

AY 2023-24

III Semester

CSL306

**A PROJECT REPORT
ON**

3rd Eye for Blind and Blindkart Website

Bachelor of Technology
in
School of Computing

By
AYUSH (22118)



SCHOOL OF ELECTRONICS/COMPUTING

**INDIAN INSTITUTE OF INFORMATION TECHNOLOGY UNA
HIMACHAL PRADESH**

DECEMBER 2023

BONAFIDE CERTIFICATE

This is to certify that the project titled *3rd Eye for Blind and Blindkart Website* is a bonafide record of the work done by

Ayush (22118)

in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in BRANCH NAME of the INDIAN INSTITUTE OF INFORMATION TECHNOLOGY UNA, HIMACHAL PRADESH, during the year 2022 - 2026.

under the guidance of

Dr.Neha Sharma

Project viva-voce held on: _____

Internal Examiner

External Examiner

ORIGINALITY / NO PLAGIARISM DECLARATION

We certify that this project report is our original report and no part of it is copied from any published reports, papers, books, articles, etc. We certify that all the contents in this report are based on our personal findings and research and we have cited all the relevant sources which have been required in the preparation of this project report, whether they be books, articles, reports, lecture notes, and any other kind of document. We also certify that this report has not previously been submitted partially or as whole for the award of degree in any other university in India and/or abroad.

We hereby declare that, we are fully aware of what constitutes plagiarism and understand that if it is found at a later stage to contain any instance of plagiarism, our degrees may be cancelled.

Ayush (22118)

ABSTRACT

This project introduces an innovative solution aimed at enhancing the mobility and accessibility of visually impaired individuals. This project has an object detection mechanism using an ultrasonic sensor and a feedback system with a buzzer. Beside of that it creates visually impaired-friendly website for shopping.

The hardware component employs ultrasonic sensors. The ultrasonic sensor measures the distance to objects, providing real-time data to the microcontroller. Upon detection of obstacles, a feedback system is initiated, featuring a buzzer to alert the user, aiding in timely obstacle avoidance.

In the software component, the project extends its functionality to an shopping website. Users can independently purchase hardware components voice navigation and receive auditory feedback for each key press. This inclusive approach not only enhances the overall user experience but also empowers individuals with visual impairments to independently explore and shopping.

In summary, this project provides a holistic solution by combining a hardware-based object detection system with a feedback mechanism and an shopping website. By addressing both physical and digital aspects of accessibility, the initiative aims to significantly improve the daily lives of visually impaired individuals.

ACKNOWLEDGEMENT

We would like to thank the following people for their support and guidance without whom the completion of this project in fruition would not be possible.

We would like to express our sincere gratitude and heartfelt thanks to [Project Guide Name] for their unflinching support and guidance, valuable suggestions and expert advice. Their words of wisdom and expertise in subject matter were of immense help throughout the duration of this project.

We also take the opportunity to thank our Director and all the faculty of School of Computing/Electronics, IIIT Una for helping us by providing necessary knowledge base and resources.

We would also like to thank our parents and friends for their constant support.

Ayush (22118)

TABLE OF CONTENTS

Title	Page No.
ABSTRACT.....	iii
ACKNOWLEDGEMENT.....	iv
TABLE OF CONTENTS.....	v
LIST OF ACRONYMS.....	vi
LIST OF TABLES.....	vii
LIST OF FIGURES.....	viii
1 Introduction.....	1
1.1 Overview of Project.....	1
1.1.1 Object detector mini project.....	1
1.1.2 Visually impaired friendly website.....	1
2 Review of Literature	3
2.1 Arduinio mini project.....	3
2.2 Website.....	4
3 Methods and Result.....	5
3.1 Arduino Mini Project.....	5
3.1.1 Components Required.....	5
3.1.2 Project Setup.....	6
3.2 Website.....	7
3.2.1 Project Setup.....	8
4 Future.....	12
4.1 3rd Eye for blind future improvement.....	12
4.2 Blindkart Website.....	13

References.....	14
Appendices.....	15

LIST OF ACRONYMS

HTML	Hyper Text Markup Language
CSS	Cascading Style Sheets
JS	Javascript
JSON	Javascript Object Notation
DOM	Document Object Model
SQL	Structured Query Language
CLI	Command Line Interface

LIST OF TABLES

3.1	MySql order table.....	11
-----	------------------------	----

LIST OF FIGURES

2.1	Example project.....	3
3.1	Arduino.....	5
3.2	HC-SR04 ultrasonic sensor.....	6
3.3	Buzzer.....	6
3.4	Circuit for hardware setup.....	7
3.5	Front Page.....	10
3.6	Billing Page.....	10
3.7	Thank you page.....	11

Chapter 1

Introduction

1.1 Overview of project

In a world where visual signals are frequently crucial, people with visual impairments encounter particular difficulties when navigating their environment. The coming together of innovation and technology has made it possible to address these issues and work toward making the world a more welcoming place for all. One such effort is the creation of an object detector mini project and has a website that is accessible to the blind for shopping.

1.1.1 Object detector mini project

This project aims to empower people with visual impairments by utilizing proximity detection and audio notifications to help them navigate their environment more confidently and easily.

The buzzer alerts the user to the presence of an impediment by producing unique noises when it detects one within a predetermined range. An inconspicuous but incredibly useful navigational assistance for people with visual impairments is created by the smooth integration of ultrasonic detection and audio feedback, which improves their spatial awareness and encourages independent mobility.

1.1.2 Visually impaired friendly website

In the era of digital commerce, online shopping has become an integral part of our daily lives, offering convenience and accessibility to a wide range of products. However, for individuals with visual impairments, traditional online shopping platforms can be a formidable challenge. Recognizing the need for this, we present an initiative – the Visually

Impaired Shopping Website, a platform that is designed to cater to the unique needs of users with visual impairments.

Clear and instructive voice prompts follow every click and mouseover action on the website, providing users quick audio feedback. With the help of this voice-guided navigation, people who are visually impaired can confidently make accurate choices by easily exploring product categories.

In addition to voice navigation, the website uses inclusive and interactive typing. A clear and audible sound is produced in response to each keystroke, improving the user experience by giving them instant feedback on their input. This feature makes typing more precise and effective, but it also makes online interactions easier and multisensory.

Chapter 2

Review of Literature

2.1 Arduino mini project

The use of ultrasonic sensors to identify obstacles in the surroundings and deliver immediate feedback via an Arduino-based system is demonstrated by the authors in [1][2]. This project illuminates the capability of Arduino and ultrasonic sensors to improve accessibility for individuals with visual impairments. Furthermore, the work presented in [3] showcases the implementation of ultrasonic sensors in robotics, utilizing Arduino for effective obstacle avoidance. The findings in [4] contribute significantly to establishing a foundation for the project.

These types of projects are available on ecommerce platforms like amazon ,flipkart,etc.

