

BYTEBAY

Website Documentation

Abstract

Bytebay is an online business website that retails laptops from industry-leading brand from Acer, Dell, Lenovo, and MSI. This website is created using HTML, CSS, and JavaScript.

CARL MATTHEW ONGKIT
DCPET 1-2

Table of Contents

1 Introduction to the Portfolio	1
1.1 Introduction.....	1
1.2 Business Background.....	1
2 Website Screens	2
2.1 Home	2
2.2 About Us	3
2.3 Shop.....	4
2.4 Contact Us	5
2.5 Partners	6
3 Functionalities	8
3.1 Search Function	8
3.2 Shopping Cart	10
3.3 Database.....	13
3.4 Update Product Details	14
3.5 Translator	15

Introduction to the Portfolio

Introduction

ByteBay is an online store where you can buy laptops. When you visit ByteBay, you'll see a website with different pages that help you find and buy the laptop you want. This website was made using Hypertext Markup Language (HTML), JavaScript (JS), and Cascading Style Sheets (CSS), this helped me deliver a dynamic and great immersive user experience. With the already existing template “Zay Shop Template” from [templatemo.com](https://www.templatemo.com) that utilizes Bootstrap 5 beta 1 I was able to ensure the responsiveness and compatibility of my website across various devices and web browsers.


The hosting infrastructure of my website is supported by GitHub, leveraging the GitHub Pages feature for deployment. This is free and accessible to everyone and is good option for a student like me whose studying HTML and other coding languages. I was able to successfully create and add more features to websites thanks to the widely available resources online like GitHub, YouTube Videos, Reddit, and Stack Overflow. This resources that I mention greatly benefited me in debugging and fixing bugs that I encountered in creating this website.

Business Background

ByteBay offers an online platform dedicated to laptop sales, providing a convenient avenue for customers to browse and purchase their desired devices. It's all about making it easy for people to find and get the laptops they need. I chose the name “ByteBay” because I think the name itself is catchy and shows that the website is all about technology. I picked this part of the industry because laptops are everywhere, they are used for work, school, and even play. Also, this field is very related to my course, so it feels like a perfect fit whenever I wanted to start a real business in the future.

Website Screens

Home




ProArt Studiobook Pro 16
OLED
(W7604)

The new ProArt Studiobook Pro 16 OLED is a powerful mobile workstation that exceeds all expectations and goes far beyond the extraordinary. Its unstoppable 13th Gen Intel® Core™ i9-13980HX processor, pro-grade NVIDIA RTX™ 3000 Ada Generation Laptop GPU, vast amounts of memory and ultrafast storage will supercharge your content creation workflow and let your imagination do what it does best.

Featured Products

Discover excellence with our Laptops, meticulously curated to showcase the latest and most innovative offerings in the world of technology.




★★★★★ ₱79,990.00

XPS 15 Laptop

Bring your ideas to life with the XPS 15. The 16:10 4-sided InfinityEdge OLED display delivers brilliance that makes your creation as vivid as your imagination.

Reviews (24)




★★★★★ ₱99,999.00

XPS 13 Plus Laptop

Designed from the ground up to be our most powerful 13-inch XPS yet, featuring the latest Intel® 12th gen processors at 28W.

Reviews (48)



★★★★★ ₱178,880.00


ALIENWARE X16

Elevate your style and get used to stares with the new, head-turning Legend 3 design, AlienFX lighting and full metal parts that ensures - win or lose - you'll draw attention.

Reviews (74)


Brands of The Month

Products from these brands showcase cutting-edge products, setting the standard for innovation and quality in the tech industry.




Asus

[Go Shop](#)



Acer

[Go Shop](#)

