**ONLINE FOOD DELIVERY WEBSITE**

INTERNSHIP PROJECT REPORT

*by*

SAUMYA PANDEY (2000100100158)

**

Department of Computer Science and Engineering

United College of Engineering and Research

Naini, Prayagraj

August & 2022

**BONAFIDE CERTIFICATE**

This is to certify that this project report entitled “Food Delivery” submitted to **United College of Engineering and Research, Prayagraj**, is a Bonafede record of work done by **“Saumya Pandey”** under my supervision from “ **August 18, 2022** ” to “ **September 4, 2022**”.

Mr. Aashish Kapoor

****

**Declaration by Authors**

This is to declare that this report has been written by us. No part of the report is plagiarized from other sources. All information included from other sources have been duly acknowledged. We aver that if any part of the report is found to be plagiarized, we are shall take full responsibility for it.

Saumya Pandey

Roll Number: 2000100100158

**TABLE OF CONTENT**

1. Project:
2. Description
3. Scope
4. Solving Problem
5. Technology Used
6. What is Full Stack Development?
7. Scope of Full Stack Development
8. Benefits
9. HTML:
   1. Description
   2. Role
   3. History
   4. DOM
   5. Basic Body Structure
   6. Basic tags
   7. Tables
   8. List
   9. Form
10. CSS:
    1. Introduction
    2. Syntax
    3. Ways to implement
    4. Selectors
    5. Basic Properties
    6. Text- Property
    7. Font-property
    8. Margin and padding
    9. Background
    10. Advanced Properties
        1. Position
        2. Animation
        3. Keyframe
        4. Pseudo class
        5. Media query

* Implementation
* Description of Project with each steps
* Purpose of modules in your project
* Screenshots with each LOC Description

**PROJECT**

Description:

Our Project is a food delivery website. In this you Customer can place his/her order and he will receive the delivery in next half an hours.

Scope:

In this project we can add cookies, authentication through google, Wishlist ,Payment Gateway .

Problem solving-

We live in an era where everyone is living a fast-paced life. People, especially in cities, live on takeaways instead of cooking their meals. Still, a lot of time is wasted in going and picking up the order, so we’ve made an online ordering food website which allows people to order their food. Our website helps people in browsing menu online and choosing accordingly while sitting and enjoying comfort of their home.

It’s true that nowadays people value their time over anything plus, they want to avoid unnecessary traveling as much as they can. So, our website also provides home delivery.

**Technology Used:**

For the front-end we used:

* + HTML
  + CSS

**HTML**

Description

HTML stands for Hyper Text Markup Language. It is the basic building block of front-end web content.

ROLE

As said previously, HTML is basic building block of a web page. Hypertext refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext. As its name suggests, HTML is a Markup Language which means you use HTML to simply “mark-up” a text document with tags that tell a Web browser how to structure it to display.

HISTORY

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists and so forth and so on. First version of html was written by Tim Berners-Lee in 1993. Most pages on the today were built using either HTML

4.01. HTML 5 is the latest and the current version.

DOM

The Document Object Model (DOM) is a programming interface for web documents. It is an interface that represents how our HTML and XML documents are read by the browser. It allows a language (JavaScript) to manipulate structure and style out website as it allows JavaScript to add change or remove HTML elements and attributes.

BASIC BODY STRUCTURE

<!DOCTYPE html>  
<html>  
<head>  
<title>Page Title</title>  
</head>  
<body>  
  
<h1>My First Heading</h1>  
<p>My first paragraph.</p>  
  
</body>  
</html>

HTML BASIC TAG

1. <!Doc1. Heading tags make the text bigger and bolder. There are six heading tags in html, h1, he, he, h4, he and h6. H1 stands for most important tag whereas h6 for the least one.

<h1>Heading</h1>

<h2>Heading</h2>

<h3>Heading</h3>

<h4>Heading</h4>

<h5>Heading </h5>

<h6>Heading </h6>

1. P tags defines a paragraph. The web browser automatically inserts a single blank line before and after each <p> element to make the text more readable.

<p> Hello, there. Order food from our website. </p>.

1. 3. Img tag allows us to insert images into our web page. It has no closing tab. It includes – src (source), length, width, height.
2. . Anchor tags allow us to link one web page to another page or a section of the same page.

