School of Computing, Engineering and Mathematics

CI504 Coursework assignment 2022-23

Module Title:	**Databases Management Systems**
Module Code:	**CI504**
Author(s)/Marker(s) of Assignment	Jennie Harding
Assignment No:	Parts A and B
Assignment Title:	Individual Coursework – Bet's Pets' Vets
Assignment weighting:	Part A 50% of module mark Part B 50% of module mark
Module Learning Outcome/s Covered	LO1 Analyse and model database designs, including more complex design elements LO2. Create a database to meet the requirements of an organisational information system, justifying the choices made LO3. Use advanced features of a database management system (e.g. triggers, stored procedures, backup and restore)
Assignment Brief and Assessment Criteria:	
See attached sheets for tasks and Assessment Criteria and Case Study	
Date of issue:	w/b 10/10/2022
Deadline for submission:	**The assignment is in two parts. Each part is equally weighted.** **Part A Design: HAND IN: 19/01/2023 at 15:00h** via TurnItIn (PDF format of design document) **Part B Implementation: HAND IN: 11/05/2023 at 1500h** hours via TurnItIn (PDF format of technical user manual in single document) *Note - Students are allowed to submit work within two weeks of the published deadline or the last working day immediately prior to the feedback date if this is shorter than two weeks – late work is capped at the pass mark.*
Date feedback will be provided	Feedback will be provided electronically through TurnItIn within 4 weeks of the hand-in date

Assessment Tasks – Part A - Design

Write a formal report which documents the logical design of a database to support the Information System requirements described in the case study (Appendix 1, Appendix 2).

1) Evidence of normalization.
Normalise the supplied document (Appendix 3) from the case study (removing repeating groups, resolving functional dependencies). The outcome of the normalisation should inform your ERD.

2) Entity-Relationship Diagram
Produce a key-only Entity-Relationship Diagram (ERD) showing primary and foreign keys to meet the requirements, showing all entities and relationships (with relationship name, cardinality and optionality).

3) Data Dictionary
For each table, list the table name, a description of the entity that is implemented, the primary key of the table (name and data type) and other identified attributes (names / data types / attribute description / default and any other relevant information).

Assessment Tasks – Part B – Implementation

Write a formal report demonstrating that you have created a database to meet the requirements of the case study. You should use text and images to demonstrate this.

1. *Create tables using SQL DDL*

   Write SQL statements to create the tables that implement the database you designed in part A (or the part model answer that you will be given after Part A was handed in). You must show the SQL CREATE TABLE statements and images of the tables created.

2. *Populate tables using SQL INSERT*

   Write SQL statements to populate all the tables you have previously created, with at least five records each. Your report must show the SQL INSERT statements and images of the tables when they have been populated. *NOTE: you should consider creating a formal test data set where you know what the outcome of a given SQL query will be.*

3. *Retrieve information using SQL queries*

   Write SQL queries to demonstrate that your system can address the requirements in Appendix A. You must show the SQL code and the result of executing each query. *These should be clearly labelled with the requirement number and the test data should be sufficient to demonstrate that the SQL works.*

4. *Implementation of DBMS functionality*

   Design appropriate stored procedures, functions, triggers and application to implement the functionality defined in case study. Your report must show the SQL code and the result of execution.

5. *Business Justification*

   A short (max: 500 words) explanation of how the implemented features will enable the business to meet its objectives. *[NOTE: this is not about the database itself: benefits not features.].* You may suggest additional features that you think would be of use to the business.

Appendix 1: Case Study

Bet's Pets' Vets

Bet's Pet's Vets is a small chain of vets operating in the South East. There are 8 surgeries employing 30 vets and vet nurses. Staff may be required to work in any branch.

They require a small database to manage appointment and payments.

Pet owners book appointments with either a vet or vet nurse. These usually take place at a surgery. Home visits are offered for larger animals, e.g. horses. Some owners have more than one pet – but each appointment slot can only be for one animal.

Most animals are seen regularly – mainly dogs, cats etc. for regular vaccinations. There is an emergency callout service. For each animal the species is recorded, in order to allocate the correct staff -– though they do have an 'other' category to cover the odd animal e.g. axolotl, seahorse, that they have been asked to treat over the years. Examples shown in table.

petTypeID	petTypeDesc
CT	cat
DG	dog
HR	horse
RB	rabbit
RD	rodent
OT	other

Each appointment is allocated a treatment code, which fixes the cost. Examples shown in table.

