

MINI PROJECT

(2020-21)

WEBSITE DEVELOPMENT

MID-TERM REPORT



Institute of Engineering & Technology

Submitted By:-

SACHIN GUPTA(181500599)

ASHISH CHAUDHARY(181500140)

ABHISHEK PANDEY(181500020)
ARUN PRATAP SINGH(181500132)

Supervised By:-

Ms. Priya Agarwal

Trainer

DEPARTMENT OF COMPUTER ENGINEERING & APPLICATIONS

CERTIFICATES





CONTENTS

Abstract

1.Introduction

1.1 General Introduction to the topic

1.2 Area of Computer Science

1.3 Hardware & Software Requirements

2. Problem definition

3. Objectives

4. Implementation Details

5. Progress till Date & The Remaining work

6. Some Screenshots

7. References

ABSTRACT

This thesis project consists of the development of a contributory web application to display and gather information on Web Development . The goal of this application is to create the foundation for a web-based spatial inventory of existing . The inventory is meant to advise policymakers and heritage organizations on priority resources to protect the existing base of resources as well as to create a historic record of the historic urban landscape. This is important because the push to modernize infrastructure in emerging nations often leads to the destruction of the colonial heritage fabric in urban areas. An inventory of what currently exists, and possibly what existed in the past, will help to digitally record these sites. As a secondary purpose, the web application was developed to be replicable by others who might choose to take and adapt the web application model for their own heritage conservation-

related purposes. As such, the web application employs easy-to-use, relatively inexpensive, cloud-based tools and services, such as Persona, Heroku, Bitbucket, and various Google products. In order to use many of these services together, the web application was purposebuilt using the well-documented and flexible Python web development framework, in conjunction with the templating system, along with the standard HTML, CSS, and JavaScript programming languages. After instructional documentation is written and further development occurs as specified in the application's roadmap, the code, which is currently stored in a private repository on Bitbucket will be opened for download and collaboration by others wanting to use and improve the application model.

Introduction to Web Development



What is Web Development?

Web development is the coding or programming that enables website functionality, per the owner's requirements. It mainly deals with the non-design aspect of building websites, which includes coding and writing markup.

Web development ranges from creating plain text pages to complex web-based applications, social network applications and electronic business applications.

The web development hierarchy is as follows:

- Client-side coding
- Server-side coding
- Database technology

Among web professionals, "web development" usually refers to the main non-design aspects of building web sites: writing markup and coding. Web development may use content management systems (CMS) to make content changes easier and available with basic technical skills.

For larger organizations and businesses, web development teams can consist of hundreds of people (web developers) and follow standard methods like Agile methodologies while developing websites. Smaller organizations may only require a single permanent or contracting developer, or secondary assignment to related job positions such as a graphic designer or information systems technician. Web development may be a collaborative effort between departments rather than the domain of a designated department. There are three kinds of web developer specialization: front-end developer, back-end developer, and full-stack developer. Front-end developers responsible for behavior and visuals that run in the user browser, while back-end developers deal with the servers.

A basic website or may be a highly professional website have two ends :

- Front End – User Interface
- Back End – Website Functionality

Basic Languages used in Front end development

- HTML
- CSS
- Bootstrap

Basic Languages used in back end development

- SQL
- PHP

What is Front End Web development ?

Front-end web development, also known as client-side development is the practice of producing HTML, CSS and JavaScript for a website or Web Application so that a user can see and interact with them directly. The challenge associated with front end development is that the tools and techniques used to create the front end of a website change constantly and so the developer needs to constantly be aware of how the field is developing.

The objective of designing a site is to ensure that when the users open up the site they see the information in a format that is easy to read and relevant. This is further complicated by the fact that users now use a large variety of devices with varying screen sizes and resolutions thus forcing the designer to take into consideration these aspects when designing the site. They need to ensure that their site comes up correctly in different browsers (cross-browser), different operating systems (cross-platform) and different devices (cross-device), which requires careful planning on the side of the developer.

A front-end developer architects and develops websites and applications using web technologies (i.e., HTML, CSS, DOM, and JavaScript), which run on the Open Web Platform or act as compilation input for non-web platform environments.