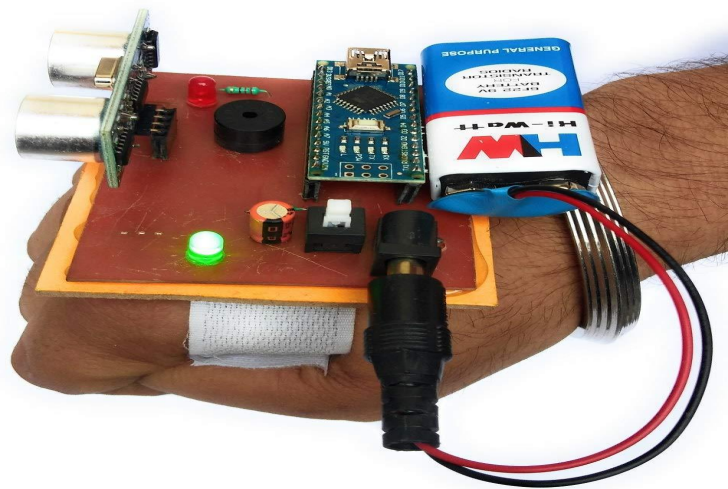


Figure 2.1: Example project.

2.2 Website

I was able to create the front end of websites because of the source that taught me HTML and CSS [5]. I implemented functions like data transmission between files, dynamic element addition, and audio playback using JavaScript [6]. I gained knowledge about text-to-speech processing from [9]. I was able to seamlessly transfer data to MySQL databases by connecting static websites to backends [7]. I established server connections using MySQL commands [8]. With the help of this all-inclusive learning path, I was able to gain a comprehensive understanding of front-end and back-end development, which helped me create dynamic, data-driven web applications.

Chapter 3

Methods and Result

3.1 Arduino Mini Project

The ultrasonic sensor works on the principle of sending and receiving ultrasonic waves to measure the distance between the sensor and an object. The sensor has a transmitter that sends out ultrasonic pulses and a receiver that detects the reflected waves. By measuring the time it takes for the waves to travel to the object and back, the distance can be calculated using the formula: $\text{Distance} = (\text{Speed of Sound} * \text{Time}) / 2$.

By measuring the distance, instruction is set to high the tone of buzzer according to distance measured.

3.1.1 Components Required:

- **Arduino:**

The microcontroller processes the data from the ultrasonic sensor and controls the buzzer based on the detected distance. Arduino, with its user-friendly programming environment, is commonly used for this purpose.



Figure 3.1: Arduino.

- **Ultrasonic Sensor:**

The HC-SR04 ultrasonic sensor is employed to measure the distance between the user and potential obstacles. It emits ultrasonic waves and calculates the time taken for the waves to bounce back after hitting an obstacle, determining the distance.



Figure 3.2: HC-SR04 ultrasonic sensor.

- **Buzzer:**

A piezo buzzer is integrated into the system to provide auditory feedback to the user. The intensity and frequency of the sound vary based on the proximity of obstacles, aiding the user in understanding their surroundings.



Figure 3.3:Buzzer.

3.1.2 Project Setup:

1. Connect the VCC pin of the ultrasonic sensor to the 5V pin on the Arduino.
2. Connect the GND pin of the ultrasonic sensor to the GND pin on the Arduino.
3. Connect the Trig pin of the ultrasonic sensor to digital pin 9 on the Arduino.
4. Connect the Echo pin of the ultrasonic sensor to digital pin 10 on the Arduino.
5. Connect one leg of the buzzer to digital pin 11 on the Arduino.
6. Connect the other leg of the buzzer to the GND pin on the Arduino.

3.1.1 Images

The images created with the help of TinkerCAD.com are presented in Fig. 3.1.

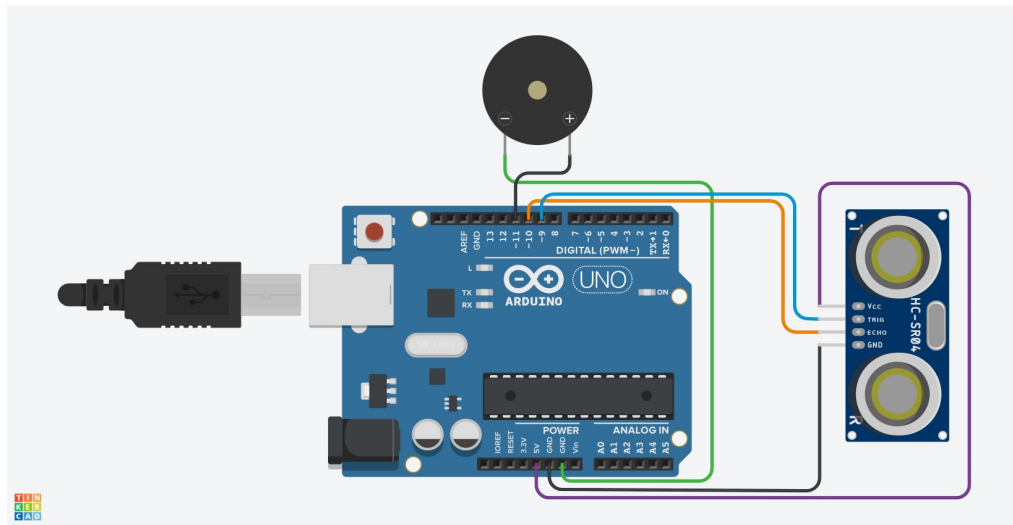


Figure 3.4: Circuit for hardware setup.

3.2 Website

The website's functionality is driven by a combination of user keyboard inputs (keypress) and internal events within the website.

The front end is crafted using basic HTML and CSS to create a visually appealing interface.

Within the HTML structure, I created div blocks to organize content, inserted images to enhance visual appeal, and utilized input tags for user inputs. Buttons were added for submission and managing product interactions within the cart.

CSS was applied to style the elements, introducing colors, resizing images, and providing a cohesive layout. Additionally, I implemented color-changing capabilities to enhance the interactive experience. Positioning was carefully adjusted for each block to ensure proper alignment and a visually pleasing display.

The addition of dynamic features is accomplished through JavaScript, including sound production triggered by clicks and mouseovers, error handling to manage an empty cart, fetching and dynamically inserting data from local storage, implementing animated responses to keypress events for an onscreen keyboard, and sending data to a billing form via JSON file and URL parameter.

To handle server-side operations, Node.js and Express.js are employed to host the server locally. Various modules such as MySQL, BodyParser, and Express are utilized. MySQL is responsible for storing user data, encompassing product details and contact information for product delivery. This seamless integration enables the website to efficiently collect and organize user data. Server requests are managed using the Fetch keyword, ensuring smooth communication between the client and server components. In summary, the combined use of HTML, CSS, JavaScript, Node.js, Express.js, and MySQL results in a fully functional website with an interactive front end and robust server-side capabilities.

3.2.1 Project Setup:

HTML and CSS:

- Create a basic shopping website with HTML and CSS.
- Integrate the keyboard layout from <https://codepen.io/gschier/pen/VKgYaY>.

Text-to-Speech (TTS):

- Generate audio files using the TTS feature from a website like <https://ttsmaker.com/>.

Image Files:

- Save all required image files for the website.

JavaScript:

- Implement JavaScript to play audio and add animations on the website.
- Send requests to the billing page.