Code	Treatment Name	Treatment Description	Cost (£)	Staff Required
T40	30 min vet	Home Visit	100	Vet
T41	D-VAX-1	Dog - first dose standard	70	Vet or Nurse
T42	D-VAX-2	Dog - booster annual standard dog	50	Vet or Nurse
T43	C-VAX-1	Cat - first dose standard	60	Vet or Nurse
T44	C-VAX-2	Cat - booster annual standard	35	Vet or Nurse
T45	diagnostic	Examination to determine cause and suggest treatment	45	Vet
T46	D-teeth	Dog - examine and monitor teeth / mouth	30	Vet or Nurse
T47	test	Biopsy, blood test etc.	33	Vet or Nurse
T48	Minor Operation	Small op under local anesthetic	150	Vet

Each single appointment immediately generates a single invoice, which must be paid in full within 28 days. Some owners pay in full at the appointment, and some make a number of payments to settle a single invoice. (If a payment is to cover more than one invoice, the oldest is recorded as paid, and then the balance applied to more recent invoices).

Most staff have an identified line manager (but not all; the most senior vet at each surgery in line-managed via Head Office, which is beyond the scope of this system).

A full list of requirements is shown in Appendix 2.

Appendix 2: System Requirements

After consulting *Bet's Pets' Vets*, the following requirements have been identified and prioritized. For all queries, consider the order of the results. The data storage system:

MUST

1. Record information for each pet, including its type / species, approximate age (DoB not always known), comments.
2. Enable one owner to be shown for multiple pets
3. Record staff information, identifying whether they are VETs or VET NURSEs.
4. Record appointment information, showing pet and staff information as well as date and time. Allow space for outcome of appointment to be recorded.

SHOULD

5. Show all appointments for today (including home visits) for a given staff member. Include time, pet, owner, staff, and treatment codes and cost.
6. Record payments made, including the invoice information, amount paid, method of payment etc. A payment can only be applied to one invoice (see Case Study). Generate a unique receipt number for each payment.
7. Show staff information, showing their line managers. Include staff who do not have line managers.
8. Calculate the total owed by each owner (whether invoices are overdue or not).

COULD

9. Identify any unpaid invoices dated more than 28 days ago, showing the name and address of the responsible owner. Identify the owners that owe the most.
10. Calculate the total amount owed to each surgery (consider how Home visits will be included).
11. Find the average income generated by each staff member per month (ignore unpaid invoices).

WOULD BE NICE

12. Create a stored procedure to input new pet information, including associating them with an existing *or* new owner (given the owner's name).
13. Create a trigger that will check how much a given owner owes when they make an appointment. If there are any unpaid invoices more than 28 days old, do not make the booking.
14. Create an app capable of running the queries.

Appendix 3: Sample Document For Normalisation

BET's PETs Vets — **Today's Appointments**

STAFF NAME:	Gurkiran Greenwell				**StaffID**	s50	**STAFF TYPE ID**	**1**		**STAFF TYPE DESC**	VET			
Date/ Time	*Surgery Code*	*Surgery Details*	*Pet ID*	*Pet Name*	*Pet Type ID*	*Pet Type Desc*	*Owner ID*	*Owner Name*	*Treatment Code*	*Treatment Name*	*Cost (£)*	*Outcome*	*Invoice Number*	*Paid*
13/11/22 09:00h	WG	Worthing Surgery Montague Place	P2003	Ratty	RD	RODENT	tt031	Trevor Tatler	T47	Blood test …	33	biopsy sent	FFF1234	Y
13/11/22 1145h	WG	Worthing Surgery Montague Place	P2004	Carrots	RB	RABBIT	bb902	Bettina Bishop	T45	Diagnostic	45	Annual check - no issues	GGG1234	N
13/11/22 13:15		Home Visit	P2005	Pinchbeck	HR	HORSE	ss011	Sally Smith	T40	Home Visit	70	*NULL*	*NULL*	*NULL*
13/11/22 15:00	BN	Brighton Surgery Old Steine Place	P2002	Sleeky	CT	CAT	yy721	Yolande Yearling	T45	Diagnostic	45	*NULL*	*NULL*	*NULL*
13/11/22 19:00	WG	Worthing Surgery Montague Place	P2003	Ratty	RD	RODENT	tt03	Trevor Tatler	T48	Small op…	150	*NULL*	*NULL*	*NULL*

