# SYSC 4504: Fundamentals of Web Development
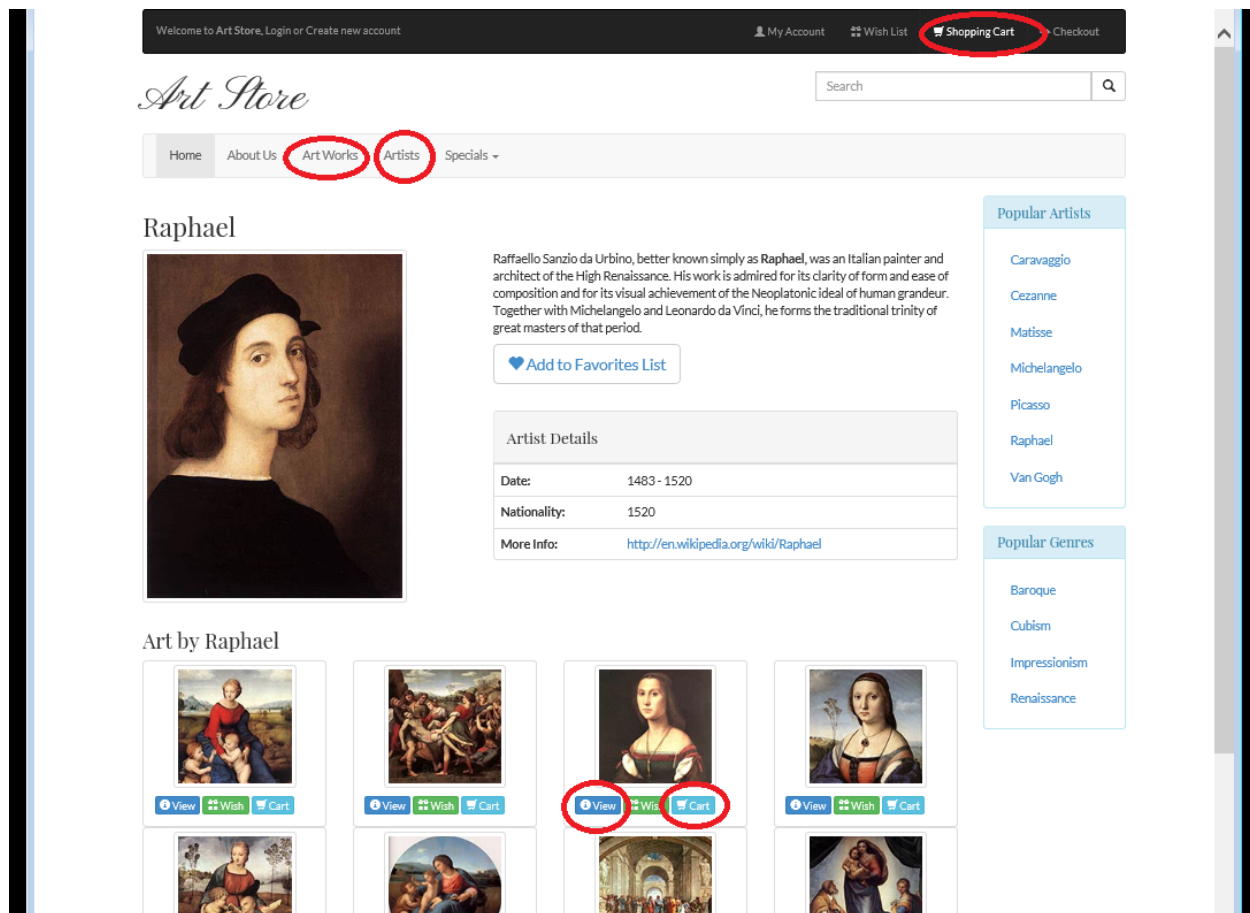# Programming Assignment 3:
