This assignment is to be completed individually. No group work is allowed.

This assignment provides the first step toward developing a dynamic web application by creating a set of static HTML5 pages for several core parts of the application flow. These pages serve as a look-and-feel design prototype for later functional / dynamic application development. This assignment is intended to (re)familiarize students with HTML development and navigation, as well as application design, deployment requirements, and assignment submission in the context of the course.

Course Standards for Web Page Design

This assignment introduces the fundamental standards for web page design in the course. In this and all subsequent assignments, the following design standards apply.

1. Pages must be valid HTML5 standard.
2. You must not use site-builder or automatic site generation tools. You must code your HTML directly, by hand.
3. All pages must validate cleanly as HTML5.
4. All pages must render sensibly cross-browser.
5. All pages must have full and complete separation of CSS into separate files – only the link(s) to load the CSS should be present in the HTML document. No style definitions within the page.
6. All pages must have full and complete separation of JavaScript into separate files – only the link(s) to load should be present in the HTML document. No scripting within the
7. All resources (CSS, JavaScript, Media) must be locally stored with the web application (e.g., local copy of JavaScript libraries).
8. Each assignment will consider coding efficiency and design choices, which includes coding elegance or style. Efficiency is required to make the best use of resources (many full pages are viewed on mobile devices). Good code design is required for maintenance, understanding, reusability, and extensibility of applications. Proper indentation for nested elements is just one example. Course staff are the final arbiters on efficiency and design choices in this context.
9. Page design choices must be sensible for use by a broad audience. Course staff are the final arbiters on sensibility in this context. Ridiculous and laughable branding / inventory may be fine. Ridiculous and laughable site design / implementation is not.

Even if not explicitly or specifically mentioned in assignment requirements, points may be deducted on an assignment for poor efficiency / (code or user interface) design practices in submissions.

Assignment Description:

In this assignment you will develop four HTML pages, according to the following specifications.

