Palestine Polytechnic University

College of Engineering & Technology
Electrical & Computer Engineering Department

Graduation Project

E-Services for HEPCO

Project Team

Hiyam M.Zuher Badr
Tahanee Moh. Darabee'
Abeer Mahdy Aljabary

Project Supervisor

Dr. Salman Talahmeh

This project is submitted in partial fulfillment of the requirements for the degree of B.Sc. in Computer Systems Engineering in College of Engineering and Technology in Palestine Polytechnic University

Hebron-Palestine

January, 2005

Dedication

To my beloved mother Mariam. who stood by me and gave me all the support that I needed in all my life.

To the soul of my great father.....

Tahanee

To my parents, sisters, brothers, friends, teachers and every one granted me the encouragement and support.

Hiyam

To my father Mahdy Aljabary

To my mother

To our supervisor DR.Salman Talahma

Abeer

Acknowledgment

We would like to give our thanks to every one participated in the success of this project, whether by an advice or encouragement or useful material related to the project.

First of all, we are so grateful to our supervisor Dr. Salman Talahmeh, who gave us a lot of his precious time to put us on the right track during our work in the project, and he is never being hesitated or late to give us from his knowledge and experience.

Also, our thanks go to every one helped us in Hebron Electric Power Company employees, and for the library staff in the university.

Table of Contents

Table of Figures

List of Tables

Initiation

Hebron Electric Power Company (HEPCO) is a private shareholding company that was established in 2002. HEPCO is responsible for distributing electrical energy in the Hebron & Halhul regions. The total number of consumers is 27000 approximately. it became independent from Hebron municipality in 2004.The primary concern of HEPCO in delivering electricity to its customers in its concession area is to ensure safe, efficient and reliable supply of electricity.

Until today the company is under construction in terms of software applications, managements, and data collections. it keeps data in oracle DB. and they have a separate application for entering readings. The company requests for a computer system that will be accessible via a web page in the internet. This system will be able to organize the process of paying bills that are issued by the electric company; a user name and a secret password are given to the user in order to be able to see his own bills only. In addition the user will be able to pay bills. by accessing the webpage of the Arab bank. making it possible to transfer money from the users' accounts to the account of the electrical company. Also the software will give the ability for the client to fill in a form that provides the company with his personal data. This process allows the user to benefit from company provided services. Moreover the user can inform via the web about any problem in the electrical network.

Over all of this. the company requests for a computer system that will give the facility for any customer to pay the bills via the web. any time and any where. only he needs account in a certain bank. and a login id for the HEPCO web which show him his own bills and a login id for the credit bank web page which supports the paying process. And at once the customer transfers the value of the bill, the money will be discounted from his account in the bank immediately then transferred to the account of the company later. After that. there will be a certain communication between the electrical company and the bank, to tell the company who has paid the electricity bill via the web, the system also is to have the ability to fill in a form for registration in the company to get services provided by the company. also reporting their problems and feedbacks

The system should be with a comfort user interface, and easy to deal with. also the numbers in the web is to be in 6 digits. The system should follow the regulations that are followed by other similar international systems in terms of security, user interface and the number of login id's characters.

Chapter

One

Introduction

1.1 Overview

By entering the new millennium we have witnessed many changes in the world. Perhaps the most noticed one is the tendency and huge increase of using the Internet in everything. Two of the most important Internet applications are e-bill and e-Payment services.

Paper bills, invoices, statements and other documents are rapidly becoming a thing of the past, as consumers and corporate clients demand the convenience and flexibility of electronic access to account information. In return, financial institutions and large size companies are looking for ways to satisfy the customers' needs while capitalizing on the potential for cost reduction. This presents a great opportunity for existing mail service providers, payment service providers, or new billing service providers to expand business and create new revenue streams [1].

Online bills. invoices and statements...electronic payment...and online customer care can make this happen. These services reduce call center costs and volume, and save big-time with decreased paper and postage requirements.

E-Bill is a secure online billing service. Nowadays, receiving and paying bills online (E-Bill) is becoming a necessity in our society. It is expected that E-Bill systems can provide a fast, easy and secure system. Just by a click or hit of a button the user can make his payment and finishes very fast before noticing that. Indeed, the user will love the time he saves and the tediousness he avoids.[2]

Electronic billing and payment are efficient and cost-effective ways to provide online presentment of virtually any kind of document; from traditional bills, insurance notices and complex invoices, to flexible software solution that addresses your requirements for speed to market, resource efficiency, low cost and simplicity [3].

Using E-bill and E-payment can provide the following services:

- Receiving notes and details of the bill by the company web (in most cases, depending on your method of enrolling).
- Viewing bills online. Once the customer set up his payment account, he will be able to pay it with just a mouse click or hit of a button.
- Scheduling e-Bill payments and adjust payment amounts to fit customer's needs.
- Paying bills wherever you are -- at home, at work or on the road, any time you have an extra two minutes.
- Downloading bills so customer can print it, the customer will be able to save bills detailed statement on his PC.
- There is no stamps or trips to the post office

The figure (1.1) illustrates the e-bill /e-payment process [4]:

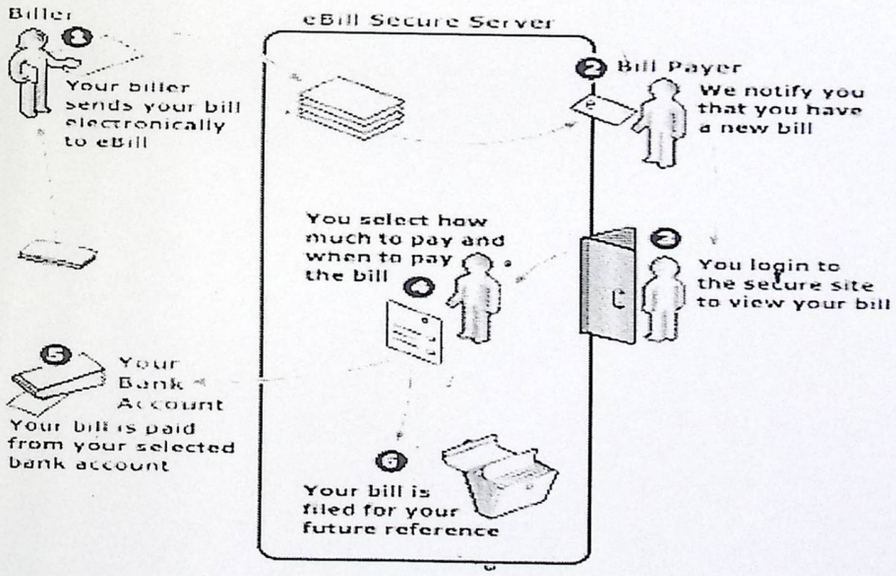


Figure 1.1 E-bill /e-payment process

1.2 Importance of E-Bill and E-Payment System

E-Bill and E-Payment System is the easy, smart way for consumers to securely receive, manage and pay their bills on the Internet. Using this system to get their services [5]:

- The system can automate many processes that managers and their subordinates must accomplish
- Time and many can be saved by using this system to do what previously could be done only manually
- High level software let managers develop their own tools for daily monitoring and control of their operation
- The system supports more and more task benefits to end any job at the high quality and little time.

1.3 Hebron Electric Power Company (HEPCO)

Hebron Electric Power Company (HEPCO) is a private shareholding company that was established in 2002 HEPCO. It is responsible for distributing electrical energy in the Hebron & Halhul regions. The total number of consumers is approximately 27000.

The primary concern of HEPCO in delivering electricity to its customers in its concession area is to ensure a safe, efficient and reliable supply of electricity.

Since the company is under construction in terms of software applications, managements, and data collection, it faces a lot of difficulties in sending customers bills in time, having accurate statistics about customers and getting customers' reports and comments

So the Company likes to enable customers to make payments on line using different methods, in order to accelerate the process of delivering bills and paying them and keep the customers in a closer contact with the company; receiving their comments and improving its services [5].

Chapter

Two

System Specification

2.1 Introduction

After several meeting with the company administration and understanding the specification of the system requirements, we demonstrate here that specifications which include: project objectives, benefits, functional and non-functional requirements, constrains and validations, development requirements, feasibility study, risks evaluation, project plan and scheduling, and finally summary and recommendations.

2.2 Project Main Objectives

These are the main objectives of the project:

- Enable customers to view the electricity bill online.
- Enable customers to retrieve up to six of his bills and payments.
- Enable customers to get detailed electricity consumption report.
- Enable customers to make payment online
- Enable HEPCO to put announcements for the customers.
- Enable customers to send any comment or complains.

2.3 Project Benefits

- Benefits for the company:

- Facilitating the communication process between the company and its costumers.
- Reducing the number of employees needed for delivering bills and dealing with costumers.
- Reducing company expenses such as papers, pencils, stamps.
- Allowing the company to get costumers' feedbacks frequently in order to improve its work.
- Allowing the company to get benefits the new trends of technology

- Benefits for the development team:

- Improving our abilities to develop a web-based database system.
- Increasing our works opportunities
- Granting us the soul of working as a group

-Society Benefits:

- Accelerating bills' delivering and payment
- Allowing costumers to report their problems easier.

- Giving the opportunity to more costumers to subscribe to get company services.

2.4 Functional Requirements

This system should provide the following functions:

2.4.1 Public User Functional Requirements

The following functions should be provided for any one uses the system:

1. Registration for Online Service: The public user should get an ID and a password to get online services.

2. Login: The public user should be able to log in the system to get e-services.

3. Viewing Announcements and Advertisements: The public user should be informed of the company announcements.

4. Viewing Help: The public user should get help.

5. Viewing FAQ: The public user should be able to view the frequently asked questions about the system.

6. Reporting Comments: The public user should be able to report comments and feedbacks.

7. Viewing Company Contact information: The public user should be able to view the company's address, phone #, fax #, and location.