Typically, a person enters into the field of front-end development by learning to develop HTML, CSS, and JavaScript which commonly runs in a web browser but can also run in a headless browser, WebView, or as compilation input for a native runtime environment.

What is HTML ?

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

HTML is not a programming language, meaning it doesn't have the ability to create dynamic functionality. Instead, it makes it possible to organize and format documents, similarly to Microsoft Word.

When working with HTML, we use simple code structures (tags and attributes) to mark up a website page. For example, we can create a paragraph by placing the enclosed text within a starting `<p>` and closing `</p>` tag.

1. `<p>`This is how you add a paragraph in HTML.`</p>`
2. `<p>`You can have more than one!`</p>`

Overall, HTML is a markup language that is really straightforward and easy to learn even for complete beginners in website building. Here's what you'll learn by reading this article:

- [The History of HTML](#)
- [How Does HTML Work?](#)
- [Overviewing The Most Used HTML Tags](#)
 - [Block-Level Tags](#)
 - [Inline Tags](#)
- [HTML Evolution, What Differs Between HTML and HTML5?](#)
- [Pros and Cons of HTML](#)
- [How are HTML, CSS, and JavaScript related?](#)
- [So...What is HTML?](#)

The History of HTML

HTML was invented by [Tim Berners-Lee](#), a physicist at the CERN research institute in Switzerland. He came up with the idea of an Internet-based hypertext system.

Hypertext means a text that contains references (links) to other texts that viewers can access immediately. He published the first version of HTML in 1991, consisting of 18 HTML tags. Since then, each new version of the HTML language came with new tags and attributes (tag modifiers) to the markup.

According to Mozilla Developer Network's [HTML Element Reference](#), currently, there are 140 HTML tags, although some of them are already obsolete (not supported by modern browsers).

Due to a quick rise in popularity, HTML is now considered an official web standard. The HTML specifications are maintained and developed by the World Wide Web Consortium (W3C). You can check out the latest state of the language anytime on [W3C's website](#).

The biggest upgrade of the language was the introduction of **HTML5** in 2014. It added several new semantic tags to the markup, that reveal the meaning of their own content, such as `<article>`, `<header>`, and `<footer>`.

How Does HTML Work?

HTML documents are files that end with a `.html` or `.htm` extension. You can view them using any web browser (such as Google Chrome, Safari, or Mozilla Firefox). The browser reads the HTML file and renders its content so that internet users can view it.

Usually, the average [website includes several different HTML pages](#). For instance: home pages, about pages, contact pages would all have separate HTML documents.

Each HTML page consists of a set of **tags** (also called **elements**), which you can refer to as the building blocks of web pages. They create a hierarchy that structures the content into sections, paragraphs, headings, and other content blocks.

Most HTML elements have an opening and a closing that use the `<tag></tag>` syntax.

Below, you can see a code example of how HTML elements can be structured:

```
1. <div>
2. <h1>The Main Heading</h1>
3. <h2>A catchy subheading</h2>
4. <p>Paragraph one</p>
5. 
6. <p>Paragraph two with a <a href="https://example.com">hyperlink</a></p>
7. </div>
```

- The outmost element is a simple division (`<div></div>`) you can use to mark up bigger content sections.
- It contains a heading (`<h1></h1>`), a subheading (`<h2></h2>`), two paragraphs (`<p></p>`), and an image (``).
- The second paragraph includes a link (`<a>`) with a href attribute that contains the destination URL.
- The image tag also has two attributes: *src* for the image path and *alt* for the image description.

Overviewing The Most Used HTML Tags

HTML tags have two main types: **block-level** and **inline tags**.

1. Block-level elements take up the full available space and always start a new line in the document. Headings and paragraphs are a great example of block tags.
2. Inline elements only take up as much space as they need and don't start a new line on the page. They usually serve to format the inner contents of block-level elements. Links and emphasized strings are good examples of inline tags.

Block-Level Tags

The three block level tags every HTML document needs to contain are `<html>`, `<head>`, and `<body>`.