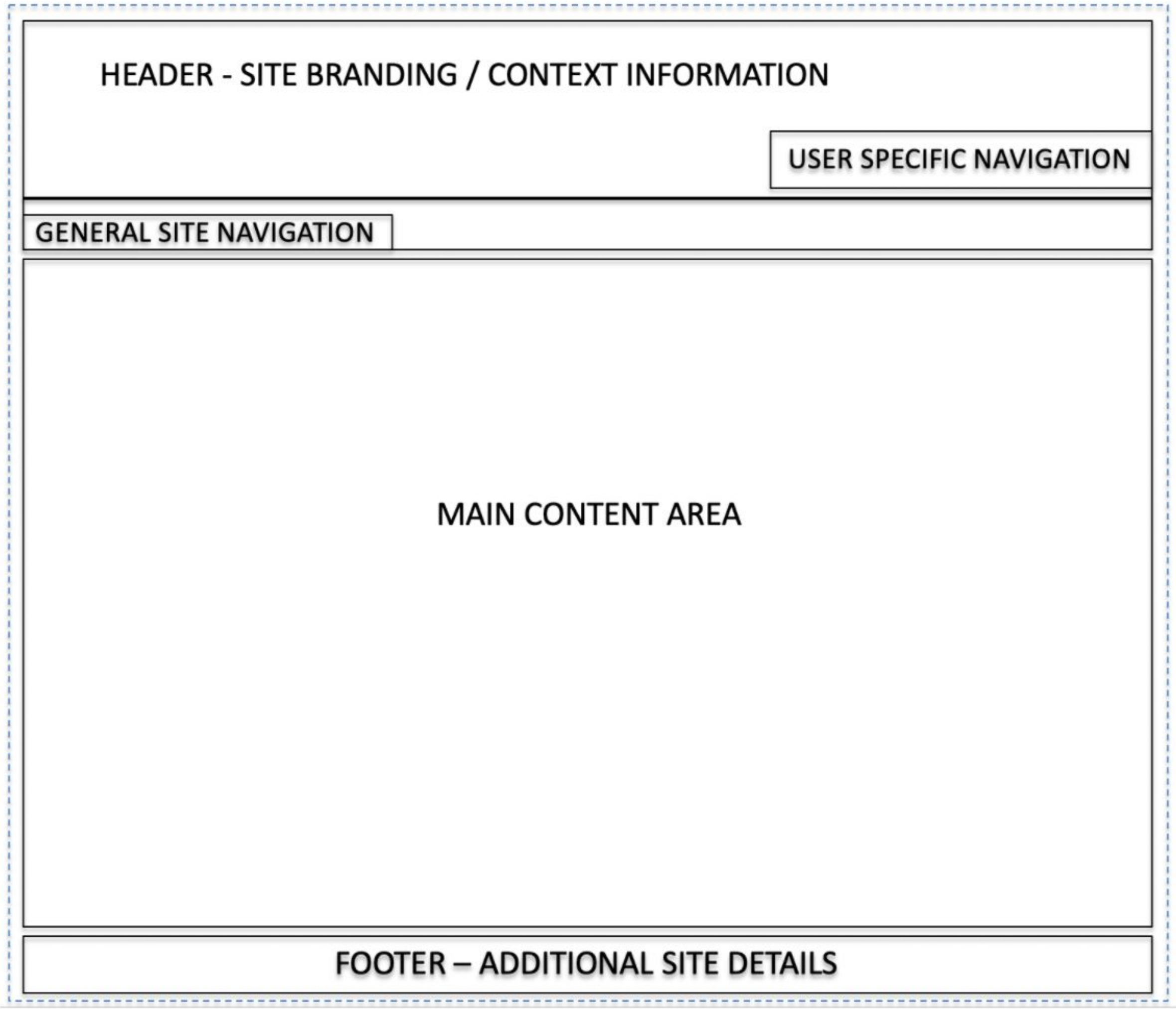
1. Choose an event / meetup domain and categories of interest to you. You must develop your own unique site with branding, categories, and events representative of that domain. Creativity is highly encouraged.
2. You are not allowed to select previously created branding / content. You must create your own. In particular, you must not use any branding / content from (1) the assignment itself; (2) the textbook; or (3) any other students in any of the sections of the course.
3. Each of the pages specified must be a standalone, individual HTML page. And not, for example, framed or one single page with main content section visibility switching.
4. Each of the pages must be standalone, but must have a common design to preserve the look-and-feel of the site. So, some of the common look-and-feel elements will be repeated across pages. Common look-and-feel elements must render exactly the same, so there are seamless transitions with no jitter between pages for common elements.
5. You must conform to the required overall page layout structure.
6. Within each of the sections of the overall page layout structure, you may style the content differently than the representative examples (color scheme, alignment, etc.), as long as the illustrated types of content are present and represented in a reasonable and understandable way.
7. All of the content must be represented using HTML directly (no JavaScript DOM generation).
8. For this assignment, we will consider the following specific design standard requirements:
    1. Validation of HTML5 by the following tool: http://html5.validator.nu/

    2. Cross-browser checking for Firefox and Chrome

Required Page Structure

All pages must use the following foundational layout structure, including:

- Full-width containers of sensible height for (1) page header area, (2) area for navigation elements, (3) page footer area.
- Full-width container for the main content area.