2.4.2 Online Registered User Functional Requirements

The following functions should be provided for HEPCO customers (who have been registered online):

1. Logout: The customer should be able to log out the system.

2. Changing Login Password: The customer should be able to change password

3. Viewing Announcements and Advertisements: The customer should be informed of the company announcements.

4. Viewing Help: The customer should get help.

5. Viewing FAQ: the customer should be able to view the frequently asked questions about the system.

6. Reporting Comments: The customers should be able to report their comments and feedbacks.

7. Viewing Company Contact information: The customer should be able to view the company's address, phone #, fax #, and location.

8. Viewing Bills: The customers should be able to view, save and print their bills.

9. Viewing Consumption Report: The customers should be able to view their consumption reports.

10. Viewing Bills Status: The customer should be able to view bills status for 12 months.

11. Prompting to Bank Website: The customer should be prompted to bank website to pay his bills.

12. Bank Login: The customer should be able to have a bank ID and password.

13. Bank Help: The customer should get a help from the bank website.

14. Online Payment Using Bank Website: The customer should be able to pay bills by bank website

15. Bank Confirmation: The customer should be confirmed by the bank that the bill was paid.

16. Online Payment Using Credit Card: The customer should be able to pay the bill online using credit card.

2.4.3 Administrator Functional Requirements

The following functions should be provided for the system administrator:

1. Logout: The administrator should be able to log out the system

2. Changing login password: The administrator should be able to change password.

3. Viewing Comments: The administrator should be able to view comments sent by customers and users.

4. FAQ Maintenance: The administrator should be able to add, modify and delete FAQ's (Frequently Asked Questions).

5. Reports Generation: The administrator should be able to generate reports.

6. Announcements Maintenance: The administrator should be able to add, modify and delete announcements.

7. Bill generation: the administrator should be able to generate bills.

2.4.4 Transparent Functional Requirements

There are two transparent functions:

1. The bank should generate XML transactions to HEPCO.
2. HEPCO should read the XML files and update its database.

2.5 Non-Functional Requirements

Non Functional Requirements have also been called the 'ilities' because they are most simply expressed like this:

- Usability
- Reliability
- Interoperability
- Scalability
- Security

There are other ones:

- Time to market
- Cost
- Speed

Functional Requirements are either met or not met. Non-functional requirements tend to be things that you can measure.

It helps if you can translate each one into some measurable property of the final product otherwise the client can always claim that the product is not good enough.

Product Requirements:

- Reliability: the system is to be reliable and can be used in the environment with no problems.
- Portability: the system is to be accessed every where and to be portal to any desired place or environment.
- Usability: the system is to be easy to use since the software will be used by different cultured levels of people.
- The speed of the system must be acceptable, which means that the users or employees can input or retrieve information quickly.
- The menus must be clear.
- Buttons are to have names and supported by tips.

Process Requirements:

- The system is to be built with asp.net ,it is the most modern developing for the web from Microsoft, even there is similarity between asp.net and asp (active server Pages),but asp.net is redesigned upon .NET framework.
- Should be submitted at the end of the current course, January/2005.
- The database in the system is to be built using SQL server 2000.
- A sufficient hardware is needed.

External Requirements:

- Legislative requirements, the system is to be under the rules of the company and the lows of the country, and be sure no one gets hurt by it.
- Ethical requirements, is to save the personal data about the customer who registered in the system, and allow the user to access his own data only
- The system will be in the internet server, so it must be easy to use as the users do not have a high background in the system.

2.6 Project's Constraints

In our software some constrains are to be considered:

- Changing password every 6 months to insure the security.
- The characters' number of the login id and password is to be not less than 6 characters.
- The time of submitting the software is not to exceed the time that we determine with the company.
- The software must be applied in HEPCO only.
- In case of any problem met in the system the software developers must be informed.
- The constrains related to any delay in submitting the time of the project, will be negotiated later with HEPCO.
- The website should follow the rules of the company in terms of the data that can be showed in the web.
- In order to use the HEPCO E-Bill Service you must be a customer of HEPCO.
- Each member can't login the system without ID and password.
- If the user does not use the password for 6 months, he will lose the E-bill service.
- The user can see only the last 6 months.
- The bill will be displayed on the screen with some details about consuming
- The user must have bank ID and password in order to pay his bill
- The data base of the bank and HEPCO webs are separated from the real database.

2.7 Allocation and Trade offs

Table 2.1 Allocation and Trade offs

Requirements	Allocation
Registration	Software
Login / Logout	Software
Viewing announcements and advertisements	Software
Viewing help	Software
Viewing FAQ	software
Reporting comments	Software
Viewing company address	Software
Bills Generator	Software
Viewing bills status	Software
Viewing consumption Report	Software
Online Payment Using Bank	Software
Bank Login	Software
Bank Help	Software
Bank Confirmation	Software
Online Payment Using Credit Card	Software
Viewing Comments (for administrator)	Software
FAQ maintenance(for administrator)	Software
Generating reports (for administrator)	Software
The bank should generate xml transactions to HEPCO	Software
HEPCO should read the xml files and update its database	Software

2.8 Development Requirements and cost

2.8.1 Hardware

Table 2.2 Hardware Development Resources

Hardware Component	Cost
1 PC with Pentium 4, 2400 MHz 40GB HDD, 128 MHZ RAM, monitor 17", keyboard and mouse.	$1300
Total	$1300

2.8.2 Software

Table 2.3 Software Development Resources

Software Component	Cost
Windows XP professional	$100
Microsoft Visual Studio.net	$700
DBMS (MS SQL Server 2000)	$400
Adobe Photoshop 6	$100
Total	$1300

2.8.3 Human

Table 2.4 Human Development Resources

Work Team	Cost
Tahani Aldarabee'	*
Hiyam Badr	*
Abeer Aljabary	*

* No money compensations due to educational purposes

2.8.4 Other Resources

Table 2.5 Other Development Resources

Resources	Cost
Books	$500
Internet	$100
Transportations	$100
Total	$700

2.8.5 Total Resources:

Table 2.6 Total Development Resources

Total Development Resources	Total Cost
Hardware	$1300
Software	$1300
Human	None
Others	$700
Total	$3300

2.9 Costs-Benefit Analysis

There are tangible and non-tangible benefits for the system:

2.9.1 Tangible benefits

In order to calculate the cost and the profits we need to focus on the fixed and variable costs that arise from this project. On the other hand, we need to consider fixed and variable profits. Both costs and profits depend on the number of the system's users (clients). The total number of customers in HEPCO is 27,000.

The possible costs and profits for the coming 5 years have been estimated. The profits were considered depending on the possibility of cutting down expenses by discharging employees; when an employee is no longer needed, because of the increasing number of the clients who use the web service that provided by this project.

The costs are divided into fixed and variable costs. The fixed costs are defined by the price of the software, while the variable costs are related to the price of advertising about the new service and the hosting web server, including maintenance, and the bank commission for each bill that paid by bank via web.

With regard to the price of the software, it is considered to be part of the costs during the first 5 years.

- **Costs:**

The costs that will appear when the software is installed are as following:

1. Fixed costs: The price of the software which is ($4000); will be considered in calculating as $1000 for each year in the first 4 years.
2. Variable costs: These costs are grouped by
   - Web server hosting, $100 each year (including maintenance).
   - Bank commission, $0.1 for each paid bill.
   - Advertising about the new service, $100 each year

- **Benefits**

The benefits in the project are totally depending on the number of clients who use this service.

The main benefit is discharging some employee: since that each customer comes to the company will take 5 minutes from the employee's time, but when 1000 customers pay by the bill through the web this will keep 5*1000 minutes which are 83 hours which are 10 work days, and if the employee salary is $30 each day this give $311.25 will be considered as benefits.

Table 2.7a the detailed costs for each year

Costs/year		Hosting the project in server	Advertising about the new service	S.W price $4000	Bank commission $0.1/bill
year	Customers #				
1	500	$100	$100	$1000	$50
2	1500	$100	$100	$1000	$150
3	2500	$100	$100	$1000	$250
4	3500	$100	$100	$1000	$350
5	5000	$100	$100	$0	$500

Table 2.7b The total benefits for each year

Year /benefits		Discharging employee
year	Customers #	
1	500	$ 200
2	1500	$460
3	2500	$780
4	3500	$1093
5	5000	$1562

The total cost is calculated by subtracting the benefits from the cost for each year and the results are shown in table 2.8.