Billing Page:

- Design a billing page and handle requests from the main page.
- Redirect to the backend upon submission.

Backend (Node.js and Express):

- Create a backend using Node.js and Express.js.
- Host the website on localhost.

MySQL Server:

- Set up a MySQL server.
- Create tables with desired fields to store billing information.

Backend Logic:

- Implement JavaScript logic in the backend to fetch data from the billing page.
- Use MySQL commands to store data in the MySQL server.

Thank You Page:

- Design a "Thank You" page for successful submissions.

Testing:

- Test the website locally to ensure functionality.

3.2.1 Images

The image is front page of the shopping website is presented in Fig. 3.5.



Figure 3.5: Front Page.

The image on the billing page of the shopping website is presented in Fig. 3.6.

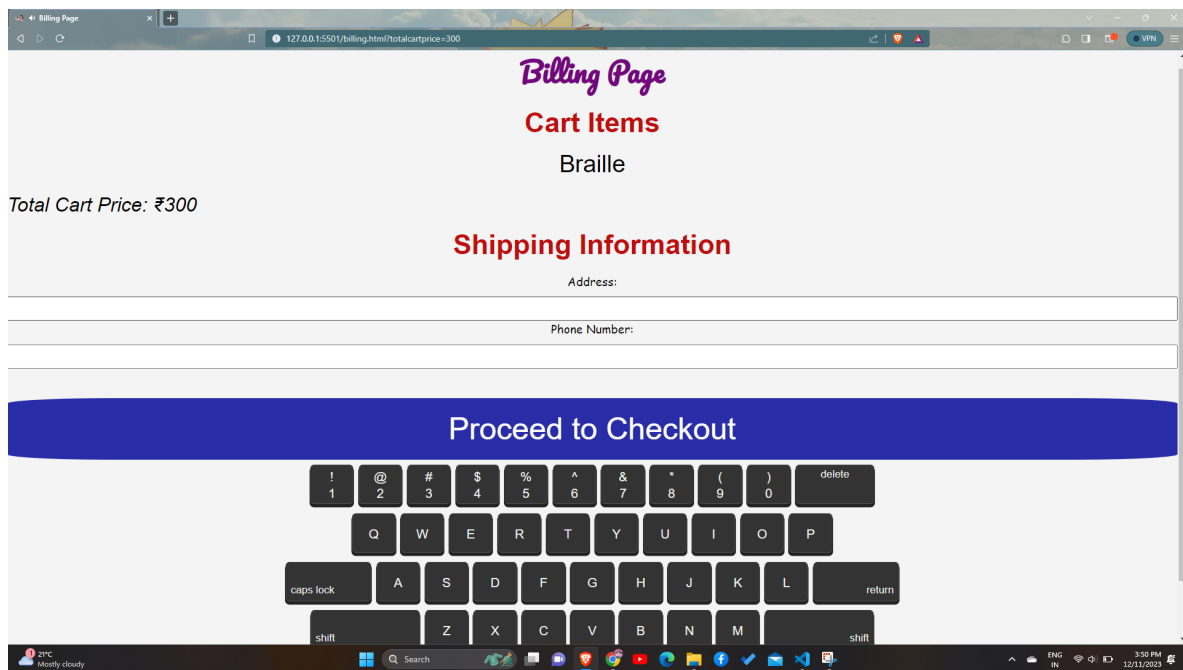


Figure 3.6: Billing Page.

The image is thankyou page(confirmation page) of the shopping website is presented in Fig. 3.7.



Figure 3.7: Thank you page.

2.1.2 Tables

The tables are created in mysql server for order details.

total_price	cart_items	address	phone
1200.00	Braille,Keyboard,Myproject	iiit una,himachal	9932122345
1400.00	Myproject stick Keyboard	hanuman nagar ,bihar	7352677279
700.00	Braille Keyboard	faridabad,ncr	9797845269

Table 3.1: MySql order table

Future:

4.1 3rd Eye for blind future improvement:

a) Miniaturization:

- Work on making the device smaller and more compact, possibly by using sequential circuit designs.

b)Integration with Other Sensors:

- Consider incorporating additional sensors, such as cameras ,GPS ,to provide more detailed environmental information.

c)Improved Obstacle Detection:

- Experiment with advanced signal processing techniques to improve obstacle detection accuracy and use long distance sensor like lidar sensor,IR sensor.
- Incorporate machine learning algorithms to better recognize different types of obstacles or complex environments.

d)Smart Navigation Assistance:

- Develop algorithms that can analyze the environment and suggest optimal paths or directions to the user.

e)Wireless Connectivity:

- Integrate wireless connectivity (Bluetooth, Wi-Fi) to enable communication with smartphones or other devices, opening up possibilities for remote assistance, map data, or additional functionalities.

f)Accessibility Features:

- Implement features like voice commands or gesture recognition for hands-free operation.

- Explore compatibility with existing assistive technologies and accessibility standards.

4.2 Blindkart Website

a)Voice Search and Commands:

- Implement advanced voice search functionality to allow users to find products and navigate the website using voice commands.

b)Customizable Text and Display Settings:

- Provide options for users to customize text size, color contrast, and overall display settings based on their preferences.

c)AI-Powered Personalized Recommendations:

- Implement artificial intelligence algorithms to provide personalized product recommendations based on user preferences and previous purchases.

d)Accessible Product Information:

- Include detailed audio descriptions for products, including size, color, material, and any other relevant information.

References

- [1] A. Yadav, M. Kumar, V. Gupta, and S. Bhushan, "Arduino Based Third Eye for Blind People," IJRASET, vol. 10, pp. 1-2, May 2022. Available: <https://doi.org/10.22214/ijraset.2022.43422>.
- [2] H. Kumar, D. G, and N. J, "An IoT based Obstacle Detection and Alerting System in Vehicles using Ultrasonic Sensor," An IoT based Obstacle Detection and Alerting System in Vehicles using Ultrasonic Sensor, vol. 5, issue 20, pp. 1-2, 2017. Available: <https://www.ijert.org/research/an-iot-based-obstacle-detection-and-alerting-system-in-vehicles-using-ultrasonic-sensor-IJERTCONV5IS20002.pdf>.
- [3] R. Mothe, S. T. Reddy, G. Sunil, and C. Siddhardha, "An IoT Based Obstacle Avoidance Robot Using Ultrasonic Sensor and Arduino," Warangal, India, pp. 9-10, October 2020. Available: [10.1088/1757-899X/981/4/042002](https://doi.org/10.1088/1757-899X/981/4/042002).
- [4] A. Carullo and M. Parvis, "An ultrasonic sensor for distance measurement in automotive applications," IEEE Sensors Journal, vol. 1, no. 2, pp. 143-147, August 2001. Available: https://www.researchgate.net/publication/3430936_An_ultrasonic_sensor_for_distance_measurement_in_automotive_applications.
- [5] Thomas A. Powell, "Introduction to HTML5, Introduction to CSS" HTML & CSS: The Complete Reference, vol. 5, pp. 055-663, 2010. Available: <https://www.dcehvp.com/E-Content/BCA/BCA-II/Web%20Technology/the-complete-reference-html-css-fifth-edition.pdf>
- [6] Dr. Axel Rauschmayer, "Using JavaScript: the big picture, Syntax," Abbrev. JavaScript For Impatient Programmers (ES2022 edition), pp. 073-183, 2022. Available: <https://exploringjs.com/impatient-js/downloads/impatient-js-preview-book.pdf>.
- [7] Sufiyan Akbani. "How to Connect Node JS with MySQL Database," perfcient, August 3rd, 2023. Available: <https://blogs.perfcient.com/2023/08/03/how-to-connect-node-js-with-mysql-database/>.
- [8] Kodrat Iman Satoto, R. Rizal Isnanto, Rinta Kridalukmana, "Optimizing MySQL Database System on Information Systems Research," ICITAACEE, Vol-3, pp. 1-5, Oct 19-21. Available: <https://core.ac.uk/download/pdf/286381187.pdf>.
- [9] Thierry DUTOIT., "High-quality text-to-speech synthesis" ,pp. 1-21, 1997.