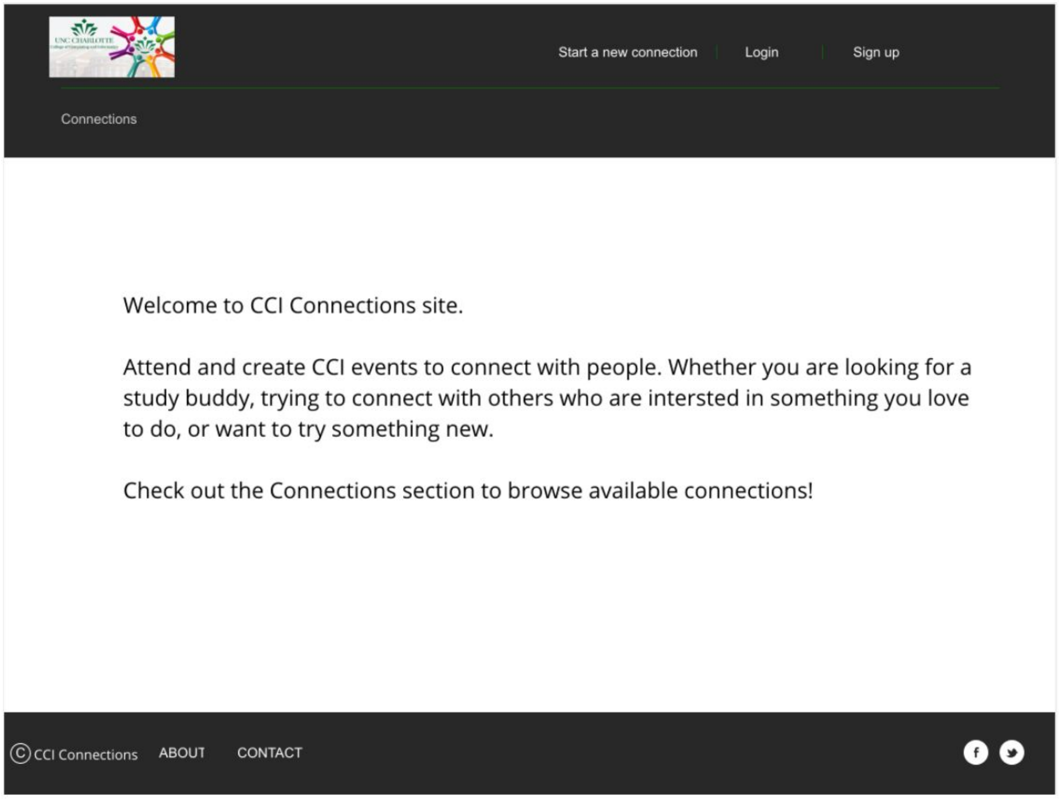

HEADER - SITE BRANDING / CONTEXT INFORMATION
USER SPECIFIC NAVIGATION
GENERAL SITE NAVIGATION
MAIN CONTENT AREA
FOOTER – ADDITIONAL SITE DETAILS

Page 1 – Home Page

Common content elements described here (navigation, header, footer) also apply to the other pages. This page should provide the following content / functionality:

- Filename: index.html
- Logo / primary branding in the header
- Placeholder for login name
- Representative links for user-specific navigation (login-in, signup, etc.)
  - Log in and sign up links must provide navigation to saved connections page
  - The link for the "Start a new connection" can be inactive for this stage.
- Representative links for general site navigation (home, connections, about, etc.)

    - Connections link must provide navigation to connections page
    - Logo image must link to home page
- Representative footer information (copyright, etc.)
- Main content area should provide an introduction and overview to the site. May be more than simple text.



Welcome to CCI Connections site.

Attend and create CCI events to connect with people. Whether you are looking for a study buddy, trying to connect with others who are intersted in something you love to do, or want to try something new.

