast.error('password length greater 8',{

position: "top-center",

});

} else {

console.log("data added succesfully"); history("/login")

localStorage.setItem("user\_Signup",JSON.stringify([...data,inpval]));

}

}

return (

<>

<div className='mainbox'>

<div className="container mt-3">

<section className='d-flex justify-content-between'>

<div className="left\_data mt-3 p-3" style={{ width: "100%" }}>

<div className='s2'></div>

<center>

<h1 className='text-center col-lg-6'>SIGN UP</h1>

<Form >

<Form.Group className="mb-3 col-lg-6" controlId="formBasicEmail">

<Form.Control type="text" name='name' onChange={getdata}

placeholder="Enter Your Name" style={{width:'300px',height:'40px',fontSize:'25px',borderRadius:'30px'}}/>

</Form.Group>

<br/>

<Form.Group className="mb-3 col-lg-6"

controlId="formBasicEmail">

<Form.Control type="email" name='email' onChange={getdata}

placeholder="Enter email"

style={{width:'300px',height:'40px',fontSize:'25px',borderRadius:'30px'}}/>

</Form.Group>

<br/>

<Form.Group className="mb-3 col-lg-6"

controlId="formBasicEmail">

<Form.Control onChange={getdata} name='phone\_no' type="text"

placeholder="phone number"

style={{width:'300px',height:'40px',fontSize:'25px',borderRadius:'30px'}}/>

</Form.Group>

<br/>

<Form.Group className="mb-3 col-lg-6"

controlId="formBasicPassword">

<Form.Control type="password" name='password'

onChange={getdata} placeholder="Password" style={{width:'300px',height:'40px',fontSize:'25px',borderRadius:'30px'}}/>

</Form.Group>

<br/><Button className='submit' onClick={addData} type="submit"style={{background:"black",color:"lime",borderRadius:"50px",border:"6p x solid lime", width:'150px',fontSize:'25px'}} >

SUBMIT

</Button><br/>

<Button className="loginbtn" variant="contained" style={{background:"black",color:"red",borderRadius:"50px",border:"6px solid red", width:'150px', fontSize:'25px'}} onClick={()=>{history("/login")}}>BACK</Button>

</Form>

</center>

</div>

</section>

<ToastContainer />

</div>

</div>

</>

)

}

export default Register;

#### Product page:

export default function Product(props) { const{product,onAdd}=props;

return (

<main>

<div className='productpage'>

<div className='card'>

<div className='images'>

<img className="small" src={product.img} alt={product.name}></img>

</div>

<div className='details'>

<h1>{product.name}</h1><br />

<h1>{product.price} rupees</h1>

<Link to="/cart"><button className='addtocartbutton' onClick={()=>onAdd(product)}>ADD TO CART</button></Link>

</div>

</div>

</div>

</main>

)

}

#### Cart page:

export default function Cart(props) {

const {cartItems, onAdd,onRemove}=props;

const itemsPrice =cartItems.reduce((a,c)=>a+c.price\*c.qty,0);

return (

<div className='cartpage'>

<div className='namesidehalfproductcart'>

<h1>CART</h1>

</div>

<div className='rightsidehalfproductcart'>

{cartItems.length ===0 && <div><h2 className='cartempty'>Cart is empty</h2></div>}

{cartItems.map((item)=>(

<div key={item.id} className="row">

<div><h2 className='details\_name'>{item.name}</h2></div>

<div>

<button onClick={()=>onAdd(item)} className="Add">+</button>

<button onClick={()=>onRemove(item)} className="Remove">-</button>

</div>

<div>

{item.qty} x {item.price.toFixed(2)} rupees

</div>

<div>

{itemsPrice} rupees

</div>

</div>

))}

</div>

</div>

)

}

# CONCLUSION



## CHAPTER 7 CONCLUSION

This chapter tells about the conclusion that anyone can drive from the project and the learning we learnt by taking over this project.

### CONCLUSION

The completion of the custom PC selling website project is an important milestone for our company. We have worked hard to build a website that is easy to use, secure, and fulfills the needs of our customers. We have provided our customers with a fast, reliable, and secure way to purchase custom PCs and have given them access to a wide variety of components to choose from. We are proud of the website we have created and are looking forward to a successful launch. We have been able to make the website user-friendly and secure by implementing the latest security measures and using the most up-to-date coding standards. We have also been able to provide our customers with a wide selection of components to choose from, making it easier for them to find the perfect custom PC. We have also provided them with helpful guides and tutorials to make the purchasing process as easy as possible. Overall, we are proud of the work we have done on the custom PC selling website project. We have been able to create a website that is secure, user-friendly, and has a wide selection of components to choose from. We look forward to a successful launch and continued success in the future.

### FUTURE SCOPE

Customized PC Selling Website project has a great potential for the future. With the advancements in technology and the ever-growing demand for more powerful and efficient PCs, this project has the potential to become one of the most successful websites in the world. With the help of customized options, customers can design their own PCs with the exact specifications

that they need. This will enable them to get the best performance for the lowest price. Moreover,

customers can also make use of various add-ons which can enhance the overall performance of their PCs.

The website can also provide customers with information regarding the latest hardware, software and other components which can help them make an informed decision. It can also provide them with tips and tricks on how to get the most out of their PCs. Furthermore, customers can also avail of discounts and offers which can help them save on their purchase.

In addition to this, the website can also provide customers with detailed reviews on the products they are interested in. This will help them make an informed decision and ensure they get the best product for their needs. Furthermore, the website can also provide customers with an online community where they can discuss their experiences and share their opinions regarding the products they have purchased.

# REFERENCES



### REFERENCES

* + 1. GeeksforGeeks, “Types of E-commerce”, <https://www.geeksforgeeks.org/types-of-e-commerce/>
    2. Shopify, “How to Create a Custom PC Building Website”, <https://www.shopify.ca/blog/custom-pc-building-website>
    3. Smashing Magazine, “Designing For E-Commerce”, <https://www.smashingmagazine.com/2016/11/designing-for-e-commerce/>
    4. TutsPlus, “How to Create an Ecommerce Website with WordPress and WooCommerce”,

https://webdesign.tutsplus.com/tutorials/how-to-create-an-ecommerce-web site-with-wordpress-and-woocommerce--cms-31938