Appendices

Appendix A

Code Attachments

The following is the partial / subset of the Arduino.

A.1 Code for Arduino project

```
const int trigPin = 9; // Trigger pin of the ultrasonic sensor
const int echoPin = 10; // Echo pin of the ultrasonic sensor
const int buzzerPin = 11; // Buzzer pin

void setup() {
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(buzzerPin, OUTPUT);
}

void loop() {
  long duration, distance;

  // Send ultrasonic pulse
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Measure the time it takes for the pulse to return
  duration = pulseIn(echoPin, HIGH);

  // Calculate distance in centimeters
  distance = duration * 0.034 / 2;

  // Print distance to Serial Monitor
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  // Control the buzzer based on the distance
  if (distance <= 10) {
    // Object is very close, sound the buzzer loudly
    tone(buzzerPin, 1000); // Adjust frequency for your buzzer
  } else if (distance > 10 && distance <= 50) {
    // Object is approaching, sound the buzzer moderately
    tone(buzzerPin, 800); // Adjust frequency for your buzzer
  } else if (distance > 50 && distance <= 100) {
    // Object is approaching, sound the buzzer moderately
```

```

    tone(buzzerPin, 700 );
}
else if (distance > 100 && distance <= 200) {
    // Object is at a moderate distance, sound the buzzer softly
    tone(buzzerPin, 600); // Adjust frequency for your buzzer
}
else if (distance > 200 && distance <= 500) {
    // Object is approaching, sound the buzzer moderately
    tone(buzzerPin, 500 );
}
else if (distance > 500 && distance <= 2000) {
    // Object is approaching, sound the buzzer moderately
    tone(buzzerPin, 300 );
}
else {
    // Object is far away, turn off the buzzer
    noTone(buzzerPin);
}

delay(100); // Adjust the delay as needed
}

```

A.2 Sample Code for Website

A.2.1. Code for First Page of Website

A.2.1.1. HTML

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  </head>
  <body>
    <div class="blindupper">Blindkart</div>
    <div class="product-box1">
      <div class="product" id="product1">
        
        <br />
        <span>stick</span>
        <strong class="price">500</strong>
        <br />
        <button class="add-to-cart-btn" value="1">Add to Cart</button>
        <button class="remove-from-cart-btn" value="1">Remove from
Cart</button>
      </div>
      <div class="product" id="product2">
        

```

```

        <span>myproject</span>
        <strong class="price">500</strong>
        <br />
        <button class="add-to-cart-btn" value="2">Add to Cart</button>
        <button class="remove-from-cart-btn" value="2">Remove from
Cart</button>
    </div>
</div>

<div class="product-box2">
    <div class="product" id="product3">
        
        <span>braille</span>
        <strong class="price">300</strong>
        <br />
        <button class="add-to-cart-btn" value="3">Add to Cart</button>
        <button class="remove-from-cart-btn" value="3">Remove from
Cart</button>
    </div>
    <div class="product" id="product4">
        
        <span>keyboard</span>
        <strong class="price">400</strong>
        <br />
        <button class="add-to-cart-btn" value="4">Add to Cart</button>
        <button class="remove-from-cart-btn" value="4">Remove from
Cart</button>
    </div>
</div>
<button id="checkout-btn">Checkout</button>
<div class="shopping-cart">
    <h2>Shopping Cart</h2>
    <ul id="cart-items">
        <!-- Cart items will be added dynamically -->
    </ul>
</div>
<script src="store.js"></script>
</body>
</html>

```

A.2.1.2 CSS

```

* {
    padding: 0;
    margin: 0;
    box-sizing: border-box;
    /* Used to demonstrate vision of user */
    /* filter: blur(10px); */
}

body {
    font-family: Arial, sans-serif;
    background-color: #e1e1f2;
}

```

```

    color: #333;
}

.blindupper {
  height: 100px;
  font-family: Verdana, Geneva, Tahoma, sans-serif;
  font-size: 60px;
  width: 100%;
  color: #ffffff;
  background-color: rgb(23, 24, 27);
  position: fixed;
  top: 0;
  left: 0;
  padding: 10px;
  display: flex;
  justify-content: center;
  align-items: center;
  z-index: 1000;
}

.product-box1 {
  display: flex;
  justify-content: space-around;
  flex-wrap: wrap;
  margin: 20px;
  margin-top: 110px;
  border-radius: 2vh;
}

.product-box2 {
  display: flex;
  justify-content: space-around;
  flex-wrap: wrap;
  margin: 20px;
}

.product {
  border: 1px solid #ddd;
  padding: 10px;
  text-align: center;
  background-image: linear-gradient(
    rgb(82, 19, 178),
    rgb(103, 12, 173),
    rgb(16, 160, 185)
  );
  margin: 10px;
  max-width: 300px;
  box-shadow: 0 4px 8px rgba(120, 70, 70, 0.1);
  border-radius: 2vh;
  color: #fff;
}

.product img {
  max-width: 100%;
  height: 270px;
  box-shadow: #a21c1c;
  border-radius: 20px;
}

```

```

.product p {
  margin: 10px 0;
}

.product button {
  background-color: #00ff51;
  color: black;
  border: none;
  padding: 8px;
  cursor: pointer;
  height: 45px;
}

.product button:hover {
  background-color: #0056b3;
}

.shopping-cart {
  position: fixed;
  top: 60px;
  right: 10px;
  background-color: #fff;
  padding: 20px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
  z-index: 900;
}

.shopping-cart h2 {
  font-size: 20px;
  margin-bottom: 10px;
  text-align: center;
}

.shopping-cart {
  position: relative;
  justify-content: center;
}

#cart-items {
  list-style: none;
  padding: 0;
  margin: 0;
}

#cart-items li {
  margin-bottom: 10px;
}

#checkout-btn {
  background-color: #28a745;
  color: #fff;
  border: none;
  padding: 10px 20px;
  font-size: 50px;
  height: 100px;
  width: 100%;
  display: block;
  border-radius: 10%;
}

```

