

Web and Multimedia Technologies – Project Report

Surname	Bottini
Name	Mattia
Student ID	507798
E-mail	mattia.bottini01@universitadipavia.it
Date of written test	22/02/2022

Project title (website subject)
National Museum of Colorado – Website for a science museum

HTML version
<input checked="" type="checkbox"/> HTML5 <input type="checkbox"/> HTML4 <input type="checkbox"/> XHTML

Client-side technologies used
Javascript

Server-side technologies used
PHP

Website content description

Describe the website in terms of:

- *Content of the single pages* and/or of the *main sections* (e.g., “The products page lists all the...”, “The ‘News’ section presents...”)
- *Graphic design* (e.g., “The navigation menu is located in the left part of the page, immediately below the header, and...”)

Specify whether, for the definition of the content and/or of the graphic structure of your site, you have “drawn inspiration” from one or more existing websites (indicating their URLs).

This section can be extended to the next page, if necessary—but possibly avoid exceeding 50 lines.

The project has been created to simulate a possible website for a Natural Science Museum in Colorado.

In the Home Page there is a recap of the main contents of the whole website. There is a brief description, using different articles, of the variety of collections present in the museum (Botany, Zoology and Mineralogy), a remainder of which information a user can find in “*Info*” section, surrounded by the possible type of visits that can be chosen by visitors and the ways to contact the museum if further information is required. There is also a disclaimer put in the bottom-right part to remind the use of the mask.

The three collections in the museum are described in separated sections: “*Botany*”, “*Zoology*” and “*Mineralogy*”.

In the “*Botany*” page there’s a table containing all the plants available in the museum classified by: “Name”, “Scientific Name”, “Family”, “Color”, “Origin”, “Conservation Status” (in nature), “Museum Section” (in which they can be watched in practice) with the addition of an image of the relative species.

In the “*Zoology*” section there’s a table containing all the animals that can be observed in the museum classified by:

“Name”, “Scientific Name”, “Family”, “Color”, “Habitat”, “Conservation Status” (in nature), “Museum Section” (in which they can be seen in practice) with the addition of an image of the relative species.

In the “*Mineralogy*” webpage there’s a table containing plenty of minerals that can be noticed in the museum classified by:

“Name”, “Category”, “Color”, “Hardness”, “Museum Section” (in which they can be distinguished in practice) with the addition of an image of the relative species.

In “*Info*” section guests can find general information about the visit to the museum. In the upper part of the page there’s an article distinction between the “Basic Visit” and the “Guided Visit”.

The first one has a price distinction specified by age and doesn’t require a reservation, instead the second type costs the same for every independently by the age and a reservation is necessary. In order to make this specification fully understandable there’s an immediate link to the “*Book a visit*” page.

Just under this content there are the three possible ways, characterized by a horizontal page division into three areas: reaching the museum divided by mean of transport (train, plane and car) and the steps to complete in order to arrive at the museum without much mind effort.

The “*Book a visit*” page has been developed to allow the user to reserve a guided visit for the museum choosing the preferred date and timeslot. In the upper part of this section there is the form that must be compiled to make the reservation official.

The main fields are “First Name”, “Family Name”, “Email”, “Number of people”, “Date” and “Timeslot” (the preferred hour in which the user wants to visit the museum).

Just under the form, separated by a horizontal break line, there’s a timetable containing the opening days and hours of the museum. It is specified that guided visits are only available at morning and the afternoon is dedicated to basic visits.

When the form has been sent there’s a response page characterized by a confirmation message or a refuse based on the availability of that day and timeslot, this is just above a big representation of the museum logo.

In “*Contact*” there’s a form composed by these fields: “First Name”, “Family Name”, “Email”, “Subject”, “Message”. This form can be used to contact the museum to receive further information and to ask for cancelling visit reservations.

When the form has been sent there’s a response page characterized by a confirmation message, this is just above a big representation of the museum logo.

All the pages are characterized by a header present at the top characterized by the logo of the museum and the name of the it. Just under the header there’s a dropdown navbar with all the links to the other pages, with the text aligned at the center of the page. In the bottom of the page there’s always a black footer with white text containing the description of copyright rights of the museum.

