**Final Project**

**CS 6314**

Group Size: 4 members

You are to develop either an E-commerce site or a Social Media site.

**E-commerce Requirements:**

* User signup: Register a new user to the system.
  + Check the new username or email address if it’s already registered to the system using AJAX.
  + Form validation: Check if all mandatory fields are filled out. (JQuery)
  + Check if password is strong enough. Define the rules of having a strong password. (JQuery). User passwords should be hashed and hashed version of passwords should be stored in the database.
* User login
* List available products/services in the system. User should be able to filter the results based on at least one criterion (i.e. category). Also, should be able to “Search” for a specific item. Search and filtering should be integrated together.
* You will have paging functionality for listing products/services.
* Add item(s) to the cart and checkout. For the online shopping sites, pricing will be included but payment functionality doesn’t need to be implemented. However, you will keep track of inventory. User should be able to update items in the cart (remove items from the cart, update their quantity).
* Show the history of purchases or saved lists for the user.
* For Admin user(s):
  + List all items
  + Add new item
  + Delete item
  + Update item

Admin user will have the same interface with regular users, except that he/she will be provided extra features (buttons/links) for update and deletion of products/services.

While adding items to the system, admin user should be able to upload pictures for the items and while updating items, should be able to change pictures.

For delete, soft-delete should be implemented.

Optional (You can implement Wish list or Favorites and save those listings.)

**Social Media Requirements:**

* User signup: Register a new user to the system.
  + Check the new username or email address if it’s already registered to the system using AJAX.
  + Form validation: Check if all mandatory fields are filled out. (JQuery)
  + Check if password is strong enough. Define the rules of having a strong password. (JQuery). User passwords should be hashed and hashed version of passwords should be stored in the database.
* User login
* Handle privacy settings for each user. What other people are allowed to see (public, private, friends only, friends of friends).
* Upload pictures to their account.
* Post Comments on their page or friend’s pages. Comments include pictures, videos, and text. If user settings does not allow the publishing of content by other people without approval, you need to notify the user to allow them to accept or reject the content.
* Search for other user’s on the site you should be able to filter these results.
* Search the content of a user’s page you should be able to filter the results.
* Users should be able to send private messages to other users that only they can see.
* For Admin user(s):
  + Access User accounts
  + Update User accounts
  + Delete User accounts
  + Delete specific content of the user account

Admin user will have the same interface with regular users, except that he/she will be provided extra features (buttons/links) for accessing user accounts and the deletion of user accounts.

**Both Designs:**

* You need to come up with a unique feature that is not part of the requirements above. This feature should set your website apart from everybody else.

**Front-end design**

You can use Bootstrap and Bootstrap templates for user interface design. You can also use design tools (i.e. FrontPage, Dream Weaver). You are flexible for page designs but it is important to have a professional look for all pages (i.e. the same menu or navigation bar appears on top of every page etc.).

You can also use frameworks (i.e. Angular JS) for client-side scripting.

**Back-end design**

You can use relational databases or MongoDB for database component. In case a relational database system is used, tables should be normalized to Third Normal Form.

For server side scripting, you can use PHP, Ruby on Rails and Node.JS You are also allowed to use PHP frameworks such as Laravel, CodeIgniter, CakePHP, etc.

**Do not use a pre-built shopping cart system or back-end system. This must be developed using your own code.**

**If you use a platform that was not covered in class every group member is responsible for understanding how that platform works.**

**Your project must be new. If you have develop a system in the past do not use it for this project.**

**Project Report**

Project Report should include following items:

* Project Title
* Name for the Website
* Project Description: a brief description of the website. If you make any assumptions, make sure to include those assumptions.
* Database Design: Database structure including the tables and primary and foreign key definitions on tables. Alternatively, you can include your “Create Table” statements for this. In case you use MongoDB, provide information about your data model.
* Languages/frameworks used for implementation
* Screenshots for main functionalities (5-6 screenshots will be sufficient)
* Team Members: Names and Net-IDs for team members
* Work division among team members: who has completed which part (be specific and try to give page names). Please note that every team member should be actively involved in every step of the project development (database design, client-side and server side scripting) and work should be equally divided between team members.

**Deliverables:**

You will show a live demo of your project in the classroom. Presentations will be done on **April 25, April 30, and May 2, 2019** during regular class time.

Each team is responsible for sending following items for final project:

* Project report (File name should be Coursenumber-Teamnumber-ProjectTitle.pdf. Example: CS6314.0U2-Team4-OnlinePizzaOrders.pdf)
* Source files (zip your home directory for your project)
* Sql dump file for your database if you are using a relational database

Due date for submission is **April 24, 2019, 11:59 PM.**