



Zagazig University
Faculty Of Computers And Informatics
Information System Department

Question Pool System 2014-2015

Under Supervision:-
Prof. Dr. Ayman Mostafa

Teaching Assistant:-
Eng. Mohamed Maher

Project Team Work

Project Team Work

- 1. Hossam Baz Ali Hussien**
- 2. Mostafa Hussien Mohamed Sopieh**
- 3. Heba Salah El-Din Ahmed**
- 4. Aya Salah Bahgat Mostafa**
- 5. Aya Saeed Ali**
- 6. Radwa Raafat Hussein Mohammed**



Frist of all see deeply thank "Allah" for being kind, helpful and supporting to us during our life and specially to achieve this project.

Also, we would like to prevent thankfulness to our families for giving us their attention, supporting and carefulness to accomplish this project.

We express our deep since of gratitude and sincere to our supervisors :

Prof. Dr. Ayman Mostafa

Eng. Mohamed Maher

For their guidance, valuable suggestions, generous help and careful encouragement given through this project.

Finally, we hope that our project obtain your attention and specification.

Thanks

Team work Project

Table Of Contents

Chapter (1)Introduction

Introduction to QPS System	8
----------------------------	---

Chapter (2) Information System

2.1.What is an Information System	13
2.2.Basic Components of Information System	13
2.3.Components Of Information System	14
2.4.Information System Development	14
2.5.System Development Cycle	15
2.6.The Data Flow Diagram	16
2.7.Functional Decomposition	25
2.8.Context Diagram	26
2.9.Entity Relationship Diagram	28
2.10.System Design and Construction	32

Chapter (3) Normalization

3.1.DataBase Normalization	39
3.2.Normalization	40

3.3. First Normal Form	42
3.4. Secound Normal Form	42
3.5. Third Normal Form	42
3.6. Database Tables And Normalization	43
3.7. The Need For Normalization	44
3.8. The Normalization Process	45
3.9. First Norm Form	47
3.10. Secound Norm Form	48
3.11. Third Norm Form	53
3.12. Boyce-Codd Norm Form	53

Chapter (4) QPS Analysis

4.1. QPS ERD	55
4.2. QPS DFD	56
4.3. Functional Decomposition	58
4.4. Context Diagram	59
4.5. Oracle Database	59
4.5.1. Creating Database	60

Chapter (5) How To Run QPS Project

5.1.How to run QPS	70
5.2.The Employee	70
5.3.The Student	71
5.4.The Instructor	71

Chapter (6) Design Of QPS

5.1.Design of QPS	74
5.1.1.Design Include	74
5.1.2.Design Input Forms	74
5.1.3.Design Output Forms	86
5.1.4.Design User Interface Forms	88

Chapter (7) Network

7.1.Setting up Windows Server 2003 enterprise edition	92
7.2. To Install or reinstall the Windows Server 2003 Administration Tools	92
7.3.Setting up network Servers	94
7.4.Configuring the network Card	101
7.5.Adding computer to a secured wireless network	102

7.6.Configure a network card Windows 2003	102
7.7.Configure a network card in windows XP	103
7.8.Configure a network card in windows 95/98/ME	104
7.9.Adding Computer to the domain	104
7.10.Running the Router setup Wizard	105
7.11.The LAN Settings screen is used to specify the LAN IP address of the router, and to configure the DHCP server	107
7.12.Assign Or Unassigned an IPSec Policy on a computer	114
7.13.Installing Oracle 9i data base server	124

Introduction

→ Chapter 1 :

Introduction

1.1:in paper and pencil tests, the instructors prepare a number of questions according to the syllabus given to the students.

According to the time available to answer the test, the number of questions is prepared to suit that time. The instructor then photo copies the test according to the number of students to be tested. In the time of test, the instructor gives a copy from the test to each student asking them to answer it. At the end of the time of the test, the instructor collects all the copies of the test from the students. Then he/she starts to evaluate the answer of each student to assess the grade required. Then, the final result is announced to students. If the test is a final test it will be passed to some reviewers to check up the grades assessed by the instructor of the topic.

1.2:When analyzing the procedures of the paper and pencil tests, one can very easily finds out the following drawbacks:

- a-** The heavy burden on the instructor each time a test is required because he/she has to set and think of once more on the questions to be include on the test and then he/she writes them down on papers.
- b-** The cost of test papers to be copied to be given to students to answer.
- c-** The procedures required to keep the test papers safe before giving them to students to answer and after evaluating the answers to students.
- d-** Because of the time the students will take to write down the answers, this neceditates that a small number of questions should be include in the test which means that sometimes not all the points of the given syllabus are covered in the test.

e- A number of observers who are needed to keep an eye on the students during answering the test to prevent bad works of some students (as cheating from one another or using external resources such as papers or mobiles...etc.

f- The time needed by the instructor to evaluate the test has a great effect especially if the test is held at the end of the period assigned for doing tests and the date required to announce the final results is short. In this case may be and we say may be the evaluation will not reflect the exact evaluation. This is not an accusation but it is a possibility because of the burden on the instructors, and we appreciate the work of all our instructors.

g- We all are human beings and we like and dislike things.

This means that may be during the test evaluation the instructor is biased to certain student. If this bias is positive the answer of this student will be highly evaluated, contrast to if the bias is negative.

h- There is also, another problem with the paper and pencil test, this is the time wasted to distribute the test papers to students to answer it. The wasted time too, taken by students to review and clarify the bad typed question or the mysterious ones.

1.3:Computers and computers network may be used to remove and get rid of the problems gained by using paper and pencil tests. By using computers networks for example the test is passed directly to each student simultaneously. This saves the time wasted for distributing the test paper to each student.

1.4 : Using computers each student can know his/her grade after finishing answer. Computer can evaluate the answer by comparing the student's answer with the correct answer stored in its storage memory.

1.5 :This project presents the Questions Pool System QPS.

The system consists of a computers wireless network and a database management systems base on oracle database. This database is used to build the questions pool and to handle all questions and evaluation of answers.

1.6 :In Chapter one Which is a small one , a definition of QPS is given the benefits of using computers in management tests are presented.

1.7:Chapter two introduces the analysis of information system.

1.8 :a brief review of Normalization of the project is presented in chapter three.

1.9:In chapter four we presented the analysis of whole project.

1.10: In chapter five we give a review how to run the QPS.

1.11:The logical design of the QPS using oracle presented in chapter six.

1.12:We introduce in chapter seven the computer network.

End of Chapter 1

Thanks



→ Chapter 2 : **INFORMATION SYSTEM**

2.1 :What is an Information System?

- An **information system** is an arrangement of people, data, processes, interfaces, networks, and technology that interact for the purpose of supporting and improving both day-to-day operations in a business (sometimes called **data processing**), as well as supporting the problem solving and decision making needs of management (sometimes called **information services**).

2.2:BASIC COMPONENTS OF INFORMATION SYSTEM

Every system can be considered a black box, which has input, process, files and output.

- **Data input**

Data input to a computer must be structured (organized) so that they can be accepted by the computer data are normally grouped in to sets or chains called records.

- **Data stored**

Data can be stored in a variety of ways and in a variety of structures.

- **Output data**

Output data are that are read by people. As such, they must be presented to the reader in a format and structure that he can use.

- **File maintenance**

File maintenance is the processing of input data to update the data stored in the file.

- **Output production**

In output production, programs or modules take data from a file or files and produce “reports,” such as video screens, invoices checks, control reports, and files.

- **File / data base content and structure**

It also follows that changing the file in content or organization has a direct impact on file maintenance and output production.

2.3 : Components of information systems

An information system is essentially made up of five components hardware, software, database, network and people. These five components integrate to perform input, process, output, feedback and control.

Hardware consists of input/output device, processor, operating system and media devices. Software consists of various programs and procedures. Database consists of data organized in the required structure. Network consists of hubs, communication media and network devices. People consist of device operators, network administrators and system specialist.

2.4 :INFORMATION SYSTEM DEVELOPMENT

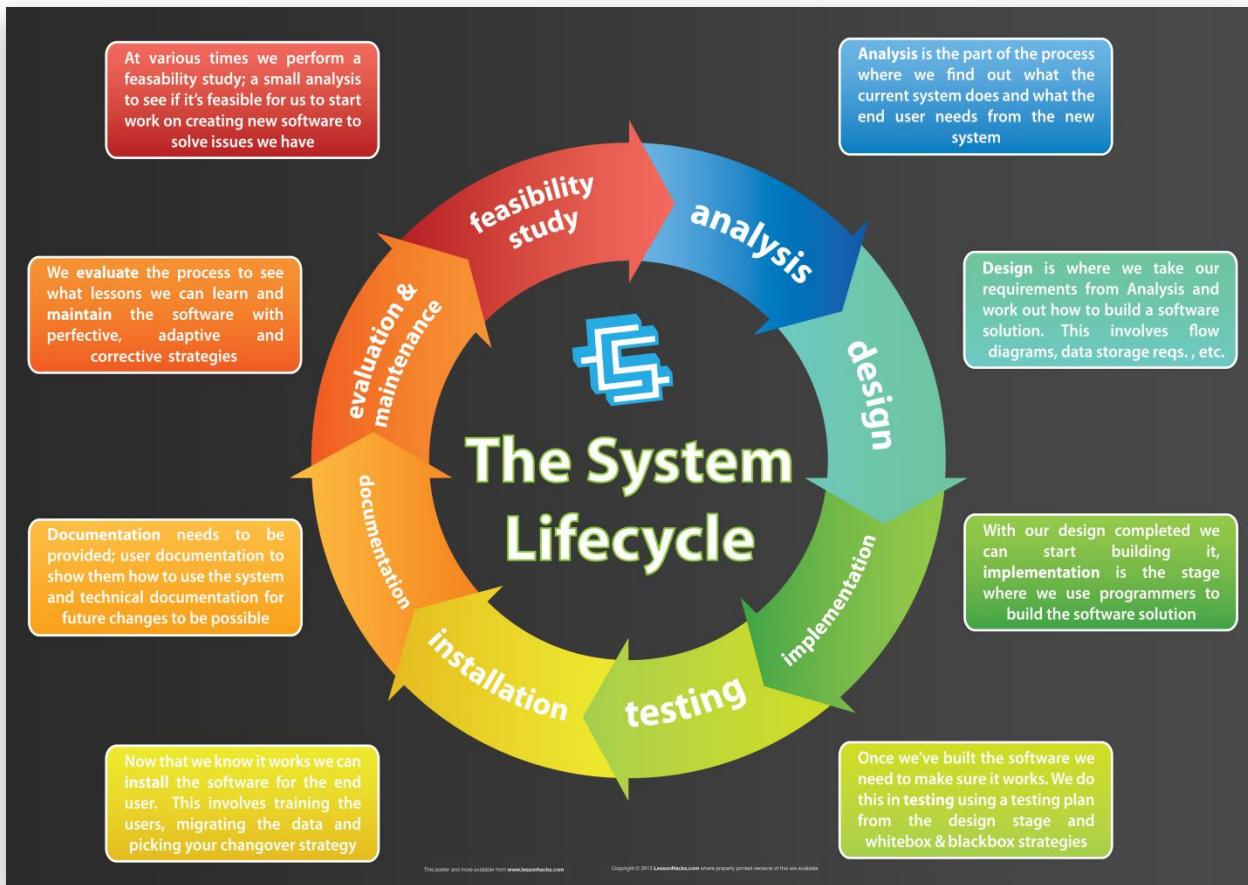
- **System Development Life Cycles and Methodologies**

- The process used to develop information systems is called a methodology.

2.5 :A system development life cycle (SDLC)is a logical process by which systems analysts, software engineers, programmers, and end-users build information systems and computer applications to solve business problems and needs.

❖ **The SDLC usually incorporates the following general-purpose problem solving steps:**

- **Planning:**Identify the scope and boundary of the problem, and plan the development strategy and goals.
- **Analysis:**Study and analyze the problems, causes, and effects. Then, identify and analyze the requirements that must be fulfilled by any successful solution.
- **Design:**-If necessary, design the solution not all solutions require design.
- **Implementation:**-Implement the solution.
- **Support:**- analyze the implemented solution, refine the design, and implement improvements to the solution. Different support situations can thread back into the previous steps.



2.6 :The Data Flow Diagram (DFD) :

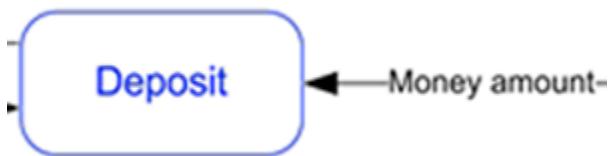
The Data Flow Diagram (DFD) provides a graphical representation of the flow of data through a system. It shows logically what information is exchanged by our system processes and external interfaces or data stores, but it does not explicitly show when or in what sequence the information is exchanged.

Data Flow Diagrams are one of the three essential perspectives of the Structured Systems Analysis and Design Method (SSADM) that predates the more recent object oriented design methods and notations such as UML. This does not mean that the DFD has lost its usefulness even for new analysis endeavors, and any business analyst

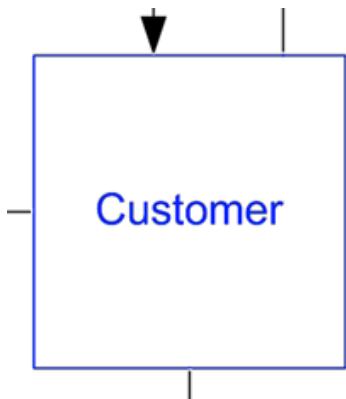
is bound to encounter them while reviewing the original design documentation for 'legacy' systems.

Diagram Elements

The diagram elements listed below and in the subsequent worked example are based on the Gane-Sarson symbol set (or notation) for Data Flow Diagrams. There are other symbol sets such as Yourdon-Coad, which comprise the same four element types albeit represented using different shapes.



This lozenge shape represents a system Process which typically consumes data from an Interface or Data Store (see below), transforms it in some way, and then feeds out the end result. to an Interface or Data Store.



This rectangle shape represents an external Interface, which is any external system or human actor that interacts with our system processes. In some alternative notations the Interface shape may be known as a Terminator, an Input/Output or an Entity.



A Data Flow line shows data flowing from a Process to an external Interface or Data Store, or data flowing from an external Interface or Data Store to a Process. The data flows in the direction of the arrow.



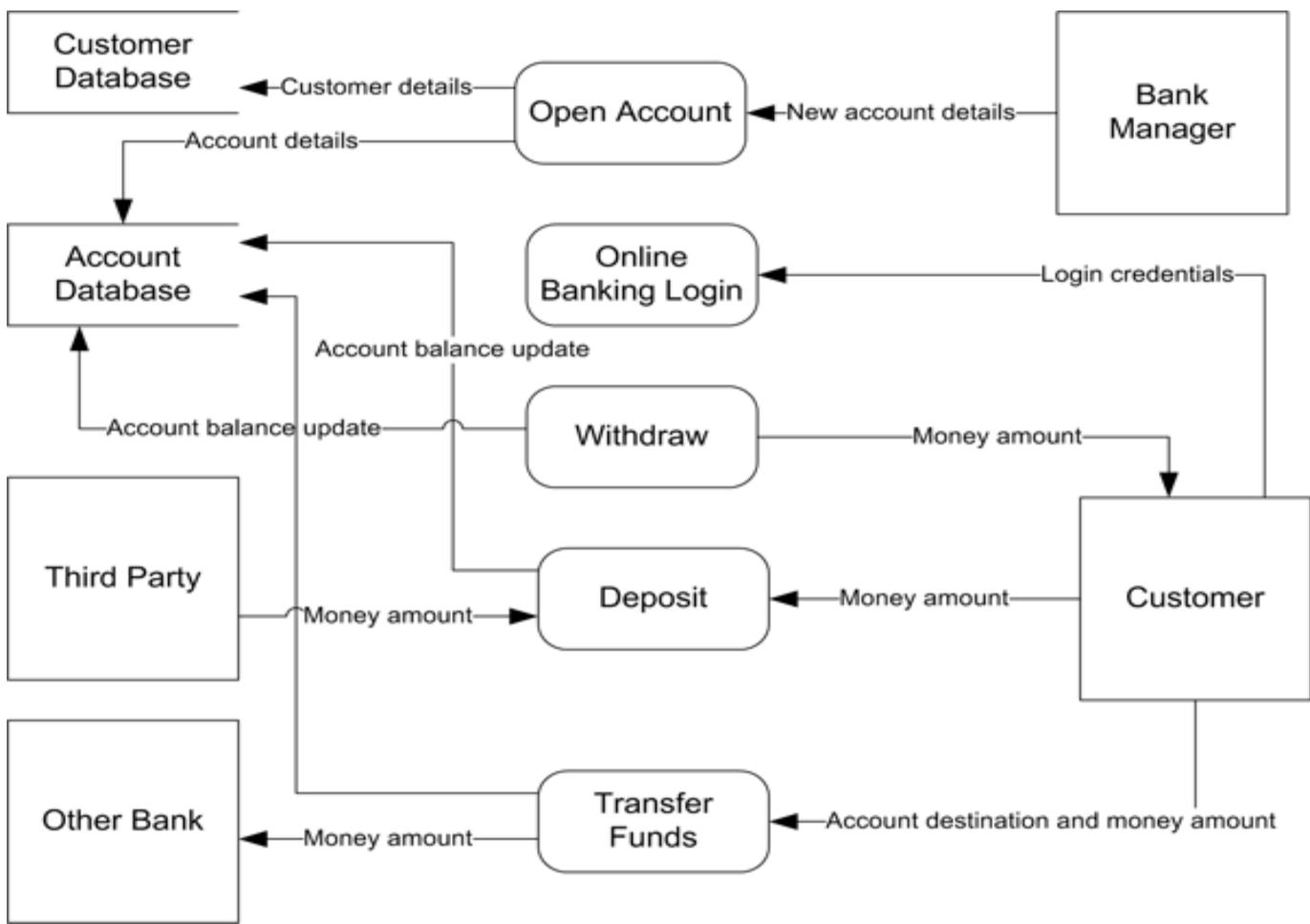
A Data Store may represent an entire database or a more specific entity within a database or other persistent data store.

While this table of diagram elements is informative, the only way to truly appreciate the role of the Data Flow Diagram is via a concrete worked example.

Worked Example

The figure below shows a Data Flow Diagram that was drawn in Microsoft Visio using the Gane-Sarson symbol set.

A good Data Flow Diagram should be easy to comprehend and intuitively obvious to the lay person; ideal for reviewing with non-technical project stakeholders. So take time to interpret the diagram yourself, and then read the description that follows.



This article describes the Data Flow Diagram devised by Larry Constantine in the 1970s as part of the Structured Analysis movement. It follows logically from the Context Diagram article in which we used a much simplified Data Flow Diagram to show a proposed system in the context of its external interfaces and actors.

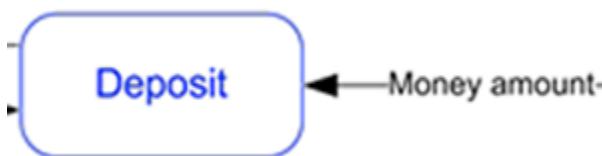
Introduction

The Data Flow Diagram (DFD) provides a graphical representation of the flow of data through a system. It shows logically what information is exchanged by our system processes and external interfaces or data stores, but it does not explicitly show when or in what sequence the information is exchanged.

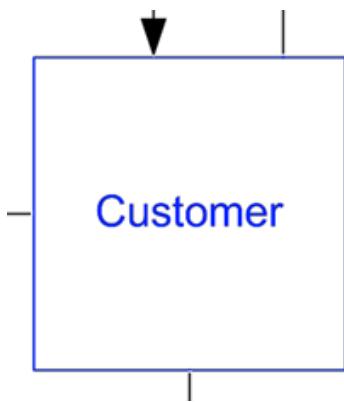
Data Flow Diagrams are one of the three essential perspectives of the Structured Systems Analysis and Design Method (SSADM) that predates the more recent object oriented design methods and notations such as UML. This does not mean that the DFD has lost its usefulness even for new analysis endeavors, and any business analyst is bound to encounter them while reviewing the original design documentation for 'legacy' systems.

Diagram Elements

The diagram elements listed below and in the subsequent worked example are based on the Gane-Sarson symbol set (or notation) for Data Flow Diagrams. There are other symbol sets such as Yourdon-Coad, which comprise the same four element types albeit represented using different shapes.



This lozenge shape represents a system **Process** which typically consumes data from an **Interface** or **Data Store** (see below), transforms it in some way, and then feeds out the end result to an **Interface** or **Data Store**.



This rectangle shape represents an external **Interface**, which is any external system or human actor that interacts with our system processes. In some alternative notations the **Interface** shape may be known as a **Terminator**, an **Input/Output** or an **Entity**.



A **Data Flow** line shows data flowing from a **Process** to an external **Interface** or **Data Store**, or data flowing from an external **Interface** or **Data Store** to a **Process**. The data flows in the direction of the arrow.

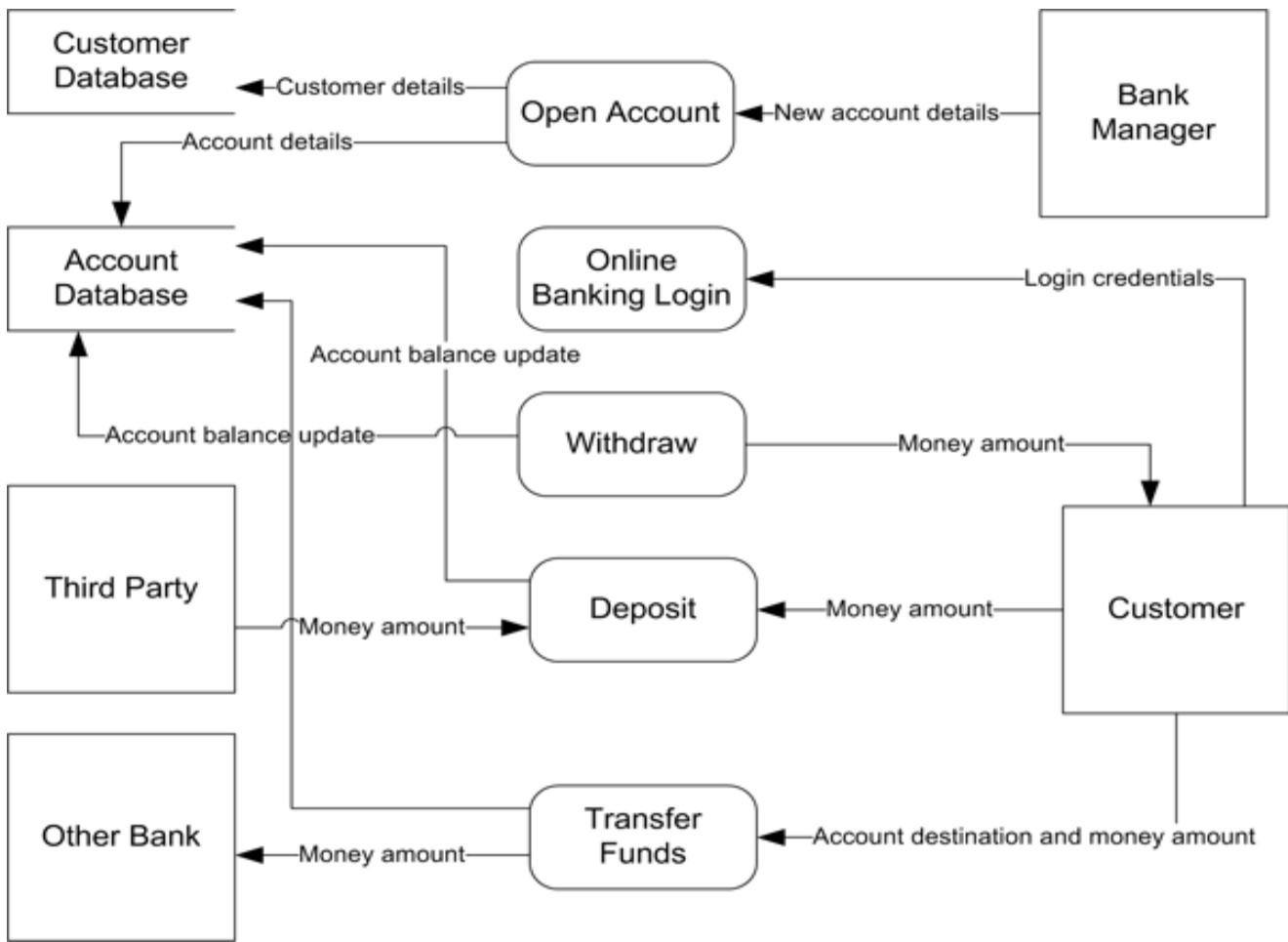


A **Data Store** may represent an entire database or a more specific entity within a database or other persistent data store.

While this table of diagram elements is informative, the only way to truly appreciate the role of the Data Flow Diagram is via a concrete worked example.

Worked Example

The figure below shows a Data Flow Diagram that was drawn in Microsoft Visio using the Gane-Sarson symbol set. A good Data Flow Diagram should be easy to comprehend and intuitively obvious to the lay person; ideal for reviewing with non-technical project stakeholders. So take time to interpret the diagram yourself, and then read the description that follows.



This worked example DFD comprises five processes, four external interfaces / actors, and two data stores. It is not meant to be an exhaustive representation of the data flows in a banking system, but sufficiently comprehensive to give a good feel for how a DFD might be constructed.

A **Bank Manager** actor provides **New account details** to the **Open Account** process which results in **Customer details** being persisted in the **Customer Database** data store and **Account details** being persisted in the **Account Database** data store. Although we have used the phrase 'results in' as part of this explanation, the DFD implies no such cause and effect; all it shows is that the **Open Account** process can read in data from the **Bank Manager** interface and write out data to the **Customer Database** and **Account Database** data stores in no particular order.

A **Customer** actor using the **Online Banking** Login process must provide some data in the form of a set of **Login credentials** such as a user name and password.

A **Customer** actor can receive a **Money amount** from the **Withdraw** process and can supply a **Money amount** to the **Deposit** process; in either case causing (although this causation cannot be explicitly modeled) an **Account balance update** to the **Account Database** data store.

A **Customer** actor can initiate the **Transfer Funds** process, to which he or she must provide an **Account destination and money amount**. The **Transfer Funds** process can send a **Money amount** to another bank via the **Other Bank** interface.