1. The `<html></html>` tag is the highest level element that encloses every HTML page.
2. The `<head></head>` tag holds meta information such as the page's title and charset.
3. Finally, the `<body></body>` tag encloses all the content that appears on the page.

```
1. <html>
2. <head>
3. <!-- META INFORMATION -->
4. </head>
5. <body>
6. <!-- PAGE CONTENT -->
7. </body>
```

8. `</html>`

- Headings have 6 levels in HTML. They range from `<h1></h1>` to `<h6></h6>`, where h1 is the highest level heading and h6 is the lowest one. Paragraphs are enclosed by `<p></p>`, while blockquotes use the `<blockquote></blockquote>` tag.
- Divisions are bigger content sections that typically contain several paragraphs, images, sometimes blockquotes, and tags for ordered lists and `` for unordered ones. Individual list items must be enclosed by the `` tag. For example, this is how a basic unordered list looks like in HTML: other smaller elements. We can mark them up using the `<div></div>` tag. A div element can contain another div tag inside it as well.
- You may also use ``

1. ``
2. `List item 1`
3. `List item 2`
4. `List item 3`
5. ``

Inline Tags

Many inline tags are used to format text. For example, a `` tag would render an element in **bold**, whereas `` tags would show it in *italics*.

Hyperlinks are also inline elements that require `<a>` tags and **href** attributes to indicate the link's destination:

1. `Click me!`

Images are inline elements too. You can add one using `` without any closing tag. But you will also need to use the **src** attribute to specify the image path, for example:

1. ``

If you want to learn more HTML tags, consider checking our [complete HTML cheat sheet](#) (which is also available for download).

HTML Evolution. What Differs Between HTML and HTML5?

Since the first days, HTML has gone through an incredible evolution. W3C constantly publish new versions and updates, while historical milestones get dedicated names as well.

HTML4 (these days commonly referred to as “HTML”) was published in 1999, while the latest major version came out in 2014. Named **HTML5**, the update has introduced many new features to the language.

One of the most anticipated features of HTML5 is native support for audio and video embedding. Instead of using Flash player, we can simply embed videos and audio files to our web pages using the new `<audio></audio>` and `<video></video>` tags. It also includes in-built support for scalable vector graphics (SVG) and MathML for mathematical and scientific formulas.

HTML5 introduced a few semantic improvements as well. The new semantic tags inform browsers about the meaning of content, which benefits both readers and search engines.

The most popular semantic tags

are `<article></article>`, `<section></section>`, `<aside></aside>`, `<header></header>`, and `<footer></footer>`.

What is CSS ?

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

Advantages of CSS

- **CSS saves time** – You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
- **Pages load faster** – If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.
- **Easy maintenance** – To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
- **Superior styles to HTML** – CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
- **Multiple Device Compatibility** – Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.
- **Global web standards** – Now HTML attributes are being deprecated and it is being recommended to use CSS. So its a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

Who Creates and Maintains CSS?

CSS is created and maintained through a group of people within the W3C called the CSS Working Group. The CSS Working Group creates documents called specifications. When a specification has been discussed and officially ratified by the W3C members, it becomes a recommendation.

These ratified specifications are called recommendations because the W3C has no control over the actual implementation of the language. Independent companies and organizations create that software.

NOTE – The World Wide Web Consortium, or W3C is a group that makes recommendations about how the Internet works and how it should evolve.

CSS Versions

Cascading Style Sheets level 1 (CSS1) came out of W3C as a recommendation in December 1996. This version describes the CSS language as well as a simple visual formatting model for all the HTML tags.

CSS2 became a W3C recommendation in May 1998 and builds on CSS1. This version adds support for media-specific style sheets e.g. printers and aural devices, downloadable fonts, element positioning and tables.

What is Bootstrap ?

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation and other interface components.

Bootstrap, originally named Twitter Blueprint, was developed by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden.

Why Use Bootstrap?

- **Mobile first approach** – Bootstrap 3, framework consists of Mobile first styles throughout the entire library instead of in separate files.
- **Browser Support** – It is supported by all popular browsers.



- **Easy to get started** – With just the knowledge of HTML and CSS anyone can get started with Bootstrap. Also the Bootstrap official site has a good documentation.
- **Responsive design** – Bootstrap's responsive CSS adjusts to Desktops, Tablets and Mobiles. More about the responsive design is in the chapter Bootstrap Responsive Design.