Table 2.8 total costs benefits

Net\year	1	2	3	4	5
Costs-benefits	$1050	$596	$668	$457	$100

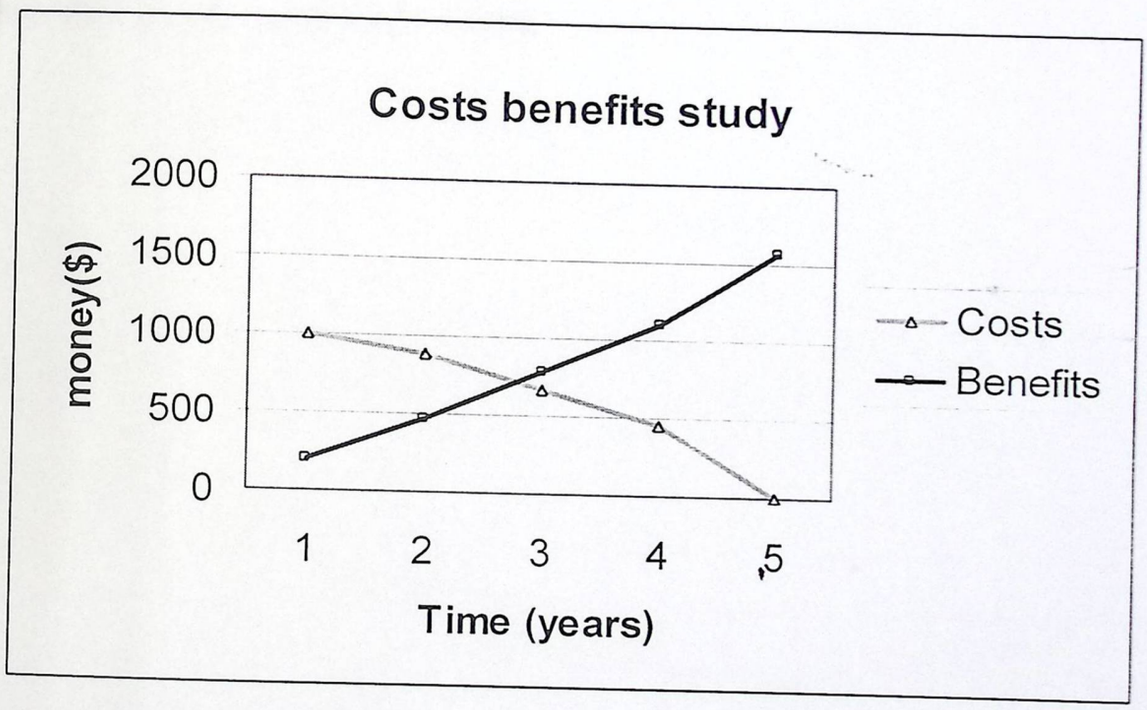


Figure 2.1 Cost benefits analysis

We can see that the costs are decreasing by increasing the benefits as shown in figure 2.1.

2.9.2 Non-tangible benefits

- Saving time by accelerating billing and payment processes.
- Allowing HEPCO to follow new technologies.
- Facilitating dealing with customers.

2.10 Feasibility Study

2.10.1 Economic Feasibility

The total cost of development resource is $3300 as mentioned in section 2.8 and since we are able to have all of these resources the system is economically feasible.

2.10.2 Technical feasibility

Since we have all the development resources mentioned in section 2.8 we have the ability to develop this system, so it is technically feasible.

2.10.3 Legal Feasibility

The system follows the regulations and the rules of the company and obeys the law of the country, so it's legally feasible.

2.12 Project Plan and Scheduling

Table 2.10 Project Activities and Their Dependencies

Activities(tasks)	symbol	Dependences
Determining project subject and writing the introduction	T1	-
System specification	T2	-
Software Requirement specification	T3	T2
Design	T4	T3,T2
Coding and implementation	T5	T4,T2
Testing	T6	T5
Maintenance Plan	T7	T9
Writing the documentation	T8	T2, T3,T4, T4, T5, T6, T7, T8, T9,

Table 2.11 Project Scheduling

Week / Task	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
T1																
T2																
T3																
T4																
T5																
T6																
T7																
T8																

2.13 Summary and Recommendation

According to the mentioned system specification which is approved by the company and the supervisor of the project, the work team decided to continue with the software requirements specification phase.

Chapter

Three

Software Requirements Specification

3.1 Introduction:

This chapter will cover the system requirements in more details and in more technical terms, it describes the following

- Functional description of the system, in which all the supported functions and services will be identified.
- Behavioral models in which the behavior of the system will be modeled by using a data flow diagram.
- Data dictionary in which a complete description of each system entity will be provided, with its name, type, and description.
- Database requirements and the required features of the database will be identified.
- Database dictionary: a detailed description of the contents of each table in the database

3.2 Functional Detail Description

Here is a detailed description for systems functional requirements:

3.2.1 User Functional Requirement

Function: Registration for online service.

Description The public user should get an ID and a password to get online services.

Inputs: ID, Password, identity card number, email

Source: User (web form)

Outputs: Registration confirmation.

Destination: HEPCO database, screen

Pre-condition: The user can't login to get e-services.

Post-condition: The user can login and get e-services.

Side effect: none.

Validation: 1. All inputs must be filled in .

2. The ID must equal HEPCO customer no.
3. The password must be at least 6 characters
4. ID mustn't equal the password
5. Password mustn't include special characters.

Function: Login

Description: The registered user and administrator should be able to log in the system to use e-services.

Inputs: ID, Password.

Source: registered user or administrator (web form)

Outputs: the main web page of e-services

Destination: screen

Pre-condition: The user must visit HEPCO e-services website, and must have been registered online.

Post-condition: The user can use e-services.

Side effect: none

Validation: 1. ID and password must be filled in.

2. ID and password must have been registered in HEPCO database.

Function: Logout

Description The registered user and administrator should be able to log out the system

Inputs: selecting to log out

Source: Registered user or administrator (web form)

Outputs: HEPCO main website.

Destination: screen.

Pre-condition: Registered user or administrator must have been logged in HEPCO e-service website.

Post-condition: Registered user or administrator should return to the main page in the website.

Side effect: none

Validation : Registered user or administrator must have already logged in.

Function: Changing login password

Description The registered user and the administrator should be able to change login password.

Input: ID, old password , new password and password confirmation

Source: registered user or administrator (web form).

Output: changing password confirmation.

Destination: HEPCO database.

Pre-condition: the registered user or the administrator must have been logged in HEPCO e-services.

Post-condition: the registered user or the administrator must logout.

Side effect: none

Validation: 1. The new password must be at least 6 characters long and doesn't include special characters, and mustn't equal the user ID.

2. New password= password confirmation.

Function: Viewing announcements and advertisements.

Description: The user (public or registered) should be informed of the company announcements.

Inputs: Announcements and advertisements

Source: HEPCO database

Outputs: Announcements and advertisements list.

Destination: Screen.

Pre-condition: The user must visit HEPCO e-services website, and request to view announcements and advertisements.

Post-condition: none

Side effect: none

Validation: none

Function: Viewing Help

Description: The user should get help.

Inputs: Help information.

Source: -

Outputs: help contents

Destination: Screen.

Pre-condition: The user must visit HEPCO e-services website, and request for online help

Post-condition: none

Side effect: none

Validation: none

Function: Viewing FAQ's

Description The user should be able to view the frequently asked questions about the system.

Inputs: FAQ's

Source: HEPCO database

Outputs: FAQ's list.

Destination: Screen.

Pre-condition: The user must visit HEPCO e-services website, and request to view FAQ's

Post-condition: none

Side effect: none

Validation: none

Function: Reporting comments

Description The users should be able to report their comments and feedbacks.

Inputs: Comment, questions and user name and email, and HEPCO subscription number.

Source: User

Outputs: Confirmation screen.

Destination: comments to HEPCO database and confirmation on screen

Pre-condition: The user must visit HEPCO e-services website, and request to write comments and questions.

Post-condition: None

Side effect: None

Validation: 1. Message mustn't be blank

2. The user must provide his email and his name and HEPCO subscription number.

Function: Viewing company contact information

Description The user should be able to view the company's address, phone #, fax #, and location.

Inputs: -

Source: -

Outputs: company's address, phone #, fax #, and location.

Destination: Screen.

Pre-condition: The user must visit HEPCO e-services website, and request to view company address.

Post-condition: none

Side effect: none

Validation: none

Function: Viewing bills

Description The registered user should be able to view, save and print bills.

Inputs: Bills information, bill date

Source: HEPCO database, web form

Outputs: Bill is displayed, printed or saved.

Destination: screen , printer, or computer hard desk

Pre-condition: The customer must have been logged in HEPCO e-services website, and request to view his bill.

Post-condition: None

Side effect: None

Validation: bill date must be less than or equal the current date

Function: Viewing Consumption Report

Description The registered user should be able to view their consumption reports.

Inputs: Consumption report, service ID.

Source: HEPCO database, user(web form)

Outputs: Consumption report for the last 6 months is viewed.

Destination: screen.

Pre-condition: The registered user must have been logged in HEPCO e-services website, and request to view his consumption report.

Post-condition: None

Side effect: None

Validation: none

Function: Viewing Bills Status

Description The registered user should be able to view bills status for 12 months.

Inputs: bills information, service ID

Source: HEPCO database, user(web form)

Outputs: Bills Status for the last 6 months is viewed.

Destination: screen.

Pre-condition: The registered user must have been logged in HEPCO e-services website, and request to view his Bills Status.

Post-condition: None

Side effect: None

Validation: none.

Function: Prompting to Bank Website

Description The registered user should be prompted to bank website to pay his bills.

Inputs: Select to go to bank website

Source: Registered user

Outputs: Bank website

Destination: screen.

Pre-condition: The registered user must have been logged in HEPCO e-services website and have a bank account.

Post-condition: The registered user should logout.

Side effect: None

Validation: None

Function: Bank Login

Description The registered user should be able to log in the bank website.

Inputs: ID and password

Source: Registered user

Outputs: bank e-payment screen

Destination: bank database and screen

Pre-condition: The registered user must have been logged in HEPCO e-services website, and select to pay by bank, and must have and ID and password which is provided manually by bank.

Post-condition: registered user should be able to pay his bill.

Side effect: None

Validation: ID and Password must be provided and must be valid.

Function: Bank Help

Description the registered user should get a help from the bank website.

Inputs: Help information

Source: -

Outputs: help viewed.

Destination: screen

Pre-condition: The registered user must visit the bank website, and request for help.

Post-condition: None

Side effect: None

Validation: None.

Function: Bank Login

Description The registered user should be able to log in the bank website.

