CSE 445/598 Assignment / Project 5 (100 Points)

Fall 2013

Part 1 Due: Saturday, November 23, 2013, 11:59pm (Arizona Time),
Part 2 Due: Saturday, November 30, 2013, 11:59pm (Arizona Time). No Grace Period for Part 2.

Introduction

The aim of this assignment is to make sure that you understand and are familiar with the concepts covered in the lectures, including the Web application architecture, components and structure, controls, and state management.

This is a team project with two, three, or four team members. You are expected to use the same team that you used in assignment 3, so that you can continue to use the allocated server site for this assignment.

Although a team works together to complete the project in a collaborative and coordinated manner, a large part of the project has to be done independently. A declaration must be given, which identifies the portion of individual efforts in the joint part of the project. An overall percentage of contribution of each member (e.g., 35%, 35%, and 30%) must be given, which will be used to scale team member's grades of the assignment. The entire project must be deployed into the given server. One member of the team must submit the entire project in a single zip file into the Blackboard submission site.

Exception: For those who did not work well with the other team members in project 3, you may choose to leave your team under the following conditions:

- You declare that: (1) you leave your team, (2) you will not access the WebStrar and V-Lab account of your team. You must send the instructor an email and carbon copy the email to all the team members, and you must place the declaration in the Discussion Board, in the Forum "Team Building" and modify the team members in the Group page.

- The remaining members of a team do not need to make any declaration. You just need to be aware that you will be working with fewer members, and possibly become a one-member team. Please check the discussion board "Team Building" forum to see if any of your team members have left the team in case the email did not land into your email box.

- Those who declared to leave a team are responsible for finding your own server to deploy your application. If you do not have a server, you will lose the points for the deployment.

- The new team may use the services published by its previous team, if the service is publically available.

- The **deadline** for leaving a team and forming a new team is before the due date of Part 1 submission, not including the grace period!

Practice Exercises (No submission required)

No submission is required for this part of exercises. However, doing these exercises can help you better understand the concepts and thus help you in quizzes or exams.

1. Reading: Textbook Chapter 5 and Chapter 6 section 6.2.

2. Do the multiple choice exercises in text Sections 5.8 and 6.5

3. What kinds of Web-based computing models exist? Where is the computation (client side or server side) done in each of the models?

