

# **COMPUTERIZATION OF WAPDA UNIVERSITY TOWN II PESHAWAR**



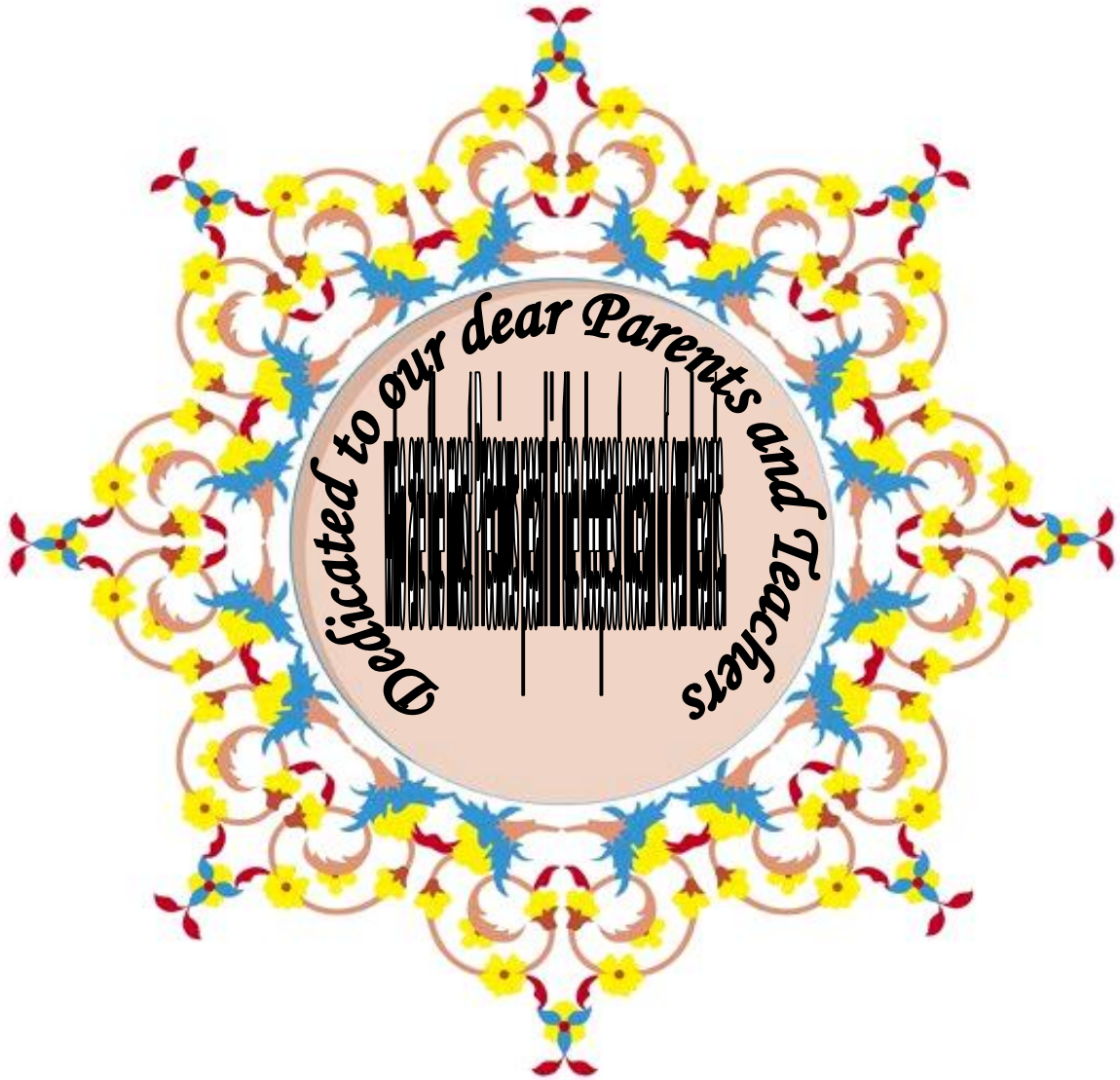
**By**  
**Hafiz Ur Rahman**  
**Muhammad Ishaq**  
**Islahuddin**

**Supervised By**  
**Dr. Sara Shahzad**

**Submitted to the Department of Computer Science**  
**University Of Peshawar (UoP)**  
**For the partial fulfillment of Bachelor of Computer Science (BCS)**  
**Degree.**

**Session 2008-2011**

# DEDICATION



## **PROJECT APPROVAL**

The committee certifies that this thesis forwarded by Mujtaba Hussain, Salman Jafer and Najeeb Ullah is up to the standard and we here by approve it for its acceptance as a partial fulfillment for the award of Bachelor of Computer Science degree.

### **Committee**

**External Examiner**

**Signature:** \_\_\_\_\_

**Remarks:** \_\_\_\_\_

**Internal Supervisor**

Mr. Ibrar Ahmad

Assistant Professor

Department of Computer Science

University of Peshawar

**Signature:** \_\_\_\_\_

**Remarks:** \_\_\_\_\_

**Chairman**

Dr. Saeed Mehfooz

Associate Professor

Department of Computer Science

University of Peshawar

**Signature:** \_\_\_\_\_

**Remarks:** \_\_\_\_\_

## *PREFACE*

We are living in an era of information technology. The fifth generation computers, Internet Technology, and the new development in hardware have changed the shape of the world, and a new world is coming to existence which is totally based on Information Technology. Soon the definition of an illiterate person will be “a person who is ignorant of computer”.

We, being the pioneers of that technology, introduced and expand it to every walk of life. But the revolution has not yet reached fully to the east, and although computer is widely used in the east but still there are certain fields where works is being carried out manually as in the prehistoric age.

Information technology, the fusion of computing and communication, is creating far-reaching changes in the way we work. Keeping in mind the emerging trends in the field of computer science we took the challenge of developing software, which we will submit as a partial fulfillment of our degree along a project report to fulfill our course requirement.

In this project attempt has made to develop such a system that would prove beneficial to an organization. We start our project to fulfill our course requirement but it will become learning experiences for all of us.

The goal and objective of our project is to computerize the WAPDA University Town II Peshawar, which uses the traditional system in which bulky registers are used for entry due to which it become headache for the clerk in case of searching a record to determine whether the person is a defaulter or not. It becomes very difficult task for the clerk to search the person in several registers.

The software is such that it will overcome the problems of searching and catching defaulters. This software’s main advantage is its user friendliness and will prove to be extremely easy for a person having preliminary computer operation knowledge to operate it properly.

**Mujtaba Hussain**  
**Salman Jafer**  
**Najeeb Ullah**

# **ACKNOWLEDGMENT**

We are grateful to Almighty Allah, Whose blessings have given us the ability to complete this task. We wish our heartiest thanks to our parents who encouraged us at every stage of life and remembered us in their prayers. We can never turn down their efforts they put forward for our bright and prosperous future. May Allah Almighty give them with affection and forgiveness in this life and afterward.

We can never forget the efforts of our teachers and very thankful to them who helped us during course of study and make our emotion strong for higher education. We are thankful to Mr. Ibrar Ahmad for his cooperation in the project.

**THANKS**

# PROJECT

**Project Title** : Inventory Management

**Organization** : WAPDA University Town II

**Development Team:** Hafiz ur Rahman

Muhammad Ishaq

Islah Uddin

# Table Of Contents

<b>Chapter 1</b>		
<b>01</b>	<b>Introduction</b>	
1.1	The Customer	01
1.2	The project	01
1.2.1	New Meter Information System	01
1.2.2	D & R Information System	01
1.3	Project Objective	02
1.4	Scope of the Project	02
1.5	Tools Used in Project	02
1.5.1	Oracle 10g	03
1.5.2	Developer 10g	03
<b>Chapter 2</b>		
<b>02</b>	<b>Customer Statement of Requirement</b>	
2.1	Customer Statement of Requirement	04
2.1.1	New Meter Section	04
2.1.2	Disconnection and Reconnection	05
<b>Chapter 3</b>		
<b>03</b>	<b>Requirement Engineering</b>	
3.1.	Functional Requirement Specification	06
3.2	WAPDA University Town II Functional Specification	06
3.2.1	Data to be Held	07
3.2.1.1	New Meter Section	07
3.2.1.2	Disconnection and Reconnection Section	08
3.2.2	Operational Scenario	08
3.3	Stakeholder	08
3.4.	Actor	09
3.5.	Use Cases	09
3.5.1.	Use Case Diagram	10
3.5.2	Element of Use Case Diagram	10
3.5.2.1	Actor	10
3.5.2.2	Use Case	11
3.5.2.3	Association	11
3.5.2.4	System Boundary Boxes	11
3.5.2.5	Packages	12
3.6	WAPDA University Town II Use Case Diagram	13

---

## Chapter 4

<b>04</b>	<b>Domain Analysis</b>	
4.1	Domain Analysis	16
4.2	Domain Model	16
4.2.1	Data Analysis and Modeling	16
4.2.1.1	Data Objects/Entities	17
4.2.1.2	Attributes	17
4.2.1.3	Relationship	17
4.2.2	DOD of New Meter Section	17
4.2.3	DOD of D & R Section	23
4.2.4	ERD	25
4.2.5	WAPDA University Town II ERD	26

## Chapter 5

<b>05</b>	<b>Data Design</b>	
5.1	Introduction	28
5.2	Backend	28
5.3	Table: Applicant Information	28
5.4	Table: Property Information	28
5.5	Table: Application Information	28
5.6	Table: DoubleConnection Information	29
5.7	Table: Service Estimate Information	29
5.8	Table: Demand Notice Information	29
5.9	Table: SCO Information	30
5.10	Table: Meter Description Information	30
5.11	Table: Account Information	30
5.12	Table: Install Remarks Information	31
5.13	Table: Meter Store Information	31
5.14	Table: MCO Information	31
5.15	Table: MCO Meter Remove Information	32
5.16	Table: MCO Meter Install Information	32
5.17	Table: Send Challan Information	32
5.18	Table: Receive Challan Information	32
5.19	Table: CP78 Information	33
5.20	Table: DCO Information	33
5.21	Table: ERO Information	33
5.22	Table: CP79 Information	33
5.23	Conclusion	34



## Chapter 6

<b>06</b>	<b>System Design</b>	
<b>6.1.</b>	System Interaction	34
<b>6.1.1</b>	Sequence Diagram	34
<b>6.1.2</b>	Collaboration Diagram	35
<b>6.2</b>	Class Diagram	36
<b>6.2.1</b>	Class Name	36
<b>6.2.2</b>	Class attribute list	37
<b>6.2.3</b>	Class operations list	37
<b>6.2.4</b>	Inheritance	37
<b>6.2.5</b>	Association	38
<b>6.2.5.1</b>	Bi-directional Association	38
<b>6.2.5.2</b>	Uni-directional Association	39
<b>6.2.6</b>	WAPDA University Town II Class Diagram	40
<b>6.3</b>	User Interface Design	41
<b>6.3.1</b>	Log In	42
<b>6.3.2</b>	Main Page	43
<b>6.3.3</b>	New Meter Section	44
<b>6.3.4</b>	Disconnection and Reconnection	45
<b>6.3.5</b>	Searching	46
<b>6.3.6</b>	Reports	47
<b>6.4</b>	Implementation	48
<b>6.4.1</b>	Implementation Strategy	48
<b>6.4.1.1</b>	Proposed System Requirement	48
<b>6.4.1.2</b>	Language Selection	48
<b>6.4.3</b>	System Implementation	49

## Chapter 7

<b>07.</b>	<b>History of Work</b>	
<b>7.1</b>	History of Work	51
<b>7.1.1</b>	Project Selection	51
<b>7.1.2</b>	Requirement Gathering	51
<b>7.1.3</b>	Data Analysis	51
<b>7.1.4</b>	Data Modeling	51
<b>7.1.5</b>	Interface Design	52
<b>7.1.6</b>	Coding	52
<b>7.2</b>	Key Accomplishments	52

## Chapter 8

<b>08</b>	<b>Future Works</b>	
<b>8.1</b>	Future Work	53
<b>8.2</b>	References	54

**Chapter 9**

**09**

Glossary

55

# **CHAPTER 1**

## **INTRODUCTION**

## INTRODUCTION

### 1.1 The Customer

The Water And Power Development Authority (WAPDA) is a government-owned organization of Pakistan maintaining many Dams, including Tarbela, Mangla, Warsak, Monda, Shangla etc dams among its resources. Its headquarter is in Lahore. It was established in February 1959.

WAPDA has various electric supply companies in which one of them is PESCO which distributes electricity in all districts of Khabar Pukhtoonkhwa including Peshawar, Mardan, Kohat etc.

Peshawar District has many Electric Grid Stations in which one of them is University Town II Grid station located on the University Road near Gora Qabristan. It provides electricity to all residential of University Town II. It is managed under a qualified SDO and other devoted staff.

### 1.2 The Project

There are different sections of the office. Details of one's concerned with our project are given below

#### 1.2.1 New Meter Information System

The New Meter section handles all the records of customers who apply for the issuance of meters. This section handles the application information and also Applicant information, Demand Notice, Service Connection Order, Meter Description.

#### 1.2.2 D & R information System

This section handles the information of customers who want to remove the meter or want to change the meter and as well as those customers who want reconnection. This section also handle those customers whose meters or connection has been damaged due to some disaster.

### 1.3 Project Objective

The objectives of the proposed system are as follows:

1. The new system should be more efficient
2. It should provide complete security to the users and the administration
3. The system should be reliable
4. The system should be user friendly
5. The system should be flexible enough to accommodate any type of change in future such as to update, refresh system
6. It should be complete and replace the manual system maximum data integrity, correctness and avoids data duplication
7. The system can replace paper work and space needed to save record to some extent
8. Data maintenance in terms of accuracy should be excellent
9. Customers data entry should be efficient
10. It should be design as long-life system

### 1.4 Scope of the Project

The system as to be designed for the University Town II Grid Station. So it must fulfill the requirements of this organization which mainly include:

1. Admin can perform Data Modification Language on customer records
2. Admin can perform Data Definition Language on customer records

### 1.5 Tools Used in Project

Following tools used in the project

1. Oracle10g for database development
2. Developer 10g for interface design

#### 1.5.1 Oracle 10g

As our project is strongly based on database, so we choose oracle10g for database design which provides following facilities:

1. It provides maximum security of the data
2. Each record inserted is consistently

3. Data updating / deletion and addition is easier
4. Easy and efficient data processing
5. Efficient data retrieval
6. security in distributed environment
8. There is much less chance of data corruption
9. Data accuracy is 100%

### **1.5.2 Developer 10g**

According to the requirement of the project we need a User-friendly interface which can be used easily by end-user.