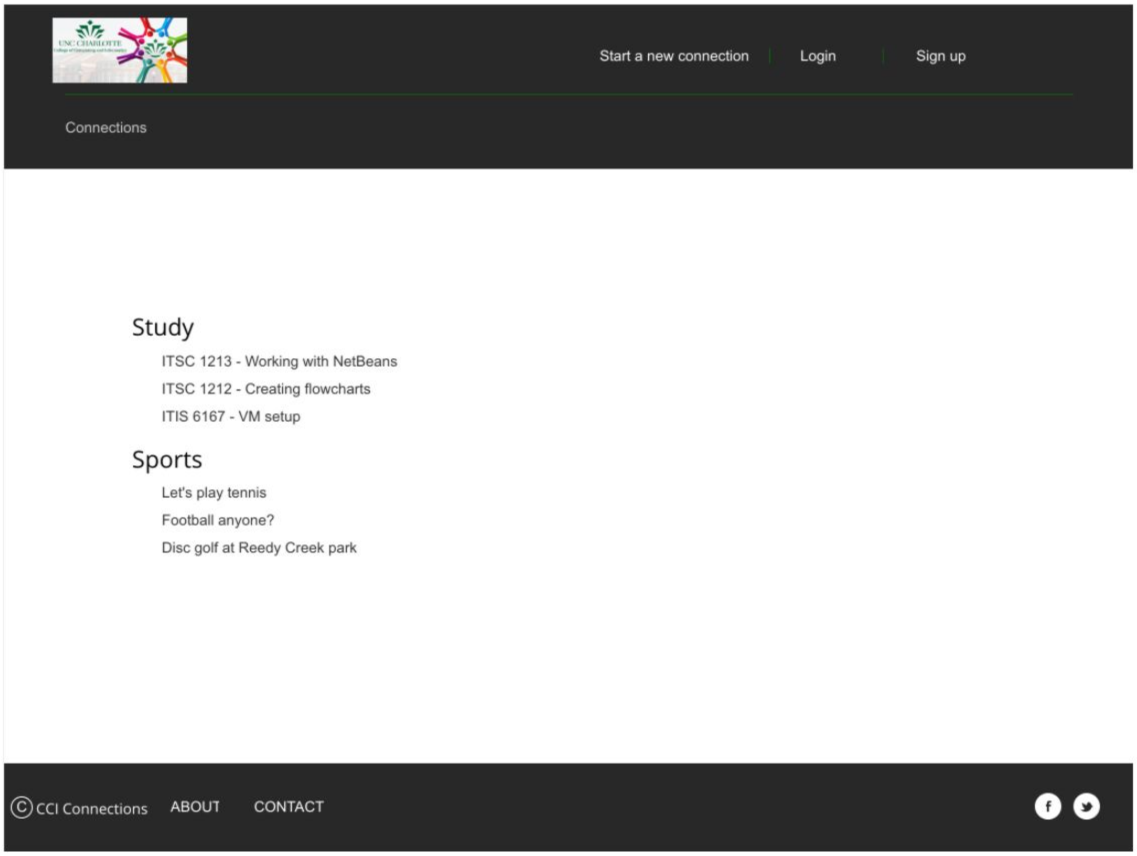
Check out the Connections section to browse available connections!

Page 2 – Connections Page

This page should provide the following content / functionality in addition to common content:

Filename: connections.html

- At least two categories of connections
- At least three connections per category/topic
- All connections listings should link to the same connection page, as a design placeholder
- Note that within the main content container, category/topic organization may be fancier than a plain list.



Study

ITSC 1213 - Working with NetBeans
ITSC 1212 - Creating flowcharts
ITIS 6167 - VM setup

