CISC 3650, Spring 2018, Assignment #1

Prof. Kletenik

Website Project

The goal of this project is to create a website that collects user information for some purpose. That purpose is up to *you.* For example:

- A website to allows college students to upload information about themselves with the goal of creating an online collection of resumes
- A website that allows users to upload favorite recipes
- A website that collects voter information for political polls
- A website that collects information about applicants for auto insurance
- etc.

Your goal is to create a well-designed user interface using *client-side* programming. You do not need to store any information on the server – i.e., you may ask the user for their username and password, but you don't need to store that information anywhere and will never need to retrieve it.

You *are* asked to use some client-side session storage. This is not something that you'd ever use for long-term storage, or for anything that you'd need to access on your own.

The emphasis of this project is on designing the interface in an intuitive and clear way that maximises the user experience. In Section 2, you are given a list of client-side programming functionalities that must be worked into your site. (Yes, you will need to be creative to work it all in!) Read through the section before beginning to design so that you know what you will need to incorporate. However the emphasis should be on the user design.

1 Design Process

1. Write a brief synopsis of what your website is intended to do. Is it targeted at a specific group of users? What are the demographics of that group?

2. Refer to Schneiderman's Eight Golden Rules as well as the principles for interface design in the reading provided. Which of them will you incorporate and how? (You will need to defend these choices in Section 3.)

3. Create several (at least three) sketches / wireframes / mockups for the design of your UI. (Note: we are using the three terms interchangeably and the representations can be as high- or as low-fidelity as you think makese sense, within reason.) Some things to consider as you sketch your designs (adapted from `http://www.columbia.edu/itc/visualarts/r4110/f2000/week04/04_06_Assignment_3.pdf`):

   (a) What is the grid and layout system you're using to organize the information on screen?

   (b) Are the buttons and other elements consistently placed on screen? Is their placement reasonably intuitive to the average user? Could/should you use an icon instead of/in addition to text?

   (c) Which form fields should be required fields? Do any form fields have constraints on the contents (e.g. only alphanumeric text)? Why?

   (d) Are you presenting the content in a clear, accessible way? Are you presenting too much material at once? Not enough?

   (e) When giving feedback to the user, what is the best way to convey information? Color? Text? Pictures? What should happen after the user submits the page?

   Your designs should include:

(a) Placement and basic appearance of buttons and other controls, and description of rollover and "click" behavior,

(b) Rough text and image placements and styles,

(c) May include mention of colors/patterns of text, background, etc.

(d) May include audio components (prompts, background music, interface sounds)

At least *one* of your sketches must be created in Balsamiq (`https://balsamiq.com/`, choose the 30 day free trial); the others may be drawn by hand. As you create your sketches, evaluate which ones seem to be best and why. Note: although not required for this submission, it would be an excellent idea to have other people (that fit the profile of your targeted users) evaluate your sketches and give their opinion.

When you have chosen a design that you think is a good UI design, proceed to develop it using HTML, CSS and Javascript.

2 Programming Component

Your website must contain at least *three* webpages (a main page and at least two content pages).

You must use the following (some design questions to think about are also suggested):

1. Use of CSS with HTML to make your page look good. (*2 points.*)

   (a) use multiple `<div>` tags to divide up your page into sections

   (b) use the `position` property to place each `<div>` in a distinct location on your page

   (c) set the *color* properties of each `<div>` (background and text color)

   (d) set the *font* properties of each `<div>` (and make sure each `<div>` contains some text)

   (e) set the *border* properties of each `<div>`

   (f) set the *margin* and *padding* properties of each `<div>`

   (g) create a `list` and use CSS to modify 2 or more default properties of the list

   (h) create a `table` and use CSS to modify 2 or more default properties of the table

2. Use Javascript to process form elements and generate some sort of functionality for the website. (*3 points.*)

   (a) use a `<button>` to change the text in one of the `<div>`s that you defined above

   (b) use a `<button>` to change the background color of one of the `<div>`s that you defined above

   (c) create a series of fields that auto-tab between them after X characters are entered (for example, a social security number field that auto-tabs after 3 characters, then 2, then 4)

   (d) create two `password` elements and ask the user to enter the same password twice; use a JavaScript function to verify that the user typed the same password in both fields. Use `valid` and `invalid` css pseudoclasses. To think about: what is the best way to provide textual user feedback?

   (e) create at least one other field that validates user input (e.g. a username field that can only include alphanumeric characters). Convey user feedback as appropriate.

   (f) use `<input type="text">` to create a single-line text field; use a JavaScript function to convert the text that the user enters to all upper-case

   (g) create a `submit` button that should be disabled if the user has not fulfilled the requirements (e.g. required text, valid form elements).

3. Use of HTML and HTML5 forms (*2 points*)

   (a) use `<select>` and `<option>` to create a drop-down list from which the user can pick one item
   (b) use `<select multiple>` and `<option>` to create a drop-down list from which the user can select more than one item
   (c) use a `<input type="date">` tag to request and read a date
   (d) use a `<input type="email">` tag to request and read the user's email address
   (e) use a `<input type="tel">` tag to request and read the user's telephone number
   (f) use a `<input type="number">` tag to request and read a numeric value from the user
   (g) use a `<input type="range">` tag to request and read a range of numeric values from the user
   (h) use the `required` HTML tag on at least one field
   (i) use a `<input type="textarea">` and the `<input type="maxlength">` to limit the number of characters that the user can enter. Think about user feedback!

4. Use of advanced HTML5/HTML (*3 points.*)

   (a) Use of geolocation capabilities Use the HTML5 Geolocation feature to get the user's latitude and longitude.
   (b) Use of visual choices Use the HTML5 Drag and Drop feature to let the user select one from a set of images.
   (c) Use of **either** a video or audio file in your site.
   (d) Use of a Google Map to your site.
   (e) Use of HTML web storage to your site with the use of the `window.sessionStorage` object.

3 Written Component: Website Design

(*5 points.*)

This is essentially the writeup of your design process in Section 1.

1. Write a brief synopsis of what your website is intended to do. Is it targeted at a specific group of users? What are the demographics of that group?

2. Include pictures of your wireframes (there must be at least 3, at least one of which must have been created in Balsamiq). Justify your final design choice: why did you make the design decisions that you did? You do not need to write an answer to every question asked in Section 1 but you must provide a detailed, well-reasoned justification. Some things to include: how did you decide on colors? placement of items? user feedback? directions?

3. Read Chapter 7 in the reading assigned. Refer to the list of Schneidermans Eight Golden Rules for interface design (covered in lecture 3; refer to the lecture slides as well). Choose *four* of them and explain how your website adheres to those guidelines.

4. Section 7.1 in the reading refers to principles of interface design. There are three main categories: learnability, flexibility, robustness. Choose *one* principle from each category (e.g. for learnability, you can choose predictability, synthesizability, familiarity, generalizability, or consistency) and explain how your website adheres to that principle. You may have some overlap with your answer to the previous question.

Submission

Zip your website files (html, css, javascript) into a single zipped folder. Save the written component (including pictures of wireframes) as a PDF. Submit both the zipped folder and the PDF through this Blackboard link. Note: do NOT zip the PDF together with the website files!

This assignment is due on Tuesday March 5 at 11:59 PM.