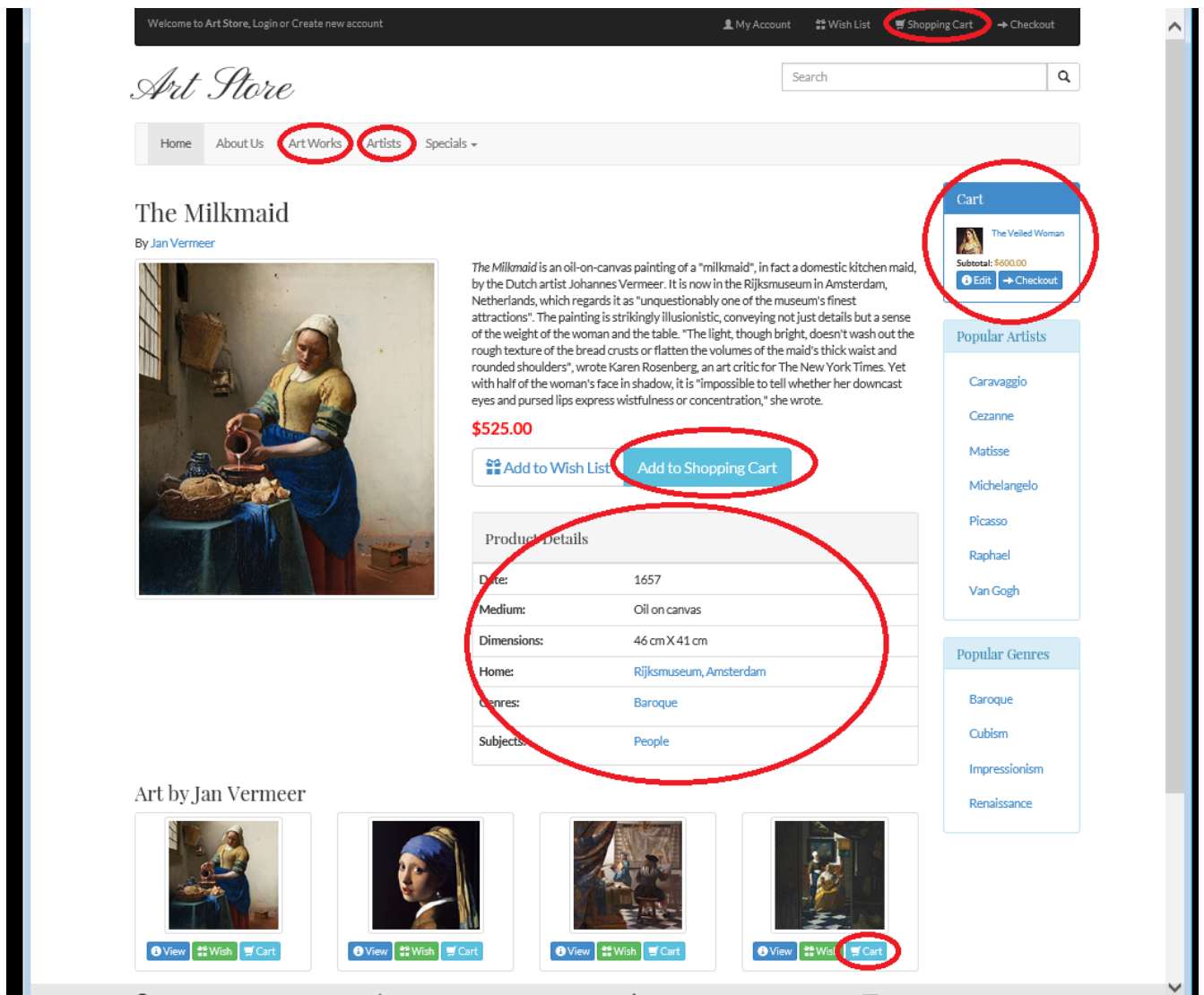# Server-Side Application Development
# Due: December 5, noon