Just like the **Customer** actor, a **Third Party** actor can make use of the **Deposit** process (but obviously not the **Withdraw** process) by supplying a **Money amount**.

Tips and Tricks

Although our focus is on computer systems and software implementations, the DFD has wider uses in modeling non-computerised company processes and exchanges of information. The abstract symbol set could be used to model manual processes and physical data stores such as a filing cabinet. But we're computer analysts, right?

Data flows between external interfaces and data stores should not be shown, for the simple reason that these are considered to be external and 'out of scope'. The analyst should have no knowledge of the interconnections between external entities.

Notice how in the worked example, when modeling the data flow from the **Customer** to the **Login** process we chose to label the data flow with the phrase **Login credentials** rather than (for example) **username and password**. This gives us some flexibility in defining elsewhere what the required login credentials are without invalidating the diagram. In the future we may require the customer to supply an email address and PIN in

order to log in. Note, however, that this is a personal preference and some analysts may prefer to be absolutely explicit when labeling data flows.

The external interfaces and actors in this DFD correspond with those shown on the Context Diagram in the previous article, so all we have really done here is to decompose the all-encompassing Bank System process from the Context Diagram into a set of internal processes for specific tasks. We have defined these processes with a view to making each one a discrete use case on a UML Use Case Diagram, with each data flow between an **Interface** and a **Process** in this diagram suggesting an association between an **Actor** and a **Use Case**. This is not obligatory and is merely a suggestion for aiding traceability between the various systems analysis diagrams and artifacts.

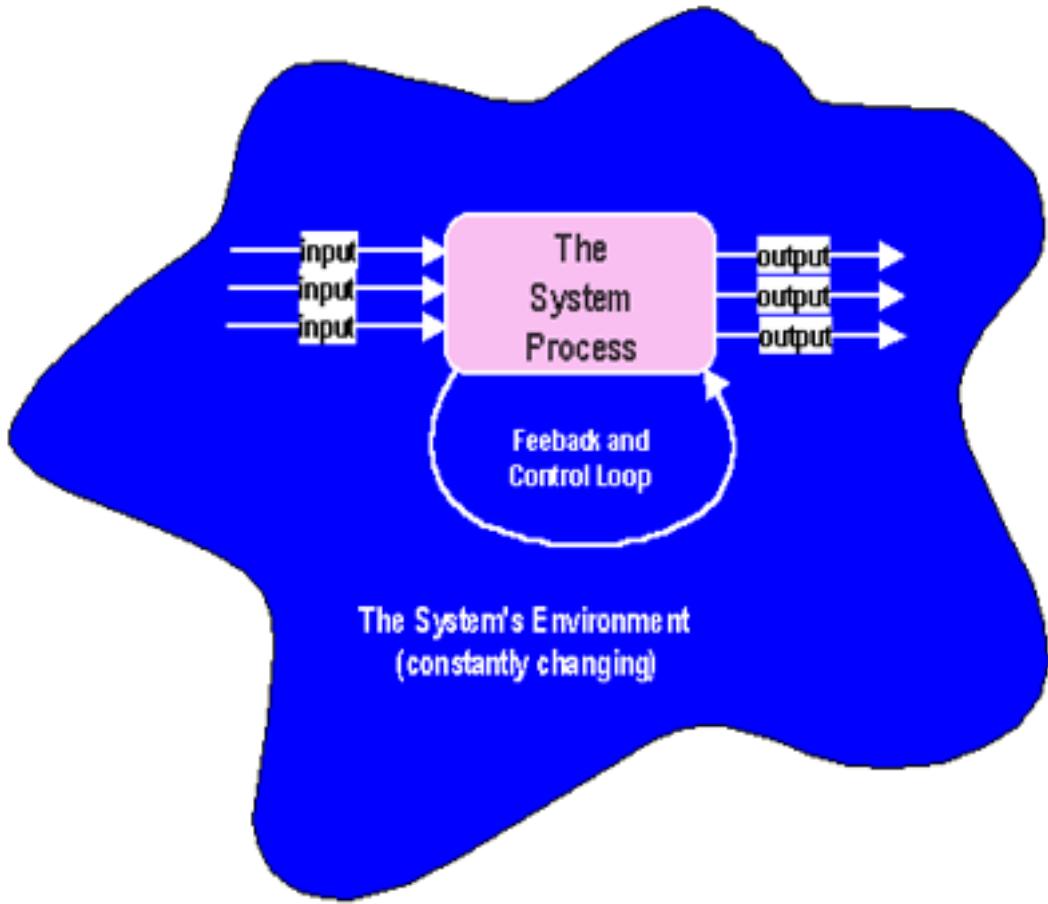
The DFD might also drive the creation of another UML diagram: the UML Activity Diagram which would show the order in which the **Processes** -- to be re-branded as **Activities** -- would be performed. This would resolve the problem of the DFD showing what data is exchanged but not when.

Data Flow Diagram (DFD)

- a systems analysis process model consists of data flow diagrams (dfds).

▣ A DATA FLOW DIAGRAM (DFD)

- is a tool that depicts the flow of data through a system and the work or processing performed by that system. synonyms include bubble chart, transformation graph, and process model.



the classical process model of the system.

2.7 :Functional Decomposition :

What is Functional Decomposition?

Functional decomposition corresponds to the various functional relationships as how the original complex business function was developed. It mainly focusses on how the overall functionality is developed and its interaction between various components.

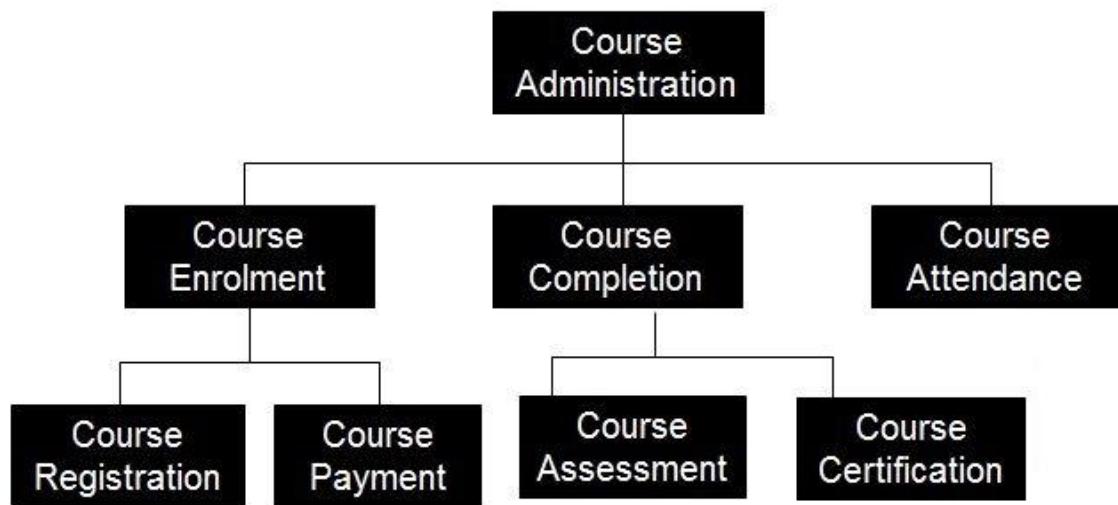
Large or complex functionalities are more easily understood when broken down into pieces using functional decomposition.

When and How?

- Functional decomposition is mostly used during the project analysis phase in order to produce functional decomposition diagrams as part of the functional requirements document.
- Functional Decomposition is done after meeting with business analysts and subject matter expertise.
- Decompose the first level components with their functions and continue to decompose to lower levels until sufficient level of detail is achieved
- Perform an end-to-end walk-through of the business operation and check each function to confirm that it is correct.

Example:

The below example, best describes the Functional Decomposition:

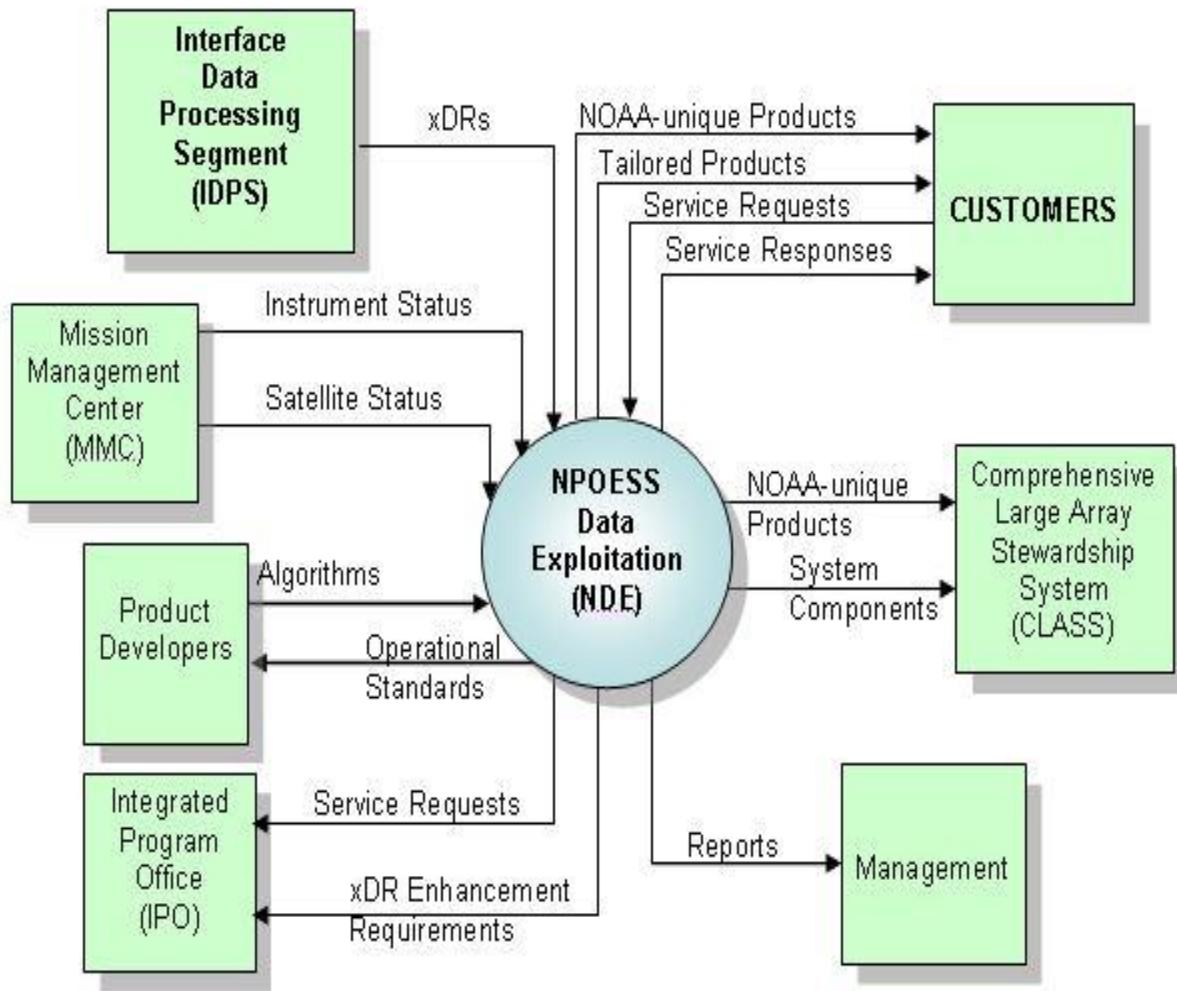


1.8 :Context Diagrams :

Context diagrams depict the environment in which a software system exists. The context diagram shows the name of the system or product of

interest in a circle with the circumference of the circle representing the system boundary. Rectangles outside the circle represent external entities which could be user classes, actors, organizations, other software systems or hardware devices that interface to the system.

The interfaces between the system and the external entities are shown with labeled arrows. The context diagram depicts the project scope at a high level of abstraction but reveals nothing about the system functionality, architecture, or look-and-feel. Nor does it explicitly identify the features or functionality that are in or out of scope. The functional behavior of the system is merely implied by the labeled flows that connect the system to the external entities. Corresponding data inputs and outputs imply the types of operations the system will perform, but these aren't shown explicitly in the context diagram. Context diagrams serve as a useful tool to help the project stakeholders communicate about what lies outside the system boundary.



2.9 :Entity-Relationship Diagram (ERD) :

This solution extends Concept Draw PRO v9.4 software with the ability to describe a database using the Entity-Relationship model.

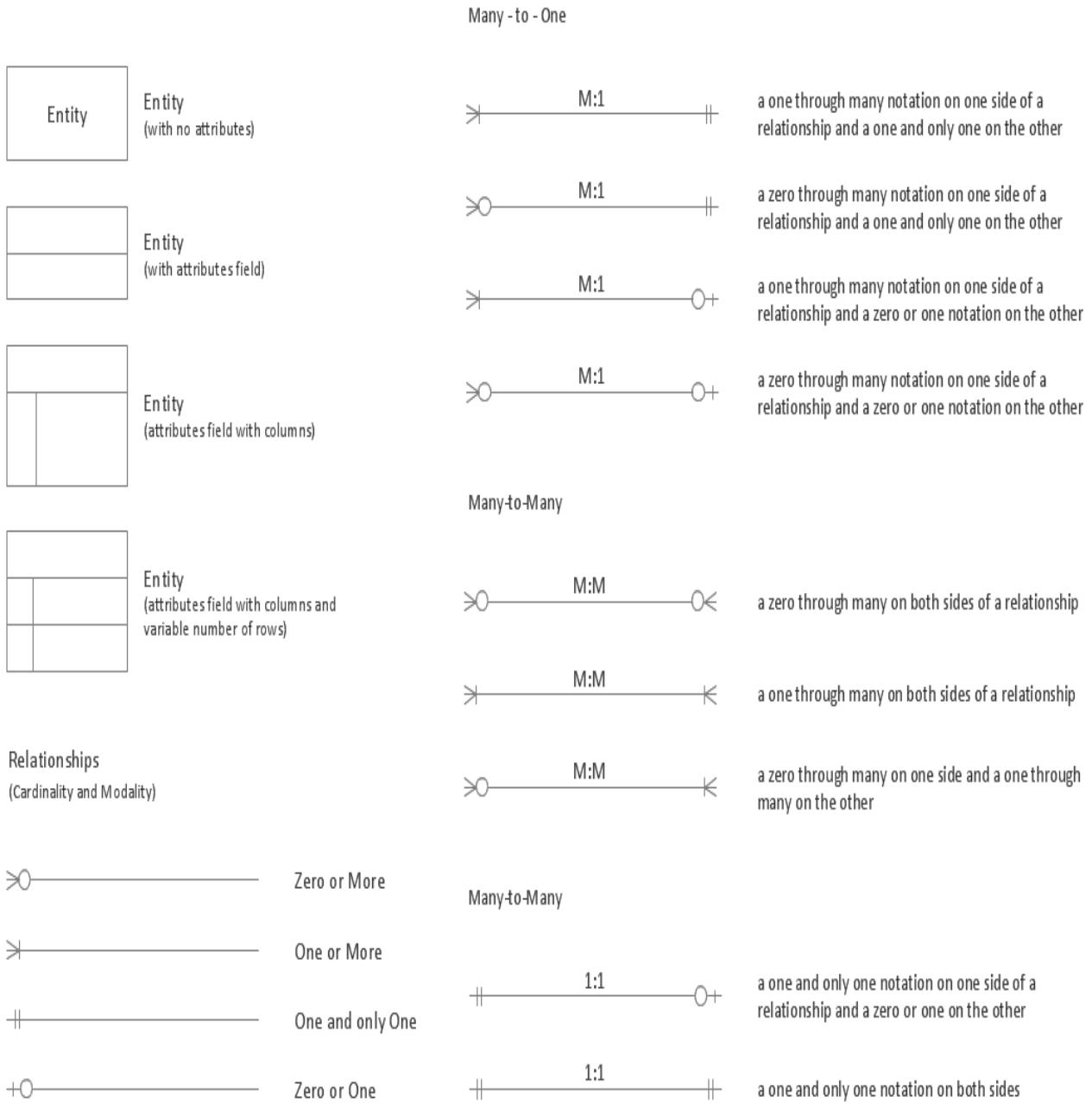
This solution includes icons advocated by Chen's and Crow's Foot notation that can be used when describing a database.

The vector graphic diagrams produced when using this solution can be used in your whitepapers, presentations, datasheets, posters or any technical material.

Design Elements

2 libraries, 45 objects included in Entity-Relationship Diagram (ERD).

Crow's Foot ERD



Chen ERD

Entity	Entity	Attribute	Attribute	Participations Cardinality can be shown or hidden	Recursive Relationship Cardinality can be shown or hidden																														
	Entity		Attribute	Mandatory																															
	Weak Entity		Key attribute	<table> <tr><td>1</td><td>1</td><td>(1:1)</td></tr> <tr><td>N</td><td>(0:N)</td><td></td></tr> <tr><td>1</td><td>N</td><td>(1:N)</td></tr> <tr><td>M</td><td>(0:M)</td><td></td></tr> <tr><td>1</td><td>M</td><td>(1:M)</td></tr> </table>	1	1	(1:1)	N	(0:N)		1	N	(1:N)	M	(0:M)		1	M	(1:M)	<table> <tr><td>1</td><td>1</td><td>(1:1)</td></tr> <tr><td>N</td><td>(0:N)</td><td></td></tr> <tr><td>1</td><td>N</td><td>(1:N)</td></tr> <tr><td>M</td><td>(0:M)</td><td></td></tr> <tr><td>1</td><td>M</td><td>(1:M)</td></tr> </table>	1	1	(1:1)	N	(0:N)		1	N	(1:N)	M	(0:M)		1	M	(1:M)
1	1	(1:1)																																	
N	(0:N)																																		
1	N	(1:N)																																	
M	(0:M)																																		
1	M	(1:M)																																	
1	1	(1:1)																																	
N	(0:N)																																		
1	N	(1:N)																																	
M	(0:M)																																		
1	M	(1:M)																																	
	Relationship		Weak key attribute																																
	Identifying Relationship		Derived attribute	Optional																															
	Associative Entity		Multivalue attribute	<table> <tr><td>1</td><td>1</td><td>(1:1)</td></tr> <tr><td>1</td><td>N</td><td>(0:N)</td></tr> <tr><td>1</td><td>N</td><td>(1:N)</td></tr> <tr><td>1</td><td>M</td><td>(0:M)</td></tr> <tr><td>1</td><td>M</td><td>(1:M)</td></tr> </table>	1	1	(1:1)	1	N	(0:N)	1	N	(1:N)	1	M	(0:M)	1	M	(1:M)																
1	1	(1:1)																																	
1	N	(0:N)																																	
1	N	(1:N)																																	
1	M	(0:M)																																	
1	M	(1:M)																																	

Examples :

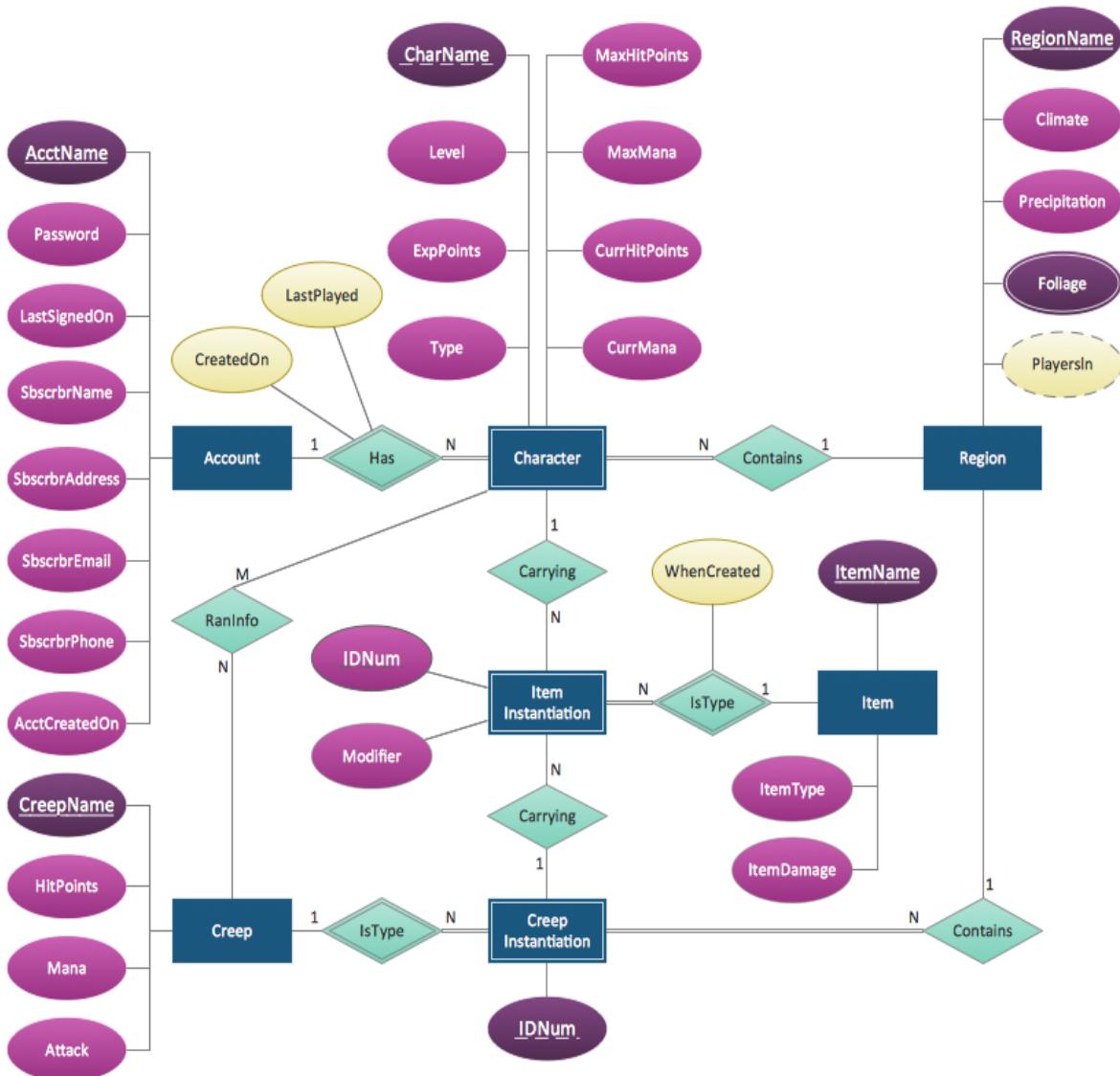
The samples you see on this page were created in Concept Draw PRO using the Entity-Relationship Diagram (ERD) solution, and demonstrate some of the solution's capabilities and the results you can achieve.

All source documents are vector graphic documents. They are available for reviewing, modifying, and converting to a variety of available formats (PDF file, PowerPoint Presentation, image, or MS Visio XML) from the Template Gallery of Concept Draw PRO, for all users who have installed the Entity-Relationship Diagram (ERD) solution.

Example 1: Entity-Relationship Diagram Using Chen's Notation

This diagram was created in Concept Draw PRO using the Chen ERD library from the Entity-Relationship Diagram (ERD) solution. An experienced user spent 20 minutes creating this sample.

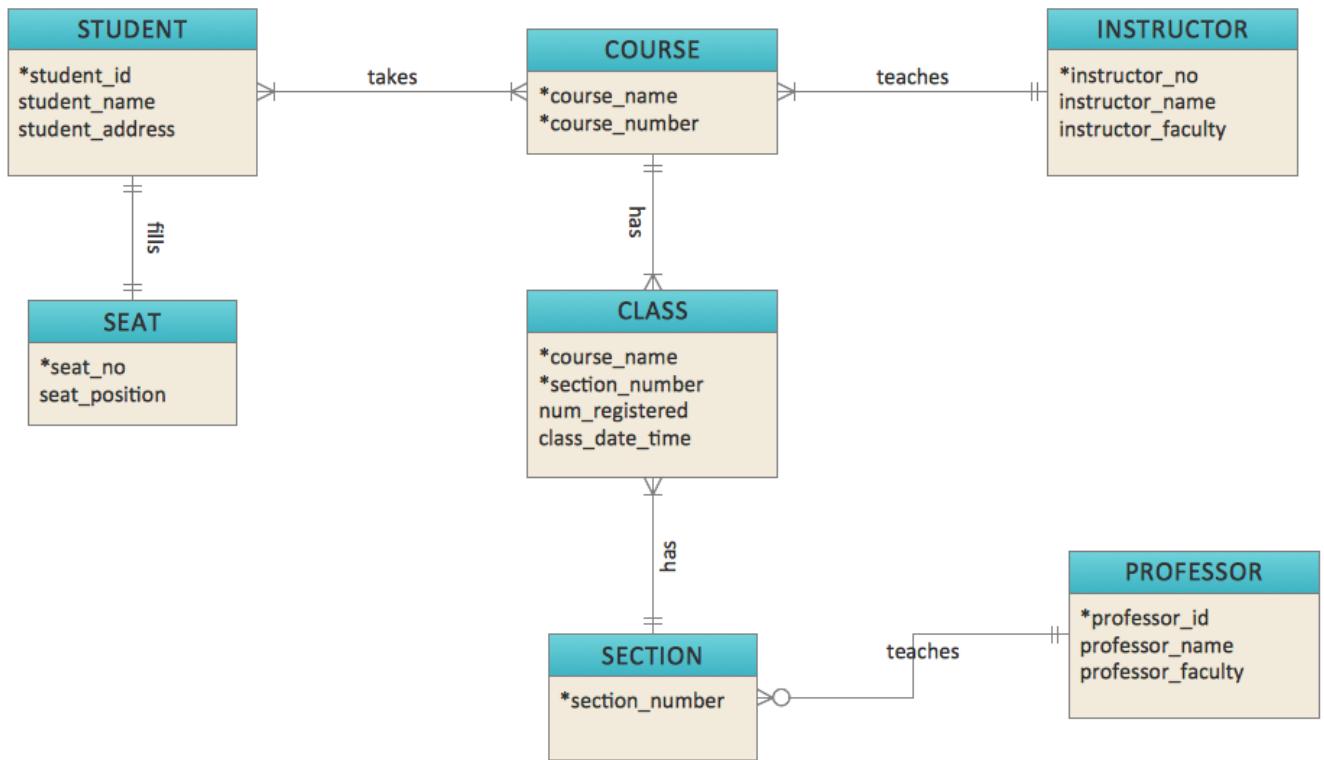
This sample diagram using the Entity-Relationship Diagram (ERD) Solution shows the type of icons and graphics you can use to develop a model of a database of arbitrary complexity according to Chen's notation.



