CSI3670 // Winter 2019
Term Project Requirements

This document outlines the requirements for your final project. Overall the project is worth **150 points.**

As a reminder, you are required to provide a **software-as-a-service**. I will leave it to you to define how your project exactly fits this definition – don't worry, it can be flexible.

Al project materials will be due on 04/17 (last day before finals officially begin).

SUMMARY OF IMPORTANT DATES

Presentations:	**04/11, 04/16**
Project report due:	**04/17**
Project deliverables due:	**04/17**

SERVICE: 75 points

Your service must be minimally working by the time of the final presentation, and fully working when the project report is due. To avoid confusion, here are the definitions **I** will be using to quantify these terms.

Minimally working: able to demonstrate the core concepts of the service
Fully working: project is functioning as specified, including single-sign on

This is the most important part of the project, so make sure it is functional.

(Advanced) Requirement: Implement Single Sign-On: 25 points (of the 75)
Example: Your website login and service login are tied to a single user account, either through Active Directory, OpenID, or another similar technology. Note: Active Directory SSO will need to be cleared with me as special permissions will need to be given.

Now I don't care how you implement SSO, as long as you can make the claim that you did it. If you're installing a forum, see if it supports LDAP/AD/OpenID. Or, create a central database table and allow a user to login to that for your implementation. No hard requirement other than it is there in some form.

PROJECT PRESENTATION: 25 points

The last two classes of the semester will be team project demonstrations. These should run at a maximum of 10 minutes.

You must:

1) Introduce your team

2) Introduce your project

a) Describe what it is
b) Describe any technologies you used
c) Overview your server environment
d) If applicable, how to connect to the service

3) Demonstration

a) You must have a minimally-working demonstration. At the very least you should have a video of your project functioning. Failure to show what your service does will result in a fairly hefty points loss here.
b) Ideally, your project should be **accessible** by the class. This is not a requirement, but heavily suggested if your project is hosted on the OU VMs.

4) Q&A

Make sure you leave a couple of minutes for us to ask you questions

PROJECT WEBSITE: 50 points

Your website must be **database-driven**, either by Microsoft SQL, MySQL, MariaDB, or some other database. It must be accessible outside of the VPN (meaning you need to open up the appropriate ports on your server).

Please note that only providing a basic website will get you half the points here (so your lab is intended as a starter, not the whole aspect).

1) Website basically functional (e.g., you did the Wiki lab) – (10 points)

If you decide to go the Ubuntu route, you could install something like WordPress, Drupal, etc.

2) Content (30 points)

Consider that the content of your website should be similar to that found in a term project report. As such, your site must have pages dedicated to the following topics:

a) Project overview, including the URL of where somebody could find the service themselves, and how it classifies as a software-as-a-service.
b) Screenshots and/or videos of your project in action (whatever best demonstrates it)
c) Your team members and their roles
d) How to access your project
e) The steps one would need to take to setup and run your project on their own (i.e., what are the steps you took to set it up and run it)

3) Readability / Grammar (10 points)

Your site must be readable and free from spelling/grammar errors. I will be reviewing this point as I would a technical report. The whole site does not to be prose, but it should look like a reasonable website.

Additionally, a separate Moodle assignment will be given that requires you to evaluate yourself and your team members' contributions to the project. This will be a separate score that I use to factor everybody's grades. It will be subjective, based on your reports and my observations.

Possibilities for extra credit:

The following identifies some possibilities for going above and beyond. If you have another idea you must first **clear it with me**, otherwise I will consider it to be a normal aspect of your project. To clear an extra credit suggestion you must send an email to ensure that there is a paper trail of your request.

Possibility 1) Implement failover clustering
Example: Your service is clustered amongst two nodes and you take Node 1 down, with automated failover to Node 2 to demonstrate no interruption of service.

Possibility 2) Implement community interaction
Example: You install SharePoint, you install a forum, etc. Pointswise, SharePoint > forum.

Possibility X) Something above and beyond – discuss with me first.