- Provides a clean and uniform solution for building an interface for developers.
- It contains beautiful and functional built-in components which are easy to customize.
- It also provides web based customization.
- And best of all it is an open source.

What Bootstrap Package Includes?

- **Scaffolding** – Bootstrap provides a basic structure with Grid System, link styles, and background. This is covered in detail in the section **Bootstrap Basic Structure**
- **CSS** – Bootstrap comes with the feature of global CSS settings, fundamental HTML elements styled and enhanced with extensible classes, and an advanced grid system. This is covered in detail in the section **Bootstrap with CSS**.
- **Components** – Bootstrap contains over a dozen reusable components built to provide iconography, dropdowns, navigation, alerts, pop-overs, and much more. This is covered in detail in the section **Layout Components**.
- **JavaScript Plugins** – Bootstrap contains over a dozen custom jQuery plugins. You can easily include them all, or one by one. This is covered in details in the section **Bootstrap Plugins**.
- **Customize** – You can customize Bootstrap's components, LESS variables, and jQuery plugins to get your very own version.

What is Back End Web Development ?

So what makes the front end of a website possible? Where is all that data stored? This is where the back end comes in. The back end of a website consists of a server, an application, and a database. A back-end developer builds and maintains the technology that powers those components which, together, enable the user-facing side of the website to even exist in the first place.

SKILLS AND TOOLS

In order to make the server, application, and database communicate with each other, back-end devs use server-side languages like PHP, Ruby, Python, Java, and .Net to build an application, and tools like MySQL, Oracle, and SQL Server to find, save, or change data and serve it back to the user in front-end code. Job openings for back-end developers often also call for experience with PHP frameworks like Zend, Symfony, and CakePHP; experience with version control software like SVN, CVS, or Git; and experience with Linux as a development and deployment system.

Back-end devs use these tools to create or contribute to web applications with clean, portable, well-documented code. But before writing that code, they need to collaborate with business stakeholders to understand their particular needs, then translate those into technical requirements and come up with the most effective and efficient solution for architecting the technology.

What is MySQL?

SQL (Structured Query Language) is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS). It is particularly useful in handling structured data where there are relations between different entities/variables of the data. SQL offers two main advantages over older read/write APIs like ISAM or VSAM. First, it introduced the concept of accessing many records with one single command; and second, it eliminates the need to specify *how* to reach a record, e.g. with or without an index.

What is PHP?

PHP: Hypertext Preprocessor (or simply **PHP**) is a general-purpose programming language originally designed for web development. It was originally created by Rasmus Lerdorf in 1994; the PHP reference implementation is now produced by The PHP Group. PHP originally stood for *Personal Home Page*, but it now stands for the recursive initialism *PHP: Hypertext Preprocessor*.

PHP code may be executed with a command line interface (CLI), embedded into HTML code, or used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a

PHP interpreter implemented as a module in a web server or as a Common Gateway Interface (CGI) executable. The web server outputs the results of the interpreted and executed PHP code, which may be any type of data, such as generated HTML code or binary image data. PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control.

Common uses of PHP

- PHP performs system functions, i.e. from files on a system it can create, open, read, write, and close them.
- PHP can handle forms, i.e. gather data from files, save data to a file, through email you can send data, return data to the user.
- You add, delete, modify elements within your database through PHP.
- Access cookies variables and set cookies.
- Using PHP, you can restrict users to access some pages of your website.
- It can encrypt data.

Characteristics of PHP

Five important characteristics make PHP's practical nature possible –

- Simplicity
- Efficiency
- Security
- Flexibility
- Familiarity

"Hello World" Script in PHP

To get a feel for PHP, first start with simple PHP scripts. Since "Hello, World!" is an essential example, first we will create a friendly little "Hello, World!" script.

As mentioned earlier, PHP is embedded in HTML. That means that in amongst your normal HTML (or XHTML if you're cutting-edge) you'll have PHP statements like this –

[Live Demo](#)

```
<html>

<head>
  <title>Hello World</title>
</head>

<body>
  <?php echo "Hello, World!";?>
```

```
</body>
</html>
```

It will produce following result –

Hello, World!