Technical description of the website

Describe the website implementation in terms of:

- *Folders (directories) and files*: list the folders that are part of the project, describing their content (e.g., “The folder `img` contains all the images of the website, while the folder...”). Clearly specify the name of the Home Page file
- *HTML structure of pages*: indicate whether there are “structures” shared by all or many pages of the website, such as headers (e.g., `<header>` tag), generic containers (`<div>`, `<section>`), navigation sections (`<nav>`), main content (`<article>`), footer (`<footer>`), etc.
- *CSS*: indicate the main selectors for which you have defined styles (shortly describing them), including possible classes or pseudo-classes (e.g., “The style for the `#main` selector defines an absolute positioning and...”; “The generic class `.ital` specifies that...”)
- *JavaScript (and other possible client-side technologies used)*: illustrate the purpose of the employed JavaScript code (as well as of other possible client-side codes), shortly describing it (without necessarily providing all technical details, however); e.g., “The JavaScript function `slideSh()` allows to display a sequence of images which are placed in the folder ...”). Indicate the URL(s) of the possible page(s) from which the JavaScript (or other technology) code has been “copied” and adapted to your own needs
- *Server-side technologies*: indicate the purpose of the employed server-side code (e.g., PHP, ASP, Node.js, etc.), shortly describing it (without necessarily providing all technical details, however); e.g., “The PHP code at the beginning of the `<body>` of the page is used to...”
- *Development tools employed*: HTML/CSS/JavaScript/etc. editors, possible image editing tools (e.g., Gimp, Photoshop, ...), etc.

Specify whether, for the creation of the website, you have “drawn inspiration” from existing websites or templates (indicating their URLs), and whether you have used specific frameworks (such as Bootstrap). Also indicate anything you deem useful to make your implementation choices clear.

This section can be extended to the next page, if necessary—but possibly avoid exceeding 90 lines.

The main folder is composed by all the webpages of the website, in particular by the Home Page called “*index.php*”, and contains some subfolders. The “*css*” subfolder contains the stylesheet of the pages and a subfolder called “*font*” composed by the font used in whole website. The “*javascript*” folder contains the file composed by all the functions used in some pages like “*index.php*”, “*contact.html*” and “*book.php*”.

The “*images*” folder includes all the images used in the website and the “*logos*” folder contains the museum logo and the other logos used for example in “*index.php*” and “*info.html*” to describe the different sections.

All the pages share some structures such as header, navbar, articles and footer.

The **<header>** tag has been used to define the header in the top part of the page, and it has been filled with the museum logo (using **** tag) and a text link that brings to the Home Page.

Nested **<div>** tags have been used to create the dropdown navbar positioned just under the header and containing links, non-clickable buttons and nested links to the different sections in order to make every page accessible wherever the user is.

In the main content of the page many **<article>** tags have been used in order to create different areas, each one concerning different subjects. Inside the **<article>** tag there's a division in columns using the **<div>** tags, this was made to create a column of text just beside an image describing what is the content of the text column.

In some pages, like *"index.php"* and *"info.html"* this was made to give different types of the same subject such as types of collections or possible means of transport to reach the museum.

In the collection pages and for the timetable the **<article>** tag has been used with the **<table>** tag to represent into a table the list of species of the corresponding sections.

In *"contact.html"* and *"book.php"* the **<article>** tag has been used with the **<form>** tag to make possible the reservation for guided visits and contact inserting the user's data into the **<input>** fields.

The **<footer>** tag has been used in the bottom part of all the pages to communicate the copyright policy about the website and the name of museum.

The *"mycss.css"* file provides a height of 80px and a background color obtained using *rgb (196, 164, 132)* similar to burlywood. The logo used in header section has been set with a width of 100px and a height of 50px, then vertical align has been set to middle in order to bring it in the correct position of the header area. This was made using *".headerImage"* class. The link in **<header>** has been set using the *"a.header"* class in order to make the text color equal to *#eee* and realize a black text-shadow to make the link more visible.

The **<div>** navbar part has been set with absolute positioning just under the **<header>** section with a black background color and a center-aligned text using the *".navbar"* class. The link inside the navbar has been put using relative positioning, with an inline-block position, a white text color and removing the text-decoration through the *".navbar a"* class.