Inputs: ID and password

Source: Registered user

Outputs: bank e-payment screen

Destination: bank database and screen

Pre-condition: The registered user must have been logged in HEPCO e-services website, and select to pay by bank, and must have and ID and password which is provided manually by bank.

Post-condition: registered user should be able to pay his bill.

Side effect: None

Validation: ID and Password must be provided and must be valid.

Function: Bank Help

Description the registered user should get a help from the bank website.

Inputs: Help information

Source: -

Outputs: help viewed.

Destination: screen

Pre-condition: The registered user must visit the bank website, and request for help.

Post-condition: None

Side effect: None

Validation: None.

Function: Bank Confirmation.

Description: The registered user should be confirmed by the bank that the bill was paid.

Inputs: none.

Source: bank database

Outputs: message tells the registered user that the bill was paid.

Destination: screen.

Pre-condition: registered user logged in bank site and had paid.

Post-condition: None

Side effect: None

Validation : None

3.2.2 Administrator Functional Requirements

Function: Viewing Comments

Description: The administrator should be able to view comments sent by users.

Inputs: comments

Source: HEPCO database

Outputs: comments list and details.

Destination: screen

Pre-condition: The administrator should have been logged in HEPCO website as administrator and select to view comments.

Post-condition: none

Side effect: none

Validation: none

Function: FAQ Maintainace

Description The administrator should be able to add, delete and modify FAQ.

Inputs: Questions and answers.

Source: HEPCO database

Outputs: FAQ list

Destination: HEPCO database

Pre-condition: the administrator should have been logged in HEPCO website as administrator and select to maintain FAQ

Post-condition: none

Side effect: none

Validation: none

Function: Announcements Maintenance

Description: The administrator should be able to add, modify and delete announcements.

Inputs: announcements

Source: HEPCO database and administrator

Outputs: announcements list

Destination: HEPCO database

Pre-condition: the administrator should have been logged in HEPCO website as administrator and select to maintain announcements

Post-condition: none

Side effect: none

Validation: none

3.2.3 Transparent Functions

Function: Generating XML transaction

Description The bank should generate XML transactions to HEPCO.

Inputs: bank transactions

Source: bank database

Outputs: XML file

Destination: bank hard desk

Pre-condition: not reported

Post-condition: reported

Side effect: none

Validation: None

Function: Reading XML file

Description: HEPCO should read the XML files and update its database

Inputs: XML files

Source: Bank hard desk

Outputs: Confirmation.

Destination: HEPCO database

Pre-condition: The bank should have sent XML transactions to HEPCO

Post-condition: none

Side effect: none

Validation: None

3.3 Project Constraints Details

In our software some constrains are to be considered:

- Changing password every 6 months to insure the security, to do this the user should enter the id ,old password and new password
- In registration ID, password and identity card number must be filled in, ID must equal HEPCO customer no.
- Password must be at least 6 characters, ID mustn't equal the password, Both ID and password mustn't include special characters.
- The login ID and password must have been registered in HEPCO database.
- Both customer and administrator can't logout unless being already logged in.
- The user must provide his email and his name and HEPCO customer number in order to report his comments, and the message mustn't be blank, and the size of the message must be limited.
- In order to use the HEPCO E-Bill Service you must be a customer of HEPCO.
- In order to view the bill or consumption report, you should choose the month, that should be less than or equal the current month.
- If the user does not use the password for 6 months, he will lose the E-bill service.
- The user can see only the last 6 months' bills.
- The bill will be displayed on the screen with some details about consuming
- To pay the bill the user must login the bank with valid ID and password (provided from the bank) and the customer number must be filled in also, the registered user must have an account greater or equals the bill value, in order to perform the process of paying successfully.
- The constrains that are related to database are :

  1. The database related to HEPCO e-service web is separated from real HEPCO database.
  2. The database related to the bank web is separated from the real bank database.

- The time of submitting the software is not to exceed the time that we determine with the company.
- The software must be applied in HEPCO only.
- In case of any problem happens in the system the software developers must be informed.

- The constraints related to any delay in submitting the time of the project, will be negotiated later with HEPCO.
- The website should follow the rules of the company in terms of the data that can be showed in the web.

- The constraints related to any delay in submitting the time of the project, will be negotiated later with HEPCO.
- The website should follow the rules of the company in terms of the data that can be showed in the web.

3.4 System Data Flow Diagram

The overall description of data flow diagram is shown in the following figure:

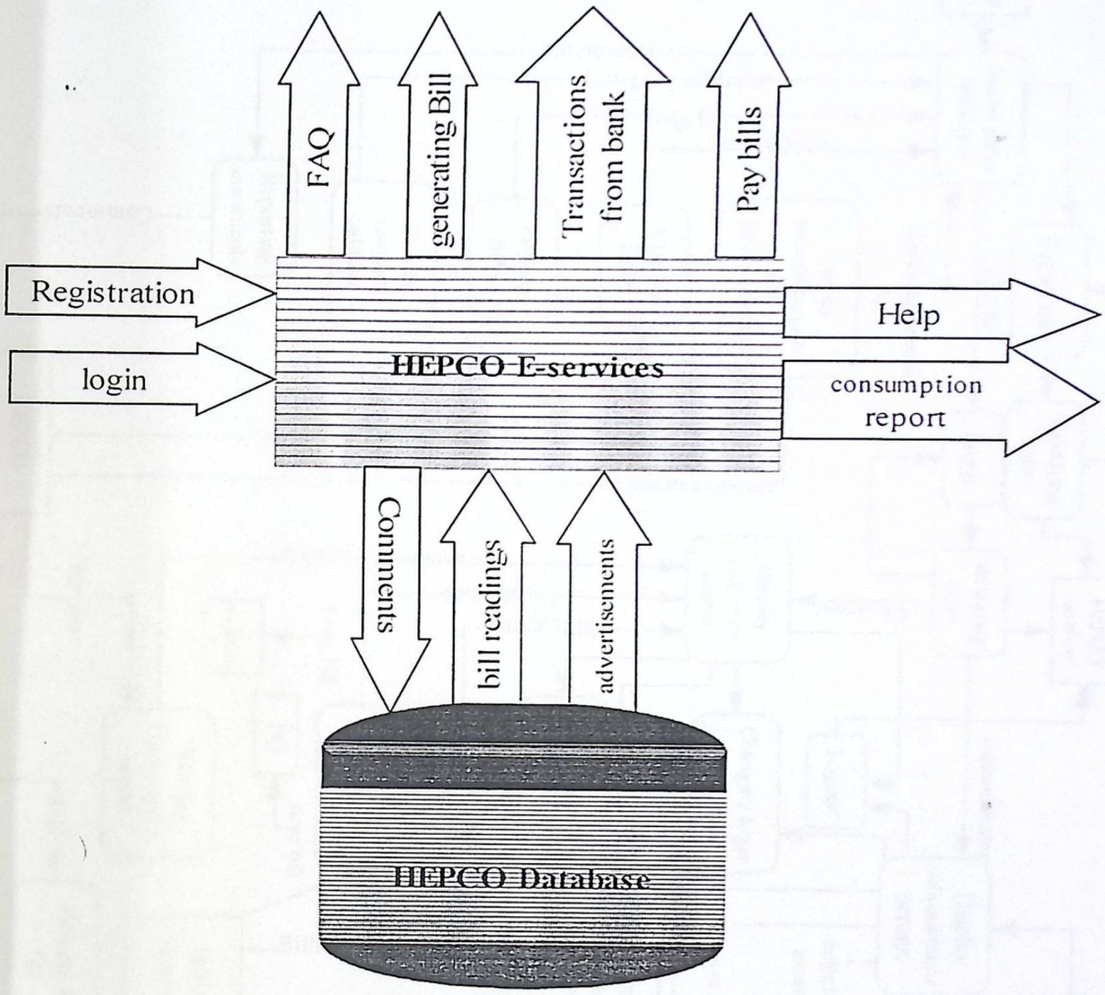
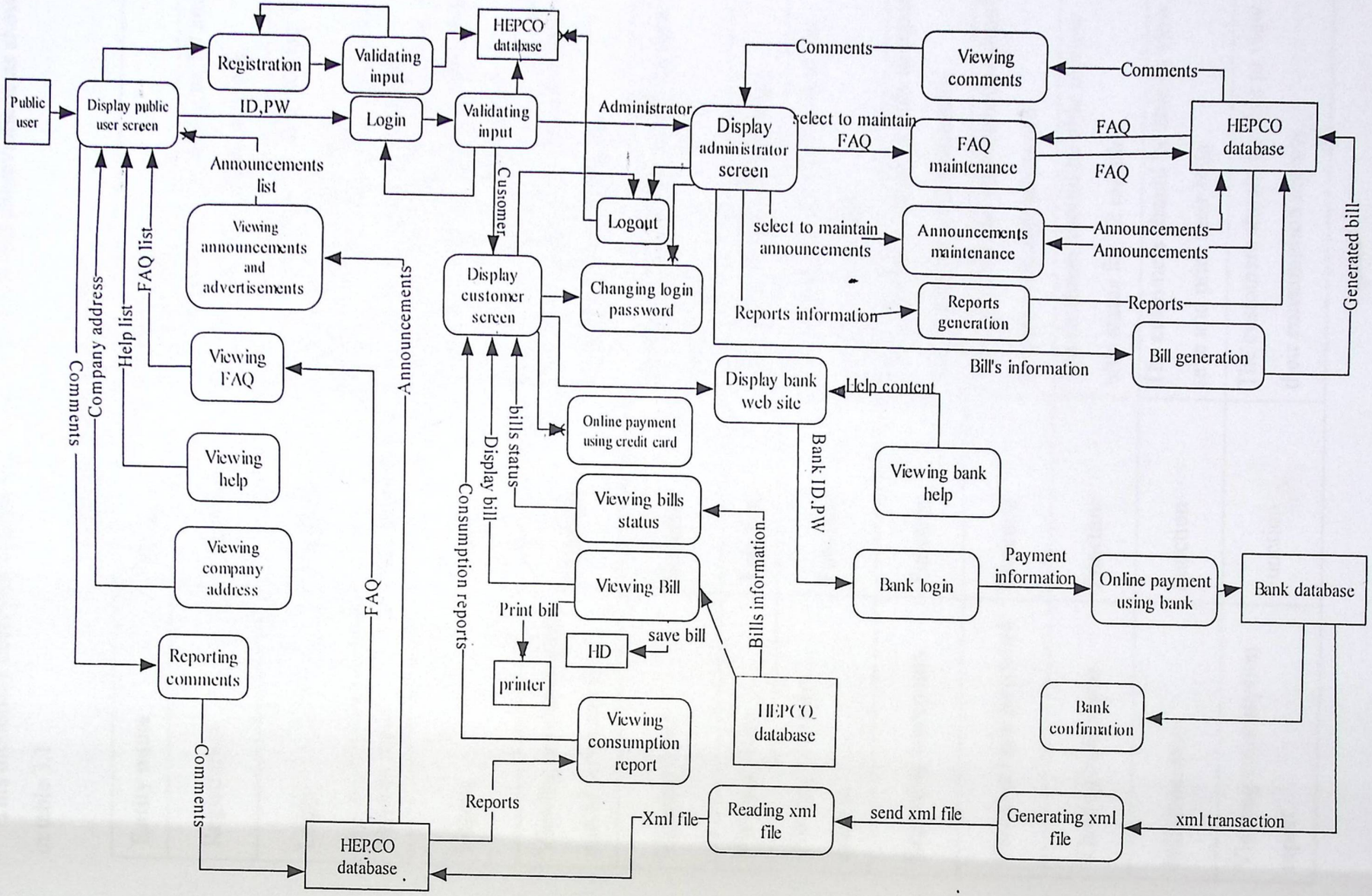