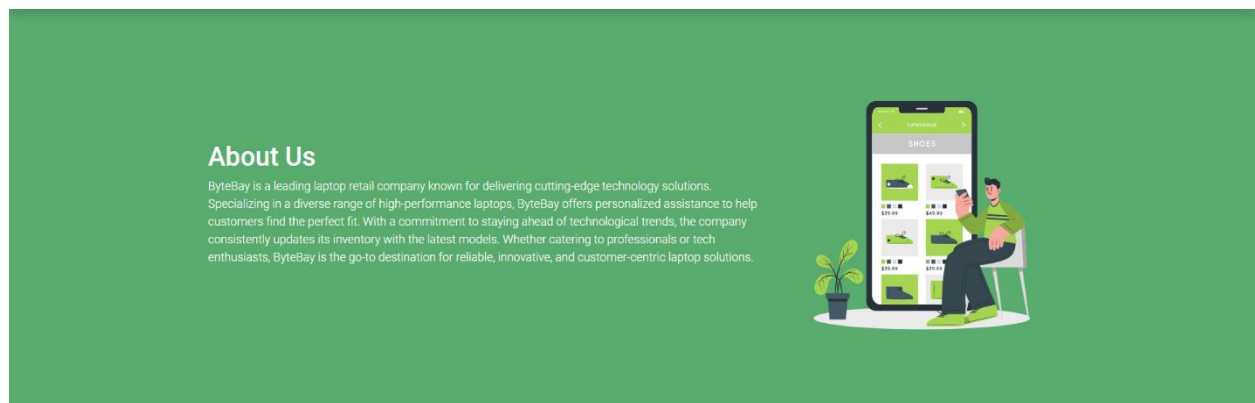


MSI

[Go Shop](#)

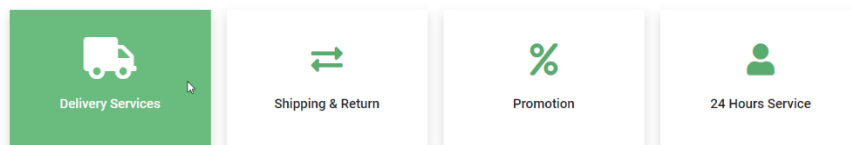
When you land on ByteBay's homepage, you're greeted by a captivating display of products, each sliding into view one after another. These products showcase the latest and most sought-after items in the market. As you scroll down, you'll encounter the 'Brands of the Month' section, featuring three prominent brands currently in the spotlight. Further down the page, the 'Featured Products' section highlights the top three products with the highest reviews on the website. This layout ensures that visitors can easily spot the trendiest and most recommended products available on ByteBay. The homepage is designed to engage users and streamline the process of discovering the best offerings from the website.

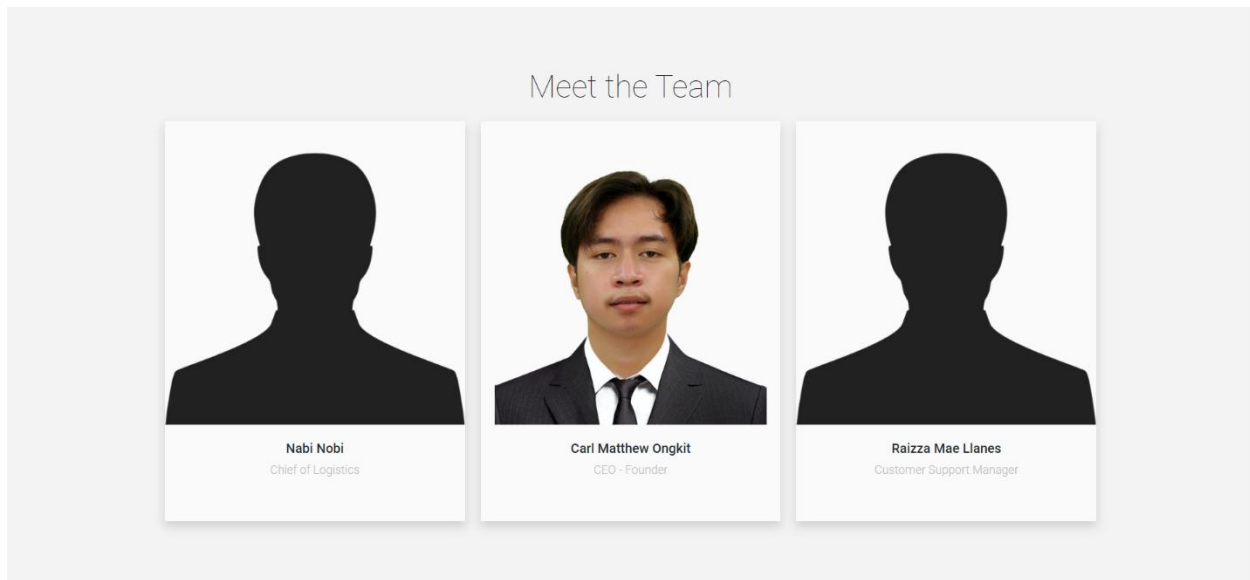
About Us



Our Services

Our services encompass expert guidance in selecting the ideal laptop, ensuring cutting-edge technology tailored to meet your specific needs.