4. Explain how the files in ASP .Net Website application are organized in the application folder.

5. Explain what types of files exist in an ASP .Net Website application and what the functions of each type file are.

6. How do we create a user control, and how do we include the user control into a Web page?

7. What is the most frequently used function of the Global.asax file?

8. What kinds of state-saving mechanism exist, and what are the main features of each kind state-saving mechanism?

9. What is the most general state variable in ASP .Net? What type of data can be stored in this kind of state variable?

10. Compare and contrast the state management mechanisms: View State, Cookie, and Session State variables.

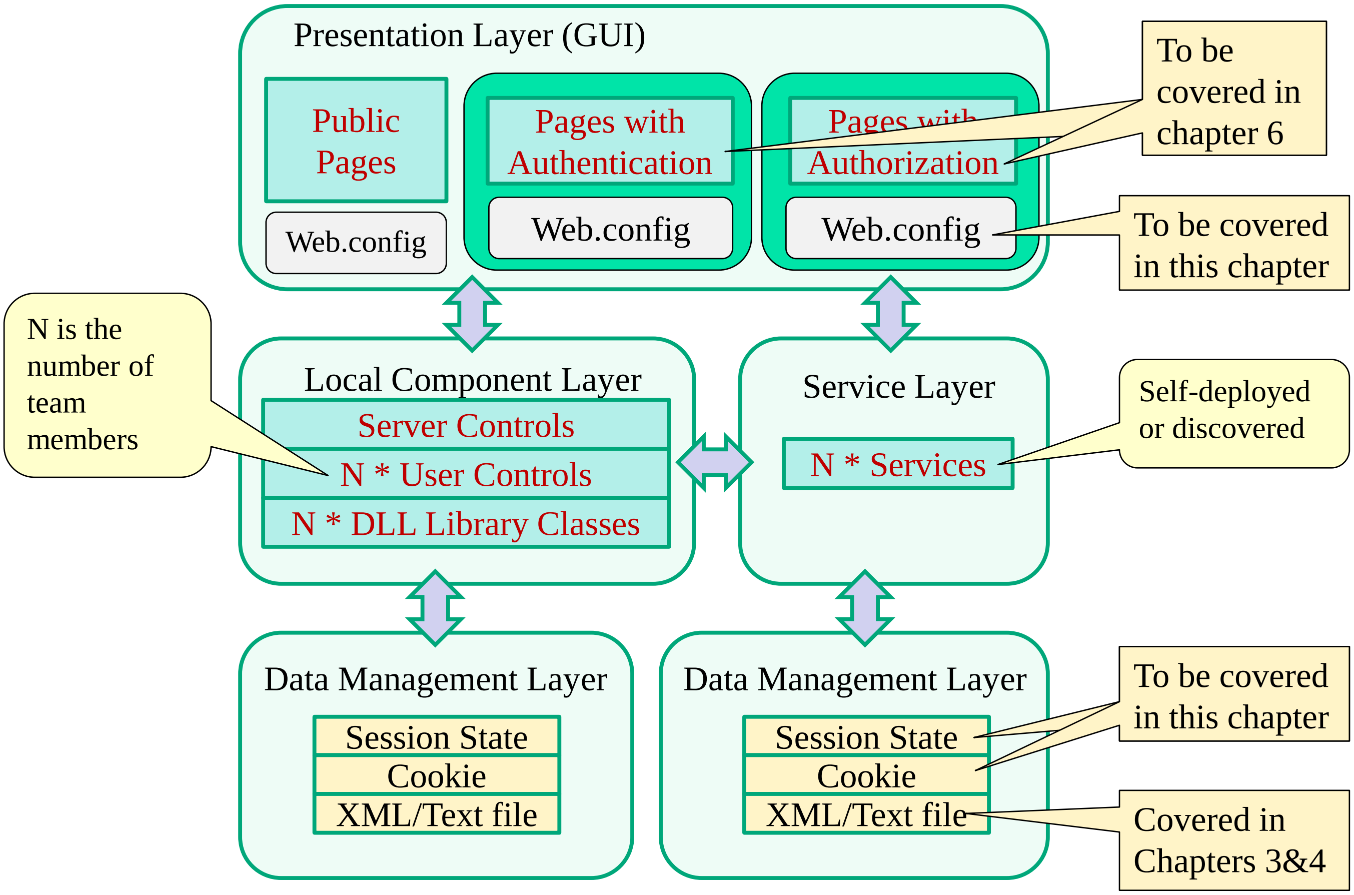
11. How are dynamic graphics generated and maintained in ASP .Net environment? Read text section 5.6.

12. What is the execution model of ASP .Net application in the tightly managed Web server?

13. Study for the questions 2 through 12 in text section 5.8, and study the questions 2 through 15 in text section 6.5. Make sure that you understand these questions and can briefly answer these questions. Study the material covered in these questions can help you prepare for the quiz and exam, and can help you understand the homework assignment.

Project Description (Submission Required)

In this assignment, you will develop a service-oriented Web application. The application must simulate a realistic application for the end users. A sample architecture is shown in the figure below.



The system must be implemented as a **Website** application and must be deployed to a Web server. You may implement the application that you have outlined in project 3. You may choose to implement a different application. The code must be well commented. The application implemented must meet the following organizational, architectural, and functional requirements. The composed Website must have at least the following tiers of components.

1 Presentation/GUI tier consisting of ASPX pages and server controls which allow users to interact with the application. The application must have at least three ASPX pages. Among the three ASPX pages, one page is the public page and the other two pages are password protected using forms security. The two password protected pages must have different authorization privileges, i.e., some people can access one page, and other people can access the other page.
Note, for at least one of the two protected pages, you must use Web.config file to store the user names and passwords. For the other page, you could use the built-in account management class to store the user names and passwords in the built-in database in the Account folder and let users to register and to obtain access to the protected page. You can also use an XML file for storing the account information. [10 points for Part 1 and 15 points for Part 2]

2 Local component tier (individual work), consisting of
(1) ASCX user controls and
(2) DLL class library modules, and each library module must have at least one class and two methods.
This tier must have at least N User Controls and N classes in the class library (DLL files), where N is the number of members in the project team. Each member must be responsible of developing at least one of the user controls and one of the library modules. [20 points for Part 1]

