Database Programming Phase 3

**PostITT**

**Introduction**

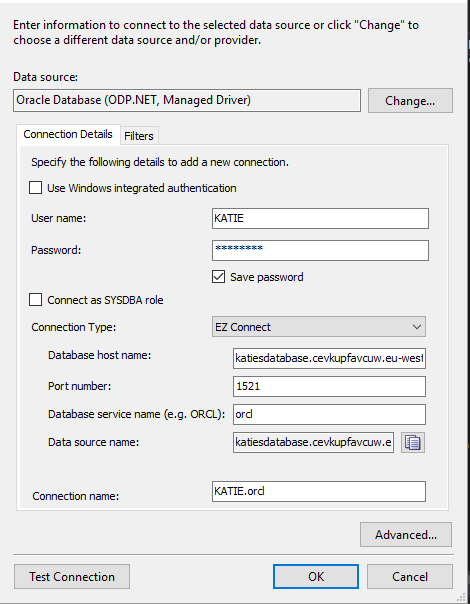
For this phase of the group project we each had to design and implement working UI’s. When discussing with framework to use each member of the group had their own decision to do between Visual Studio C# and PHP. The design for the database was to create and use our own instances on amazon cloud. This gave us the chance to be able to use the tables/procedures/triggers without having the worry of effecting anybody else’s work. We all still worked as a team when creating those instances and discussing how to set up the basis of the project.

When I started the research for phase three of the project, I had made a decision to use PHP and do the project using Larval and AWS. After sometime I had major difficulty in getting the oracle PHP extension working in windows, I felt that this trial and error approached would likely have to be repeated to get the project working for examination and was therefore not suitable. I came to the realisation that I wouldn’t be able to use PHP so I had to resort to another database implementation. My choice in the end was to use Visual C# with Entity frame work and AWS for my database.

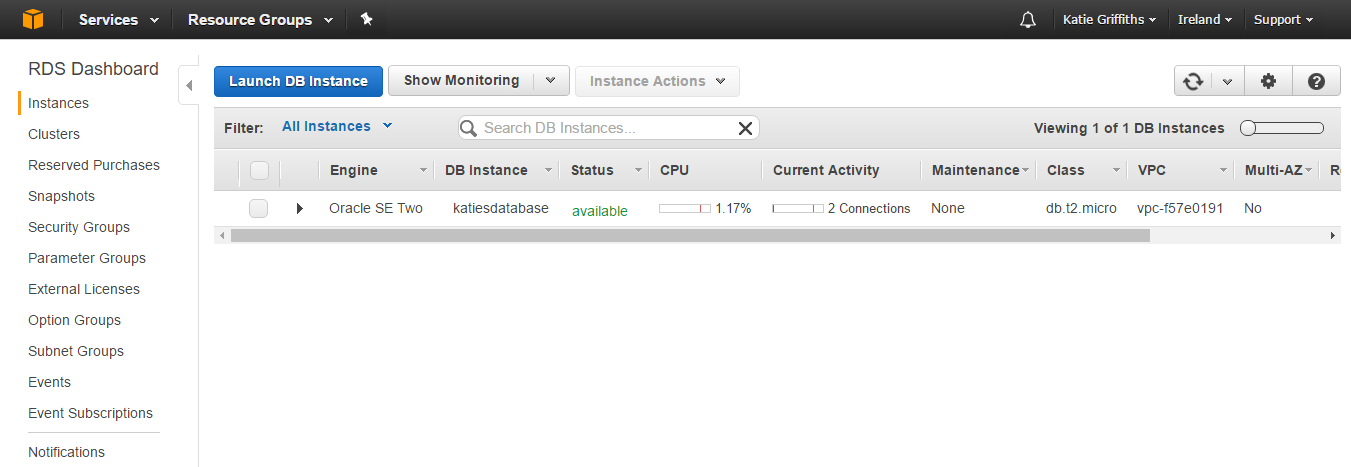
Having used Visual Studio and oracle in a previous semester I was able to move around the framework and quickly design forms. These forms that I have designed where kept close to the design mock-ups in which I made in phase two of the project. There were slight alterations in the designs that I had to change due to Visual not having the properties available and also that our database did not have the functions or procedures available.

While using Visual studio I researched up on LINQ and I found it easier to implement with the procedures and triggers.