STAFF NAME:	Sarah Smith				**StaffID**	s53	**STAFF TYPE ID**	**2**		**STAFF TYPE DESC**	VET NURSE			
Date/ Time	*Surgery Code*	*Surgery Details*	*Pet ID*	*Pet Name*	*Pet Type ID*	*Pet Type Desc*	*Owner ID*	*Owner Name*	*Treatment Code*	*Treatment Name*	*Cost (£)*	*Outcome*	*Invoice Number*	*Paid ?*
13/11/22 10:45h	WG	Worthing Surgery Montague Place	P2001	Rover	CT	CAT	pp091	Paul Preston	T46	D-teeth	30	Advised chew sticks …	CCC123	N
13/11/22 13:30h	WG	Worthing Surgery Montague Place	P2006	Bugs	RB	Rabbit	pp231	Praney Patel	T47	Blood test…	33	*NULL*	*NULL*	*NULL*

NOTE: there is currently no location code for a home visit.

This is the view of the appointments at 1300h. Outcome, Invoice Number and Payment Status are recorded after the appointment.

Appendix 4: Marking Criteria Part A

	A	B	C	D	E	F
Normalisation 30%	All repeating groups have been removed. All functional dependencies have been resolved. Tables are in 3NF.	Most repeating groups have been removed. Most functional dependencies have been resolved. Tables are in 3NF. Most repeating groups have been removed. Most functional dependencies have been resolved. Tables are in 3NF.	An attempt has been made to normalize the tables that shows reasonable understanding of the principles - for example, attributes have been used that are not in document supplied.	An attempt has been made to normalize the tables that shows some understanding of the principle - some elements are normalised correctly, though there are errors.	An attempt has been made at normalisation, but it does not show an understanding of many of the principles.	An unsatisfactory attempt at normalisation, showing little awareness of the principles.
KEY ONLY ERD 35%	Diagram properly drawn using a CASE tool. *All* necessary tables are defined. Diagram matches the tables and relationships identified in other parts of the documentation. Primary and foreign keys are shown correctly. Multiplicity and optionality indicated on relationships. This work is completed to a near professional standard.	There are SOME minor issues with the ONE of the following: -ER diagram produced using a standard notation -all necessary tables are defined -using primary and foreign keys to make relationships -diagram properly drawn using a CASE tool -diagram largely matches the tables and relationships identified in other parts of the documentation -multiplicity and optionality indicated on relationships	There are SOME minor problems with TWO or more of the following, or major issues with ONE: -ER diagram produced using a standard notation -all necessary tables are defined -using primary and foreign keys to make relationships -diagram properly drawn using a CASE tool -diagram largely matches the tables and relationships identified in other parts of the documentation -multiplicity and optionality indicated on relationships	There are SOME minor problems with THREE of the following or major problems with TWO: -ER diagram produced using a standard notation -all necessary tables are defined -all necessary tables are defined -using primary and foreign keys to make relationships -diagram largely matches the tables and relationships identified in other parts of the documentation -multiplicity and optionality indicated on relationships	There are MAJOR problems with THREE or more of the following: -ER diagram produced using a standard notation -all necessary tables are defined -all necessary tables are defined -using primary and foreign keys to make relationships -diagram largely matches the tables and relationships identified in other parts of the documentation -multiplicity and optionality indicated on relationships	There are MAJOR problems with FOUR or more of the following: --ER diagram produced using a standard notation -all necessary tables are defined -all necessary tables are defined -using primary and foreign keys to make relationships -diagram largely matches the tables and relationships identified in other parts of the documentation -multiplicity and optionality indicated on relationships
Data Dictionary 25% **Presentation 10%**	Primary and foreign keys are identified across tables. Data dictionary matches ERD. Datatypes are well-chosen. Constraints are sensible. Default data is suggested as appropriate. Field names are useful. Useful sample data is supplied. This work is completed to a near professional standard. Work is presented to an excellent standard. Clear relationship between different parts of the assignment. Assumptions are reasonable and explained.	Primary and foreign keys are identified across tables. Data dictionary matches ERD. Datatypes are well-chosen. Constraints are sensible. Default data is suggested as appropriate. Field names are useful. Useful sample data is supplied. There are some *minor* issues with the above. Work is presented to a good standard. Clear relationship between different parts of the assignment. Assumptions are reasonable and explained.	There are SOME minor problems with TWO or more of the following, or major issues with ONE: -Primary and foreign keys are identified across tables -Data dictionary matches ERD -Datatypes are well-chosen -Constraints are sensible -Default data is suggested as appropriate -Field names are useful -Useful sample data is supplied Work is presented to a reasonable standard. There are some mismatches between different parts of the assignment. Assumptions may contradict case study, or be unstated.	There are SOME minor problems with THREE of the following or major problems with TWO: -Primary and foreign keys are identified across tables -Data dictionary matches ERD -Datatypes are well-chosen -Constraints are sensible -Default data is suggested as appropriate -Field names are useful -Useful sample data is supplied There are some major issues with the presentation of this work. There are mismatches between different parts of the assignment. Assumptions may contradict case study, or be unstated.	There are MAJOR problems with THREE or more of the following: -Primary and foreign keys are identified across tables -Data dictionary matches ERD -Datatypes are well-chosen -Constraints are sensible -Default data is suggested as appropriate -Field names are useful -Useful sample data is supplied Work is unstructured, assumptions are unstated or contradictory. Below standard expected at this level.	There are MAJOR problems with FOUR or more of the following: -Primary and foreign keys are identified across tables -Data dictionary matches ERD -Datatypes are well-chosen -Constraints are sensible -Default data is suggested as appropriate -Field names are useful -Useful sample data is supplied Work is very poorly presented; well-below standard expected at this level.

