Unit-4: Design.

DATE

Design is one of the phases of SDLC. It involves
→ Designing Databases
→ Designing forms and reports
→ Designing Interfaces and Dialogues.

4.1 Designing Databases

* The database design process typically involves identifying the data that needs to be stored, determining the relationship between the data, and creating a logical model of the database using a modeling tool.

→ The main objective of database designing is to first build the logical model of the database and then it can be translated into a physical database using a DBMS.

→ The process of database design includes.

i) Logical design. (defines what system contains)
→ Based upon the conceptual data model. It has 4 steps
a) Develop a logical data model for each known UI
b) Combine normalized data requirements from all UI into one logical database model.
c) Translate the conceptual ER model into normalized data requirements
d) Compare the logical db model with translated ER model & produce final logical db model for the application

ii) Physical design
→ Based upon the logical database design (defines how system works)
a) Choosing storage format for each attribute from logical db. model
b) Grouping attributes from the logical DB model into physical records
c) Selecting proper secondary memory for storing records
d) Selecting proper structures (ex. tables) for storing data so that it can be easily accessed.

classmate

PAGE

In summary, logical design is based upon the conceptual model of data i.e. it defines what system contains & relationships between data, normalizing the data, and so on. It doesn't concern with how the data will be stored or where it will be stored physically. ex: ER diagram design. is a conceptual then we convert it into logical (tables)

whereas, the physical database involves translating the logical DB design of the database onto a physical media using hardware resources and software systems like DBMS. It defines how the system works by specifying the storage formats, structures, indexes and access methods that will be used to implement the database. The logical design is used as the blueprint to create the physical design

* The outcomes and deliverables of this stage are the logical & physical database designs, database schema (a detailed description of db structure, including the tables, field, keys, indexes and constraint).

* Relational Database Model

→ This model describes the database a collection of relations where a relation is nothing but a ~~value of~~ table of values

→ In this model, data are stored as tables, where every rows represent an entity and every columns represent an attribute.

→ The tables are also called relations in relational model

→ The relationship between tables is maintained by using a primary key & foreign key.

DATE

Terms in relational model:

- **Table:** Data is stored in table format. Table is used because it can represent the data in the simplest form possible making data retrieval very easy.
- **Attribute:** Each column in a table. Any relation have definite properties called as attribute
- **Tuple:** Rows of table represent the tuple which contains the data records
- **Domain:** Set of values which is indivisible. i.e. It is a set of acceptable values that a column is allowed to contain. Ex: value of date of birth should be > 0.
- **Relation:** A relation in relational data model represents the respective attributes & relationships between them.

Ex: Student (relation / table)

Std_id	Std_name	Std_address
x	x	x
x	x	x

attributes

record or tuples

→ There are four basic update operations performed on relational database model: Insert, Delete, Modify, Select. (Constraints must be followed while using these)

→ A well structured relation must have the following characteristics

→ Advantages of using relational model:
Simplicity, Structural independence, Easy to use, data independence, Scalable

classmate

PAGE

→ Disadvantage:

- Few relational databases have limits on field lengths which can't be exceeded.
- Relational db can sometimes become complex as the amount of data grows.

→ Characteristics of well structured relation.

- Each relation in a db must have a distinct or unique name which would separate it from other relations in the db
- A relation mustn't have two attributes with same name
- Duplicate tuples mustn't be present in a relation
- Each tuple must have exactly one data value for an attribute
- Tuples & relations don't have to follow any significant order as the relation is not order sensitive

DATE

* Normalization

→ Normalization is the process of reducing a table (relation) into multiple simple tables for minimizing the redundancy of data. It divides the larger tables into smaller tables and links them using relationship.

→ Normal forms → Normalization works through a series of steps called normal forms. The normal forms apply to individual relations. A relation is said to be in particular normal form if it satisfies its constraints. Normal forms indicate the degree to which it has been normalized.

→ Advantages of Normalization
- Helps in maintaining data integrity
- Helps in simplifying the structure of the table.
- Eliminates insertion, deletion, & update anamolies
- Eliminates redundant data so, less storage is required
- Better understanding of of data

→ Denormalization is the process of joining the normalized tables to form a lower normal form.

→ Super key: Attributes or a set of attributes that is used to uniquely identify all other attributes in a relation. All super keys can't be candidate but reverse is true.

→ Candidate key: uniquely identify all other attributes (minimal super key)

→ Prime attribute: attributes which are candidate keys

→ Non-prime attribute: which are not candidate keys

→ Functional dependencies: (FD)

If R is a relation with attributes X and Y. Then, the functional dependency between the attributes are represented as X→Y, which specifies Y is functionally dependant on X. For each value of X, there is exact one value of Y & one value of Y can have multiple value of X.

