CS 6314 – Web Programming Languages
Fall 2016
Dr. Mithun Balakrishna
Course Project

A. Project Steps and Deadlines:

- **Project Group Formation**:
  - Due by **Friday, October 21st 2016, 11:59pm**
  - A maximum of three (3) students per project group
  - The group should decide on an appropriate group name
  - One group member should submit a document containing the group name and the group member information i.e. Group name and Group member names, via eLearning
    - Please name the document following the convention "ProjectGroupInfo-GROUPNAME.pdf", where GROUPNAME is your project group's name.
    - Submit the document to the "Group Information Submission" assignment inside the "Final Project" folder listed in the course home page on eLearning.
    - Students that want to work on the project individually should also submit this document
  - Students that need help to form a group should meet the Instructor on **Friday, October 21st 2016** at **6pm** in the Instructor's office
    - Students that want to work on the project individually do NOT need to do this
- **Computing Resources**:
  - Deadline: **Friday, October 28th 2016, 7pm**
  - Please talk to the Instructor if your group does not have the computing resources (i.e. a laptop/desktop with internet connection and root/administrator privileges) to support implementation of this project.
- **Project Demo**:
  - Due date: **TBA**
  - Demo sign-up details: **TBA**
  - Submit your project source code and report via eLearning before your group's allocated demo session:
    - One group member should submit a single zip file containing the following via eLearning:
      - Project source code/script file(s)
      - A ReadMe file with instructions on how to access the project demo
      - Project report in PDF or MS Word document format.

        - Please name the zip archive document following the convention "ProjectFinalSubmission-GROUPNAME.zip", where GROUPNAME is your project group's name.
        - Submit the document to the "Project Final Submission" assignment inside the "Final Project" folder listed in the course home page on eLearning.
    - Please hand over a hard copy of the project report before the start of your group's demo session with the TA

B. Project Description:

Please design and implement a **responsive web site** and **scalable web application** based on the **service-oriented architecture** (SOA).

Mandatory Requirements

A. **HTML/CSS/JavaScript**: You are required to build your web site's client side Graphical User Interface (GUI) using HTML/CSS/JavaScript. You are required to use responsive HTML/CSS/JavaScript templated such as Bootstrap (http://getbootstrap.com) and Foundation 3 (http://foundation.zurb.com), etc.

B. **Server-side Programming**: You can use any programming language for your web site's server-side implementation and your web application's Web Services implementation.

C. **Domain**: The students can pick any domain, application, or service of their choice for their web site/application. Example: implement a project related to e-commerce (i.e. electronic selling and buying of products or services) such as online book store, online flight reservation system, etc.

D. **Functionalities**: Your web site and web application should support the following functionalities:
    1. New user registration
    2. Existing user login and logout
    3. User profile information display and editing
    4. User login information
        i. Last time, date, and location of valid login
    5. Ability to post items that you want (e.g. physical/knowledge/monetary items that you need)
    6. Ability to bid for items (i.e. to sell) that somebody wants (e.g. physical/knowledge/monetary items that some user needs)
    7. Page listing all the bids for your post (display should conform with Table display requirements in point 9)

8. Search for items that you would like to bid for (display should conform with Table display requirements in point 9)
9. Table display:
    i. Results (with at least four properties) should be displayed in a sortable table (i.e. allowing resulting to be sorted on any column)
    ii. Search results filtering capabilities on at least four result item properties
10. Shopping cart and order purchase submission:
    i. ability to add items
    ii. ability to remove items
    iii. ability to update item counts
    iv. Submission of order purchase request should result in a "purchase order received" confirmation email being sent to the purchaser and the bidder.
    - You are free to use existing third-party libraries to implement the shopping cart
11. Accessible any unavailable page should retrieve a pretty and generic 404 page

E. **Database:** It is mandatory that your project use a database to store all data. There is no restriction on what type of database to use. Any NoSQL database or RDBMS is fine.

The database SQL or ORM request and response information should be available in the Web-Service web/app server logs for the TA to review the implementation of this feature. In addition, the TA might inspect the database's content getting updated via a database SQL console.