```

    margin: 0 auto;
}

#checkout-btn:hover {
    background-color: #218838;
    justify-content: center;
}

img {
    width: 400px;
    height: 300px;
}

```

A.2.1.3 JAVASCRIPT

```

var audioPlayed = false;
var totalcartprice = 0;
var audioo;
document.body.addEventListener("click", function () {
    if (!audioPlayed) {
        var audio = new Audio("sounds/blindkartintro.mp3");
        audio.play();
        audioPlayed = true;
    }
});

document.addEventListener("DOMContentLoaded", function () {
    const cartItems = document.getElementById("cart-items");
    const checkoutBtn = document.getElementById("checkout-btn");
    const cartimage = document.querySelectorAll("img");
    for (var ip = 0; ip < cartimage.length; ip++) {
        (function (index) {
            cartimage[index].addEventListener("mouseover", function () {
                this.style.transform = "scale(1.1)";
                makeSoundImg(cartimage[index].alt);
            });

            cartimage[index].addEventListener("mouseleave", function () {
                this.style.transform = "scale(1)";
                stopSoundImg();
            });
        })(ip);
    }
    function makeSoundImg(a) {
        switch (a) {
            case "Product 1":
                audioo = new Audio("sounds/stickdisc.mp3");
                audioo.play();
                break;
            case "Product 2":
                audioo = new Audio("sounds/myprojectdisc.mp3");
                audioo.play();
                break;
            case "Product 3":
                audioo = new Audio("sounds/brailledisc.mp3");

```

```

        audioo.play();
        break;
    case "Product 4":
        audioo = new Audio("sounds/keyboarddisc.mp3");
        audioo.play();
        break;
    default:
        console.log("no sound for", a);
        break;
    }
}

function stopSoundImg() {
    if (audioo) {
        audioo.pause();
        audioo.currentTime = 0;
    }
}

const savedCartItems = [];
function renderCartItems() {
    cartItems.innerHTML = "";
    savedCartItems.forEach((item) => {
        const li = document.createElement("li");
        li.textContent = item.description;
        cartItems.appendChild(li);
    });
}

// Function to handle adding items to the cart
function addToCart(productDescription) {
    const newItem = { description: productDescription };
    savedCartItems.push(newItem);

    // Save the updated cart to local storage
    localStorage.setItem("cartItems", JSON.stringify(savedCartItems));

    // Update the displayed cart items
    switch (productDescription) {
        case "keyboard":
            var aud = new Audio("sounds/keyadd.mp3");
            aud.play();
            break;
        case "stick":
            var aud = new Audio("sounds/stickadd.mp3");
            aud.play();
            break;

        case "myproject":
            var aud = new Audio("sounds/eyeadd.mp3");
            aud.play();
            break;
        case "braille":
            var aud = new Audio("sounds/brailleadd.mp3");
            aud.play();
            break;
    }
    renderCartItems();
}

```

```

}

// Function to handle removing items from the cart
function removeFromCart(productDescription) {
  // Find the index of the item in the cart
  const index = savedCartItems.findIndex(
    (item) => item.description === productDescription
  );

  // Remove the item if found
  if (index !== -1) {
    savedCartItems.splice(index, 1);

    // Save the updated cart to local storage
    localStorage.setItem("cartItems", JSON.stringify(savedCartItems));
    console.log(productDescription);
    switch (productDescription) {
      case "keyboard":
        var aud = new Audio("sounds/keyrem.mp3");
        aud.play();
        break;
      case "stick":
        var aud = new Audio("sounds/stickrem.mp3");
        aud.play();
        break;

      case "myproject":
        var aud = new Audio("sounds/eyerem.mp3");
        aud.play();
        break;
      case "braille":
        var aud = new Audio("sounds/braillerem.mp3");
        aud.play();
        break;
    }

    // Update the displayed cart items
    renderCartItems();
  }
}

// Event listener for Add to Cart buttons
const addToCartButtons = document.querySelectorAll(".add-to-cart-btn");
for (var ip = 0; ip < addToCartButtons.length; ip++) {
  (function (index) {
    addToCartButtons[index].addEventListener("mouseover", function () {
      audioo = new Audio("sounds/tapadd.mp3");
      audioo.play();
    });

    addToCartButtons[index].addEventListener("mouseleave", function () {
      audioo.pause();
      audioo.currentTime = 0;
    });
  })(ip);
}
addToCartButtons.forEach((button) => {

```

```

        button.addEventListener("click", function () {
            const productDescription =
                this.parentNode.querySelector("span").textContent;
            const price = this.parentNode.querySelector("strong").textContent;
            totalcartprice += parseFloat(price);
            addToCart(productDescription);
        });
    });

    // Event listener for Remove from Cart buttons
    const removeFromCartButtons = document.querySelectorAll(
        ".remove-from-cart-btn"
    );
    for (var ip = 0; ip < removeFromCartButtons.length; ip++) {
        (function (index) {
            removeFromCartButtons[index].addEventListener("mouseover", function
() {
                audioo = new Audio("sounds/tapremove.mp3");
                audioo.play();
            });

            removeFromCartButtons[index].addEventListener("mouseleave", function
() {
                audioo.pause();
                audioo.currentTime = 0;
            });
        })(ip);
    }

    removeFromCartButtons.forEach((button) => {
        button.addEventListener("click", function () {
            const productDescription =
                this.parentNode.querySelector("span").textContent;
            const price = this.parentNode.querySelector("strong").textContent;
            totalcartprice -= parseFloat(price);
            removeFromCart(productDescription);
        });
    });

    // Initial rendering of cart items
    renderCartItems();
    checkoutBtn.addEventListener("mouseover", function () {
        audioo = new Audio("sounds/checkouttap.mp3");
        audioo.play();
    });

    checkoutBtn.addEventListener("mouseleave", function () {
        audioo.pause();
        audioo.currentTime = 0;
    });
    // Event listener for Checkout button
    checkoutBtn.addEventListener("click", function () {
        if (savedCartItems.length === 0) {
            var audio = new Audio("sounds/emptykart.mp3");
            audio.play();
            totalcartprice = 0;
        } else {

```

```

        // window.location.href = 'billing.html';
        window.location.href =
`billing.html?totalcartprice=${totalcartprice}`;
    }
    });
});

```

A.2.2. Sample code for BILLING page of website

A.2.2.1 HTML

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"
  />
    <title>Billing Page</title>
    <link rel="preconnect" href="https://fonts.googleapis.com" />
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
    <link
      href="https://fonts.googleapis.com/css2?family=Pacifico&display=swap"
      rel="stylesheet"
    />
    <link rel="stylesheet" href="style.css" />
    <link rel="stylesheet" href="billing.css" />
  </head>
  <body>
    <h1>Billing Page</h1>

    <div>
      <h2>Cart Items</h2>
      <ul id="cart-items">
        <!-- Cart items will be added dynamically -->
      </ul>
    </div>
    <div class="totalprice">
      <!-- totalprice will be added dynamically -->
    </div>
    <div>
      <h2>Shipping Information</h2>
      <form id="billing-form">
        <label for="address">Address:</label>
        <input type="text" id="address" name="address" required />

        <label for="phone">Phone Number:</label>
        <input
          type="tel"
          id="phone"
          name="phone"
          pattern="[0-9]{10}"
          required
        />
      </form>
    </div>
  </body>
</html>