Ex. Emp_id | Emp_name | Emp_dept | Salary

Here, If we know the value of Emp_id, we can determine Emp_name, Emp_dept, Salary, Hence, we can say that Emp_name, Emp_dept, Salary are functionally dependent on Emp_id

- Partial F.D → Given X→Y, Y is said to be partially F.D on X If 'Y' is functionally dependent on X and also all of X's subsets. ie. If removal of some attributes from X, the dependency still holds.

- Transitive F.D → A F.D is said to be transitive If it is indirectly formed by 2 FDs. ie X→Z is T.D If following F.D are true. X→Y, Y→X, Y→Z then X→Z

→ Normal forms:

1. First Normal form

A relation is said to be in 1NF If and only If

- Each cell has single value. ie. multivalued attribute is not allowed in a single cell
- There are no duplicate rows in the table
- Data for a particular column are of similar kind

Ex:

C_id	Color
1	red, green
2	yellow

=>

C_id	Color
1	red
1	green
2	yellow

DATE

2) Second Normal Form

A relation is said to be in 2NF if

- It is in 1NF
- No partial dependencies exist between non-prime (non-key) attributes and prime (key) attributes
- Its main aim is to ensure that all information in a relation is only about one thing

Ex:

T_Name	Tsubject	Age
Sushil	C	40
Sushil	OOP	40
Balk.	DS	42
Balk	CG	42
Shiva	Stat	43

⇒

T_Name	Tsubject
Sushil	C
Sushil	OOP
Balk	DS
Balk	CG
Shiva	Stat

Tnam	Tage
Sushil	40
Balk	42
Shiva	43

3) Third Normal form

A relation is said to be in 3NF if

- It is in 2NF
- No transitive dependencies exist between non-key & key attributes. ie. no non key attributes is functionally dependent on key attribute.

Roll no	Game	Fee
1	Basketball	500
2	Basketball	500
3	Cricket	600
4	Cricket	600

⇒

Roll	Game
1	B
2	B
3	C
4	C

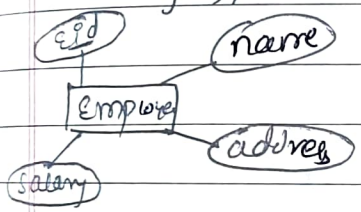
Game	Fee
B	500
C	600

Transforming ER diagrams into relations

→ ER model is a description of real world entities.
→ It represents the conceptual level of database design meanwhile the relational schema is the logical level for the database design.
→ Transforming E-R diagrams into relations can be accomplished in four steps:

1) Represent entities

- Each entity type in the ER diagram becomes a relation (table)
- The primary attribute of the entity type becomes the primary key of the relation & other attributes of the entity type become non-primary key attributes of the relation



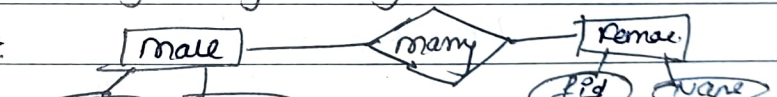
⇒ Employee

Eid	name	salary	address

2) Represent relationship

- 1-1 relation:- we set primary key of any one of the entity type as foreign key.

Ex:



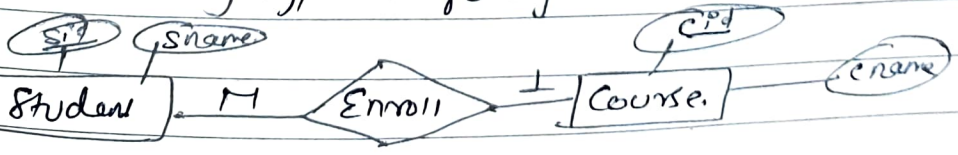
⇒

Mid	Name

Fid	Mid	Name

- 1-N relation:- Include primary key of one side entity type to many side entity type as foreign key.

Ex:



⇒ Student

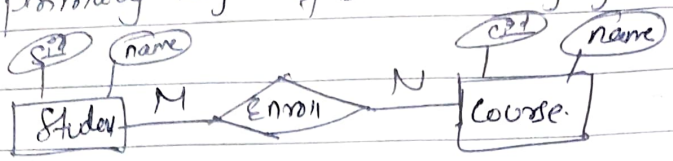
Sid	SName	Cid

Course.

Cid	name

DATE

- M:N relation: separate relation is created for the primary keys of both entity type.



=> Student

S_id	name

Course

C_id	name

Enroll

S#d	C#d

3) Normalize the relations.

