Design Document

A technical tour to UPhotos

Nick Broadbent and Weiran Song

Spring 2015
CS50
May 11, 2015

Table of Contents

Broadbent and Song

Spring 2015

Design Document

A technical tour to UPhotos

Bootstrap and HTML Templates

Suspendisse potenti.

We choose to use Bootstrap and its template to create the template files. (Link provide at the References section) At first we were planning to the MVC framework that we use in Pset7. By reaching to friends for advice, Bootstrap is decided to be used because first, it is easy to learn and start. So it can take less time for us to build a website and offer good quality, and it is responsive! Second, many basic HTML elements are already included and has been stylized, and JavaScript libraries are already included. In addition, website created using Bootstrap will work well with all Chrome, Firefox, IE, and it mobile friendly. Moreover, source code can be easy to find and we can easily get rid of the parts we don't need and transplant needed source code in, so we can customize out own website (Information can be found at http://www.sitepoint.com/11-reasons-to-use-twitter-bootstrap)

So after searching online and looking for proper templates for our project, we decided to use the Business casual and 3 Col Portfolio templates for our project (links can be found at Reference section). To create the basic framework for UPhotos, we took a look at how the templates actually works and how those websites looks like. Then, we based on the Business Casual template and keep those elements we needed such as the slide show on home page(use the origin photos included in the template) , box element to wrap the text, photos, and placeholders. Then delete those elements we don't need such as page number. Then choose elements from the 3 Col Portfolio template to create a sub-website to display photos uploaded by users. An important of our website allows users to up vote or down vote for the photos they like or dislike, thus a vote button need to be create for each photo. After trying different vote button design, we choose the one we currently use from jQuery2dotnet (link can be found at Reference section) and add button for each photos. We also used Pset7

postmortems to create login and register sub-webpage (username, password, and password confirmation). Then, edit content for each part of the webpage. For instance, a brief introduction of the website and how it works. Last step is make sure that all sub-webpages are connected correctly, and make sure there pages are all using the same style including headers and footers to make the website looks constant.

Database and PHP

We made use of a database in order to store account information, such are usernames and passwords, and information for photos which should be used in order to display the information on the website. So, we used the pset7 postmortem login and register pages in order to guide us in developing such pages for our own website. We also made use of functions.php and congif.php in order to use functions such as render and be able to query the database easily, making the coding a lot nicer looking and easier. We decided to make the majority of the files PHP in order to take advantage of all of the functions that PHP has to offer. As well as it being a very popular language, any problems that we've encountered using PHP would have occurred before to someone else, so it would be very easy to find the reason for the problem online, as well as a solution.

After, making the register, login pages work with the database, we used logout.php from pset7 to log users out. And then we worked on the upload.php file, which allows users to upload images to the server. We decided to store images in the server, so that there wouldn't require as much data to be transferred from the database and the server, making the website faster and more efficient. In order to create the upload file, we did some research online on how to get the users file locally, then transfer it to the server, then move it into a permanent folder. We found php.net, w3schools and other forums to be useful for completing this.

Creating the photos.php, which displays the photos, was done by getting some photo information from the database and using this information to access the photos location on the server and display the title and user who uploaded it alongside the photo. We used a php for loop and decided to display 9 pictures every time. Looping over some html tags using the bootstrap outline to format the photos nicely.

We used the database to make two tables, to store user data, such as username, password, id and number of votes, and to store photo info, such as title, filename, votes and owner. So, we stored the entire image data on the database so that we could find the image stored on the server when necessary.

References

Business Casual HTML Template and pictures from Bootstrap:
http://startbootstrap.com/template-overviews/business-casual/

3 Col ortfolio HTML Template and pictures from Bootstrap:

http://startbootstrap.com/template-overviews/3-col-portfolio/

Vote button: http://www.jquery2dotnet.com/2013/11/vote-up-and-down-script-using-bootstrap.html

File upload:

http://php.net/manual/en/features.file-upload.post-method.php

http://www.w3schools.com/php/php_file_upload.asp

MVC framework: PSet 7 postmortem