CIS 3308 Web Application Programming Syllabus (Spring 2020 – Kyvernitis)
(Upper Level CS Elective)

Course Description
This course explores techniques that are used to design and implement web applications – both server side and client side code. Using open source (free) development tools such as Netbeans (multi-language editor with context sensitive error messages), Apache (web server), Tomcat (JSP application server), MySQL (Database Management System) and MySQL Workbench (GUI to help you create SQL to interact with a database), students write code in the following languages:

- HTML (structure and content of web page within browser), CSS (styling of web page)
- JavaScript (code that runs in the user’s browser)
- Java/JSP (server side code that we use to create Web APIs that provide database access using SQL within Java).
- SQL (to create and modify data).

In this course, students shall:

- create a web application UI (User Interface) using HTML and CSS.
- use JavaScript to achieve user interface code reuse.
- create their own MySQL database and populate it with data.
- learn how to publish and test their web applications using internet protocols such as ftp, sftp, http, and https.
- create Web APIs (server side Java/JSP code that access their database) then write the client side code (JavaScript/AJAX) to invoke those Web APIs. The Web APIs produce data in the JSON web format which is easily converted to JavaScript objects (client side) and to Java objects (server side).
- learn about server side request and response objects (request being a representation of what the browser requested, along with any URL input parameters, and response being the server’s response to the client’s request).
- implement log on and security, by using server side objects: session (to store user information such as logon status), and response (to redirect the user to an error page if they are not logged on).
- write code according to software design patterns such as MVC (Model-View-Controller), SRP (Single Responsibility Principle), DRY (Don’t Repeat Yourself), and Dependency Injection (don’t allow code module B to reference an object of code module A unless A has provided B with a reference to that object).

Although each student’s web application will employ HTML/CSS to provide some aesthetic appeal, web design is not the major thrust of this course. Instead, this course focuses on designing and implementing client side and server side code to create a reliable, secure, extensible, and maintainable web application. We will focus on the enduring basics of web development (HTML, CSS, JavaScript, Server Side Programming), but we will also learn a bit about ReAct which is (at this moment) a popular JavaScript Single Page Application Framework. Students must recognize that frameworks come and go but the basics are enduring.

Prerequisites

- Grade of C- or better in CIS 2107 Computer Systems and Low-Level Programming
- Grade of C- or better in CIS 2168 Data Structures

The only assumption is that students have knowledge and skills obtained in the pre-requisite courses listed just above. All other topics will be introduced as new material, even though some students may already have had some exposure. **Students with more experience can add extra functionality to their weekly assignments (if they wish), as long as they meet all homework requirements and submit on time.**
Textbook
There is no text book. Web references and other materials will be posted online.

Course Format

- **Homeworks.** Almost every week, there will be a programming assignment that is highly related to (reinforcing) lecture topics. To get a grade for your homework you must:
  - Complete the homework assignment and test it locally.
  - Publish it and test what you published.
  - Upload a zip file of your WHOLE web application (NetBeans project) into Canvas.
  - If you have not completed your homework by the due date, you can still complete it by the following week (with a -20% penalty). After that, homeworks are not accepted, but you still have to complete the work to avoid further deductions on your project grade. (The project is just the culmination of all your labs, regression tested so that everything all works together.)
  - If there is ever any question about a Homework grade, we will go by the code that you have uploaded into Canvas.

- **Project.** Your project is the culmination of all your labs (all combined into a SINGLE, regression tested, web application). Your project grade is based on functional testing plus a code review of all your source code. So, each week, be sure to keep your code well designed/organized, bug free, using self documenting names and adequate comments. EVERY WEEK, your NetBeans Project shall grow – don’t create a new project each week.