Example 2: Entity-Relationship Diagram Using Crow's Foot Notation

This diagram was created in Concept Draw PRO using the Crows Foot ERD library from the Entity-Relationship Diagram (ERD) solution. An experienced user spent 20 minutes creating this sample.

This sample diagram using the Entity-Relationship Diagram (ERD) Solution and shows the type of icons and graphics you can use to develop a model of a database of arbitrary complexity using Crow's Foot notation.



2.10 :SYSTEM DESIGN AND CONSTRUCTION

System design includes the evaluation of alternative solutions, preparation of detailed computer-based specifications that will fulfill the requirements specified during system analysis, and construction of system prototypes.

- **System design** is the evaluation of alternative solutions and the specification of a detailed computer-based solution. It is also called **physical design**.

STRATEGIES FOR SYSTEM DESIGN

There are many popular strategies or techniques for performing system design. These techniques can be used in combination with one another.

- ◆ **Modern Structured Design**, a technique that focuses on processes.
- ◆ **Information Engineering (IE)**, a technique that focuses on data and strategic planning to produce application projects.
- ◆ **Prototyping** a technique that is an iterative process involving a close working relationship between designers and users to produce a model of the new system.
- ◆ **Joint Application Development (JAD)**, techniques that emphasizes participative development among system owners, users, designers, and builders. During JAD sessions for system design, the system designer takes on the role of the facilitator.
- ◆ **Rapid Application Development (RAD)**, a technique that represents a merger of various structured techniques with prototyping and JAD to accelerate systems development.
- ◆ **Object-Oriented Design (OOD)**, a new system strategy that follows up object-oriented analysis to refine object requirement definitions and to define new design specific objects.

Also there are several strategies and techniques that discussed in the previous chapter such as Structured Requirement Definition (SRD), Jackson System Development (JSD), Higher-Order Software (HOS) used for system design.

APPLICATION ARCHITECTURE AND PROCESS DESIGN

Information application architecture and process design include techniques for distributing data, processes, and interfaces to network locations in a distributed computing environment.

An **application architecture** defines the technologies to be used by (and to build) one, more, or all information systems in terms of its data, process, interfaces, and network components.

- The prevailing computing model is currently **client / server** wherein networks of clients, single-user computers, are connected to and interoperates with servers, multiple-user computers that share their services. This is also called **distributed computing**.
- Centralized computing, distributed presentation, distributed data, distributed data and logic, and Internet/intranet computing are flavors of **client/server computing**.
- Client/server computing can be based on different network topologies including bus, ring, star, and hierarchical networks.
- Data storage is typically implemented using distributed relational database technology that either partitions data to different servers or replicates data on multiple servers.
- Processes are implemented using highly integrated tool kits called software development environments.
- Physical data flows diagrams model an information system's application architectures and processes. Because they show the planned implementation of all processes, data stores, and data flows, they serve as a general design or blueprint for subsequent detailed design, prototyping, and construction.
- System flowcharts are a lesser used diagram to show the same implementation features as a physical data flow diagrams. Although they are rarely drawn today, many older, legacy systems use them for documentation.

DATABASE DESIGN

Data storage is a critical component of most information systems. Some people consider it to be the critical component. The data captured by an information system is stored in files and databases. A file is a collection of records. A database is a collection of interrelated files. Many legacy systems were built with files technology because files were built for specific applications their design was optimized for those applications. This close relationship between the files and their applications made it difficult to restructure the files to meet future requirements. And because many applications use the same data, it is not uncommon to find redundant files with data values that do not always match.

Database design is the process of translating logical data models into physical database schemas. The smallest unit of meaningful data that can be stored is called a field. There are four types of fields:

- a) A *primary key* is a field that uniquely identifies one and only one record in a file or table.
- b) A *secondary key* is a field that may either uniquely identify one and only one record in a file or table or identify a set of records with some common, meaningful characteristic.
- c) A *foreign key* is a field that points to a related record in a different table.

Fields are organized into records, and similar *records* are organized into files or tables. A *database* is a collection of tables (files) with logical pointers that relate records in one table to records in a different table.

Database architecture is built around a database management system (DBMS) that provides the technology to define the database structure and then to create, read, update, and delete records in the tables that make up that structure. A DBMS provides a data language to accomplish this. That language provides at least two components:

- a) A *data definition language* to create and maintain the database structure and rules.
- b) A *data manipulation language* to create, read, use, update, and delete records in the database.

Today, relational database management systems are used to support the development and reengineering of the overwhelming number of information systems. Relational databases store data in a collection of tables that are related via foreign keys.

- a) The data definition and manipulation languages of most relational DBMSs are consolidated into a standard language known as SQL.
- b) High-end relational database management systems support triggers and stored procedures, programs that are stored with the tables and callable from other SQL-based programs.

Data analysis and normalization are techniques for removing impurities from a data model as a preface to designing the database. These impurities can make a database unreliable, inflexible, and non-salable.

Normalization involves checking each entity (table) for first, second, and third normal form impurities.

- a) An entity is in first normal form if it contains no repeating attributes.
- b) An entity is in second normal form if it contains no partial dependencies.
- c) An entity is in third normal form if it contains no derived attributes or no transitive dependencies.

Database integrity should be checked and, if necessary, improved to ensure that the business and its users can trust the stored data.

- a) Key integrity ensures that every record will have a unique, non-null primary key value.
- b) Domain integrity ensures that appropriate fields will store only legitimate values from the set of all possible values.
- c) Referential integrity ensures that no foreign key value points to nonexistent primary key value.

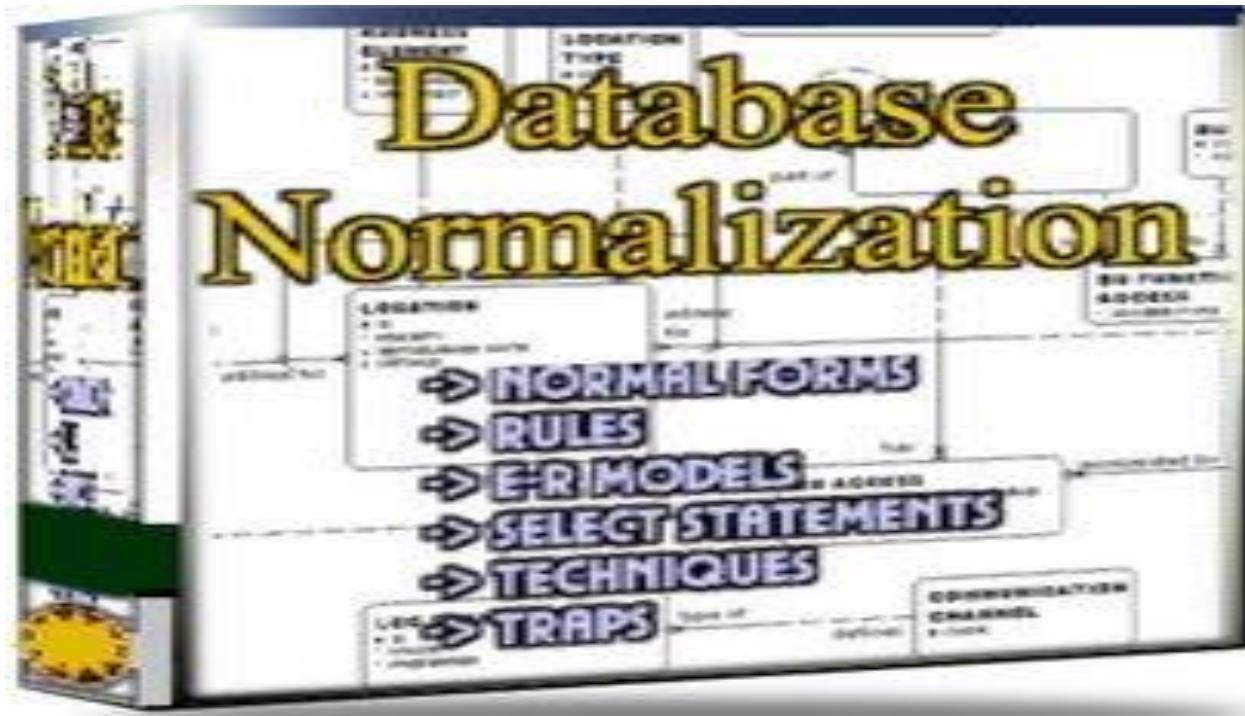
A deletion rule should be specified for every relationship with another table.

Summary

- An **information system** is an arrangement of people, data, processes, interfaces, networks, and technology that interact for the purpose of supporting and improving both day-to-day operations in a business, as well as supporting the problem solving and decision making needs of management.
- .
- Every system can be considered a black box, which has input, process, files and output.
- A system development life cycle (SDLC) is a logical process by which systems analysts, software engineers, programmers, and end-users build information systems and computer applications to solve business problems and needs.

End of Chapter 2

Thanks



→ Chapter 3 :

Normalization

3.1 : Database normalization is the process of organizing the attributes and tables of a relational database to minimize data redundancy.

Normalization involves decomposing a table into less redundant (and smaller) tables but without losing information; defining foreign keys in the old table referencing the primary keys of the new ones. The objective is to isolate data so that additions, deletions, and modifications of an attribute can be made in just one table and then propagated through the rest of the database using the defined foreign keys. Edgar F. Codd, the inventor of the relational model (RM), introduced the concept of normalization and what we now know as the First normal form (1NF) in 1970. Codd went on to define the Second normal form (2NF) and Third normal form (3NF) in 1971, and Codd and Raymond F. Boyce defined the Boyce-Codd Normal Form (BCNF) in 1974. Informally, a relational database table is often described as "normalized" if it is in the Third Normal Form.^[4] Most 3NF tables are free of insertion, update, and deletion anomalies. The relational model separates the logical design from the physical design: DBMS performance is a matter of physical designer using indexes, view materialization, big buffers, etc. It is not a matter of changing the logical design.

A typical example of normalization is that an entity's unique ID is stored everywhere in the system but its name is held in only one table. The name can be updated more easily in one row of one table. A typical update in such an example would be the RIM company changing its name to BlackBerry. That update

would be done in one place and immediately the correct "BlackBerry" name would be displayed throughout the system.

3.2 : normalization is In creating a database, normalization is the process of organizing it into tables in such a way that the results of using the database are always unambiguous and as intended. Normalization may have the effect of duplicating data within the database and often results in the creation of additional tables. (While normalization tends to increase the duplication of data, it does not introduce redundancy, which is unnecessary duplication.) Normalization is typically a refinement process after the initial exercise of identifying the data objects that should be in the database, identifying their relationships, and defining the tables required and the columns within each table.

- Normalization is a process in which we systematically examine relations for anomalies and, when detected, remove those anomalies by splitting up the relation into two new, related, relations.
- Normalization is an important part of the database development process: Often during normalization, the database designers get their first real look into how the data are going to interact in the database.

- Finding problems with the database structure at this stage is strongly preferred to finding problems further along in the development process because at this point it is fairly easy to cycle back to the conceptual model (Entity Relationship model) and make changes.
- Normalization can also be thought of as a trade-off between data redundancy and performance. Normalizing a relation reduces data redundancy but introduces the need for joins when all of the data is required by an application such as a report query.

Recall, the Relational Model consists of the elements: relations, which are made up of attributes.

- A relation is a set of attributes with values for each attribute such that:
 1. Each attribute (column) value must be a single value only.
 2. All values for a given attribute (column) must be of the same data type.
 3. Each attribute (column) name must be unique.
 4. The order of attributes (columns) is insignificant
 5. No two tuples (rows) in a relation can be identical.
 6. The order of the tuples (rows) is insignificant.

Normalization degrees of relational database tables have been defined and include:

3.3 :First normal form (1NF) :

This is the "basic" level of normalization and generally corresponds to the definition of any database, namely:

- It contains two-dimensional tables with rows and columns.
- Each column corresponds to a sub-object or an attribute of the object represented by the entire table.
- Each row represents a unique instance of that sub-object or attribute and must be different in some way from any other row (that is, no duplicate rows are possible).
- All entries in any column must be of the same kind.

3.4 :Second normal form (2NF) :

At this level of normalization, each column in a table that is not a determiner of the contents of another column must itself be a function of the other columns in the table.

3.5 :Third normal form (3NF) :

At the second normal form, modifications are still possible because a change to one row in a table may affect data that refers to this information from another table.

3.6 :DATABASE TABLES AND NORMALIZATION :

Having good relational database software is not enough to avoid the data redundancy discussed in Chapter 1, Database Systems. If the database tables are treated as though they are files in a file system, the RDBMS never has a chance to demonstrate its superior data-handling capabilities. The table is the basic building block in the database design process. Consequently, the table's structure is of great interest. Ideally, the database design process explored in Chapter 4, Entity Relationship (ER) Modeling, yields good table structures. Yet it is possible to create poor table structures even in a good database design. So how do you recognize a poor table structure, and how do you produce a good table? The answer to both questions involves normalization. Normalization is a process for evaluating and correcting table structures to minimize data redundancies, thereby reducing the likelihood of data anomalies. The normalization process involves assigning attributes to tables based on the concept of determination you learned about in Chapter 3, The Relational Database Model. Normalization works through a series of stages called normal forms. The first three stages are described as first normal form (1NF), second normal form (2NF), and third normal form (3NF). From a structural point of view, 2NF is better than 1NF, and 3NF is better than 2NF. For most purposes in business database design, 3NF is as high as you need to go in the normalization process. However, you will

discover in Section 5.3 that properly designed 3NF structures also meet the requirements of fourth normal form (4NF). Although normalization is a very important database design ingredient, you should not assume that the highest level of normalization is always the most desirable. Generally, the higher the normal form, the more relational join operations required to produce a specified output and the more resources required by the database system to respond to end-user queries. A successful design must also consider end-user demand for fast performance. Therefore, you will occasionally be expected to denormalize some portions of a database design in order to meet performance requirements. Denormalization produces a lower normal form; that is, a 3NF will be converted to a 2NF through denormalization. However, the price you pay for increased performance through denormalization is greater data redundancy.

3.7 : THE NEED FOR NORMALIZATION :

To get a better idea of the normalization process, consider the simplified database activities of a construction company that manages several building projects. Each project has its own project number, name, employees assigned to it, and so on. Each employee has an employee number, name, and job classification, such as engineer or computer technician. The company charges its clients by billing the hours spent on each contract. The hourly billing rate is dependent on the employee's position. For example, one hour of computer

technician time is billed at a different rate than one hour of engineer time.

3.8 : THE NORMALIZATION PROCESS :

In this section, you learn how to use normalization to produce a set of normalized tables to store the data that will be used to generate the required information. The objective of normalization is to ensure that each table conforms to the concept of well-formed relations, that is, tables that have the following characteristics: Each table represents a single subject. For example, a course table will contain only data that directly pertains to courses. Similarly, a student table will contain only student data. No data item will be unnecessarily stored in more than one table (in short, tables have minimum controlled redundancy). The reason for this requirement is to ensure that the data are updated in only one place. All nonprime attributes in a table are dependent on the primary key – the entire primary key and nothing but the primary key. The reason for this requirement is to ensure that the data are uniquely identifiable by a primary key value. Each table is void of insertion, update, or deletion anomalies. This is to ensure the integrity and consistency of the data. To accomplish the objective, the normalization process takes you through the steps that lead to successively higher normal forms. The most common normal forms and their basic characteristic are listed in Table . You will learn the details of these normal forms in the indicated sections.

Normal Forms	
NORMAL FORM	CHARACTERISTIC
First normal form (1NF)	Table format, no repeating groups, and PK identified
Second normal form (2NF)	1NF and no partial dependencies
Third normal form (3NF)	2NF and no transitive dependencies
Boyce-Codd normal form (BCNF)	Every determinant is a candidate key (special case of 3NF)
Fourth normal form (4NF)	3NF and no independent multivalued dependencies

From the data modeler's point of view, the objective of normalization is to ensure that all tables are at least in third normal form (3NF). Even higher-level normal forms exist. However, normal forms such as the fifth normal form (5NF) and domain-key normal form (DKNF) are not likely to be encountered in a business environment and are mainly of theoretical interest. More often than not, such higher normal forms usually increase joins (slowing performance) without adding any value in the elimination of data redundancy. Some very specialized applications, such as statistical research, might require normalization beyond the 4NF, but those applications fall outside the scope of most business operations. Because this book focuses on practical applications of database techniques, the higher-level normal forms are not covered.

3.9 :First norm form (1NF) :

Table format

No repeating group

PK identified

emp_id	emp_name	emp_gender	emp_image	emp_address	emp_phone	emp_email	emp_pass	emp_band_id	band_name	item_id	item_title	dept_id	dept_name	stu_id	stu_stu	stu_stu	stu_stu	inst_id	inst_name	inst_inst	inst_inst	inst_inst	inst_inst	course_id	course_name	exam_result	exam_resid_id	exam_id	exam_type	answ1	answ2	answ3	answ4	answ5	correct_ques_total
1	Ali Ahmed	Male	Tasneem 012222	Empal Egypt 3456	102004	01	College	01	first	1	SS 05	Aliya	Female	10-00	011545	Study	Abbas	03	Rafia	Female	Menna	012456	Dr.	123456	01	Security	1	1	1	choice	1	15			
2	Hossam female	Bal	Tasneem 010000	Emp. Aski2ka Egypt 3345	02	College	02	Second	2	IS 06	mostafa male	Belbes	012456	Stu	0103	04	Heba	Female	Tasneem 012456	DR	010305	02	Data	2	1	3	True&False	1	15						
3	Bla	female	Tasneem 010023	Egypt 23244 01	01055	03	College	03	Third	3	IT 08	Omar male	Tasneem 012113	Dtu	FSDW	05	Ali Ahmed	Male	Tasneem 011122	DR	23225	03	Graphic	5	1	7	choice	1	15						

3.10 :Second norm form (2NF) :

1NF

No partial dependences

Write each key component :

Emp_id → Employee

Band_id → Band

Term_id → Term

Dept_id → Department

Stu_id → Student

Inst_id → Instructor

Course_id → Course

Question_id → Exam

Exam_res_id → Exam-result

Assign corresponding dependent attributes :

Employee (emp_id , emp_name , emp_gender , emp_image ,
emp_address , emp_phone , emp_user name , emp_password)

emp_id	emp_name	emp_gender	emp_image	emp_address	emp_phone	emp_user_name	emp_pass
1	ali ahmed	male		zagazig egypt	012222 3456	emp.ali	120AA HHS
2	hossam baz	female		zagazig egypt	010000 33445	emp. hossam	askj12d kjhh
3	aya salah	female		zagazig egypt	010023 23344	emp. aya	12sfwE DW56

Band (band_id , band_title)

band_id	band_title
01	college
02	college
03	college

Term (term_id , term_title)

term_id	term_title
01	first term
02	second term
03	third term

Department (dept_id , dept_name)

dept_id	dept_name
1	CS
2	IS
3	IT

Student (stu_id , stu_name , stu_gender , stu_image ,
 stu_phone , stu_address , stu_user name , stu_password)

stu_id	stu_name	stu_gender	stu_image	stu_address	stu_phone	stu_user name	stu_pass
05	aya saeed	female		10-mn ramadan	011154 327	stu.aya	AaAas 123
06	mostafa ali	male		belbes zagazig	01234 5678	stu. mostafa	QWE3 RT4
08	omar ali	male		zagazig egypt	012113 33332	stu. omar	FSVXD 43Vgk

Instructor (inst_id , inst_name , inst_gender , inst_image ,
 inst_phone , inst_address , inst_user name , inst_password)

inst_id	inst_name	inst_gender	inst_image	inst_address	inst_phone	inst_user name	inst_pass
03	radwa ali	female		menya elamh	012345 345	DR- radwa	123456 ASDGH
04	heba salah	female		zagazig- egypt	012345 3455	DR- heba	QYHDS 4FF
05	ahmed salah	male		zagazig egypt	011122 34333	DR- ahmed	23122s gfgQE

Course (course_id , course_name)

course_id	course_name
01	secuirty
02	data base
03	grafic

Exam (question_id , exam_id , question_type , question , answer 1 , answer 2 , answer 3 , answer 4 , correct_answer , question_degree , total_degree)

ques_id	exam_id	ques_type	ques_ answ1	ques_ answ2	ques_ answ3	ques_ answ4	correct_ answ	ques_degree	total_degree
1	1	choice					1	15	
1	3	true& false					1	15	
1	7	choice					1	15	

Exame result (exam_res-id)

exam_result	exam_res-id
	1
	2
	5

3.11: Third norm form (3NF) :

2NF

No transitive dependences

3.12 :Boyce-codd norm form (BCNF) :

Every determinate is candidate key

End of Chapter 3

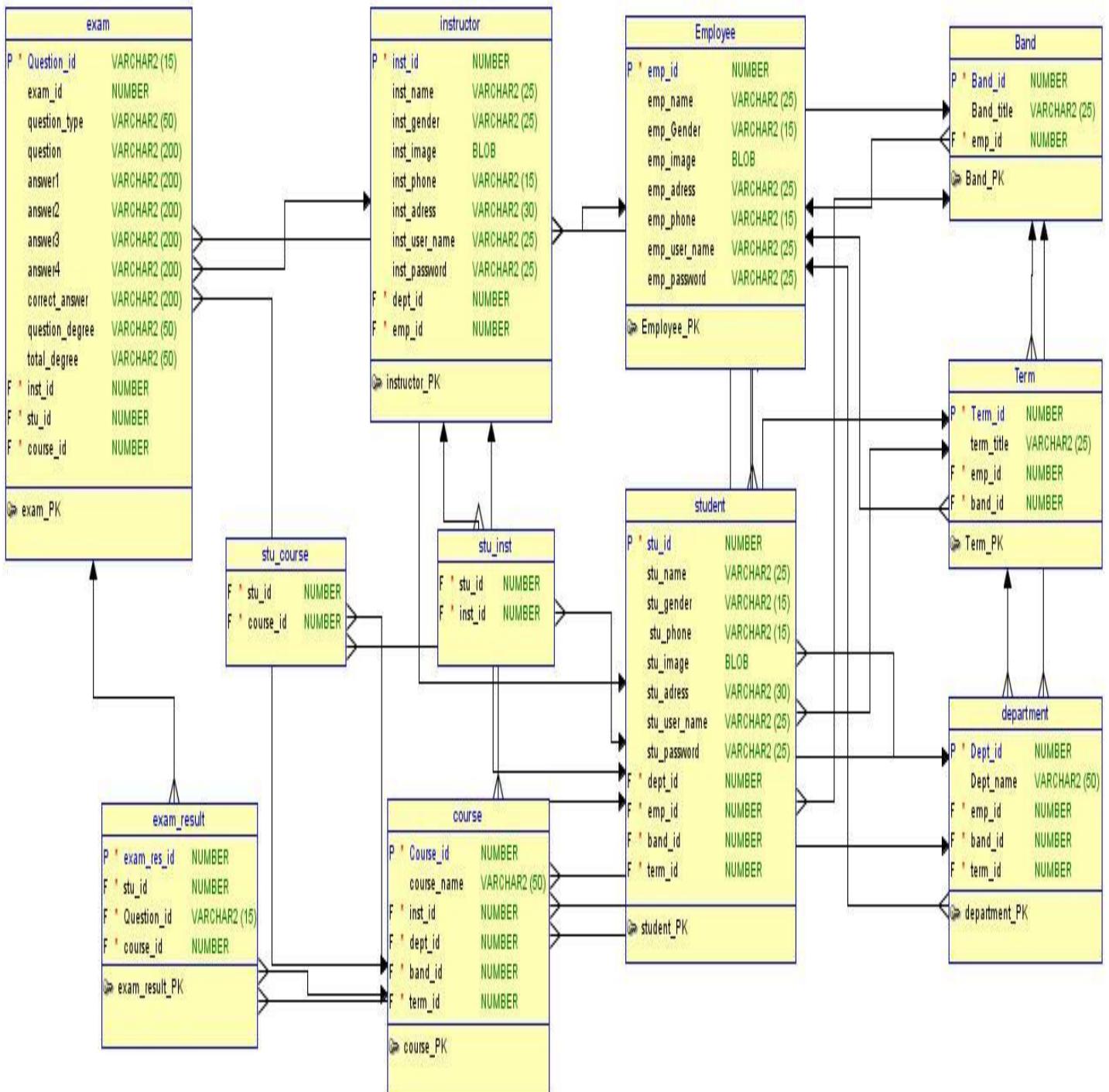
Thanks



→ Chapter 4 :