<a></a>

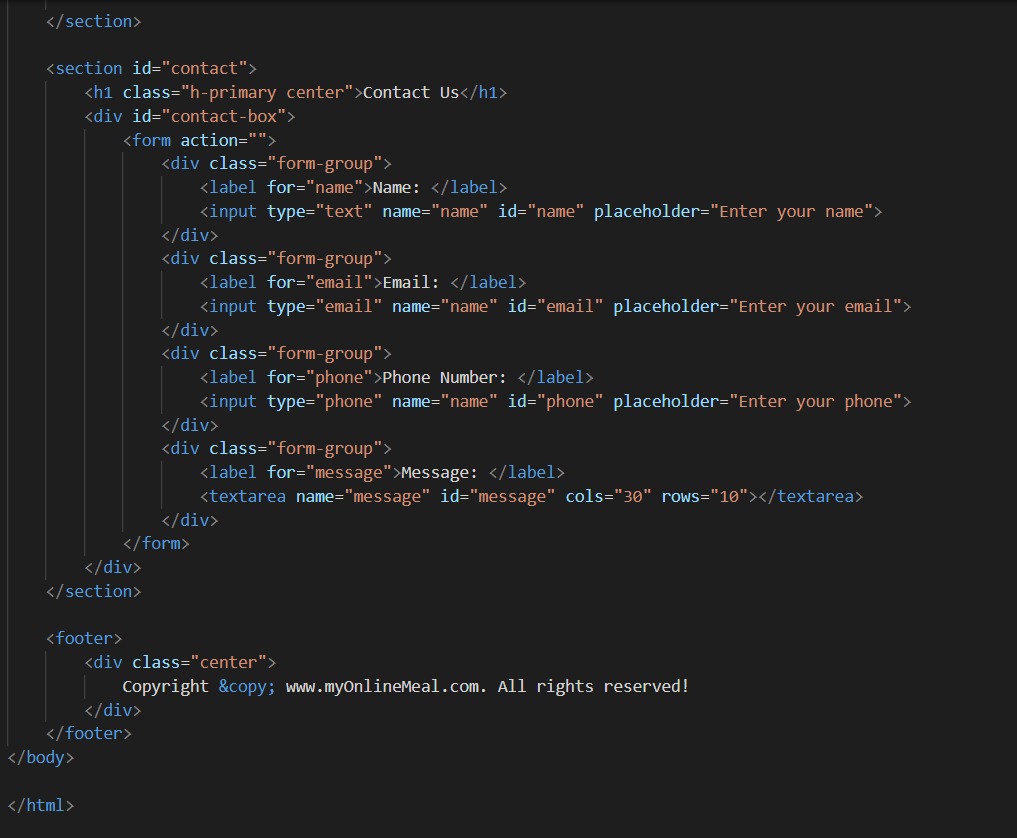
5. Comment tags help programmers to understand the HTML source code. They’re not visible on the web page on browser.

<! —This is a comment🡪

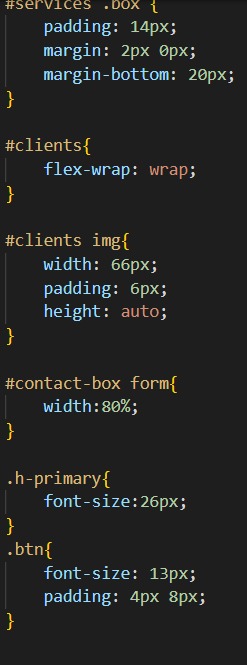
Tables

* HTML to arrange data into rows and columns.
* A table in HTML consists of table cells inside rows and columns.
* Each table cell is defined by a <td> and a </td> tag.
* Each table row starts with a <tr> and ends with a </tr> tag



* 





* Sometimes you want your cells to be table header cells. In those cases use the <th> tag instead of the <td> tag:

Example:

<table>  
  <tr>  
    <th>Company</th>  
    <th>Contact</th>  
    <th>Country</th>  
  </tr>  
  <tr>  
    <td>Alfreds Futterkiste</td>  
    <td>Maria Anders</td>  
    <td>Germany</td>  
  </tr>  
  <tr>  
    <td>Centro comercial Moctezuma</td>  
    <td>Francisco Chang</td>  
    <td>Mexico</td>  
  </tr>  
</table>



LIST

Html lists allow web developers to group a set of related items in lists.

An unordered HTML list starts with the <ul> tag. Each list Item starts with <li> tag. All the items are marked with the bullets automatically.

Plum

Vaseline

Mama earth

An ordered HTML list:

An ordered list starts with <ol> and list item begins with <li> tag and items are marked with numbers by default.