- **Lab Activities.** Almost every week, during your lab period, there will be a graded lab activity.
  - Lab activities are short exercises that are started in lab. Before the end of lab, you must demo your work to the Lab Assistant for grading and also upload the code to Canvas. (Any questions about Lab Activity grade will be based on the code submitted into Canvas.)
  - The activities are typically designed to get you started on your upcoming homework programming assignment for the week. So it is not uncommon that a Lab Activity asks you to do a subset of the homework assignment. Even if you do not completely understand everything that you are doing in the lab activity, try to keep a reasonable pace so that you can finish as many milestones as possible. You can always repeat your lab activity (and experiment, digging deeper) after the completion of the lab period.
  - If you miss a lab activity, there is no make-up (but if a student has a long term documented illness, let the instructor know – perhaps an accommodation can be reached). If you miss a lab activity, it is likely that you will have trouble getting started on your homework assignment.
  - To account for any unexpected problem, your lowest Lab Activity grade will be dropped.

- **Tests.** There will be two tests. The only way to do well on the tests is to experiment a lot with sample code and homework code.
• **Working Together Option.** Two students may elect to work together on their Homework/Project. Here are the requirements for students who work together:
  o Each student must do their own lab activity.
  o The footer must include names and and NetAccess username (e.g., tua12345) of BOTH students.
  o Each student must have their own database - the data should be different between the two databases but the design must be exactly the same. Each time they publish, each student must ensure that the database connection wrapper class (DbConn) directs to their database credentials.
  o Each student must publish the code to their own web site, The code can be the same as their partners – all except the database connection string.
  o Each student must upload a zip file of the code to Canvas by the homework due date. When uploading code to canvas for partner submissions, each student must indicate what percentage of the work was done by each student. If one student does all the work one week, the other student must do all the work the next week or both students can do half the work each week.
  o Remember that if you do not work with your partner (you allow your partner to do everything), your test grade will suffer and you may not pass the course. It is your responsibility to learn what is being taught in each homework.
  o If one student is not “pulling his/her weight”, the other student may elect to “go solo” at any time – by just removing the other student’s name from the footer and mentioning the new arrangement in the Canvas submission. If you elect to go solo, you must inform the other student by the end of the lab activity associated with the first solo homework assignment. Copy your instructor and Lab Assistant on the “going solo” email so that there is no miscommunication.
  o If you have worked with a partner for any homeworks, then you must select the tutorial option (see below) and each student must do their own tutorial work.
  o I’m sure there will be many scenarios and questions that will come up about working together. I guess we’ll address them as they arise.

• **Tutorial Option.** Most of the homeworks provide a fair amount of sample code which is necessary so that we can cover the vast amount of material that is needed for even beginning web development. However, many students prefer to write code from scratch (which I totally understand). You are in luck – select the Tutorial option where you can write your own code any way you like (as long as it adheres to the good software design practices covered in class) and you can pick from a large array of topics. If you select the tutorial option, you are excused from a couple of homeworks.
  o In order to get a course grade better than B you must select the tutorial option.
  o If you are struggling with homeworks, it may be best not to select the tutorial option because it is a very open ended assignment with no sample code.
  o The tutorial involves writing extensive JS code (consumer/provider style), explaining the code, and presenting to the class.
Tentative (and approximate) Grade Weights

| Lab Activities, must be completed in lab (approx. 11, drop lowest) | 15% |
| Programming Homeworks - approx. 9 OR if Tutorial selected - approx. 7 HWs and Tutorial | 20% |
| Project (culmination of all Homeworks/Tutorial plus code review) | 15% |
| First Test | 25% |
| Second Test | 25% |

100%

EXCEPTIONS:

- If the average of your test grades is less than C-, your course grade will also be less than C- (and you need at least C- to be able to count this course towards CS major requirements).
- To get a course grade higher than B, you must select the Tutorial Option.

Temple Grade Scale

| 93-100: A | 83-86: B | 73-76: C | 63-66: D |
| 90-92: A- | 80-82: B- | 70-72: C- | 60-62: D- |
| 87-89: B+ | 77-79: C+ | 67-69: D+ | 0-60: F |

