CSCI 11062 - Introduction to Database Management Systems
Department of Software Engineering
Faculty of Computing and Technology, University of Kelaniya

Assignment (Individual)

Appendices

Appendix I: Cover Page for the report.

Faculty of Computing and Technology, University of Kelaniya

Department of Software Engineering

Introduction to Database Management Systems
[CSCI 11062]

Assignment – Realizing database design process
Book Publisher System

CS/2020/024 –Weerarathna Piyumi Nadeesha

Date of submission: 22/11/2022

CSCI 11062 - Introduction to Database Management Systems

Department of Software Engineering

Faculty of Computing and Technology, University of Kelaniya

Assignment (Individual)

Appendix II: Declaration page.

Declaration

I hereby certify that this report does not contain any material previously published or written by another person except where due reference is made in the text.

(Signature of the student)

Date: 22/11/2022

CSCI 11062 - Introduction to Database Management Systems

Department of Software Engineering
Faculty of Computing and Technology, University of Kelaniya

Assignment (Individual)

Learning Outcomes covered:

- LO1: Familiarize with the basic concepts of a database system
- LO2: Design a relational database system using the ER and EER Model
- LO5: Query the database using SQL

*****This assessment contributes 10% to the Final grade.**

Tasks to be done:

Realizing the database design process

Requirement gathering

1. Select a business process that you know. Example: Food delivery system, Attendance management system, Student registration system etc. You can imagine the business/ system and a real client is not essential.