```

```

<button type="submit">Proceed to Checkout</button>

<div class="keyboard">
  <div class="keyboard__row">
    <div class="key--double" data-key="49">
      <div>!</div>
      <div>1</div>
    </div>
    <div class="key--double" data-key="50">
      <div>@</div>
      <div>2</div>
    </div>
    <div class="key--double" data-key="51">
      <div>#</div>
      <div>3</div>
    </div>
    <div class="key--double" data-key="52">
      <div>$</div>
      <div>4</div>
    </div>
    <div class="key--double" data-key="53">
      <div>%</div>
      <div>5</div>
    </div>
    <div class="key--double" data-key="54">
      <div>^</div>
      <div>6</div>
    </div>
    <div class="key--double" data-key="55">
      <div>&</div>
      <div>7</div>
    </div>
    <div class="key--double" data-key="56">
      <div>*</div>
      <div>8</div>
    </div>
    <div class="key--double" data-key="57">
      <div>(</div>
      <div>9</div>
    </div>
    <div class="key--double" data-key="48">
      <div>)</div>
      <div>0</div>
    </div>

    <div class="key--double-right key--word key--w4" data-key="8">
      <div>delete</div>
    </div>
  </div>
  <br />
  <div class="keyboard__row">
    <div class="key--letter" data-char="Q">Q</div>
    <div class="key--letter" data-char="W">W</div>
    <div class="key--letter" data-char="E">E</div>
    <div class="key--letter" data-char="R">R</div>
    <div class="key--letter" data-char="T">T</div>
  </div>
</div>

```

```

        <div class="key--letter" data-char="Y">Y</div>
        <div class="key--letter" data-char="U">U</div>
        <div class="key--letter" data-char="I">I</div>
        <div class="key--letter" data-char="O">O</div>
        <div class="key--letter" data-char="P">P</div>
    </div>
    <br />
    <div class="keyboard__row">
        <div class="key--bottom-left key--word key--w5" data-key="20">
            <span>caps lock</span>
        </div>
        <div class="key--letter" data-char="A">A</div>
        <div class="key--letter" data-char="S">S</div>
        <div class="key--letter" data-char="D">D</div>
        <div class="key--letter" data-char="F">F</div>
        <div class="key--letter" data-char="G">G</div>
        <div class="key--letter" data-char="H">H</div>
        <div class="key--letter" data-char="J">J</div>
        <div class="key--letter" data-char="K">K</div>
        <div class="key--letter" data-char="L">L</div>

        <div class="key--bottom-right key--word key--w5" data-key="13">
            <span>return</span>
        </div>
    </div>
    <br />
    <div class="keyboard__row">
        <div class="key--bottom-left key--word key--w6" data-key="16">
            <span>shift</span>
        </div>
        <div class="key--letter" data-char="Z">Z</div>
        <div class="key--letter" data-char="X">X</div>
        <div class="key--letter" data-char="C">C</div>
        <div class="key--letter" data-char="V">V</div>
        <div class="key--letter" data-char="B">B</div>
        <div class="key--letter" data-char="N">N</div>
        <div class="key--letter" data-char="M">M</div>

        <div class="key--bottom-right key--word key--w6"
data-key="16-R">
            <span>shift</span>
        </div>
    </div>
    <br />
    <div class="keyboard__row keyboard__row--h3">
        <div class="key--bottom-left key--word">
            <span>fn</span>
        </div>
        <div class="key--bottom-left key--word key--w1" data-key="17">
            <span>control</span>
        </div>
        <div class="key--bottom-left key--word key--w1" data-key="18">
            <span>option</span>
        </div>
        <div class="key--bottom-right key--word key--w3" data-key="91">
            <span>command</span>
        </div>
    </div>

```

```

        <div
            class="key--double key--right key--space"
            data-key="32"
            data-char=" "
        >
            &nbsp;
        </div>
        <div class="key--bottom-left key--word key--w3"
data-key="93-R">
            <span>command</span>
        </div>
        <div class="key--bottom-left key--word key--w1"
data-key="18-R">
            <span>option</span>
        </div>
    </div>
</div>
</form>
</div>
<script src="billing.js"></script>

<script src="keyboard.js"></script>
</body>
</html>

```

A.2.2.2 CSS

- **For Keyboard:**

```

h1 {
    text-align: center;
    font-size: 1.8em;
    font-family: monospace;
    padding: 0.7em;
}

.keyboard {
    text-align: center;
    font-size: 14px;
    font-family: sans-serif;
}

.keyboard__row {
    display: inline-block;
    height: 3em;
    margin: 0.2em;
}

.keyboard__row--h3 {
    height: 3.3em;
}

.keyboard__row > * {

```

```

position: relative;
background: #333;
text-align: center;
color: #eee;
float: left;
border-radius: 0.3em;
margin: 0.2em;
padding: 0.2em;
width: 3.3em;
height: 100%;
box-sizing: border-box;
cursor: pointer;
/* -webkit-user-select: none; */
border: 1px solid #444;
box-shadow: 0 0.2em 0 0.05em #222;
border-bottom-color: #555;
}

.keyboard__row--h1 > div {
width: 3.5em;
}

.keyboard__row > div[data-pressed],
.keyboard__row > div:active {
background: #2a2a2a;
color: #aaa;
position: relative;
top: 0.2em;
box-shadow: 0 0 0 0.05em black;
}

.key--w3 {
width: 4.6em;
}

.key--w4 {
width: 6em;
}

.key--w5 {
width: 6.5em;
}

.key--w6 {
width: 8.3em;
}

.key--space {
width: 18em;
}

.key--double > * {
padding-top: 0.1em;
}

.key--letter {
line-height: 2.8em;
}

```

```

}

.key--bottom-left > * {
  position: absolute;
  text-align: left;
  bottom: 0.4em;
  left: 0.4em;
}

.key--bottom-right > * {
  position: absolute;
  text-align: right;
  bottom: 0.4em;
  right: 0.4em;
}

.key--word > * {
  font-size: 0.8em;
}

.key--arrow--tall > *,
.key--arrow {
  height: 1.8em;
  margin-top: 1.7em;
  line-height: 0.5em;
}

.key--arrow--tall > * {
  padding-top: 0.2em;
}

.keyboard > .keyboard__row {
  text-align: center;
}

```

● For Main page:

```

body {
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
  background-color: #f4f4f4;
}
h2 {
  color: #bf0c0c;
  font-size: 45px;
  display: flex;
  justify-content: center;
  margin-top: 2vh;
  margin-bottom: 2vh;
}
h1 {
  color: #74067d;
  font-size: 45px;
}

