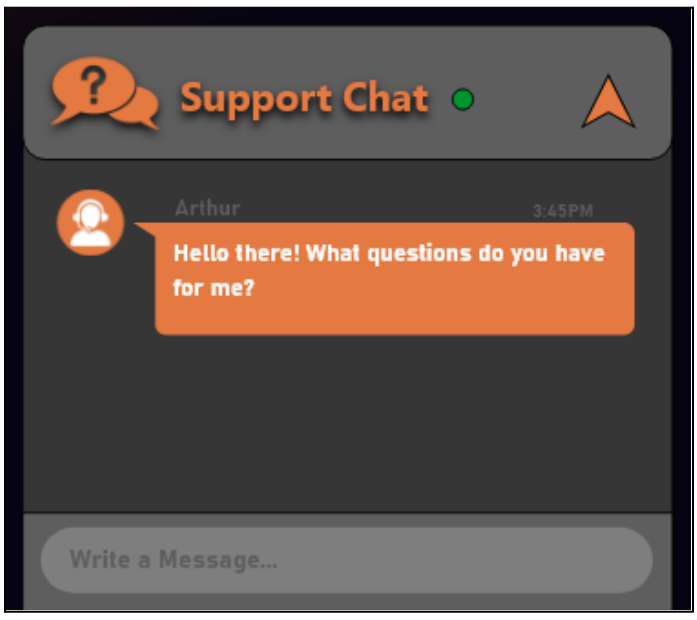
Arthur Levitsky
Dr. Zhou
CSIT335 - Intro to Human Computer Interaction
April 2, 2021

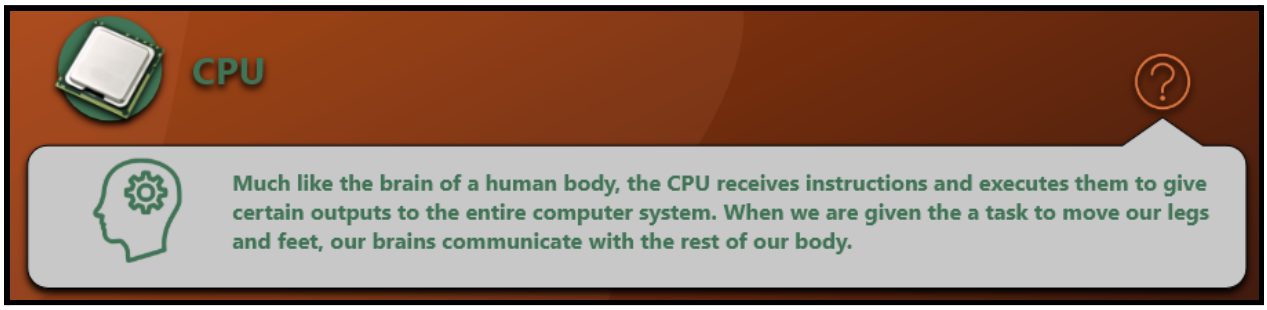
Module 5: Project Implementation (Part IV)

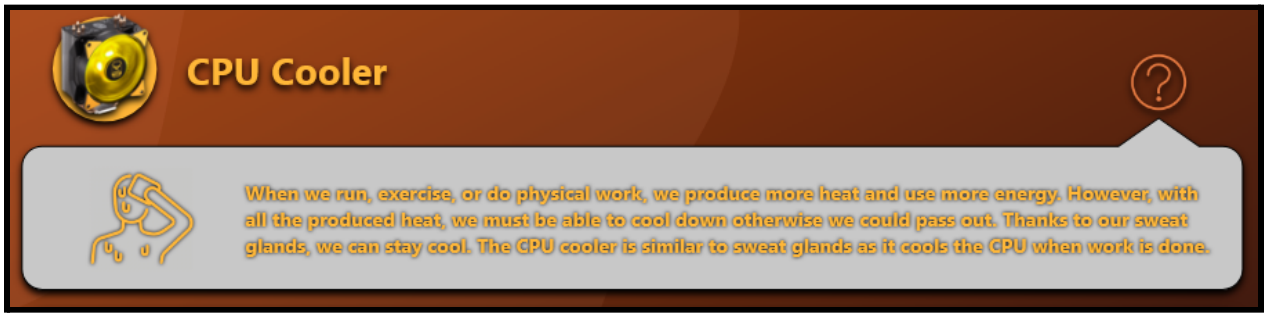
Within the *EZPC Builder* website, the main focus was to construct the *System Builder* section. In this section I have shown **seven features** that explain how the computer builder should work. The features are:

- Support Chat
    - For the support chat, many websites use this feature where you can open a small tab that will allow you to speak to customer service representatives. This feature will require some networking to be done and once the user opens the Support Chat bubble, a secured connection will be established between the customer and the support team. The interface allows the user to send messages and see messages. Once the user sends a message, the message will go through some secured and reliable Web Chat API. The API will have to be programmed in with various languages such as Javascript to allow this to work and HTML and CSS will be used to dynamically display the styling for the user.



