**55-5445 Database Systems for Software Applications**

**Coursework Assignment Brief – February 2014:**

**Space & Equipment Management**

A university not far away needs an application to help manage and monitor the usage of offices, classrooms, labs, and other spaces, and the equipment deployed in them.

The university has several buildings of various sizes on the campus, and all spaces within a building which can contain equipment or people are within the scope of the system. This clearly means offices, classrooms and labs, but also includes kitchens, toilets, shower rooms, catering outlets, corridors, stairwells, lift-shafts, atriums etc.

It is essential to know what furniture there is in each office and teaching space, as that detemines the capacity, which in turn affects allocation of rooms. But it is also important to know what other equipment is in each space, and this may include common items such as PC’s, printers, projectors, photocopiers, telephones etc., as well as other things such as fire-extinguishers, display boards, visualisers, OHP’s, filing cabinets, cookers, fridges, coffee machines, water coolers, water heaters, lifts, - in fact anything.

Many spaces are used as offices housing both administrative and academic staff, and the system needs to be able to record which staff are based in which rooms, what their internal phone number is, and which PC they use (this will help the IT staff to install appropriate software). It must also record which PC’s are in which teaching rooms, and distinguish the lectern PC (if there is one).

Effective utilisation of space is a key objective. To help monitor utilisation, every event planned to occupy a space is to be recorded by the system, so that unused space can be identified easily. Events include timetabled teaching sessions in classrooms and labs, but may also include staff or staff-student meetings in conference rooms, or public events such as open days or exhibitions in the larger spaces. Of course it must also include exams. All such events will be pre-booked though the space administrator.

Some specific facts which came to light during the investigation include the following:

The capacity of some spaces can vary according to the configuration, for example

desks/tables may be set out in rows, as islands, in a horseshoe, or individually as for an exam.

In most cases the number and quantity of equipment in a space is sufficient, but for some types of equipment it is required to identify items individually – eg. for PC’s, photocopiers, etc.

All events have a staff member as the principal organiser who may be contacted for queries or amendments. If the event is a teaching session the title will include the module code and session, but this will be supplied by the organiser when booking – there is no requirement to link these events to the (separate) timetabling system.

It is not required to consider which software is installed on IT equipment.

**Coursework Assignment Tasks**

**Individual Tasks**

EACH MEMBER OF THE GROUP must complete two individual tasks as part of this assignment.

**Individual Task 1**

1. Normalise the business data provided in Appendix 1 and draw a corresponding entity relationship diagram.
2. Also produce an outline entity relationship diagram for the entire system based on the notation the discussed in the lectures of January and February 2014.

This piece of work will attract 10% of the project marks. It will not be marked until later, but you will receive formative feedback on it which may influence the group solution.

***In order to receive useful feedback this piece of work must***

***be handed in by the end of w/c March 3rd***

**Individual Task 2**

Write an individual critical reflection of the entire project from a personal perspective. This should consider how the project work was organised; whether it worked it well or badly; your own role in the project and what you have learned from it; and/or what might be done differently if you had to do it again.

The critical reflection should be no more than 500 words, and will attract 10% of the project marks. It should be handed in at the walkthrough of the group work.

**Group Tasks**

1. Produce a final ERD for the system. This should be based on a consolidation of the individual normalisation products, and the individual diagrams for the entire system. It must include a written explanation of any differences and how they were resolved.
2. Specify the tables and any other objects (eg. views, indexes, sequences) which you would expect to implement to create the database. This means identifying for each table; - its name, attributes, datatypes, and any constraints.
3. Create an SQL script containing the code to create the objects, including comments to explain the purpose of main statement. Hand in a print-out at the walkthrough, but have the script available to run if required.
4. Run the script to create the database (using Oracle), then load the business data supplied in the spreadsheets. Provide a brief written explanation of the strategy and methods used to load the data, and how any problems were solved. Hand in an SQL script containing the code for the INSERT statements for all of the data loaded.
5. Provide a brief written explanation of the strategy and methods used to test that the data has been loaded correctly. Be prepared to demonstrate the validity of the data at the walkthrough, and explain how your method would scale to larger volumes.
6. Write SQL queries to accomplish the following list of business requirements. You may demonstrate the queries with SQL Developer, but demonstration by submission of a prepared statement from a web-page will attract extra marks\*.

a) Derive a phone list of all the people in a user-specified building.

b) Produce a list identifying the items in every teaching space of a user-specified

building.

c) Find one or more vacant rooms to seat a user-specified number of people in a

user-specified building on a given date and time.

d) Insert a new event: a meeting, with a user-specified title and organiser.

e) Insert a room booking for that event, using a room from (c).

\* Does not need to demonstrate user input via HTML forms

***Your group work documentation will be handed in, and the solution demonstrated to tutors, at a walkthrough to be held during w/c April 28th***

***.***

Checklist for documentation required to be handed in at the walkthrough:

**Group task:**

* Entity relationship diagram
* Specification of database tables and other objects (including primary keys, foreign keys, constraints).
* SQL script(s) to implement the database objects.
* Written description of data load strategy / procedure
* SQL script of INSERT statements to reload the data.
* Written description of strategy / procedure used to test the data load
* SQL script(s) containing the code for the queries.

**Individual Task 2:**

* Individual critical reflection – as specified above.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Space & Equipment Management: Group Assignment – Marking Scheme** | | | | | | |
| **Group:  No of Members:** | <40% Fail | 40 - 50 Weak Pass | 50 - 60 Pass | 60 - 70 Good Pass | 70 - 80 Distinctive | 80 + Very Distinctive |
| **Entity Relationship Diagram of entire system, plus commentary / justification**  **[10]** | Missing or demonstrates little or no understanding | Important entities missing or poorly described in terms of attributes. Many incorrect relationships | Most important entities present. Sensible and adequate attributes shown.  Relationships mostly correct. | Most PK’s and FK’s indicated. May be some minor discrepancies between keys and relationships. | All relationships shown correctly, with labels, and supported by PKs and FKs. | + closely models the problem domain |
|  |  |  |  |  |  |
| **Database Specification**  **(table definitions showing attributes datatypes, constraints)**  **[10]** | Missing or demonstrates little or no understanding | Basic description of tables – names and attributes only. | Data types shown for most attributes. PK and FK constraints shown. | Not Null, Check, Unique, and Default constraints shown where appropriate. PK and FK constraints mostly correct. | PK and FK constraints named. Evidence of naming scheme. | + use of comments explaining purpose of columns and/or constraints |
|  |  |  |  |  |  |
| **Database Creation Script**  **(commented code for tables and other objects)**  **[10]** | Missing or demonstrates little or no understanding | Implementation of a basic set of tables. May be some inappropriate data types | All tables specified and constraints implemented. Mostly correct data types. Non-table objects also defined. | A comprehensive solution with all required PK’s and FK’s correctly implemented. | + code well formatted.  DROP statements +  GRANT sttmnts + | + use of comments to document the script |
|  |  |  |  |  |  |
| **Data Take-on**  **[10]** | Little or no data loaded | Some data loaded. | Most data correctly loaded. Data not loaded is clearly identified. | Script supplied to correctly reload all of the data in correct sequence. Outline written description of strategy / procedure | Detailed written description of data loading strategy and procedure. | Data mapping document relating source data to database tables/columns. |
|  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | <40% Fail | 40 - 50 Weak Pass | 50 - 60 Pass | 60 - 70 Good Pass | 70 - 80 Distinctive | 80 + Very Distinctive |
| **Data Take-on Testing**  **[10]** | Little or no evidence of testing | Simple manual checks only. | A few examples of SQL scripts to check certain aspects. | Script(s) to run/re-run systematic and comprehensive checks. | Outline description of data loading strategy and procedure. | Detailed description of data loading strategy and procedure, including problem resolution. |
|  |  |  |  |  |  |
| **SQL Queries**  **[30]** | No working query. | Query code exists as script but returns errors. | Query executes but returns an invalid result.  Awareness of flaws + | Query executes and returns a valid result. Evidence of testing + | Output appropriately organised and titled. | Implemented from web-page as prepared statement. |
| Phone list of all the people in a user-specified building. [5] |  |  |  |  |  |  |
| Produce a list identifying the items in every teaching space of a user-specified building. [5] |  |  |  |  |  |  |
| Find one or more vacant rooms to seat a user-specified number of people in a user-specified building on a given date and time. [10] |  |  |  |  |  |  |
| Insert a new event: a meeting, with a user-specified title and organiser. [5] |  |  |  |  |  |  |
| Insert a booking for that event, using a room from (c). [5] |  |  |  |  |  |  |

**Space & Equipment Management: Individual Marking Scheme**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | <40% Fail | 40 - 50 Weak Pass | 50 - 60 Pass | 60 - 70 Good Pass | 70 - 80 Distinctive | 80 + Very Distinctive |
| Entity Relationship Diagram  (entire system)  [5] | Missing or demonstrates little or no understanding | Important entities missing or poorly described in terms of attributes. Many incorrect relationships | Most important entities present. Sensible and adequate attributes shown.  Relationships mostly correct. | Most PK’s and FK’s indicated. May be some minor discrepancies between keys and relationships. | All relationships shown correctly, with labels, and supported by PKs and FKs. | + closely models the problem domain |
|  |  |  |  |  |  |
| Submit an individual critical reflection of the entire project from a personal perspective.  [10] | Little or no critical reflection | A description of problems and successes is given. | Problems and successes are identified, with some evaluation and suggestions for improvement. | Provides an appraisal of the degree of completeness of the project, and evidence of personal learning. | + suggests alternative avenues that could have been taken. | + evaluation of own learning |
|  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| NB: RANGES >> | <20% Fail | 20 - 40 | 40 - 50 | 50 - 60 | 60 - 80 | 80 + |
| Normalisation Products  (derivation of 3rd normal form, plus a corresponding ERD)  [5] | Missing or demonstrates little or no understanding | Mostly correct UNF attributes. PK identified. Repeating groups identified (if present) | + mostly correct 1NF. PK’s identified. RG’s separated (if present) | + mostly correct 2NF. PK’s identified. PKD’s removed (if present) | + mostly correct 3NF. PK’s & FK’s identified. NKD’s removed (if present).  Corresponding ERD | Fully correct normalisation with all keys marked. Correct ERD well named and labelled. |
|  |  |  |  |  |  |