If you examine the HTML output of the above example, you'll notice that the PHP code is not present in the file sent from the server to your Web browser. All of the PHP present in the Web page is processed and stripped from the page; the only thing returned to the client from the Web server is pure HTML output.

All PHP code must be included inside one of the three special markup tags ATE are recognised by the PHP Parser.

```
<?php PHP code goes here ?>
<?  PHP code goes here ?>
<script language = "php"> PHP code goes here </script>
```

A most common tag is the <?php...?> and we will also use the same tag in our tutorial.

From the next chapter we will start with PHP Environment Setup on your machine and then we will dig out almost all concepts related to PHP to make you comfortable with the PHP language.

Hardware Requirements

- 2.0 GHz processor for faster processing power.
- 1 GB RAM
- 2-3 GB Hard Disk Storage
- OS : Windows, MacOS or Linux

Software Requirements

- Sublime text editor
- JQuery, Bootstrap – for the front end &
- WAMP server – To handle communication with Website and to store data from the website

INTRODUCTION TO PROJECT

Through the six week of the training program, I have been working on a creating some website landing pages. There are of 2 different HTML pages. The website will consist of HTML, CSS, BOOTSTRAP, PHP and MYSQLI code. The HTML, CSS and BOOTSTRAP part of your code decides how your website will look, while the PHP and MySQL part decides how it will function. There are 2 major tasks during this program. They will be structured as follows :

- 1) HTML , CSS AND BOOTSTRAP : where I have design the look of each page of your website.
- 2) PHP &MySQL: where I willadd functionality to the website. This will allow you to allow users to register and buy flowers on the website.

The main pages that need to be designed are:

1. Blog page
2. Startup Landing page

Working

HTML code for blog page

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>blog</title>
```

```
    <link rel="stylesheet" type="text/css" href="blog.css">
```

```
</head>
```

```
<link
```

```
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:400,700&display=swap"
rel="stylesheet">
```

```
<body>
```

```
    <div class="date">JUNE 13 2019</div>
```

```
<div class="heading"><h2>This Is My First Article</h2></div>
```

```
<div class="quote"><p>Bacon ipsum dolor amet capicola strip steak landjaeger, biltong
spare ribs rump cow ground round andouille sirloin pork. Short ribs pig prosciutto swine.
Flank turducken turkey rump, leberkas shoulder bresaola ham hock tail drumstick corned
beef. Venison pork chop beef jowl short ribs.</p></div>
```

<p>Bresaola short ribs pastrami, beef ribs spare ribs kielbasa ham tongue kevin landjaeger chicken ball tip. Pork chop beef kevin strip steak, chicken pork belly pastrami ham hock flank shoulder chuck turkey ribeye andouille ball tip. Leberkas ham ham hock pork loin. Filet mignon bacon pancetta leberkas turducken fatback tongue frankfurter jowl. Shoulder tenderloin chicken shank bacon shankle sirloin.</p>

<p>Pork pig pork loin prosciutto meatball turkey beef ribs ground round. Pork belly salami shank pork chop turducken ribeye swine shoulder tri-tip fatback cupim short loin chuck strip steak. Rump pork chop t-bone.</p>

<hr>

<div class="date">AUGUST 15 2019</div>

<div class="heading"><h2>This Is Another Article</h2></div>

<div class="quote"><p>Bacon ipsum dolor amet capicola strip steak landjaeger, biltong spare ribs rump cow ground round andouille sirloin pork. Short ribs pig prosciutto swine. Flank turducken turkey rump, leberkas shoulder bresaola ham hock tail drumstick corned beef. Venison pork chop beef jowl short ribs.</p></div>

<p>Shankle beef ribs tongue strip steak flank landjaeger capicola hamburger chuck pancetta kevin. Sirloin landjaeger chicken, bresaola brisket swine short ribs turkey short loin ball tip porchetta ham hock. Capicola frankfurter jowl short loin. Kevin flank hamburger, beef venison shankle short loin bresaola frankfurter</p>

<p>Bresaola short ribs pastrami, beef ribs spare ribs kielbasa ham tongue kevin landjaeger chicken ball tip. Pork chop beef kevin strip steak, chicken pork belly pastrami ham hock flank shoulder chuck turkey ribeye andouille ball tip. Leberkas ham ham hock pork loin. Filet mignon bacon pancetta leberkas turducken fatback tongue frankfurter jowl. Shoulder tenderloin chicken shank bacon shankle sirloin.</p>