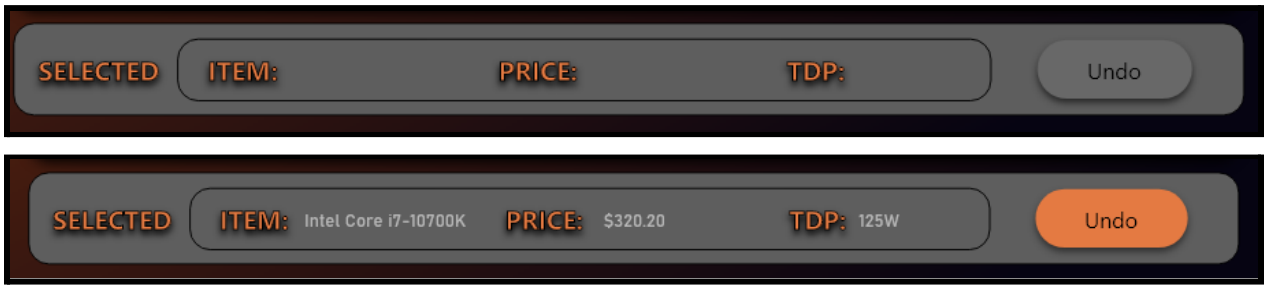
- Question Mark Bubble
    - Users will be able to click on this icon to read more about the component that they are looking at. The information needs to be concise and supportive with analogies for the average user to be able to get an idea. The implementation for the question mark bubble will be very simple. This will solely require the use of HTML and CSS. We will use a button functionality for the icon and when pressed, the CSS will use fading animations to show the contents that the container holds. Once the user is done reading, the button will be pressed again where the animation will fade away and disappear to avoid clutter on the page. The information for each component will be stored in a database like MySQL and can be edited by admins in the future in case needed.



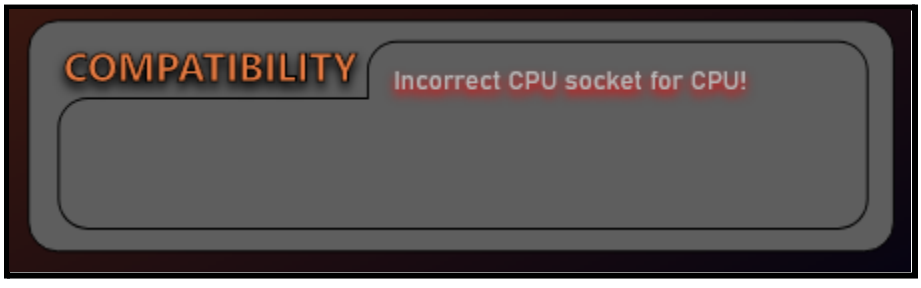


- Browse Component Products
    - The implementation for this will also be fairly simple. For the EZPC Website, we will require a database that stores all the components that the website sells. Each database category will be based on the components (CPU, Motherboard , RAM, etc.). We will most likely use MySQL to store those components and their details. We will use PHP code to help fetch all of these components based on the component section we are in. So if we are in the CPU section, the system should only fetch CPU components that the user can look at. We will have an HTML container that will display all of these components and have a scrollbar function where the user can scroll down to see the full list of components.