```

```

    font-family: "Pacifico", cursive;
    padding: 0px;
    margin-top: 1vh;
    margin-bottom: 1vh;
}
.container {
    max-width: 800px;
    margin: 0 auto;
    padding: 20px;
}
#cart-items {
    list-style: none;
    padding: 0;
    display: flex;
    flex-direction: column-reverse;
    align-content: center;
    align-items: stretch;
    justify-content: space-evenly;
    flex-wrap: wrap;
}
#cart-items li {
    margin-bottom: 10px;
    font-size: 41vh;
}
form {
    margin-top: 20px;
}
label {
    margin-bottom: 1.2vh;
    font-size: larger;
    font-family: cursive;
    display: flex;
    justify-content: space-around;
    align-content: flex-end;
}
input {
    width: 100%;
    padding: 10px;
    box-sizing: border-box;
}
button {
    background-color: #282ca7;
    color: #fff;
    border: none;
    padding: 10px 20px;
    font-size: 50px;
    margin-top: 5vh;
    height: 100px;
    width: 100%;
    display: block;
    border-radius: 9%;
}
button:hover {
    background-color: #2e5cb3;
    align-content: center;
}
.totalprice {

```

```

    font-size: xx-large;
    font-style: italic;
}

```

A.2.2.3. JAVASCRIPT

- **For keyboard:**

```

function getKey(e) {
    var location = e.location;
    var selector;
    if (location === KeyboardEvent.DOM_KEY_LOCATION_RIGHT) {
        selector = ['[data-key="' + e.keyCode + '-R"]'];
    } else {
        var code = e.keyCode || e.which;
        selector = [
            '[data-key="' + code + '"]',
            '[data-char="' + encodeURIComponent(String.fromCharCode(code)) +
            '"]',
        ].join(",");
    }
    return document.querySelector(selector);
}

function pressKey(char) {
    var key = document.querySelector('[data-char="' + char.toUpperCase() +
    '"]');
    if (!key) {
        return console.warn("No key for", char);
    }
    key.setAttribute("data-pressed", "on");
    setTimeout(function () {
        key.removeAttribute("data-pressed");
    }, 200);
}

var h1 = document.querySelector("h1");
var originalQueue = h1.innerHTML;
var queue = h1.innerHTML;

function next() {
    var c = queue[0];
    queue = queue.slice(1);
    h1.innerHTML = originalQueue.slice(0, originalQueue.length -
    queue.length);
    pressKey(c);
    if (queue.length) {
        setTimeout(next, Math.random() * 200 + 50);
    }
}

h1.innerHTML = "&nbsp;";
setTimeout(next, 500);
//Sound generate for onscreen clicking

```

```

var keys = document.querySelectorAll(".key--double, .key--letter,
.key--word");

for (var i = 0; i < keys.length; i++) {
  keys[i].addEventListener(
    "click",
    (function (index) {
      return function () {
        if (index >= 0 && index <= 10) {
          var buttonInnerHTML = keys[index].dataset.key - 48;
          console.log(buttonInnerHTML);
          makeSound(buttonInnerHTML.toString());
        } else {
          var buttonInnerHTML = keys[index].dataset.char;
          console.log(buttonInnerHTML);
          makeSound(buttonInnerHTML.toLowerCase());
        }
      };
    })(i)
  );
}

document.addEventListener("keypress", function (event) {
  if (event.key >= 0 && event.key <= 10) {
    makeSound(event.key);
  } else {
    makeSound(event.key.toLowerCase());
  }
});

function makeSound(key) {
  switch (key) {
    case "q":
      var tom1 = new Audio("sounds/8d82b5_Letter_Q_Sound_Effect.mp3");
      tom1.play();
      break;

    case "w":
      var tom2 = new Audio("sounds/8d82b5_Letter_W_Sound_Effect.mp3");
      tom2.play();
      break;

    case "e":
      var tom2 = new Audio("sounds/8d82b5_Letter_E_Sound_Effect.mp3");
      tom2.play();
      break;

    case "r":
      var tom2 = new Audio("sounds/8d82b5_Letter_R_Sound_Effect.mp3");
      tom2.play();
      break;

    case "t":
      var tom2 = new Audio("sounds/8d82b5_Letter_T_Sound_Effect.mp3");
      tom2.play();
      break;

    case "y":
      var tom2 = new Audio("sounds/8d82b5_Letter_Y_Sound_Effect.mp3");
      tom2.play();
      break;
  }
}

```

```

case "u":
    var tom2 = new Audio("sounds/8d82b5_Letter_U_Sound_Effect.mp3");
    tom2.play();
    break;
case "i":
    var tom2 = new Audio("sounds/8d82b5_Letter_I_Sound_Effect.mp3");
    tom2.play();
    break;
case "o":
    var tom2 = new Audio("sounds/8d82b5_Letter_O_Sound_Effect.mp3");
    tom2.play();
    break;
case "p":
    var tom2 = new Audio("sounds/8d82b5_Letter_P_Sound_Effect.mp3");
    tom2.play();
    break;
case "a":
    var tom2 = new Audio("sounds/8d82b5_Letter_A_Sound_Effect.mp3");
    tom2.play();
    break;
case "s":
    var tom2 = new Audio("sounds/8d82b5_Letter_S_Sound_Effect.mp3");
    tom2.play();
    break;
case "d":
    var tom2 = new Audio("sounds/8d82b5_Letter_D_Sound_Effect.mp3");
    tom2.play();
    break;
case "f":
    var tom2 = new Audio("sounds/8d82b5_Letter_F_Sound_Effect.mp3");
    tom2.play();
    break;
case "g":
    var tom2 = new Audio("sounds/8d82b5_Letter_G_Sound_Effect.mp3");
    tom2.play();
    break;
case "h":
    var tom2 = new Audio("sounds/8d82b5_Letter_H_Sound_Effect.mp3");
    tom2.play();
    break;
case "j":
    var tom2 = new Audio("sounds/8d82b5_Letter_J_Sound_Effect.mp3");
    tom2.play();
    break;
case "k":
    var tom2 = new Audio("sounds/8d82b5_Letter_K_Sound_Effect.mp3");
    tom2.play();
    break;
case "l":
    var tom2 = new Audio("sounds/8d82b5_Letter_L_Sound_Effect.mp3");
    tom2.play();
    break;
case "z":
    var tom2 = new Audio("sounds/8d82b5_Letter_Z_Sound_Effect.mp3");
    tom2.play();
    break;
case "x":

