Responsive Design Lab

Overview: In this lab, you will create a small (3 page) web site that uses "single page design" and "roll your own" responsive design. You will learn how to create a HTML form to gather user input (and send it to a server side program). Your form will use some HTML5 validation (client side validation). You will learn how to use a bit of jQuery functionality (to implement a collapsible navigation bar). While Bootstrap is a great responsive design framework, we will not use any Bootstrap for this lab. Instead, you will learn the basics of what would be involved to write a framework like Bootstrap.

Definitions:

- "Single page design" is where you have several virtual pages inside of a single physical page. The user can scroll up/down to go from virtual page to virtual page, or they can click in the fixed nav bar to navigate (internal links) to the virtual pages.
- "Responsive design" is when web pages adjust according to the width of the viewing device, e.g., one layout for desktop versus another layout for mobile phone. You test this by narrowing and widening your browser to see how the page elements realign themselves.
- HTML form tags have been around for a long time. Inside a form attribute you have one or more <input> tags plus a submit button. When the user clicks the submit button, all of the user's input values (that are in the same form as the submit button) are sent to the page that is specified by the form's "action" attribute. HTML5 (released not too long ago) introduced some new client side validation features.
- jQuery is an extensive library of JavaScript code. Its two main benefits are (1) you write less code than if you used just straight JavaScript, and (2) jQuery works well regardless of which browser (e.g., chrome vs. firefox vs. IE; old versions vs. new versions). jQuery tests which browser is being used and provides "graceful degradation of functionality" for older browser versions that do not support the "latest and greatest" features.

Lab Requirements:

1. **Single Page Design.** You shall create a single file "02_advLayout/index.html". This page will act like a small web site containing three virtual pages within the single physical page, using "single page design".
   - You may choose any topic for your "small web site" for this week, but it would be best if your topic is the same as the topic you selected for your web application for the whole semester.
2. **Style Sheet.** Your single page shall reference an internal style sheet (and no external style sheet). Come up with a totally new (and more exiting/professional) layout than what you had in your last lab.
3. **First Virtual Page - Landing Page.** The first virtual page (within "02_advLayout/index.html") shall be a "landing page" (like a home page for your small web site) that provides the user with an idea of what your "small web site" is about.
4. **Second Virtual Page - Informational Page.** The second virtual page (within "advLayout/index.html") shall be an informational page with at least 3 areas within it. Each of these areas shall have a title, an image, and some explanatory text. In desktop view, this virtual page shall display multiple columns side by side. In mobile view each area should be full width. Images should be styled with percentage width so that they are responsive images.
5. **Third Virtual Page - Form.** The third virtual page shall be a form with a submit button. In desktop view, the form shall have two columns, but in mobile view the form shall have just one column. Note: to make this form look OK, you should probably have your prompt on one line, followed by the input, followed by a blank line (and repeat). There is not enough width on a cell phone to have a prompt and input on the same line.
   - The form tag shall contain at least one of each of the following HTML input types: text box, text area, checkbox, select list, radio group, and number (new in HTML5) and a submit button. Each of these input elements shall be “well named” so that you get a descriptive email when the user submits their input.
   - At least two of the input tags shall be required using the “required” attribute (or have some other type of HTML5 validation). If any input does not pass validation (and the user clicks submit), your page shall pop up a HTML5 validation bubble error message.
   - When the user clicks submit button (and if all the HTML5 validation conditions have been met), the user’s input shall be sent to your Temple email address using the Temple mail program (see appendix).

6. **JQuery.** "02_advLayout/index.html" shall have some animation implemented with jQuery that uses the animate effect that was provided in the sample code.

7. **Fixed Title Nav.** "02_advLayout/index.html" shall have a fixed Title/Nav bar which is always visible. Remember that any fixed element has the potential to overlap with other elements. Apply a higher z-index to your title nav so that it is top of other elements. When you link to each of your virtual pages, be sure that you "land" correctly at the top of that virtual page (not too high, not too low) – check this in desktop width as well as mobile.

8. The **navigation bar** shall have the following links:
   - `<a href="#home">Home</a>` (landing page described above, internal link, not your main home page)
   - `<a href="#info">Info</a>` (the informational page described above, internal link)
   - `<a href="#form">Form</a>` (the form described above, internal link)
   - `<a href="../index.html">Back</a>` (gets you back to your main home page that is stored up one folder)

9. **Footer.** "02_advLayout/index.html" shall have a footer with your name in it (can be fixed - always at the bottom - or visible only at the end of your last virtual page, as you wish).