- The relationship created by representing Entities & representing relationship may have redundancy
- So, these relationship are normalized to make them well structured

3) Merge the relationship

- If there are redundant relations (two or more relations that describe the same entity type), that must be merged & re-normalized to remove the redundancy

Ex: Employee1 (Emp_ID, Name, Address, Phone)
Employee2 (Emp_ID, Name, Address, Jobcode, No. of years)
Because these two relations have same primary key & describe the same entity, they should be merged into one relation.

Employee (Emp_ID, Name, Address, Phone, Jobcode, No. of years)

Physical file & database design

→ The purpose of physical database design is to translate the logical description of data into the technical specifications for storing and retriving data.

→ We create a physical ~~file or~~ database for storing data that will provide adequate performance & ensure database integrity, security and recoverability.

→ A physical file is a named portion of secondary memory (e.g. hard disk) for the purpose of storing physical records

→ Designing physical files & databases requires certain information that should have been collected during prior SDLC phase

- Normalized relations & definitions of each attribute
- Descriptions of when & where data are used.
- Description of technologies used for implementing the files & database.

* Designing Fields

→ Field is the smallest unit of data in database. Corresponding to a simple attribute from the E-R diagram

→ Field design process includes.

i) Choosing data types. ⇒ data types should be of
   - minimum storage possible
   - represent all possible values
   - improve data integrity
   - support all data manipulations

ii) Coding techniques

iii) Controlling data integrity

iv) Handling missing data

DATE

* Designing Physical Tables

A physical table is a named set of rows and columns that specifies the fields in each row of the table. A physical table may or may not correspond to one relation. The normalized tables (relations) posses well-structured relations whereas the design of a physical table has two goals different from those of normalization.

i) Efficient use of secondary storage : It relates to how data are loaded on disks. It depends on factors such as operating system parameters, outside the control of each DBS.

ii) Data processing speed : Data are most efficiently processed when they are stored close to one another in secondary memory, thus minimizing the input/output (I/O) operations that must be performed.

12 Designing Forms & Reports

I Introduction (process of designing)

→ Designing forms and reports involves creating the user Interface (UI) elements that will be used to interact with and display data stored in the database.

→ This includes determining the layout and design of forms, which are used to input & edit data, as well as the layout and design of reports, which are used to present the data in a summarized or detailed format.

→ The forms & reports should be designed in such a way that they are easy to understand & use, with clear and consistent layout & intuitive navigation.

→ An Example: Every input form is associated with a data flow entering a process on a DFD and every output form or report will be a data flow produced by a process in a DFD.

→ Designing forms & reports is a user-focused activity that typically follows prototyping approach. We understand the intended users & several wh-questions must be answer

- Who will use form/report?
- What is the purpose of form/report?
- When is the form/report needed or used?
- Where & whom to deliver the form/report & use?
- How many people need to view the form/report?

→ Gaining the understanding of these questions is required 1st step in the creation of any form or report. After collecting the initial requirements, you structure & refine this information into a prototype. Finally, you ask users to review & evaluate prototype. Then make changes if needed.

DATE

* Deliverables & outcomes

Design specifications is the major outcome. It has three sections

a) Narrative overview : This section contains a general overview of the target users, tasks, system and environment in which the form or report will be used. The purpose is to explain to those who will actually develop the final form, why this form exists, & how it will be used so that they can make the appropriate decisions

b) Sample design: It is the simple hand made design or using CASE tools design of forms & reports. It's just a prototype & will keep on improving by the feedback of users.

c) Testing & usability assesment : Depends on speed, accuracy and satisfaction

* Formatting Forms & Reports

1) General formatting guideline:

i) Meaningful titles :- Should be clear & specific describing content & use

ii) Meaningful information :- only necessary information with no need of modification should be included by form

iii) Balance of layout :- Adequate space & margins and all data & entry fields must be labelled

iv) Easy navigation → Clearly show how to move forward & backward, show where you are (page no.) & notify a user when on a last page of a multipage sequence.

classmate

PAGE

2) Guidelines for displaying contents

i) Highlighting information: It will enhance the appearance of output. It can be carried out using different methods such as color difference, intensity difference, underlining, font and size differences, etc.

ii) Using color: Use of appropriate colors while designing has several advantages like. It strikes the eye, draws attention to warnings & use of color in graphs & charts helps in better understanding, etc.

iii) Displaying text: We should use appropriate punctuations wherever required. The text should be properly spaced & there should be blankline between paragraphs

iv) Designing tables & lists: we should use meaningful labels to all columns & rows & separate labels from other information by using highlighting. Displayed data should be sorted in order

Paper based vs Electronic reports:

Paper based reports	Electronic reports
→ They are stored locally in filing cabinets	→ Stored in file systems of computer
→ Editing, & copying is difficult	→ Easier
→ For multiple users, each needs a copy of the report.	→ Multiple users may access single report simultaneously
→ It can be delivered by mail or any other physical method	→ It can be delivered by networks, disks, flash memory, etc.
→ Paper reports take time for searching	→ Electronic reports are faster to search.