3 Remote service tier (individual work), consisting of sensible Web services developed by each team member and/or discovered from the internet. You must have at least N services and each member is responsible for (developing or using) at least one service. The services can be developed by the team members or discovered from the public repository. At least one service must be developed by the team members and at least one service must be discovered from the public repository. Self-developed services must be deployed into a public-accessible Web server (e.g., WebStrar or v-lab). For the discovered service, an html page must be written to introduce the service with at least following information: Service name, service description, list of operations available, return type and parameter types of each operation, sample code used to invoke the service. This html page must be linked to the TryIt page (See Part 1 Submission Section).
It is the developers' (your) responsibility to make sure that the services are available and reliable when we grade the assignment. [10 points for Part 1]

4 Data management tier (group work), consisting of **both** temporary states (e.g., using session state management) **and** permanent states, e.g., using text file or XML file. Using Web.config file for storing the access control list does not count in this question. The team must implement functions that use:

1) Permanent states (XML file or text file), for example, for storing user data management, catalog, shopping cart, recommendation list; [10 points]

2) Cookie for storing user profile; [5 points]

3) Session state for storing temporary states for sharing among the sessions [5 points]

5 Your application must have the Global.asax file with at least two sensible event handlers or functions in the file. [5 points]

6 Deploy the application to the server. [10 points]

7 **User Manual**
This document must be submitted with your project into the blackboard. It is the starting file of grading. In this document, you must provide the following information. [10 points]

7.1 Description of your project

7.2 A diagram that depict your application and its components. The diagram must be specific to your application.

7.3 The URL of your deployed application in the server.

7.4 Percentage of the contribution to Part 2. The total must add to 100%. You must fill out this table even if the percentages are equal.

Name: ______________________________ Percent _____

Name: ______________________________ Percent _____

Name: ______________________________ Percent _____

Name: ______________________________ Percent _____

The percent of contribution will be used to scale the score of each member in this part.

Part 1 Submission (Individual Work, 40 Points)

Part 1 Due: Saturday, November 23, 2013, 11:59pm (Arizona Time),

Submit questions 1 (10 points), 2 (20 points), and 3 (10 points) into **Blackboard**. The application and components can be deployed on the localhost or deployed into the server.

For question 1, you do not need to include all the functions described in this question. Instead, you just need to include a part of the functions. The minimum requirement is to implement an ASPX page that will invoke the components in questions 2 and 3. You can consider this page as a TryIt page for questions 2 and 3. However, this page must be a part of the overall page that your team will integrate in the Part 2 Submission.

Part 2 Submission (Group Work, 60 Points)

Part 2 Due: Saturday, November 30, 2013, 11:59pm (Arizona Time). No Grace Period.

Integrate all the parts must be zipped into a single file and submit it into the **Blackboard**. Only one team member should submit the project. The entire project must be deployed into the **Server**. We will read the code from your blackboard submission and test your code from the server deployment.

Grading of computer programs

The TA will grade your program following these steps:

(1) The TA will read your program and give points based on the points allocated to each component, the readability of your code (organization of the code and comments), logic, inclusion of the required functions, and correctness of the implementations of each function.

(2) Compile the code. If it does not compile, 40% of the points given in (1) will be deducted. For example, if you are given 20 points in step (1), your points will become 12 if the program fails to compile.

(3) If the code passes the compilation, the TA will execute and test the code. If, for any reason, the program gives an incorrect output or crashes for any input, 20% of the points given in (1) will be deducted.

(4) The TA will test the deployed application in the server assigned to your team.

Please notice that the TA will not debug your program to figure out how big or how small the error is. You may lose 40% or 20% of your points for a small error such missing a comma or a space!

Final Project Submission Policy:

For Part 1 submission, the normal late submission policy applies.

For Part 2 submission, the submission policy applies:

- No grace period;
- Three percent (3%) grade deduction for every hour after the submission due date! Notice that the penalty is 3%, instead of 1% in the previous assignment, as we do not have buffer zone at the end of the semester for any delay. For example, if you submit Part 2 at 10pm on Sunday, you are 22 hours late and your lose 66% of your grade.