Jacobs tasty website!

Internet Applications, ID1354

Jacob Kimblad jacobki@kth.se

2018-11-05

1 Introduction

The first requirement for the tasty recipes web site is that it should constist of four different pages. An index page, a calendar page, a meatballs recipe page and a pancakes recipe page. The second requirement is that from each of the four web pages you should be able to navigate to any of the other three web pages directly. The third requirement is that they should all have a similar layout which should be put into consideration when choosing fonts, colors, mouse hoovering, and link behaviour. In addition to all of the pages having similar layout some of their properites may not be assigned their default values. These properites are page layout, font family, font size, font style, foreground and background color, mouse hovering and link behavior. Beneath follows the requirements for each of the four individual pages.

1.1 Index page

The index page should be informative and welcoming by promoting the calendar page and providing a link to it in text-format.

1.2 Recipe pages

There should be an individual page for each of the two recipes (pancakes and meatballs). The recipe page should contain a picture of the prepared dish, its ingredients, instructions for preparing, its name and hard coded user comments.

1.3 Calendar page

The calendar page should, as the name suggests, present the user with a calendar of the month. Two days in the calendar should each contain a unique picture of the recipes (meatballs and pancakes). These should be clickable links that redirects the user to their respective recipe page.

1.4 Tasks

The tasks to be followed which should meet the requirements begin with creating HTML and CSS files to implement each of the different pages of the tasty recipes web site. It is necessary that all of the HTML and CSS files pass the W3C validation. It is also necessary to use a reset style sheet. When designing the website it is necessary to follow five of the ten heuristics for user interface design which were discussed during a lecture. All of the pages must also have identical behavior in the following four browsers: Chrome, Explorer (or Edge), Firefox and Safari.

1.5 Summary

The following bulle list summarises the requirements and functionality of the complete web page.

- Design and functionality:
  1. Contain index, calendar and recipe pages.
  2. Navigate to any page directly from any other page.
  3. Pages should use similar layout.
  4. Properties that must not have default values.
- Index page:
  1. Informing and welcoming.
  2. Contains a link to the calendar page.
- Recipe pages:
  1. One page for each recipe.
  2. Contain the name of the dish.
  3. Contain a picture of the prepared dish.
  4. Contain a list of required ingredients.
  5. Contain steps of how to prepare the dish.
  6. Contain comments from users about the dish.
- Calendar page:
  1. Present a calendar of one month.
  2. Present a picture of each recipe in two unique days.
  3. Make the pictures of the recipes clickable links to their respective page.
- Tasks:
  1. Implement the described pages using CSS and HTML.
  2. All CSS and HTML files should pass W3C validations.

3. Use a CSS reset sheet.
4. Follow the five given heuristics for user interface design.
5. All pages must have identical behaviour in the four common browsers.

2 Literature Study

For basic CSS and HTML syntax and tutorials reference *W3Schools Online Web Tutorials* [1] provides good overview of possible syntax for both languages. It also provides easy to understand examples and the limits and possibilities of different tags in HTML and selectors, properties and values in CSS. It also provides templates for websites using HTML and CSS which is a great resource to gather inspiration and examples of actual websites.

It is predicted that the hardest part to complete will be to get the formating right for the calendar using HTML and CSS. The stackoverflow post shown in *html - What is this I hear about the table tag being deprecated?* [2] explains how the HTML table tag should not be used to format layout of the page. It should only be used for data that is very tabular in nature. When looking at the reference document for the table tag *HTML table tag* [3] we see that a lot of table attributes have been deprecated in HTML5. Reference *Using ¡div¿ Tags for Layout* [4] explains how using tables for layout is now offcially wrong and instead div elements should be used. To help with constructing a layout for the calendar a div table generator such as *Div Based Table Generator* [5] can be used.

