CEWP 1002 – Intro to Web Programming
Concordia University, Spring 2017

Web application project guidelines
Worth 50% of your final grade

The aim of this project is to develop a Web Application implemented in PHP and using a MySQL Database. At the end of the semester, you will be able to showcase your work as a portfolio item. Below are the guidelines and deliverables for the project.

1. Project Guidelines

From the beginning of class, you can start thinking about an original web application idea that you will be implementing during the semester. Using your client-side programming skills (HTML, CSS, Javascript), you will be able to design and prototype the static elements of the website such as forms and navigation.

Upcoming lectures on different server-side topics using PHP and MySQL will help you integrate the dynamic elements of the site. Below are the list of features expected (section 1.1) and some example projects (section 1.2) to help you get a good idea for your project. You can implement alternative features with the approval of your instructor. You are allowed to work individually or in teams of 2 students.

1.1 List of expected features

Required Features (For a passing grade)

- User Registration and login with username and password
- Special user access (Admin) to manage products in the case of an online store
- Product Description driven by the database (or equivalent)
- Each form should use client-side and server-side validation
- Functional shopping cart (or equivalent)

Advanced Features (For a better grade, teams must implement 2 advanced features to pass)

- Encrypted passwords and other sensitive information in the database
- Email or SMS confirmation for user registration
- Inventory System (or equivalent)
- Google Map integration
- Frontend implemented solely in html, all requests to webserver using AJAX
- Deployment of the application online

1.2 Example Project Ideas with main feature breakdown

Below is a list of example projects you can inspire yourself with. Your project doesn't need to match exactly what is listed below. Ideal projects should be useful web applications.

- Example 1 : Online store
  - User account for online shopping
  - Admin account to manage products, receive orders and send shipping confirmation
  - Inventory driven from Database
  - Shopping Cart Implementation
  - Checkout System with email/SMS confirmation
  - Invoices can be visualized by user

- Example 2 : Restaurant online deliveries
  - User accounts for placing orders
  - Admin Account to update menu, receive orders, send confirmation for deliveries
  - Online Menu driven from Database
  - Items added to shopping cart
  - Checkout System with email/SMS confirmation
  - All orders can be visualized by the user

- Example 3 : Appointment system for a service (Eg. Barber shop, Dentist, etc.)
  - User accounts to make appointments using a Calendar
  - Admin accounts to update services, receive and confirm appointments with users
  - Calendar to schedule appointments, each time slots driven by database
  - User can make an appointment by selecting an available time slot
  - Email/SMS confirmation for making appointments
  - Previous appointments can be visualized

- Anything else you feel is appropriate, you can verify your idea with your instructor.

2. List of Deliverables

Below is a breakdown of the work that must be submitted to Moodle throughout the semester.

2.1 Project Frontend Prototype

No PHP Programming is required for this deliverable. You need to mock-up all the screens for your project in HTML, CSS and Javascript (if necessary). Your prototype should include navigation between the various pages of your application, all the forms required by your application with validation, and example of the dynamic elements that will be driven by your database for the final deliverable.

2.2 Database Implementation

The project must involve Inserting, Deleting, Updating data in a database, as well as querying the database for output data. For this deliverable, you need to submit a SQL script that will generate all your tables, and a few records for each table. Your database should not contain redundant items and tables should be properly related.

2.3 Project Presentation

You must showcase your Web Application and present it to your classmates. You must present all the elements from the Project Report (described below). You are allowed to use Powerpoint. You have a total of 15-20 minutes for your presentation including questions. You will be interrupted if you run out of time.

2.3 Project Report

At the end of the project, you need to write a report (approximately 1500 words) in Word, PDF or Webpage format. The report should contain the following sections:

- **Project outline**
  - Purpose (why is it being developed?)
  - Applicability (who will use the system, and how will they benefit?)

- **Project feature breakdown and development schedule**
  - For each major feature
    - Who implemented it (if done in a team)
    - How many hours did it take to develop
    - Highlights and limitations (future works)

- **Results**
  - Screenshot for the main screens of your project

2.5 Final Project Code

At the end of the semester, when the project is implemented, you must submit all the code required for your web application, including all the PHP files, html, css, javascript, images as well as the database script to generate your application.

2.6 Deliverables Schedule

Item	Due	Worth
Project Frontend Prototype (HTML, CSS, JS)	May 8	5%
Database implementation	May 15	5%
Final Project Presentation	May 31	10%
Final Project Code submission	May 31	20%
Final Project Report	May 31	10%

You must submit all assignments by the deadline, or it will be marked 0. You will be granted an extension only if you have a valid excuse such as a doctor's note.