10. **Responsive Design.** All three virtual pages shall employ responsive design, that is to say, all elements and components shall re-order and/or resize nicely according to the browser’s screen size. Your grade will heavily depend on how well your webpage responds/looks at different screen sizes. Check the appendix to see an example that shows responsive design.

11. **Media Queries.** Your style shall have been designed "mobile first", which means most of your design rules (that are outside any media query) apply in mobile view (mostly everything in single column layout). This should be followed by a "desktop only" media query where you apply the minimal number of styles that are needed to make it look good in desktop width (many things in multi-column layout). You might then need a second media query for mobile only, but try to avoid this, if at all possible.

12. **Collapsible Nav Bar.** The navigation bar shall be collapsible when in mobile view (using jQuery) and always visible in desktop view.

13. **No Angular.** This lab shall not use Angular for code reuse (e.g., of title/nav or footer).

14. **Blog.** For this lab (and all labs), your labs page shall include a blog that describes and links to the work you did in the lab. In the blog, describe your experience doing the responsive design lab. Which aspects were easy? Which were hard? What were the most important things you learned in this lab?

15. "**For All Labs**". All code for this lab shall follow the requirements listed at the top of the 3344 labs page in a section entitled "Requirements for All Labs" (mostly just "good programming and design practices").
Submission:

1. After completing all the requirements, test locally (and syntax check – right click and View Source from Firefox), then publish and test what you published.
2. Then submit into blackboard a zip file of your whole web site (meaning all the work you have done in all the labs so far). Make sure to include all the necessary files and folders in your zip file.

Grading:

1. We will visit your labs page from your published web site. We will check that your blog for this week includes what was asked (what was easy? hard? important?).
2. We will link from your blog to the landing area of the single page “web site” that you created for this week's lab. In desktop width browser
   a. We will scroll down top to bottom, checking if all three virtual pages look professional. Does the first virtual page (landing area) entice users to visit your Single Page Designed (3 page) “web site”? Is there a clear visual division between each virtual page? Is each virtual page always at least as tall as the height of the browser? Do all three virtual pages have a consistent message and do they look like they could belong to a web site that people would like to visit?
   b. We’ll check that all links in the nav bar work (with smooth scrolling) and that the nav bar is always visible. Clicking on each link, we’ll check that the virtual page lays out nicely (heading not too high, not too low, etc).
   c. We'll check that the informational virtual page has multiple columns, each with picture, label and information.
   d. We'll check that the form lays out in two columns and has the required input elements (text box, text area, check box, select list, radio group, number). We’ll click submit without entering anything and see if we get a HTML5 validation bubble error message (at least two fields were to have HTML5 validation). After filling in all fields and clicking submit, we'll check the page that says an email that was sent to you. From this page, we'll check that you named your user input tags well.
3. We will slowly narrow the browser until your media query changes your layout to mobile view.
   a. We'll check that all elements on all three pages adjusted nicely (typically all the areas "one up", some fonts/padding/margins changed).
   b. We'll check that the nav bar collapses (desktop nav bar changes to hamburger menu) and opens/closes when clicked (without causing the content below the nav bar to realign).
   c. We will check that all three internal links (from the mobile nav bar) work and correctly position to the right spot at the top of each virtual page.
4. We will view source to check for no syntax errors, good naming, code indentation, some comments. Basically we are looking for neat, organized code with nothing that's unnecessary.
5. We will test your back link to make sure it navigates back to your main index page (one folder up).
Suggested Approach:

1. Decide the **topic** of your single page web site for this week’s lab. Try to align it with your overall web site theme.

2. Find several **images** that are related to the topic. If the size of any image file is more than 500K or so, use the picture editing techniques (from the "Web Design" tutorial page) to make it smaller (or pick a smaller image to start with – google images allows you to filter by file size). Find or create a paragraph of text for each image and come up with a label for each image.

3. There are several sample pages referenced by this lab – single page design, responsive design, jquery (collapsible nav bar), and forms. Your job is to **combine the code from all three of these examples**. In the lab activity (in case you didn’t do it), I had students start out with a copy of single-page-design.html, copy the responsive design (London – Paris – Tokyo) into the second virtual page, copy the forms example into the third virtual page.

4. **Design a form** that makes sense for your small web site's purpose. Use all the different input types that were required. Name each one well. Make sure that the action attribute of the form tag references your own temple email address. Test the form to see if it sends you an email. To make the form work well in mobile layout, place the input element on a new line after the prompt and then use a blank line as separation – there is not enough width to place the prompt and input element on the same line. Then group your input elements into two divs that you can later style side by side (but only in desktop).

