

CS 3800 – Web Development I

Final Project – Fall 2019

Your final project for this course is individual, meaning that each student will be designing and implementing a website of their choosing. Keep the following in mind when deciding on the purpose of your site:

Your site should be some sort of business that requires interaction with a customer via the website. Some examples are an e-commerce site, a site that makes reservations or schedules appointments, or a restaurant that allows customers to place take-out orders. You may not use sites that are similar to ones done as class examples or for earlier examples.

Be sure you are following good website design as described in the Website Guidelines document. In addition:

- Use separate files for items that are the same on all pages, such as the header, footer, and navigation sections
- Use functions for all main tasks and for all database queries
- Be sure to validate input as needed (for example, if you ask for a phone number, be sure it's in the correct form)
- Be sure to filter input to avoid injection attacks
- All inventory or description of services, etc., must be stored in a database and retrieved through a query. Do NOT hard-code any specific items within your website.

Your website must use these features:

- User interaction that involves purchasing/scheduling or similar
- You must use at least one form as part of the user interaction.
- Use images and text to display items to be purchased/scheduled
- A shopping cart or similar that shows the user what they have purchased or scheduled. The shopping cart will be stored in a database table and must retain its contents between logins.
- Login/logout as described in part 3. You may want to leave this feature until late in your implementation so you do not have to deal with logins during testing.
- At least one admin page that allows the admin to modify inventory or similar operation
- Styling: all pages must have a professionally-styled fluid layout.
- Functionality for the user to view past orders/appointments. For an ecommerce or restaurant site, include only completed orders, not items left in the cart. For scheduling sites, include only completed appointments (this may require an admin to mark appointments as completed rather than just assuming if the date has passed it is completed).

Wow! points. Correctly meeting the requirements above will result in a score of 90%. You must include other advanced features or enhancements for the final 10%. You do not need pre-approval for the features you use, but you may want to discuss your plan with the instructor to be sure it satisfies this requirement before your implementation is complete.

Presentation – you will present your work to the class during the week of December 2.

Weekly progress checks – Dates and requirements for each progress check are described below. Students may complete the progress check early by appointment. No late progress checks will be conducted. Students who do not attend class the day of the progress check will receive a 0 for this check unless it was completed early or unless there is as documented emergency. In the case of such an emergency, the student must contact the instructor to schedule a makeup.

Overall Grading (300 points possible):

- ¹Progress checks 1-6: 6 @ 25 points each = 150 points
- Completed website/database: 100 points
- ²Presentation: 25 points
- Is able to answer instructor questions regarding design and implementation during grading appointment: 25 points

¹Progress check points will be given using the following standard:

- Fully meets all requirements – 25 points
- Meets most of the requirements – 20 points
- Completed but with significant missing elements – 15 points
- Minimal progress - 5 points
- Very minimal or no progress - 0 points

²Presentation expectations are described at the end of this document.

Progress Check 1 – Tuesday, October 22

1. Choose a subject for your site that meets the requirements listed on the first page of this document.
2. Create a **written plan** for your website. Include a description of the basic content/functionality for each page in your site.
3. Create a **written plan** for your database. Indicate the tables you plan to use (you must have at least three) and what fields (columns) will be needed for each table. Be sure to indicate the primary key for each table as well as any foreign keys that are used.
 - One table in your database must be a users table, which should have columns for username, password (type varchar(255)), and an account type column that indicates the type of account (admin or customer). The customer account type should be set as the default. Do NOT include additional fields; other information for clients/customers must be stored in a separate table with a foreign key for username.

Progress Check 2 – Tuesday, October 29

For this step, you will create the database for your website.

1. Incorporate any changes needed to your design based on your previous progress check.
2. Write a script that creates and populates your database tables. Be sure your script begins with statements to drop the tables if they already exist. This will allow you to run the script to re-create the database with the original data whenever you need. Note: The first line of the script should be a USE clause with the name of the database. For this project, use a database with the name *yourLastName*.
 - Note that you will need a table in your database called 'users' with a username and user_password column. Be sure the password column is of type varchar(255). You must also add to this table a 'usertype' column, to indicate whether the user is an *admin* or a *customer*. Set a default of *customer* when creating that table so the login functions that create a new user do not need to be modified. (You may choose a different value for the usertype instead of customer, such as client or patron, to suit your database.) Do not add any other fields to this table.
3. Prepare sample queries that you can run from phpMyAdmin during your progress check. You should have these queries stored in a document from which you can simply copy-and-paste. Prepare at least one query of each type: SELECT, INSERT, UPDATE, DELETE. At least one query should require an INNER JOIN.