```

```

        var tom2 = new Audio("sounds/8d82b5_Letter_X_Sound_Effect.mp3");
        tom2.play();
        break;
    case "c":
        var tom2 = new Audio("sounds/8d82b5_Letter_C_Sound_Effect.mp3");
        tom2.play();
        break;
    case "v":
        var tom2 = new Audio("sounds/8d82b5_Letter_V_Sound_Effect.mp3");
        tom2.play();
        break;
    case "b":
        var tom2 = new Audio("sounds/8d82b5_Letter_B_Sound_Effect.mp3");
        tom2.play();
        break;
    case "n":
        var tom2 = new Audio("sounds/8d82b5_Letter_N_Sound_Effect.mp3");
        tom2.play();
        break;
    case "m":
        var tom2 = new Audio("sounds/8d82b5_Letter_M_Sound_Effect.mp3");
        tom2.play();
        break;
    case "1":
        var tom2 = new Audio("sounds/8d82b5_The_Number_1_Sound_Effect.mp3");
        tom2.play();
        break;
    case "2":
        var tom2 = new Audio("sounds/8d82b5_The_Number_2_Sound_Effect.mp3");
        tom2.play();
        break;
    case "3":
        var tom2 = new Audio("sounds/8d82b5_The_Number_3_Sound_Effect.mp3");
        tom2.play();
        break;
    case "4":
        var tom2 = new Audio("sounds/8d82b5_The_Number_4_Sound_Effect.mp3");
        tom2.play();
        break;
    case "5":
        var tom2 = new Audio("sounds/8d82b5_The_Number_5_Sound_Effect.mp3");
        tom2.play();
        break;
    case "6":
        var tom2 = new Audio("sounds/8d82b5_The_Number_6_Sound_Effect.mp3");
        tom2.play();
        break;
    case "7":
        var tom2 = new Audio("sounds/8d82b5_The_Number_7_Sound_Effect.mp3");
        tom2.play();
        break;
    case "8":
        var tom2 = new Audio("sounds/8d82b5_The_Number_8_Sound_Effect.mp3");
        tom2.play();
        break;
    case "9":
        var tom2 = new Audio("sounds/8d82b5_The_Number_9_Sound_Effect.mp3");

```

```

        tom2.play();
        break;
    case "0":
        var tom2 = new Audio("sounds/8d82b5_The_Number_0_Sound_Effect.mp3");
        tom2.play();
        break;
    case "Enter":
        var tom2 = new Audio("sounds/enter.mp3");
        tom2.play();
    default:
        console.log("No sound for", key);
    }
}

document.addEventListener("keydown", function (event) {
    if (event.code === "Backspace") {
        var backspaceSound = new Audio("sounds/backspace.mp3");
        backspaceSound.play();
    }
});

document.body.addEventListener("keydown", function (e) {
    var key = getKey(e);
    if (!key) {
        return console.warn("No key for", e.keyCode);
    }
    key.setAttribute("data-pressed", "on");
});

document.body.addEventListener("keyup", function (e) {
    var key = getKey(e);
    key && key.removeAttribute("data-pressed");
});

function size() {
    var size = keyboard.parentNode.clientWidth / 90;
    keyboard.style.fontSize = size + "px";
    console.log(size);
}

var keyboard = document.querySelector(".keyboard");
window.addEventListener("resize", function (e) {
    size();
});
size();

```

● For Main page:

```

var audioPlayed = false;
var audio;
document.body.addEventListener("click", function () {
    if (!audioPlayed) {
        var audio = new Audio("sounds/billingpagefront.mp3");
        audio.play();
        audioPlayed = true;
    }
}

```

```

});

document.addEventListener("DOMContentLoaded", function () {
    const cartItemsElement = document.getElementById("cart-items");
    const phonetag = document.querySelectorAll("input")[1];
    const addresstag = document.querySelectorAll("input")[0];

    phonetag.addEventListener("click", function () {
        var audio = new Audio("sounds/enterphonenummer.mp3");
        audio.play();
    });
    addresstag.addEventListener("click", function () {
        var audio = new Audio("sounds/enteraddress.mp3");
        audio.play();
    });
    const billingForm = document.getElementById("billing-form");
    const totalpriceElement = document.querySelector(".totalprice");
    var cartproduct = "";
    const urlParams = new URLSearchParams(window.location.search);
    const totalcartprice = urlParams.get("totalcartprice");

    // Display totalcartprice on the billing page
    if (totalcartprice !== null) {
        totalpriceElement.textContent = `Total Cart Price: ₹${totalcartprice}`;
    }
    // Retrieve cart items from local storage
    const savedCartItems = JSON.parse(localStorage.getItem("cartItems")) ||
[];

    // Function to render cart items on the billing page
    function renderCartItems() {
        savedCartItems.forEach((item) => {
            cartproduct = cartproduct + item.description + " ";
            const li = document.createElement("li");
            li.textContent = item.description;
            cartItemsElement.appendChild(li);
        });
    }
    const submitbutton = document.querySelectorAll("button")[0];
    submitbutton.addEventListener("mouseover", function () {
        audioo = new Audio("sounds/checkouttap.mp3");
        audioo.play();
    });

    submitbutton.addEventListener("mouseleave", function () {
        audioo.pause();
        audioo.currentTime = 0;
    });
    // Event listener for form submission
    billingForm.addEventListener("submit", function (event) {
        event.preventDefault();

        // Retrieve shipping information
        const address = document.getElementById("address").value;
        const phone = document.getElementById("phone").value;
        const requestData = {
            totalcartprice: totalcartprice,

```

```

        cartItems: cartproduct,
        address: address,
        phone: phone,
    };
    fetch("http://localhost:3000/checkout", {
        method: "POST",
        headers: {
            "Content-Type": "application/json",
        },
        body: JSON.stringify(requestData),
    })
    .then(function (response) {
        if (response.ok) {
            console.log("Server is running!");
        } else {
            window.location.href = "sorry.html";
        }
    })
    .catch((error) => console.error("Error:", error));
    window.location.href = "thankyou.html";
});
renderCartItems();
});

```

A.2.3. Code for THANK YOU page of website

- A.2.3.1 HTML

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Document</title>
    <style>
      * {
        margin: 0;
        padding: 0;
        box-sizing: border-box;
        background-image: url("/images/thankyou.png");
        background-repeat: no-repeat;
        background-size: cover;
      }
    </style>
  </head>
  <body></body>
</html>

```

A.2.2 Code for backend of website

```

import express from "express";
import bodyParser from "body-parser";
import mysql from "mysql2";
const app = express();
const port = 3000;
const db = mysql.createConnection({
  host: "localhost",
  user: "root",
  password: "root",
  database: "server",
});

db.connect((err) => {
  if (err) {
    console.error("Error connecting to MySQL:", err);
  } else {
    console.log("Connected to MySQL");
  }
});

app.use(bodyParser.urlencoded({ extended: true }));
app.use(bodyParser.json());
app.use(express.static("C:\\Users\\ASUS\\Downloads\\pro"));
app.post("/checkout", (req, res) => {
  const totalcartprice = req.body.totalcartprice;
  const cartItems = req.body.cartItems;
  const address = req.body.address;
  const phone = req.body.phone;
  const sql =
    "INSERT INTO orders (total_price, cart_items, address, phone) VALUES"
    "(?, ?, ?, ?)";

  db.query(sql, [totalcartprice, cartItems, address, phone], (err, result)
=> {
    if (err) {
      console.error("Error inserting into database:", err);
    } else {
      console.log("Order inserted successfully");
    }
  });
});

app.get("/", (req, res) => {
  res.sendFile("C:\\Users\\ASUS\\Downloads\\pro\\store.html");
});

app.listen(port, () => {
  console.log(`Server is running on port ${port}`);
});

```

A.2.3. Database creation code and display code for MySQL

```

CREATE DATABASE SEVER;
USE SEVER;
CREATE TABLE ORDERS (total_price DECIMAL(10,2), cart_items TEXT, address
TEXT, phone TEXT);

```

```
SELECT * FROM ORDERS;
```