Hack Week Project

CS4830
Fall Semester 2016
Assigned Thursday, November 10, 2016
Due Thursday, December 1, 2016 11:59:59 PM

Overview

For this assignment, you are to create a classic 3-tier web application. You must have a database which stores your data. You must have at least one web service that reads and writes data from/to the database. You must have a web application, which can only interact with the user and with the web service layer (it may not interact directly with the database).

You may write your application in any language you choose (e.g., C#, PHP, Javascript, F#, a combination of languages, etc.). The web service should support SOAP, REST, or both.

Detailed Specifications

Basically, there are no detailed specifications. This is an opportunity to express your creativity. Think of an application which fits nicely into a 3-tier architecture, and implement it.

Hints

I suggest you make a simple feature work end-to-end, add another feature, and repeat. That is, don't come up with a grand plan and then get only 1/3 of the way through it so nothing works. Make something simple, and then make that thing better. That way, you will always have something you could hand in if you run out of time.

Web Application Layer

- You could write the entire web application layer client-side, calling the web service via AJAX.
- You could use ASP.NET MVC.
- You could use PHP.
- You could use ASP.NET Web Forms.
- You could use Silverlight, again doing the whole "application" client-side.
- You could use a little client-side stuff with a little server-side stuff.

Web Service Layer

- You could use WCF for the service.
- You could write a RESTful service in PHP.
- You could use ASP.NET MVC.
- You could use ASP.NET Web API.

Database Layer

- You could use SQL Server.
- You could use MySQL.
- You could use Postgres.
- You could use an ORM (like Entity Framework).
- You could use parameterized queries.
- You could build up a SQL string (but watch out for SQL injection).

Grading

Post the URL for your running application to Canvas. Also, upload your source code as a zip file to Canvas.

Your source code should include:

- All client-side code (Javascript, HTML, CSS, Silverlight code, etc.)
- All server-side code (C#, PHP, etc.)
- All service code (C#, PHP, etc.)
- SQL scripts necessary to create the schema (including any indexes, integrity constraints, etc. that you might have)

Your grade will be based on:

1) Having something that works. You must submit a functioning program to get significant points!
2) Implementing the 3-tier architecture properly.
3) The quality of your code. Quality includes an appropriate level of commenting, good understandable names for variables and methods, etc.
4) Creativity – Is the problem you tackled interesting and unusual?
5) Difficulty / Scope – Did you just do the basics or did you try something a bit more difficult with more than a few features?