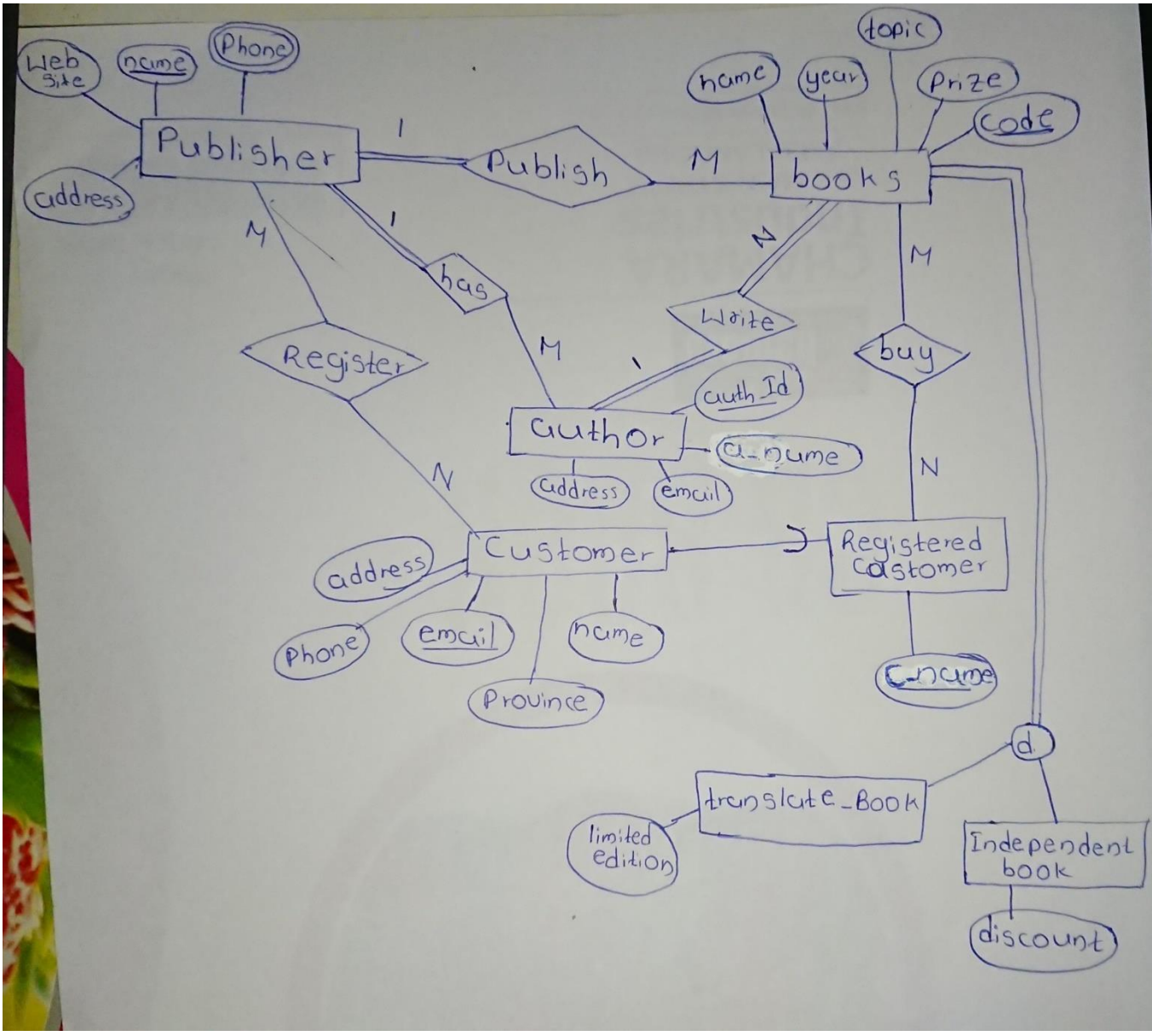
   Books Publisher System

2. Briefly describe your hypothetical business process by identifying its main functions and its data requirement for the system of the selected business (100 - 200 words)
   ***Above task 1 and 2 represent the requirement gathering stage.

Books Publisher Companies publish books that are written by authors. All authors are identified by using unique author_id. Additions to that author's name, email, address are recorded. A books publisher has several authors and who join at most one publisher. Publishers have a unique name. Additions to that have address, phone_numbers, web site are recorded. Every printed book by publisher has a name, year, topic, prize and unique code. Book should have one author. He/She may be a real author or translated author. An author should have written/translated minimum one book. Customers buy these books. For that each customer must be registered relevant publisher's web site. Each customer is identified by the using email, addition to that name, province, address, phone_num are recorded.and registered customers have a name(c_name) that given by publisher. Each publisher publishes more books. Some translated books have a type and other book(independent) have a discount.

Design the database for the system

3. Model the Data requirements using the Entity Relationship (ER) Model (There should be at least 4 entities and there relationships)



4. Convert the ER Model to derive the Relational Data Model (Schema of the Database. There should be minimum 4 relations)

PUBLISHER (name, web_site, address)

PUB_PHONE (name, phone_num)
BOOK (code, year, topic, name, prize, aauth_id, Pname)
)
AUTHOR (auth_id, a_name, email, address, pname)
TRANSLAT_BOOK (code, limi_edition)
INDEPENDENT_BOOK (code, discount)
REGISTERED_CUSTOMER (email, c_name)
CUSTOMER (email, name, province, phone, address)

5. Write SQL commands (DDL) to create database tables (for the relations identified in the relational model).

```
CREATE TABLE Publisher (
        name       CHAR (20) PRIMARY KEY,
        web_site   CHAR(50),
       address     CHAR(25),
      CONSTRAINT must_be_different UNIQUE(web_site));

CREATE TABLE Pub_Phone (
        name            CHAR (20) ,
        phone_num  INTEGER,
        PRIMARY KEY(name, phone_num), FOREIGN KEY(name) REFERENCES
        Publisher(name));

CREATE TABLE Author (
       auth_id      CHAR(6)   PRIMARY KEY,
       a_name       VARCHAR (20)  NOT NULL,
       email        VARCHAR(50) NOT NULL  UNIQUE,
       address      CHAR(25),
       pname        CHAR (20),
       FOREIGN KEY(pname) REFERENCES Publisher(name));

CREATE TABLE Book (
        Code          INTEGER PRIMARY KEY,
        year          INTEGER   CHECK (year BETWEEN 1980 AND 2020),
        topic         VARCHAR(50),
       name          CHAR(25),
       prize         REAL  NOT NULL,
       aauth_id      CHAR(6)  ,
       Pname         CHAR (20),
       FOREIGN KEY(aauth_id) REFERENCES Author (auth_id),
       FOREIGN KEY(pname) REFERENCES Publisher(name));

CREATE TABLE Translate_book (
        Code             INTEGER PRIMARY KEY,
        Limi_edition  CHAR(4)  CHECK (limi_edition= 'Yes' OR  limi_edition= 'No'));


CREATE TABLE Independent_book (
        Code         INTEGER PRIMARY KEY,
        discount     REAL  NOT NULL  CHECK (discount > 0));

CREATE TABLE Customer (
        email       CHAR(20) PRIMARY KEY,
        name        CHAR(20)  ,
       province   CHAR(10) NOT NULL,
       phone      INTEGER,
       address    CHAR(25)
 );
```

```
CREATE TABLE Registered_Customer (
        email       CHAR(20) PRIMARY KEY,
        c_name      CHAR(20)
 );
```

