

CIS 484 Spring 2022 Lab #3 – **VERSION 2**

This assignment is due by **Thursday, March 10th by 11:59 p.m. (Evening)**

**I HAVE ALTERED THE DELIVERABLE REQUIREMENTS BELOW:
HIGHLIGHTED IN YELLOW**

Requirements

This assignment builds on your Lab #2. Be sure to save your Lab2 Visual Studio folder (backup copy) in another location before starting this lab in case any questions come up with your Lab #2 grade. You will use ASP .Net, C#, and SQL Server again. You will reuse the database structure and files from Lab2 and make any NEW database changes needed to implement the listed requirements.

- **Your SQL Server database for this lab should be named “Lab3”.**
- **Your Visual Studio Solution/Project should be named “Lab3”.**

You can work with up to TWO other students **from your section** on this Lab assignment.

The following requirements will need to be met before grading will begin:

- All submissions will include a DukeGroup member login with a username of “**admin**” and password of “**password**”. You will need to create this in its **hashed form** and include this with your test INSERT statements.
- All SQL must be error free and written by you (no tool generated scripts).
- **Include Separate TEXT files with your submission that each contain the following as described – Failure to include any or all of these will result in a resubmission and late penalty:**
 - A Text file containing
 - all usernames and passwords that you would like me to test out. Ideally, this will be “admin”, one Student, and one Member.
 - Short instructions as to which page to run first when I launch your site.
 - Your name and your Partner(s) names clearly stated in the text file.
 - A Text File containing your SQL: CREATE and INSERT statements only. No comments. No DROP statements.
 - A SEPARATE text file containing your AUTH DB CREATE and INSERT statements.
 - A SEPARATE text file for each of your Stored Procedures and SQL Triggers. One per file.
- **NO ID’s will be displayed or required anywhere in the application. This will result in a resubmission and late penalty.**
- **All editing of individual records must be done in a dedicated web form. No editing via GridView or any other “View” control will be allowed (Resubmission and Late Penalty).**

Requirements and Rubric:

Use of parameterized queries wherever input comes from a TextBox to prevent SQL Injection attacks (#6 below is the one exception to this).	5 Points
Use of the PBKDF2 Hashing technique and files to securely store any Passwords.	5 Points
A new Student should be able to register for an account and then log in afterwards. Prompt the user for all the appropriate information, including a password, which you will encrypt.	5 Points
A new Member should be able to register for an account and then log in afterwards. Prompt the user for all the appropriate information, including a password, which you will encrypt.	5 Points
A senior DukeGroup member (for our application: “admin”) will need to approve new accounts created for Students and Members before they are allowed to log in.	5 Points
Store user account information and passwords in a separate database called “AUTH.” I will already have this DB on my machine but it will be empty. You will also provide the SQL statements that will create the tables for this second database.	5 Points
Web.config file contains connection strings for all databases.	5 Points
Use at least one Stored Procedure that is passed parameters from your application to process login attempts. Include this stored procedure in a separate text file (your application will not automatically create the stored procedure). The name of the stored procedure AND the name of the file you list it in will be “Lab3StoredProcedure”. Make sure you use this name in your application code when calling it.	5 Points
Use proper encoding to prevent Cross-Site Scripting for TextBoxes.	5 Points
Give students the ability to upload a PDF Resume to the site, to replace the resume with an updated one, and to view it in their account.	5 Points
Give DukeGroup members the ability to search for a Student, to view their details, and view their most recently uploaded resume file.	10 Points
Total:	60 Points

Quality of Code

I will judge your programs both by how well they work and based on the elegance of the code. The code should be well documented and efficient. Proper indentation, descriptive variable names, and comments are essential. No extraneous variables or extra unnecessary lines of code. You must include your name near the top of all of your files. I will use the above grading rubric when grading your files (it serves as a checklist for all functionality that must be present).

Deliverables

- A folder, zipped, containing:
 - Your solution folder

- **A single text file with the SQL to CREATE and INSERT the main DB tables and their test data.**
- **A second single text file with the SQL to CREATE and INSERT your AUTH db and its table(s) data.**
- A third text file, named “JeremyEzellLab3” containing your stored procedure by the same name.
- Text files containing any other Stored Procedures or SQL Triggers as discussed above.
- Another text file letting me know what your start page is, login information (should be “admin” and “password” as seen in the initial requirements above), and any other usage information you think I should know, as discussed above.