TOPIC 2 in IM 1

COURSE NUMBER: **CC 105**
COURSE TITLE : INFORMATION MANAGEMENT 1

COURSE LEARNING OUTCOME

- *Analyze an existing database system with respect to quality issues: Reliability, scalability, efficiency, effectiveness and security.*

STUDENT LEARNING OUTCOMES

1. *Discuss the quality issues of database systems and its relevance to our daily transactions.*

LEARNING CONTENT:

1. Database Management System (DBMS)
2. XAMPP
3. DBMS Environment
4. Roles in the Database Environment

INTRODUCTION

Welcome to the second week session of this course. In every organization that use a computerized system, there should be a powerful software that manage all the activities of the system especially the storing, defining, manipulating, and sharing of data to multiple users simultaneously. In our course, we are going to use MySQL as our DBMS.

A. LESSON CONTENT

Unlocking Definitions:

XAMPP is a package software created by apache friends. A group of programmers who let their creation be used by students in running their web project without internet connection. When you install this into your device and once it is running, your device will then become a localhost. Let us try to define the meaning of the acronym XAMPP.

X stands for cross-platform. This software package is a cross-platform package that could run in any platform whether you have a Linux, Windows, or Mac system.

A stands for apache. This is a server software that needs to run into your device so that it will become a localhost or local server that acts as the storage of the data.

M stands for MySQL. This is the DBMS that used in this package. In every database that you need to create, you need a DBMS.

P stands for PHP. This is a scripting language that allows you to create a dynamic web pages.

P stands for Pearl/Perl. Another scripting language that allows you to create dynamic web pages.

Database Management System (DBMS) – is a collection of program that enables users to create and maintain a database. It is a *general purpose software system* that facilitates the processes of ***defining***, ***constructing***, ***manipulating***, and ***sharing*** databases among various users and applications.

- **Defining** a database involves specifying the table name, column names, data types, and constraints of the data to be stored in the database. The database definition or descriptive information is also stored in the database in the form of a database catalog called meta-data.
- **Constructing** the database is the process of storing the data on some storage medium that is controlled by the DBMS.
- **Manipulating** a database includes functions such as querying the database to retrieve specific data, updating the database to reflect changes in the mini-world, and generating reports from the data.
- **Sharing** a database allows multiple users and programs to access the database simultaneously.

DBMS provides the following facilities:

1. It allows user to define a database through a DDL or data definition language
2. It allows user to insert, update, delete, retrieve data from the database through DML or data manipulating language (SQL)
3. It provides controlled access to the database like;
    - Security System, which prevents unauthorized users from accessing the database.
    - Integrity System, which maintains the consistency of the stored data
    - Concurrency Control, which allows shared access of the database
    - Recovery Control System, which restores the database to a previous consistent state following a hardware or software failure.
    - User-accessible Catalog, which contains description of the data in the database.
4. It provides view mechanism which allows user to have his or her own view of the database. View offers the ff. benefits:
    - Reducing complexity, by letting the user see the data in the way they want.
    - Provides level of security, can be set up to exclude data that some users should not see.

- Provides a mechanism to customize the appearance of the database.
- Can present a consistent, unchanging picture of the structure of the database even if the underlying database is changed.

Database Terminologies:

Application programs – accesses the database by sending queries or requests for data to the DBMS.

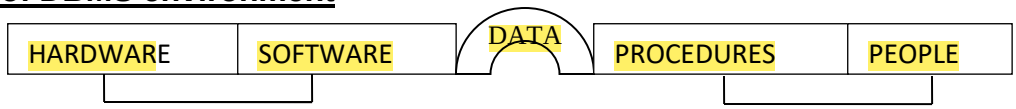
Query – causes some data to be retrieved.

Transaction – may cause some data to be read and some data to be written into the database. The concept of transaction has become central to many database applications. It is an executing program or process that includes one or more databases accesses, such as reading or updating of database records. Each transaction is supposed to execute a logically correct database access if executed in its entirety without interference from other transactions. The DBMS must enforce several transaction properties, like

1. **Isolation property** ensures that each transaction appears to execute in isolation from other transactions, even though hundreds of transactions may be executing concurrently.

2. **Atomicity property** ensures that either all the database operations in a transaction are executed or non are.

Components of DBMS environment



Hardware – the DBMS and the applications require hardware to run. It can range into personal computer, mainframe or a network of computers. It requires an amount of main memory and disk space to run. The part of the DBMS that manages and controls access to the databases is known as the **backend**. The part of the DBMS that interfaces with the user with several computers at different locations is known as the **frontend.** In client-server architecture: the backend is the server, the frontend is the client.

