



**Department of Information and Communication Technology**

**Faculty of Technology**

**University of Ruhuna**

**Database Management Systems**

**ICT 1213**

**Assignment 02**

**Group 02**

**Submitted to: Mr.P.H.P. Nuwan Laksiri**

**Submitted by:**

Joseph Rasanjana

TG/2019/525

Umeen Rathnayake

TG/2019/516

Niraj Dilshan

TG/2019/491

Waruna Dissanayake

TG/2019/515

# Table Of Content

Brief introduction about the problem/group project .....	1
Brief introduction to the solution .....	1
Proposed ER/EER diagram.....	2
Proposed Relational mapping diagram .....	3
Table structure of your solution .....	5
Architecture of your solution .....	8
Tools and technologies that you have used .....	8
Security measures that you have taken to protect your DB .....	8
Brief description about DB Accounts/Users and the reasons for creating such Accounts/Users .....	8
Code snippets to support your work .....	9
Problems that you faced during the development of the solution .....	10
Solutions/how you have overcome the above identified problems .....	10
New database technologies/trends that you have used to develop the backend.....	11
If you are going to host your backend where are you going to host it and reasons for the selection.....	11
If you are going to host your backend in a cloud environment what are the things/changes that you have to do in your backend .....	12
Individual contribution to the backend development .....	12
References .....	13

## **Brief introduction about the problem/group project**

We are required to develop a database for the faculty of technology for recording student marks and attendance. Our system should include user accounts like Admin, Dean, Lecturer, Demonstrator, Technical Officer and Student. Our system should include tables to store Quiz marks, Assessment Marks, Mid Semester marks(Practical and Theory), Final Marks(Practical and Theory) etc. Our system should include tables to store data about course modules. Our system should include tables to store data about all the users Admin to Student. We need to go through the DDLC process and produce all the related documents for the final presentation (Requirement document, ER diagrams etc). We should store data for at least 10 students, at least 04 lecturers, at least 04 instructors, at least 04 technical officers and other mentioned roles. We should store basic data for above roles including at least 04 different attributes. We should record attendance details for 06 subjects within 15 days.

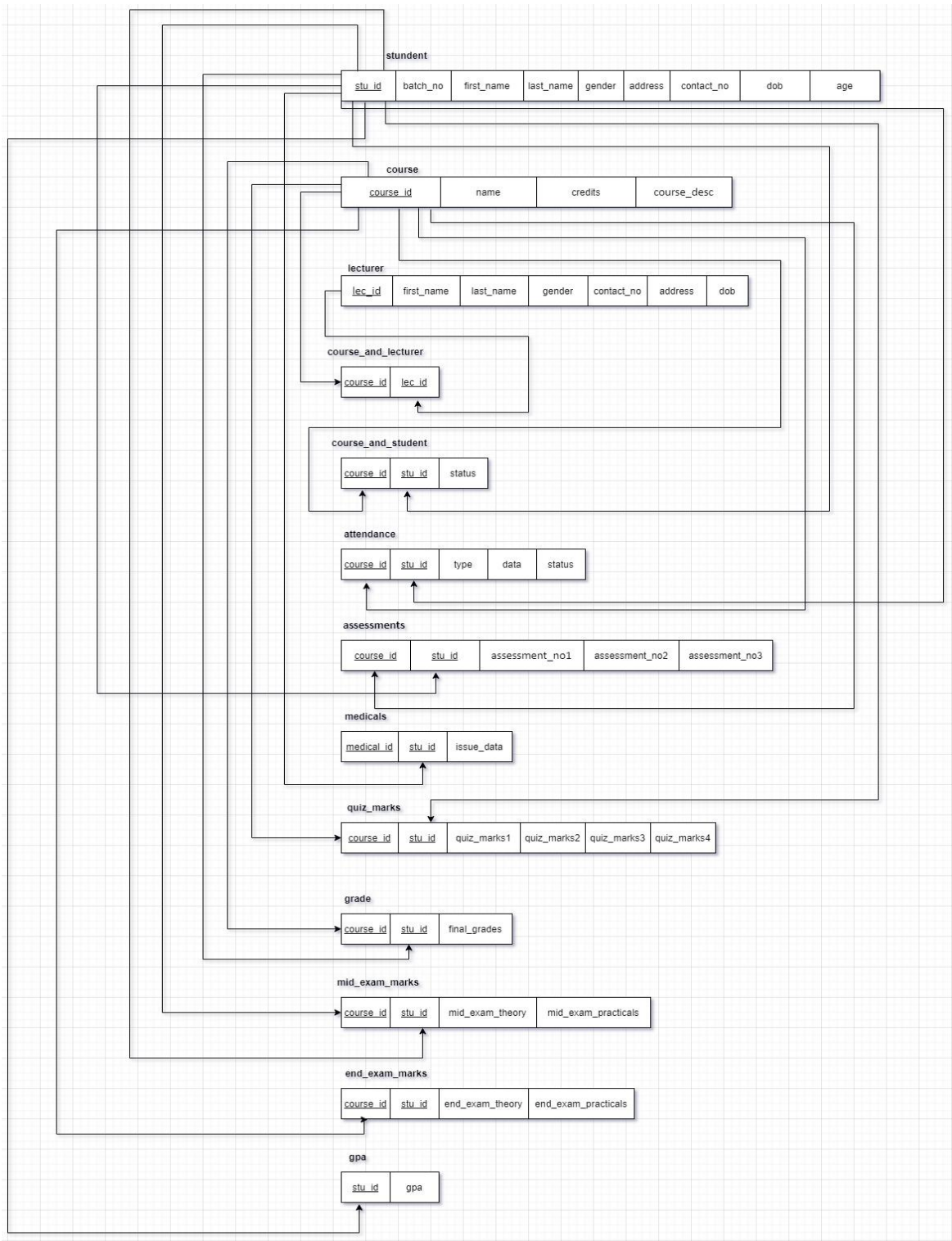
## **Brief introduction to the solution**