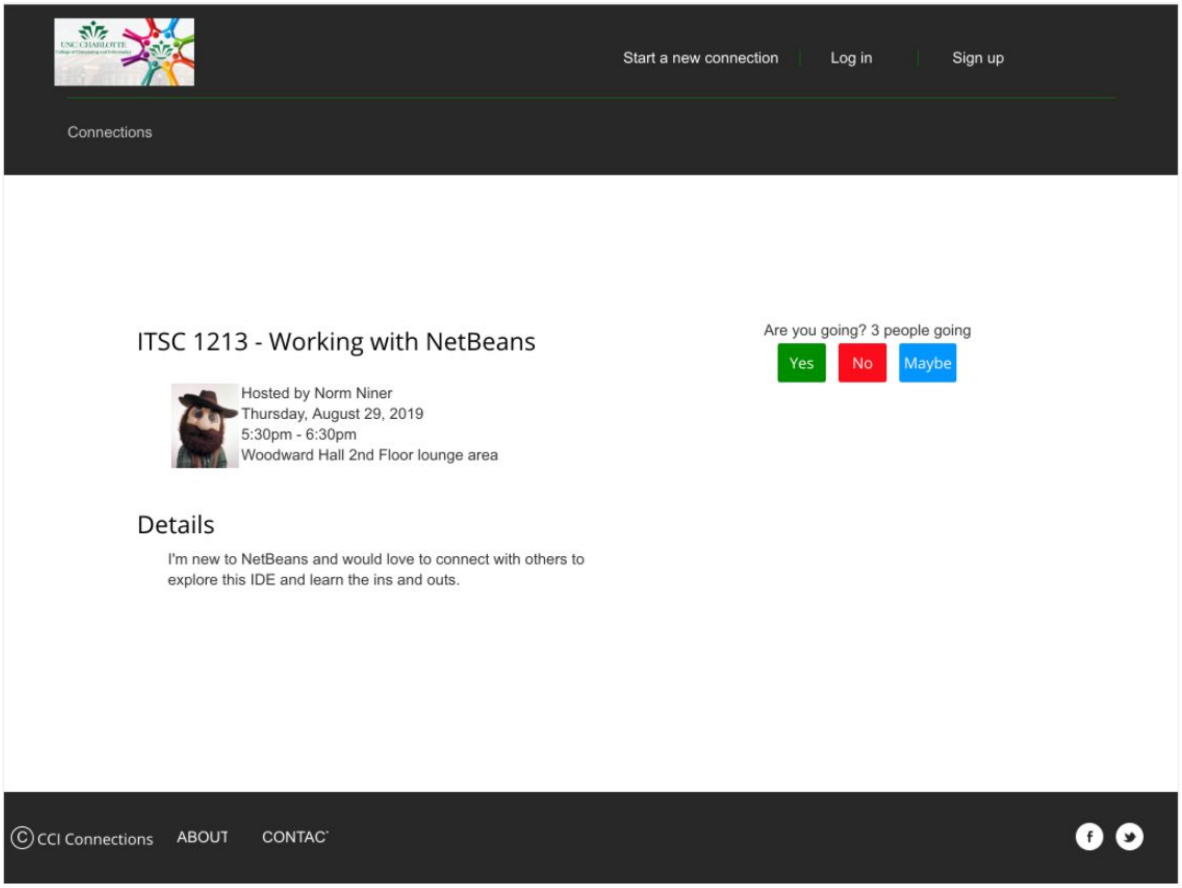
Sports

Let's play tennis
Football anyone?
Disc golf at Reedy Creek park

Page 3 – Connection Detail

This page should provide the following content / functionality in addition to common content: (Note: only one connection page is needed as a representative page – not a different one for every item in the connections list)

- Filename: connection.html
- Picture of host user
- Connection highlights, including category/topic, host user name, time / date, and location
- Connection detail description
- Buttons to add the Connection to a saved connections list
  - Button action should link to the saved connections page



ITSC 1213 - Working with NetBeans

Are you going? 3 people going

Hosted by Norm Niner
Thursday, August 29, 2019
5:30pm - 6:30pm
Woodward Hall 2nd Floor lounge area