At ByteBay's "About Us" page, you'll find a brief story about our company and what we stand for. As you scroll down, you'll see the different services we offer, and if you hover over each service, it'll turn green to catch your eye. Towards the end, you'll meet our team: Nabi Nobi, our Logistics Chief, me, the CEO and Founder, and Raizza Mae Llanes, our Customer Support Manager.

Shop

Categories







All

Featured

Brands

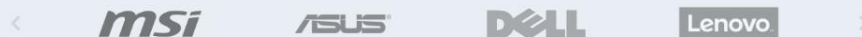
Acer
Asus
Dell
Lenovo
MSI



 <p>Swift 3 SF314-512-50XM OPI Acer</p> <p>★★★★★ ₱43,699.00</p>	 <p>A515-56G-579G P.SILVER core i5 Acer</p> <p>★★★★★ ₱39,899.00</p>	 <p>Aspire Vero AV15-51R-5822 Acer</p> <p>★★★★★ ₱38,949.00</p>
 <p>Swift 3 SF314-43-R06N Acer</p> <p>★★★★★</p>	 <p>Aspire Vero AV15-51-78G8 Acer</p> <p>★★★★★</p>	 <p>Zenbook Pro Duo 15 OLED Asus</p> <p>★★★★★</p>

Partnerships

ByteBay collaborates with industry leaders to enhance our commitment to delivering cutting-edge technology solutions, fostering innovation and ensuring a diverse range of high-performance laptops for our customers.

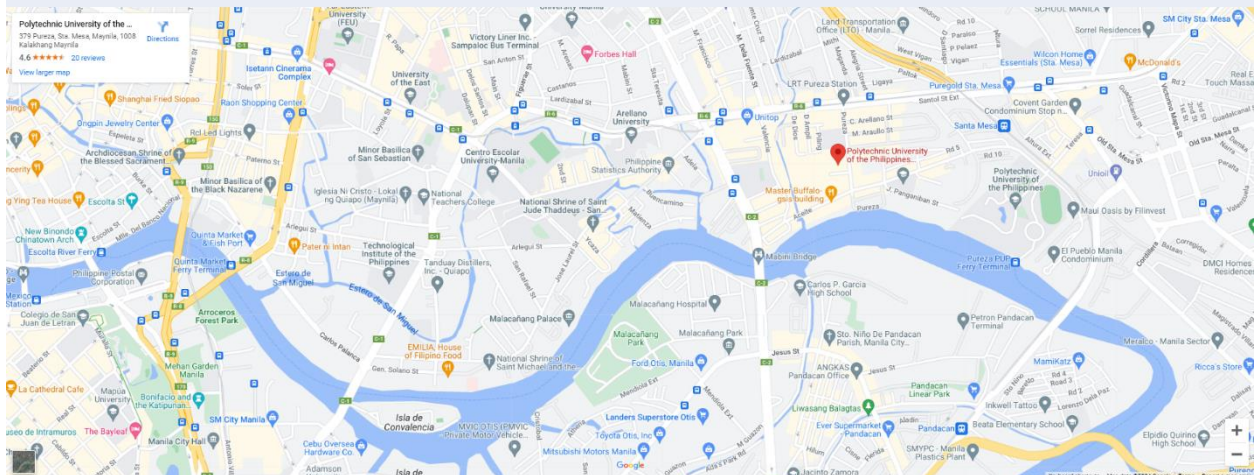


On ByteBay's "Shop" page, you'll find an extensive collection of products available for purchase, complete with star ratings, images, brand names, and prices, making it easy to find the perfect laptop. Navigate through different categories to filter products by brand, ensuring you discover exactly what you're looking for. As you explore the last section of the website, you'll encounter the "Partnerships" section, showcasing the brands we've partnered with to ensure the buyers that they are buying from reputable brands.

Contact Us

Contact Us

Contact us at ByteBay for personalized assistance and expert guidance in selecting the perfect laptop, as we are dedicated to providing top-notch technology solutions tailored to your specific needs.