Appendix 5: Marking Criteria Part B

	A	B	C	D	E	F
CreateTables 15%	SQL syntax correctly used in each statement with evidence ofsuccessful execution. Proper useof a full range of constraints (data types, primary key, foreignkey, not null, unique and check, where appropriate), default values and generated keys. May be some very minor issues.	SQL syntax correctly used in MOST statements with evidence of successful execution. Good use of a range of constraints (data types, primary key, foreign key, not null, unique and check, where appropriate),default values and generated keys	SQL syntax correctly used in SOME statements with evidenceof successful execution. SOME use of a range of constraints (data types, primary key, foreignkey, not null, unique and check, where appropriate), default values and generated keys.	SQL syntax correctly used inSOME statements. SOME attributes have reasonable data types and primary keys are defined. There may be some issues with foreign keys.	There are major issues with the SQLused OR attributes and primary keys.	There are major issues with the SQL used AND attributesAND primary keys. Extremely limited.
PopulateTables 15%	SQL syntax correctly used in each statement with evidence ofsuccessful execution. A range of data values to test MOST constraints on attributes. May be some very minor issues.	SQL syntax correctly used in each statement with evidence of successful execution. There is a good attempt to use a range of data values to test SOME constraints on attributes.	SQL syntax correctly used in each statement with evidence of successful execution. There is an attempt to use a range of data values to test constraints, but this may be partial.	SQL syntax correctly used ininserting data for SOME tables created.	SQL syntax used in inserting data to SOME tables created but contain many significant errors.	Extremely limited attempt at insertingdata using SQL.
SQL Queries 30%	Correct SQL syntax to answerthe query with evidence of successful execution for ALL queries (MUST, SHOULD and COULD). May be some very minor issues.	Correct SQL syntax to answer the query with evidence of successful execution for MOST queries.	Correct SQL syntax to answer the query with evidence of successful execution for SOME queries.	SQL syntax correctly usedmainly correctly to attempt allMUST queries	SQL syntax used to attempt SOME queries, but there are some major issues and may contain some significant errors.	Extremely limited attempt at writingSQL queries
ProceduresAnd Triggers 20%	Correct SQL syntax to performALL defined actions with evidence of successful execution. May be some very minor issues.	Correct SQL syntax to perform MOST of the defined actions with evidence ofsuccessful execution. There may be a MAJOR error in any ONE of the procedures, or a MINOR error in TWO.	Correct SQL syntax to perform SOME defined actions with evidence of successful execution.There may be a MAJOR error in two of the procedures, or MINOR errors all three.	Correct type of procedural program selected for the defined functionality. Someattempt to write the code as demonstrated in lab sessions.	Some attempt to create the definedfunctionality. Limited attempt to write the code as demonstrated in lab sessions.	Very limited attempt to create the defined functionality - may be incomplete or very muddled. Extremely limited.
Application 10%	Excellent attempt at creating application	Reasonable attempt (but not completely successful) to create application	Some attempt, partially successful, at creating application	Some attempt at creatingthe application, limited success	Some attempt tocreate the application, very limited success	No attempt - will begraded as F-
Business Justification 10%	Excellent justification for how the business will be improved by the implemented database	Good justification for how the business will be improved by the implemented database	Reasonable justification for how the business will be improved by the implemented database	Some justification for howthe business will be improved by the	Little justification forhow the businesswill be improved by	No attempt - will begraded as F-