Progress check: Demonstrate the creation of your database by importing your.sql file in phpMyAdmin and running the queries from step 3.

Progress Check 3 – Tuesday, November 5

For this check, you must have your home page (index.php) and shopping cart (or similar) completed. Include a (non-functioning) login button at this time.

Your home page should include user interaction (buy merchandise, schedule appointments, or similar). User selections should be stored in a database table. The user should be able to view her selections by going to a cart page. User should be able to modify her order from the cart page.

Progress check: Demonstrate user purchases/modifications through a working home and cart page.

Progress Check 4 – Tuesday, November 12

For this check, you must have log-in functionality working for your site, using the following procedure:

1. Add the provided log-in **folder (the entire folder, not just the files)** to your website. Information on using these files, and describing what is and isn't allowed as far as modifying these files, may be found in "Using the log in files" document provided within the files on Pilot.
2. Add functionality to the log-in button on your home page. When the user clicks the button, she should be directed to the login_start.php page (this page is part of the provided files). Once the create account/log-in process is completed, the user should be returned to the home page. The create account/log in button on your home page should change to a logout button when there is a logged-in user. An example of this implementation may be found in Content->Lecture Activities->Encrypt_password_hash.zip.
3. When testing your work, create at least two users. When they are inserted, they will both be created with the type "customer" (or client, patron, etc., if that is what your default type is). Then go to phpmyadmin to set one of the accounts to an admin account. You should NOT be able to create an admin type from your website.
4. If the logged-in user is an admin, display an additional button(s) on the home page that links to an "admin" page. This page should only be accessible by a logged-in admin. You do not need any functionality on this page at this time, but include enough content so that when testing it is obvious that it is an admin page.
5. Add functionality to your home page so that only a logged-in user may purchase/schedule items. Also, only a logged-in user should be able to view the cart or pages in your site other than the home page.

Progress check: On your local machine, demonstrate the creation of user accounts and the behavior of the site when each user type (user and admin) attempts to go to the admin page.

Progress Check 5 – Tuesday, November 19

For this step, you should have a complete web site that meets the requirements as given on the Project Overview document. This part of the project will vary based on your site's purpose, but should include the following:

- Your home page should be in its final form
- You should have completed specific admin functions
- Most (90% or more) of the functionality and styling should be complete.

Progress check requirements: You will be asked to demonstrate your website on your local machine.

Progress Check 6 – Tuesday, November 26

For this step, all functionality and styling should be complete.

You should have created your database and uploaded your website to the course server.

Progress check requirements: For this progress check you will be asked to do the following:

1. Demonstrate your website **on the course server** to verify that all pages display correctly and that the site is able to successfully connect to and interact with the database.
2. Provide a list of features that you have used to meet the “Wow! Points” requirement.
3. Sign up for a grading appointment to take place after the final deadline.

Note: You may do final ‘tweaking’ of your site and re-upload it before the final deadline of Monday, December 2, 11:59 pm.

Final Presentation - Tuesday, December 2 or Thursday, December 4

Expectations:

- professional demeanor
- organized plan¹
- able to answer questions on design and implementation
- presentation is within the time limit (8-10 minutes). You will be penalized 1 point for every minute your presentation (not counting questions) is under 8 minutes or over 10 minutes.²
- Attentive participant for other presenters

¹ The organized plan should include a description of the site's purpose and a walk-through as both a demo user and an admin. You should include a way to demonstrate that the database updates when appropriate (if you don't have something in your site that clearly shows this has happened, you can go to phpmyadmin to show it).

² I suggest practicing your presentation to target for 10 minutes. Presenters tend to go a little faster during the "real thing." You might also prepare a little extra material, to use only if needed. You must demo your site from the course server, NOT from your local machine. Questions from the “audience” do not count in the time limit.