Figure 3.1 System top level diagram

The system data flow diagram is shown in the following figure



3.5 Data Dictionary

This dictionary describes system functions, procedures and abbreviations as shown in table 3.1

Table 3.1 Data Dictionary

Entity name	Type	Description
Registration	Function	The public user should get an ID and a password to get online services.
Login	Function	The user should be able to log in the system to get e-services.
Validate input	Stored procedure	Validate user ID and password
Logout	Function	The customer and administrator should be able to log out the system
Viewing announcements and advertisements	Function	The user should be informed of the company announcements.
Viewing FAQ	Function	The customer should be able to view the frequently
Viewing Help	Function	The customer should get help
Viewing company address	Function	The user should be able to view the company's address.
Reporting comments	Function	The customers should be able to report their comments and feedbacks.
Changing login password	Function	The customer and administrator should be able to change password
Viewing bills status	Function	The customer should be able to view bills status for 12 months.
Bill generator	Function	The customers should be able to view, save and print their bills.
Viewing consumption report	Function	The customers should be able to view their consumption reports.

Viewing comments	Function	The administrator should be able to view comments sent by customers and users.
FAQ maintenance	Function	The administrator should be able to add, modify and delete FAQ's (Frequently Asked Questions).
Announcements maintenance	Function	The administrator should be able to add, modify and delete announcements.
Reports generation	Function	The administrator should be able to generate reports.
Bills generations	Function	The administrator should be able generate bills.
Online payment using credit card	Function	The customer should be able to pay the bill online using credit card.
Online payment using bank	Function	The customer should be able to pay bills by bank website
Bank login	Function	The customer should be able to have a bank ID and password.
Viewing bank help	Function	The customer should get a help from the bank website
Bank confirmation	Function	The customer should be confirmed by the bank that the bill was paid.
Generating xml transaction	Function	The bank should generate XML transactions to HEPCO
Reading xml file	Function	HEPCO should read the XML files and update its database
FAQ	Abbreviation	Frequently Asked Questions
HEPCO	Abbreviation	Hebron Electric Power company
Xml	Abbreviation	Extensible Markup Language
E-bill and e-payment	Abbreviation	Electronic bill and electronic payment

3.6 Database Requirements

System database will include the following tables and fields:

1-HEPCO database tables

Customer information:

- Customer ID: A unique number of the customer.
- Customer name: The name of the customer (first name and last name).
- Address: The customer address.
- IDC number: Customer identification card number (haweyah).
- Phone: The phone number of the customer.
- Valid: A flag that determines whether the customer is still get service from the company or not.

Web account information

- Account ID: A unique number for each registered customer on the web.
- Customer ID: A unique number for each customer in HEPCO.
- Password: A unique password for customer and it is encrypted.
- Web Registration date: The date when the user registered.
- Status: Te customer is login or not.
- Number of login times: How many times did the customer login.
- Last login: The date of last login.
- User type: If the user is a customer or an administrator.
- E-mail: The email of the customer.

Bill Read:

- Bill number: The number of the generated bill (each bill has a unique number).
- Service number: The number of the each service in the company that a bill is generated to.
- Issue date: The date of bill generating.
- Due date: The date that the customer should pay before.
- Meter no: The number which identifies the meter that has been read.
- Current reading: Current reading of meter.
- Current reading date: The date of the current reading of meter.
- Paid: This field determines whether this bill is paid or not.
- Payment source: The source by which the customer chose to pay.
- Payment date: The date when the bill is paid.

Services:

- Service number: The number of the each service in the company that a bill is generated to.
- Customer ID: A unique number for each customer in HEPCO.
- Service Type ID: The type of the provided service.
- Establishment Date: The date of establishing this service.
- Service Beneficiary: The name of the person who gets benefits from this service (the service may be rented or owned by some one else).

Services types:

- Service Type ID: The type of the provided service.
- Type Name: The name of the service type (one Phase, two phases, or three phases).
- Type cost: The cost of each type per KW (kilo watts).

Advertisement information:

- Advertisement NO: A unique number for each advertisement.
- Advertisement title: The title of the advertisement.
- Advertisement description: Contents of the advertisement.
- Expiration Date: The date after which the advertisement mustn't be viewed
- Viewing date: The date on which the advertisement must be viewed
-

FAQ information:

- FAQ number: The number of FAQ.
- FAQ title: The question that is asked by the customer.
- FAQ description: The answer of the question.
- FAQ Modification date: The date when the FAQ is updated

Comments:

- Comment number: the number of the comment.
- Comment contents: the contents of the comment.
- User name: the name of the user who reported the comment.
- User email: The e-mail of the user who reported the comment.

2-Bank's database Tables

Bank Login:

- Customer bank ID: ID that identifies the customer which is given by bank
- Password: customer password which is given by bank (unique for each customer)

Bank accounts:

- Customer bank ID: ID that identifies the customer which is given by bank
- Bank account number: the number of the bank account for the customer
- Name: the name of the bank agent
- Account value: the amount of money that the customer has in his account.

Bank transaction

- Transaction Number: each transaction has a unique number.
- Customer ID: A unique number for each customer in HEPCO.
- Services number: The number that identifies the service.
- Bill number: number of the invoice
- Payment date: the date when the customer paid the bill.
- Money paid: amount of money that is paid.

3.6 Database Data Dictionary

This dictionary describes the database fields, their names, types, lengths and description.

Table 3.2 Database Dictionary

Field Name	Type	Length	Description.
Customer ID	Varchar	20	User login ID.
Customer password	Varchar	20	User login password.
Status	Boolean	1	Determines whether the customer is logged in or not .
Number of logins	Integer	4	Number of time the customer login the system
Last login	Date	8	The date of last login
User type	Char	1	Determines whether user or administrator
E-mail	Varhchar	50	Customer e-mail
Registration date	Date	8	Date of the customer registration.
First name	Varchar	20	The first name of the customer
Last Name	Varchar	20	The last name of the customer
Address	Varchar	100	The customer's address.
Phone number	Varchar	7	Customer phone number
Valid	Boolean	1	Determines whether the customer is still getting service from the company or not.
IDC Number	Varchar	9	Identity card number
Service Number	Integer	4	Number of the service.
Service type	Integer	4	The type of the provided service
Type name	Varchar	20	The name of the service type

Meter Number	Integer	4	Number of meter.
Current reading	Decimal	9	The current reading of meter.
Current reading date	Date	8	The date of The current reading.
Beneficiary	Varchar	50	The name of the customer who get services
Bill number	Decimal	9	Number of invoice
Paid	Boolean	1	Determines whether the bill is paid or not
Payment source	Varchar	50	The source of payment
Payment date	Date	8	The date on which the bill is paid
Issue date	Date	8	Date of invoice.
Due date	Date	8	.last date for payment.
FAQ number	Integer	4	Frequently ask question number
FAQ Description	Varchar	500	Frequently ask question contents
FAQ title	Varchar	50	Title of the FAQ
FAQ date	Date	8	The date when the FAQ is modified
Advertisement number	Integer	4	Advertisement number
Advertisement title	Varchar	50	The title of the advertisement
Advertisement description	Varchar	500	The content of the advertisement
Expiration Date	Date	8	The date after which the advertisement mustn't be viewed
Viewing date	Date	8	The date after which the advertisement should be viewed
Comment number	Integer	4	Comment number
Comment contents	Varchar	200	Comment contents
User name	Varchar	50	Name of user who reported the comment
User email	Varchar	100	Email of user who reported the comment
Customer bank ID	Varchar	50	ID that identifies the customer which is given by bank
Bank Password	Varchar	50	Customer password which is given by bank
Bank account number.	Varchar	50	The number of the bank account for the customer
Account value	Float	8	The amount of money in customer's account.
Transaction Number	Integer	4	Unique number for each bank transaction
Money paid	Float	8	Amount of money paid to bank

3.8 Summary and Recommendation

After understanding functional and database requirements in details which is approved by the supervisor the team will continue working in the next phase which is system design

Chapter

Four

Design

4.1 Introduction

This chapter will cover the design of the system including; including output/input design, database design, and functional design.

4.2 Output/Input Design

4.2.1 HEPCO screens

The following screens are the designed screens for HEPCO web based system.

Login screen that is shown in figure 4.1, is used when the user wants to login

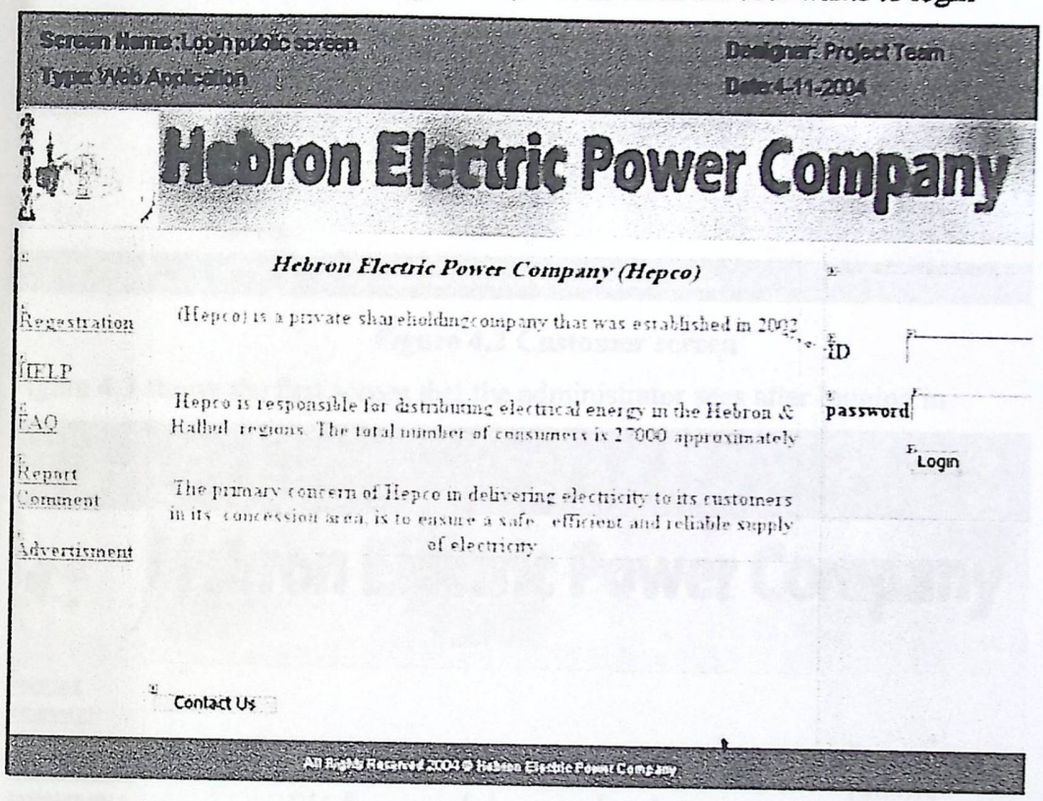


Figure 4.1 Login public screen

Figure 4.2 shows the first screen to the customer

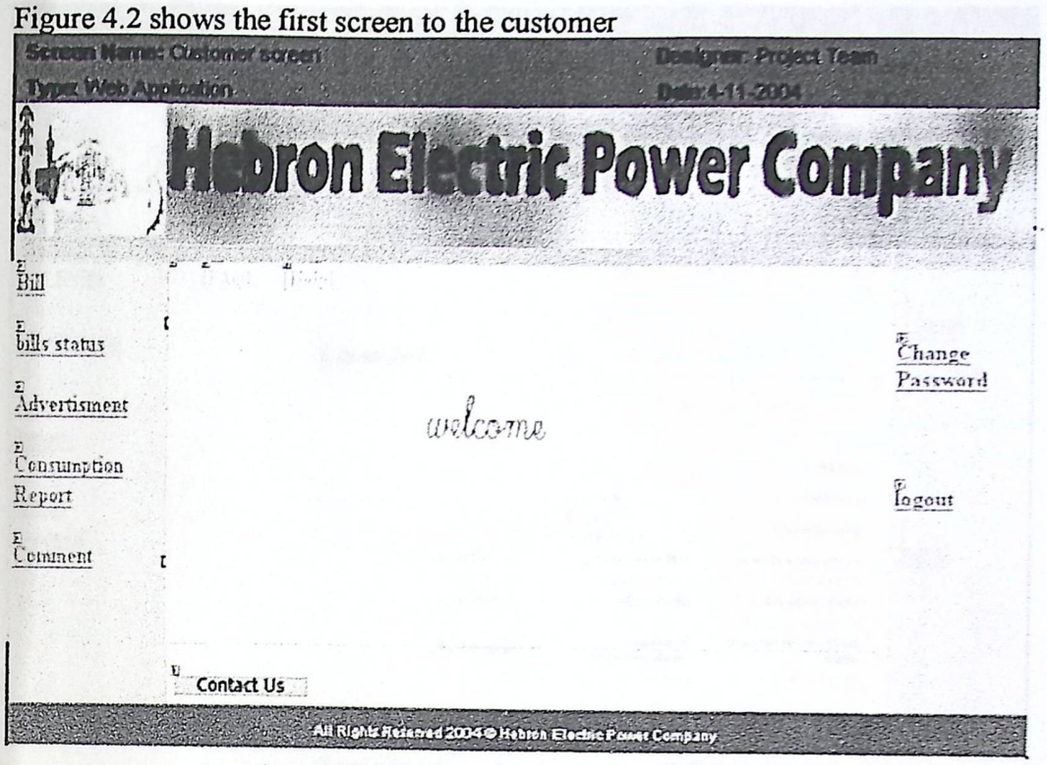