```
</body>
```

```
</html>
```

CSS code for blog page

```
body{  
  
    border: 20px solid #bdc3c7;  
  
    margin:20px auto;  
  
    max-width: 700px;  
  
    width: 80%;  
  
    padding:20px;  
  
    font-family: Source Sans Pro ;  
  
    padding-bottom: 10px;  
  
}  
  
.heading {  
  
    color:#2c3e58;  
  
    font-size: 2rem;  
  
}  
  
.quote{  
  
    border-left: 5px solid #bdc3c7;  
  
    padding-left: 2px;  
  
}  
  
.date{  
  
    text-transform: uppercase;
```



```
        letter-spacing: 5px;

        color: #3498db;

        font-weight: bold;
    }

hr {

    border: 0;

    height: 1px;

    background-image: -webkit-linear-gradient(left, #f0f0f0, #8c8b8b, #f0f0f0);

    background-image: -moz-linear-gradient(left, #f0f0f0, #8c8b8b, #f0f0f0);

    background-image: -ms-linear-gradient(left, #f0f0f0, #8c8b8b, #f0f0f0);

    background-image: -o-linear-gradient(left, #f0f0f0, #8c8b8b, #f0f0f0);

}
```

Output

JUNE 13 2019

This Is My First Article

Bacon ipsum dolor amet capicola strip steak landjaeger, biltong spare ribs rump cow ground round andouille sirloin pork. Short ribs pig prosciutto swine. Flank turducken turkey rump, leberkas shoulder bresaola ham hock tail drumstick corned beef. Venison pork chop beef jowl short ribs.

Bresaola short ribs pastrami, beef ribs spare ribs kielbasa ham tongue kevin landjaeger chicken ball tip. Pork chop beef kevin strip steak, chicken pork belly pastrami ham hock flank shoulder chuck turkey ribeye andouille ball tip. Leberkas ham ham hock pork loin. Filet mignon bacon pancetta leberkas turducken fatback tongue frankfurter jowl. Shoulder tenderloin chicken shank bacon shankle sirloin.

Pork pig pork loin prosciutto meatball turkey beef ribs ground round. Pork belly salami shank pork chop turducken ribeye swine shoulder tri-tip fatback cupim short loin chuck strip steak. Rump pork chop t-bone.

AUGUST 15 2019

This Is Another Article

Bacon ipsum dolor amet capicola strip steak landjaeger, biltong spare ribs rump cow ground round andouille sirloin pork. Short ribs pig prosciutto swine. Flank turducken turkey rump, leberkas shoulder bresaola ham hock tail drumstick corned beef. Venison pork chop beef jowl short ribs.

Shankle beef ribs tongue strip steak flank landjaeger capicola hamburger chuck pancetta kevin. Sirloin landjaeger chicken, bresaola brisket swine short ribs turkey short loin ball tip porchetta ham hock. Capicola frankfurter jowl short loin. Kevin flank hamburger, beef venison shankle short loin bresaola frankfurter

Bresaola short ribs pastrami, beef ribs spare ribs kielbasa ham tongue kevin landjaeger chicken ball tip. Pork chop beef kevin strip steak, chicken pork belly pastrami ham hock flank shoulder chuck turkey ribeye andouille ball tip. Leberkas ham ham hock pork loin. Filet mignon bacon pancetta leberkas turducken fatback tongue frankfurter jowl. Shoulder tenderloin chicken shank bacon shankle sirloin.

HTML code for Startup landing page

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Image Gallery</title>
```

```
    <link rel="stylesheet"
```

```
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
```

```
integrity="sha384-
```

```
BVYiISiFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"
```

```
crossorigin="anonymous">
```

```
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.4.0/css/font-awesome.min.css">
```

```
    <link rel="stylesheet" type="text/css" href="image gallery.css">
```

```
</head>
```

```
<body>
```

```
    <nav class="navbar navbar-inverse navbar-fixed-top">
```

```
    <div class="container">
```

```
        <!-- Brand and toggle get grouped for better mobile display -->
```

```
        <div class="navbar-header">
```

```
<button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-
target="#bs-example-navbar-collapse-1" aria-expanded="false">
```

```
<span class="sr-only">Toggle navigation</span>
```

```
<span class="icon-bar"></span>
```

```
<span class="icon-bar"></span>
```

```
<span class="icon-bar"></span>
```

```
</button>
```

```
<a class="navbar-brand" href="#"><span class="glyphicon glyphicon-picture" aria-
hidden="true"></span> IMGS</a>
```

```
</div>
```

```
<!-- Collect the nav links, forms, and other content for toggling -->
```

```
<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
```

```
<ul class="nav navbar-nav">
```

```
<li class="active"><a href="#">About <span class="sr-
only">(current)</span></a></li>
```

```
<li><a href="#">Contact</a></li>
```

```
</li>
```

```
</ul>
```

```
<ul class="nav navbar-nav navbar-right">
```

```
<li><a href="#">Sign Up</a></li>
```

```
<ul class="nav navbar-nav navbar-right">
```

```
<li><a href="#">Login</a></li>
```

```
</ul>
```

```
</div><!-- /.navbar-collapse -->
```

```
</div><!-- /.container-fluid -->
```

```
</nav>
```

```
<div class="container">
```

```
<div class="jumbotron">
```

```
<h1><i class="fa fa-camera-retro" aria-hidden="true"></i> The Image Gallery</h1>
```

```
<p>A bunch of beautiful images that I didn't take( like no one:) )</p>
```

```
</div>
```

```
<div class="row">
```

```
<div class="col-lg-4 col-md-6 col-sm-6">
```

```
<div class="thumbnail">
```

```