5. **Using "Mobile First" design approach**, work in narrow browser width and enter your CSS style rules (inside of the <style> tag in the header) as you did for last lab (no media query). Create your landing page and your informational page (3 images each with label/title and explanatory paragraph). Each page needs to be visually separated from the other. This can be done by alternating page colors or adding borders etc.

6. Once the layout looks good in mobile width, add a media query and put in the minimal style rules to make it look good in **desktop width**: 3 columns for the informational page and 2 columns for the form, make sure the nav bar still looks good.

7. Make sure that the **fixed nav bar** is always visible and that each link gets you to the right spot on the page. You might have to add different anchor tags and reference these from the nav bar.

8. **Add in the jquery code to make your nav bar collapse** (using the jquery example)

9. Remember that throughout the semester, you need to have a **single project that includes all of your homeworks**. If you have not already done so, place your single page design “web site” in 02_advLayout/index.html. Test linking from your blog down to the single page website. Test the back button from there making sure you can get back to your main home page.

10. **Test** everything as listed in the grading section above.

11. **Publish** and test what you published.
Sample Deductions

• -9 if no zip file submitted into Canvas (by the due date).
• -9 if no pages published by the due date.
• -2 if blog missing.
  • -9 If the blog is missing and you have not named your homework exactly as specified, we will not be able to find your work and the grade will be the same as if you didn’t publish.
  • Up to -2 if blog shows no effort.
• Up to -1 if landing area shows lack of effort.
• -0.5 if your name is missing from the footer.
• -1 if images load slowly – you are supposed to use the easy image editing techniques under my web site “tutorials – front end – web design” (section entitled “working with images”).
• Up to -3 for missing or non functioning Nav Bar links (home to the landing area for this week’s work, info, form, labs back to your blog page).
• Up to -3 for informational page lacking elements (need 3 sections, each with title, image, and text – the sections should show in 3 columns in desktop width, single column in mobile).
• -2 if form does not work when you submit it.
  • -2 if form does not have 2 column layout in desktop and single column layout in mobile.
• Up to -5 for lack of professionalism.
  • Your page must look good in desktop as well as narrow/mobile viewport width.
  • Is the layout fluid (wraps nicely at all viewport widths).
  • Do you utilize "white space" effectively to emphasize important aspects of the page?
  • Have you used margins and/or padding to keep text away from all visible edges?
  • Is all text legible (including link text before clicking, after clicking, and while hovering)? Is it large enough to be easily read? Does it have enough contrast with its background?
  • Is your nav bar always visible/usable?
  • Is your layout substantially different from sample layouts that were provided (and from your classmates)?
• Up to -3 for lack of originality.
  • Your code may not be overly similar to sample code that is provided, nor overly similar to any of your classmates.
  • Your code may not be copied from the internet.
• When we “View Source”:
  • Up to -2 for HTML/CSS syntax errors which would show in red font from Firefox View Source.
  • Up to -2 for poor coding style. Your code should be neat, properly indented, well named, and with no unnecessary code.
Appendix: Example Design

**Desktop View**

This website is for a mexican restaurant. The site is going to provide information to customers as well as employees. The customers will have access to menus and general information. Employees will be able to maintain a work schedule as well as inventory and order records.

**Chef Info**

**Alain Ducasse**
French-born chef Laurent Orsola uses the cuisine of his homeland to inspire the dishes he prepares in his latest restaurant, L’O. After training at the Côte d’Azur, France, Orsola spent time in Europe’s top kitchens before coming to the United States.

**Nancy Silverton**
One of the country’s top chefs pays her respects to a simple ingredient: bread. Tharren, W. A. F. Focus on the simple. “We do bread,” Silverton says, “almost as much like the bakers she bakes.

**Charlie Palmer**
Charlie Palmer has come a long way from the dining halls and fast-foods of Schenectady, NY, where he grew up. Now a culinary success, Palmer owns more than 10 restaurants, a hotel, and a wine retail chain.

**A form so we can better serve you.**

- What is your Name? (required)
- What is your Email? (required)
- What is your favorite menu item?
- Do you prefer Lunch or Dinner?

Do you like coupons? [ ]
How many people do you like to eat with? [ ]

**Comments or Suggestions?**
Type your comments here.

[Click Here to send email]

**Mobile View**