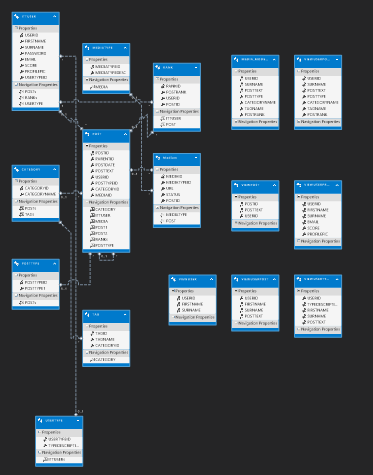
**Connecting to the database:**



Connecting Visual Studio to AWS Database.

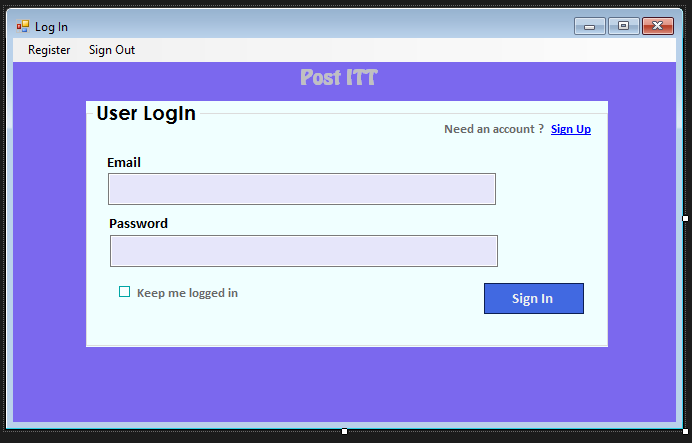


**Entity frame work**



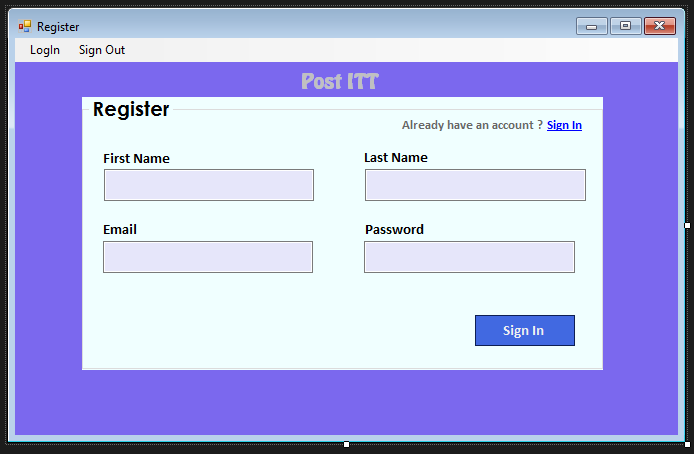
UML Diagram auto generated in Visual Studio when tables, views, indexes, procedures and triggers are brought in and set up.

**UI design**



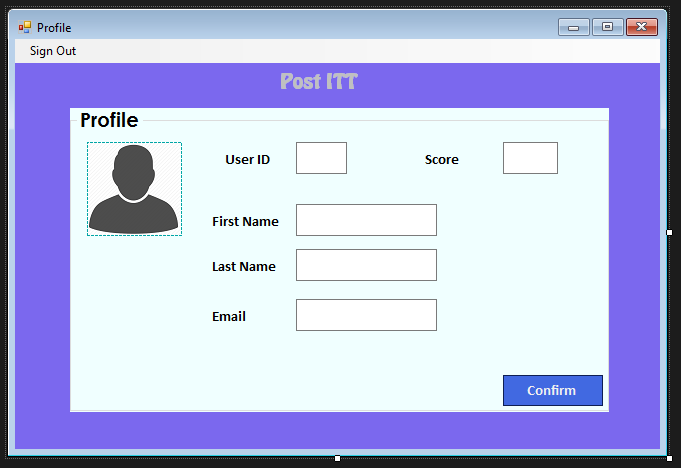
This form was generated in Visual studio, I wanted to keep the UI design as close as possible to the sample UI’s that I made in phase two of the group project. This form allows the user to fill in their email and password that they registered with once the information has been entered, the user will click on the “Sign In” button that will call the procedure to Login.

The form also allows the user to register for an account by clicking on the “Sign Up” link. This will then take the user to the register form.



This Register form was generated in Visual studio, I wanted to keep the UI design as close as possible to the sample UI’s that I made in phase two of the group project. This form allows the user to fill in their First name, last name, Email and password to generate an account for PostITT. Once the information has been entered, the user will click on the “Sign In” button that will call the procedure to create a user. If the user enters an invalid email a trigger is called and the user must enter in a valid email to proceed.