We are Building a database to address the problems introduced in the project. This solution database includes, user accounts like Admin, Dean, Lecturer, Demonstrator, Technical Officer and Student, tables to store Quiz marks, Assessment Marks, Mid Semester marks(Practical and Theory), Final Marks(Practical and Theory), tables to store data about course modules, tables to store data about all the users Admin to Student.

### Proposed ER/EER diagram

<https://app.diagrams.net/#G1exfX2gSDh2V5OuMR4Ssj3aZ6WFFOMLhm>

# Proposed Relational mapping diagram



[https://drive.google.com/file/d/10vB\\_-  
Wp\\_doo9x92M0atqwj8eIsrvckE1/view?ts=60dff4a6](https://drive.google.com/file/d/10vB_-Wp_doo9x92M0atqwj8eIsrvckE1/view?ts=60dff4a6)

## Table structure of your solution

According to the ER diagram, we prepared the structure of the table. Considering the table structure we create the table with all details that provide the guideline of the project. We created 25 tables including views and the base tables of project.

Here are the tables that we created

1. admin
2. assessment\_top\_marks
3. assessments
4. attendance
5. attendance\_eligibility
6. ca\_eligibility
7. course
8. course\_and\_lecturer
9. course\_and\_student
10. demo
11. end\_exam\_marks
12. final\_grade\_temp
13. final\_grade\_temp\_scaled
14. grades
15. lecturer
16. medicals
17. mid\_exam\_marks
18. mid\_exam\_overall\_marks
19. quiz\_mark\_avgs
20. quiz\_marks
21. sgpa
22. sgpa\_temp
23. student
24. student\_sgpa
25. T\_o

These are our base tables

- admin
- assessments
- attendance
- course
- course\_and\_lecturer
- course\_and\_student
- demo, end\_exam\_marks
- lecturer
- medicals
- mid\_exam\_marks
- quiz\_marks
- student
- student\_sgpa
- t\_o

And these are views table

- assessment\_top\_marks
- attendance\_eligibility
- ca\_eligibility
- final\_grade\_temp
- final\_grade\_temp\_scaled
- grades
- mid\_exam\_overall\_marks
- quiz\_mark\_avgs
- sgpa
- sgpa\_temp

Here is one example for the base table and view table, consider the assessments table this is the table we created to enter assessments details. Attributes are of this table stu\_id, course\_id, assessment\_1, assessment\_2, assessment\_3. Then we calculated the top marks of the assessment and created a view for it called “assessment\_top\_marks”. The assessment\_top\_marks table attributes are stu\_id,



course\_id, max1, max2. We got the maximum 2 values from the assessments table and created a view with stu\_id, course\_id, max1, max2.

## **Architecture of your solution**

We have hosted a MySQL Community Server on a VPS(Virtual Private Server) and started to develop a solution for the problem through that.

## **Tools and technologies that you have used**

MySQL WorkBench And MySql command line client We have used for creating databases, creating users, inserting data to the table, backing up the database.

## **Security measures that you have taken to protect your DB**

We have added a firewall for our VPS to protect the database.

We have added different access levels for different users.

We have added strong passwords for user accounts.

## **Brief description about DB Accounts/Users and the reasons for creating such Accounts/Users**

DB Accounts – One account for use the database (root)

### **User Accounts**

1. Admin- - With All privileges with Grant
2. Dean- With All privileges without Grant
3. Demonstrator – Read for all tables/views
4. Lecturer - All privileges without Grant and user creation
5. Student- Read permission for final attendance and final marks/Grades tables/views
6. Technical Officer- Read, write and update permissions for attendance related tables/views

## **Code snippets to support your work**

```
CREATE DATABASE db_name;
```

```
CREATE USER 'user'@'localhost' IDENTIFIED BY 'password';
```

```
GRANT ALL PRIVILEGES ON DATABASE.TABLE TO 'user'@'localhost';
```

```
CREATE TABLE table_name (column_name column_type);
```

```
INSERT INTO table_name ( field1, field2,...fieldN ) VALUES ( value1,  
value2,...valueN );
```

```
SELECT field1, field2,...fieldN FROM table_name1, table_name2... [WHERE  
Clause] [OFFSET M ][LIMIT N];
```

```
UPDATE table_name SET field1 = new-value1, field2 = new-value2 [WHERE  
Clause];
```

```
DELETE FROM table_name [WHERE Clause];
```

```
CREATE VIEW `view_name` AS SELECT statement;
```

```
LOAD DATA INFILE 'path' INTO TABLE tablename FIELDS TERMINATED  
BY ',' LINES TERMINATED BY '\n' (columns);
```