DATE

* Assessing Usability

Usability typically refers to the following three characteristics:
i) Speed: Can you complete a task efficiently?
ii) Accuracy: Does the system provide what you expect?
iii) Satisfaction: Do you like using the system?

General design guideline for usability of forms & reports

Factor	Guideline for achieving usability
Consistency	→ Consistent use of terminology, formatting titles & navigation within & across outputs
Organization	→ Text & data should be aligned and sorted for efficient navigation & entry.
Clarity	→ Outputs should be clear to the user.
Format	→ Information format should be consistent between entry and display
Flexibility	→ Information should be viewed & retrieved in a manner that is most convenient to the user.

* Measures of usability →

Designing Interfaces and Dialogues

→ It is the process of defining the manner in which humans and computers interact with each other & interchange information.

→ During this phase, interface & dialogue designers work with user experience (UX) designers to create prototypes of the interfaces & dialogues. These are used to visualize how the final product will look & function, & to gather feedback from users. & then iteratively keep on making refinements.

→ This process is parallel process of designing forms reports

→ The measures of usability are:

i) Learnability: How difficult is it for a user to perform the task for the first time

ii) Efficiency: How quickly can users perform tasks?

iii) Error rate: How many errors might a user encounter, and how easy is it to recover from those errors?

iv) Memorability: How easy is it to remember how to accomplish a task?

v) satisfaction: How enjoyable is the system to use?

→ In general sense, interfaces are the visual & interactive elements that allows users to navigate & perform tasks Ex: buttons, menus, icons, . whereas, dialogue refers to the conversations or interactions that occur between user & system Ex: prompts, messages that provide information, Error notice, etc on.

→ The deliverables & outcome from Designing interfaces & dialogs is the design specification similar to forms & report.

- Narrative overview
- Sample design
- Testing & usability assessment (tested to proper users & they are proper using it)

DATE

* Interaction methods and devices

o Methods of interaction

Methods used to interact with the system is the basic to make during the designing of UI. It includes

i) Command language → Users enter command to perform operations & interact with the system. It requires users to remember all syntax & operations - It is good for experienced users but it is rarely used these day

ii) Menu interaction → A menu is a list of options. When an option is selected by the user, a specific command or some operation is done. They are widely used

iii) Form interaction → It allows users to fill the form to get required information & to perform tasks

iv) Object-based Interaction → It is implemented through the use of icons. They may look like a button & can perform relevant action to a form like save, edit, cancel, ask for help, etc.

v) Natural language interaction → It means interacting with the system to take input & produce outputs in completely human language. It is one of the branch of AI research studies.

Designing Interfaces

→ User interface is the front-end application view to which user interacts in order to use the software.
→ UI can be graphical, text-based, audio-video based depending upon the underlying hardware & software combination
→ The UI must be Attractive, simple to use, Responsive in short time, clear to understand, consistent on all interfaces screens
→ Designing Interfaces include.

1) Designing Layouts: Standard formats similar to paper-based forms and reports should be used. users should be able to move freely between fields, & so on. Flexibility & consistency are primary goals. screen navigation should be left-to-right.

2) Structuring data entry
- Always provide default values when appropriate
- Never require data that can be computed. (eg. Age from DOB)
- Always place the caption & placeholder in the fields.
- Provide formatting Examples.

3) Controlling data input
- One objective of interface design is to reduce data entry errs.
- Designer should properly guide users to avoid, detect & correct data entry mistakes
- There can be various data errors like: Appending (adding extra characters to a field), Truncating (Loosing characters from a field), Transcipting (Entering invalid data into a field).

4) Providing feedback & help.
- Feed back about what is going on the system
- Provide help & error messages.

DATE

* Designing Dialogues

→ Dialogue is a sequence in which information is displayed to & obtained from the user.

→ For designing dialogues there are three steps

i) Design a dialogue sequence
- Define the sequence
- Have a clear understanding of the user, task, technological & environmental characteristics
- Creating dialogue diagram using box & diagrams.

ii) Building prototypes & assessing usability

* Designing Interfaces & dialogues in graphical environment

For designing interfaces & dialogues you should

- Become an expert in the GUI environment ; understand how GUI applications have been designed.
- Gain an understanding of the available resources & how they can be used. & become familiar with standards for GUI menus and form

Questions asked.

8. How do you format forms & report? Explain general guideline for formatting forms & report. (2076 (new) - 5 marks)

9. What is the purpose of database design? Compare logical design with physical design. (2078 - 5 marks)