QPS Analysis

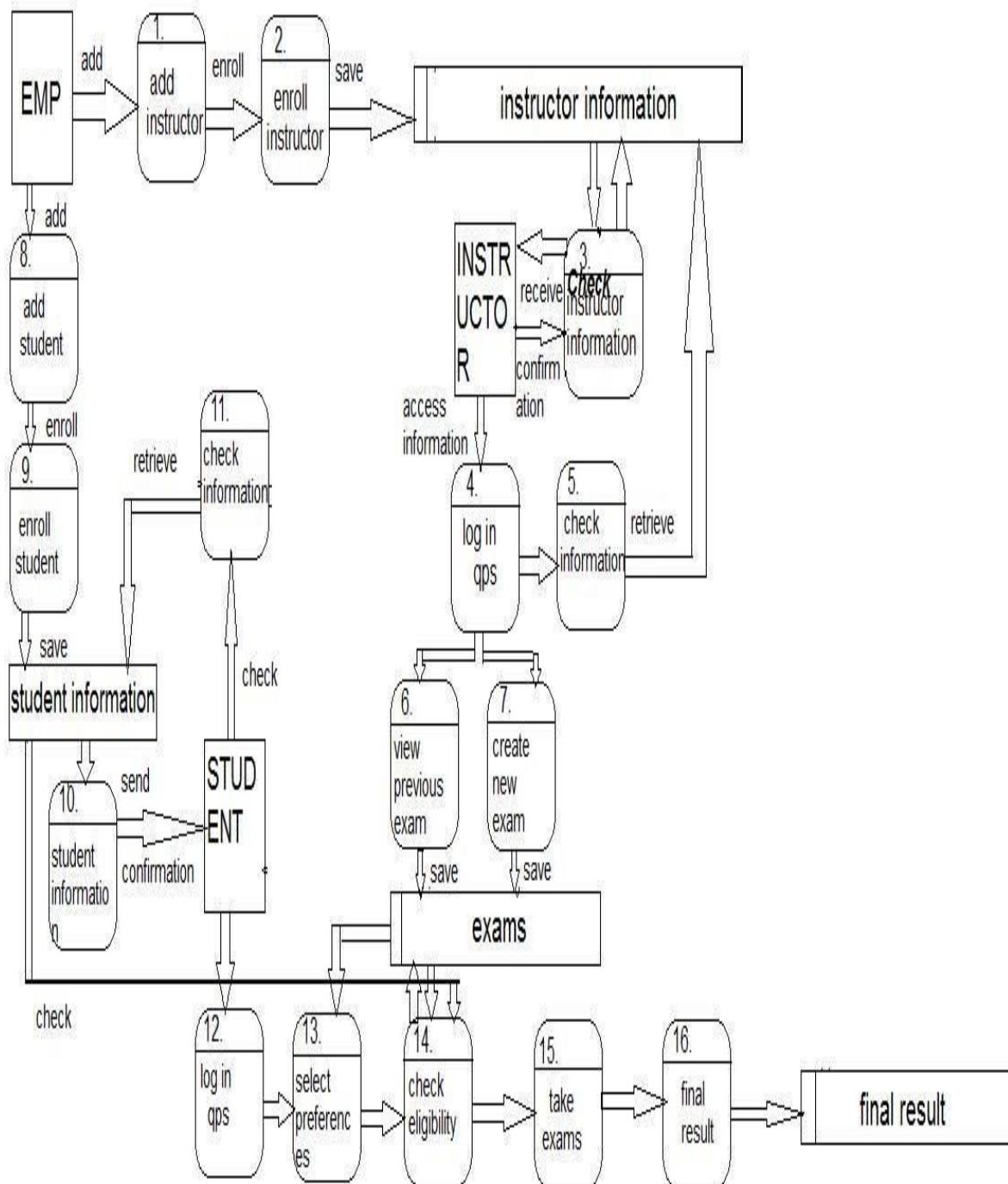
4.1 QPS ERD :



Relationships between entities

Employee 1: M Band
Employee 1: M Term
Band 1: M Term
Employee 1: M Department
Band 1: M Department
Term 1: M Department
Department 1: M Student
Employee 1: M Student
Band1: M Student
Term1: M Student
Department 1: M Instructor
Employee 1: M Instructor
Student M: M Instructor
Department 1: M Course
Band 1: M Course
Term 1: M Course
Instructor 1: M Course
Student M: M Course
Instructor 1: M Exam
Student 1: M Exam
Course 1: M Exam
Student 1: M Exam_result
Course 1: M Exam_result
Exam 1: M Exam_result

4.2 QPS DFD :



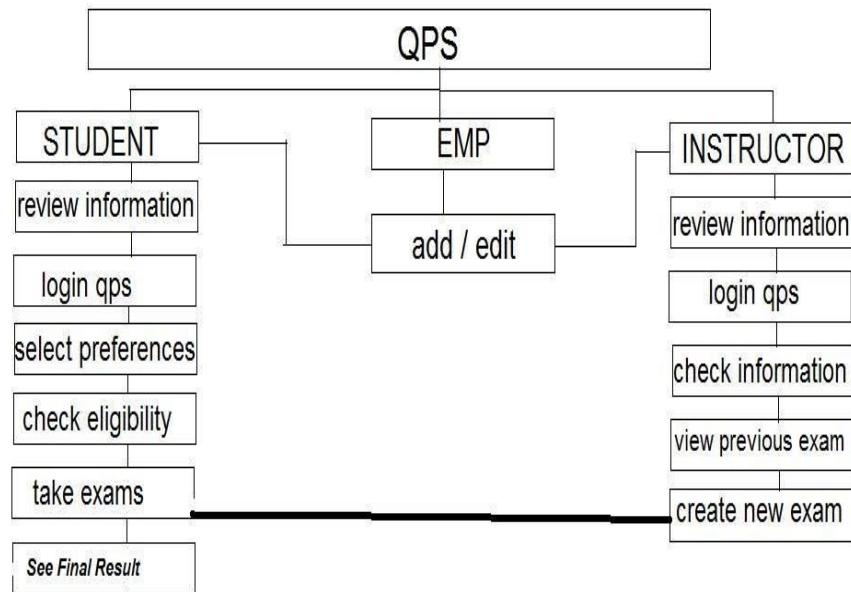
Data flow diagram (DFD) describes operations which happen inside the system

EMP adds students and lectures

Instructor adds exams and can see his information

Student can see his information , takes the exams and tests

4.3 Functional decomposition :



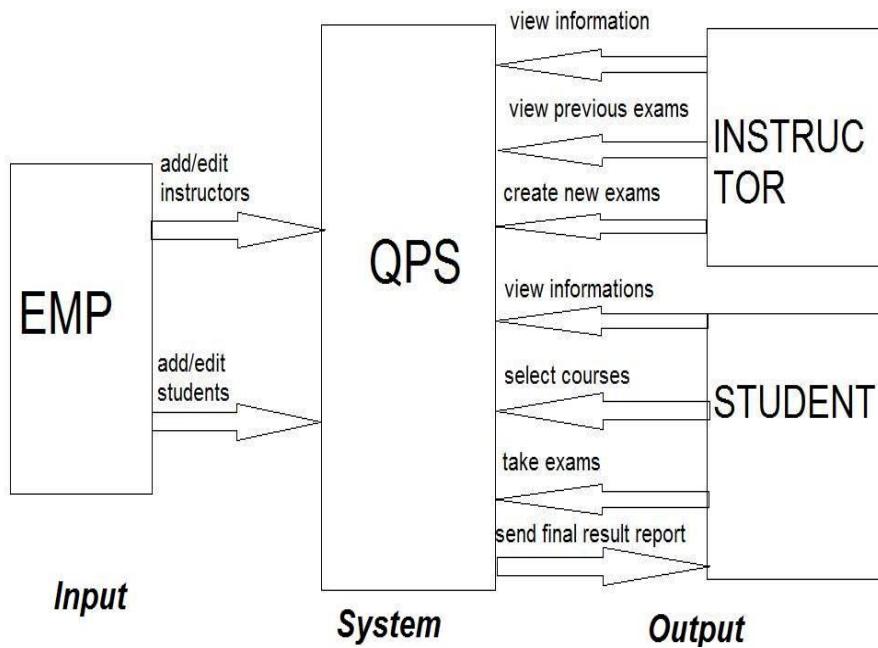
Functional decomposition contains components of the system

EMP adds students and lectures

Instructor adds exams and can see his information

Student can see his information , takes the exams and tests

4.4 context diagram :



EMP is the input of the system , instructor and student are the output from the system

4.5 :Orcle Database is the only database specially designed as an internet development and development platform , extending oracle's long-standing technology leadership in the areas of data management , transaction processing_ and data warehousing to the new medium of the internet.

We using oracle database and plsql.

4.5.1 : Creating Database:-

```
create table employee
```

```
(
```

```
    emp_id number(15),
```

```
    emp_name varchar2(25),
```

```
    emp_Gender varchar2(15),
```

```
    emp_image blob,
```

```
    emp_address varchar2(30),
```

```
    emp_phone varchar2(15),
```

```
    emp_user_name varchar2(25),
```

```
    emp_password varchar2(25),
```

```
constraint employee_emp_id_pk primary key (emp_id)
```

```
);
```

```
Create table band
```

```
(
```

```
    Band_id number(15),
```

```
    Band_title varchar2(25),
```

```
emp_id number(15),
constraint band_band_id_pk primary key (band_id),
constraint band_emp_id_fk foreign key (emp_id) references
employee (emp_id)
);
create table term
(
Term_id number(15),
term_title varchar2(25),
emp_id number(15),
band_id number(15),
constraint term_term_id_pk primary key (term_id),
constraint term_emp_id_fk foreign key (emp_id) references
employee (emp_id),
constraint term_band_title_fk foreign key (band_id) references
band (band_id)
);
```

```
create table department
(
Dept_id number(15),
Dept_name varchar2(50),
```

```
emp_id number(15),
band_id number(15),
term_id number(15),
constraint department_dept_id_pk primary key (dept_id),
constraint department_emp_id_fk foreign key (emp_id)
references employee (emp_id),
constraint department_band_id_fk foreign key (band_id)
references band (band_id),
constraint department_term_id_fk foreign key (term_id)
references term (term_id)
);
```

```
create table student
(
stu_id number(15),
stu_name varchar2(25),
stu_gender varchar2(15),
stu_phone varchar2(15),
stu_image blob,
stu_address varchar2(30),
stu_user_name varchar2(25),
stu_password varchar2(25),
```

```
dept_id number(15),  
emp_id number(15),  
band_id number(15),  
term_id number(15),  
constraint student_stu_id_pk primary key (stu_id),  
constraint student_emp_id_fk foreign key (emp_id) references  
employee (emp_id),  
constraint student_band_id_fk foreign key (band_id) references  
band (band_id),  
constraint student_term_id_fk foreign key (term_id) references  
term (term_id),  
constraint student_dept_id_fk foreign key (dept_id) references  
department (dept_id)  
);
```

```
create table instructor  
(  
inst_id number(15),  
inst_name varchar2(25),  
inst_gender varchar2(15),  
inst_image blob,  
inst_phone varchar2(15),  
inst_address varchar2(30),
```

```
inst_user_name varchar2(25),  
inst_password varchar2(25),  
dept_id number(5),  
emp_id number(15),  
constraint instructor_inst_id_pk primary key (inst_id),  
constraint instructor_dept_id_fk foreign key (dept_id) references  
department (dept_id),  
constraint instructor_emp_id_fk foreign key (dept_id) references  
employee (emp_id)  
);
```

```
create table stu_inst  
(  
stu_id number(15),  
inst_id number(15),  
constraint stu_inst_stu_id_fk foreign key(stu_id) references  
student(stu_id),  
constraint stu_inst_inst_id_fk foreign key(inst_id) references  
instructor(inst_id)  
);
```

```
create table course  
(
```

```
Course_id number(15),
course_name varchar2(50),
inst_id number(15),
dept_id number(15),
band_id number(15),
term_id number(15),
constraint course_course_id_pk primary key (course_id),
constraint course_dept_id_fk foreign key (dept_id) references
department (dept_id),
constraint course_band_id_fk foreign key (band_id) references
band (band_id),
constraint course_term_id_fk foreign key (term_id) references
term (term_id),
constraint course_inst_id_fk foreign key (inst_id) references
instructor (inst_id)
);
```

```
create table stu_course
(
stu_id number(15),
course_id number(5),
constraint stu_course_stu_id_fk foreign key(stu_id) references
student(stu_id),
```

```
constraint stu_course_course_id_fk foreign key(course_id)
references course(course_id)

);
```

```
create table exam

(
Question_id varchar2 (15),
exam_id number(15),
question_type varchar2(50),
question varchar2(200),
answer1 varchar2(200),
answer2 varchar2(200),
answer3 varchar2(200),
answer4 varchar2(200),
correct_answer varchar2(200),
question_degree varchar2(50),
total_degree varchar2(50),
inst_id number(15),
stu_id number(15),
course_id number(15),
constraint exam_Question_id_pk primary key (Question_id),
```

```
constraint exam_inst_id_fk foreign key (inst_id) references
instructor (inst_id),
constraint exam_stu_id_fk foreign key (stu_id) references student
(stu_id),
constraint exam_course_id_fk foreign key (course_id) references
course(course_id)
);
```

```
create table exam_result
(
exam_res_id number(15),
stu_id number(15),
Question_id varchar2 (15),
course_id number(5),
constraint exam_result_exam_res_id primary key (exam_res_id),
constraint exam_result_stu_id foreign key (stu_id) references
student (stu_id),
constraint exam_result_course_id foreign key (course_id)
references course (course_id),
constraint exam_result_Question_id foreign key (Question_id)
references exam (Question_id)
);
```

create or replace view student_information_desplay
as
select
stu_id,stu_name,stu_gender,stu_phone,stu_image,stu_adress,stu
_user_name,stu_password,band_title,term_title,dept_name

from student,department,band,term
Where department.dept_id = student.dept_id and band.band_id
= student.band_id and term.term_id=student.term_id ;

create or replace view instructor_information_desplay
as
select inst_id,inst_name,inst_gender,inst_image,inst_phone,inst_a
dress,inst_user_name,inst_password,dept_name
from instructor,department
Where department.dept_id = instructor.dept_id ;

End of Chapter 4

Thanks



**HOW
TO...**

→ Chapter 5 :

HOW TO RUN QPS SYSTEM

5.1.How to run QPS System :

A- Powering up the network switch.

B- Power on the network server:

- (1)When logon screen appear
- (2)Type user name: fciqpsteam_2015
- (3)Type password: fciqpsteam_2015

C- Power on the application server:

- (1) When logon screen appear
- (2)Type user name: fciqpsteam_2015
- (3)Type password:fciqpsteam_2015

D- Then the employee , instructor and student can run the system

In this chapter we will show the role of each one can use QPS System .

5.2.The Employee:

-He is the main role in the system , he only can control the whole System and any person will use the System and he only can remove or add any thing in system . The employee has the full authority to deal with the system :

- (1) Employee has password only used by him no one can used it .
- (2) Employee add instructor to the system , instructor can't use system before going to employee to take its data to complete the Registration process ,then employee will give instructor its user name and password .

- (3) Employee also add student to the system , employee take all information about student like email , phone , address , its grade and department , then employee give him its user name and password.
- (4) Employee can add courses that the students will examin in the system .

-Employee is an inner admin and the manager to all process will run in QPS System.

5.3. The Student:

After employee give student its user name and password he can use the system ,he can see its information that the employee take it and if he find any mistake he can report the employee to check the mistake and correct it .

Student can start the exam that the instructor made and student will start the exam when instructor determine the time that the exam will done .

Student will open the system , instructor will send the exam and he will examine in a determined time instructor given after finish exam the system will calculate the result and show the student its mistakes in the exam and the final result of this exam or the pervious exams .

5.4. The Instructor:

Instructor also like student can see its information and if find any mistake he can report employee to correct it .

Instructor put the exam questions by choosing the course he want to put its exam and he can see the exams that he have put before .

If instructor hasn't time to put new exam , he can chosen any exam that he make before so he will save time and effort .

This Project presents the Questions Pool System QPS . The System consists of a computers wireless network and a database management systems base on oracle database. This database is used to build the questions pool and to handle all questions and evaluation of answers.

End of Chapter 5

Thanks



→ Chapter 6 :

Design of QPS

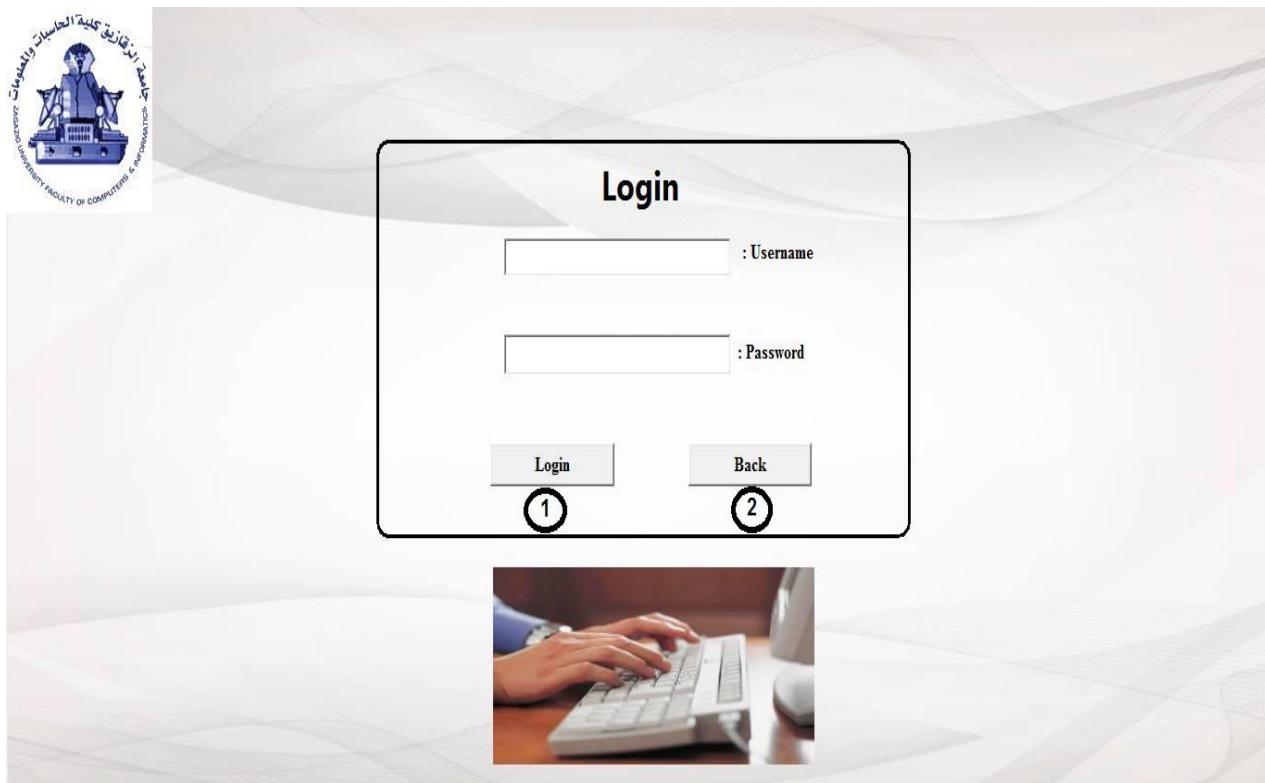
6.1 Design of QPS:

6.1.1. Designing include :

- 1-design input form.
- 2-design output form.
- 3-design user interface form.

6.1.2. Design input forms :

During analysis we have recognized the following input forms



- 1- Press this button to close the screen
- 2- Press this button to back to the main menu

This is the code of this form:

```

declare

    y varchar2 (60);
begin
    select stu_user_name , stu_password into :GLOBAL.x,y
    from student
    where stu_user_name = :STUDENT.STU_USER_NAME and
        stu_password = :STUDENT.STU_PASSWORD;

    if :GLOBAL.x=:STUDENT.STU_USER_NAME and y= :STUDENT.STU_PASSWORD then
        OPEN_FORM('STUDENT.fmx',ACTIVATE);

    end if;
end;

```

To Add Student :

The form is titled "جامعة العلوم والتكنولوجيا البحرينية" and "UNIVERSITY FACULTY OF COMPUTING & INFORMATION". It contains the following fields:

- Student Image:** An input field with a placeholder "Add image".
- Student ID:** An input field.
- Student Name:** An input field.
- Student Username:** An input field.
- Password:** An input field.
- Employee ID:** An input field.
- Term ID:** An input field.
- Student Phone:** An input field.
- Department ID:** An input field.
- Band ID:** An input field.
- Address:** An input field.
- Gender:** An input field.

At the bottom are four buttons:

- Exit (4):** A button labeled "Exit" with a circled number 4 below it.
- Save (3):** A button labeled "Save" with a circled number 3 below it.
- delete (2):** A button labeled "delete" with a circled number 2 below it.
- Add (1):** A button labeled "Add" with a circled number 1 below it.

1-press this button to add new student to the database.

2-press this button to delete the page in order to enter another student.

3-press the save button to save the changes you have made after entering the student details

4-press this button to exit form

The code of add button:

```
go_block('student');  
last_record;  
next_record;
```

The code of save button:

```
commit;
```

The code of delete button:

```
go_block('student');  
delete_record;  
previous_record;
```

Edit Student:

The screenshot shows a web-based application for managing student records. At the top left is the logo of the Faculty of Computer Sciences at King Fahd University of Petroleum & Minerals. To its right are two search inputs: one for 'name' and one for 'ID', both preceded by a magnifying glass icon. Below these is a 'Search' button. The main area displays a table with columns for Student Password, Student Username, Student Phone, Student Gender, Student Name, and Student ID. There are 10 rows of data in the table. At the bottom right of the page is a small navigation element containing the number '76'.

Student Password	Student Username	Student Phone	Student Gender	Student Name	Student ID

In this form employee will search to find student by name and ID and it will appear its information then employee can edit the information about student.

The code of this form:

```

declare
  x varchar2(20);
  y varchar2(20);
  z varchar2(20);
  m varchar2(20);
begin
  x := 'stu_name like'|| chr(39) || :student.name || chr(39) ;
  z := 'stu_id like' || chr(39) || :student.id || chr(39);
  y := :student.id;
  m := :student.name;
  if y is null then
    SET_BLOCK_PROPERTY('student',DEFAULT_WHERE, upper(x));
    execute_query;
  elsif m is null then
    SET_BLOCK_PROPERTY('student',DEFAULT_WHERE, upper(z));
    execute_query;
  elsif y is null and m is null then
    message('enter data');
    message('enter data');
  else
    SET_BLOCK_PROPERTY('student',DEFAULT_WHERE, 'stu_name like '''||upper (:student.NAME)||''' and '|| 'stu_id like '''|| upper(:student.ID)|| '''' );
    execute_query;
  end if;

end;

```

Add Instructor:

- 1-press this button to add new instructor to the database.
- 2-press this button to delete the page in order to enter another instructor.
- 3-press the save button to save the changes you have made after entering the instructor details.
- 4-press this button to exit form.

Edit Instructor:

In this form employee will search to find instructor by name and ID and it will appear its information then employee can edit the information about instructor.

The code of this form :

```

declare
  x varchar2(20);
  y varchar2(20);
  z varchar2(20);
  m varchar2(20);
begin
  x := 'inst_name like'|| chr(39) || :instructor.name || chr(39) ;
  z := 'inst_id ='|| chr(39) || :instructor.id || chr(39);
  y := :instructor.id;
  m := :instructor.name;
  if y is null then
    SET_BLOCK_PROPERTY('instructor',DEFAULT_WHERE, upper(x));
    execute_query;
  elsif m is null then
    SET_BLOCK_PROPERTY('instructor',DEFAULT_WHERE, upper(z));
    execute_query;
  elsif y is null and m is null then
    message('enter data');
    message('enter data');
  else
    SET_BLOCK_PROPERTY('instructor',DEFAULT_WHERE, 'inst_name like '''||upper (:instructor.NAME)||''' and '|| 'inst_id ='''|| upper(:instructor.ID)||'''');
    execute_query;
  end if;

end;

```

Add Course:

The screenshot shows a web page titled "Add Course". At the top left is the logo of the "Cairo University Faculty of Computers & Information". The page has a light background with a wavy pattern. There are six input fields labeled from left to right: "Course ID", "Course Name", "Instructor ID", "Band ID", "Department ID", and "Term ID". Below these fields is a row of four buttons, each with a circled number below it: "Exit" (4), "Save" (3), "Delete" (2), and "Add" (1). The "Add" button is highlighted with a red oval.

1-press this button to add course to the database.

2-press this button to delete the page in order to enter another course.

3-press the save button to save the changes you have made after entering the course details.

4-press this button to exit form.

Add Band Title, Band id:

Employee ID	Band Title	Band ID

[Next](#) [Add](#)

1-press this button to add band title and band id.

2-press this button to add department id and title.

Term ID	Band ID	Employee ID	Department Name	Department ID
TERM_ID	BAND_ID	EMP_ID	DEPT_NAME	DEPT_ID
TERM_ID	BAND_ID	EMP_ID	DEPT_NAME	DEPT_ID
TERM_ID	BAND_ID	EMP_ID	DEPT_NAME	DEPT_ID

[Add student](#) |
 [Back](#) |
 [Add and save](#)

this form to add department id and department title ,after add save the changes.

The code for this forms is same but change the name of rh part we can add:

```

-- Begin default enforce data integrity constraint DEPARTMENT_DEPT_ID_PK section
--

declare
    cursor primary_cur is select 'x' from QPS.STUDENT
        where DEPT_ID = :DEPARTMENT.DEPT_ID;
    primary_dummy char(1);
begin
    if ( (:DEPARTMENT.DEPT_ID is not null) ) then
        open primary_cur;
        fetch primary_cur into primary_dummy;
        if ( primary_cur%found ) then
            message('Cannot delete master record when matching detail records exist.');
            close primary_cur;
            raise form_trigger_failure;
        end if;
        close primary_cur;
    end if;
end;

-- End default enforce data integrity constraint DEPARTMENT_DEPT_ID_PK section
--

-- Begin default enforce data integrity constraint DEPARTMENT_DEPT_ID_PK section
--


declare
    cursor primary_cur is select 'x' from QPS.INSTRUCTOR
        where DEPT_ID = :DEPARTMENT.DEPT_ID;
    primary_dummy char(1);
begin
    if ( (:DEPARTMENT.DEPT_ID is not null) ) then
        open primary_cur;
        fetch primary_cur into primary_dummy;
        if ( primary_cur%found ) then
            message('Cannot delete master record when matching detail records exist.');
            close primary_cur;
            raise form_trigger_failure;
        end if;
        close primary_cur;
    end if;
end;

-- End default enforce data integrity constraint DEPARTMENT_DEPT_ID_PK section
--


-- Begin default enforce data integrity constraint DEPARTMENT_DEPT_ID_PK section
--


declare
    cursor primary_cur is select 'x' from QPS.COURSE
        where DEPT_ID = :DEPARTMENT.DEPT_ID;
    primary_dummy char(1);
begin
    if ( (:DEPARTMENT.DEPT_ID is not null) ) then
        open primary_cur;
        fetch primary_cur into primary_dummy;
        if ( primary_cur%found ) then
            message('Cannot delete master record when matching detail records exist.');
            close primary_cur;
            raise form_trigger_failure;
        end if;
        close primary_cur;
    end if;
end;
delete_record;
-- End default enforce data integrity constraint DEPARTMENT_DEPT_ID_PK section
--

```

Add Employee:



The form interface includes:

- An "Add image" button above a placeholder box for the Employee Image.
- A text input field for Employee ID.
- A text input field for Employee Name.
- A text input field for Employee Username.
- A text input field for Password.
- A text input field for Gender.
- A text input field for Address.
- A text input field for Phone.
- Four buttons at the bottom: Exit (labeled 4), Save (labeled 3), Delete (labeled 2), and Add (labeled 1).

1-press this button to add new employee to the database.