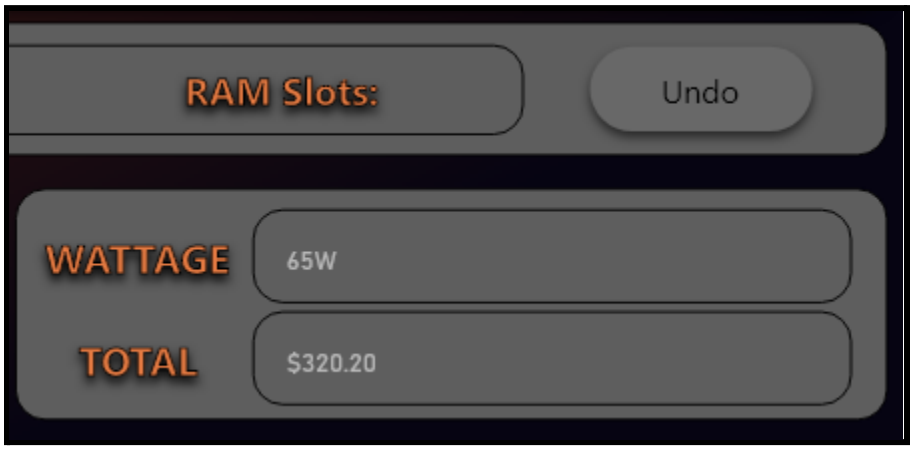
- Select and Undo Components
    - When a user selects a component, the selected component will display in a box. The user can't select the component again once it has been selected, but the user can undo their selection and choose a different component. This all can be done with HTML, CSS, and Javascript. Primarily, we are looking at Javascript to help prevent the user from accessing other pages until they have selected a certain component. Because we don't need to use PHP and the database, Javascript is great at dynamically displaying text once it is already fetched on the page. Javascript code will hold conditions so that the user can feel guided in his computer building journey.

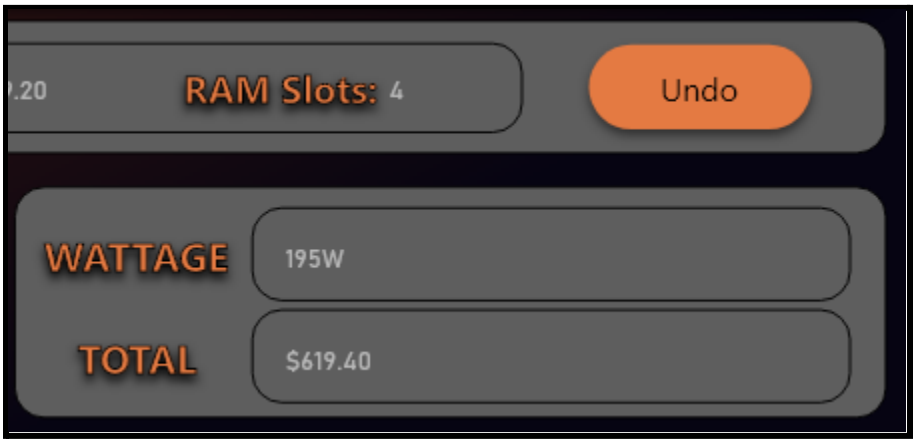


- Compatibility and Availability
    - The compatibility and availability features are extremely important for the *System Builder* section of the website. These features will help the user understand what is compatible with the components that they are choosing and what isn't. This will help eliminate confusion in the long run. In terms of implementation, we will be using Javascript and PHP. PHP will be used to fetch information from the database. The database will store all the possible information that each component has. For example, when we load the component pages, the Javascript code will request PHP to fetch information about the Intel Core i7 CPU that we had selected. When we are in the motherboard page, Javascript will request information from Intel Core i7 CPU regarding the CPU Sockets it can work with. PHP code will then go through each motherboard in the database and check the type of CPU socket that the motherboards require. If the sockets are the same, then they are compatible, if they aren't, then they are not compatible and we would have to output to the user a red *X* mark for compatibility. This implementation would be by far the hardest implementation for the *System Builder* website as it will require numerous algorithms to check for compatibility. The availability will be easy to construct as PHP code can just go through the list of components in the database and see which components have more than '0' for their quantity. If they have '0', then there is no availability and PHP will print out a red *X* mark for availability.