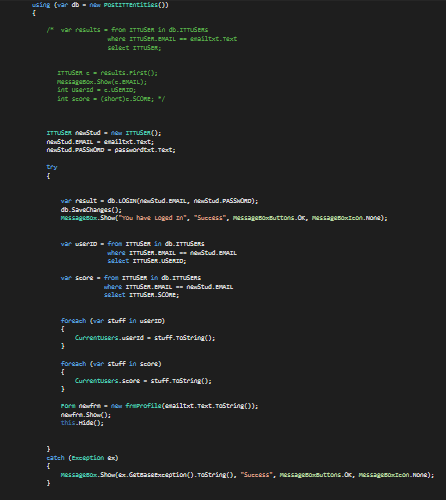
The form also allows the user to Login to their account by clicking on the Login in the menu bar. This will bring the user to the login form.



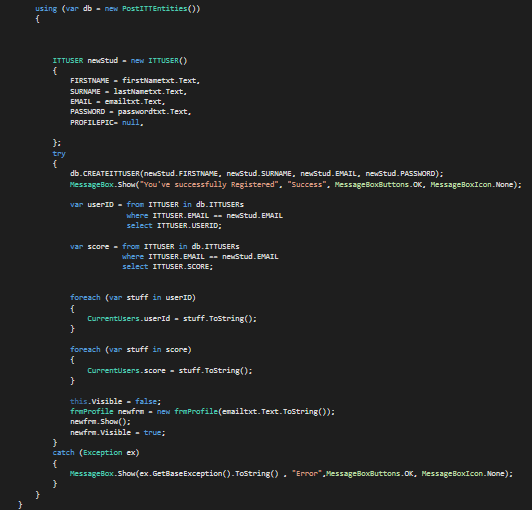
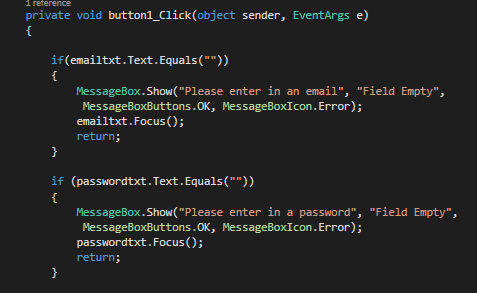
This profile form was generated in Visual studio, I wanted to keep the UI design as close as possible to the sample UI’s that I made in phase two of the group project, and this was a problem when designing the profile UI. This form allows the user view their information that they initially registered with. The form displays the users ID that was generated by a sequence, There score from posts that they had been scored on by generating a post and that post being given an up vote, their first name, last name and email. Once the information has been displayed I was hoping for the user to be able to edit there name or email by calling an update function, once that had been done by the user they would select the confirm button and that would then alter their information in the database.

The form also allows the user to sign out by clicking on the sign out at the menu.

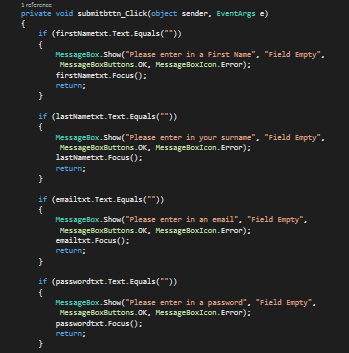
**Code in visual Studio**



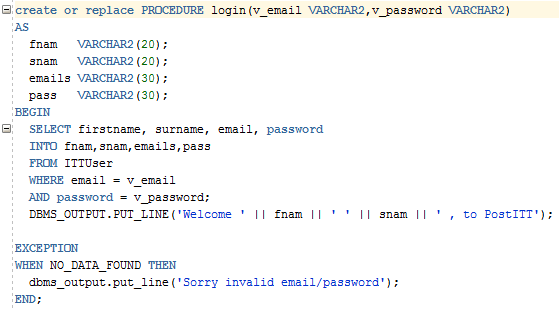
Login procedure being called using LINQ in visual Studio



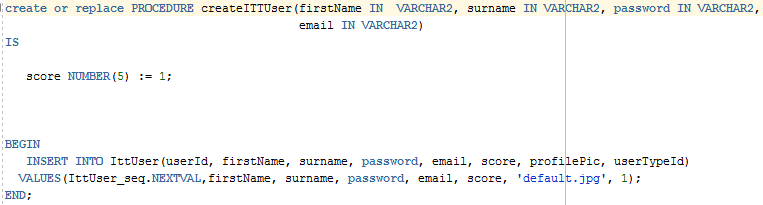
Registering a user by calling the create itt user procedure also validation is set up to check that all the fields have been entered with information.