Following facilities are provided by Developer 10g

1. It provides easy connectivity with database
2. Data drag and drop object make code and UI design more efficient
3. It includes a lot of build-in tools, triggers and controls application development
4. Form wizard provides easy way for building and designing data entity and built-in steps which include required text items to enter and view data
5. Developer 10g provides facility of making procedure such as scroll, item\_enable\_disable, menu scroll procedure to perform task effectively and accurately
6. The built-in libraries function in Developer 10g provides the facility to reuse same code at many places with no changes
7. User has to write code for one time, saved in library and attach this library at many places to perform task
8. The property class function can be used to set visual attributes
9. Oracle 10g is web based, so it is easy to connect from remote area through only IP address
10. It's runtime environment is Microsoft internet explorer

# **CHAPTER 2**

## **CUSTOMER STATEMENT OF REQUIREMENTS**

## 2.1 Customer Statement of Requirement

The customer statement of requirement is a document written at the starting of the new project to define the project requirements. It is also a document which is referred to constantly throughout the projects to act as a benchmark for scoping the project. In other words, if there are changes to the SOR, a new document must be drawn up to accommodate these new requirements and there will be a new negotiated price for this as a result of the additional scope of work.

The WAPDA of Pakistan is running in loss for many years due to illegal use of electricity and non-payment of bills by the people.

WAPDA usually uninstalls electricity meters from defaulters. Due to some loop holes in the existing systems. These defaulters easily get new meter without payment of dues. There is no proper check system in grid stations which can catch these defaulters. In these grid stations, one of them is University Town II Peshawar. In University Town II, the people also using electricity illegally which become headache for the University Town II grid station.

The SDO of University Town II grid station desired for such a system to have, which can catch these defaulters and thieves of the electricity and reduce the losses of WAPDA therefore he thought it appropriately to computerize the current traditional system into computerized system which also has the facility to not letting the customer to install more than one meter to same property.

To catch these defaulters it is necessary to computerize the following two sections of University Town II.

### 2.1.1 New Meter Section

The New meter Section maintains the records of new meters. If the customer wants to apply for new meter he/she should consult this section. This section allot a meter to customer after fulfilling the following formalities.



First the customer gets application for new meter and fill it out. The application contains the Customer name, CNIC, Father name, Property number, Property address, Customer address, Connection type and Connection load etc.

After filling the application form the new meter section provides service estimation which show the customer that how much the customer will pay for the installation of new meter.

The demand notice is issued to customer on the basis of which the customer will pay the lumsumcost and the security.

After the issuance of the demand notice the service connection order is issued to the customer.

Account is created on the basis of which the new meter is allotted to customer and the line superintendent install the new meter for customer.

### **2.1.2 Disconnection And Reconnection Section (D&R)**

In this section we handle the information of customers who want to remove the meter or want to change the meter and as well as those customers who want reconnection. This section also handle those customers whose meters or connection has been damaged or burnt due to natural disasters.

# **CHAPTER 3**

## **REQUIREMENT ENGINEERING**

### 3.1 Functional Requirement Specification

The Functional Requirement Specification document also called Functional requirement or Functional specification, which defines the capabilities and functions that a system must be able to perform successfully.

Functional Requirements should include:

- Descriptions of data to be entered into the system
- Descriptions of operations performed by each screen
- Descriptions of work-flows performed by the system
- Descriptions of system reports or other outputs
- Who can enter the data into the system.
- How the system meets applicable regulatory requirements

The functional specification is designed to be read by a general audience. Readers should understand the system, but no particular technical knowledge should be required to understand the document.

### 3.2 WAPDA University Town II Functional Specification

The new proposed system is computerized to overcome the existing System's drawbacks. In proposed system electronic data processing methods are used to make system more efficient, more economical and error free.

The proposed system will perform the following functions:

1. The new system should less time consuming and more efficient
2. It should provide maximum flexibility to handle any type of changes
3. It should eliminate paper work to some extend
4. It should provide maximum security to user
5. The data stored should be consistent and error free
6. The system should be user friendly and reliable
7. It requires less resource to develop and update
8. Repetition of data should be eliminated

9. Data should be retrieved and update easily
10. Certain checks are applied to avoid errors and mistakes

### **3.2.1 Data to be Held**

The proposed system of WAPDA University Town II consists of two sections.

#### **a. New Meter Section**

The New Meter section handles all the record of customers who apply for issuing new meters as well as the description of new meters. This section handles the application information, Applicant information, Demand Notice, Service Connection Order, Meter Description, Service Estimation, Account, Property Information and Installation Remarks etc

1. Applicant info table which contains the Applicant Name, Applicant NIC, Applicant Father Name, Applicant Address etc
2. Application info table which contains the Application Number, Date of Application, NewExtChange field which determines whether the applicant change the meter or apply for new meter or for extension of meter etc. This also contain a field which act as a reference and refer to the applicantinfo table.
3. Property Info table contains the Property Number, Property Address and Substation concern field which shows that this property is under the substation e.g University Town II substation.
4. Service Estimate table contains the Service Estimate No, Date issued, Application No as a referential field is used to refer that for that application no this service estimation is done and Cost Estimate field is used to show that how much cost, the customer will pay for this connection.
5. Demand Notice Payment table contains the Demand Notice No, LumsumCost, Security, Slip No and Service Estimate No field is used as a referential field.
6. Service Connection Order table contains the SCO No, Connection type either commercial or local etc
7. Account table contains the Demand Notice No as a referential field, Account No, Meter No and Service Connection Order No etc.

8. Meter Description table is used to enter all the information relating to meter. This table contains Meter No, Meter Company, Initial Reading, Meter type etc.
9. Installation Remarks table contains SDO sign and Name, Line Superintendent name and signature and Date of installation etc.

#### **b. Disconnection And Reconnection (D&R) Section**

In this section we handle the information of customers who want to remove the meter or want to change the meter and as well as those customers who want reconnection. This section also handle those customers whose meters or connection has been damaged or burnt due to natural disasters.

1. MCO table contains MCO No, MCO Date, Account No, Batch No and Reason Change field is used to determine why the meter is change etc
2. MCOmeterRemove table contains MCO No, Multiplying Factor, Advance Unit, Final Reading etc
3. MCOmeterinstall table contains MCO No, Loc Code, Meter Code etc
4. DCO table contains DcoNo ,DcoDate etc
5. RCO table contains RcoNo, RcoDate etc

#### **3.2.2 Operational Scenario**

The system can be used in central command environment where a single person can use the system and work as administrator .Administrator can perform the following action.

1. Admin can perform DML on all customer Records
2. Admin can perform DDL on all customer Records

The DBA of WAPDA University Town II will assign a user or clerk each section with certain privileges. New meter section will be assigned to one user and the Disconnection And Reconnection (D&R) section to other user with privileges i.e is selection or insertion only and will not let these user to update or modify etc the customer records.

### 3.3 Stakeholders

A person, group or organization, that has direct or indirect share in an organization because it can affect or be affected by the organization's actions, objectives and policies. Key stakeholders in WAPDA University Town II are:

1. WAPDA University Town II **SDO**, who is the head of that grid station and wish to reduce the losses and catch the defaulters easily.
2. WAPDA University Town II New Meter Section **Clerk** or **User**, who handles the new meter section and insert the records of those customers who apply for new meter. He is interested in this to get rid of bulky registers.
3. WAPDA University Town II Disconnection And Reconnection (D&R) section **Clerk** or **User**, who is responsible for keeping track of old meter and as well as new meters. He is also interested to get rid of bulky registers and easily search a record when he is asked.

### 3.4 Actors

Actors are parties outside the system that interact with the system. An actor may be a class of users, roles users can play, or other system distinguishes between primary and secondary actors. A primary actor is one having a goal requiring the assistance of the system. A secondary actor is one from which the system needs assistance. The primary actor in WAPDA University Town II is the **SDO** who has goal to catch the defaulters and search a record easily so the SDO needs the assistance of such a system which help in achieving the goal of the SDO while as the secondary actor in WAPDA University Town II is the **Clerk** whose responsibility is to insert and maintains the customer records which means that the clerk helps the system for maintaining and achieving the goal of the primary actor. The other actor is the **Customer** who applies for the new connection or change the old connection.

### 3.5 Use Cases

A use case defines a goal-oriented set of interactions between external actors and the system under consideration. A use case is initiated by a user with a particular goal in mind and completes successfully when that goal is satisfied. It describes the sequence of

interactions between actors and the system necessary to deliver the service that satisfies the goal. It also includes possible variants of this sequence, e.g. alternative sequences that may also satisfy the goal, as well as sequences that may lead to failure to complete the service because of exceptional behavior, error handling etc. The system is treated as a “black box”, and the interactions with system including system responses, are as perceived from outside the system. Thus, use cases capture who (actor) does what (interaction) with the system, for what purpose, without dealing with system internals. A complete set of use cases specifies all the different ways to use the system, and therefore defines all behavior required of the system, bounding the scope of the system.

### 3.5.1 Use Case Diagram

Use case diagrams overview the usage requirements for a system. They are useful for presentations to management and/or project stakeholders, but for actual development you will find that use cases provide significantly more value because they describe "the meat" of the actual requirements.

### 3.5.2 Elements Of Use Case Diagram

#### 3.5.2.1 Actor

An actor is a person, organization, or external system that plays a role in one or more interactions with your system. Actors are denoted as stick figures.

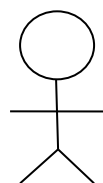


Figure 3.5.2.1 Actor

#### 3.5.2.2 Use Case

A use case describes a sequence of actions that provide something of measurable value to an actor and is denoted as a horizontal ellipse.



Figure 3.5.2.2 Use Case

### 3.5.2.3 Association

Associations between actors and use cases are indicated in use case diagrams by solid lines. An association exists whenever an actor is involved with an interaction described by a use case. Associations are modeled as lines connecting use cases and actors to one another, with an optional arrowhead on one end of the line. The arrowhead is often used to indicating the direction of the initial invocation of the relationship or to indicate the primary actor within the use case. The arrowheads are typically confused with data flow and as a result I avoid their use.



Figure 3.5.2.3 Association

### 3.5.2.4 System Boundary Boxes (Optional)

You can draw a rectangle around the use cases, called the system boundary box, to indicate the scope of your system. Anything within the box represents functionality that is in scope and anything outside the box is not.

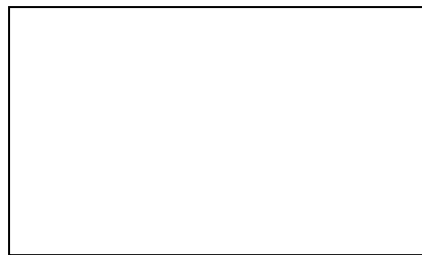


Figure 3.5.2.4 System Boundary Box



### 3.5.2.5 Packages (Optional)

Packages are UML constructs that enable you to organize model elements (such as use cases) into groups. Packages are depicted as file folders and can be used on any of the UML diagrams, including both use case diagrams and class diagrams

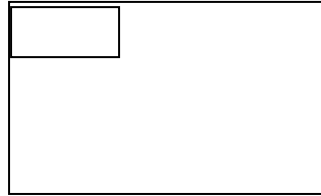
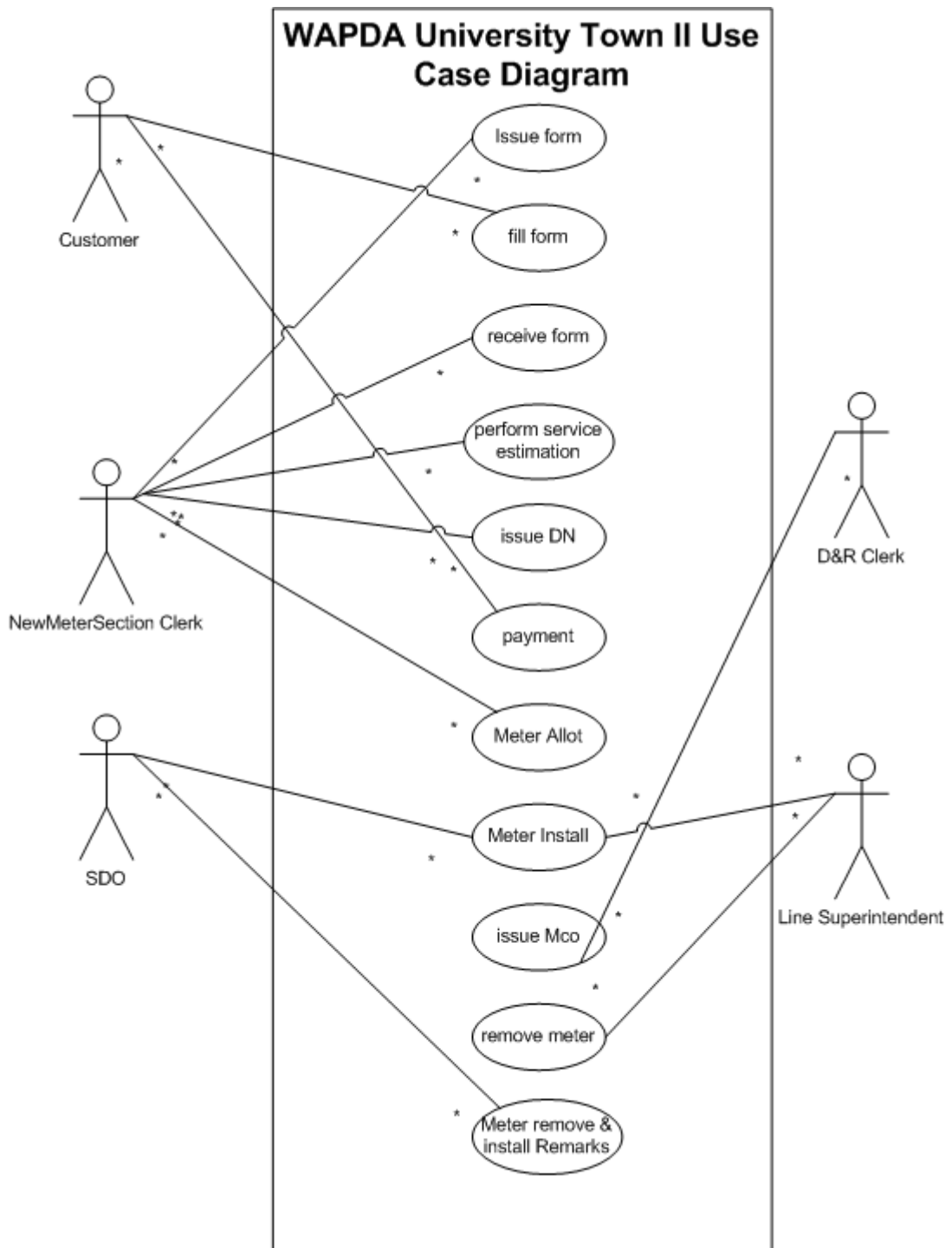


Figure 3.5.2.5 Package

3.6 WAPDA University Town II Use Case Diagram



### 3.7 Non Functional Requirement

In systems engineering and requirements engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. Non-functional requirements are often called qualities of a system. Other terms for non-functional requirements are "constraints", "quality attributes", "quality goals", "quality of service requirements" and "non-behavioral requirements". Qualities, that is non-functional requirements. Non-functional requirements include constraints and qualities. Qualities are properties or characteristics of the system that its stakeholders care about and hence will affect their degree of satisfaction with the system. Constraints are not subject to negotiation and, unlike qualities, are off-limits during design trade-offs. Contextual constraints are characteristics of the “super-system” (i.e., the larger system into which the system under development will fit) or the development organization that constrain the development in some way. Examples include the target operating system or hardware platform in the case of the user environment, or the skill-set of available developers in the case of the development organization.

### 3.8 Sources of Non-Functional Requirements

Run-time non-functional requirements arise from the operating environment, the user(s), and competitive products:

#### 3.8.1 System Constraints

Here one is looking for elements of the environment into which the system must fit, that may serve as constraints on the system. This may be true of the installed infrastructure (e.g., hardware and OS platforms) or legacy applications, or may be in the form of organizational factors or the process that the system will support.

#### 3.8.2 User Objectives, Values and Concerns

In establishing the run-time qualities for a system, it is important to identify all the categories of user (including other systems) that will interact with the system, and understand what quality attributes they care about. A quality attribute such as performance may surface for one user as a concern, and another as a value, so it is useful to direct elicitation of both values and concerns for any (group of) user(s). It is important to focus on creating *just* what users want, with the qualities they care about—low priority

whiz-bang features or qualities only increase complexity for the development organization and/or for the user. The requirements team should nonetheless be alert to requirements that users take for granted or are not able to articulate directly. Understanding the users' objectives and forces that impact their success and sense of utility, will help surface and establish the priorities of system qualities—as well as functionality, of course. In addition to discovering what qualities are important to users at the system level, qualities associated with particular functionality/user goals should be elicited. The qualities may need to be translated by developers from user-level objectives, values and concerns into specific technical quality requirements. For example, a user's requirement not to be impeded by slow system performance in conducting a task may be translated into requirements on transaction throughput and network latency.

### **3.8.3 Competitive Analysis of Features**

Run-time qualities are often associated with product features. Features are generally thought of as the characteristics of the product that establish its competitiveness, frequently distinguishing the product functions (base-line and unique product differentiators) with at least one quality attribute. For example, many web-based catalog services have on-line payment options. To allay market concern, the electronic payment feature includes transaction security as an essential attribute.

## **3.9 Tools for Capturing and Documenting Non-Functional Requirements**

Use cases have been widely used to specify functional requirements. By simply extending use cases with a field for all the non-functional requirements associated with the use case, run-time qualities associated with particular functionality can be captured conveniently

Since development-time requirements such as extensibility or reuse generally relate to future run-time functionality, they could theoretically be captured using use cases too. However, exhaustively exploring requirements on future releases or product variants are out of the question (the first product would be delayed indefinitely), so development-time requirements need to be treated differently. Because so much, is unknown and unpredictable about the future run-time environment and user needs, “what if” scenarios are used to explore the system's robustness to future requirements and the ease of adapting it to altering technologies and user goals.

# **CHAPTER 4**

# **DOMAIN ANALYSIS**

## 4.1 Domain Analysis

Domain analysis can be conceived of as an activity occurring prior to systems analysis and whose output (i.e A Domain Model) supports system analysis in the same way that systems analysis output (i.e Requirements analysis and specification document) supports the system's designer tasks. In the conventional waterfall model of software development, a systems designer's task is to produce a particular design from a set up requirements and specifications. A systems analyst's task, in the same model, is to create a model of in existing system and proposed alternative for automation or improvement. Both activities focus on a specific model for a particular system. In domain analysis in contrast we try to generalize all system in an application domain by means of a domain model that transcends a specific application. Domain analysis is thus at a higher level of abstraction than the system analysis. In domain analysis, common characteristics from similar systems or generalized, objects and operations common to all systems within the same domain or identified, and models are defined to describe their relationships.

If we succeed in identifying the objects and operations in a domain, the next step is to define a domain specific language. This language becomes our domain model and is used to describe objects and operations common to that domain.

## 4.2 Domain Model

A domain model in problem solving and software engineering can be thought of as a conceptual model of a domain of interest (often referred to as a problem domain) which describes the various entities, their attributes, roles and relationships, plus the constraints that govern the integrity of the model elements comprising that problem domain.

### 4.2.1 DATA ANALYSIS AND MODELING

It is the first activity in the analysis model. The following things are determined precisely in data analysis and modeling.

1. Data objects/entities
2. Attributes of data objects
3. Relationships between data objects

ERD is created from these three discussed above which is the graphical Representation of data analysis and modeling.

**4.2.1.1 DATA OBJECTS/ENTITIES**

Anything which have some properties (attributes) is called data object .The DOD incorporates the data object and all of its attributes.

**4.2.1.2 ATTRIBUTES**

Attributes are the properties of an object e.g., name of a candidate is an attributes of a candidate.

**4.2.1.3 RELATIONSHIP**

Relationship is a logical association between entity types. There are required for relationship.

- a. Degree of Relationship
- b. Cardinality of a Relationship
- c. Modality of Relationship

**i. Degree of Relationship**

Degree shows that how many entity are participating in a relationship

**ii. Cardinality of Relationship**

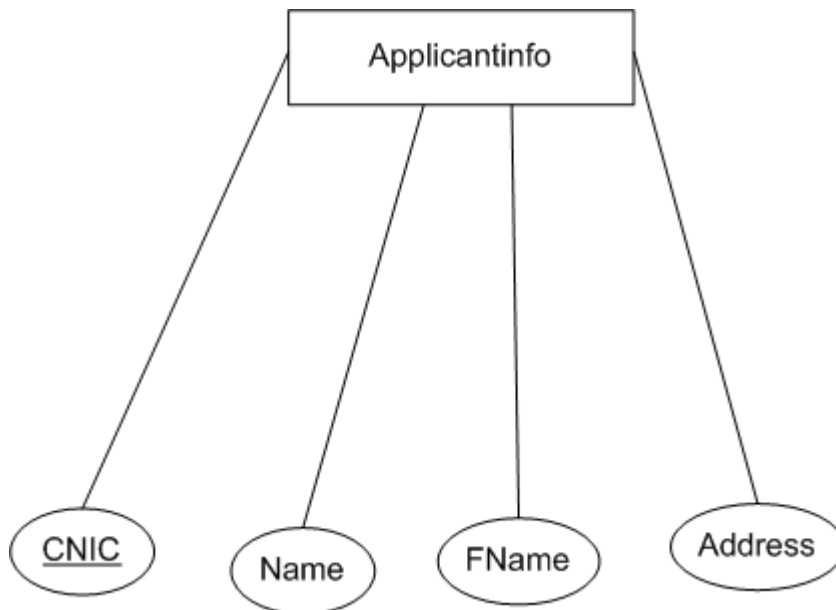
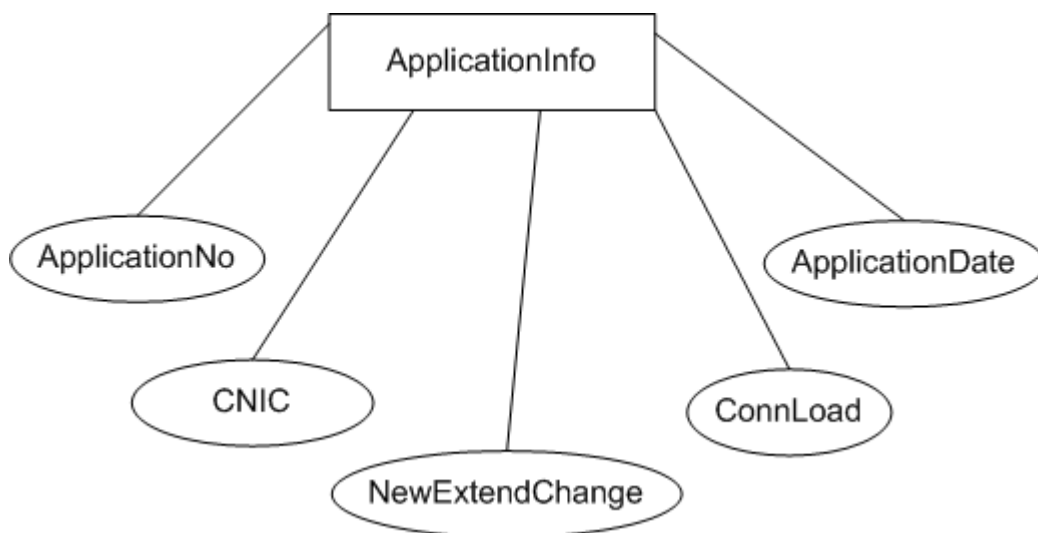
Cardinality of a relationship shows that how many instances of one entity related to how many instances of other entity.

**iii. Modularity Relationship**

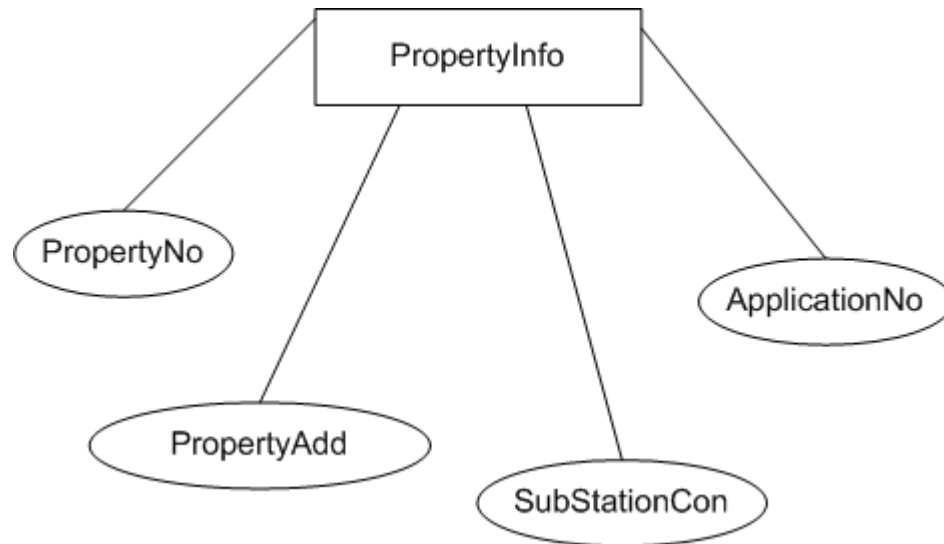
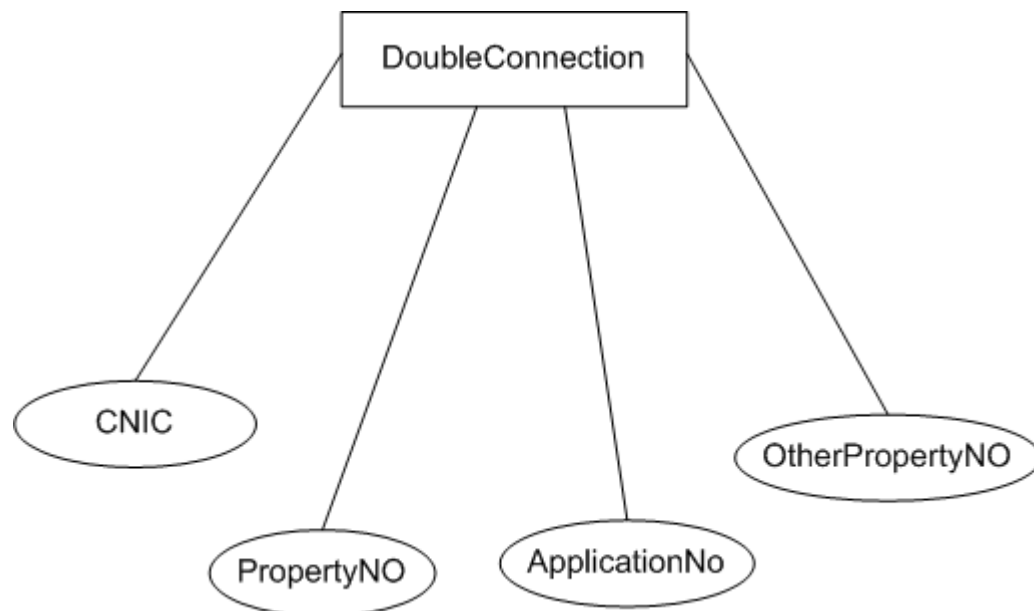
Modality of relationship shows that either the relationship is mandatory or optional.

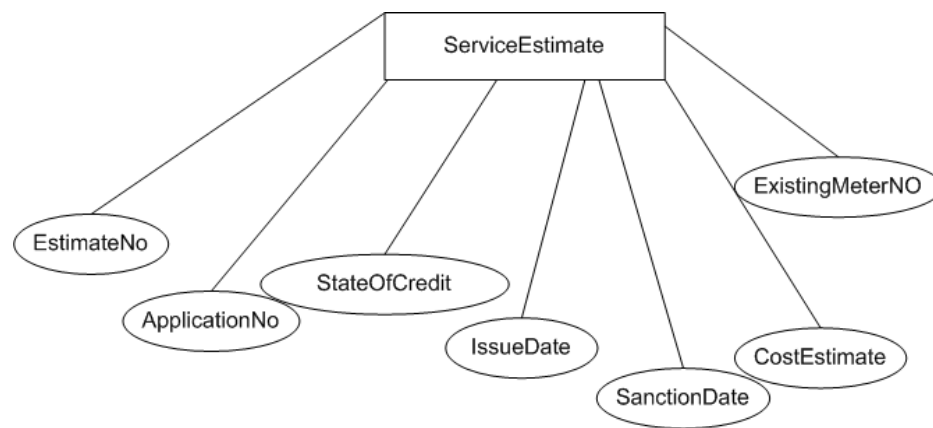
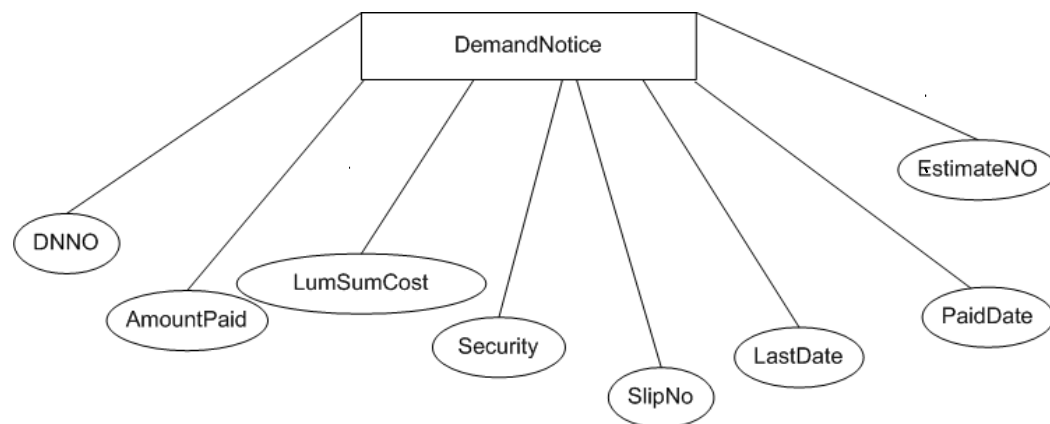
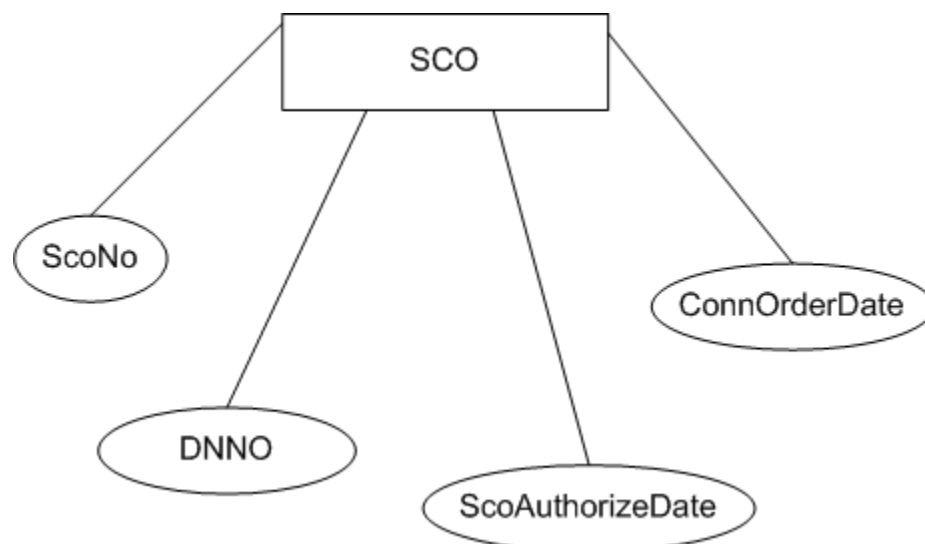
**4.2.2 DOD (Data Object Description) of New Meter Section**

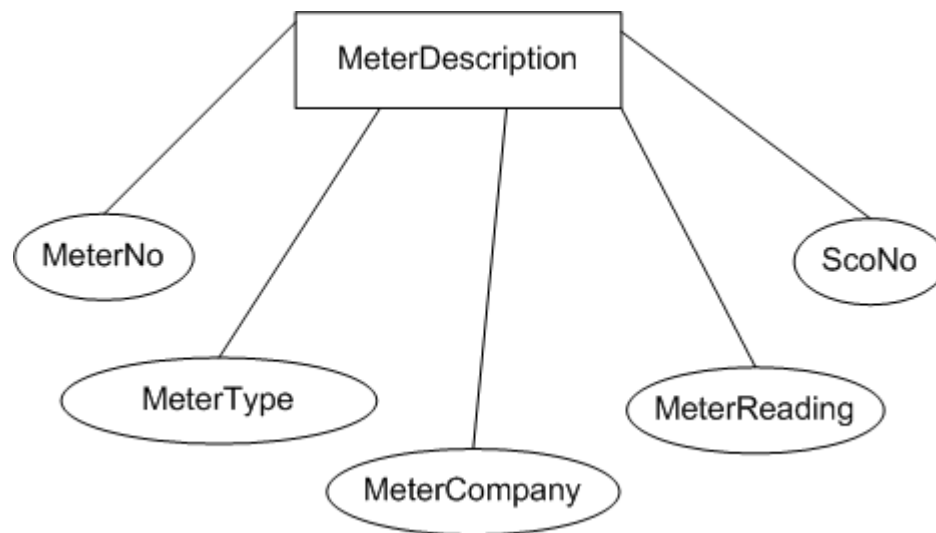
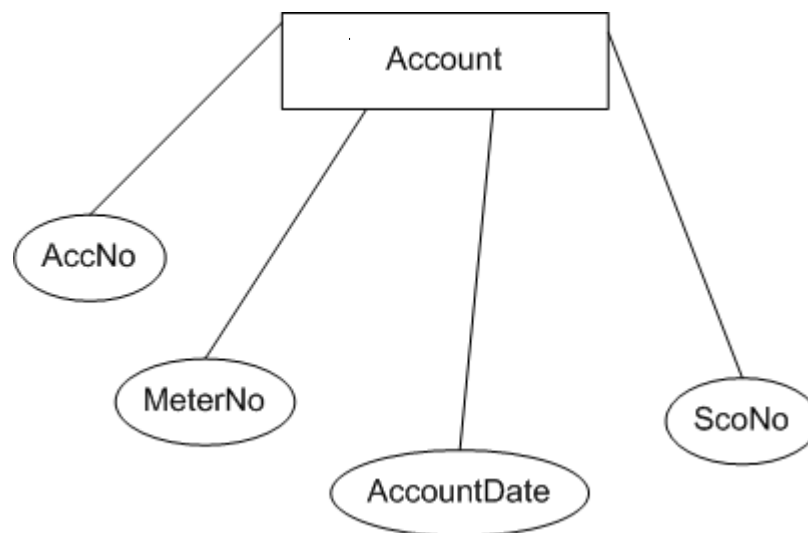
Data object description contains the attributes of each entity/data object. The New Meter Section contains the following entities.

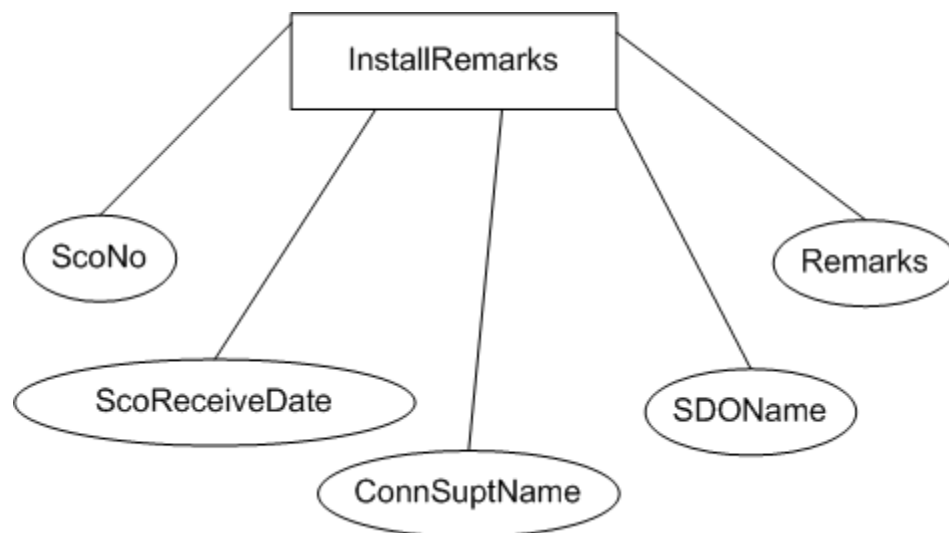
**4.2.2.1 ENTITY: Applicantinfo****4.2.2.2 ENTITY: Applicationinfo**



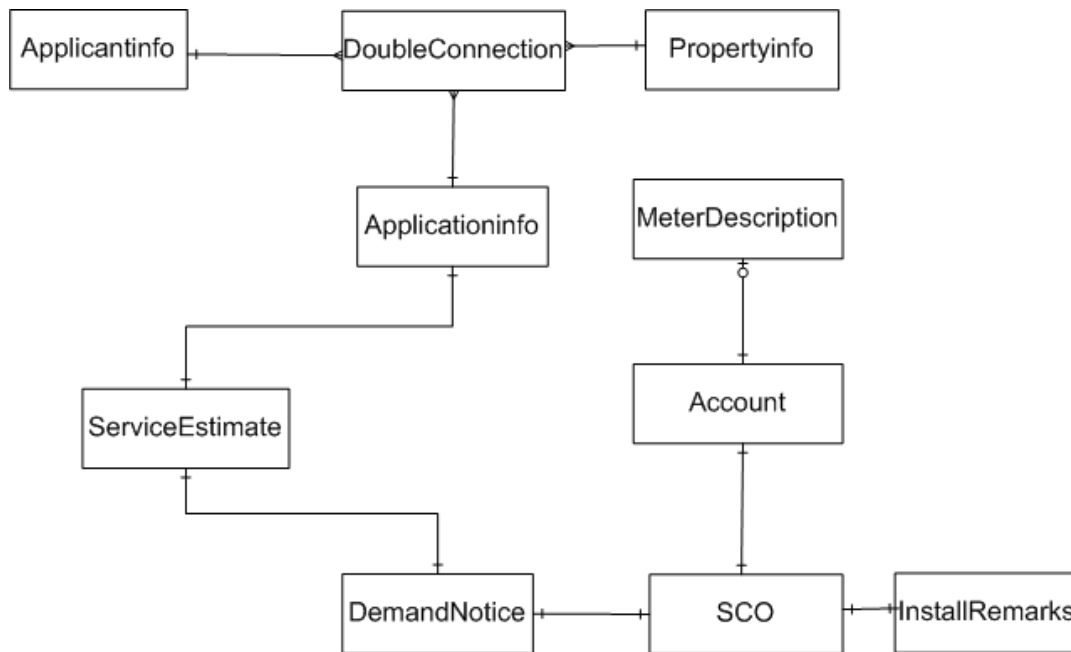
**4.2.2.3 ENTITY: Propertyinfo****4.2.2.4 ENTITY: DoubleConnection**

**4.2.2.5 ENTITY: ServiceEstimate****4.2.2.6 ENTITY: DemandNotice****4.2.2.7 ENTITY : SCO**

**4.2.2.8 ENTITY : MeterDescription****4.2.2.9 ENTITY : Account**

**4.2.2.10 ENTITY : InstallRemarks**

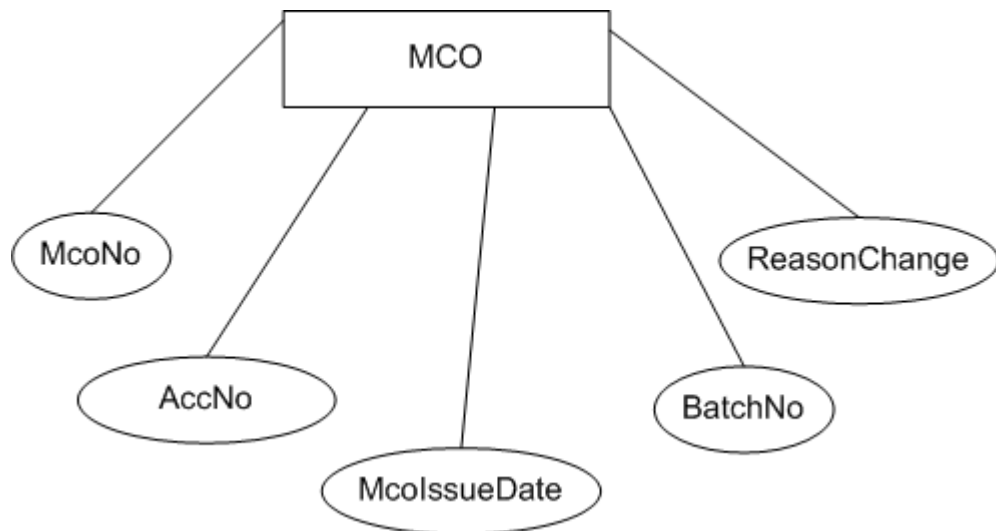
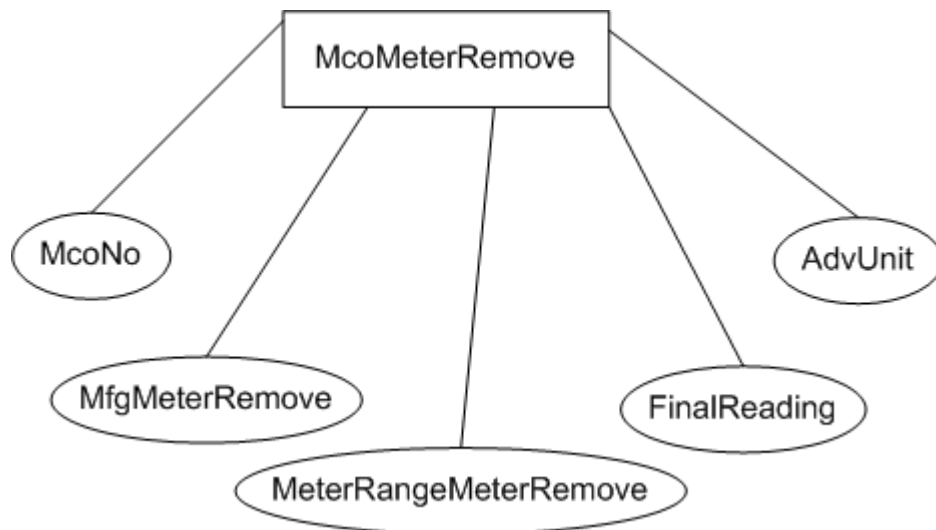
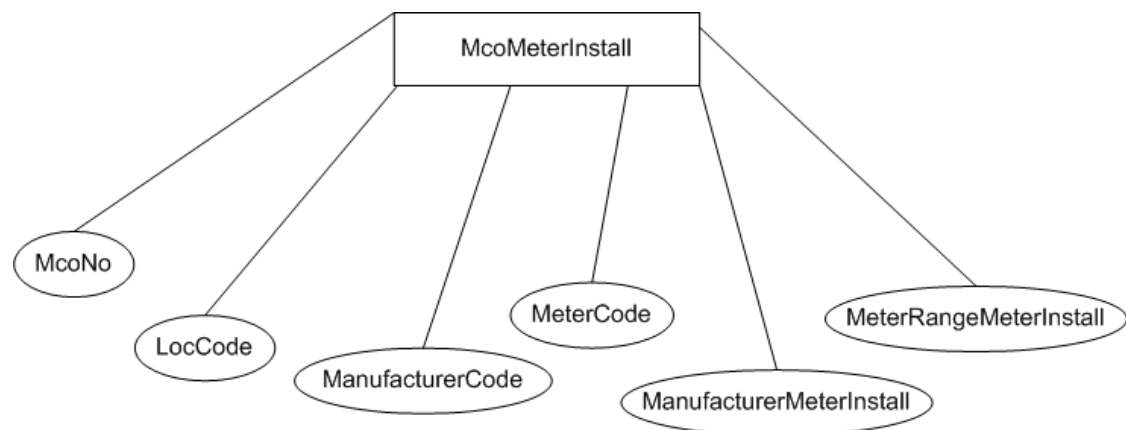
### 4.2.2.11 ERD For New Meter Section

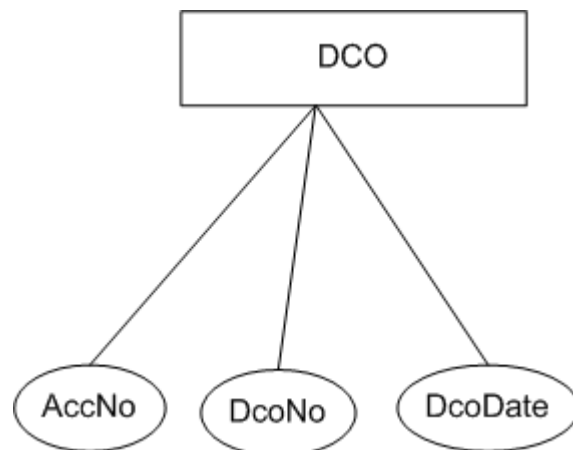
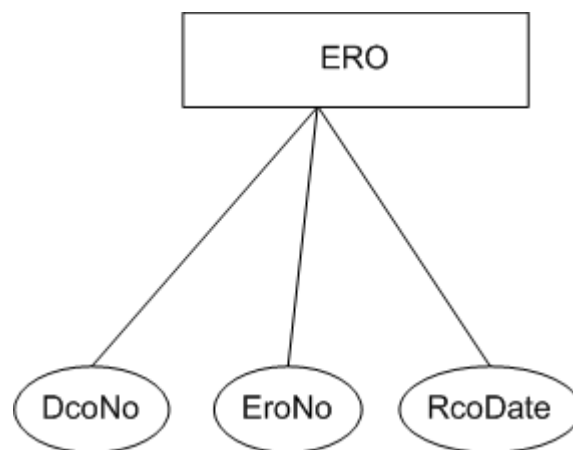


ERD For WAPDA  
University Town II New  
Meter Section

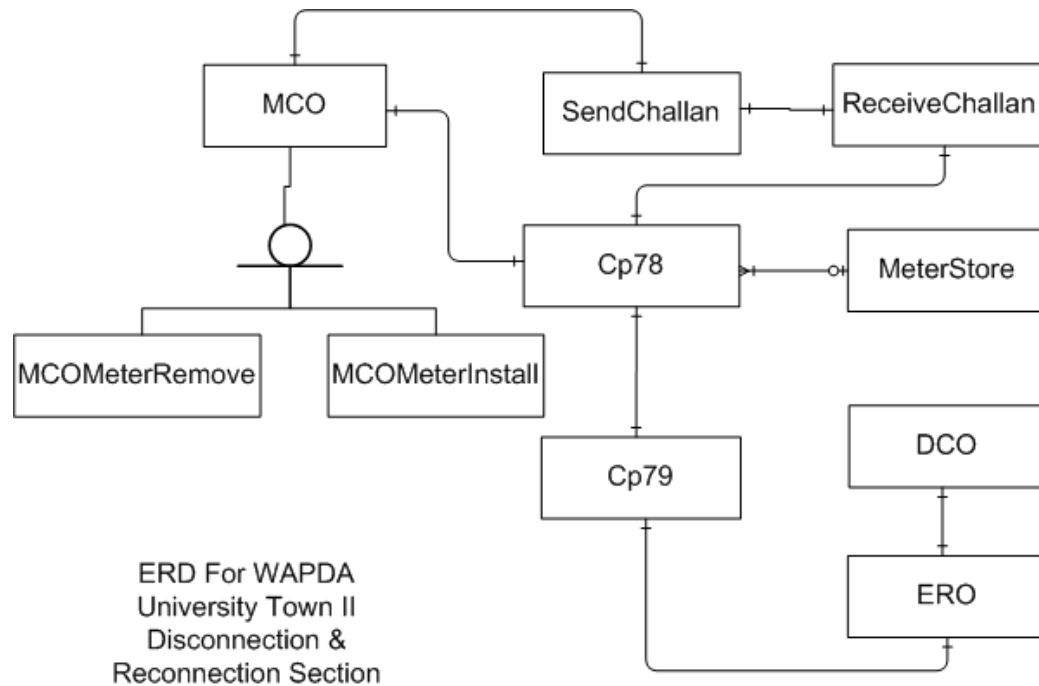
### 4.2.3 DOD (Data Object Description) of D & R (Disconnection and Reconnection) Section

Data object description contains the attributes of each entity/data object. The D & R section contains the following entities.

**4.2.3.1 ENTITY:MCO****4.2.3.2 ENTITY:MCOMeterRemove****4.2.3.3 ENTITY:MCOMeterInstall**

**4.2.3.4 ENTITY: DCO****5.2.3.5 ENTITY: RCO**

#### 4.2.3.6 D&R (Disconnection and Reconnection) ERD

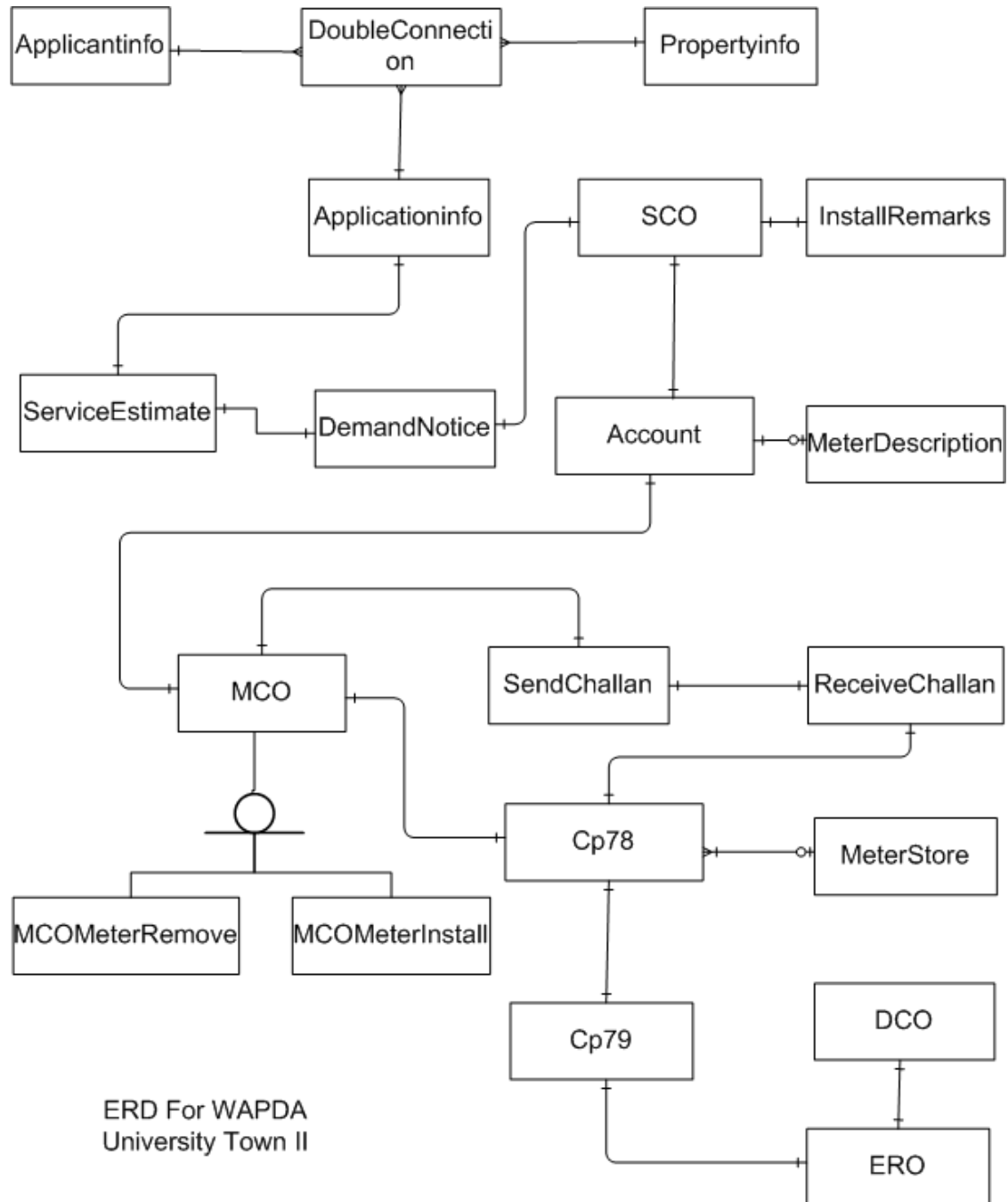


#### 4.2.4 ERD (Entity Relationship Diagram)

An entity-relationship (ER) diagram is a specialized graphic that illustrates the interrelationships between entities in a database. ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes.



4.2.5 WAPDA University Town II ERD



# **CHAPTER 5**

# **DATA DESIGN**

## 5.1 Introduction

We discussed the working of the Data Analysis and Molding in detail. This chapter is focused on the backend of the system. Here we create all the tables and their structure for the backend system.

## 5.2 BACKEND

The important step in backend is creating the tables. These tables will be the database for the program. But we should understand what we want to do before creating the tables and we should select the data.

First, create and select the tables. There are 20 tables in our project. These are created using oracle 10g.

We selected the tables but how we can create them. The steps of creating the tables are as follows:

## TABLES OF NEW METER SECTION

### 5.3 TABLE: APPLICANT INFORMATION

```
CREATE TABLE applicantinfo
(
  CNIC   VARCHAR2(15)          constraint appli_cnic_pk PRIMARY KEY,
  Name   VARCHAR2(40) ,
  FName  VARCHAR2(40) ,
  Address VARCHAR2(80)
);
```

### 5.4 TABLE: PROPERTY INFORMATION

```
CREATE TABLE propinfo
(
  propno   VARCHAR2(15) constraint PropNo_uniq unique ,
  CNIC     VARCHAR2(15) constraint propno_fk references applicantinfo(CNIC),
  Propadd  VARCHAR2(50) ,
  SubstationCon VARCHAR2(40)
```

);

### 5.5 TABLE: APPLICATION INFORMATION

CREATE TABLE appinfo

```
(
  AppNo    NUMBER(15)    constraint app_appno_pk  PRIMARY KEY,
  CNIC     VARCHAR2(15)  constraint app_cnic_fk  references applicantinfo(CNIC),
  New_Ext_Change VARCHAR2(20) constraint app_nec_nn  not null,
  ConnLoad  VARCHAR2(10),
  AppDate  DATE,
  StateOfCredit VARCHAR2(15)
);
```

### 5.6 TABLE: DOUBLECONNECTION INFORMATION

CREATE TABLE main

```
(
  CNIC     VARCHAR2(15),
  PropNo   VARCHAR2(20),
  AppNo    NUMBER(15) ,
  PropNo2  VARCHAR2(20)    constraint main_pro2_uniq unique ,
  constraint main_cpa_cpk  primary key (CNIC,PropNo,AppNo)
);
```

### 5.7 TABLE: SERVICE ESTIMATION INFORMATION

CREATE TABLE serv\_estm

```
(
  EstimateNo    NUMBER    constraint serv_estm_estmno_pk  PRIMARY KEY,
  AppNo         NUMBER(10) constraint serv_estm_appno_fk references appinfo(AppNo),
  SOCredit      VARCHAR2(10),
  DateIssue     DATE,
  DateSanction  DATE,
  CostEstimate  NUMBER(5)  constraint serv_estm_costestm_nn not null,
  ExistMeterNo  NUMBER(15)
);
```

### 5.8 TABLE: DEMAND NOTICE INFORMATION

```

create table demandnotice
(
  DnNo      number      constraint demnots_dnno_pk   primary key,
  AmountPaid number      constraint demnots_amontpay_nn not null,
  LumsumCost number,
  Security  number,
  SlipNo    number      constraint demnots_slipno_nn not null,
  LastDate  date,
  PayDate   date,
  EstimateNo number      constraint demnots_estmno_fk references
serv_estm(EstimateNo)
);

```

### 5.9 TABLE: SCO INFORMATION

```

create table serv_conn_ord
(
  SCONo    number      constraint sco_scono_pk   primary key ,
  DnNo     number      constraint sco_dnno_fk    references demandnotice(DnNo),
  SCO_Authr_date date,
  conn_load varchar2(12),
  Conn_date date
);

```

### 5.10 TABLE: METER DESCRIPTION INFORMATION

```

create table meterdesc
(
  MeterNo      number      constraint meter_metrno_pk primary key,
  MeterType    varchar2(20),
  MeterComp    varchar2(20),
  Status       varchar2(15)
);

```

### 5.11 TABLE: ACCOUNT INFORMATION

```

create table account
(
  AccNo    number      constraint acc_accno_pk   primary key,

```

```

MeterNo    number    constraint acc_metrno_fk    references meterdesc(MeterNo),
SCONo     number    constraint acc_scono_fk    references serv_conn_ord(SCONo),
AccDate        date
);

```

### 5.12 TABLE: INSTALL REMARKS INFORMATION

create table installremark

```

(
  SCONo     number    constraint install_scono_fk references serv_conn_ord(SCONo) ,
  ScoReceiveDate  date,
  ConnSuprName  varchar2(40) constraint install_suprvisn_nn not null,
  SdoName       varchar2(40) constraint install_sdon_nn    not null,
  remarks       varchar2(40)
);

```

## TABLES OF D&R (Disconnection and Reconnection) SECTION

### 5.13 TABLE: METER STORE INFORMATION

create table store

```

(
  MeterNo      number          constraint store_metrno_pk primary key,
  MeterType    varchar2(20),
  MeterComp    varchar2(20),
  Status       varchar2(15)
);

```

### 5.14 TABLE: MCO INFORMATION

create table mco

```

(
  McoNo      number    constraint mco_mcono_pk    primary key,
  AccNo      number    constraint mco_accno_fk    references account(AccNo),
  McoIssueDate  date,
  BatchNO    number,
  Address    varchar2(40),
  ReasonChange  varchar2(40) constraint mco_roch_nn    not null
);

```

);

### 5.15 TABLE: MCO METER REMOVE INFORMATION

create table mco\_mtrre

```
(  
  McoNo          number  constraint mcomtr_mcono_fk  references mco(McoNo) ,  
  Mf_Mr          varchar2(20),  
  MtrRange_mr   number,  
  FinalReading  number,  
  AdvUnit       number  
)
```

### 5.16 TABLE: MCO METER INSTALL INFORMATION

create table mco\_mtrins

```
(  
  McoNo          number  constraint mcomtrins_mcono_fk  references mco(McoNo),  
  MnfgCode      varchar2(20),  
  MeterCode     varchar2(20),  
  Mf_mi         varchar2(20)  
);
```

### 5.17 TABLE: SEND CHALLAN INFORMATION

create table sentchallan

```
(  
  SentChNo      number  constraint sench_sentchno_pk  primary key,  
  ChDate        date,  
  McoNo         number  constraint sentch_mcono_fk    references mco(McoNo),  
  MeterNo       number  
);
```

### 5.18 TABLE: RECEIVE CHALLAN INFORMATION

create table receivechallan

```
(  
  ReceivChNo   number  constraint recch_rechno_pk      primary key,  
  ReceivChDate date,  
  SentChNo     number  constraint recch_sentchno_fk   references sentchallan(SentChNo),  
  mt_remarks   varchar2(20)
```

);

### 5.19 TABLE: CP78 INFORMATION

create table cp78

```
(
  SNoCp78      number    constraint cp78_sno_pk    primary key,
  MeterNo      number    constraint cp78_mtrno_fk  references store(MeterNo),
  McoNo        number    constraint cp78_mcono_fk  references mco(McoNo),
  ReceivChNo  number    constraint cp78_rechno_fk references receivechallan(ReceivChNo)
);
```

### 5.20 TABLE: DCO INFORMATION

create table dco

```
(
  DcoNo        number    constraint dco_dcono_pk  primary key,
  DCODate      date ,
  AccNo        number    constraint dco_accno_fk  references account(AccNo),
  MeterNo      number
);
```

### 5.21 TABLE: ERO INFORMATION

create table ero

```
(
  EroNo        number    constraint ero_dcono_pk  primary key,
  DcoNo        number    constraint ero_dcono_fk  references dco(DcoNo),
  ERODate      date,
  FinalReading number,
  Balance      number
);
```

### 5.22 TABLE: CP79 INFORMATION

create table cp79

```
(
  SNoCp79      number    constraint cp79_sno_pk    primary key,
  SNoCp78      number    constraint cp79_snocp78_fk  references cp78(SNoCp78) ,
  dateRemove   date ,
  DcoNo        number    constraint cp79_dcono_fk  references dco(DcoNo) ,
);
```



```
    remarks    varchar2(40)
);
```

### 5.23 Conclusion

This chapter discussed the Backend system from its requirements point of view.

# **CHAPTER 6**

# **SYSTEM DESIGN**

## 6.1 SYSTEM INTERACTION

As its name indicates that, the diagram is used to describe some type of interactions among the different elements in the model. So this interaction is a part of dynamic behavior of the system. This interactive behavior is represented in UML by two diagrams known as Sequence diagram and Collaboration diagram. The basic purposes of both the diagrams are similar. Sequence diagram emphasizes on time sequence of messages and collaboration diagram emphasizes on the structural organization of the objects that send and receive messages. The purposes of interaction diagrams are to visualize the interactive behavior of the system. Now visualizing interaction is a difficult task. So the solution is to use different types of models to capture the different aspects of the interaction. That is why sequence and collaboration diagrams are used to capture dynamic nature but from a different angle. So the purpose of interaction diagram can be describes as:

- a) To capture dynamic behavior of a system
- b) To describe the message flow in the system
- c) To describe interaction among objects
- d) To describe structural organization of the objects

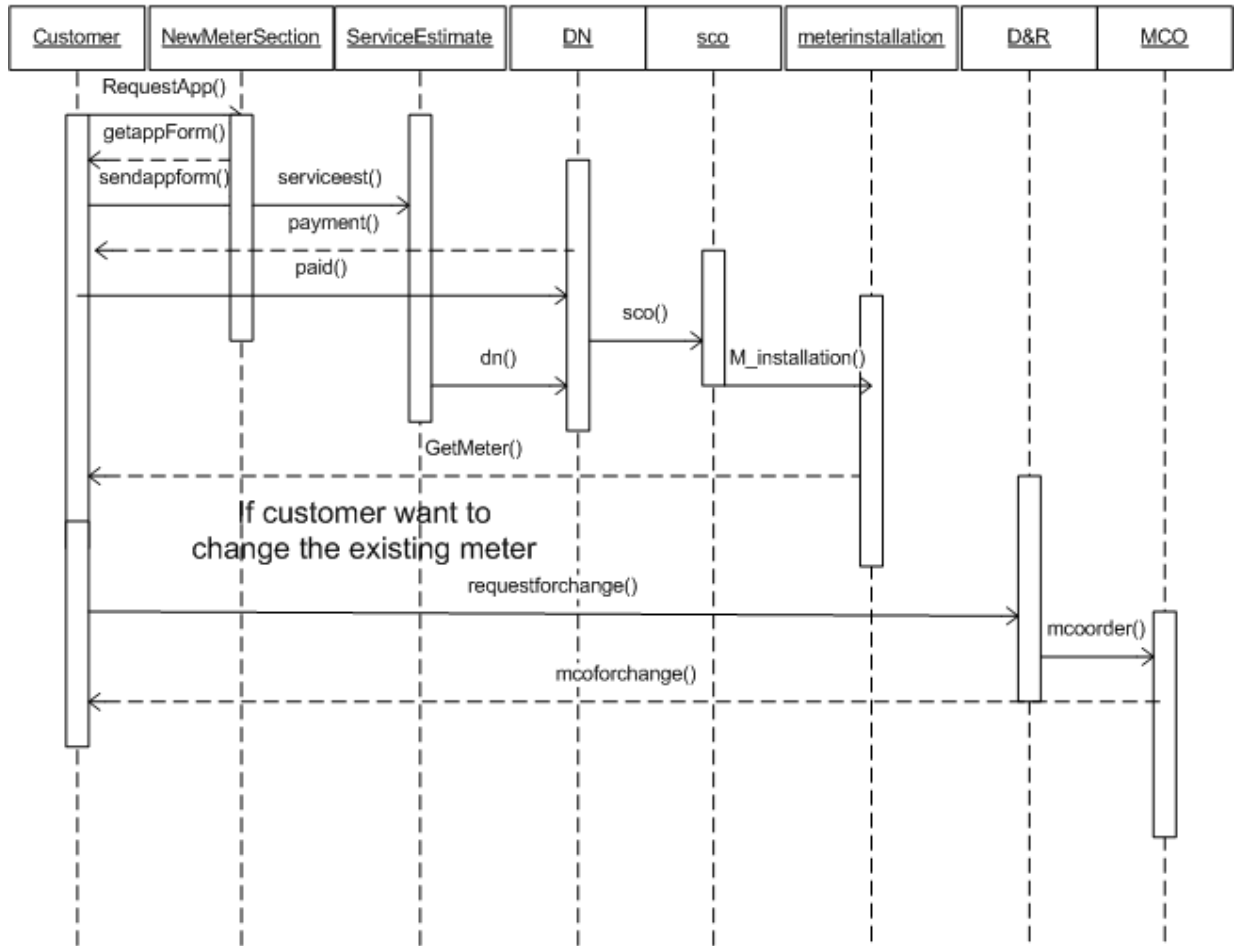
We have two types of interaction diagrams in UML. One is Sequence Diagram and the second one is Collaboration Diagram.

Now to choose between these two diagrams the main emphasis is given on the type of requirement. If the time sequence is important then sequence diagram is used and if organization is required then collaboration diagram is used. Both are discussed below.

### 6.1.1 Sequence Diagram

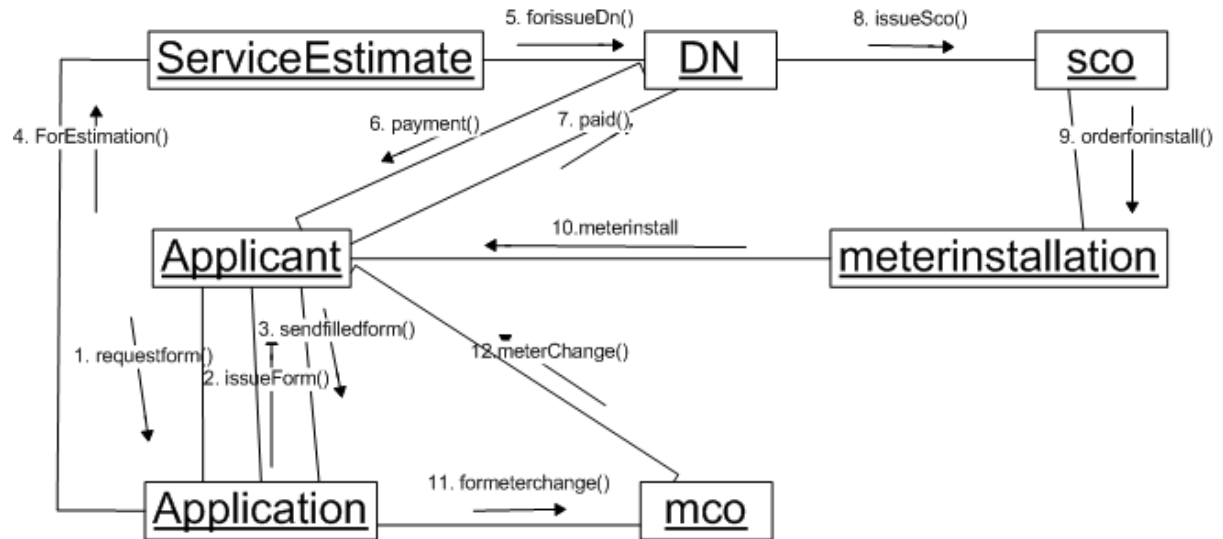
The sequence diagram captures the time sequence of message flow from one object to another.

Sequence Diagram of WAPDA University Town II



### 6.1.2 Collaboration Diagram

The collaboration diagram describes the organization of objects in a system taking part in the message flow. Here in collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the WAPDA University Town II system to describe the collaboration diagram.



WAPDA University Town II  
Collaboration Diagram

The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization where as the collaboration diagram shows the object organization.

## 6.2 Class Diagram

The purpose of the class diagram is to show the types being modeled within the system. In most UML models these types include:

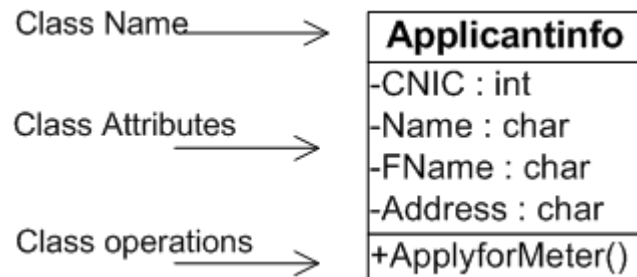
- a class
- an interface
- a data type
- a component.

UML uses a special name for these types: "classifiers." Generally, you can think of a classifiers as a class, but technically a classifier is a more general term that refers to the other three types above as well.

### 6.2.1 Class name

The UML representation of a class is a rectangle containing three compartments stacked vertically, as shown in Figure 6.1. The top compartment shows the class's name. The

middle compartment lists the class's attributes. The bottom compartment lists the class's operations. When drawing a class element on a class diagram, you must use the top compartment, and the bottom two compartments are optional.



**Figure 6.1 Applicant info class**

### 6.2.2 Class attribute list

The attribute section of a class (the middle compartment) lists each of the class's attributes on a separate line. The attribute section is optional, but when used it contains each attribute of the class displayed in a list format. The line uses the following format:

Name : attribute type

CNIC Integer

### 6.2.3 Class operations list

The class's operations are documented in the third (lowest) compartment of the class diagram's rectangle, which again is optional. Like the attributes, the operations of a class are displayed in a list format, with each operation on its own line. Operations are documented using the following notation:

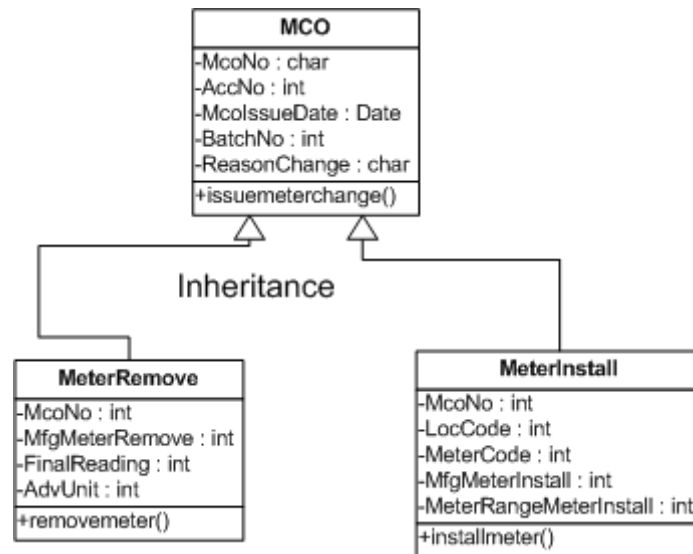
name(parameter list) : type of value returned

ApplyforMeter()

### 6.2.4 Inheritance

A very important concept in object-oriented design, inheritance, refers to the ability of one class (child class) to inherit the identical functionality of another class (super class), and then add new functionality of its own. (In a very non-technical sense, imagine that I inherited my mother's general musical abilities, but in my family I'm the only one who

plays electric guitar.) To model inheritance on a class diagram, a solid line is drawn from the child class (the class inheriting the behavior) with a closed, unfilled arrowhead (or triangle) pointing to the super class



**Figure 6.2.4 Inheritance**

In Figure 6.2.4, the inheritance relationship is drawn with separate lines for each subclass, which is MeterRemove and MeterInstall.

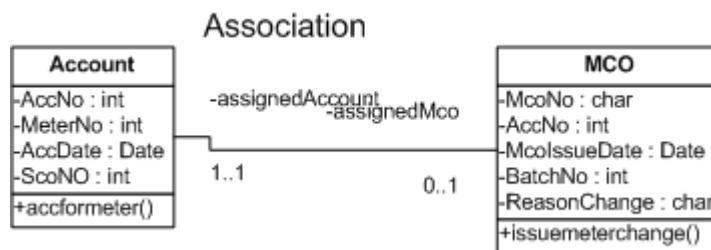
### 6.2.5 Associations

When you model a system, certain objects will be related to each other, and these relationships themselves need to be modeled for clarity. There are five types of associations. We just discuss two of them bi-directional and uni-directional associations

#### 6.2.5.1 Bi-directional (standard) association

An association is a linkage between two classes. Associations are always assumed to be bi-directional; this means that both classes are aware of each other and their relationship, unless you qualify the association as some other type.

Figure 6.2.5.1: An example of a bi-directional association between an Account class and MCO class



**Figure: 6.2.5.1 Bi-directional Association**

A bi-directional association is indicated by a solid line between the two classes. At either end of the line, you place a role name and a multiplicity value.

**Table 6.2.5.1: Multiplicity values and their indicators**

Potential Multiplicity Values

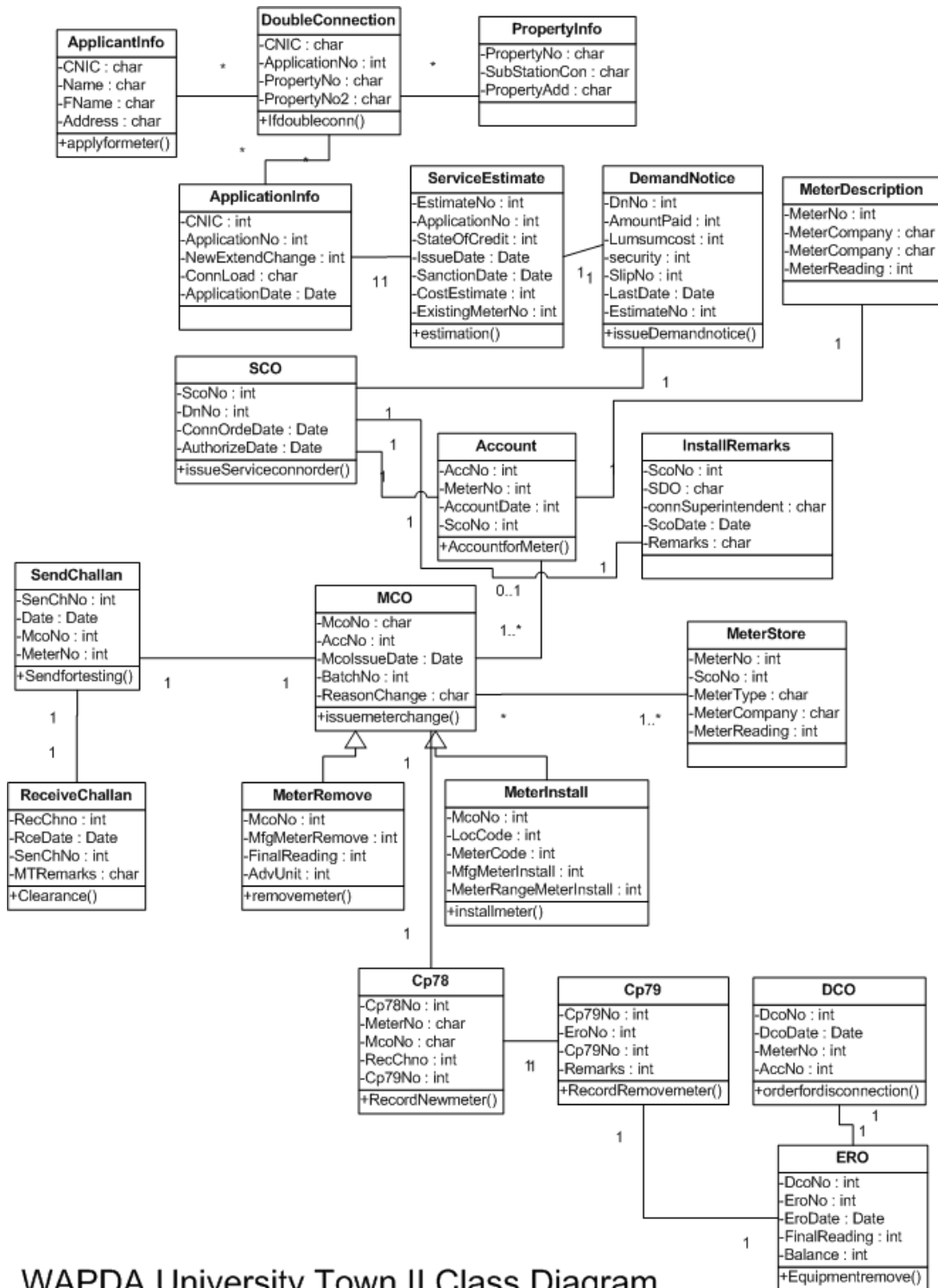
Indicator	Meaning
0..1	Zero or one
1	One only
0..*	Zero or more
*	Zero or more
1..*	One or more

### 6.2.5.2 Uni-directional association

In a uni-directional association, two classes are related, but only one class knows that the relationship exists. Figure 7 shows an example of an overdrawn accounts report with a uni-directional association.



6.2.6 WAPDA University Town II Class Diagram



WAPDA University Town II Class Diagram

### 6.3 User Interface Design

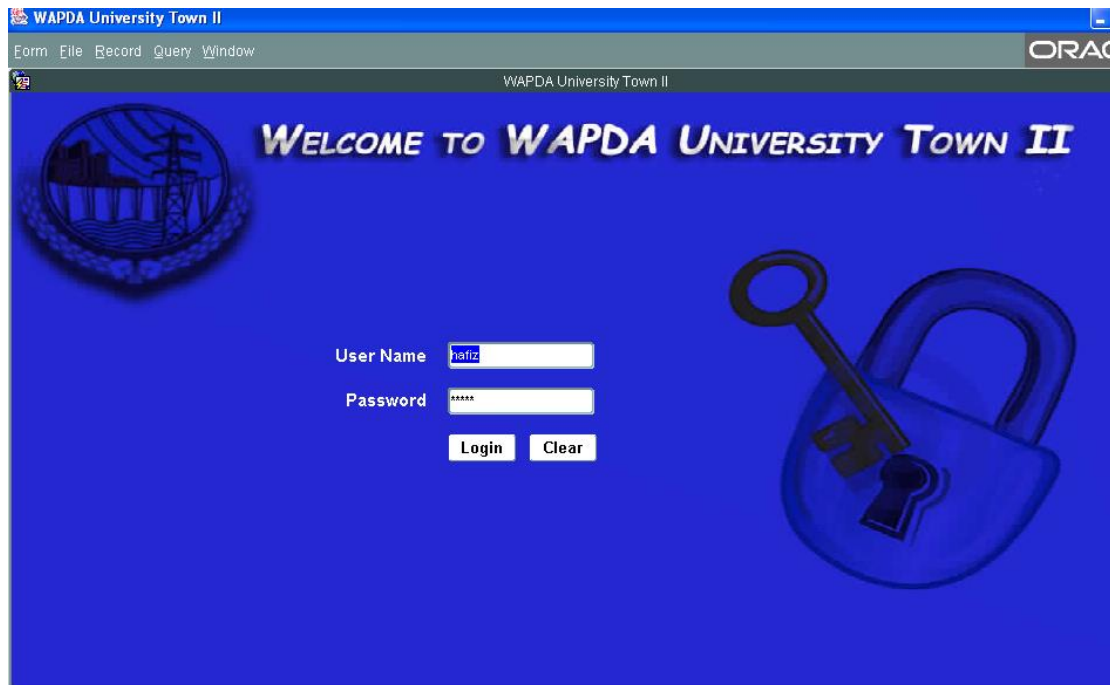
This software system is designed to keep the data of all the customer applied for meter. this software system has lot of features that have diversified and comprehensive uses. It is complete in all aspects pertaining to customers and keeping in access through World Wide Web. All the related aspects to the customer data entry and retrieval have been handled keeping the interface highly user friendly.

We have used the following graphical controls;

- Push Buttons
- Text Items
- List Items
- Display Items
- Image Item
- Radio Buttons
- Check Boxes
- LOVs
- Menus

### 6.3.1 Log In

As we start the OCJ Instance and then run the form, the user verification screen will appear. Here it asks about the username, password, and the database to which we want to connect.



**Figure: 6.3.1 Logon Page**

### 6.3.2 Main page:

On successful login, we are directed to the main page, from where we can control the overall software, the main page consists of buttons having links to different forms.

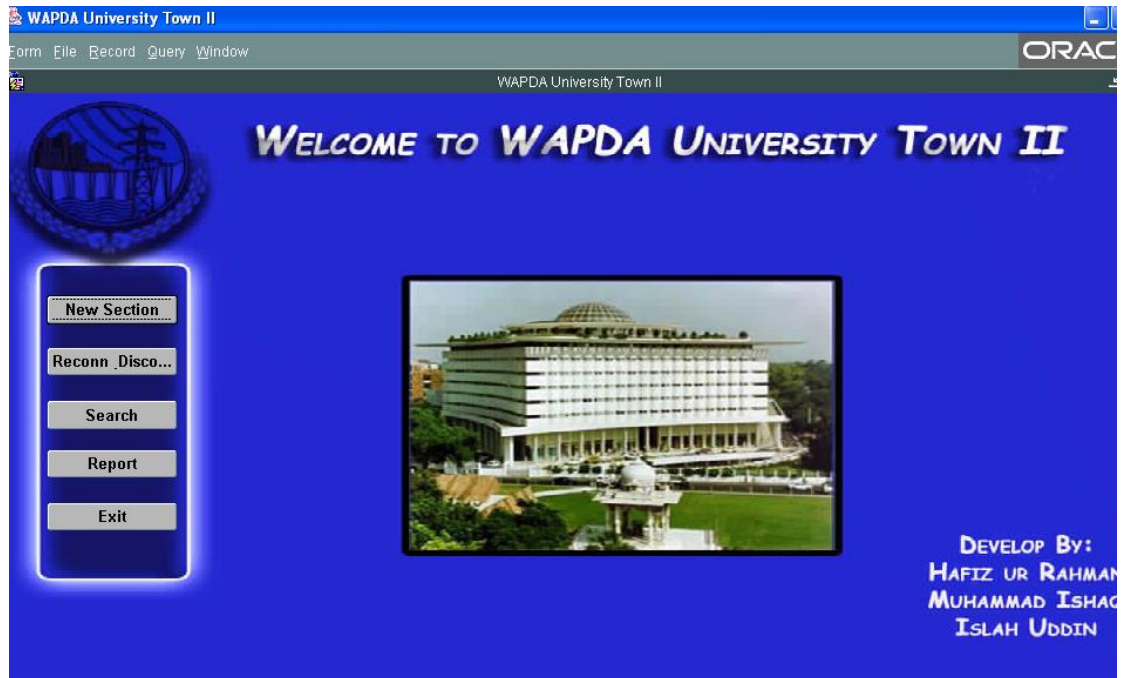


Figure: 6.3.2 Main Page

### 6.3.3 New Meter Section

The New Meter Section main form is used to trace all the tables related to the customer, which store those customer records who apply for new meter.

WAPDA University Town II


Form File Record Query Window

WAPDA University Town II

App Double Connection Estimate Account

## NEW METER SECTION

**Applicant Information**

 CNIC: 15306-1710826-5  
Name: Hafiz Rahman  
Father Name: Raz Zamin  
Address: Shaheen Town

**Property Information**

Property No: ms331  
CNIC: 15306-1710826-5  
Property Address: Koherai Malakand  
Substationcon: Town II

**Application**

Application No: 1      New Ext Change: New Meter  
CNIC: 15306-1710826-5      Connload: 250  
AppDate: 19-DEC-2011      StateOfCredit:

NewRecord Save Delete Clear Next Back Main Page Exit

P S N

Figure: 6.3.3 New Meter Section

### 6.3.4 Disconnection And Reconnection (D&R) Section

The Disconnection and Reconnection Section main form is used to trace all the tables related to the customer, which store those customer records who want to change meter due to several reasons i.e. burnt, damaged etc.

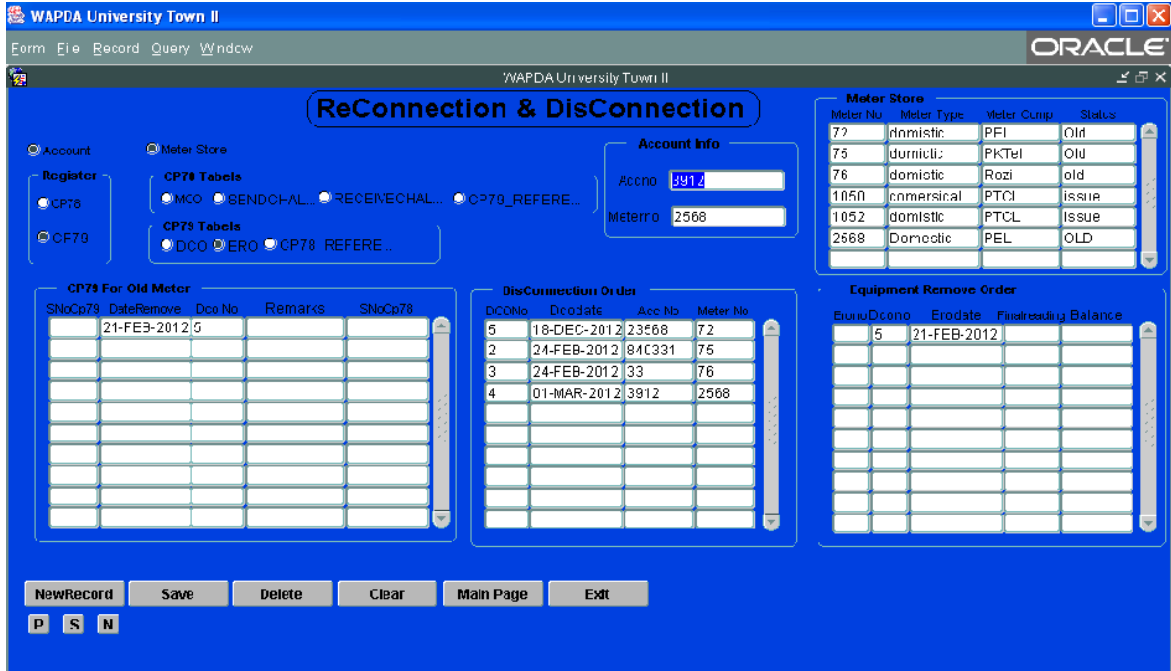


Figure: 6.3.4 New Meter Section

### 6.3.5 Searching:

Three types of searching criteria is provided to the operator, search by ID, search by Name and Search by the Father name.

On successful search it will show the personal information regarding the student as required in the form.

It also contains the fee information of the student in each and every class. And will also provide information about the concession if granted.

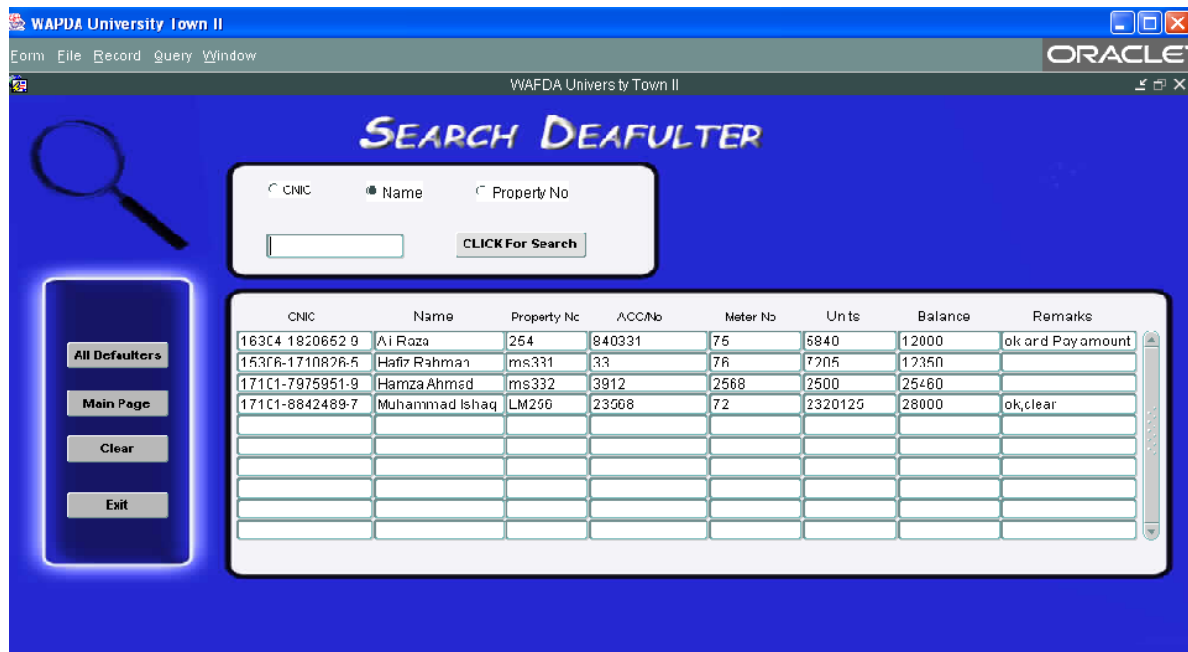


Figure: 6.3.5 Searching Page

### 6.3.6 Reports:

The report module is capable to generate different types of reports according to the given requirement.



Figure: 6.3.6 Report



## 6.4 Implementation

This section is focused on the Implementation and Maintenance of the proposed system. Here the Proposed system Requirement and Language Selection are determined for the proposed system.

### 6.4.1 Implementation Strategy

According to initial study and design phase, we have to decide about the implementation strategy which includes application type, Language selection and proposed system requirements

1. Proposed system Requirement
2. Language Selection

#### 6.4.1.1 Proposed System Requirement

According to the system requirement and end-user requirement, we have to select such tools which make such an interface that is user-friendly and helps to achieve the required goal.

The main requirement of user is that we make such software for their organization which provide maximum security, easy to operate, modifiable according to new needs, so we analyze many software development tools and found Oracle Database and Developer 2K fulfill requirement fully so we choose that tool to make this system.

#### 6.4.1.2 Language Selection

The selection of most suitable tool is important step in software development. So, such a language is chosen which is user friendly and platform independent.

##### a) Developer 10g

Developer 2K is most frequently tool used to develop database application because it gives a lot of security to end-user and also it is user-friendly. Developer 10g is event-driven language and we have to write too short code to access any function. Oracle includes built-in tools which help much to develop application easily.

The most important feature of this language is that it allows us to create database and front-end applications for most popular enterprise level database application.

The tool used to create database is Oracle10g. It is most powerful tool used to create database which maintain data accuracy and give high level of security. Each table made in database is referred as an entity. User can add, delete, and update data in oracle10g.

#### **b) Oracle 10g**

One of the most powerful and dominant feature of Oracle10g database is that it provide high level of security and data accuracy while inserting or updating Data when user work in central or single user environment, also when Oracle10g used in distributed environment when number of user are adding, deleting or updating data at same time then it also provide same level of security and replication as it provide in single user environment. Oracle 10g forms run in web environment and easily access from remote login.

### **6.4.3 SYSTEM IMPLEMENTATION**

Implementation is the final step towards the develop system. Those activities, which convert the old system to new one, are included Implementation. There may exist confusion to some extent, between these two Terminologies, conversion and implementation. Conversion is referred to as the significant and ardent relationship between old system and new one whereas the practical job of putting a theoretically designed system into practice is signified as implementation.

Implementation project involves the activities like planning and scheduling of implementation process, organizational planning and personnel administration, final system design and testing, establishment of old to new system. The most important and considerable process of implementation includes testing the system and conversion plan.

This phase starts with the acceptance of tested design to its satisfactory operation. After developing the software of “WAPDA University Town II”, we wish to implement it.

# **CHAPTER 7**

## **HISTORY OF WORK**

## 7.1 History of Work

History of work of the development software is the following milestones that we achieved

- Project Selection
- Requirement Gathering
- Data Analysis
- Data Modeling
- Interface Designing
- Coding

### 7.1.1 Project Selection

In project selection, all of our group members decided and arranged meeting to decide a project for the partial fulfillment of the BCS (Hon) degree from the computer science department, University of Peshawar. All of the members presented different domain problems but agreed upon the computerization of WAPDA University Town II. Then we went to the WAPDA University Town II and met the SDO and discussed with him and he also agreed with us. He promised to help us in every aspects of this project.

We submit the proposal of the project and chose the Dr. Sara Shahzad as a Project supervisor.

### 7.1.2 Requirement Gathering

It is a preliminary stage for database design. We started the requirement gathering after submission of the proposal. We started our struggle for the computerization of WAPDA University Town II and went there to collect data from the two sections. The clerks of these sections are cooperative and provided us all the data. We went the WAPDA University town II many times to clear the data for us in which, we had ambiguity.

### 7.1.3 Data Analysis

After the Requirement Gathering we sat and analyzed the data. In this analization we remove the duplication and redundancy from the data. In data analysis stage we analyze the gather data to identify the requirement to be included in the new database system. We structured the unstructured data. We did normalization to remove redundancy and duplication from data.

#### **7.1.4 Data Modeling**

In data modeling we created the conceptual model in which, we described the various entities, their attributes and relationships.

In data modeling we first identify the objects in our projects then their attributes and relationships among those entities. At last we individually created ERD for each section of WAPDA University Town II and also created the complete ERD for our project.

We discussed the ERD with Sir Bilal and approved our ERD.

#### **7.1.5 Interface Design**

In interface design, we designed three individual interfaces.

- Main Interface
- New Meter Section Interface
- Disconnection and Reconnect (D&R) section interface

In these interfaces, we used push buttons, text items, image items etc. The above interfaces can be accessed through buttons in our projects.

#### **7.1.6 Coding**

After approving the ERD we started coding using SQL\*plus and Oracle 10g Forms Builders.

In SQL\*Plus we created all the tables that we had in our project and did designing and other coding for buttons, triggers etc in Forms Builder.

#### **7.2 Key Accomplishments**

- Worked on New Meter Section and computerized it.
- Worked on D&R Section and computerized it.
- Created Searching Form
- Created different Reports

# **CHAPTER 8**

## **FUTURE WORK**

## 8.1 FUTURE WORK

Future work of the development software is that it can be

- **Updateable (Technology)**
- **Modules can be enhanced**
- **New modules can be define**
- **Globally Access**

### 8.1.1 Updateable (Technology)

The current developed software is in Oracle 10g it can be Update with the new release of Oracle, technology can be changed.

### 8.1.2 Modules can be Enhanced

Current developed modules can be enhanced means if the organization wants changes in front end as well as at back end, further Reports can be made if needed.

### 8.1.3 New Modules can be Define

New modules can be defined and integrate with in the current developed software.

If organization need later on any report or any new module can be implementing and integrate in current software.

E.g. Modules for Employee salaries search etc.

### 8.1.4 Globally Access

The current developed system can be accessed globally over the internet in future if the organization has a web server and static IP.

## 8.2 REFERENCES

Following are the books from which we take assistance and guidance to complete our given task

1. Oracle Developer 2000,  
By: Ivan Behros, Publisher Prentice Hall,  
Publish Year: May 2000
2. <http://www.orafaq.com>
  - a. Oracle forums for online help
3. <http://www.oracle.com/documentation>
  - a. Detail information and help about Oracle
4. <http://yahoo.oracle/groups>
  - a. Group members help regarding Oracle
5. The Complete Reference Java,  
By: Herbert Scheldt, Publisher Apress,  
Publish Year: Nov, 2004
6. <http://www.otn.com/discussionforums>
  - a. Discussion Forums and online help for Oracle
7. <http://www.google.com>
  - a. Used for overall search
8. An Introduction to DATABASE SYSTEMS  
By: C. J. Date, 7<sup>th</sup> Edition, Publisher McGraw Hill
9. SOFTWARE ENGINEERING  
By: Roger's Pressman, 5<sup>th</sup> Edition



# **CHAPTER 8**

## **FUTURE WORK**

## 8.1 FUTURE WORK

Future work of the development software is that it can be

- **Updateable (Technology)**
- **Modules can be enhanced**
- **New modules can be define**
- **Globally Access**

### 8.1.1 Updateable (Technology)

The current developed software is in Oracle 10g it can be Update with the new release of Oracle, technology can be changed.

### 8.1.2 Modules can be Enhanced

Current developed modules can be enhanced means if the organization wants changes in front end as well as at back end, further Reports can be made if needed.

### 8.1.3 New Modules can be Define

New modules can be defined and integrate with in the current developed software.

If organization need later on any report or any new module can be implementing and integrate in current software.

E.g. Modules for Employee salaries search etc.

### 8.1.4 Globally Access

The current developed system can be accessed globally over the internet in future if the organization has a web server and static IP.

## 8.2 REFERENCES

Following are the books from which we take assistance and guidance to complete our given task

1. Oracle Developer 2000,  
By: Ivan Behros, Publisher Prentice Hall,  
Publish Year: May 2000
2. <http://www.orafaq.com>
  - a. Oracle forums for online help
3. <http://www.oracle.com/documentation>
  - a. Detail information and help about Oracle
4. <http://yahoo.oralce/groups>
  - a. Group members help regarding Oracle
5. The Complete Reference Java,  
By: Herbert Scheldt, Publisher Apress,  
Publish Year: Nov, 2004
6. <http://www.otn.com/discussionforums>
  - a. Discussion Forums and online help for Oracle
7. <http://www.google.com>
  - a. Used for overall search
8. An Introduction to DATABASE SYSTEMS  
By: C. J. Date, 7<sup>th</sup> Edition, Publisher McGraw Hill
9. SOFTWARE ENGINEERING  
By: Roger's Pressman, 5<sup>th</sup> Edition

# **CHAPTER 9**

## **Glossary**

**Actor**

Actors, in use case diagram, are parties outside the system that interact with the system. They may be users or other systems. Each actor defines a coherent set of roles users of the system can distinguish between primary and secondary actors. A primary actor is one having a goal requiring the assistance of the system. A secondary actor is one from which the system needs assistance to satisfy its goal.

**Association**

Associations between actors and use cases are indicated in use case diagrams by solid lines

**Attributes**

Attributes are the properties of an object

**Attainable**

Technically feasible. What is your professional judgment of the technical “do-ability” of the requirement?

**Class Diagram**

As mentioned earlier, the purpose of the class diagram is to show the types being modeled within the system

**Collaboration Diagram**

The collaboration diagram describes the organization of objects in a system taking part in the message flow

**Composability**

Ability to compose systems from plug-and-play components

**DATA OBJECTS**

Anything which have some properties (attributes) is called data object

**DDL**

Date Definition Language

**DFD**

Data Flow Diagram

**DML**

Data Manipulation Language

**Domain Model**

A domain model in problem solving and software engineering can be thought of as a conceptual model of a domain of interest (often referred to as a problem domain) which describes the various entities, their attributes, roles and

**D&R**

Disconnection and Reconnection. It is a section in WAPDA University Town II where old meters records are stored.

**ERD**

Entity Relationship Diagram. An entity-relationship diagram is a specialized graphic that illustrates the interrelationships between entities in a database

**Evolvability**

Support for new capabilities or ability to exploit new technologies

**Functional requirement**

Functional Requirements, which define the capabilities and functions that a system must be able to perform successfully

**Inheritance**

A very important concept in object-oriented design, inheritance, refers to the ability of one class (child class) to inherit the identical functionality of another class (super class), and then add new functionality of its own

**Interaction Diagram**

As its name indicates that, the diagram is used to describe some type of interactions among the different elements in the model. So this interaction is a part of dynamic behavior of the system

**Localizability**

Ability to make adaptations due to regional differences

**Measurable**

It is possible to verify that this requirement has been met. What tests must be performed or what criteria must be met to verify that the requirement is met?

**Modifiability**

Ability to add (unspecified) future functionality

**Non Functional Requirement**

In systems engineering and requirements engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors

**Packages**

Packages are UML constructs that enable you to organize model elements (such as use cases) into groups

**Realizable**

Realistic, given the resources. Do you have the staffing? Do you have the skill? Do you have access to the development infrastructure needed? Do you have access to the run-time infrastructure needed? Do you have enough time?

**RELATIONSHIP**

Relationship is a logical association between entity types

**Reusability**

Ability to (re)use in future systems

**Smart Requirements**

Smart requirements have the characteristics: Specific: without ambiguity, using consistent terminology, simple and at the appropriate level of detail.

**System Boundary Boxes**

You can draw a rectangle around the use cases, called the system boundary box, to indicate the scope of your system

**Traceable**

Linked from, its conception through its specification to its subsequent design, implementation and test.

**Use Case**

A use case describes a sequence of actions that provide something of measurable value to an actor and is denoted as a horizontal ellipse.