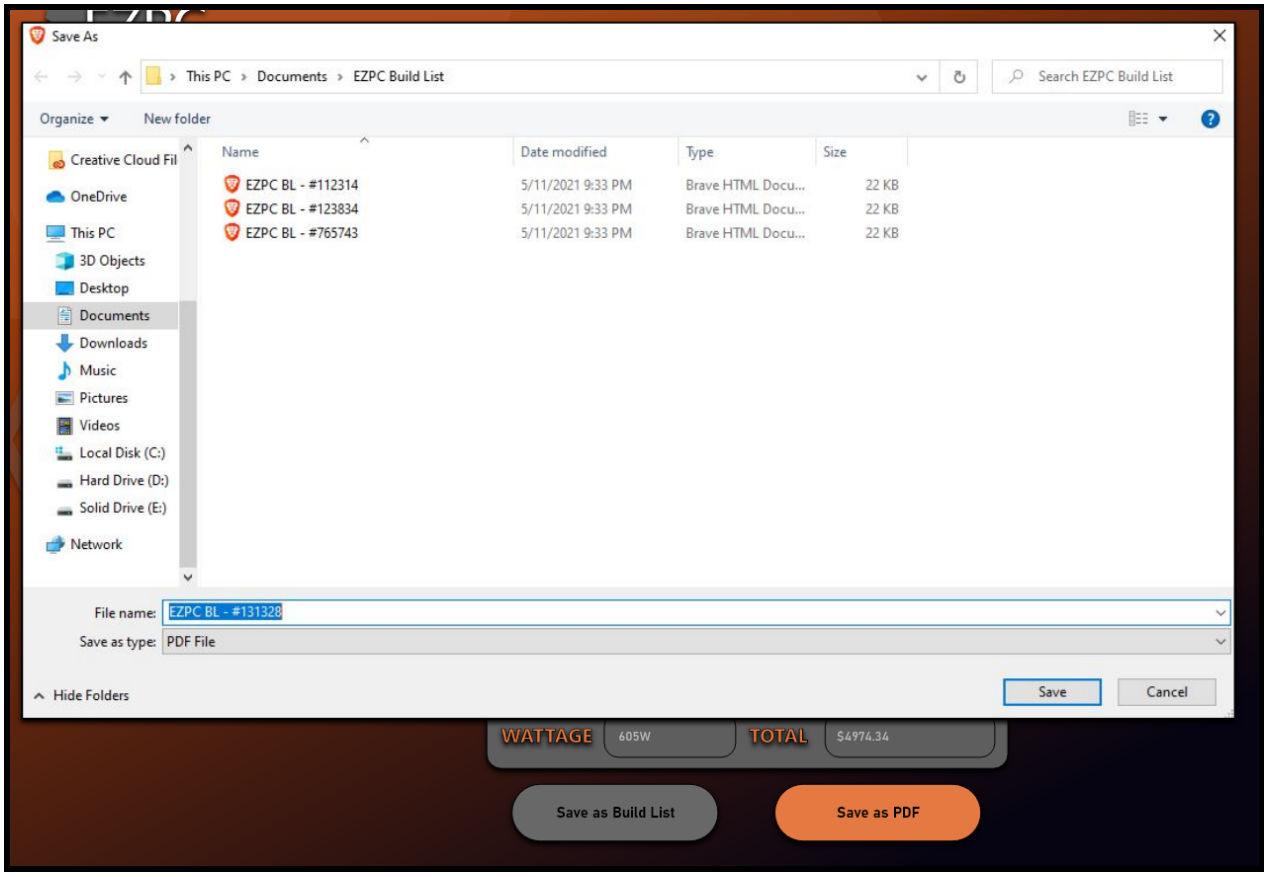
- Wattage and Total
    - Wattage and Total is very important. Each component holds a certain value for the amount of power they require. Without this number, customers might end up making mistakes in terms of choosing which power supply to get. So for MySQL, the database that holds each component will all have a column that states *Wattage* and will have a number. The database will also have a *Price* column for the price of each component. Once the user selects components, the wattage number will increase and so will the price. When other components are selected, the total current wattage will add the wattage of the selected wattage and the current total price will add the selected component's price. This can be done using PHP and Javascript.





- Save as PDF
    - When a customer is not logged into their account, they will be given an option to save their build list into a PDF file that will display all the components they selected, the wattage, and the total price. HTML, CSS, and Javascript code will all help be able to construct this. When the user clicks on Save as PDF, the style of the current page will disappear and the entire page will turn white. Basic HTML table will be created that will have rows based on the computer components and columns based on the computer component details. The HTML table will store the total wattage and total price.


Build Review
EZPC BL - #131328
Product: EVGA GeForce RTX 3090
Price: $3549.99
GPU Memory: 24GB
Case
Product: Phanteks Eclipse P360A
Price: $69.99
Form Factor: ATX
Power Supply
Product: Corsair HX Platinum
Price: $370.00
Power Efficiency: 1000W
WATTAGE
605W
TOTAL
$4974.34
Save as Build List
Save as PDF


Save As
This PC › Documents › EZPC Build List
Search EZPC Build List
Organize
New folder
Creative Cloud Fil
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Hard Drive (D:)
Solid Drive (E:)
Network
Name
Date modified
Type
Size
EZPC BL - #112314
5/11/2021 9:33 PM
Brave HTML Docu...
22 KB
EZPC BL - #123834
5/11/2021 9:33 PM
Brave HTML Docu...
22 KB
EZPC BL - #765743
5/11/2021 9:33 PM
Brave HTML Docu...
22 KB
File name:
EZPC BL - #131328
Save as type:
PDF File
Hide Folders
Save
Cancel
WATTAGE
605W
TOTAL
$4974.34
Save as Build List
Save as PDF