\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**SCHOOL OF ENGINEERING AND TECHNOLOGY**

**COURSEWORK FOR**

BSC (HONS) IN COMPUTER SCIENCE

BSC (HONS) INFORMATION TECHNOLOGY

BACHELOR OF SOFTWARE ENGINEERING (HONS)

BSC (HONS) INFORMATION SYSTEMS (DATA ANALYTICS)

BSC (HONS) INFORMATION TECHNOLOGY (COMPUTER NETWORKING AND SECURITY)

**YEAR 1;** ACADEMIC SESSION SEPTEMBER 2024

**SEG1201: DATABASE FUNDAMENTALS**

**Answer Template,SQL Script and Demonstration: Due date :Week13 Sunday (22.12.2024), 11.59 p.m)**

**SEG1201: DATABASE FUNDAMENTALS**

**STUDENT NAMES: Muhammad Umer Danka (Individual Reflection)**

**STUDENT IDS: 22104053**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

INSTRUCTIONS

* This final assessment contributes 50% to your final grade.
* This five-member group assignment (or minimum 3-member group) is primarily for Course Learning Outcome 2 - Implement a database design group project using appropriate tools such as Oracle SQL.

**IMPORTANT**

The University requires students to adhere to submission deadlines for any form of assessment. Penalties are applied in relation to unauthorized late submission of work.

**Academic Honesty Acknowledgement**

I **Muhammad Umer Danka (22104053)** verify that this paper contains entirely our own work.  We have not consulted with any outside person or materials other than what was specified (an interviewee, for example) in the assignment or the syllabus requirements.  Further, we have not copied or inadvertently copied ideas, sentences, or paragraphs from another student.  We realize the penalties *(refer to page 16, 5.5, Appendix 2, page 44 of the student handbook diploma and undergraduate programme)* for any kind of copying or collaboration on any assignment.”

**21/12/2024**(Student signatures/Date)



<<LastMinuteMaestros>> -

<<YUOR Delights>>

Individual Reflection of:

|  |  |  |  |
| --- | --- | --- | --- |
| No | Student ID | Name | Email address |
| 1 | 22104053 | Muhammad Umer Danka | 22104053@imail.sunway.edu.my |
| 2 | 23013337 | Lim Yu Bin | 23013337@imail.sunway.edu.my |
| 3 | 22071187 | Ooi Jin Li | 22071187@imail.sunway.edu.my |
| 4 |  |  |  |
| 5 |  |  |  |

# PART 3: INDIVIDUAL REFLECTION AND ANALYSIS (25 MARKS)

1. Identify any potential data anomalies (update, insertion, or deletion anomalies) that could arise from the database design proposed in your group project.

A possible update anomaly that may occur in my group’s database design could be when an order for a dine in reservation is completed in advance and may have been paid for. Consequently, if the customer then decides to ask for a delivery instead of dining in, the ordertype of ‘Dine In’ must then be changed to ‘Delivery’ and the whole order must be deleted from the dinein entity and must be placed in the delivery entity instead. This may not be very difficult however, if the order is changed since the customer food preferences for delivery may be different, then all orderitems or comboitems for that particular orderID must be changed, deleted or added based respectively and that the paymentID details must be adjusted too. This can be a challenging and may lead to inconsistencies.

Additionally, a possible insert anomaly may occur when a new ingredient is added into the database system, but no item requires that particular ingredient and it is not used. Hence the insertion of that ingredient becomes pointless

Lastly, a delete anomaly may occur in my group’s database system when the restaurant decides to delete an item or a combo. Hence all order, orderitem, ordercombo tied toward the specific item or combo must be deleted too or else the orders will not make sense. This forces the restaurant to not delete an item and concurrently keep all past combos and items in the database so that orders in the past can make sense and are not forcefully deleted from the system. Hence the menu list of items and combos cannot be directly created from the database system since not some items may have been removed due to its performance or customer preferences. This effects consistency and seamlessness that the database system for which it was initially implemented for.

1. Suggest at least 2 (TWO) specific improvements to the ERD, explaining how these changes would enhance the database design, data integrity, or query efficiency.

The ERD has great transitive connectivity throughout the diagram from containing a composite entity between orders and items to having a composite entity between items and ingredients allowing smooth and meaningful data flow. However, the Ingredient entity fails to provide comprehensive usefulness towards the database system other than listing out which ingredients make up an item. I suggest that additional attributes should be created for the ingredient entity such as ingredient stock, price per unit of measurement relevant to the ingredient and the supplier details of each ingredient. This allows transparent yet cohesive inventory management for the restaurant easing towards expense calculation, identifying added value for each item while ensuring that stock of important ingredients is always available to make sure no item runs out during working hours and popular dishes are always available; improving customer satisfaction.

On the other hand, the database system splits orders into their respective categories using an entity of dineIn and delivery. This allows additional information to be inserted such as the person count of customers which aided in average cost and item requirement per dined-in person alongside an exit time attribute which aids in calculating the time customers spend in the restaurant on average. For the delivery entity, additional information such as deliveredtime allows the restaurant to identify whether foods are being delivered quickly and on time. However, this has led to an update anomaly when a sudden change in customer requirement or order type is enforced as mentioned above. Hence keeping all orders in the orders entity itself and not splitting them into separate entities while concurrently connecting them using the ordertype attribute seems ideal. Additional attributes that are deemed to be important or useful can be created without the not null constraint. Though null is something inconvenient and it is generally avoided, this approach not only eliminates a particular update anomaly but also simplifies database structure, enhances data integrity and makes querying more efficient.

1. Reflect on your individual contribution to the group project. Describe your role, the specific tasks you were responsible for, and the challenges you faced.

My primary role in this project was creating and populating the SQL tables based on the entity relationship diagram (ERD) developed in Part 2. While my contribution to the ERD itself was minimal despite creating the combo tables in the ERD, I was mainly responsible for overlooking its logical structure and ensuring its implementable.

Additionally, I significantly contributed to the scenario write-up and developed business rules, basing them on the ERD that was primarily created by my teammates. I emphasized certain rules that better aligned with the business logic and the ERD.

I also created a few of the queries with input and ideas from my team members. A key challenge during this project was coming up with a three-member group, which became even more difficult when one member disappeared midway through the semester. The group was only structured just nine days before the initial submission deadline. Moreover, balancing this assignment with closely due tasks from other subjects required careful time management.

Despite these challenges, the assignment was both enjoyable and informative, offering valuable insights and experiences.