First item

Second item

Third item

Fourth item

FORM

Html form tag is used to create an HTML form and has attributes such as action (for passing data), method (to upload data) and target (where the result will be displayed).

There are three types of text input used on forms –

1. Single line text input controls
2. Password input controls.
3. Multi line text input controls.

CSS

DESCRIPTION

Cascading Style Sheet is a style sheet language used for describing the presentation of a document written in a Markup Language such as HTML. It’s a cornerstone technology of World Wide Web.

ROLE

CSS is used to style and layout web pages as it alters font, colour, size, etc. It also describes how elements should be rendered on scree.

HISTORY

In 1994, HÃ¥kon Wium Lie proposed the idea of CSS.

1996- The first version of CSS was invented.

1998- CSS 2 was released and work on CSS 3 began. CSS 3 was very different from the other versions, fot instead of being a single monolithic specification, it was published as a set of separate documents known as modules.

2011- CSS 2.1 was released, which fixed the errors found in CSS 2

SYNTAX

CSS syntax consists of a selector, property and its values. The selector points to the HTML element you want to style.

The declaration block contains one or more declarations separated by semicolons.

Each declaration includes a CSS property name and a value, separated by a colon.

Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces.

Example-

H1 {font-size:90px; color:red}

WAYS TO IMPLEMENT CSS

There are three ways of implementing

CSS. They’re as follows-

1. Internal/embedded system – An internal CSS is used to define a style for a single HTML page.

An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

1. External system- An external style sheet is used to define the style for many HTML pages.

To use an external style sheet, we add a link to it in the <head> section of each HTML page.

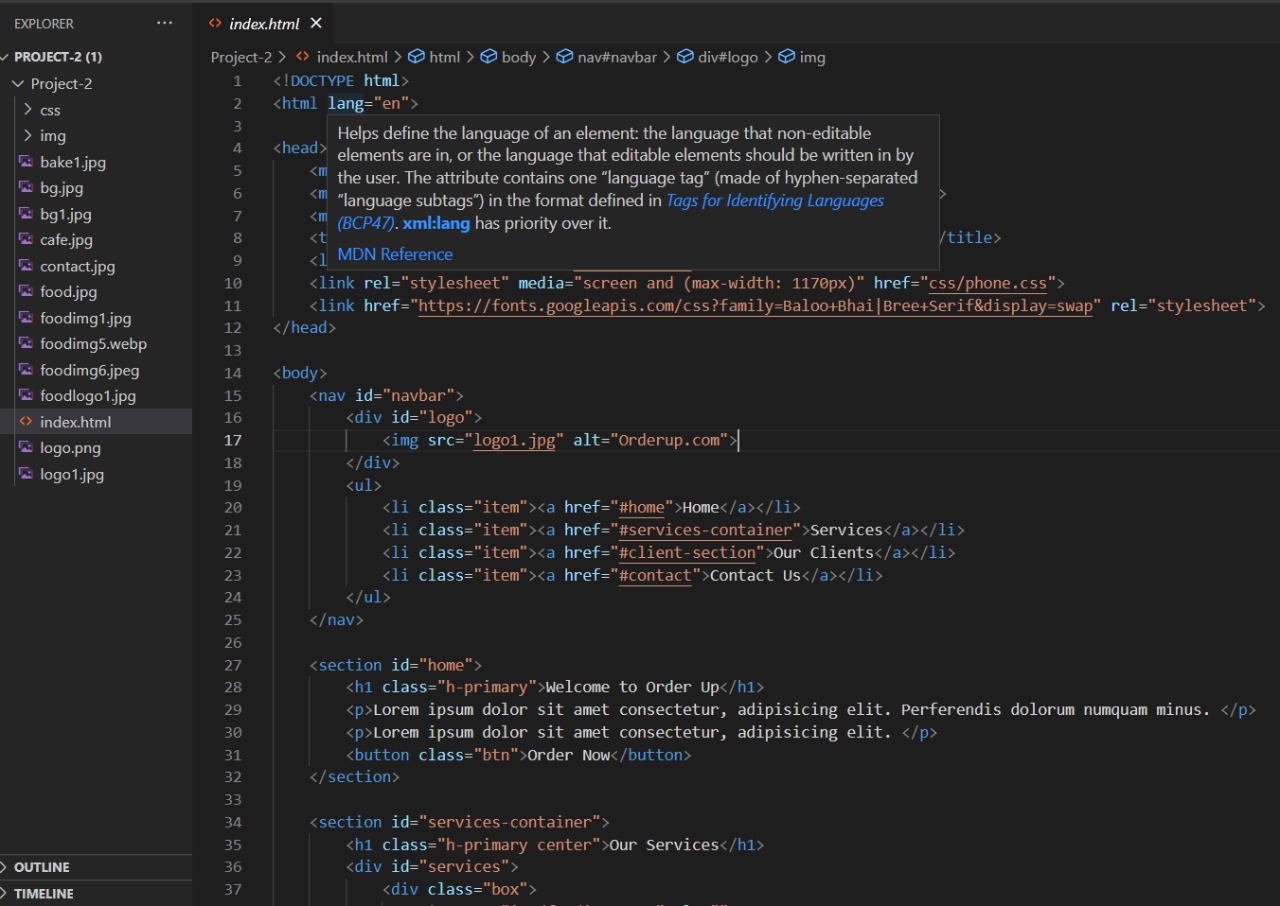
1. Inline system – An inline CSS is used to apply a unique style to a single HTML element.