This lab will bring together various elements from labs 3 and 4. It will provide a more complete server-side solution to the Art Store exercise. The course website hosts a ZIP folder with a number of files to get you started, alternatively you can also use the various partial solutions from the labs.

In this assignment, we will combine database queries to populate pages about artists and artworks with session state management to build and work with a shopping cart. The solution should use PHP to dynamically query the database server, similar to lab 4. It should also use PHP to calculate the costs of all items in the shopping cart, is done in lab 3, but this time for a dynamic collection of items. The solution consists of three related main pages:
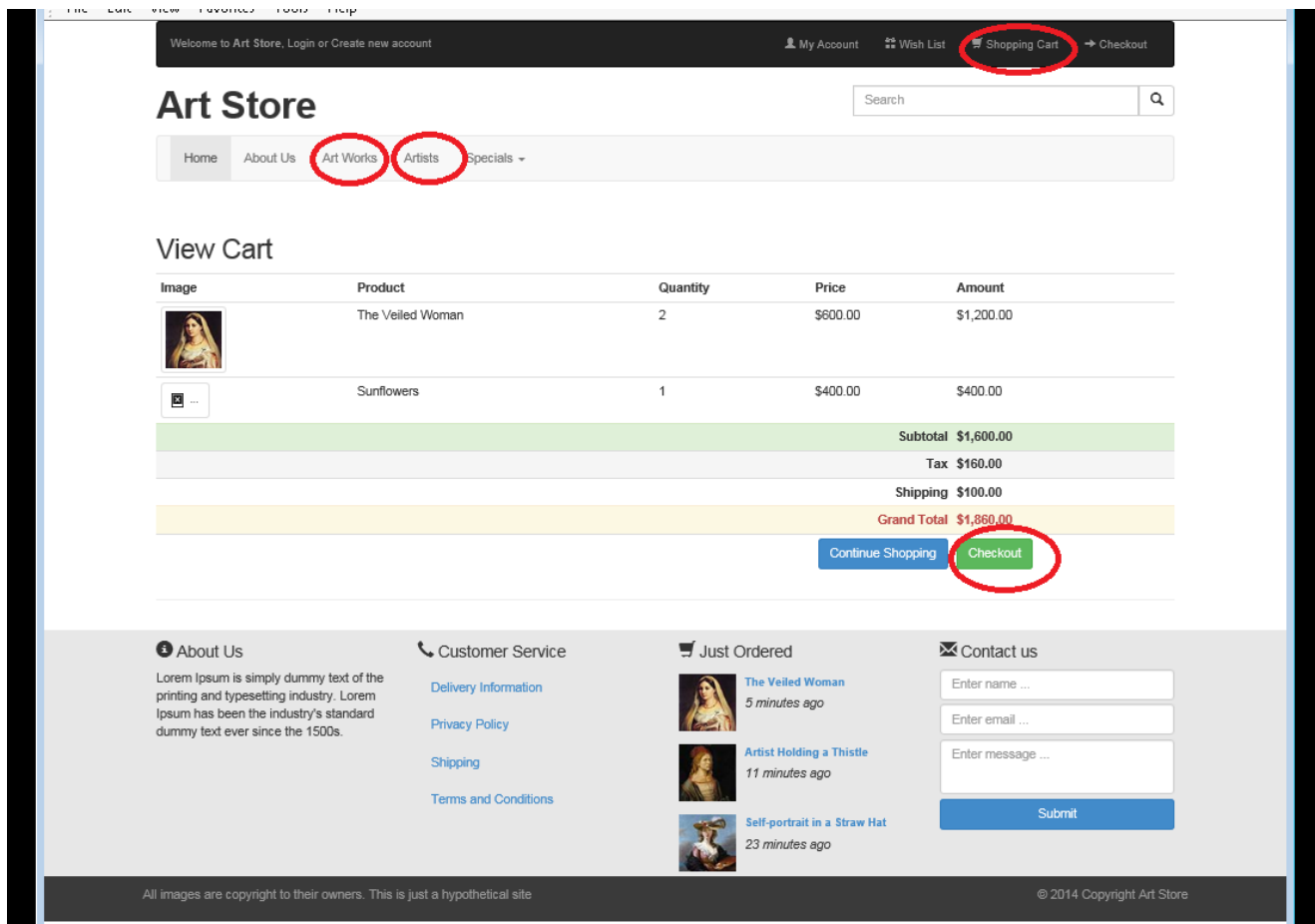
- `display-artist.php`: this page is basically what you will also work on in lab 4, with some extensions. It displays information about an artist in the centre page, with a list of paintings by this artist at the bottom. One key difference is that the links to the Shopping Cart (in the top navigation bar), Art Works, and Artists are all active, pointing to the various corresponding resources (`display-cart.php`, `display-art-work.php`, and `display-artist.php`). Selecting "View" associated with each image for this artist will bring up `display-art-work.php`, passing the Art Work ID in a query string. Selecting "Cart" will add the displayed image to the shopping cart (which is initially empty).

- `display-art-work.php`: this page display more information about the selected painting. You need to query additional tables in the ART database for this. Note that a single painting can belong to multiple `genres` and have multiple `subjects`. Below the main information for the painting, display other paintings for the same artist. When the user selects "Add to Shopping Cart" next to the painting, or "Cart" under each of the smaller images, this painting should be added to the shopping card (same as is the case for `display-artist.php`). Also note that the top left contains a condensed version of the current shopping cart content. If available, show the TINY version of an image in the card, as well as the total for all purchases.

- `display-cart.php`: This last page displays the current content and a detailed breakdown of the shopping cart total costs, as per lab 3. Notice that a user can have added the same painting multiple times to the shopping cart (in which case the count should be incremented, rather than the painting listed multiple times). When the user selects "Checkout", the shopping cart should be cleared. Where available, the page should display the TINY version of each painting in the shopping cart. We will discuss multiple ways in which we can maintain session states and you are free to choose your preferred style (some are more complex to use than others though).

## Submission Requirements

Submit your assignment using `cuLearn`. Submit all source files you wrote for this assignment as a single ZIP folder. You should NOT include the `images` folder or the `bootstrap3_defaultTheme` folder. The TA will install your submission in a subdirectory under `htdocs`, using the lab setup (Apache server, MySQL database). Make sure that, similar to the lab, all database access is done for `testuser2` with password "`mypassword`". The TA should be able to launch any of the three main pages and navigate between them using the shared/common links in the navigation header.

Marks will be based on:

- Completeness of your submission
- Correct solution to the problem
- Adhering to the submission requirements
- Proper comments in all code/resources.

The due date is based on the time of the `cuLearn` server and will be strictly enforced.