Miscellaneous

- The CIS department computer labs are NOT open 24/7. Learn the lab hours and adjust your schedule accordingly. If you want to work from home, set up your own development environment. I have tutorials and instructions that can help you with this. Those who promptly set up their own web development environment do much better in this course than those who do not.
- **Attendance:** If you must miss lecture or lab, check Canvas to see what material was presented and ask your classmates about anything else that may have been discussed.
- **Communication:** Please contact me as soon as possible if you think you are running into difficulties. Ask me, or your lab instructor, or another student for help AS SOON AS POSSIBLE.
- **Disability Disclosure:** Any student who has a need for accommodation based on the impact of a disability should contact me privately to discuss the specific situation as soon as possible. Student must provide me with a note from the office of Disability Resources and Services (100 Ritter Annex, 215-204-1280).
- **Academic Honesty and Ethics:** Temple University and I expect you to observe the highest ethical standards. When working in the lab or on your project, you may consult others, but the work you submit must be your own. Never share your answers with others. Never accept answers from others. Unless otherwise directed, all tests are closed book, closed computer. All violations of academic honesty will be handled according to university policy.
**Tentative Schedule:** For detailed HW requirements, see the class website (google “sallyk temple 3308”).

<table>
<thead>
<tr>
<th>Week of Monday</th>
<th>HW #</th>
<th>Homework</th>
<th>Notes / Lab Activity will be related to HW unless otherwise indicated</th>
</tr>
</thead>
<tbody>
<tr>
<td>20-Jan</td>
<td>2</td>
<td><strong>Routing (JS UI reuse) &amp; Database:</strong> Design &amp; populate 2 table DB, submit web app proposal (must align with db).</td>
<td>Note: MLK holiday - Monday 1/20/2020 Lab Activity on your own at home, not graded.</td>
</tr>
<tr>
<td>27-Jan</td>
<td>3</td>
<td><strong>Display Data (JS FWs)</strong></td>
<td></td>
</tr>
<tr>
<td>3-Feb</td>
<td>4</td>
<td><strong>Option A: JavaScript Slide Show</strong> OR- <strong>Option B: Tutorial Proposal</strong></td>
<td>Everyone does JS Lab Activity regardless of which HW option selected. To get course grade better than B, you must select the Tutorial Proposal now and Tutorial later.</td>
</tr>
<tr>
<td>10-Feb</td>
<td></td>
<td><strong>First Test</strong></td>
<td>Note: Midterm ratings due Mon March 9, 2020</td>
</tr>
<tr>
<td>17-Feb</td>
<td>5</td>
<td><strong>Web API:</strong> Web APIs to support Display data. User Web API is sample, students create Other Web API.</td>
<td>For HW submission, add links from blog to the Web APIs</td>
</tr>
<tr>
<td>24-Feb</td>
<td>6</td>
<td><strong>Logon (two week HW)</strong></td>
<td></td>
</tr>
<tr>
<td>2-Mar</td>
<td></td>
<td><strong>SPRING BREAK</strong></td>
<td></td>
</tr>
<tr>
<td>9-Mar</td>
<td>7</td>
<td><strong>Delete</strong></td>
<td></td>
</tr>
<tr>
<td>16-Mar</td>
<td>8</td>
<td><strong>Option A: Insert</strong> OR- <strong>Option B: Tutorial</strong> (two week HW)</td>
<td>Everyone does Insert Lab Activity regardless of which HW option selected. Remember to get course grade better than B, must select Tutorial Option.</td>
</tr>
<tr>
<td>30-Mar</td>
<td></td>
<td><strong>React Insert</strong></td>
<td></td>
</tr>
<tr>
<td>6-Apr</td>
<td>9</td>
<td><strong>Update (two week HW) and Tutorial Presentations</strong></td>
<td></td>
</tr>
<tr>
<td>13-Apr</td>
<td></td>
<td><strong>Tutorial Presentations</strong></td>
<td></td>
</tr>
<tr>
<td>20-Apr</td>
<td></td>
<td><strong>Review/Last Test</strong></td>
<td>Code Review of Project in lab - bring other work to do while you wait for your code to be reviewed.</td>
</tr>
</tbody>
</table>

**Spring 2020 IMPORTANT DATES:**

- Monday 1/27: Last Day to Drop (you don’t pay for the course and it will not appear on your transcript).
- Monday 3/9: Mid Term Ratings end.
- Wednesday 3/18: Last Day to Withdraw. If you are doing poorly and do not want to have to later retake this CS elective to improve your GPA, withdrawing is a good option for you. Grade “W” remains on your transcript but does not affect your GPA.