2-press this button to delete the page in order to enter another employee.

3-press the save button to save the changes you have made after entering the employee details.

4-press this button to exit form.

The code of this form:

```
:global.id := :employee.emp_id ;  
open_form('navigation.fmx');
```

Create New Exam:

1-Instructor will choose the course he want to create a new exam for it .

2-press button Next to put the exam questions.

Add Questions:

add questions and answers of the new exam.

The code of enter button:

```
go_block('EXAM');
```

commit;

Start Exam:



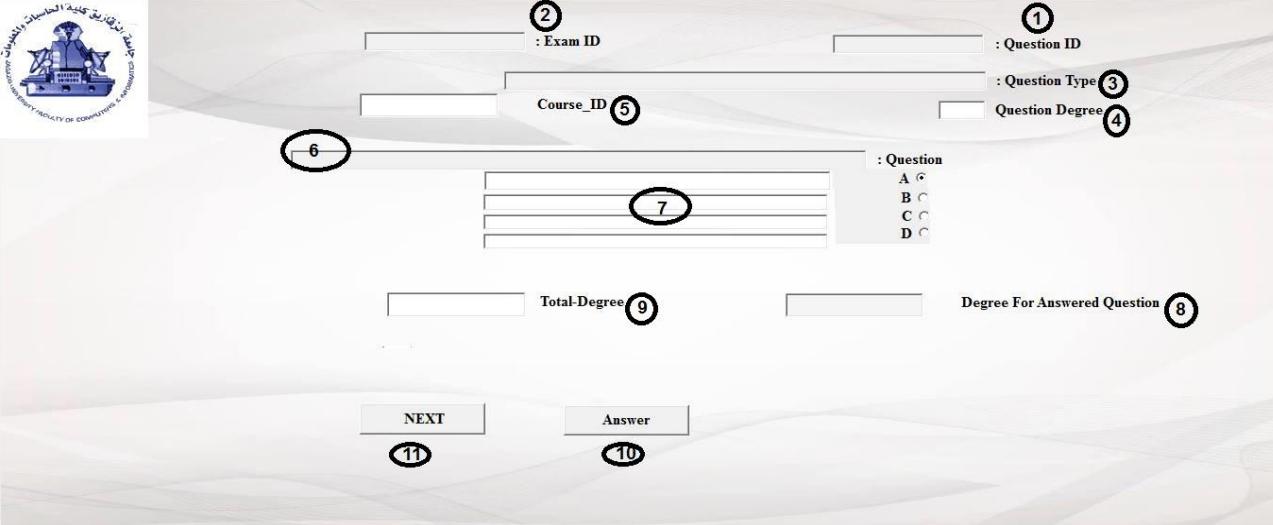
A screenshot of a web-based exam start page. At the top left is the university's logo. In the center, there is a dropdown menu labeled "Choose Course". Below the dropdown are two buttons: "Start Exam" on the left and "Back" on the right. The background features a light blue and white wavy pattern.

Student choose the name of course he want to exam.

The code of this form:

General	
■ Name	RECORD_GROUP7
» Subclass Information	
» Comments	
Functional	
■ Record Group Type	Query
■ Record Group Query	select course_name,to_char(Course_id)id from course
» Record Group Fetch Size	20
■ Column Specifications	

When Student press the button of start exam :



A screenshot of the exam interface after the student has selected a course. The screen includes several input fields and buttons, each circled with a number from 1 to 11:

- 1: Question ID
- 2: Exam ID
- 3: Question Type
- 4: Question Degree
- 5: Course_ID
- 6: Question (radio buttons A, B, C, D)
- 7: Answer input field
- 8: Degree For Answered Question
- 9: Total-Degree
- 10: Answer button
- 11: NEXT button

- 1-the question id appear.
- 2-the exam id will appear.
- 3-the question type appear.
- 4-the degree of the question.
- 5-the course id appears.
- 6-the question itself appear.
- 7-the answers to choose from.
- 8-the degree of question.
- 9-the degree of all exam.
- 10-press to answer the question.
- 11-press next to go to the next question.

The code of this form:

```

declare
  degree number(20) := 0;
  answer varchar2(100);
  ansse varchar2(100);
  TO_NUM number (20);
begin
if :EXAM.ANSWERS = 1 then
  :Global.m := :ANS1;
elsif :EXAM.ANSWERS = 2 then
  :Global.m := :ANS2;
elsif :EXAM.ANSWERS = 3 then
  :Global.m := :ANS3;
elsif :EXAM.ANSWERS = 4 then
  :Global.m := :ANS4;
else
  degree :=0;
end if;

select CORRECT_ANSWER into :Global.v from exam where QUESTION_ID = :QUESTION_ID;
:GLOBAL.counter := :Global.counter + 1 ;

if :GLOBAL.m = :global.v then

  if :Global.counter = :QUESTION_ID then
    :Global.sum_deg := :Global.sum_deg + 1;
  :DISPLAY_ITEM21 := :Global.sum_deg;
  elsif :Global.counter < :QUESTION_ID then

select Question_id,exam_id,question_type,question,answer1,answer2,answer3,answer4,question_degree into :QUESTION_ID,:EXAM_ID,:QUESTION_TYPE,:QUESTION,:ANS1
,:ANS2,:ANS3,:ANS4,:QUESTION_DEGREE from exam where Question_id = :Global.counter and course_id = :Global.selected_course_id;
  :Global.counter := :QUESTION_ID;

```

```

else
    degre :=0;
end if;

select CORRECT_ANSWER into :Global.v from exam where QUESTION_ID = :QUESTION_ID;

:GLOBAL.counter := :Global.counter + 1;

if :GLOBAL.m = :global.v then

    if :Global.counter = :QUESTION_ID then
        :Global.sum_deg := :Global.sum_deg + 1;
        :DISPLAY_ITEM21 := :Global.sum_deg;
        elsif :Global.counter < :QUESTION_ID then

            select Question_id,exam_id,question_type,question,answer1,answer2,answer3,answer4,question_degree into :QUESTION_ID,:EXAM_ID,:QUESTION_TYPE,:QUESTION,:ANS1
            ,:ANS2,:ANS3,:ANS4,:QUESTION_DEGREE from exam where Question_id = :Global.counter and course_id = :Global.selected_course_id;
            :Global.counter := :QUESTION_ID;

        else
            message ('Error');
            message ('Error');
            :Global.counter := :QUESTION_ID;

        end if ;

        degre :=1;
        :DISPLAY_ITEM16 :=degre;

    else

        degre :=0;
        :DISPLAY_ITEM16 :=degre;
    end if;

```

6.1.3.Design output form :

During analysis we have recognized the following output form:

: Student Image

: Student ID

MOHAMED : Student Name

MALE : Gender

: Phone

ZAGZAG : Address

SUCCESS : Student State

1 : Total-Degree

100 : % Percent

: Band Title

: Term Title

: Department Name

Exit

This is the final Form will appear to The student after finishing exam.

The code of this form :

```

BEGIN
SELECT stu_id,stu_name,stu_gender,stu_phone,stu_adress INTO :STU_ID,:STU_NAME,:STU_GENDER,:STU_PHONE,:STU_ADDRESS FROM STUDENT WHERE stu_user_name=:GLOBAL.x;

:DISPLAY_ITEM4 := :Global.sum_deg ;

:DISPLAY_ITEM5 := (((to_number(:Global.sum_deg))/10)*100);

if :Global.sum_deg < 6 then

:DISPLAY_ITEM6 := 'Fail';

else

:DISPLAY_ITEM6 := 'SUCCESS';

end if;
END;

```

Student Information:

The form displays the following data:

- : Student Image: Placeholder box
- : Username: MOH
- : Name: MOHAMED
- : ID: ١٢٣
- : Password: ١٢٣
- : Gender: MALE
- : Phone: ٠٠٠
- : Adress: ZAGZAIG

Buttons at the bottom left include "Back" and "Next".

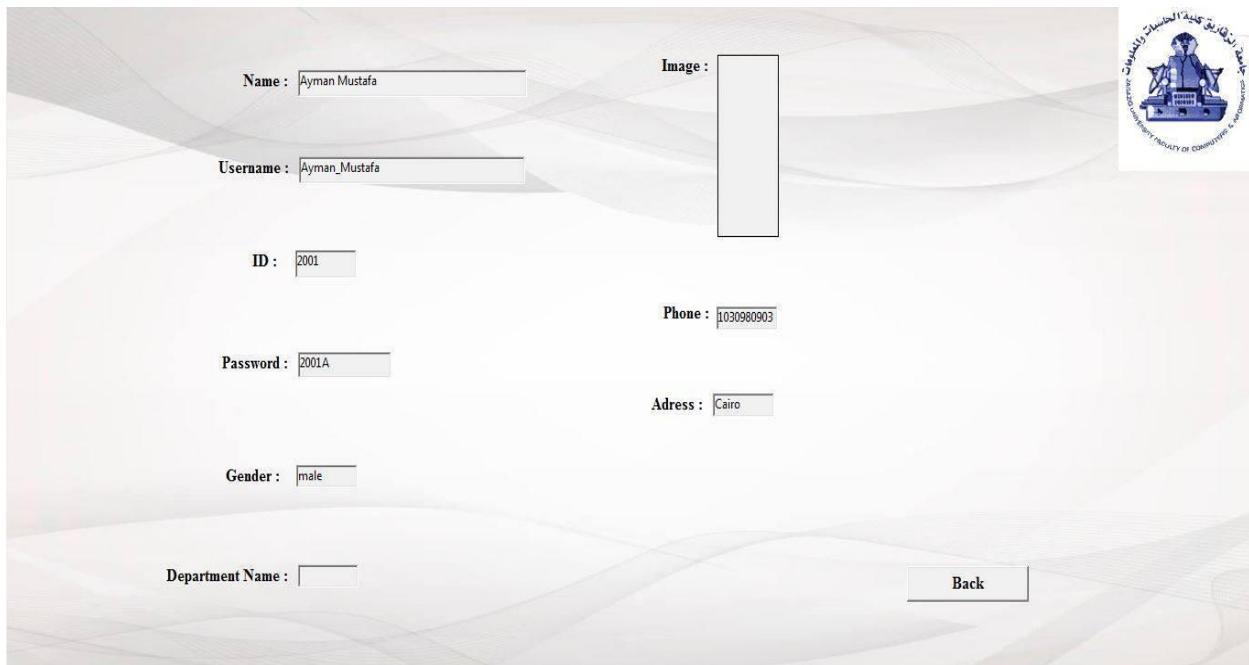
The code of this Form:

```

BEGIN
SELECT stu_id,
stu_name,
stu_gender,
stu_phone,
stu_adress,stu_user_name,
stu_password INTO :STU_ID,:STU_NAME,:STU_GENDER,:STU_PHONE,:STU_ADDRESS,:STU_USER_NAME,:STU_PASSWORD FROM STUDENT WHERE stu_user_name=:GLOBAL.x;
END;

```

Instructor Information Form:



A screenshot of an Instructor Information Form. The form includes fields for Name (Ayman Mustafa), Username (Ayman_Mustafa), ID (2001), Password (2001A), Phone (1030980903), Address (Cairo), Gender (male), and Department Name (left empty). There is also a placeholder for an Image. The background features a light blue gradient with wavy patterns. In the top right corner, there is a circular logo for "Cairo University Faculty of Computer & Information".

Name : Ayman Mustafa

Image :

Username : Ayman_Mustafa

ID : 2001

Phone : 1030980903

Password : 2001A

Address : Cairo

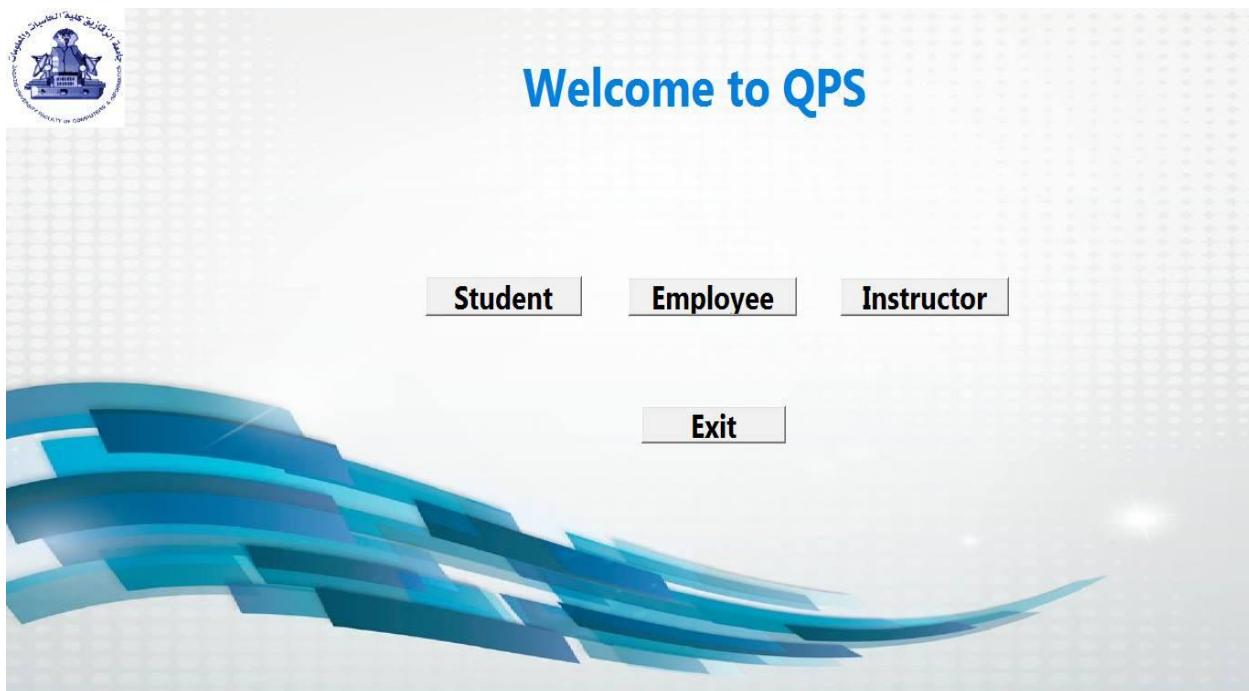
Gender : male

Department Name :

Back

6.1.4.Design user interface:

Welcome Interface:

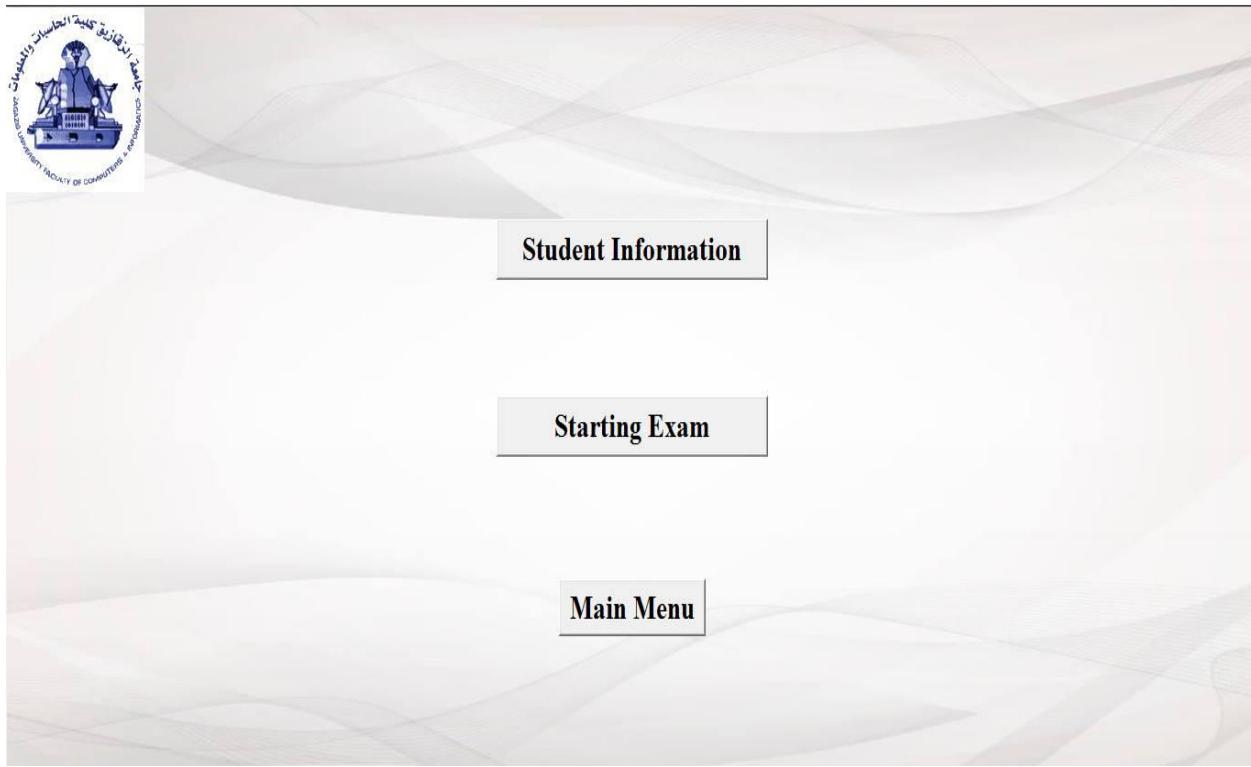


1-press student button to enter the student options to edit or add or search for student.

2-press instructor button to enter the instructor options to edit or add or search for instructor.

3-press employee button to enter the employee options to edit or add.

Student Interface:



1-press Student information to show the information about student.

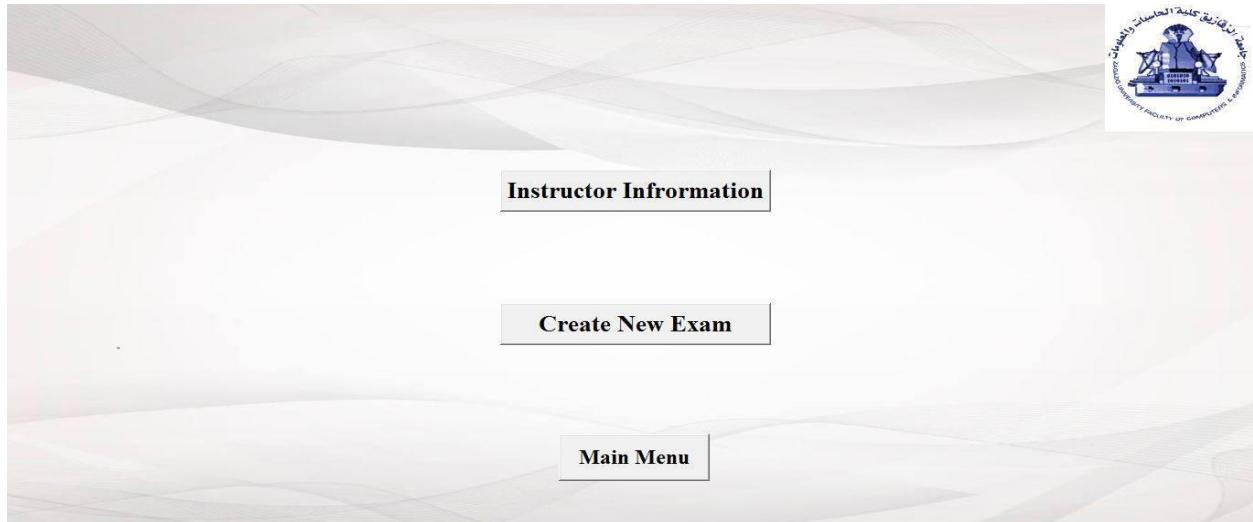
2-press starting exam to go to start exam form.

3-press main menu to go back to main menu form.

Employee interface:



Instructor Interface:



End of Chapter 6

Thanks



→ Chapter 7 :

Network

7.1 : Setting up Windows Server 2003 enterprise edition :

- A-Boots From Windows 2003 CD**
- B-Press enter to Setup afresh Copy**
- C-Press F8 to apply the policy from Microsoft**
- D-Chose Disk Drive that need to install on it and press enter**
- E-When relational and language screen appear in customize tab choose language and the place and press next**
- F- Type name and organization name and press next**
- G-Type the product key and press next**
- H-Choose the licensing mode per server or per device or per user and press next**
- I- Type Computer Name and administrative Password and press next**
- J-Choose The Time and press next**
- K-In network settings choose typical settings and press next**
- L-In workgroup or computer domain choose workgroup and press next**

7.2 : To Install or reinstall the Windows Server 2003 Administration Tools Pack :-

- A- Put the windows server 2003 CD into The CD tray of a computer running windows XP professional**
- B- The CD installation setup runs automatically .if it does not:**

- 1**-Click start , and then click Run
- 2**- in the run dialog box ,click browse
- 3**-in the browse dialog box click My computer
- 4**-Double-Click the CD drive ,and then Double -click setup exe
- 5**-IN the run dialog box ,click OK

- C**-In the Welcome to Microsoft Windows server 2003 dialog box ,click perform additional tasks
- D**- In the , what do you want to do ? Dialog box ,click Browse this CD
- E**- Double Click the i386 folder

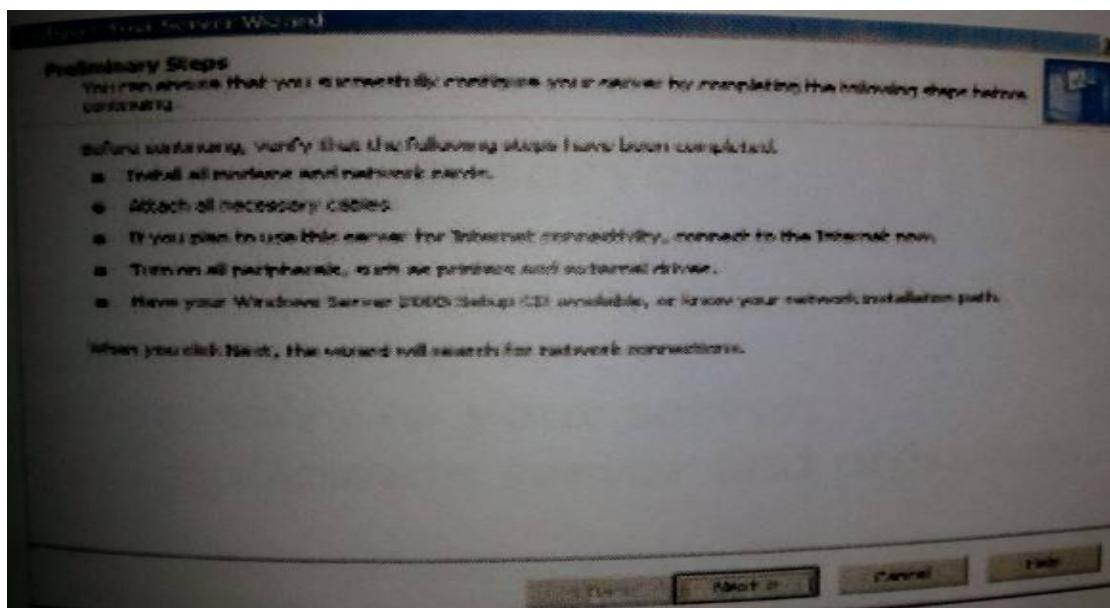


- F**-Double -Click the Adminpak.msi icon
- G**-Specify the installation location or drive where want to install the windows server 2003 Administration Tools Pack

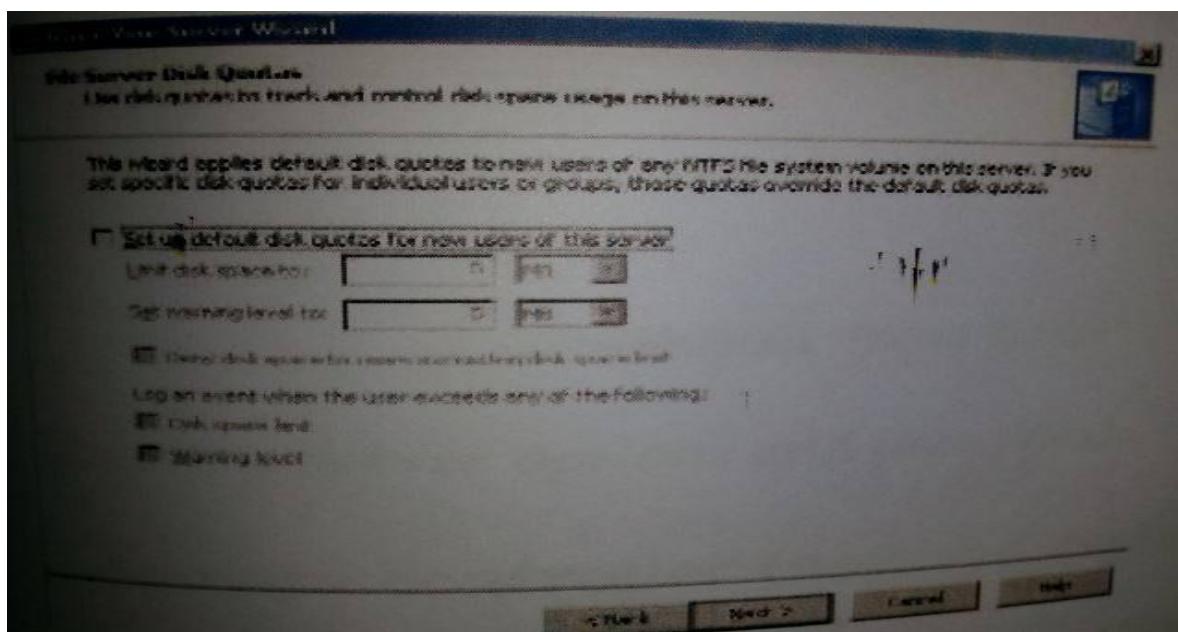
7.3 :Setting up network Servers :