6. Add necessary constraints to the relations (Example: Primary key, foreign keys, Unique, Check and not null etc.)

7. Write SQL commands (DML) to insert records (minimum 3) for the created tables.

```
INSERT INTO Publisher VALUES ('Muses', 'www.musesbooks.com', 'Ganemulla' );
INSERT INTO Publisher VALUES ('Grantha', 'www.grantha.lk', 'Moratuwa' );
INSERT INTO Publisher VALUES ('Ashirwada', 'www.ashirwadabooks.com', 'Mirigama' );

INSERT INTO Pub_Phone VALUES ('Muses',0112346777);
INSERT INTO Pub_Phone VALUES ('Muses', 0114534444);
INSERT INTO Pub_Phone VALUES ('Grantha',0116767888);

INSERT INTO Book (code, name, prize, aauth_id) VALUES (10805, 'Amazonia', 1200,
#mnjul);
INSERT INTO Book (code, prize, aauth_id)  VALUES (11256, 1800, '#keshn' );
INSERT INTO Book (code, name, prize, aauth_id, Pname) VALUES (11455,  'Pet Semetary',
1800, '#cndn', 'Muses');

INSERT INTO Author VALUES ('#mnjul', 'Manjula Senarathna',     'manjula@45.gmail.com',
'Colombo 4, 'Muses' );
INSERT INTO Author (auth_id, a_name, email ) VALUES ('#cndn', 'Chandana
Gunesekara',  'chanfana@67.gmail.com');
INSERT INTO Author (auth_id, a_name, email ) VALUES ('#keshn', 'Keshan Hareshu',
'kesh@108.gmail.com');

INSERT INTO Translate_book VALUES (10805, 'No');
INSERT INTO Translate_book VALUES (11256, 'No');
INSERT INTO Translate_book VALUES (11456, 'Yes');

INSERT INTO Independent_book VALUES (00402, 0.20);
INSERT INTO Independent_book VALUES (01255,0.10 );
INSERT INTO Independent_book VALUES (11354, 0.10);

INSERT INTO Customer (email, province)VALUES ('namal@456gmail.com', 'Western' );
INSERT INTO Customer (email, province)VALUES ('Laksh@gmail.com', 'Southern');
INSERT INTO Customer (email, province)VALUES  ('jeewanta@gmail.com','Westeren');

INSERT INTO Registered_Customer (email) VALUES ('namal@456gmail.com');
INSERT INTO Registered_Customer VALUES ('Thml@630gmail.com', '#mtama' );
INSERT INTO Registered_Customer (email) VALUES ('jeewanta@gmail.com');
```

8. Write at least 3 SQL commands (DML) to show 3 types of information useful to the business users. (You should show your capability in retrieving data by joining multiple tables, using aggregate function, Having, Order by etc.)

***Minimum requirements are mentioned above. But you may use your in-depth database knowledge to design the database and demonstrate them in the report.

1. SELECT publisher.web site. Author.email FROM Publisher, Author WHERE Publisher.name Author.pname;

2. SELECT COUNT(*) AS Num_of_Customer FROM Customer;
3. SELECT topic, SUM (prize) FROM book GROUP BY topic HAVING SUM(prize) > 2000;
4. SELECT code FROM Independent_book ORDER BY discount;

CSCI 11062 - Introduction to Database Management Systems

Department of Software Engineering
Faculty of Computing and Technology, University of Kelaniya

Assignment (Individual)

Implementation

9. Create a database instance using PostgreSQL. Run the DDL and DML commands accordingly and implement your designed database.
10. Create a database dump. (Refer to https://www.postgresql.org/docs/current/backup-dump.html)

    PostgreSQL provides the utility program **pg_dump** for this purpose. The basic usage of this command is: pg_dump *your_db_name* > *dumpFileName*

Project Deliverables

1. A report (Word document) including the details of the Project Tasks that are mentioned above (from task 1 to 8). Refer to the Appendix I below to get the cover sheet for the report. Include the declaration form (Appendix II) as the second page.

2. Database dump file. (Examiner should be able to restore the dump and check)

*** You must create a zip file of deliverables and upload it to the submission link provided in the LMS.

Submission deadline

12th week (Exact date will be informed later)

IMPORTANT:

- Zero toleration for plagiarism. (If noticed, assign zero marks)
- No extensions for the deadline.

CSCI 11062 - Introduction to Database Management Systems

Department of Software Engineering

Faculty of Computing and Technology, University of Kelaniya

Assignment (Individual)

Assignment Marking Guide:

1) Report **(80 marks)**
    I. Described hypothetical business scenario (10 marks)
        a) Process functions identification
        b) Data requirement identification (entities)
    II. ER diagram (20 marks)
    III. Relational model or Schema (20 marks)
    IV. DDL statements with constraints (10 marks)
    V. DML statements – Insert (8 marks)
    VI. DML statement – Select (12 marks)
2) Database dump **(20 marks)**

----------End of the assignment----------