Python program to create dummy data,

```
#!/usr/bin/env python
# coding: utf-8
# In[10]:
import random as r
# In[11]:
COUNT = 10
LECS = ["ICT1113", "ICT1123", "ICT1133", "ICT1143", "ICT1213", "ICT1232"]
# In[12]:
# for l in LECS:
#     for i in range(COUNT):
#         print(f"TG/2019/{400+i}", "{l}", {round(r.uniform(30.0, 100.0), 2)},
# {round(r.uniform(20.0, 85.0), 2)}),', end="")
# In[13]:
for l in LECS:
    for i in range(COUNT):
        print(f"TG/2019/{400+i}", "{l}", T, 2020-06-0{2+i}, {r.randint(0,1)}),', end="")
# In[ ]:
```

## **Problems that you faced during the development of the solution**

1. Creating an enormous amount of dummy data.
2. Adding an enormous amount of dummy data.
3. Limitations in creating the database as a group.

## **Solutions/how you have overcome the above identified problems**

1. We created a python script to randomize and create data And inserted them into a csv file.
2. We used the SQL load file to add those data in the csv file through the queries.
3. We hosted the database and logged into it through the Secure Shell(SSH).

## **New database technologies/trends that you have used to develop the backend**

We use the latest MySQL workbench for creating the database. MySQL workbench is tool for is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system. When design the database we use some few functionalities of MySQL workbench such as

1. SQL Editor
2. Server Stop/Start
3. Import/Export data

And also we setup a server for hosting the database and give the access for our four members to login the database and create tables, insert data to tables and other all functions in MySQL. When backup the database we simply used the backup option in the MySQL workbench. We use the MySQL for the development the database nowadays without NoSQL database management systems also available such as mongo DB.

## **If you are going to host your backend where are you going to host it and reasons for the selection**

We have used a cloud based host service called DigitalOcean to host our database. The reasons are listed below.

1. Allows to create multiple server instances using the same account
2. Offers great performing servers
3. Easy to set up and it is friendly to various operating systems.
4. User-friendly management interface
5. Has FAQ and tutorials to help you out
6. Great tool for tech-savvy start-ups.

## **If you are going to host your backend in a cloud environment what are the things/changes that you have to do in your backend**

We have already hosted our database in a cloud-based host service so no necessary changes or things to do in our database.

### **Individual contribution to the backend development**

Joseph Rasanjana                      -              TG/2019/525

- Create Tables
- Data Entry
- Create Views
- Create dummy data script

Umeen Rathnayake                      -              TG/2019/516

- Create Users
- Grant User Permissions

Niraj Dilshan                              -              TG/2019/491

- Create Tables
- Setup Database in cloud
- Data Entry

Waruna Dissanayake                      -              TG/2019/515

- Data Entry
- Verify Table Content

## References

- [http://teclms.ruh.ac.lk/pluginfile.php/18848/mod\\_resource/content/0/Assessment%202002%20-%20DBMS%201213%20-%20B4.pdf](http://teclms.ruh.ac.lk/pluginfile.php/18848/mod_resource/content/0/Assessment%202002%20-%20DBMS%201213%20-%20B4.pdf)
- [http://teclms.ruh.ac.lk/pluginfile.php/20282/mod\\_resource/content/0/ICT1213%20-%20DBMS%20-%20Assignment%202002%20-%20Report%20Guidelines.pdf](http://teclms.ruh.ac.lk/pluginfile.php/20282/mod_resource/content/0/ICT1213%20-%20DBMS%20-%20Assignment%202002%20-%20Report%20Guidelines.pdf)
- [http://teclms.ruh.ac.lk/pluginfile.php/17672/mod\\_resource/content/0/Practical%2001.pdf](http://teclms.ruh.ac.lk/pluginfile.php/17672/mod_resource/content/0/Practical%2001.pdf)
- [http://teclms.ruh.ac.lk/pluginfile.php/17972/mod\\_resource/content/1/Practice%2002.pdf](http://teclms.ruh.ac.lk/pluginfile.php/17972/mod_resource/content/1/Practice%2002.pdf)
- [http://teclms.ruh.ac.lk/pluginfile.php/18177/mod\\_resource/content/0/Introduction%20to%20MySQL%20Part%20III.pdf](http://teclms.ruh.ac.lk/pluginfile.php/18177/mod_resource/content/0/Introduction%20to%20MySQL%20Part%20III.pdf)
- [http://teclms.ruh.ac.lk/pluginfile.php/18175/mod\\_resource/content/0/Introduction%20to%20MySQL%20Part%20IV.pdf](http://teclms.ruh.ac.lk/pluginfile.php/18175/mod_resource/content/0/Introduction%20to%20MySQL%20Part%20IV.pdf)
- <https://dev.mysql.com/doc/>