Operating Hours: 10:00 AM - 9:00 PM

Name

Email

Subject

Message

[Let's Talk](#)


ByteBay's "Contact Us" section provides a convenient map displaying our physical store's precise location and address, courtesy of an embedded Google Map. Scroll down to view our operating hours and utilize HTML forms to input your details and address any inquiries or concerns you may have.

Partners

Partnerships


Partnering with industry-leading brands such as Asus, Acer, Dell, Lenovo, and MSI, our laptop retail company fosters collaborative relationships to provide customers with an extensive range of cutting-edge technology solutions.

As a premier supplier, Asus is renowned for its innovative approach to technology, offering a diverse range of high-performance laptops that cater to various needs. With a focus on quality and reliability, Asus continues to push boundaries, delivering cutting-edge products that redefine user experiences.



ASUS

Recognized for its commitment to accessibility and affordability, Acer stands out as a reliable supplier of laptops, providing customers with versatile solutions tailored to both professional and personal use. With a global presence and a reputation for innovation, Acer remains at the forefront of the industry, consistently delivering products that combine performance and value.



ACER



Upon entering ByteBay's "Partners" page, visitors are greeted with a lineup of logo pictures representing the reputable brands we've partnered with. These brands, including Lenovo, Asus, Acer, MSI, and Dell, are esteemed suppliers renowned worldwide for their quality products.

Functionalities

Search Function

For my Search Function, I use JavaScript and a separate HTML file for the results of my search term. In this JavaScript below, I use `.addEventListener` at the 1st line of my script. The purpose of this script is to wait for the content of the pages to load before scripting the script itself so my script can properly get the attributes and elements from the contents of my html website. So how this script works is when a submit button is clicked it changes the link or URL of my website to add 'search?q= + the input from the search bar' and then after that it uses `modal.hide()` where it closes the search bar after triggering it.

```

1  document.addEventListener('DOMContentLoaded', function () {
2      const form = document.getElementById('searchForm');
3      const input = document.getElementById('inputModalSearch');
4      const modal = new bootstrap.Modal(document.getElementById('templatemo_search')); // Initialize the modal
5
6      form.addEventListener('submit', function (event) {
7          event.preventDefault();
8          const searchTerm = input.value.toLowerCase();
9
10         // Hide the modal after handling the search (you can remove this line if you want to keep the modal open)
11         window.location.href = `search?q=${encodeURIComponent(searchTerm)}`;
12         modal.hide();
13     });
14 }, {});

```

After that, the website will now go to another HTML website where it processes the result of my JavaScript. I added a script under my html website where whenever a search term is present it will update the content of searched products on the website. This part of my script uses `fetch` function where it gets the information for the information to load when a search term and information in database matched it will show the product details.

```

document.addEventListener('DOMContentLoaded', function () {
    const productsContainer = document.getElementById('productsContainer');
    const modal = new bootstrap.Modal(document.getElementById('templatemo_search'));
    const container = document.getElementById('productsContainer');

    const searchParams = new URLSearchParams(window.location.search);
    const searchTerm = searchParams.get('q');

    if (searchTerm) {
        fetch('assets/js/database.json')
            .then(response => response.json())
            .then(data => {
                const searchResults = data.filter(product =>
                    product.name.toLowerCase().includes(searchTerm) ||
                    product.brand.toLowerCase().includes(searchTerm) ||
                    product.description.toLowerCase().includes(searchTerm)
                );

                container.innerHTML = '';

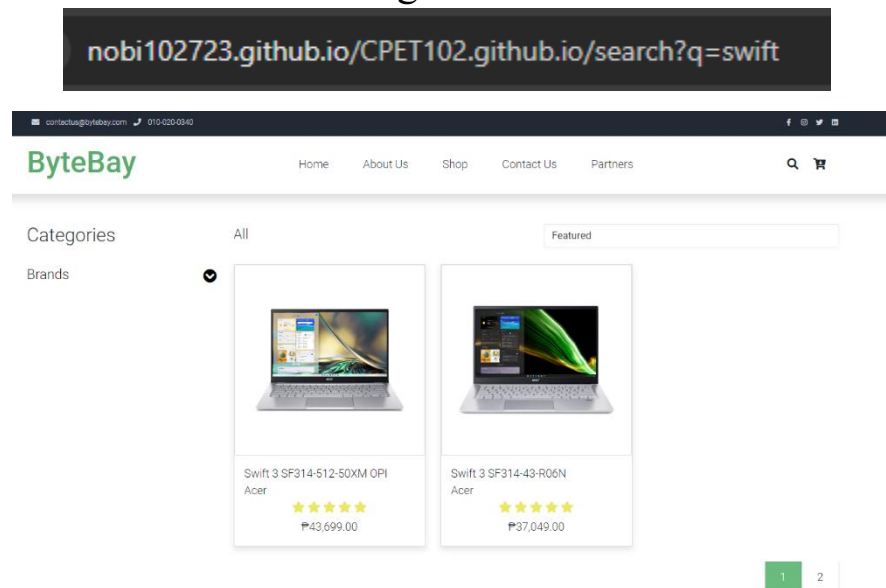
                searchResults.forEach(product => {
                    updateProductDetails(product, container);
                });
            })
            .catch(error => console.error('Error loading data:', error));
    }
});