**Procedures being used**

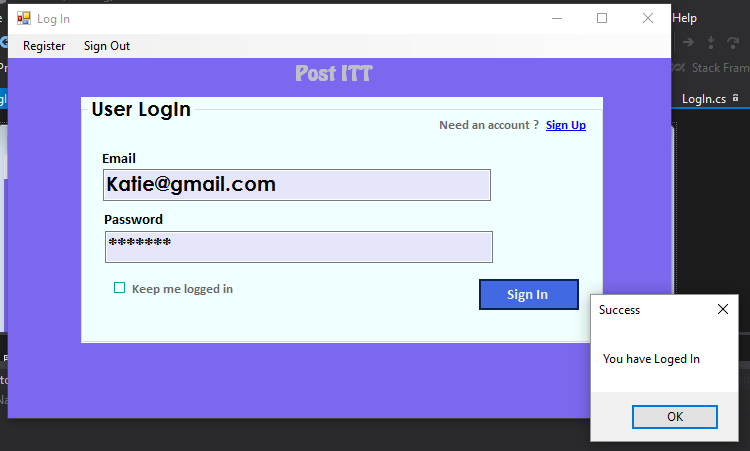
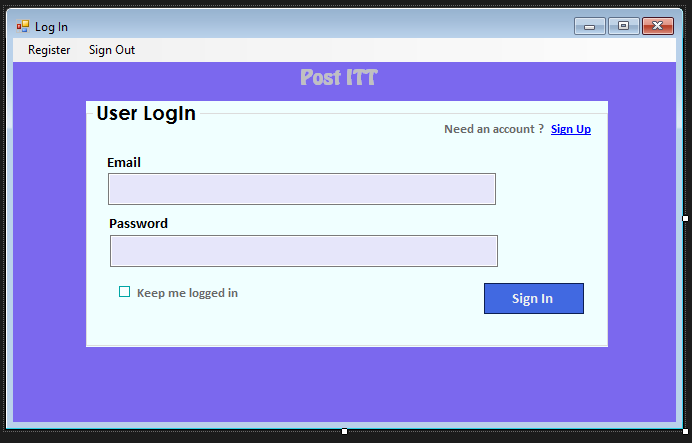


Login Procedure being called in visual studio

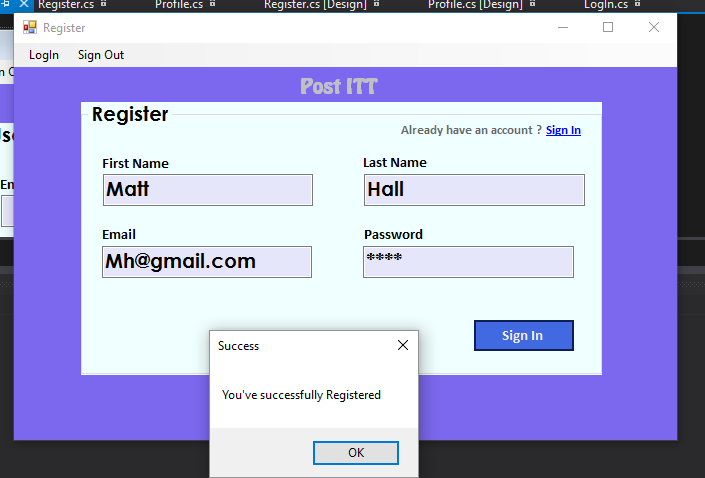


Creating a user procedure being used to register a user in visual studio.