```

```
</div>
```

```
</div>
```

```
<div class="col-lg-4 col-md-6 col-sm-6">
```

```
<div class="thumbnail">
```


</div>

</div>

<div class="col-lg-4 col-md-6 col-sm-6">

<div class="thumbnail">

</div>

</div>

</div>

<div class="row">

<div class="col-lg-4 col-md-6 col-sm-6">

<div class="thumbnail">

</div>

<div class="col-lg-4">

<div class="thumbnail">

</div>

</div>

<div class="col-lg-4 col-md-6 col-sm-6">

<div class="thumbnail">

</div>

</div>

</div>

<div class="row">

<div class="col-lg-4 col-md-6 col-sm-6">

<div class="thumbnail">

</div>

</div>

<div class="col-lg-4 col-md-6 col-sm-6">

<div class="thumbnail">

</div>

</div>

<div class="col-lg-4 col-md-6 col-sm-6">

<div class="thumbnail">

</div>

</div>

</div>

</div>

<script

src="http://code.jquery.com/jquery-3.4.1.js"

integrity="sha256-WpOohJOqMqqyKL9FccASB9O0KwACQJpFTUBLTYOVvVU="

crossorigin="anonymous"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"

integrity="sha384-

Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNlPg9mGCD8wGNlCPD7Txa"