7.3.1 : Installing file server :-

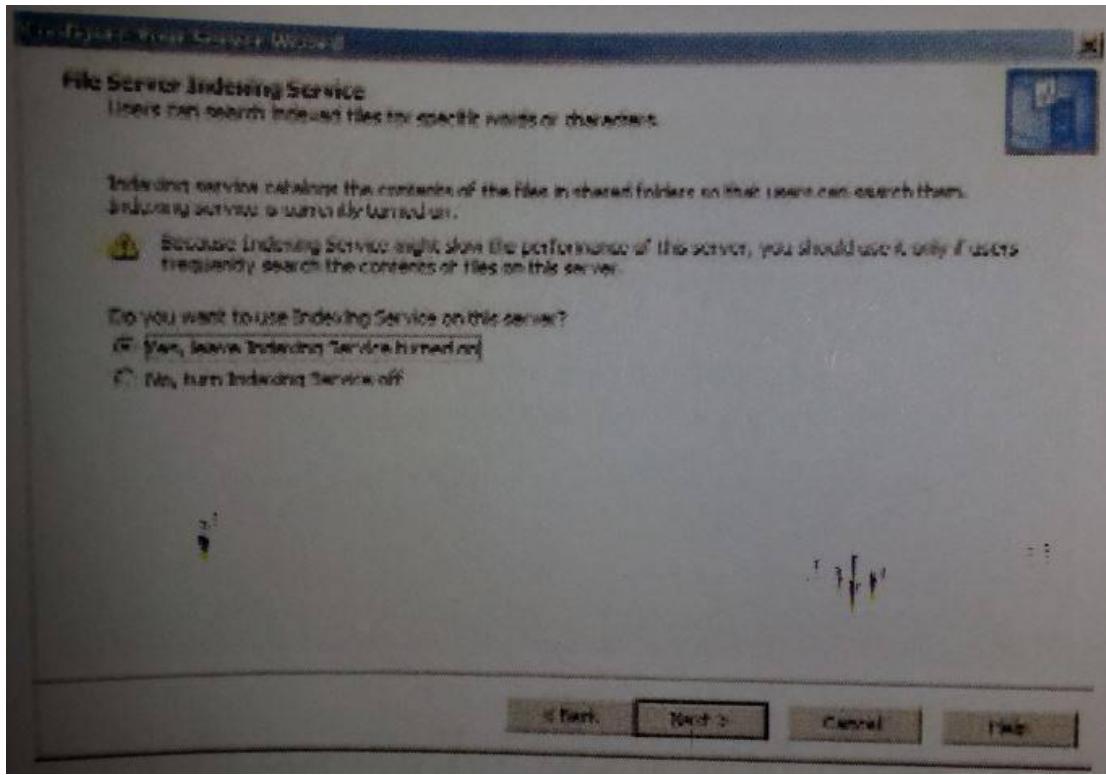
- A-From the windows start menu ,select Administrative tools
- B-Clicks on manage your server
- C-Clicks on adding roles to your server



D- Choose install file server and press next



E-Choose to run on the indexing Service on the server and press next



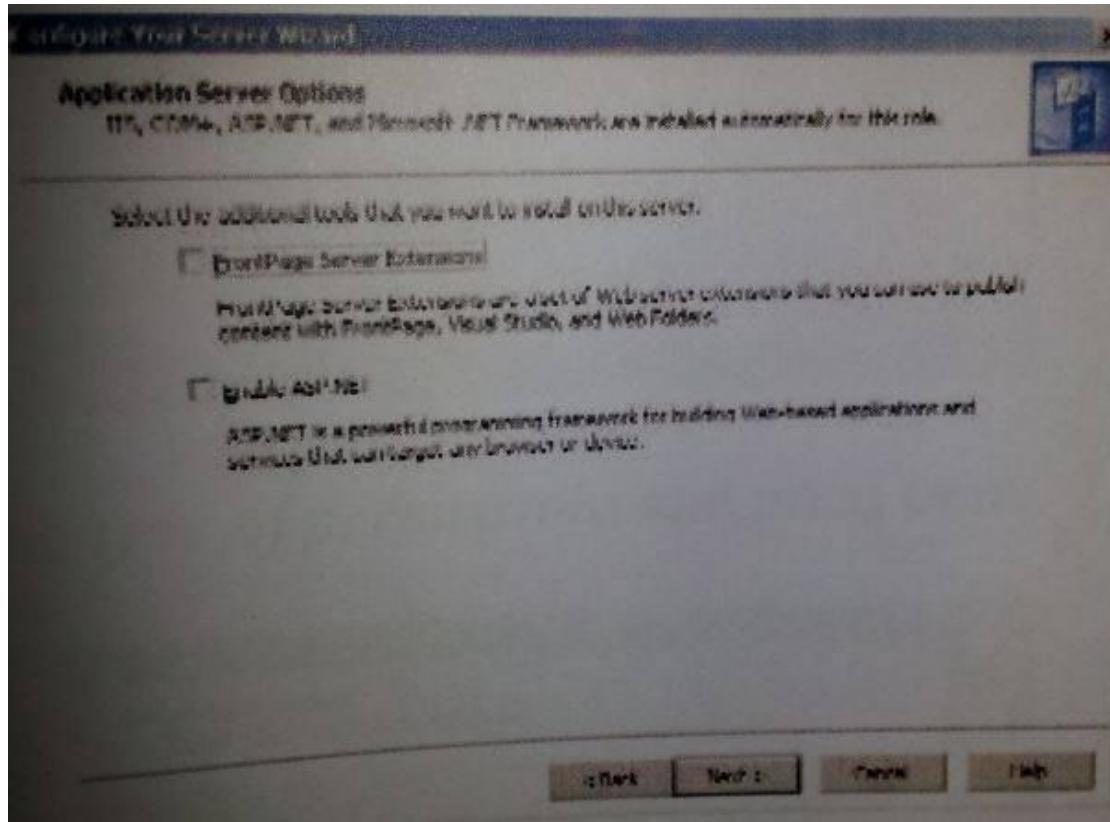
7.3.2 :Install application server :

A-from the windows Start menu , select administrative tools

B-Clicks on manage your server

C-Clicks on adding roles to your server

D-Choose install application server and press next



E-Choose application server options and press next

7.3.3-install domain controller (active directory) :

A-From the windows Start menu , Select administrative tools

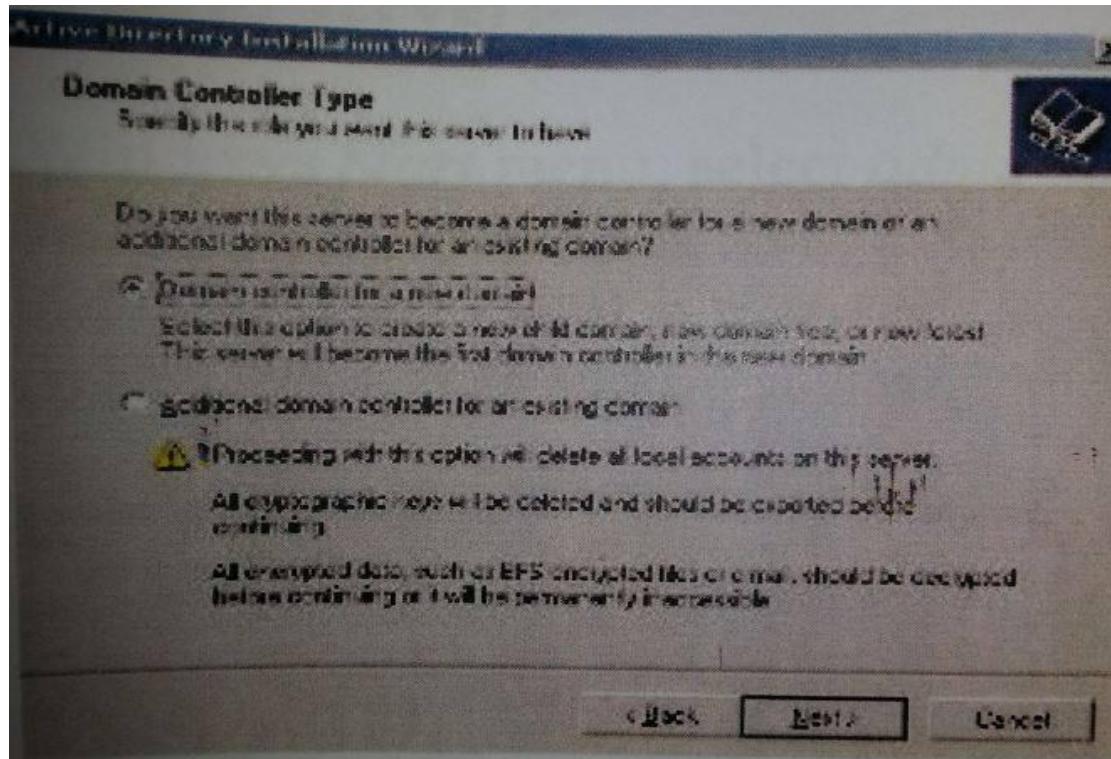
B-Clicks on manage your server

C-Clicks on adding roles to your server

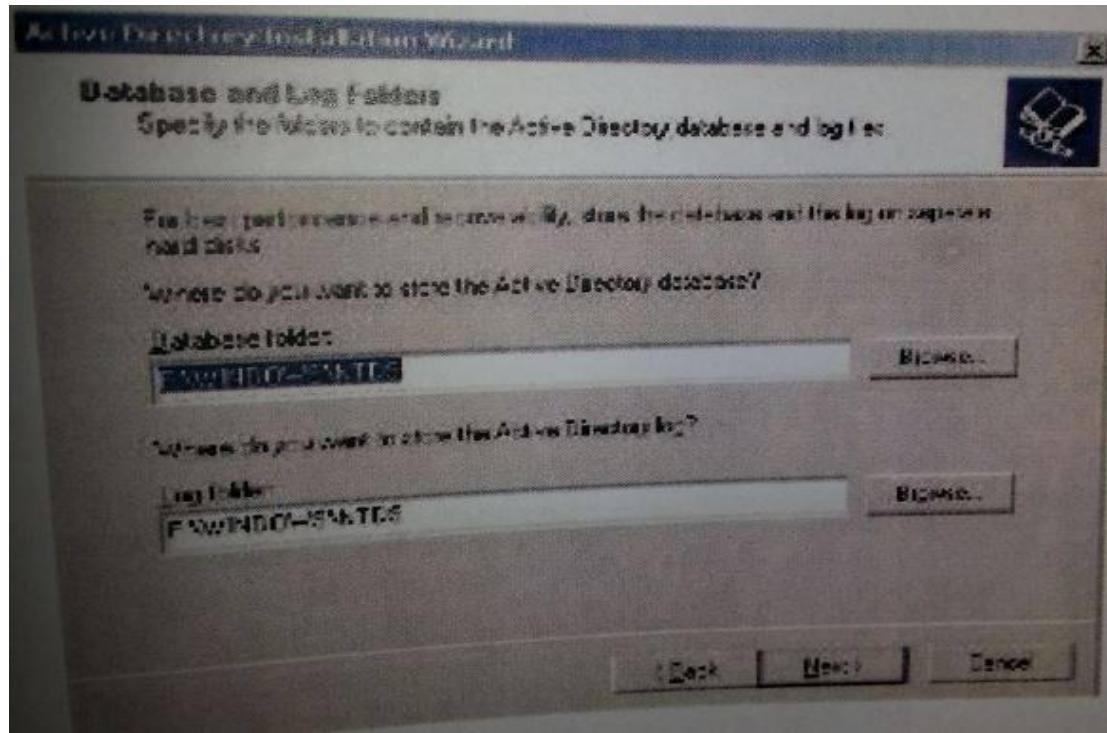
D-Choose install domain controller (Active directory) and press next

E-Choose Domain Controller type and press next

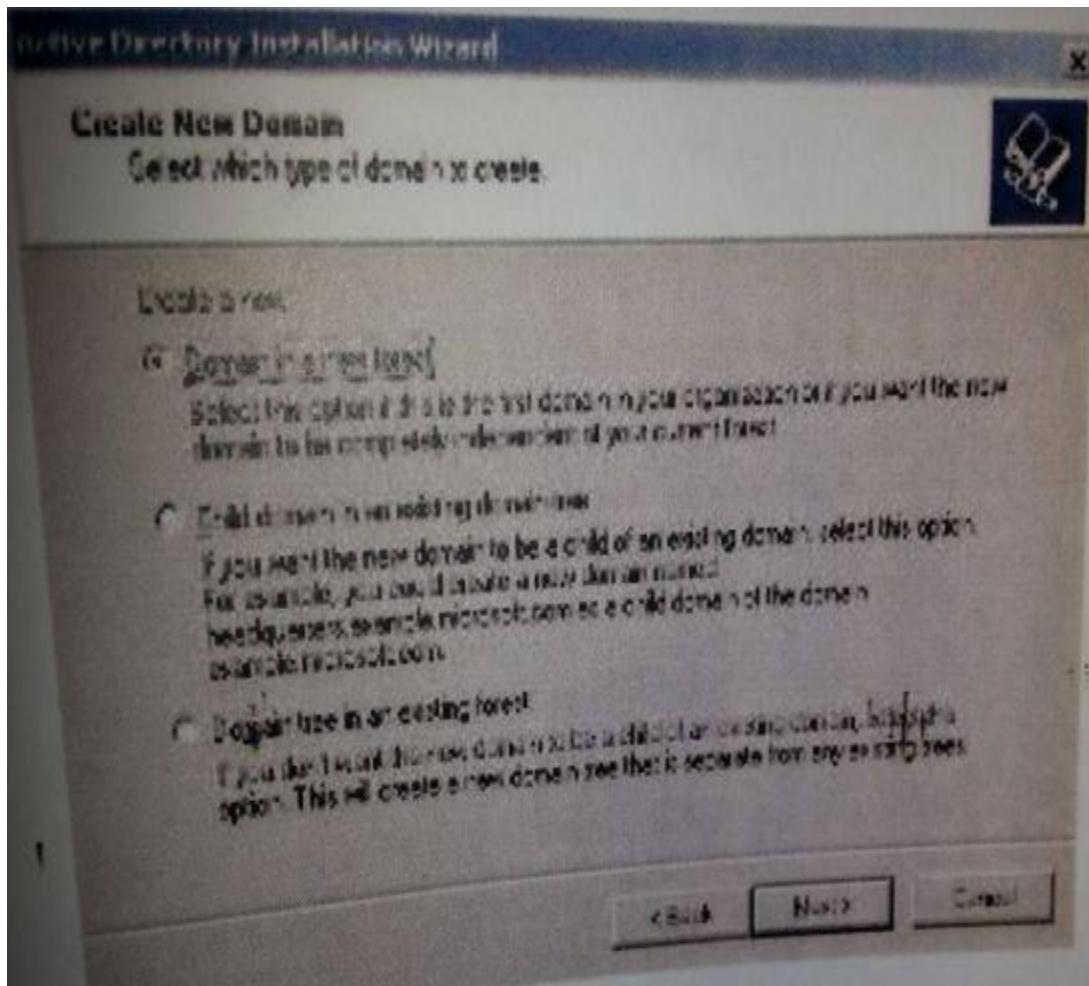
F-Choose Create domain in a new forest and press next



G-Type the domain controller name and press next
H-Specify database and log folders place and press next

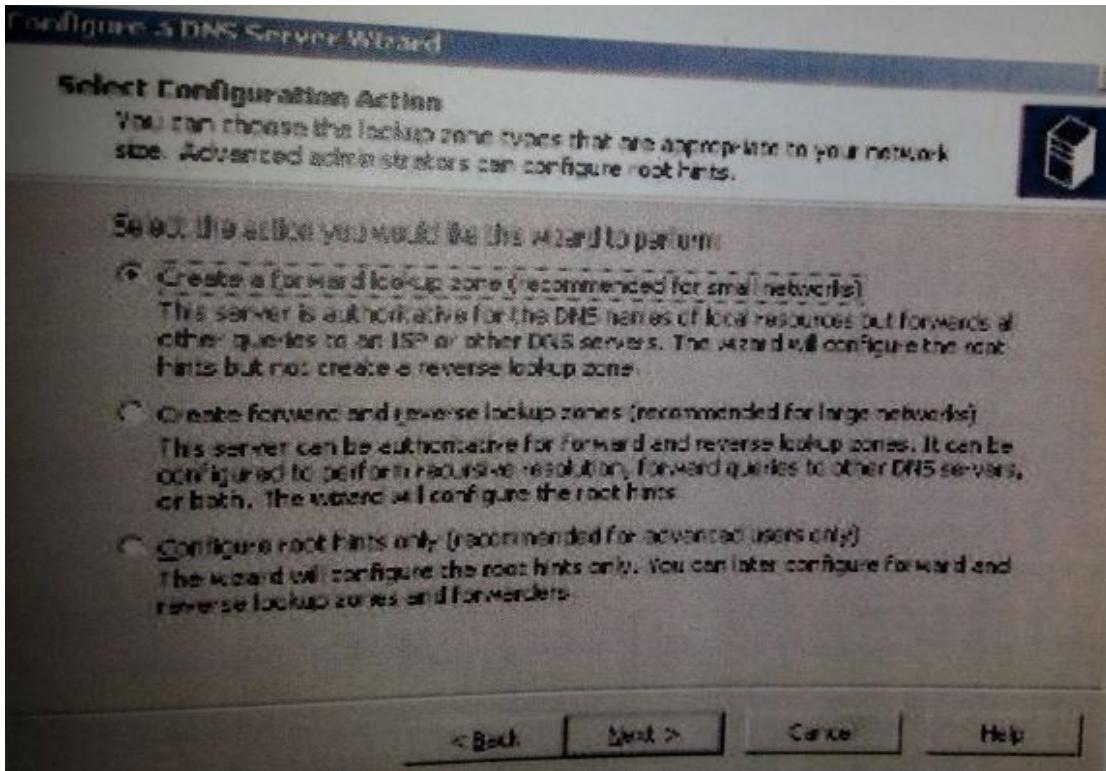


I-Choose the type of permissions and press next

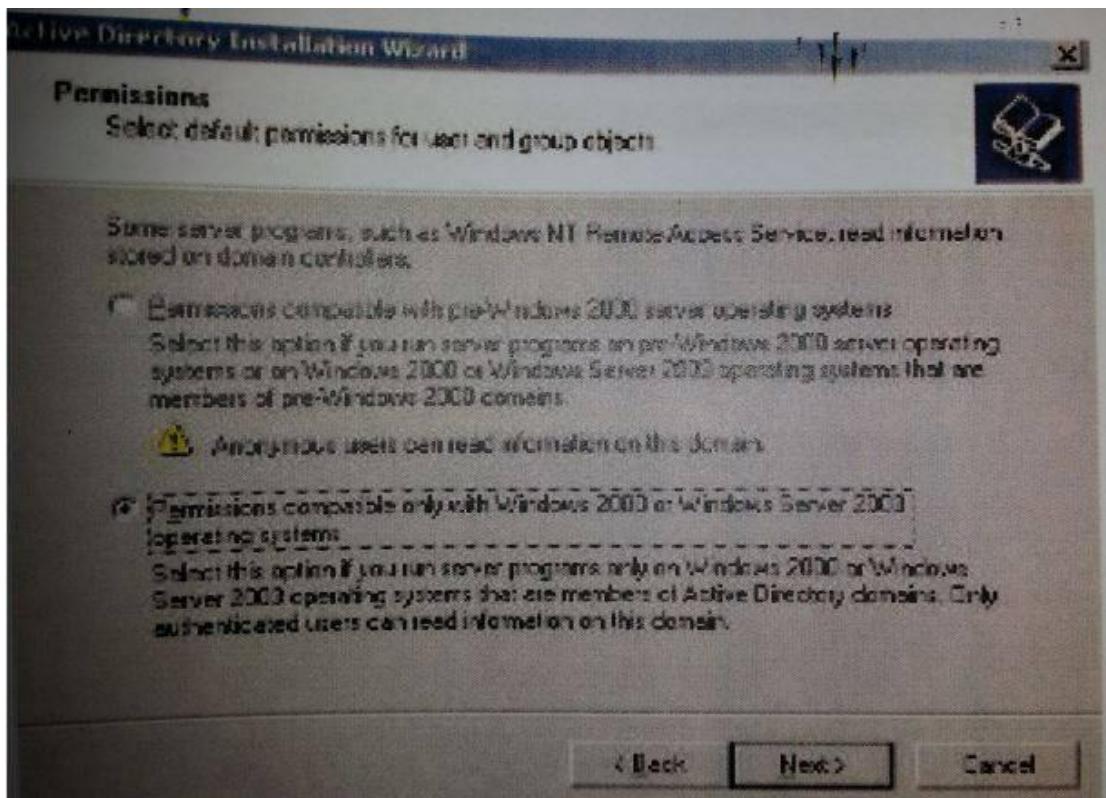


7.3.4 :Install DNS Server :

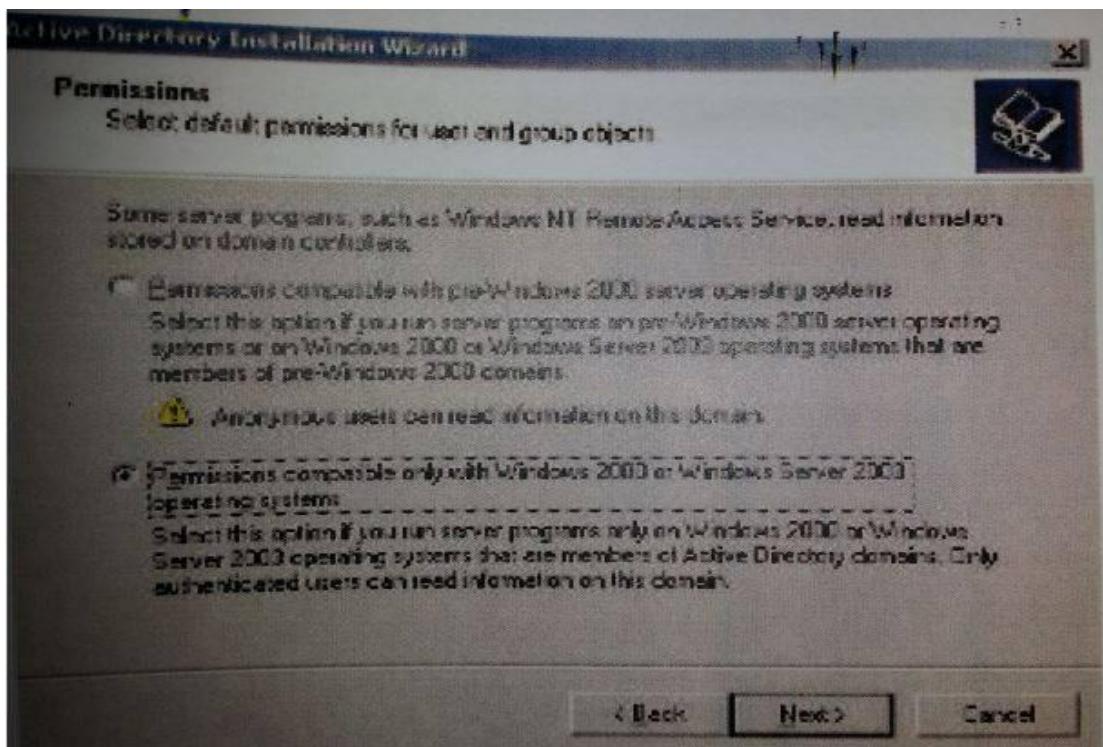
- A-From the windows start menu ,select administrative tools
- B-Clicks on manage your server
- C-Clicks on adding roles to your server
- D-Choose install DNS Server and press next



E-Choose DNS Options and press next

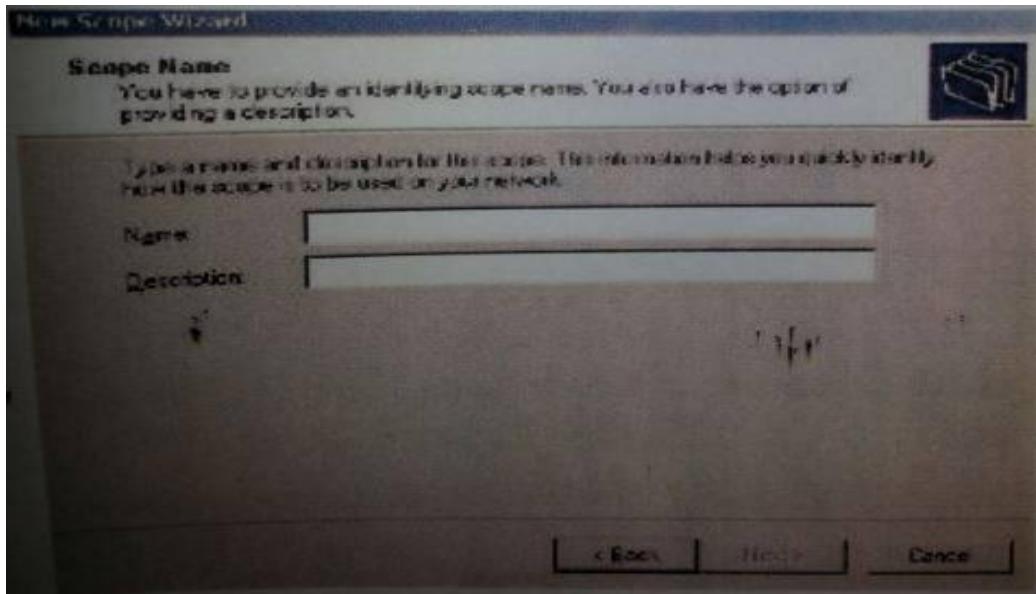


In configure DNS server wizard choose create a forward lookup zone and press next.