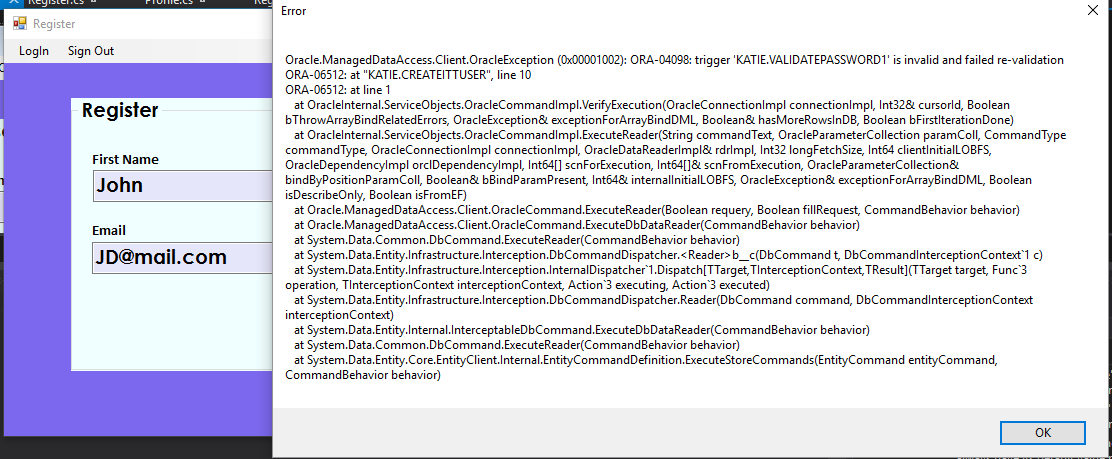
**UI’s running in Visual Studio**



Successful login procedure being called



Successful Registration calling the CREATITTUSER procedure from the database.

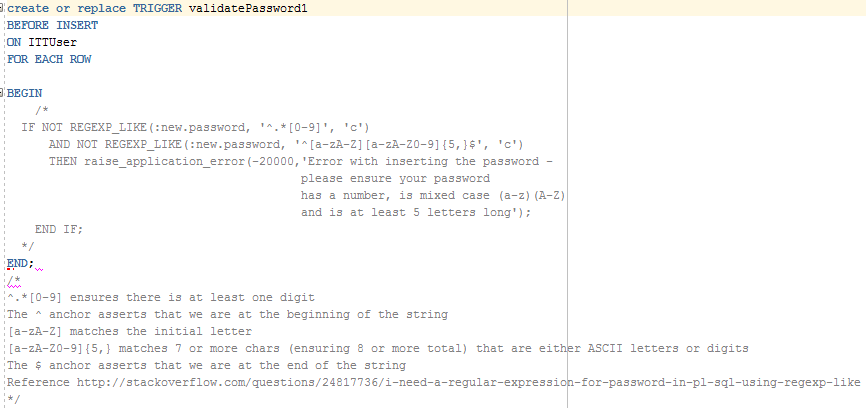
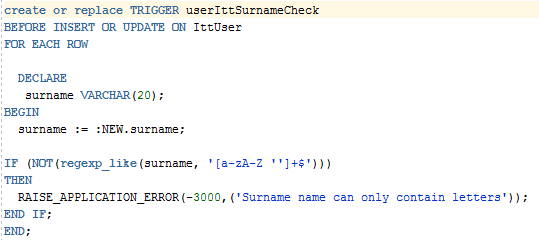


Trigger valid email being called when registering a user.

**Complications along the way**

During this project I had complications with triggers being called when the procedures where generated. The trigger for surname and password both had to be disabled for me to run the forms in visual studio.

Triggers that had to be disabled:



**Conclusion**

During the time of this project there were many challenges that were faced in trying to complete the project successfully. While each team member used their own Amazon web server, there were a few changes needed for triggers and procedures to work. Some of these became complicated to get right and some were not able to be implemented in the project. While using the same software as a previous year it gave great advance to being able to more quiet quick through the software to design and implement the forms, though it was a different challenge when coding to implement the database as well as generating the procedures and triggers.

The overall development of the project was great to see what the project looks and does physically, however I do wish I had more time to implement the project as a whole.

**References**

FALSE IS NOT NULL. (2016). *Validating data using a TRIGGER*. [online] Available at: https://falseisnotnull.wordpress.com/2012/11/06/validating-data-using-a-trigger/ [Accessed 8 Dec. 2016].

form, S. (2016). *Send values from one form to another form*. [online] Stackoverflow.com. Available at: http://stackoverflow.com/questions/1559770/send-values-from-one-form-to-another-form [Accessed 8 Dec. 2016].

Msdn.microsoft.com. (2016). *Stored Procedures*. [online] Available at: https://msdn.microsoft.com/en-us/library/bb386946(v=vs.110).aspx [Accessed 8 Dec. 2016].

Weblogs.asp.net. (2016). *ScottGu's Blog - LINQ to SQL (Part 6 - Retrieving Data Using Stored Procedures)*. [online] Available at: https://weblogs.asp.net/scottgu/linq-to-sql-part-6-retrieving-data-using-stored-procedures [Accessed 8 Dec. 2016].

www.tutorialspoint.com. (2016). *LINQ - Entities*. [online] Available at: https://www.tutorialspoint.com/linq/linq\_entities.htm [Accessed 8 Dec. 2016].