crossorigin="anonymous"></script>

</body>

</html>

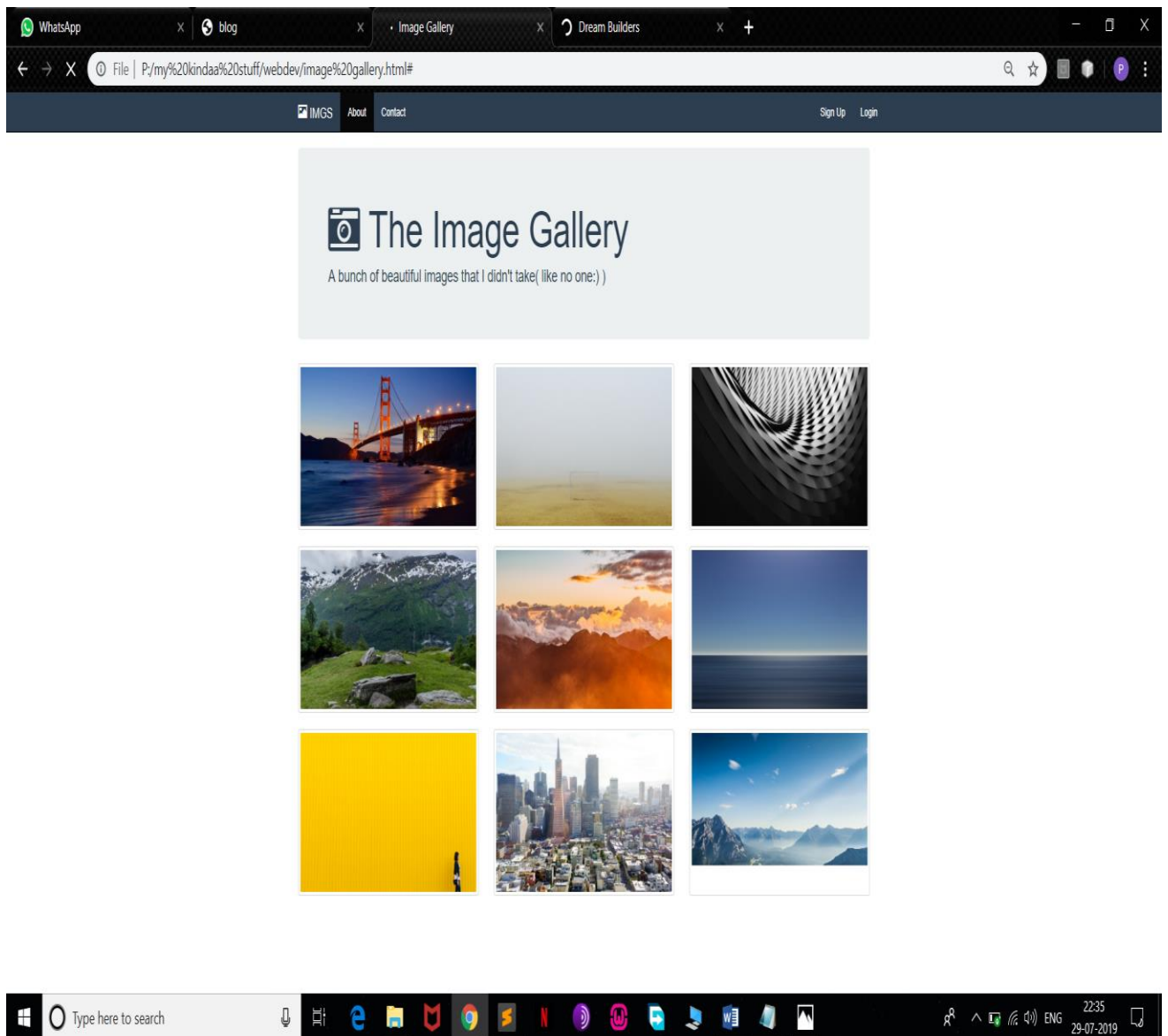
CSS code for startup landing page

```
body{  
  
    padding-top: 70px;  
  
}  
  
.jumbotron{  
  
    color: #2c3e50;  
  
    background:#ecf0f1;  
  
}  
  
.navbar-inverse{  
  
    background: #2c3e50;  
  
}  
  
.navbar-inverse .navbar-brand{  
  
    color: white;  
  
}  
  
.navbar-inverse .navbar-nav>li>a{
```

```
color:white;

}
```

Output



CONCLUSION

A Web Development project, This project may seem relatively simple as a Web Development project but I think it shows that I can work well with HTML, CSS and Bootstrap and I think the code is clean, logical and commented to make it clearer to whoever's reading it.

So our website is a very basic implementation of different web developments languages. I will be developing this project further later on and add new features.

Future Plans regarding project:

- Developing back end of this website so that it would be a complete package.
- Using PHP and MYSQL for providing backend stability.

As I said it is a very basic website , it will have some new and exciting features in future.