```

Now I will show you another function that I created where the purpose of this function is that it updates the product details on the website. This function is then called whenever a search function is used.

```
function updateProductDetails(product, container) {
  if (product && container) {
    const newHref = 'shop-single?q=' + encodeURIComponent(product.name);
    const productHTML = `
      <div class="col-md-4">
        <div class="card mb-4 product-wap rounded-0">
          <div class="card rounded-0">
            
            <div class="card-img-overlay rounded-0 product-overlay d-flex align-items-center justify-content-center">
              <ul class="list-unstyled">
                <li><a class="btn btn-success text-white" href="${newHref}"><i class="far fa-heart"></i></a></li>
                <li><a class="btn btn-success text-white mt-2" href="${newHref}"><i class="far fa-eye"></i></a></li>
                <li><a class="btn btn-success text-white mt-2" href="${newHref}"><i class="fas fa-cart-plus"></i></a></li>
              </ul>
            </div>
          </div>
        </div>
        <div class="card-body">
          <a href="${newHref}" class="h3 text-decoration-none">${product.name}</a>
          <ul class="w-100 list-unstyled d-flex justify-content-between mb-0">
            <li>${product.brand}</li>
            <li class="pt-2">
              <span class="product-color-dot color-dot-red float-left rounded-circle ml-1"></span>
              <span class="product-color-dot color-dot-blue float-left rounded-circle ml-1"></span>
              <span class="product-color-dot color-dot-black float-left rounded-circle ml-1"></span>
              <span class="product-color-dot color-dot-light float-left rounded-circle ml-1"></span>
              <span class="product-color-dot color-dot-green float-left rounded-circle ml-1"></span>
            </li>
          </ul>
          <ul class="list-unstyled d-flex justify-content-center mb-1">
            <li>
              <i class="text-warning fa fa-star"></i>
              <i class="text-warning fa fa-star"></i>
              <i class="text-warning fa fa-star"></i>
              <i class="text-warning fa fa-star"></i>
              <i class="text-warning fa fa-star"></i>
            </li>
          </ul>
          <p class="text-center mb-0">${product.price}</p>
        </div>
      </div>
    `;
  }
}
```

Example of Search Function being used:



Shopping Cart

For Shopping Cart of my website, I also use HTML and JavaScript to display the products and calculate the total price of the products added in the Shopping Cart. I also used Cookies to temporarily store Product IDs for my website so that the JavaScript and HTML can show the products and calculate it for the customer. Starting with Cookies, this is how I store the Product IDs using cookies through JavaScript. I have multiple functions retrieving and setting data in my cookie. The way it works is it checks if the Product ID is in the cookie already if it is it add an increment to that product therefore increase the quantity of that product. If not, it adds the Product ID inside the cookie and add a default quantity of 1.

```
document.querySelectorAll('.add-to-cart').forEach(button => {
  button.addEventListener('click', function() {
    const productId = this.dataset.productId;
    addToCart(productId);
  });
});

function addToCart(productId) {
  let cartItems = JSON.parse(getCookie('cart')) || {};

  if (cartItems[productId]) {
    cartItems[productId]++;
  } else {
    cartItems[productId] = 1;
  }

  setCookie('cart', JSON.stringify(cartItems));

  console.log('Product added to cart:', productId);
  console.log('Cart items:', cartItems);
}

function setCookie(name, value, days = 7) {
  const expires = new Date(Date.now() + days * 86400000).toUTCString();
  document.cookie = `${name}=${encodeURIComponent(value)}; expires=${expires}; path=/`;
}

function getCookie(name) {
  const cookieString = document.cookie;
  const cookies = cookieString.split('; ');
  for (let cookie of cookies) {
    const [cookieName, cookieValue] = cookie.split('=');
    if (cookieName === name) {
      return decodeURIComponent(cookieValue);
    }
  }
  return null;
}
```

