**Student Grading Management Sub-System**

For each subject that attended by the student, the lecture will give score to the assessment to each of their assessment. Below figure shows an Example of the assessments for course DBI202.



Students can check their results at the end of semester as following example:



Each Subject code, student can check their detailed result of as below example:



**2> The database must consist of at least six tables** that have been populated with data. The database is to support queries that would typically be submitted to the system for the topical area that you have chosen. You must do the following:

Self-investigation for the requirement of the system. Listed them all as form of reports, business rules.

* Using UML, Chen’s notation to create an Entity Relationship (ER) model for your relational database. All entity types, their attributes and relationships must be clearly shown. You will also be required to show all cardinality and participation constraints. You should use some enhanced ER features in your conceptual model where it makes sense to do so.
* Map the EER model devised in part (1) into a relational data model. It must be normalised up to at least 3rd Normal Form.
* Using appropriate SQL commands create a set of database tables in MS SQL Server 2008+. You should also show all constraints used in the creation of the tables.
* Populate the database with a small amount of data. The data should be meaningful but does not need to be extensive. The following sites may be useful for quickly generating data:
  + <http://www.databasetestdata.com/>
  + <http://www.generatedata.com/>
* Your database must contain one view, one trigger, on store procedure and an index (describe why).
* Create 10 sample queries that demonstrate the expressiveness of your database system. Your queries must demonstrate different aspects of the system.

**Final Report**

You must submit a brief final report which must include the following:

a)      A brief description of the database including any assumptions made during the design (THIS IS VERY IMPORTANT TO CLERIFY THE ASSUMTIONS in form of business rules).

b)      An ERD (Entity Relationship Diagram) that fully describes the database (giving descriptions on your work would be appreciated).

c)       The relational schema derived from the ERD that is at least in 3NF (Any detail of the process would be appreciated).

d)      The set of database statements used to create the tables used in your database. You do NOT need to include all the data and insert statements.

*e)*      *10 queries that demonstrate the usefulness of the database. Also state why and when each query would be used. The following must be demonstrated by at least one of your queries:*

o   A query that uses ORDER BY

o   A query that uses INNER JOINS

o   A query that uses aggregate functions

o   A query that uses the GROUP BY and HAVING clauses

o   A query that uses a sub-query as a relation

o   A query that uses a sub-query in the WHERE clause

o   A query that uses partial matching in the WHERE clause

o   A query that uses a self-JOIN

*f)*        *The trigger, store procedure, and the index should be added (explain why you make it)*

**Demonstration**

You will be required to briefly demonstrate your system in one of the laboratory sessions prior to submission of the report.

Relational Mapping

Step 1: Tìm normal entity, với mỗi entity tạo 1 table tương ứng với attribute tương ứng

Step 2: Tìm tất cả các quan hệ 1 – nhiều. Đối với table side detail, thêm 1 khóa ngoại (foreign key) tham chiếu (reference) tới khóa chính bên side master

Step 3: Tìm tất cả các quan hệ nhiều – nhiều. Tạo bảng tương ứng cho mỗi thực thể tham gia. Tạo bảng trung gian ứng với mỗi quan hệ. Khóa chính của bảng trung gian là tổ hợp khóa (composite key) của các bảng liên quan. Mỗi key attribute đều là khóa ngoại tham chiếu tới primary key của bảng liên quan. Thuộc tính của các bảng liên quan cũng trở thành thuộc tính của table trung gian

Step 4: Có 2 option

Option 1: Không lựa chọn để lưu lại dữ liệu lịch sử(historical data). Tìm side mandatory và triển khai như quan hệ 1 – nhiều, side mandatory là side nhiều. Nếu 2 bên cân bằng, chọn bên nào cũng được.

Option 2: Triển khai giống mqh nhiều – nhiều nhưng thêm các mốc thời gian quan trọng

Step 5: Tìm tất cả weak entity, triển khai table tương ứng, với khóa chính của table là tổ hợp giữa key attribute và khóa chính của table master

Step 6: Tìm tất cả multiple part attribute, chia thành các column ở cấp nhỏ nhất

(Name->FirstName,LastName)

Step 7: Tìm tất cả multiple value attribute, triển khai 1 table mới, khóa chính của table này là tổ hợp key giữa trường value và khóa chính của table gốc

(Department(dnum,dname, location)-> Department(dnum,dname)+Department\_location(dnum,location)