3 Method

To solve the tasks a set of development tools are selected either becuase the authors familiarity with them or because of their simplicity to use, both of which helps to speed up development time. The text editor used to edit the HTML and CSS files is Vim, described as "Vim is a highly configurable text editor for efficiently creating and changing any kind of text" *welcome home : vim online* [6]. The web server used is the http-server *http-server* [7] acquired under the NodeJS infrastructure. The shell used to run both Vim and the http-server is Zsh *ZSH - THE Z SHELL* [8] under a GNU/Linux operating system distribution known as Manjaro *Manjaro Linux* [9]. For source control Git was used with Github hosting a remote repository, this made it such that the validation tools, described in the next paragraph, could be used by just provdiging them with the URL's to the raw files hosted on Github.

Testing tools are also used to check some of the requirements of the web site are met. To make sure that the HTML files passes a W3C validation the online testing tool found in *The W3C Markup Validation Service* [10] is used. A similar testing tool is also used for the CSS files such that they also pass a W3C validation, this testing tool can be found at *The W3C CSS Validation Service* [11].

To help with debugging the code the web browser Firefox is used as it also includes a very handy set of tools, known as the Firefox Developer Tools *Firefox Developer Tools* [12], that can be used to change the source code and see the effects in real time all within the browser. This helps with speeding up development time, as there is no need to navigate around different tools to make and the see the effects of a simple change. It is also necessary to test the finalised code such that it gives the exact desired behaviour in all of the different browsers defined by the requirements. To do this another online tool found at **online browser tester** was used.

4 Result

To make sure that every other page can be reached from any of the four pages a navigation bar at the top of each page is used. This navigation bar has the same look and layout for all four pages to fulfill the requirement of similar layout on all the pages and can be seen in figure 1.

Figure 1: A sample user interface screenshot to illustrate the navigation bar at the top of each page.

This section explains *the result* of what you did.

Present the solution. Explain your code and prove that it meets the requirements. It is very important to *state each requirement that is met* and explain *how you met it*. It is also important to include links to your code in your Git repository, user interface screenshots, see Figure **fig:ui**, and other figures to illustrate your reasoning. Also remember that these figures must be referenced in the text.

5 Discussion

This section *analysis* the result presented in the previous section.

Summarize the requirements and *clearly state which of them you have met*. What lessons have you learned and what problems did you face? How were the problems solved? Should you have done something differently?

6 Comments About the Course

Any comment(s) related to this course offering or to coming offerings is much appreciated. *Please also tell approximately how much time you spent on the assignment*, including lectures and exercises. This is of great help for course evaluation.

References

[1] *W3schools Online Web Tutorials*. [Online]. Available: https://www.w3schools.com/ (visited on 11/05/2018).

[2] *Html - What is this I hear about the table tag being deprecated?* [Online]. Available: https://stackoverflow.com/questions/3176896/what-is-this-i-hear-about-the-table-tag-being-deprecated (visited on 11/05/2018).

[3] *HTML table tag*. [Online]. Available: https://www.w3schools.com/tags/tag_table.asp (visited on 11/05/2018).

[4] *Using ¡div¿ Tags for Layout*. [Online]. Available: https://www.beginnersguidetohtml.com/guides/css/layout/div-tags (visited on 11/05/2018).

[5] *Div Based Table Generator*. [Online]. Available: http://shabirhakim.net/Tools/divbasedtablegenerator.aspx (visited on 11/05/2018).

[6] *Welcome home : Vim online*. [Online]. Available: https://www.vim.org/ (visited on 11/05/2018).

[7] *Http-server*. [Online]. Available: https://www.npmjs.com/package/http-server (visited on 11/05/2018).

[8] *ZSH - THE Z SHELL*. [Online]. Available: http://zsh.sourceforge.net/ (visited on 11/05/2018).

[9] *Manjaro Linux*. [Online]. Available: https://manjaro.org/ (visited on 11/05/2018).

[10] *The W3c Markup Validation Service*. [Online]. Available: http://validator.w3.org/ (visited on 11/05/2018).

[11] *The W3c CSS Validation Service*. [Online]. Available: http://jigsaw.w3.org/css-validator/ (visited on 11/05/2018).

[12] *Firefox Developer Tools*, en-US. [Online]. Available: https://developer.mozilla.org/en-US/docs/Tools (visited on 11/05/2018).