**422 Application Development Practices Final Group Project**

Your team has been asked to take over a project for a programmer who has left the company. The application currently reads in a file that is created by Edge Diagrammer (a data modeling tool) and generates the SQL statements required to create the tables for a MySQL database. Your general overall task is to evaluate the application, refactor the code and make it as reusable and extensible as possible.

There are two goals for the finished application:

1. Have the application be able to generate the appropriate statements for any number of database management systems (e.g. MySQL, Oracle, etc.) for the current Edge Diagrammer file with minimal changes to the code. You need to think of how you can structure the code and provide it with the information it needs to create the tables on a different DBMS without major rewriting of the code. This will allow the company to switch vendors with minimal effort.
2. Have the application be able to replace Edge Diagrammer (it is expensive) with some other tool or schema description file (e.g. XML) and perform the same functions, again with minimal changes to the application itself the same as for the first objective.

These goals should be achieved by using any number of the methods we discussed or will discuss in class. You need to use a version control system for all of your work, including any documents that are required. **Host this on BitBucket and add me with admin rights as in the version control exercise.**

**Deliverables**

The following deliverables are due by the due dates on the corresponding dropbox:

1. (20%) A testing plan, test cases and test code (JUnit). This test plan and code will be done prior to receiving any code. What you are testing is given a specific input file (Edge Diagrammer, XML, etc), the application will generate an expected output file (the DDL for what the database is)
2. (15%) A description of what type of SDLC you plan on using and **why**. Include in this any milestones, backlogs, etc. that are required for the initial phases of your chosen SDLC. **Also include a FlowChart of what the code does.**
3. (45%) Refactored, abstracted and refined code to achieve the goals above. Include with the code a separate description of what you changed and **why**. Also include how your code solves the goals above – in other words, what you would have to do to use a different DBMS or modeling program’s file or some other description file.
4. (5%) Add some sort of help system.
5. (10%) A Deployment strategy and related deployment packaging.
6. (5%) Peer review including a self review.