After that I also have another JavaScript where it processes the data from the Cookie for it to properly display in my HTML. It gets the cookie from the user and searched through the contents of that cookie and then display it using another function.

```
document.querySelectorAll('.add-to-cart').forEach(button => {
  button.addEventListener('click', function() {
    const productId = this.dataset.productId;
    addToCart(productId);
  });
});

function addToCart(productId) {
  let cartItems = JSON.parse(getCookie('cart')) || {};

  if (cartItems[productId]) {
    cartItems[productId]++;
  } else {
    cartItems[productId] = 1;
  }

  setCookie('cart', JSON.stringify(cartItems));

  console.log('Product added to cart:', productId);
  console.log('Cart items:', cartItems);
}

function setCookie(name, value, days = 7) {
  const expires = new Date(Date.now() + days * 86400000).toUTCString();
  document.cookie = `${name}=${encodeURIComponent(value)}; expires=${expires}; path=/`;
}

function getCookie(name) {
  const cookieString = document.cookie;
  const cookies = cookieString.split('; ');
  for (let cookie of cookies) {
    const [cookieName, cookieValue] = cookie.split('=');
    if (cookieName === name) {
      return decodeURIComponent(cookieValue);
    }
  }
  return null;
}
```

Also, this is the last part of my script for Shopping Cart. The way this script works is it calculates and display the Total, Subtotal, and Price of the Products in real time. I achieved this by using again `.addEventListener` so whenever a quantity is changed from the dropdown it also adjusts automatically.

```
function calculateTotal() {
  const prices = document.querySelectorAll('.tdprice');
  let subtotal = 0;

  prices.forEach((price) => {
    const priceText = price.textContent.trim().replace('P', '').replace(/,/g, '');
    const priceValue = parseFloat(priceText);
    subtotal += priceValue;
  });

  const tax = subtotal * 0.12;
  const total = subtotal + tax;

  const formattedSubtotal = subtotal.toLocaleString();
  const formattedTax = tax.toLocaleString();
  const formattedTotal = total.toLocaleString();

  document.getElementById('scsubtotal').innerHTML = 'P' + formattedSubtotal;
  document.getElementById('sctax').innerHTML = 'P' + formattedTax;
  document.getElementById('sctotal').innerHTML = 'P' + formattedTotal;
}

calculateTotal();

document.addEventListener('click', function(event) {
  if (event.target.classList.contains('remove-button')) {
    const trElement = event.target.closest('tr');

    trElement.remove();

    calculateTotal();
  }
});
```

Database

The Database is the most crucial and important part of my website. It plays a vital role in how my website function and works. Because all the information of every product on the website is stored in this JavaScript Object Notation (JSON) file. Almost every script that I made in this website fetches data like Product Name or Prices from this file for it to properly process data and provide user experience for the customer.

Image of the structure of the data from my JSON file:

```
{
  "id": 1,
  "name": "Swift 3 SF314-512-50XM OPI",
  "price": "P43,699.00",
  "brand": "Acer",
  "description": "The Swift 3 is part of Acer's Swift line of laptops, which are designed to be thinner and lighter than typical laptops. The Swift 3 SF314-512-50XM OPI is 15.90 mm high, 321 mm wide, 210 mm deep.",
  "processor": "Intel® Core™ i5-1240P processor",
  "osystem": "Windows 11 Home",
  "vcards": "Intel® Iris® Xe Graphics",
  "display": "35.6 cm (14') display with IPS (In-Plane Switching) technology",
  "memory": "8 GB",
  "storage": "512GB NVMe SSD",
  "dir": "assets/products/Acer",
  "idp": "1"}],

{
  "id": 2,
  "name": "AS15-56G-579G P.SILVER core i5",
  "price": "P39,899.00",
  "brand": "Acer",
  "description": "The Acer Aspire S AS15-56G-579G P.SILVER core i5 is a pure silver notebook that measures 36.34 cm x 23.85 cm x 1.795 cm and weighs 1.9 kg.",
  "processor": "Intel® Core i5- 1135G7",
  "osystem": "Windows 11 Home",
  "vcards": "NVIDIA® GeForce® MX350",
  "display": "15.6' Acer ComfyView™ LED-backlit TFT LCD",
  "memory": "8 GB",
  "storage": "512GB NVMe M.2",
  "dir": "assets/products/Acer",
  "idp": "2"}],

{
  "id": 3,
  "name": "Aspire Vero AV15-51R-5822",
  "price": "P38,949.00",
  "brand": "Acer",
  "description": "Acer Aspire Vero is an affordable, eco-friendly laptop that has enough battery life to last most of the day. However, PCWorld says that the laptop's 3D performance is only good enough for casual gaming.",
  "processor": "Intel Core i5-1155G7",
  "osystem": "Windows 11 Home",
  "vcards": "Intel® Iris® Xe Graphics",
  "display": "15.6' FHD (1920 x 1080)",
  "memory": "8GB LPDDR4",
  "storage": "512GB NVMe M.2",
  "dir": "assets/products/Acer",
  "idp": "3"}]
```