Figure 4.2 Customer screen

Figure 4.3 shows the first screen that the administrator sees after logging in

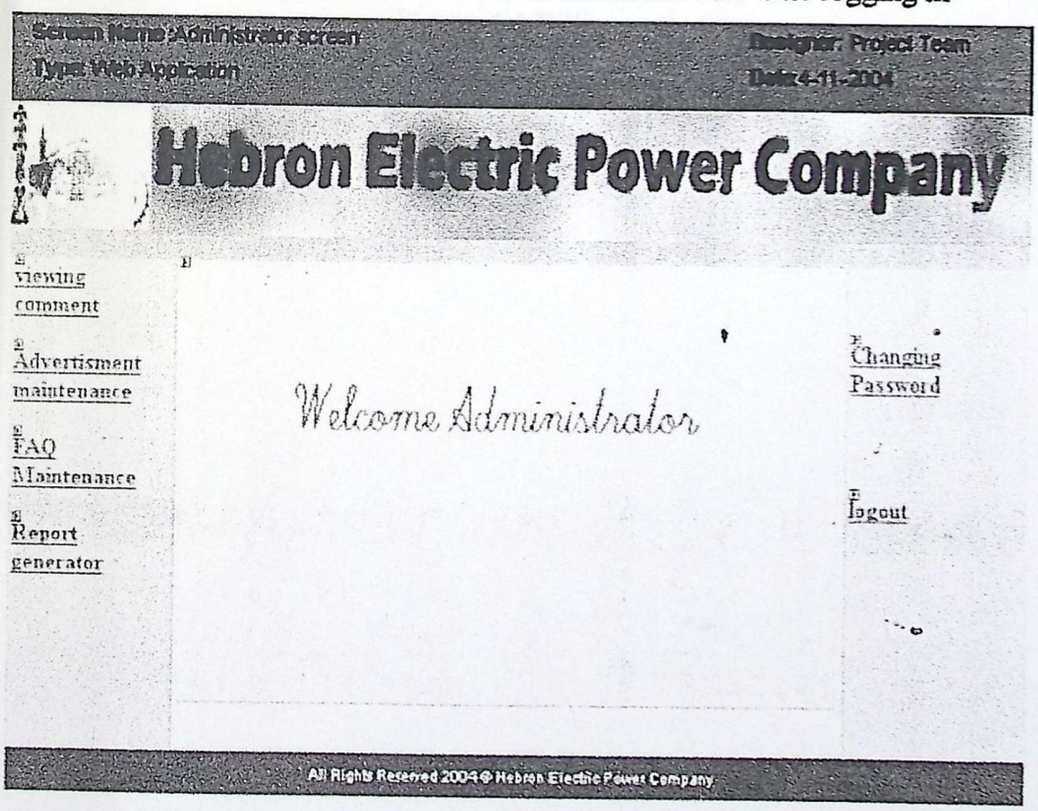


Figure 4.3 Administrator screen

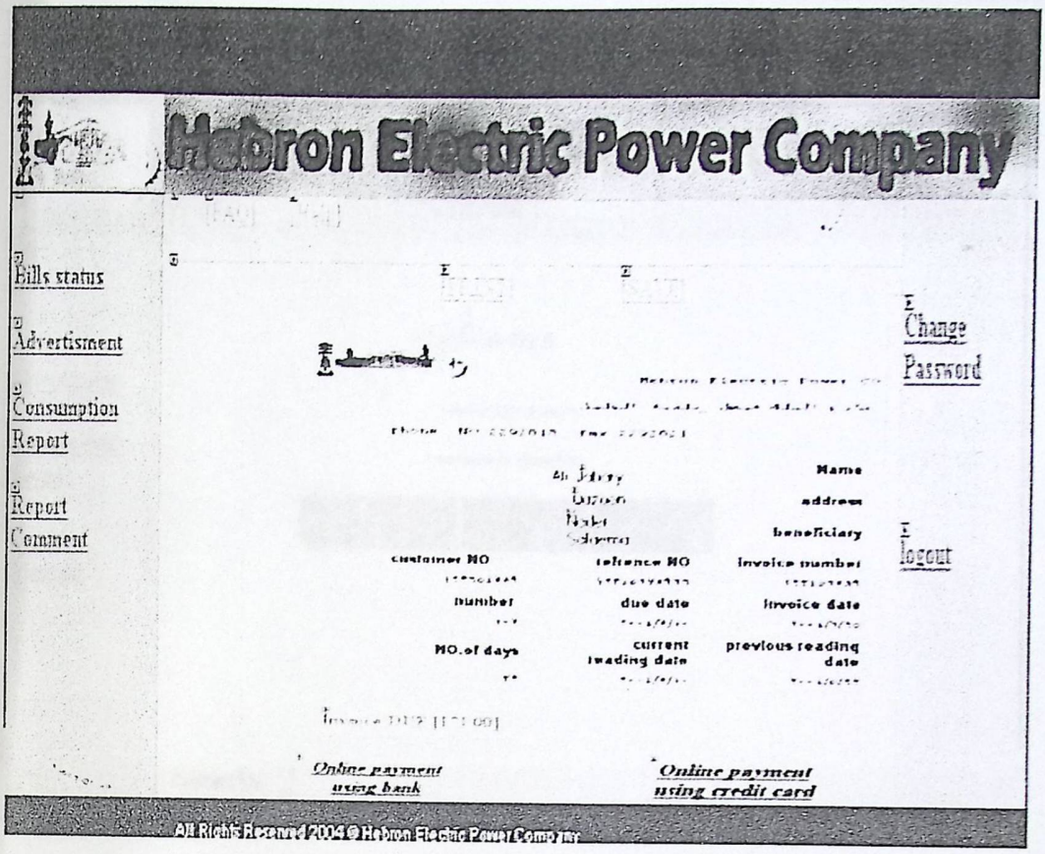


Figure 4.4 Bill generator

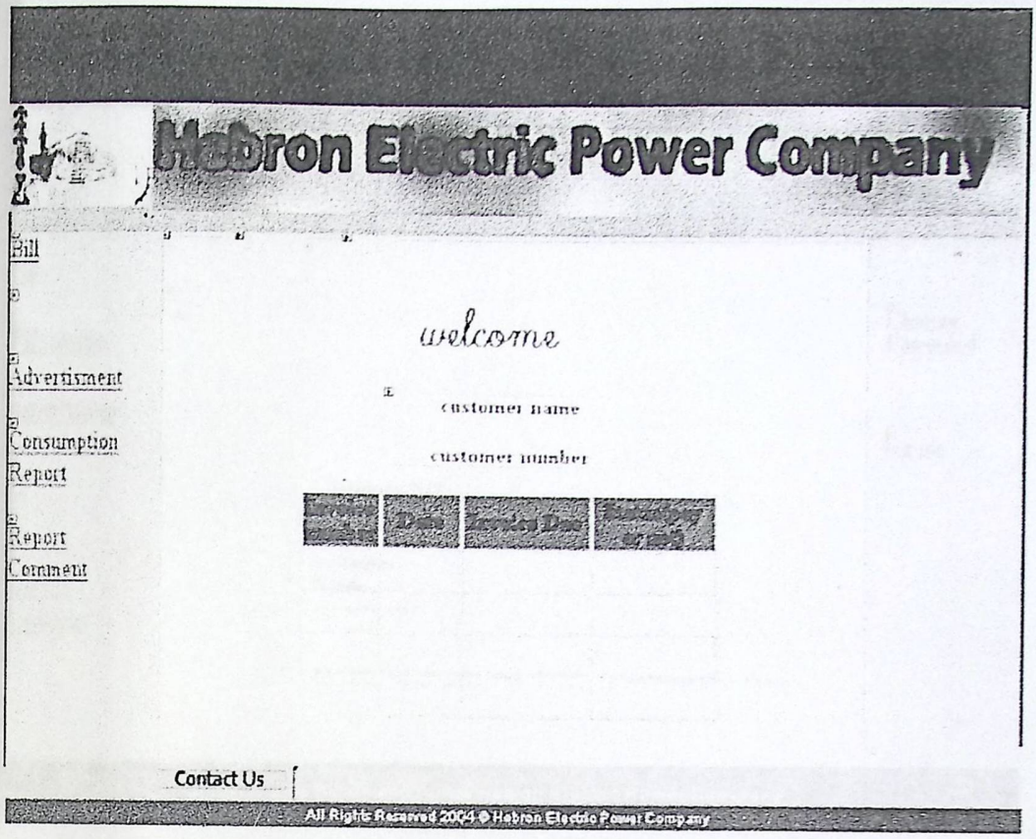


Figure 4.5 viewing bill statuses