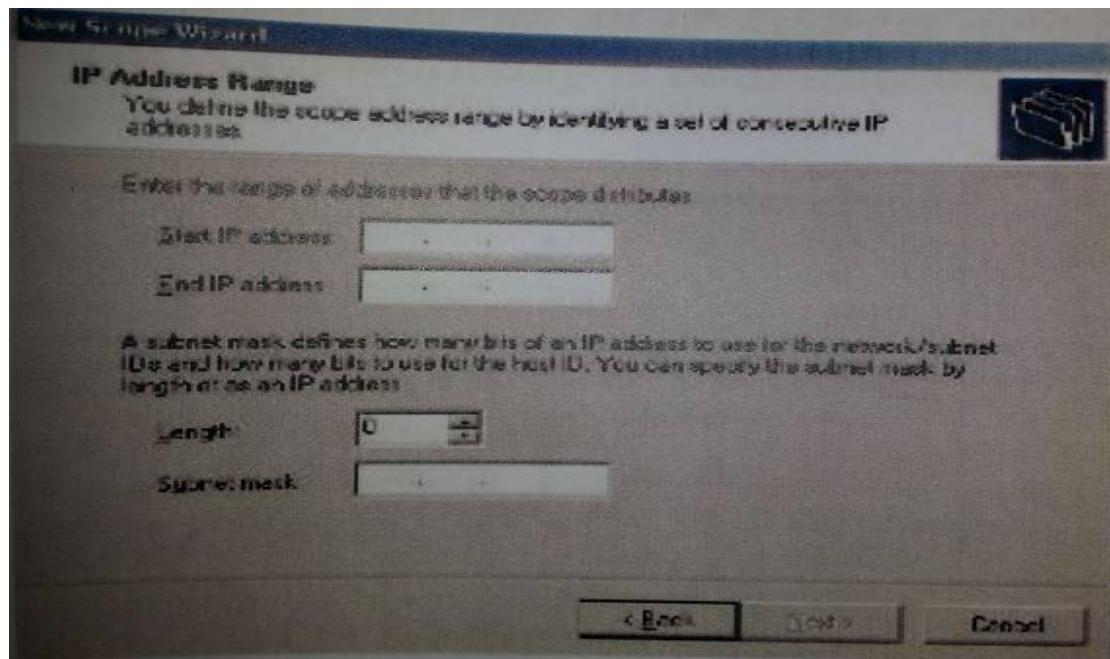


7.3.5. Install DHCP server :

- A-From the Windows Start menu , select administrative tools.
- B- Clicks on manage your server.
- C-Clicks on adding roles to your server.
- D-Choose install DHCP server and press next
- E-Choose DHCP options and press next.
- F-Type the scope name and description and press next.



Specify the scope IP type the start IP and the end IP and the length and press next.



7.4 : Configuring the network Card :

- A- Connect the card in the USB port.
- B-Double click windows start menu, select settings > control panel.

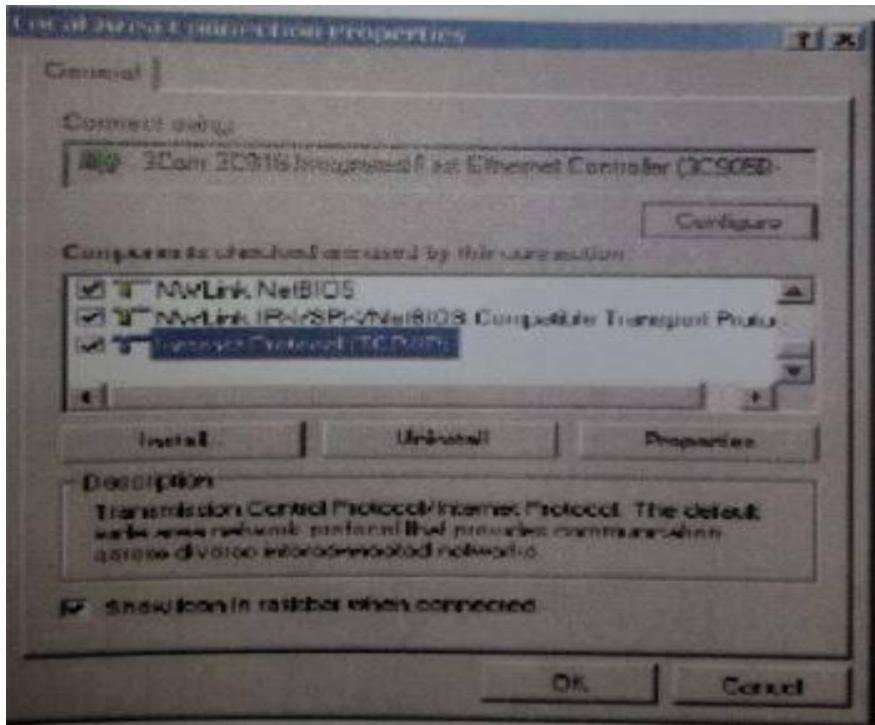
- C**-Double click on system.
- D**-Click on Hardware.
- E**-Click on Device manger.
- F**-Hit scan for hardware changes.
- G**-Put the CD in the CD Rom.
- H**-Locate the driver and press NEXT.
- I**-Click FINISH.

7.5 :Adding computer to a secured wireless network :

- A**-Double click windows start menu, select Settings > Control Panel.
- B**-Double click on network connection.
- C**-Choose the wireless connection >then properties.
- D**-Type the Encryption Key.
- E**-Press connect to be a part of this wireless network.

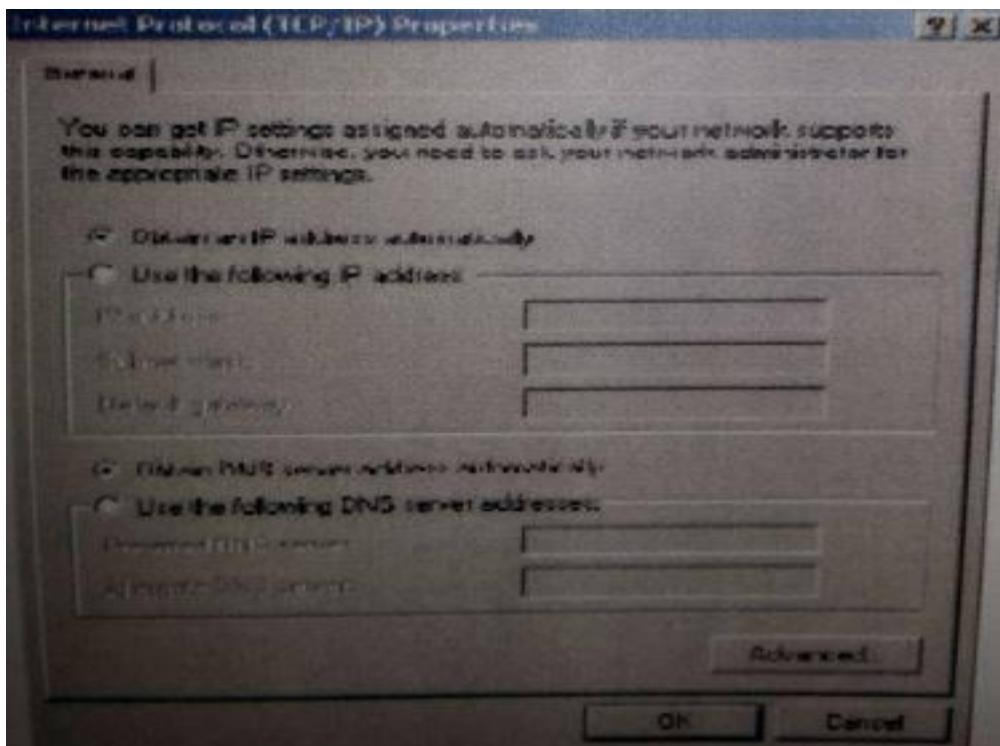
7.6 :Configure a network card Windows 2003 :

- A**-From the on Network and Dial-Up Connections.
- B**-Double click on Local Area Connection.
- C**-Double click Windows Start Menu, select Settings > Control Panel.
- D**-Clicks on Properties
- E**-A screen should be displayed. Select internet protocol TCP/IP and click on properties.
- F**-Ensure that the options obtain an IP Address automatically, and obtain DNS server address automatically is both selected, Click OK.
- G**-Restart computer.



7.7 :Configure a network card in windows XP :

- A-From the widows start menu, select control panel .
- B-Click on the network connections icon.
- C-Double click on LAN or High speed connection icon . A screen titled local Area connection status will appear.
- D-Select Internet protocol TCP/IP and click on properties.
- E-Ensure that the options obtain an IP Address automatically.
- F-Restart computer.



7.8 :Configure a network card in windows 95/98/ME :

- 1-From the windows start menu , select settings > control panel.
- 2-Double click on network. Select the TCP/IP item for network card and click on properties.
- 3-In the TCP/IP dialog, select the IP address tab, and ensure that obtain address automatically is selected. Click OK.

7.9 :Adding Computer to the domain:

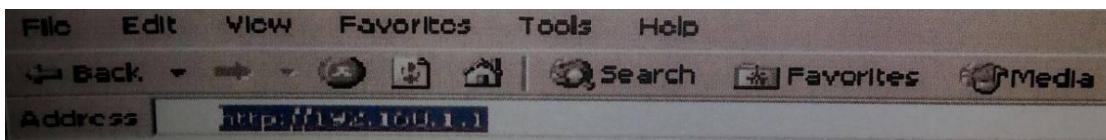
- 1-Double click windows start menu , select settings > control panel.
- 2-Double click on system.
- 3-Clicks on computer name tape.
- 4-Press change > type the domain name
- 5-Click ok and exit.

7.10 :Running the Router setup Wizard:

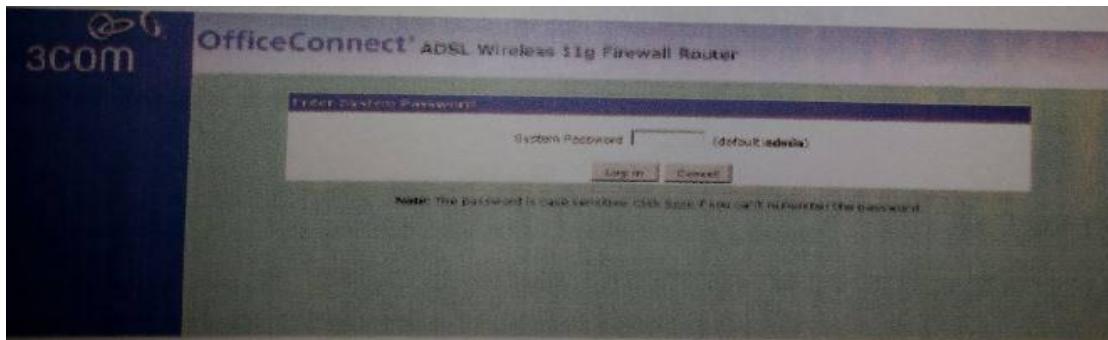
The router setup program is Web-based, which means that it is accessed through Web browser (Netscape Navigator 4.7 or higher, Internet Explorer 5.0 or higher, or Mozilla 1.2.1 or higher).

7.10.1 :To use the setup wizard:

- 1**-Ensure that have at least one computer connected to the Router.
- 2**-Launch Web Browser on the Computer.
- 3**-Enter the following URL in the location or address field of browser: <http://192.168.1.1> The login Screen Displays.



- 4**-To log in as an administrator, enter the password in the password field and click log in .



- 5**-When have logged in either :
The status screen will appear. Selected Setup wizard from the menu.

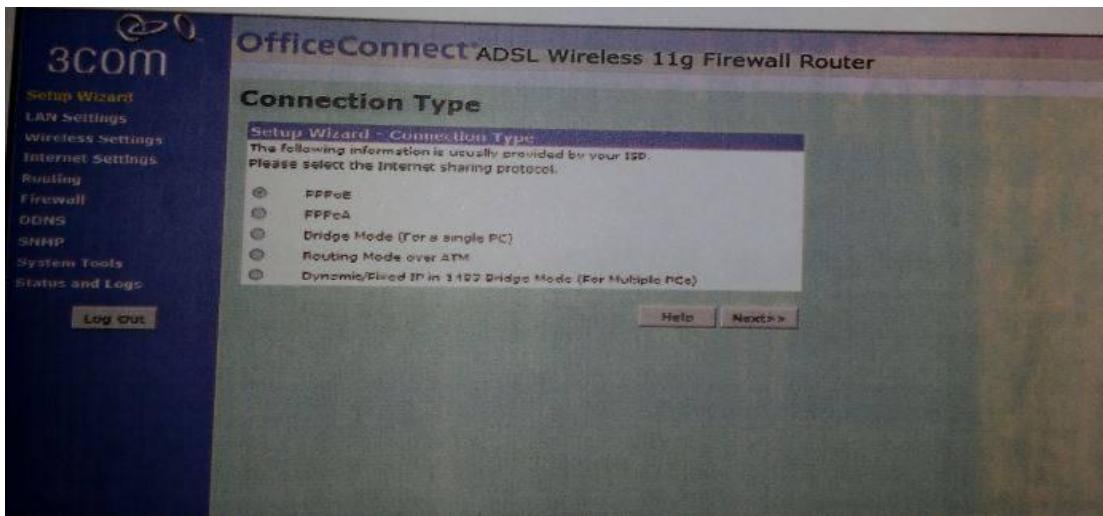
Or

If Router has not been configured before, the Wizard will launch automatically

6-Will be guided step by step through a basic setup procedure.

7.10.2 :Setup Wizard – Connection Type :

The Connection Type screen allows you to set up the Router for the type modifies Internet connection you have. Before setting up your connection type, have your account information from your ISP ready.



7.10.3 :Dynamic/Fixed IP (For Multiple PCS) :

1-Check the Get Wan IP by DCHP field if ISP allocation IP addresses using DHCP.

2-Enter Internet IP address in the WAN IP text box.

3-Enter The subnet mask in the Subnet Mask text box.

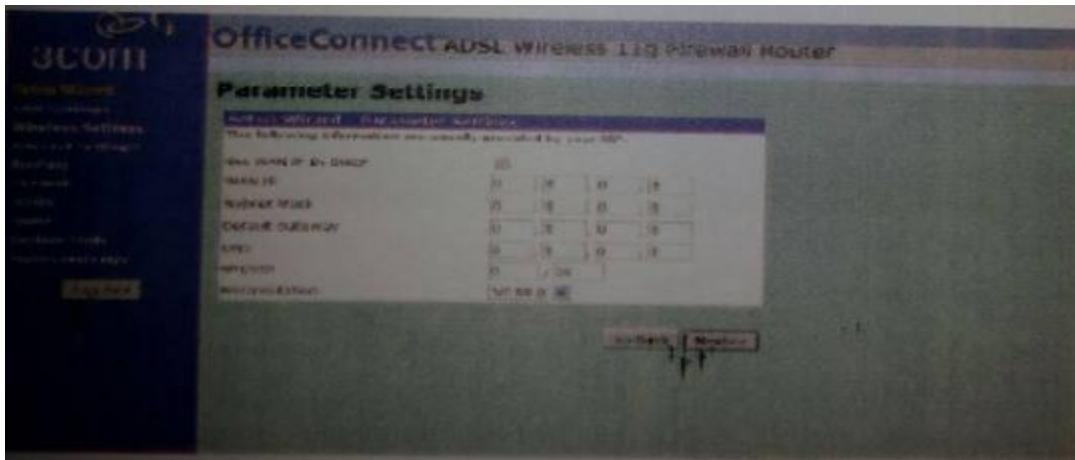
4-Enter the default router in the Default Gateway text box.

5-Enter the DNS address in the DNS text box.

6-Enter VPI and VCI information in the VPI/VCI text boxes.

7-Select the encapsulation type in the Encapsulation drop-down list.

This information will have been provided to by ISP.



8-Check all of settings, and then click next. The wireless settings screen is displayed.

9-Set the wireless channel want to use from the channel drop-down list.

10-Specify the SSID to be used by wireless network in the SSID field.

There are other wireless networks in area, should give wireless network a unique name.

7.11 :The LAN Settings screen is used to specify the LAN IP address of the router, and to configure the DHCP server :

1-Select LAN Settings and then specify the router IP Address and Subnet Mask in the appropriate fields. The default IP address of the Router is 192.168.1.1.

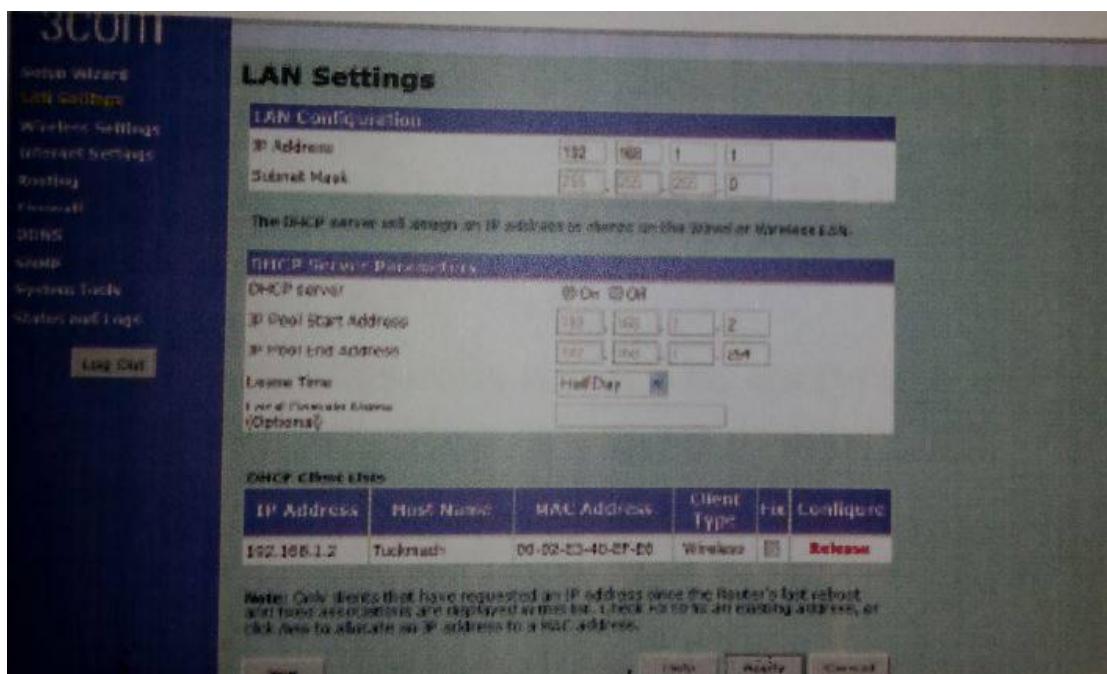
2-If want to use the Router as a DHCP server, click in the on check radio button.

3-If need to, anyone can change the range of addresses given out by the Router by changing the IP Pool start address and IP pool end address fields.

4-Specify the DHCP Lease time by selecting the required value from the lease time drop down list. The lease time is the length of time the DHCP server will reserve the IP address for each computer.

5-Specify the local domain name for the network.

6-Check all settings , and then click Apply.



7.11.1 :Configuring Wireless :

1- Set the Wireless Channel that want to use from the channel drop-down list.

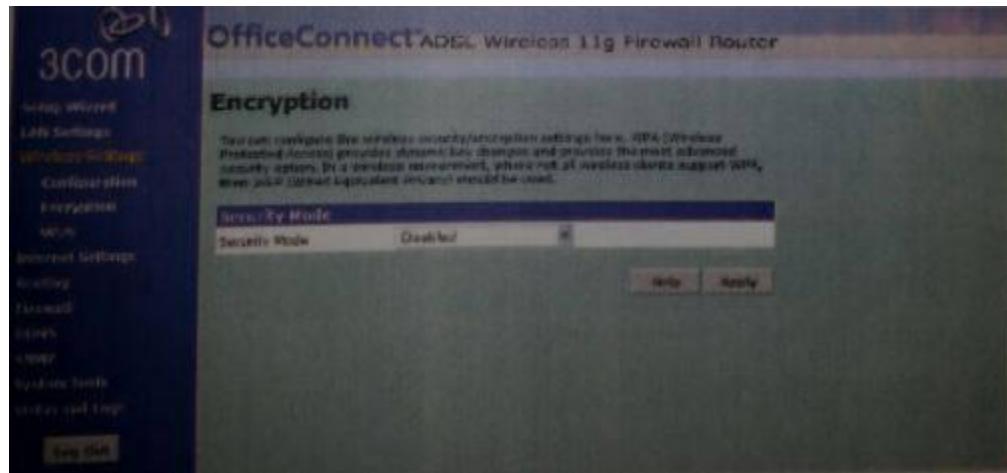
2-Specify the SSID to be used by the Wireless network in the SSID field. If there are other wireless networks in the area, should give the wireless network a unique name.

3-Enable or disable SSID broadcast.

4-In the Wireless Mode drop down list , select whether router will operate in 11b mode only, 11g mode only, or mixed 11b and 11g .

5-Clicks Apply.

7.11.2 :Wireless Security (Encryption) :



from this screen , you can select the wireless security mode that you wish to use.

there are five possible selections :

Disabled

WPA-PSK(no server)

128-bit WEP

64-bit WEP

WPA(with RADIUS server)

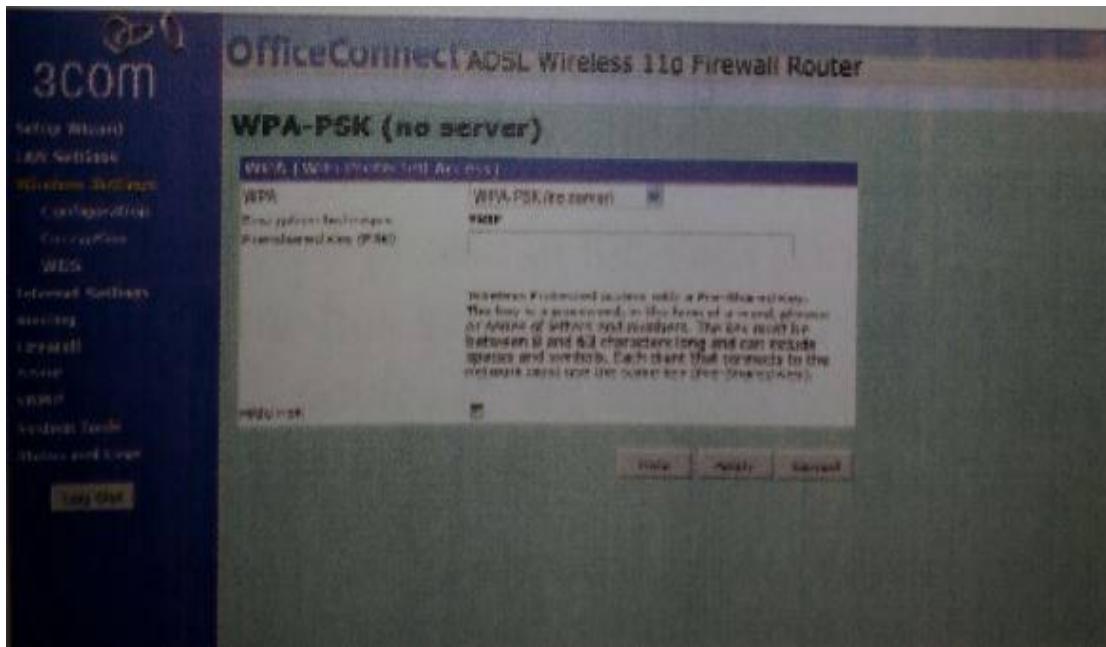
Select the required value from the drop down list, and press Apply.

DISABLED :

In this mode ,wireless transmissions will not be encrypted , and will be visible to everyone. However, when setting up or debugging wireless networks it's often useful to use this

security mode.

WPA-PSK(no server) :



WPA(WiFi Protected Access) provides dynamic key changes and constitutes the best security solution.

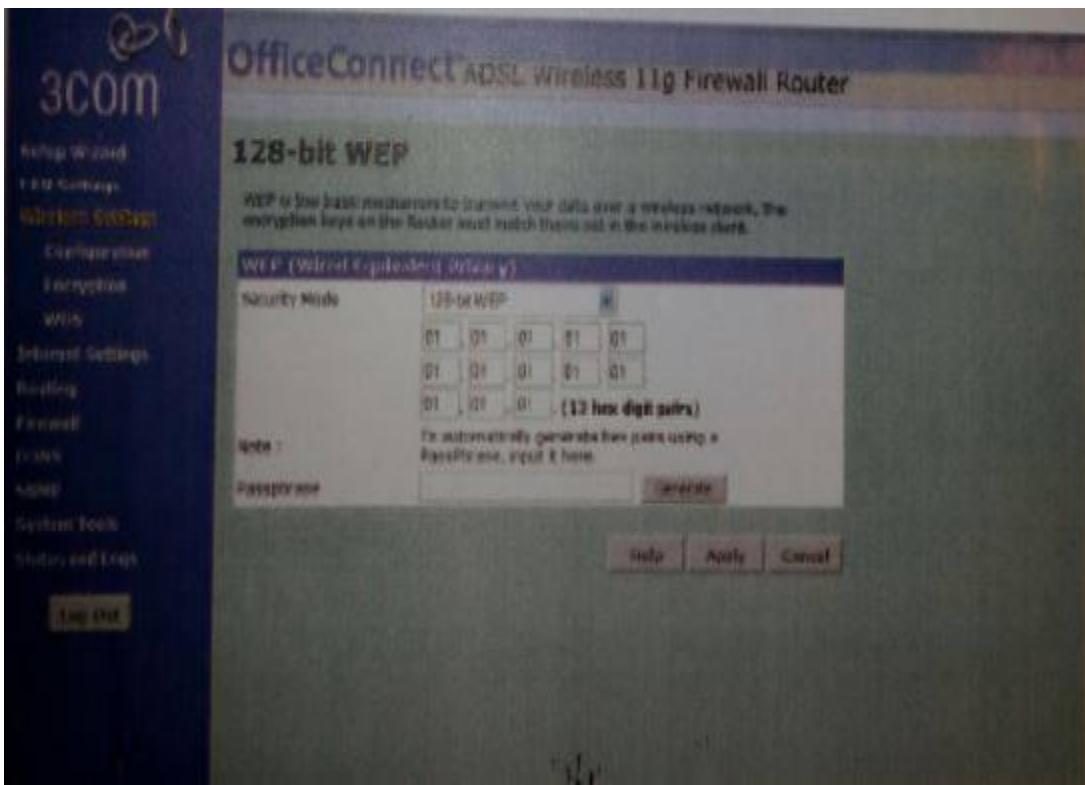
In a wireless network where not all devices support WPA, WEP (Wired Equivalent Privacy) should be used.

1 - Enter the pre-shared key (PSK) field. The pre-shared key is a password, in the form of word, phrase or series of letters and numbers. The key must be between 8 and 63 characters long and can include spaces and symbols

2- Optionally, check the Hide PSK check box, if you want the key that you enter to be shown on the screen as a series of asterisks(*).

3 - Clicks Apply.

