Homework#1

ACS575 Database Systems, Spring 2024

Dues: February 2

Instruction

- Please submit a single file named *YourLastName_YourFirstName*_ACS575_HW1.doc (.zip) for this homework.
- Organize the submission file into five subdirectories, Part I, Part II, and Part III
- Clearly number your answer, e.g., Q1 (1), Q1 (2), .. Q2, …
- Do not submit any image files of hand-drawn diagrams taken by phone/camera
- All the files you submitted should be opened by graders. Otherwise, your submission will have a score of 0.

Part I. Overview of Database Systems

Q1 The most classic paper to introduce database systems is M. Stonebraker and J. M. Hellerstein, "*What Goes Around Comes Around*". Read the paper and answer each of the following questions briefly.

(1) What is the notion of *data independence?* Why is it highly desirable?

(2) What are the key ideas behind the relational model? Why/how it improves previous model?

(3) What is the motivation of the OR (Object-Relational) model? What are the main additions in ORDB?

(4) Data can be divided to four categories: rigidly structured data, rigidly structured data with some text fields, semi-structured data, text (unstructured data). Describe each one and give an example for that.

Q2. After reading "Section 5" of M. Stonebraker et al. "One Size Fits All", give a short answer on each question following:

(1) Does one size fits all? If not, explain the reason.

(2) What components should be considered in a sensor network system with databases.

(3) Describe the characteristics of (typical) text search system which are different from conventional business-processing applications.

(4) What should be considered for scientific databases.

Q3. Name the six (/five) phases of System Development Life Cycle (SDLC), and explain the purpose and outputs of each phase.

Part II Conceptual Data Modeling

Instruction for (E)ER-diagrams

- **Naming**: Ensure that entity types, attributes, and relationship types are appropriately named.
- **Keys**: Clearly denote the key(s) for each entity type.
- **Weak Entities**: For weak entity types, identify and label a partial key.
- **Attributes**: Differentiate attributes by type (e.g., multivalued, composite, complex, required, derived) using standard notations.
- **Cardinalities**: Indicate both maximum and minimum cardinalities for each relationship.
- **IS-A Relationship**: If present, clearly denote:
  - Disjointness constraints as either distinct or overlap.
  - Completeness constraints as either total or partial.
- **Notations**:
  - Utilize **Chen's notations** for the EER diagram.
  - Avoid blending different notation sets (e.g., mixing Chen's notation for entities and attributes with Crow's foot for relationships).
  - Refrain from using relational schema notations like PK and FK.
- **Tools**: Use a database modeling tool or a drawing tool (such as MS Visio, ERwin, Rational Rose, Lucidchart,) of your choice to create your (E)ER-diagrams. Include then a copy of these diagrams in your submission file.
- Finally, if you made any assumptions while creating the EER-diagram, provide a clear explanation to ensure the grader comprehends your rationale."

Q4. Read the "**Mountain View Community Hospital**" case study provided.

(1) List all possible entity types and give a short description of each entity type.

Entity type	Description
…	…

(2) Per each entity types, list all possible attribute types. For attribute characteristics, indicate only composite, complex, multi-valued, derived, and required. If any, indicate the key attribute type(s) of the entity type. Also if available, give other specifications (e.g., domain, constraint) of the attribute type.

Entity type	Attribute type	Attribute characteristic (composite, complex, multi-valued, derived, required)	Key (Simple/Composite)	Other Specification
	…	…	…	…
…	…	…	…	…

(3) List all possible relationship types and their degree. List the participation entity types of each relationship type. Also describe the maximum cardinality and maximum cardinality of each relationship type.

Relationship type	Degree	Participation entity type	Cardinality (min, max)
…	…	…	…

(4) Develop an (E)ER-diagram on the given details of the case study.

Q5. Read the "**Musicmatic**" case study provided.

(1) Develop an (E)ER-diagram on the given details of the case study.

(2) Present the conceptual database schema of the case study using a UML class diagram. The diagram includes not only attributes but also behaviors (methods) for each entity. Include the following methods in the classes associated with them.

For the Song class, methods include:

- getDetails(): Retrieve the song's details like title, year, length, and genre.
- isSingle(): Check if the song is a single.

The Artist class methods are:

- createProfile(): Set up a new artist profile.
- getBio(): Fetch the artist's biographical data.
- listSongs(): List all songs by the artist.

For only business users, the following methods include:

- registerBusiness(): Register a new business account with a VAT number.
- getUploadedSongs(): Retrieve a list of songs uploaded by the business user.

Part III Logical Data Modeling – Relational Database Design and Normalization

Instruction for Relational Schema

In your relational database schema, ensure all primary key are prominently displayed. If present, identify foreign keys and indicate their referencing relations. If present, indicate required attributes. For this homework, there's no need to specify other domain constraints, such as attribute type or value length. Present your relational model using one of the two methods below:

Output option 1: For a graphical relational schema diagram:

- Use any drawing or data modeling tool, like Visio., ERDPlus, draw.io, or Lucidchart. For instance, MS Visio supports relational schema notations.
- Designate primary keys using 'PK' and foreign keys with 'FK'.
- Employ an arrow line to indicate the relation a foreign key references.
- Refrain from using crow foot notations (e.g.,) in the relational schema.
- Highlight required attributes with bold characters

Output option 2: For a textual presentation of each relation:

- Represent relations as $R(A_1, A_2, \ldots, A_n)$, where R is the relation name, A_{i1} is an attribute.
- Underline the primary key attributes.
- Use *italic* for foreign keys.
- Emphasize required attribute using **bold** characters.

For instance:
EMPLOYEE(SSN, **ename**, address)
EMP-PHONE(PhoneNr, *SSN*)

* Note*: SSN* in EMP-PHONE is a foreign key referencing EMPLOYEE's SSN.

Q6. Translate the (E)ER-diagram from Q5 (1) into a set of relations suitable for the database of "**Musicmatic**".

Q7. Review the case study on the "**Local Library Management System**". Based on the details provided, answer the following:

(1) Determine the normal form in which the "library" table currently resides.

(2) Enumerate all functional dependencies found in the "library" table. Note: In this scenario, a book has only one author.

(3) From the functional dependencies, identify any partial dependencies.

(4) From the functional dependencies, identify any transitive dependencies.

(5) Transform the current "library" relation into relations in the third normal form (3NF). Clearly indicate the primary key for each new relation. Additionally, specify any foreign keys and their referencing table.