The *".subnav-content"* class has been used to make the nested links not visible until the upper button is hovered by mouse using the *".subnav:hover .subnav-content"* class.

The **<article>** section made with *".content"* class and with **<div>** using *".row"*, *".column"* and *".columnArrive"* classes has been positioned just in the center of the page using margin-left and right as automatic.

The *".row"* class has been used with the flex display in order to subdivide the width of the page for every different *".column"* and *".columnArrive"* classes. The first one having a flex of 50% because of two columns

and the second set to 33.3% when there are three columns.

For all the **<article>** tags a box shadow has been used in order to make the content tridimensional respect to the full background of the page set with *“body”* using a background-image with white-green radial-gradient.

The **<footer>** tag has been modified with the *“#footer”* id that puts it into the bottom part of the page with a black background color and white text color.

The **<div>** *“#disclaimer”* id has been used to put the disclaimer into a fixed position in the bottom right part of the Home Page.

“.littleButton” and *“.bigButton”* classes have been used to create the buttons present into the different pages, in order to make the buttons with good background a visibility. These were also added with some little classes (ex. *“.redBackground”...*) to change only the background of each button. *“a: hover”* has been made to make every link darker when the mouse is over it, this permits to the user the distinction of links from normal text.

Javascript has been used for the Home Page where the h1 with *“#changeText”* id, containing the type of museum, continuously change describing the different sections. This was made using the *“changeText ()”* function that activates when the page is opened with *“window.onload”*.

“getToday ()”, *“exists ()”*, *“checkForm ()”*, *“isEmail ()”*, *“checkContactForm ()”* functions have been used for the form section. The first one is used to make the reservation dates available only the day after the current day, so the user can reserve a visit with a minimum forewarning of one day. The second one has been created to evaluate if a certain field is empty or there's at least one character, it is used with *“checkForm ()”* and *“checkContactForm ()”* to establish whether a form can be sent to the server or has to wait that other fields have to be filled. If a field is empty an appropriate alert is printed on screen.

“isEmail ()” has the purpose of checking if in the email field what it is written by the user is really an email or not (checking the presence of *“@”*).

The last function called *“closeDisclaimer ()”* allows to remove the disclaimer in the Home Page when clicking on the *“x”* button in the disclaimer itself, it permits the correct view of the page without the disclaimer.

In "*botany.php*", "*zoology.php*", "*mineralogy.php*" the server-side technologies through PHP have been used to receive the data of the different species from the corresponding tables into a database.

The different rows of the tables are written taking data from mySQL tables and also the images in are saved with that data. So, at first the connection with the database is opened and a "SELECT *" of the corresponding mySQL table is done. Then, after writing the first row of the table, the species data are written into subsequent rows.

In "*book.php*", PHP just before the header section is used to remove all the old reservations (with a date older than the current day) when the page is loaded by anyone. With this trick admin doesn't have to remove the older reservation manually.

In "*reservation.php*", PHP just before the HTML section is used to open the connection to the server and to save the data received from the form of "*book.php*" page. Then there's a control of what timeslot has been chosen by the user, according to that choice there's a check to verify if there are available seats for the visit in the date and timeslot chosen. If there's availability there's another verification, in fact the system must verify if the email chosen by the user has already been used before for another future reservation. If this last control goes well, the reservation is saved into a mySQL table.

According to the result, a message is printed on screen and the user knows how the process has gone.

In "*contactReservation.php*", PHP just before the HTML section is used to open the connection to the server and to save the data received from the form of "*contact.html*" page. Then these data are sent to the server and the contact reservation is saved into a mySQL table. A message is printed on screen and the user knows how the process has gone.

Visual Studio Code has been used for writing the HTML, CSS and PHP code. The images have been taken from the internet all with Creative Commons license, in order to be fully published on web. Many image editing tools have been used to reduce the size and the quality of images such as GIMP and Photopea, this was to decrease the time of loading for the different pages. All the HTML and CSS code has been checked using the W3C validator in order not to use deprecated tags or useless attributes.