An inline CSS uses the style attribute of an HTML element.

SELECTORS

CSS selectors are used to select the content we want to style. Selectors are part of CSS rule set. Different type of CSS selectors are –

1. CSS element selector – The element selector selects the HTML element by name.



p {  
  text-align: center;  
  color: red;  
}

1. CSS id selector – It selects the id attribute of an HTML element to select a specific element.

#para1 {  
  text-align: center;  
  color: red;  
}

1. CSS class selector- It selects elements with a specific class attribute.

.center {  
  text-align: center;  
  color: red;  
}

1. CSS universal selector – It selects all the element present on the page.

\* {  
  text-align: center;  
  color: blue;  
}

1. CSS group selector- Selects all HTML elements with the same style definitions.

h1, h2, p {  
  text-align: center;  
  color: red;  
}

BASIC PROPERTIES

TEXT

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| **Color** | Sets the color of a text | RGB, hex, keyword |
| **line-height** | Sets the distance between lines | normal, *number, length, %* |
| **letter-spacing** | Increase or decrease the space between characters | normal, *length* |
| **text-align** | Aligns the text in an element | left, right, center, justify |
| **text-decoration** | Adds decoration to text | none, underline, overline, line-through |
| **text-indent** | Indents the first line of text in an element | *length, %* |
| **text-transform** | Controls the letters in an element | none, capitalize, uppercase, lowercase |

FONT

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| **font** | Sets all the font properties in one declaration | *font-style, font-variant, font-weight, font-size/line-height, font-family,*caption, icon, menu, message-box, small-caption, status-bar, inherit |
| **font-family** | Specifies the font family for text | *family-name, generic-family,*inherit |
| **font-size** | Specifies the font size of text | xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger*, length, %,*inherit |
| **font-style** | Specifies the font style for text | normal, italic, oblique, inherit |
| **font-variant** | Specifies whether or not a text should be displayed in a small-caps font | normal, small-caps, inherit |
| **font-weight** | Specifies the weight of a font | normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900, inherit |

MARGIN AND PADDING

The CSS margin properties are used to create space around elements, outside of any defined borders.

With CSS, you have full control over the margins. There are properties for setting the margin for each side of an element (top, right, bottom, and left).

CSS has properties for specifying the margin for each side of an element:

* margin-top
* margin-right
* margin-bottom
* margin-left

All the margin properties can have the following values:

* auto - the browser calculates the margin
* *length* - specifies a margin in px, pt, cm, etc.
* *%* - specifies a margin in % of the width of the containing element
* inherit - specifies that the margin should be inherited from the parent element

The background property is a shorthand property for:

* [background-color](https://www.w3schools.com/cssref/pr_background-color.asp)
* [background-image](https://www.w3schools.com/cssref/pr_background-image.asp)
* [background-position](https://www.w3schools.com/cssref/pr_background-position.asp)
* [background-size](https://www.w3schools.com/cssref/css3_pr_background-size.asp)
* [background-repeat](https://www.w3schools.com/cssref/pr_background-repeat.asp)
* [background-origin](https://www.w3schools.com/cssref/css3_pr_background-origin.asp)
* [background-clip](https://www.w3schools.com/cssref/css3_pr_background-clip.asp)
* [background-attachment](https://www.w3schools.com/cssref/pr_background-attachment.asp)

ADVANCED PROPERTIES

**Position Property**

The position property is used to position the elements anywhere in our page or window. Position property accepts four different values and each value has its own use cases. The different values are: static, relative, absolute, and fixed. Let's discuss each of them one by one.