128-bit WEP :

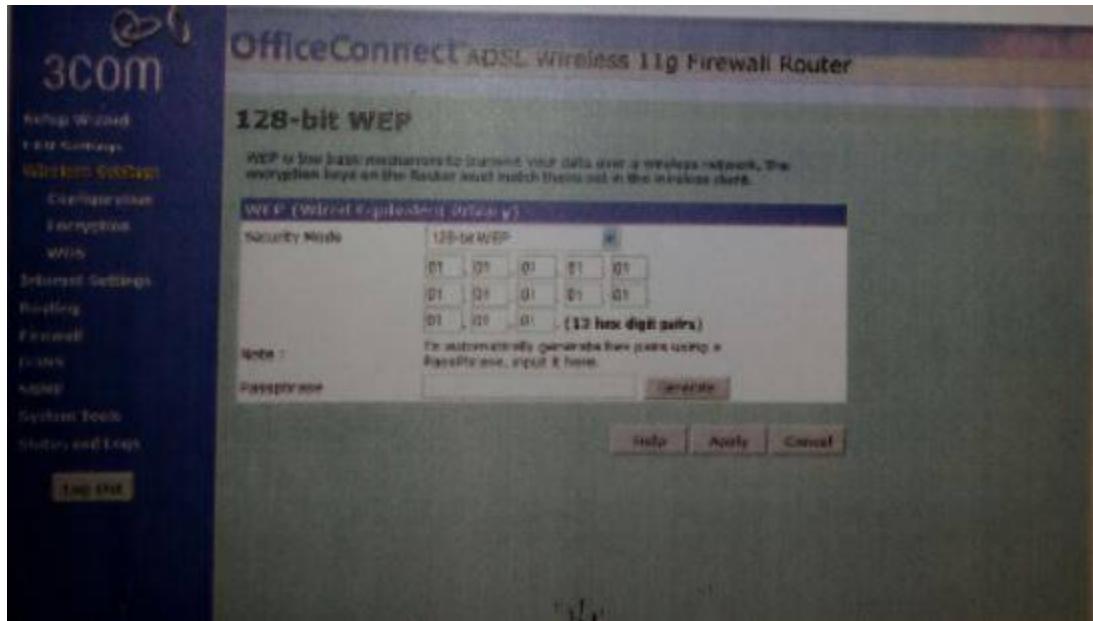


WEP is the basic mechanism transmits your data securely over the wireless network. Matching encryption keys must be set up on your Router and wireless client devices to use WEP.

A- You can either enter your WEP key as 13 pairs of hex digits(0-9,AF) , or enter a memorable pass phrase in the pass phrase box and then click Generate to Generate the hex keys from the pass phrase. The WEP keys on each device in the wirelessnetwork must be identical. In 128-bit WEP mode , only one WEP key (key 1) can be specified .

B- Clicks Apply.

64-bit WEP :

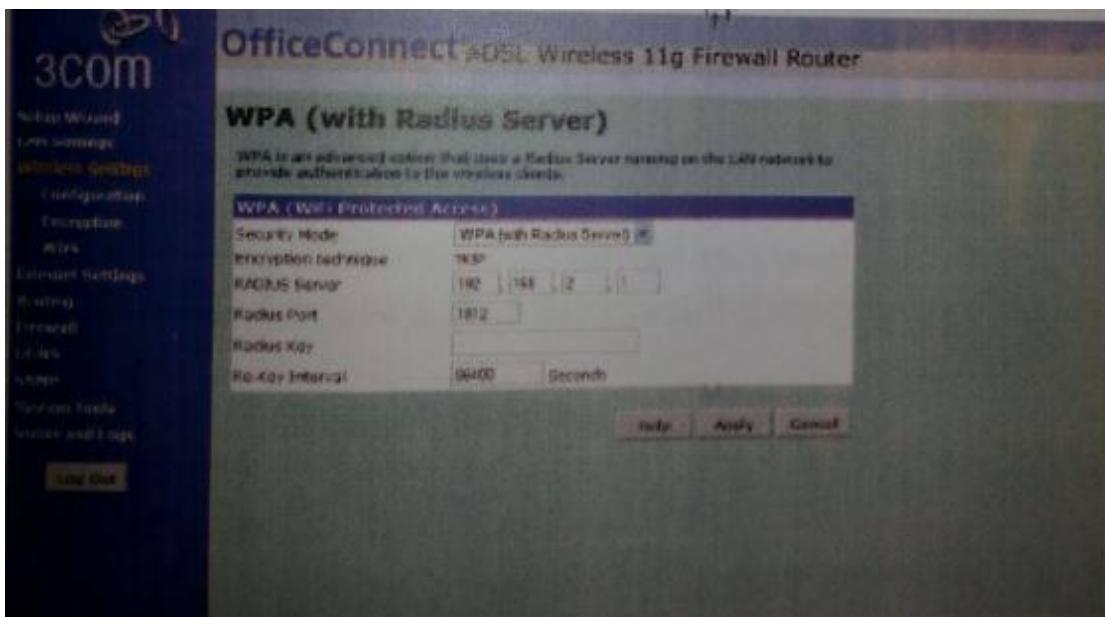


WEP is the basic mechanism to transmit your data securely over the wireless network. Matching Encryption keys must be set upon Router and wireless client devices to use WEP.

A – You can either enter your WEP key as 5 pairs of hex digits(0-9 , A-F), or enter a memorable pass phrase in the pass phrase box and then click Generate to Generate the hex keys from the pass phrase. For 64-bit WEP you can enter up to four keys , in the fields key1 to key4. The radio button in the left hand side selects the key that is used in transmission data.

B – Click Apply

WPA (with RADIUS Server) :



WPA(WiFi Protected Access) provides dynamic key changes and constitutes the best security solution. In a wireless network where not all devices support WPA, WEP (Wired Equivalent Privacy) should be used.

Wireless protected access using a server to distribute keys to the clients : this option requires that a Radius server is running on the network.

A- Enter the IP address of the RADIUS server on your network into the RADIUS server field.

B - Enter the port that the RADIUS server is operating on the RADIUS port field.

C- Enter the key of the RADIUS server in the RADIUS key field.

D- By default, the WPA keys are changed every hour, but if you wish to change this you can do so by specifying the required time in the Re-key interval field , in minutes .

E- Click Apply.

Wireless WDS settings:



The Router supports WDS (Wireless Distribution System) . WDS enables one or more access points to rebroadcast received signals to extend range and reach , though this can affect the overall throughput of data.

7.11.2 :To enable wireless repeating :

- A- Check the enable WDS check box.
- B- Enter the MAC address of one or more access points in the AP MAC Address table .

C- Clicks Apply.

To refresh the list of available access points , clicks Rescan Wireless Networking.

7.12 :Assign Or Unassigned an IPSec Policy on a computer :

To add an IP security Management Console and then assign or unassigned an IPSec policy for a local computer policy :

- A**-log on with a non-administrative user account.
- B**- Click Start, Click Run, type MMC, and then click ok.
- C**- In MMC , Click File, Click Add/Remove snap-in, and then click Add.
- D**- Click IP security policy management, and then click Add.
- E**-On the select computer or Domain page , verify that local computer is selected , and then click Finish.
- F**- In the Add Standalone Snap-in dialog box , click Close , and then click Ok.
- G**- Save the console as IPSec to the desktop, and then close the IPSec console.
- H**- On the desktop, right-click IPSec, and then select Run as.
- I**- In the Run as dialog box, select the option Theof and then type a user account and password that has the appropriate permissions to complete the task, and click Ok.
- J**- In the console tree, click IP security policies on local computer.
- K**- In the Details pane , click the IPSec policy that you want to assign or unassigned, and then do one of the following:

To assign the policy, in the details pane, right-click the appropriate policy , and then click Assign.

To unassigned the policy, in the details pane, right-click the appropriate policy , and then click Unassigned.

To assign or unassigned the an IPSec policy for an active directory-based group policy :

- A**-Open active directory users and computers.
- B**- In the console tree, right-click the domain or organizational unit for which you want to set Group Policy.

- C-** Click Properties , and then click the Group Policy tab.
- D-** Click Edit to open the Group Policy object that you want to edit . Or click New to create a new Group policy object, and then click Edit.
- E-** In the Group Policy Object Editor console, expand computer configuration , expand Windows Settings , expand Security Settings , and then click IP Security Policies on Active Directory.

7.12.1 :Working with Active Director :

To create a new Organizational Unit:

- A-** Open Active Directory Users and Computers.
- B-** In the console tree , double-click the domain node.
- C-** Right-click the domain node or the folder in which want to add the organizational unit, point to new, and then click Organizational Unit.

7.12.2 :Creating a domain user accounts :

- a-** From the windows start menu ,select administrative tools.
- b-** Clicks on active directory users and computers.
- c-** In the console tree, double-click the domain node.
- d-** In the details pane, right-click the organizational unit where want to add the user, point to New, and then click User .
- e-** In the New Object – User dialog box , in the first name box, type the user first name.
- f-** In the initials box, type the user initials.
- g-** In the Last name box, type the user last name.

- h-** In the user logon name box, type the name that the user will log on with.
- i-** From the drop-down list, click the UPN suffix that must be appended to the user logon name after the sign (@).
- j-** Click Next.
- k-** In the Password and Confirm Password boxes, type the user password.
- l-** Select the appropriate password options.

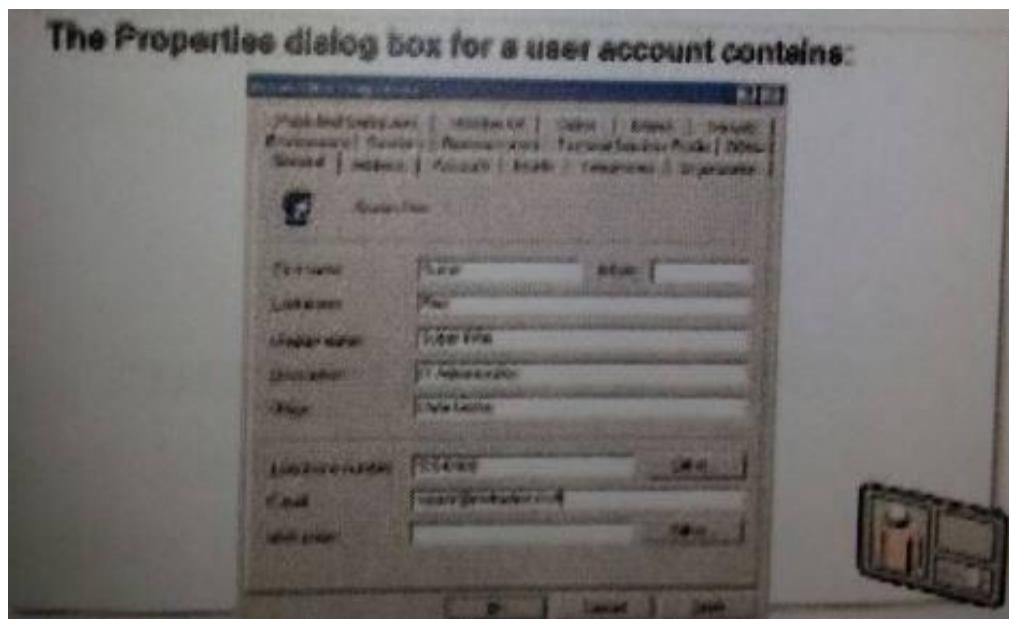
Account options	Description
User must change password at next logon	A user must change their password the next time they log on to the network
User cannot change password	A user does not have the permissions to change their own password
Password never expires	A user password is prevented from expiring
Account is disabled	A user cannot log on by using the selected account

- m-**Click Next, and then click Finish.

7.12.3 :Modifying a domain user accounts :

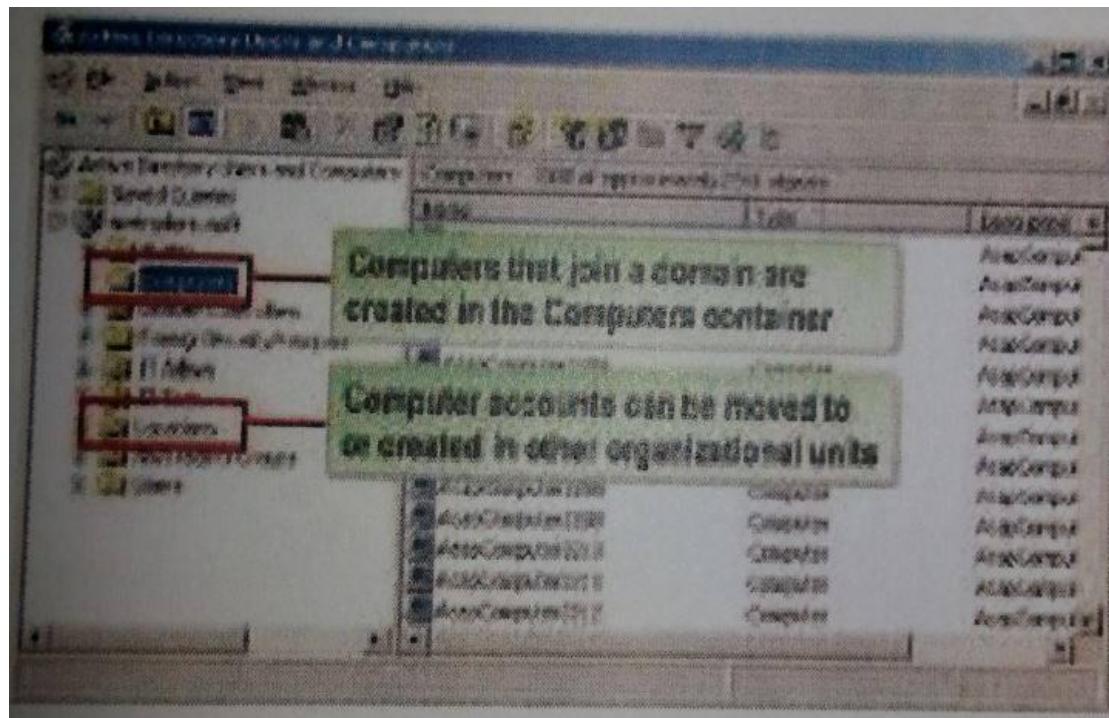
- A-**In Active Directory Users and Computers, in the console tree, navigate to the container that contains the user account that wants to modify.
- B-**In the details pane, select the user account that want to modify, right-click the selection, and then click properties.
- C-**In the properties dialog box, modify the properties of the account as Necessary.

The Properties dialog box for a user account contains:

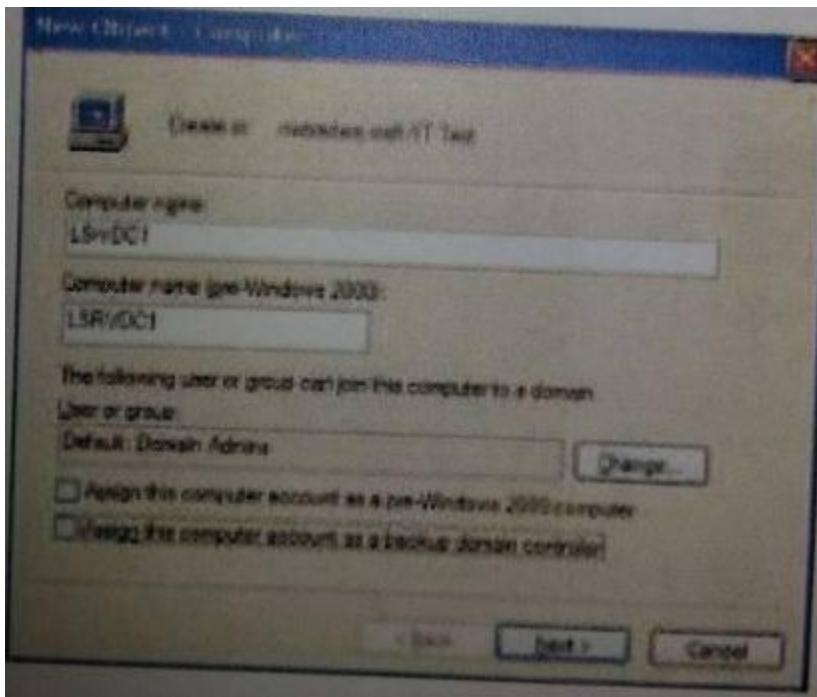


7.12.4 : Creating a computer account :

- A-From the Windows *Start menu*, select administrative tools.
- B-Clicks on active directory users and computers.
- C-In the console tree, double-click the domain node.



D-In the New Object – User dialog box , in the first name box, type the user first name.



A-Select the appropriate options, and then click Next.

B-In the managed dialog box, click Next.

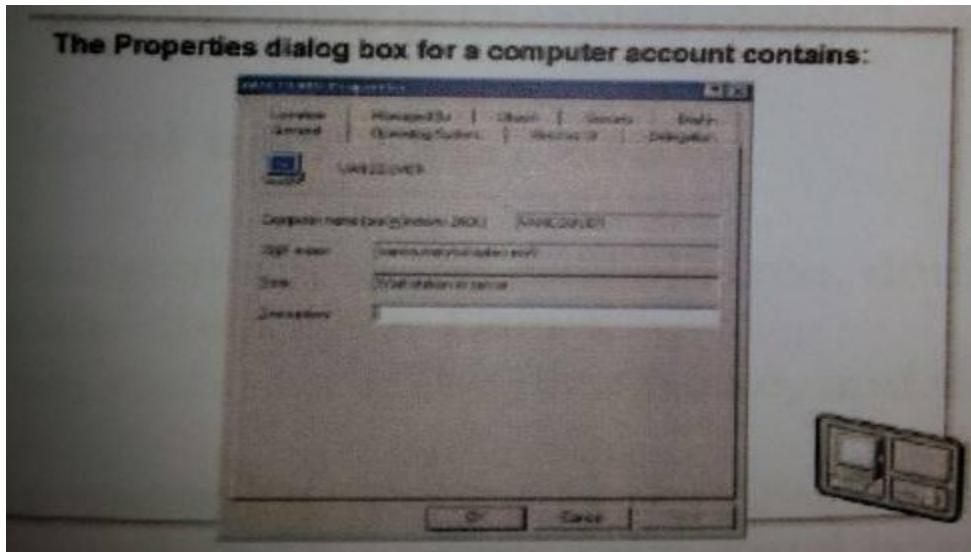
C-Click Finish.

7.12.5 : Modifying a computer accounts :

A-In Active Directory Users and Computers, in the console tree, navigate to the container that contains the computer account that wants to modify.

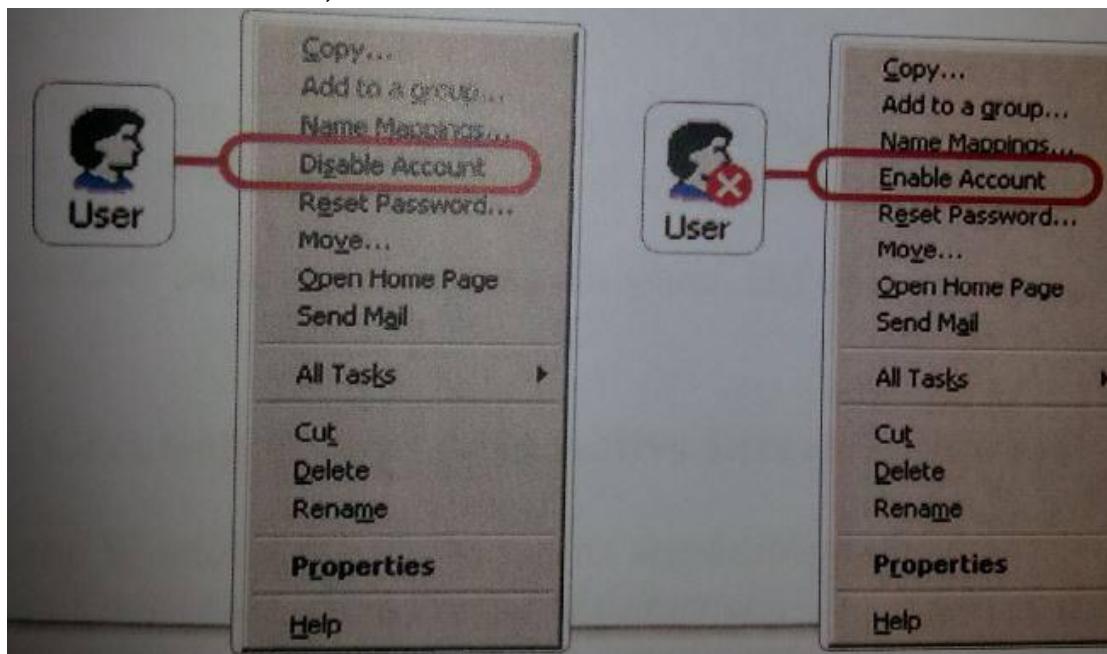
B-In the details pane, select the computer account that want to modify, right-click the selection, and then click properties.

C-In the properties dialog box, modify the properties of the account as Necessary.



7.12.6 :To enable and disable a user or computer account by using Active Directory Users and Computers :

- A-In Active Directory Users and Computers, in the console tree, select to the container or the user that contains the account to be enabled or disabled.
- B-In the details pane, right-click the user account.
- C-To disable, click Disable Account.
- D- To enable, click Enable Account.



7.12.7 :To disable or enable a local user account by using Computer Management :

- A-In computer management, expand System Tools.**
- B-In System Tools, expand Local Users and Groups, and then click Users.**
- C-Right-click the user accounts, and then click Properties.**
- D- In the Properties dialog box, to disable, select the Account is disabled check box, and then click OK.**
- E- To enable, clear the Account is Disabled check box.**

7.12.8 :To reset local user password :

- A-In computer management, in the console tree, double-click Local Users and Groups and then click Users.**
- B-In the details pane, right-click the user name and then click Set Password.**
- C-Read the warning message, if want to continue click Proceed.**
- D- In the New Password and Confirm Password boxes,type the new password and then click OK.**

7.12.9 :To reset domain user passwords :

- A-In Active Directory Users and Computers, in the console tree, click Users.**
- B-In the details pane, right-click the user name, and then click Reset Password.**
- C-In the New Password and Confirm Password boxes, type a new password and then click OK.**

7.12.10 : To reset computer accounts :

- A-In Active Directory Users and Computers, in the console tree, click Computers or the container that contains the computer that wants to reset.**
- B-In the details pane, right-click the user name, and then click Reset Account.**

7.12.11 :Creating groups in an Active Directory domain :

- A-From the Windows Start menu, select administrative tools.**
- B-Clicks on active directory users and computers.**
- C-In the console tree, double-click the domain node.**
- D- In the New Object . Group dialog box, in the group name box, type the name of the new group.**
- E- Under Group scope, click the group scope for the new group.**
- F- Under Group type, click the group type for the new group.**

7.12.12 :To add members to or remove members from a group :

- A-In Active Directory Users and Computers, in the console tree, click the folder that contains the group to which want to add a member.**
- B-In the details pane, right-click the group, and then click Properties.**
- C-In the Properties dialog box, on the Member stab, click Add.If want to remove a member from the group, click the members, and then click Remove.**
- D- In the select Users, Contact, Computers, and Groups dialog box in the Enter the object names to select box, type the name of the user, group, or computer that want to add to the group, and then click OK .**

7.12.13 : Modifying groups :

- A-In Active Directory Users and Computers, in the console tree, click the folder that contains the group.**
- B-In the details pane, right-click the group, and then click Properties.**
- C-In the Properties dialog box, on the General tab, under Group type, click the group type to change it.**
- D- Under Group scope, click the group scope to change it.**

7.12.14 :To add permissions for an object :

- A-If advanced features is not already checked in Active Directory Users and Computers, on the View menu, click Advanced Features.**
- B-In the console tree, right-click the object, and then click Properties.**
- C-In the Properties dialog box, on the Security tab, click Add.**
- D- In the Select Users, Computers, or Groups dialog box, in the Name box, type the name of the user or group to which want to grant permissions, and then click OK.**

7.12.15 :To modify an existing permission :

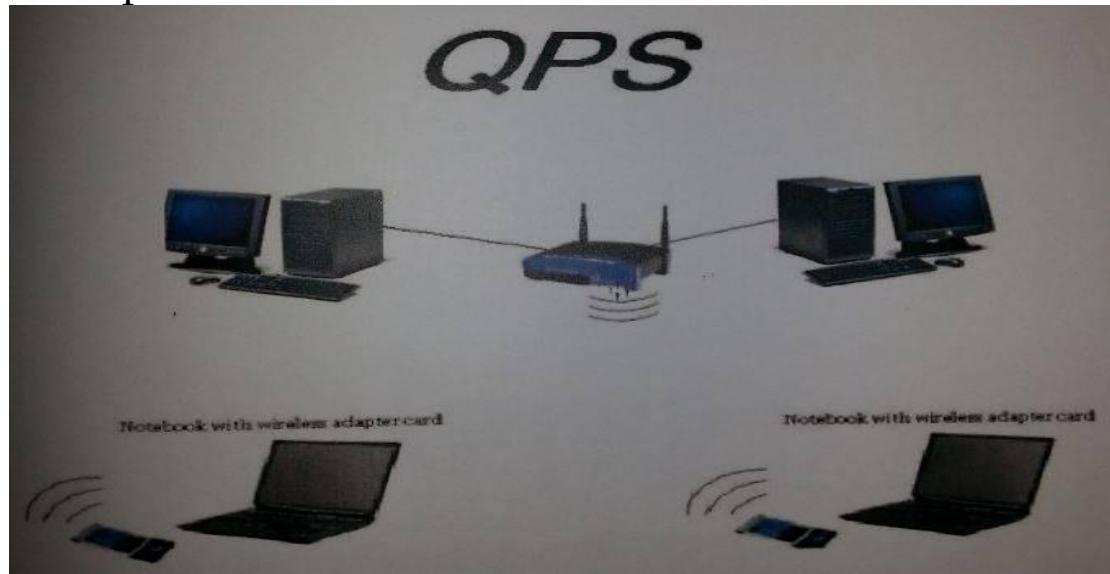
- A-If advanced features is not already checked in Active Directory Users and Computers, on the View menu, click Advanced Features.**
- B-In the console tree, right-click the object, and then click Properties.**
- C-In the Properties dialog box, on the Security tab, in the permissions box, select the Allow or Deny check box for each permission that want to allow or deny.**

7.13.16 : To view special Permissions :

- A-In** the Properties dialog box for the object, on the Security tab, click Advanced.
- B-In** the Advanced Security Settings dialog box, on the Permissions tab, click the entry that want to view, and then click Edit.
- C-In** the Permission Entry dialog box, on the Object tab, in the Apply onto box, select the option that want.

7.13 :Installing Oracle 9i data base server :

Loading oracle data base and using the whole system for the whole system to use for instructors and students is explained in chapter 3.



End of Chapter 7

Thanks