MULTI-USER DBMS ARCHITECTURE

- Teleprocessing- the traditional architecture for multi-processing systems was teleprocessing, where there is one computer with a single CPU and a number of terminals.
- File-Server- in a file server environment, the processing is distributed to a network such as LAN. The file-server holds the files required by the application and the DBMS. However the application and the DBMS run on each workstation, requesting files from the file-server when necessary. It generates a significant amount of network traffic, which can lead to performance problem. It acts as a shared hard drive.
- Client-Server- to overcome the disadvantages of the first to architecture, the client-server was developed. It refers to the way in which software

components interact to form a system. As the name suggest, there is a client process which requires some resources, and server, which provides the resources. There is no requirement the client and serve must reside on the same machine. Server holds the database and the DBMS. Clients manages user interface and runs the applications. **Alternative Client-Server Topology: *a. single server*, *single client* *b. multiple clients, single server,* *c. multiple clients, multiple servers***

Advantages include:
-wider access to existing database
-increased performance
-possible reduction of hardware costs
-reduction in communication costs
-increases consistency

Software – it comprises the DBMS software and the application programs, together with the operating system, including software if the DBMS is being used over a network. Typically, application programs are written in a third-generation programming language such as C or fourth-generation language. The target DBMS may have its own fourth-generation language tools that allow rapid development of applications through the provision of non-procedural query languages, reports generator, forms generator, graphics generator, and applications generator. Using 4GL tools can improve productivity and produce programs that are easier to maintain.

Data – the most important components of DBMS environment. It acts as the bridge between the machine components and the human components. The structure of the database is called schema.

Procedures – refer to the instructions and rules that govern the design and use of the database. The user of the database and the staff that manages the database require documented procedures on how to use or run the system. These may consist of instructions.

People – the final component is people involved with the system.

Roles in the Database Environment (or the actors of the scene)

Database and DBMS are corporate resources that must be managed like any other resource. One of the components of DBMS environment is the people. There are distinct type of people that participate in the environment.

- **Data and Database Administrators**
  - ***Data Administrator (DA)*** is responsible for the management of the data resources which includes database planning, development and maintenance of standards, policies, procedures, and conceptual logical design.
  - ***Database Administrator (DBA)*** is responsible for the physical realization of the database, including physical database design, and implementation, security and integrity control, maintenance of the operational system, and ensuring satisfactory performance for the applications and users. This

role is more technically oriented requiring detailed knowledge of the target DBMS and the system environment.

- **Database Designers**- there are two types of database designers; logical designer and physical designer.
  - ***Logical Database Designer*** is concerned with identifying the data (that is the entities and the attributes), the relationships between the data, and the constraint of the data that is to be stored in the database. This person must have a thorough and complete understanding of the organization's data and its business rules. Business rules describe the main characteristics of the data as viewed by the organization.
  - ***Physical Database Designer*** takes the logical model and decides how it is to be physically realized. This involves:
    - Mapping the logical data model into a set of tables and integrity constraint
    - Selecting a specific storage structures and access methods for the data to achieve good performance for the database activities.
    - Designing any security measures required on the data

- **Application programmers**- provider of the required functionality for the end-users. The creator of the application program use in DBMS operations for the users. Implement these specifications as programs then they test, debug, document, and maintain canned transactions.

- **System Analyst** determines the requirements of end-users, especially naïve and parametric end users, and develops specifications for canned transactions that meet these requirements. Analyst and programmers are commonly known to us as a software developer or software engineer.

- **End-users** – these are the clients of the database. The database has been designed and implemented, and is being maintained to serve their information needs. End-users can be classified according to the way they use the system:

  - ***Casual end users*** occasionally access the database, but they may need different information each time. They use sophisticated database query language to specify their request and are typically middle or high-level managers or other occasional browsers.

  - ***Naïve or parametric end users*** are typically unaware of the DBMS. They access the database through specially written application programs, which attempt to make the operations as simple as possible. Their main job function revolves around constantly querying and updating the database, using standard types of queries and updates called **canned transactions** that have been carefully program and tested. The tasks that such end

users perform are varied:

- Bank Tellers check account balances. Post withdrawals, and deposits
- Reservation Clerks for airlines, hotels, and car rental companies check availability for a given request and make reservations.
- Clerks at receiving station for shipping companies

- ***Sophisticated users*** are familiar with the structure of the database and the facilities offered by the DBMS. This kind of user may use a high-level query language such as SQL to perform the required operations. These include engineers, scientists, business analyst, and others who have familiarity of the DBMS.

- ***Standalone users*** maintain personal database by using ready-made program package that provide easy-to-use menu-based or graphic-based interfaces. An example is the user of tax package that stores a variety of personal financial data for tax purposes.

The workers behind the scene

1. DBMS system designers and implementers- design and implement the DBMS modules and interfaces as a software package.
2. Tool Developer- design and implement tools. Tool is software packages that facilitates database modeling and design, database system design, and improve performance.