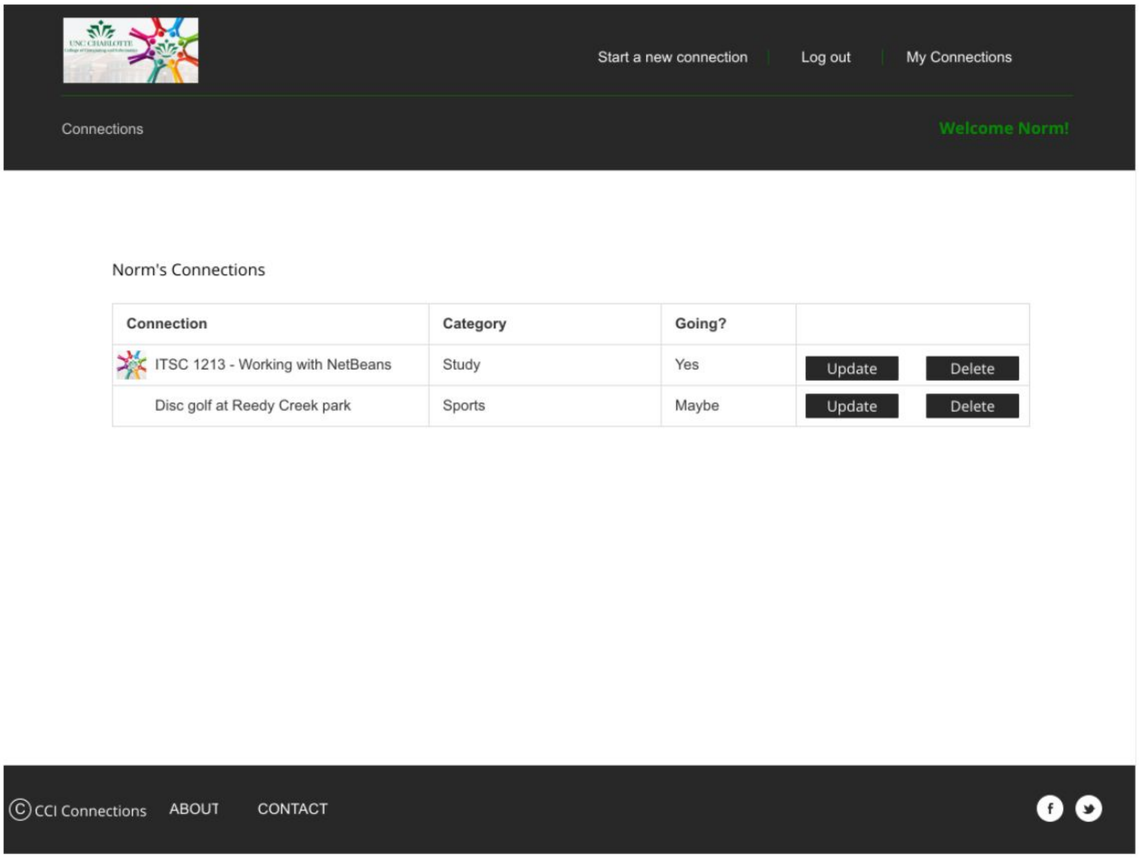
Details

I'm new to NetBeans and would love to connect with others to explore this IDE and learn the ins and outs.

Page 4 – Saved Connections

This page represents a specific application user's profile. It includes a summary of the connections they've made or interested in and saved in this application. This page should provide the following content / functionality in addition to common content:

- Filename: savedConnections.html
- Main content container should provide a table structure listing representative connections saved to user profile
- “Update” button action should link to the “connection” page
- “Delete” button should link to back to the “saved connections” page
- The link for creating new connections should be inactive for this stage.



Norm's Connections

Connection	Category	Going?	
ITSC 1213 - Working with NetBeans	Study	Yes	Update Delete
Disc golf at Reedy Creek park	Sports	Maybe	Update Delete

Assignment Submissions

What to submit using Canvas (Email submissions will NOT be accepted):

1. **Milestone1.zip** - An archive of the web application views folder stored in a standard ZIP File. **Do NOT** use another compression format such as .rar or .7zip or you risk getting a 0 on your assignment.
2. **Milestone1Info.pdf** – PDF document with the following assignment information:
    1. A screenshot of the application's home page displayed in your browser.
    2. Explanation of additional features, if any.
    3. Explanation of status, stopping point, and issues if incomplete.
    4. Discuss the easy and challenging parts of the assignment. How did you overcome all or some of the challenges?