Update Product Details

Like how I emphasized Database above, this function also plays a very important part in my website. This function is the one responsible for displaying and showing information to the HTML Website. It's a function widely used across my JavaScript because it works by fetching data from the database and using that data to insert a part of HTML code to my website.

```
function updateProductDetails(product, container) {
  if (product && container) {
    const newHref = 'shop-single?q=' + encodeURIComponent(product.name);
    const productHTML = `
      <div class="col-md-4">
        <div class="card mb-4 product-wap rounded-0">
          <div class="card rounded-0">
            
            <div class="card-img-overlay rounded-0 product-overlay d-flex align-items-center justify-content-center">
              <ul class="list-unstyled">
                <li><a class="btn btn-success text-white" href="${newHref}"><i class="far fa-heart"></i></a></li>
                <li><a class="btn btn-success text-white mt-2" href="${newHref}"><i class="far fa-eye"></i></a></li>
                <li><a class="btn btn-success text-white mt-2" href="${newHref}"><i class="fas fa-cart-plus"></i></a></li>
              </ul>
            </div>
          </div>
          <div class="card-body">
            <a href="${newHref}" class="h3 text-decoration-none" id="${product.id}">${product.name}</a>
            <ul class="w-100 list-unstyled d-flex justify-content-between mb-0">
              <li>${product.brand}</li>
              <li class="pt-2">
                <span class="product-color-dot color-dot-red float-left rounded-circle ml-1"></span>
                <span class="product-color-dot color-dot-blue float-left rounded-circle ml-1"></span>
                <span class="product-color-dot color-dot-black float-left rounded-circle ml-1"></span>
                <span class="product-color-dot color-dot-light float-left rounded-circle ml-1"></span>
                <span class="product-color-dot color-dot-green float-left rounded-circle ml-1"></span>
              </li>
            </ul>
            <ul class="list-unstyled d-flex justify-content-center mb-1">
              <li>
                <i class="text-warning fa fa-star"></i>
                <i class="text-warning fa fa-star"></i>
                <i class="text-warning fa fa-star"></i>
                <i class="text-warning fa fa-star"></i>
                <i class="text-warning fa fa-star"></i>
              </li>
            </ul>
            <p class="text-center mb-0">${product.price}</p>
          </div>
        </div>
      </div>
    `;

    container.insertAdjacentHTML('beforeend', productHTML);
  } else {
    console.error('Product or container not provided.');
```

So, the way it works is it creates a constant variable then inside that variable I create a part of html code that I need for that part of my website and uses variables from fetched data from my database.json file. After that it uses .insertAdjacentHTML method to add HTML content to the document.

Translation

For this part of my website, I simply followed a tutorial from W3Schools.com on how to easily add a Google Translate Button on my page. I achieved this by adding a division with a unique id in my footer and by adding Google Translate API in my Website.

```
<div id="google_translate_element"></div>
```

```
<script type="text/javascript" >
  function googleTranslateElementInit() {
    new google.translate.TranslateElement({pageLanguage: 'en'}, 'google_translate_element');
  }
</script>
<script type="text/javascript" src="//translate.google.com/translate_a/element.js?cb=googleTranslateElementInit"></script>
```

Image of Google Translate functioning properly in my website:

