|  |  |  |
| --- | --- | --- |
|  | Faculty of Computing, Engineering and Science |  |

**Assessment Cover Sheet and Feedback Form** 2018-19

|  |  |  |  |
| --- | --- | --- | --- |
| Module Code:  IS1S466 | Module Title:  Information Management, Assurance and Security | | Module Team:  Gaylor Boobyer |
| Assessment Title and Tasks:  Portfolio 1 Element 3 | | | Assessment No.  1 |
| Date Set:  **13-Mar-20** | | Submission Date:  **03-Apr-20** | Return Date:  **15-May-20** |

|  |  |
| --- | --- |
| ***Part A: Record of Submission (to be completed by Student)*** | |
| **Extenuating Circumstances**  If there are any exceptional circumstances that may have affected your ability to undertake or submit this assignment, make sure you contact the Advice Centre on your campus prior to your submission deadline. | |
| **Fit to sit policy**:  The University operates a fit to sit policy whereby you, in submitting or presenting yourself for an assessment, are declaring that you are fit to sit the assessment. You cannot subsequently claim that your performance in this assessment was affected by extenuating factors. | |
| **Plagiarism and Unfair Practice Declaration:**  By submitting this assessment, you declare that it is your own work and that the sources of information and material you have used (including the internet) have been fully identified and properly acknowledged as required[[1]](#footnote-1). Additionally, the work presented has not been submitted for any other assessment. You also understand that the Faculty reserves the right to investigate allegations of plagiarism or unfair practice which, if proven, could result in a fail in this assessment and may affect your progress. | |
| **Details of Submission:**  Note that all work handed in after the submission date and within 5 working days will be capped at 40%[[2]](#footnote-2). No marks will be awarded if the assessment is submitted after the late submission date unless extenuating circumstances are applied for and accepted (Advice Centre to be consulted).  **Work should be submitted to Blackboard on the submission date above but this will be treated as your submission receipt and you need to provide one printed copy on Monday 8th April 2019 before noon.**  You are responsible for checking the method of submission. | |
| **You are required to acknowledge that you have read the above statements by writing your student number (s) in the box:** | Student Number(s): |

**IT IS YOUR RESPONSIBILITY TO KEEP RECORDS OF ALL WORK SUBMITTED**

|  |
| --- |
| **Marking and Assessment** |
| This assignment will be marked out of 100%  This assignment contributes to 25% of the total module marks.  **Assessment Task:**  Using the given ERD, analyse the requirements and then:  a) Produce the table definitions (in **SQL**) for the following tables:  (Refer to the later queries in order to determine sensible attributes for each of  these tables.)   * Course Type * Allocated Student * Current Course * Allocation * Room * Trainer   Remember to define a primary key and foreign key(s) where necessary.  (Hint: Refer to ERD already given for most of these)    Produce hard copy of each of these table definitions.    Produce the Access relationship diagram generated by these table definitions.  b) Populate your tables with **sensible** data.  (Remember you have to use this data in order to satisfy the following queries.)  List the data contained in each of these tables.  (Use the datasheet view in Access)  Produce **hard copy** of all these lists.  c) Write the SQL statements necessary to satisfy the following queries. Remember to use sensible headings where appropriate:  **DO NOT use the QBE grid!**  (Produce **hard copy** of the SQL together with the corresponding datasheet view of the resulting output)  (1) Produce a list of currently running courses by course code then date order.  **(4 marks)**  (2) Produce a list of courses that have ‘Introduction’ as part of their title.  **(4 marks)**  (3) List the most expensive and the least expensive course that GBT provides. Use sensible headings.  **(6 marks)**  **Assessment Task: Cont.**  (4) Produce a list of the number of students on each of the currently allocated courses. (This should be calculated by the query NOT held as a piece of data)  **(6 marks)**  (5) Produce a list of the current courses between two dates. Make sure you choose dates that provide MEANINGFUL output from your data.  **(6 marks)**  (6) Produce a list of the number of current courses of each type between two dates. Your output should be similar to:  **Course Code Number of actual courses running**  CIMW 2  CAMW 5  CIES 5  CAES 3  **(7 marks)**  (7) Produce a list of the rooms (include type of room: e.g av facilities or not) and trainers (by name) allocated to each course for each week.  (Check your output here as this will test whether you have entered sensible data)  **(7 marks)**  (8) Produce a list of students (include student name) currently allocated to an actual course (use your data to select an actual course – E.g CIMW starting on 12/11/18)  **(7 marks)**  (9) Produce a list of the current courses together with the total revenue that each course would have/ will brought/bring in.  I.e the number of students on each course multiplied by the number of students taking that course.  E.g  **CourseCode Date Run CourseCost No of Students Revenue**  CIMW 12/11/18 100 20 2000  CIMW 12/12/18 100 25 2500  CAMW 11/01/19 200 10 2000  Etc  **(8 marks)**  **Up to 10 extra marks are available for answers to the above queries that are exceptional.** |
| **Learning Outcomes to be assessed** (as specified in the validated module descriptor [https://icis.southwales.ac.uk/](https://icis.southwales.ac.uk/studentmodules/11452/studentmodulespecifications) ):  1) To demonstrate a practical understanding of the design and implementation of information systems.  2) To demonstrate the ability to recognise any risks or safety aspects that may be involved in the operation of computing equipment within a given context. |
| |  |  |  | | --- | --- | --- | | **Marking Scheme** | **Marks Available** | **Marks Awarded** | | Table definitions in SQL (DDL) | 30 |  | | Sensible table population | 5 |  | | Specified SQL statements (DML) | 55 |  | | Exceptional elegance of SQL queries | 10 |  | | **Total** | 100 |  | |

**Appendix A**

GB Training Organisation

GBT is a training organisation that provides various training courses. The company has decided to put all the information they now keep manually into a database and to include course booking, accommodation booking, trainer allocation and room allocation in this new database.

