# CS 290 – Assignment Description

Note: Please check this document online for changes and clarifications before you begin working on an assignment.

Most homework will involve programming a webpage. It will be released every Thursday and you will have one week to complete it.

To accommodate unforeseen circumstances that keep you from completing homework on time, you are allowed two one week extension. No extensions will be granted past that. To get an extension, email me with subject line: Request for an extension – Homework#

## Presentation: Present an idea for your site

Each group/person will present their concept for the website they intend to build as part of this class project. The presentation should be between 4 to 7 minutes leaving some time for follow up questions. You will be graded out of 10 points.

## How-To Guide – 25%

- Describe how to use one library, API, feature, or tool that you discovered for web development
- Turn in a 1-3 page PDF
    - 1st paragraph: what is the name of the library/API/feature/tool; what is it good for?
    - Remainder: step-by-step instructions for how to install and use the library/API/feature/tool on an example
- Graded on completeness and clarity

**Step 1: Find an API.** The best way is to start making a website. Once you identify the features you want in your website, you will be well on your way thinking about a library, API, feature or tool that you can eventually use.

**Step 2: Become an expert.** Once you have identified an API, take time to understand it. Start with its intended purpose (which may be different than you intend to use it for), identify how to use it, the library of functions it provides and how to integrate them onto your site.

**Step 3: Write your How-To Guide.** Once you have accomplished a good understanding of this API, library, feature or tool, the next step is to write a guide that one of your classmates could use to quickly integrate into their code. Keep in mind where you started, what you needed to do to get it working, items that were confusing – and let this guide your writing process.

**Step 4: Test your Guide.** After you write your How-To Guide, ask a classmate to try it out. Have him/her walk through every step of the guide. Seek feedback from them and update your guide as you see fit. You should test their How-To Guides and provide feedback accordingly.

**Step 5: Upload it to canvas.**

## Project:

Your project will be graded on completeness and correctness.

Create one web application that includes:

- User account setup: allow creation of new accounts
- Consume new data from logged in users
    - Only logged-in users should be able to post data
- A page for listing data provided by users
    - You can require login if you want; it's up to you.
- One "cool feature" that preferably takes advantage of some other student's How-To Guide

Note: Your own creative projects are encouraged. Please list out the static/dynamic features of your site on a one page write-up and get prior approval from the teacher, in class or via email.

After you implement your project, take one screenshot for each of the following: the page where users choose the username+password and/or login, the page where users provide some sort of data, the page where users can see the data sent back from the server, and finally the page where you show off your "cool feature". Write one sentence about each of these screenshots. It is ok if you write more about each screenshot. But it should be 1 sentence minimum. Just make sure you include all the information below. Finally, turn in your 4 screenshots with 4 sentences as a 4-page PDF (to Blackboard).

The completeness criterion will include having four pages with four screenshots and four short explanations of the content, and whether you appear to have implemented the four required parts.

The first page of your PDF should show a screenshot and your explanation about the page where the user creates an account and/or logs in. In your sentence, explain what happens when the user is logged in. For example, do you set a session cookie? Then what is the name of the session cookie? How are you storing and checking passwords?

The second page of your PDF should show a screenshot and your explanation about the page where the user enters information into a form, so it can be stored on your site. The user should not be allowed to use this page unless if he or she is logged in. In your explanation, tell how you store that information (One database table? Two tables? Some files in addition to the database?)

The third page of your PDF should show a screenshot and your explanation about the page where the user gets to see some information retrieved from your site. In your explanation, tell how you retrieve the information (database query? Do you do any caching?)

The fourth page of your PDF should show a screenshot and your explanation about the page where the user gets to use your "cool feature". In your explanation, tell what the feature does and what special APIs you used to implement the feature.

If it is unclear whether you actually implemented the required parts, the instructor may require you to give a demo of your web application. For example, the instructor may require you to join him or her on a Skype call so that you can give this demo. So implement your site and do a good job of explaining the pages.

**Note: Points will be deducted for tawdry look and feel of your site. It doesn't have to be stylish but it should meet basic guidelines of a good website as discussed throughout the class.**