**Position Static**

This is the default value for the position property and this does not accept any box-offset property. Each and every element have static position property by default.

**Position Relative**

It accepts all the box off-set properties i.e left, right, top & bottom. The position relative element will always remain in the normal flow of the page, unlike floated elements. If we position any relatively it’s original position will be maintained, no elements can take its original position.

<div class="box">......</div>  
  
 .box {  
 position: relative;  
 left: 60px;  
 top: 60px;  
 }

Here, the box will move 60 pixels from left and 60px from down from its original position.

**Position Absolute**

The Absolute positioned element take reference from the element which is positioned as relative or to make an element absolutely positioned we need to make any parent element relatively positioned. When we apply absolute position property on any element then that element goes out of the normal flow and also loses its original position. The next element takes the position of that element.

<div class="box box-1">......</div>  
 <section class="parent-box">  
 <div class="box box-2">......</div>  
 </section>  
 <div class="box box-3">......</div>  
  
  
 box {  
 width: 100px;  
 height: 100px;  
 background: green;  
 }  
 .parent-box {  
 position: relative;  
 }  
 .box-2 {  
 position: absolute;  
 right: 0;  
 bottom: 0;  
 }

If we want to position the absolute element with relative to its parent then we need to position its parent relatively.

<div class="box box-1">......</div>

**Position Fixed**

Position fixed is similar to absolute positioning. The only difference is that fixed positioning always works with respect to window viewport and does not scroll with content as we scroll the page.

<div class="box">......</div>  
  
 .box {  
 position: fixed;  
 right: 60px;  
 bottom: 60px;  
 }

.box-3 {  
 left: 50px;  
 top: 50px;  
 background: green;  
 z-index: 3

ANIMATION

The animation shorthand CSS property applies an animation between styles. It is a shorthand for animation-name , animation-duration , animation-timing-function , animation-delay , animation-iteration-count , animation-direction , animation-fill-mode , and animation-play-state .

|  |  |
| --- | --- |
| **Value** | **Description** |
| [*animation-name*](https://www.w3schools.com/cssref/css3_pr_animation-name.asp) | Specifies the name of the keyframe you want to bind to the selector |
| [*animation-duration*](https://www.w3schools.com/cssref/css3_pr_animation-duration.asp) | Specifies how many seconds or milliseconds an animation takes to complete |
| [*animation-timing-function*](https://www.w3schools.com/cssref/css3_pr_animation-timing-function.asp) | Specifies the speed curve of the animation |
| [*animation-delay*](https://www.w3schools.com/cssref/css3_pr_animation-delay.asp) | Specifies a delay before the animation will start |
| [*animation-iteration-count*](https://www.w3schools.com/cssref/css3_pr_animation-iteration-count.asp) | Specifies how many times an animation should be played |
| [*animation-direction*](https://www.w3schools.com/cssref/css3_pr_animation-direction.asp) | Specifies whether or not the animation should play in reverse on alternate cycles |
| [*animation-fill-mode*](https://www.w3schools.com/cssref/css3_pr_animation-fill-mode.asp) | Specifies what values are applied by the animation outside the time it is executing |
| [*animation-play-state*](https://www.w3schools.com/cssref/css3_pr_animation-play-state.asp) | Specifies whether the animation is running or paused |
| Initial | Sets this property to its default value. |
| Inherit | Inherits this property from its parent element. [*t*](https://www.w3schools.com/cssref/css_inherit.asp) |

KEYFRAMES

The keyframes rule specifies the animation code.

The animation is created by gradually changing from one set of CSS styles to another.

During the animation, you can change the set of CSS styles many times.

Specify when the style change will happen in percent, or with the keywords "from" and "to", which is the same as 0% and 100%. 0% is the beginning of the animation, 100% is when the animation is complete.

PSEUDO CLASS

A pseudo-class is used to define a special state of an element.

For example, it can be used to:

* Style an element when a user mouses over it
* Style visited and unvisited links differently
* Style an element when it gets focus

MEDIA QUERY

Media query is a CSS technique introduced in CSS3.

It uses the @media rule to include a block of CSS properties only if a certain condition is true.

### **Example**

If the browser window is 600px or smaller, the background color will be lightblue:

@media only screen and (max-width: 600px) {  
  body {  
    background-color: lightblue

}

**Reference:**

<http://www.geeksforgeeks.org/web-technology/html-css/>

<https://www.w3schools.com/html/html_css.asp>