F. **Web Services:** Any user operation (on your web site) that requires database access (to retrieve information or add/update information in the database) must be performed via Web Services. These Web Services should be hosted as a different web application and on a different web/application server than the web/application server containing the web site. However, the two different web/application servers can reside on the same machine. Your web site will directly or indirectly consume these web services. For this project, Web Services are platform/programming-language independent, unassociated, loosely coupled units of functionalities that are self-contained and implemented via SOAP/WSDL or RESTful methodologies. **All Web Services should require authentication/authorization for clients (i.e. your web site's server or client) to access a particular Web Service's functionality.**

The web service request, database query, and response information should be available in the both the Website and Web-Service web/app server logs for the TA to review the implementation of this feature. The implementation of RESTful

WebServices and its authentication/authorization feature can also be shown to the TA via browser-based REST clients such as Postman.

G. **Other Required Features**: Your web site/application implementation should also include the following four (4) features:
   1. High Performance: perform distributed caching. Memcached is a good option for implementing a distributed caching mechanism.
      Cache miss and cache hit information should be available in the web/app server logs for the TA to review the implementation of this feature.
   2. Client-Server Communication Encryption: encrypt the communication channel between the client (i.e. browser), web site server, and Web Services server using TLS/SSL.
      The TA will check the implementation of this feature on the Website web/app server by checking if the URL in the browser address bar contains the HTTPS protocol.
      The TA will check the implementation of this feature on the Web-Service web/app server by:
      - **Examining the web/app server logs for the web service request calls being requested and responded to with the HTTPS protocol**

        OR
      - **Making HTTPS calls to the RESTful WebServices using browser-based REST clients such as Postman**

        OR
      - **Examining the capture logs of packet analyzers such as Wireshark**
   3. Request/Response Compression: perform compression (e.g. gzip) of:
      a. web site server's response to the client
         The TA will check the implementation of this feature by looking at the "Content-Encoding" HTTP response header field either in the browser debug console (a.k.a. inspect element console) or in the Website's web/app server log file
      b. web site server's request to the Web Service server
         Optional: The TA will check the implementation of this feature by looking for the "Content-Encoding" HTTP request header field in the Web-Service's web/app server log file
      c. Web Service server's response to the web site's server
         The TA will check the implementation of this feature by:
         - **looking for the "Content-Encoding" HTTP response header field in the Web-Service's web/app server log file**

           OR
         - **looking for the "Content-Encoding" HTTP response header field in the RESTful WebServices call made using browser-based REST clients such as Postman**

Extra Credit Features:

1. Single Sign-On: perform single sign-on using SAML or OpenID/oAuth
2. Object-Relation Mapping (ORM) Framework: perform mapping of object-oriented domain model to RDBMS tables using ORM frameworks such as Hibernate (Java), ADO.NET Entity Framework (.NET), Django (Python), Propel (PHP), etc. **Object serialization/unserialization and SQL query information should be available in the web/app server logs for the TA to review the implementation of this feature**.

C. Project Report

Please write a project report (5 to 10 pages) with the following details:

- An architectural diagram showing how the various components (i.e. client browser, web/application servers, database, cache, etc.) interact with each other in your project
- For each module, a clear description of the various technologies considered and the technology that was finally used in the module development. Also provide a reason why a particular technology was selected
- A clear description of the various functionalities that were available to users on your web site
- A clear description of the Web Services supported by your web application
- A summary of the problems encountered during the project and how these issues were resolved
- Please specify your group name and group member names on the document's cover/start page

D. Project Point Distribution

1. Maximum points available: 100 points
    a. Aesthetics (i.e. look and feel of web application): 5 points
    b. Web site functionality: 30 points
    c. Web Services implementation: 30 points
    d. Other required features implementation: 24 points total (8 points per feature)
    e. Group information: 3 points
    f. Project report: 8 points
2. Extra Credits: 5 points
    a. Single Sign-On: 3 points
    b. Object-Relation Mapping (ORM) Framework: 2 points