# Background

Students can apply for any of the courses provided.

Each course has one or more trainers who are skilled in providing that course. A trainer is allocated to each course occurrence (actual course).

If students attending a course require accommodation for the duration of the course, then accommodation is provided in local boarding houses which are booked by GBT and added to the overall cost of the course. They also keep a record of which students are booked into which boarding house and on which dates.

**Current System**

There is no standard method of receiving course applications. They can be made by telephone, letter or through a Course Application form which is issued with the Course Catalogue.

If a student makes a general enquiry about a course, then the Student file is checked to see if he/she is a current student. If a student is not in the Student file then he/she is sent a Student Details form to complete together with the Course Application form.

When the students return their Student Details form they are allocated unique student numbers and the students’ details are then added to the Student file.

# Course Booking

Upon receipt of a valid course application the Current Course file is checked for a vacant place on a currently organised course (actual course). If there is a place available, then the student is allocated to that course (actual course) and notification details are sent to the student.

If there is not a place currently available on a Current Course, students are allocated to the Type of Course they wish to attend. When there are enough students waiting to attend a course then a new actual course is set up and students waiting for that course are sent notification details and allocated to a Current Course (actual course).

When a student is notified of his/her acceptance onto a course, possible accommodation details are sent out and an accommodation booking is subsequently made, if required.

A training room is allocated to an actual course by the Course Booking Section. Only one room will be allocated for each course run. The training room list is checked for available rooms, an available room is allocated and the room list is updated. Trainers are then allocated to the actual course. The system must check which trainers are qualified to teach that particular course and which trainer is currently available to be allocated to the course.

**Miscellaneous**

The courses provided can be of **1,3 or 5 days** duration.

A course type may contain the following information:

Course Code: Course Title

CIMW1 Introduction to Microsoft Word

An actual course may contain the following information:

Course Code: Start Date Duration MaxNoStudents

CIMW1 12/10/19 3 10

**Appendix B**

1..\*

Course

Type

Allocated

Student

1..\*

Allocation

0..\*

Current

(Actual)

Course

1..1

1..1

1..1

0..\*

0..\*

1..1

1..1

Trainer Room

# Marking Scheme:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Fail | Narrow Fail | 3rd Class / Pass | Lower 2nd Class / Pass | Upper 2nd Class / Merit | 1st Class / Distinction |
| Table definitions in SQL (DDL) 30% | * Relations/Tables missing.Primary keys, foreign keys and not null/null used incorrectly in almost all instances | * Relations/Tables missing.Primary keys, foreign keys and not null/null used incorrectly in most instances | * No Relations/Tables missing.Primary keys, foreign keys and not null/null used incorrectly in many instances | * No Relations/Tables missing.Primary keys, foreign keys and not null/null used incorrectly in some instances | * No Relations/Tables missing.Primary keys, foreign keys and not null/null used correctly in many instances | * No Relations/Tables missing.Primary keys, foreign keys and not null/null used correctly in most/all instances |
| Sensible Table Population 5% | * None/very few tables populated with no/some data. Data inadequate to perform later queries | * some tables populated with no/some data. Data inadequate to perform later queries | * All tables populated with some data. Data mostly adequate to perform later queries | * All tables populated with a reasonable amount of data. Data suitable to perform later queries | * All tables populated with a good amount of data. Data suitable to perform later queries. Most table data size relates to expected table size based on the relationship between tables | * All tables populated with a good amount of data. Data suitable to perform later queries All table data size relates to expected table size based on the relationship between tables |
| SQL DML statements 55% | * Most SQL statements that do not meet the information retrieval requirements detailed in the scenario | * Many SQL statements that do not meet the information retrieval requirements detailed in the scenario | * SQL statements that meet some of the information retrieval requirements detailed in the scenario | * Some SQL statements that do not meet the information retrieval requirements detailed in the scenario | * SQL statements that meet many of the information retrieval requirements detailed in the scenario | * SQL statements that meet all of the information retrieval requirements detailed in the scenario |
| Exceptional elegance of SQL queries 10% | * No attempt at SQL statements that produce innovative/ exceptional coden | * Very little at SQL statements that produce innovative/ exceptional code | * Little attempt at SQL statements that produce innovative/ exceptional code | * Some attempt at SQL statements that produce innovative/ exceptional code | * Some very good attempts at SQL statements that produce innovative/ exceptional code | * Excellent attempt at SQL statements that produce innovative/ exceptional code |
|  | | | | | | |

1. University Academic Integrity Regulations [↑](#footnote-ref-1)
2. Information on exclusions to this rule is availablefrom Campus Advice Shops [↑](#footnote-ref-2)