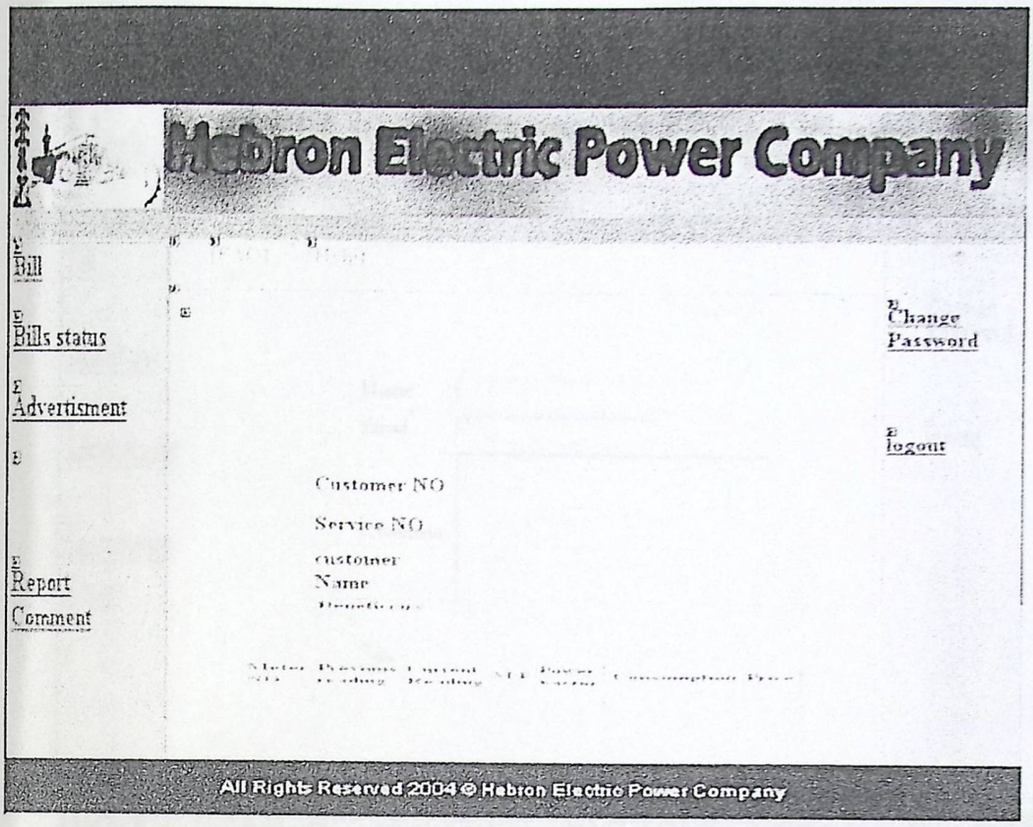


Figure 4.6 Consumption report

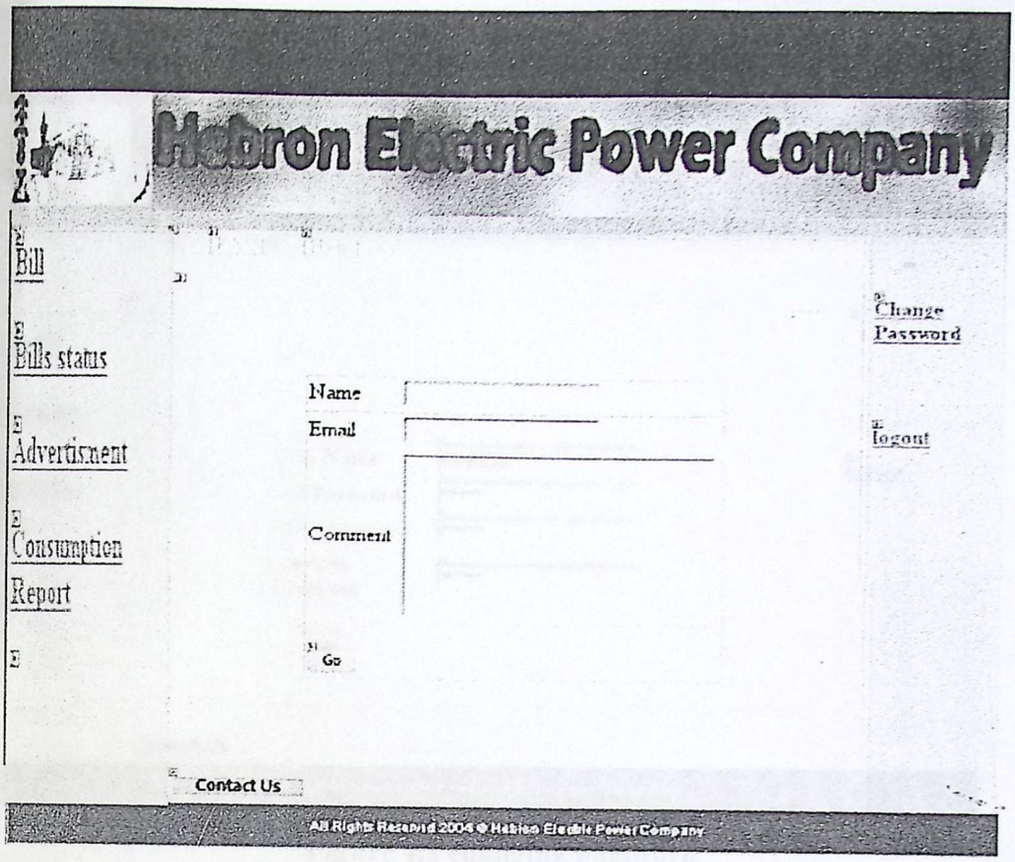


Figure 4.7 comments' reporting

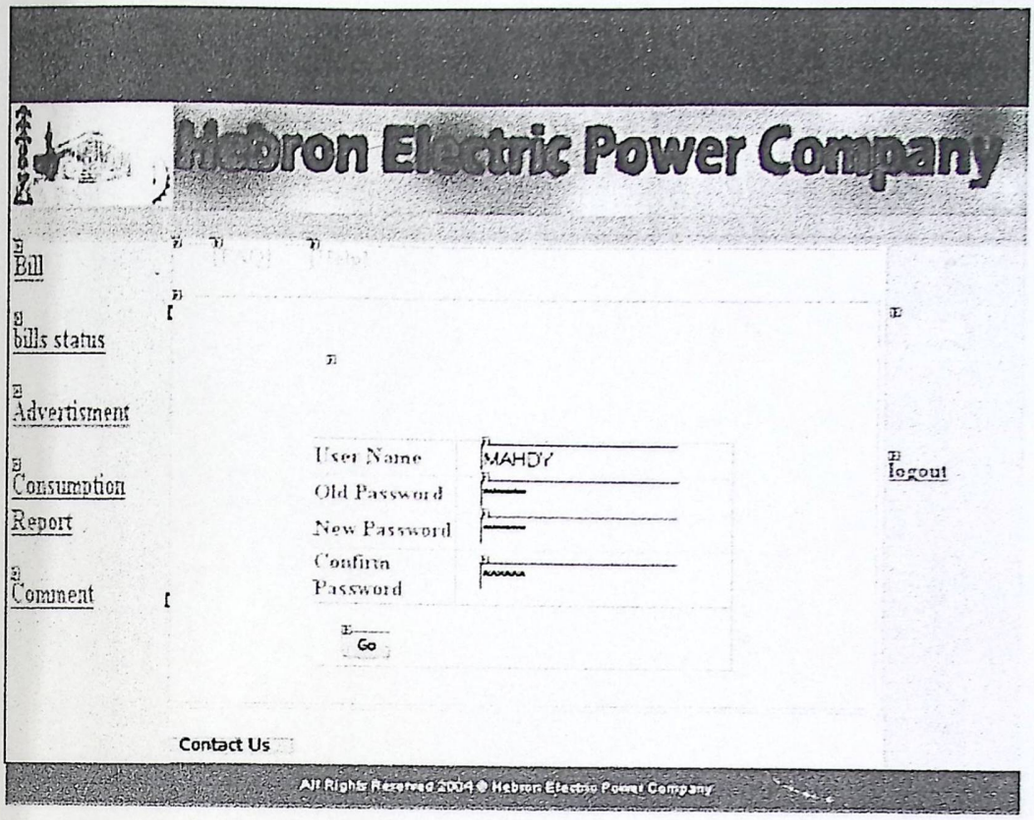


Figure 4.8 changing password

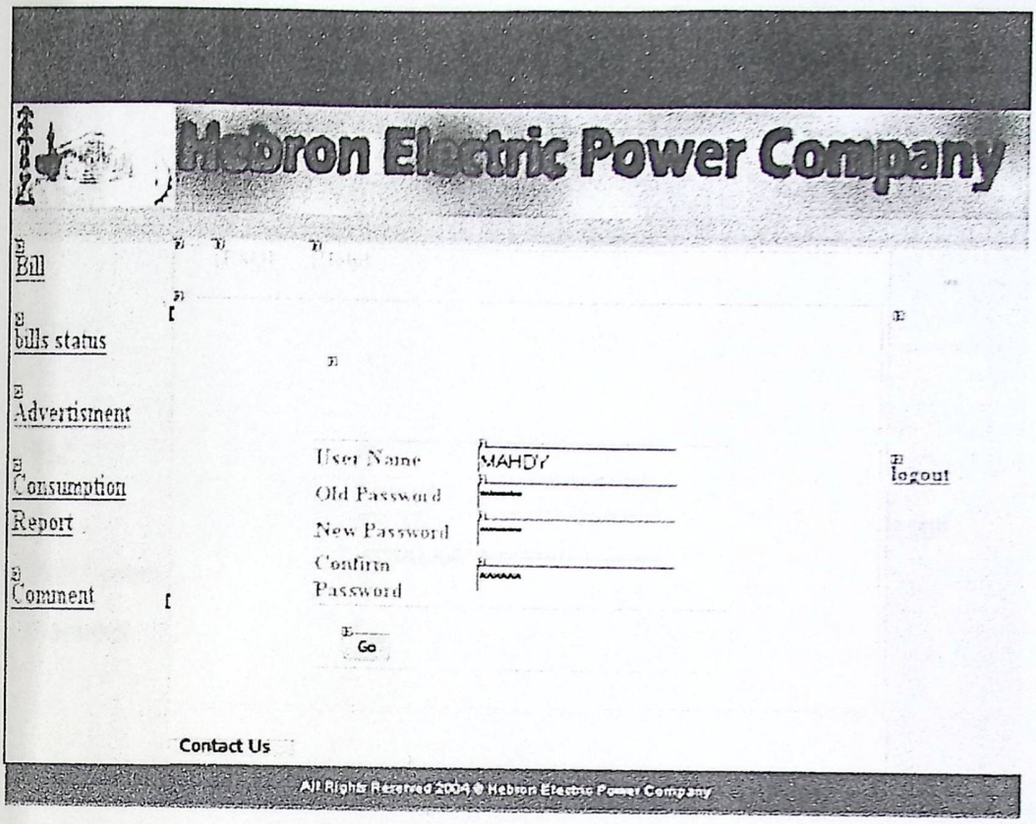


Figure 4.8 changing password

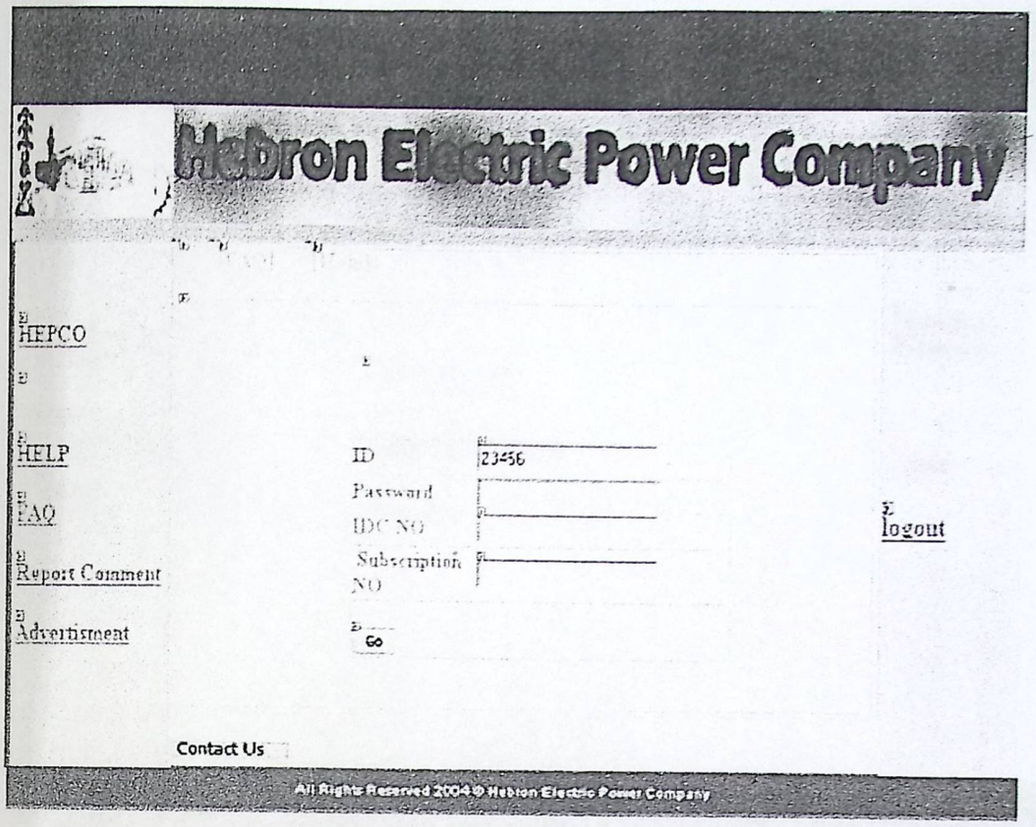


Figure 4.9 Registration

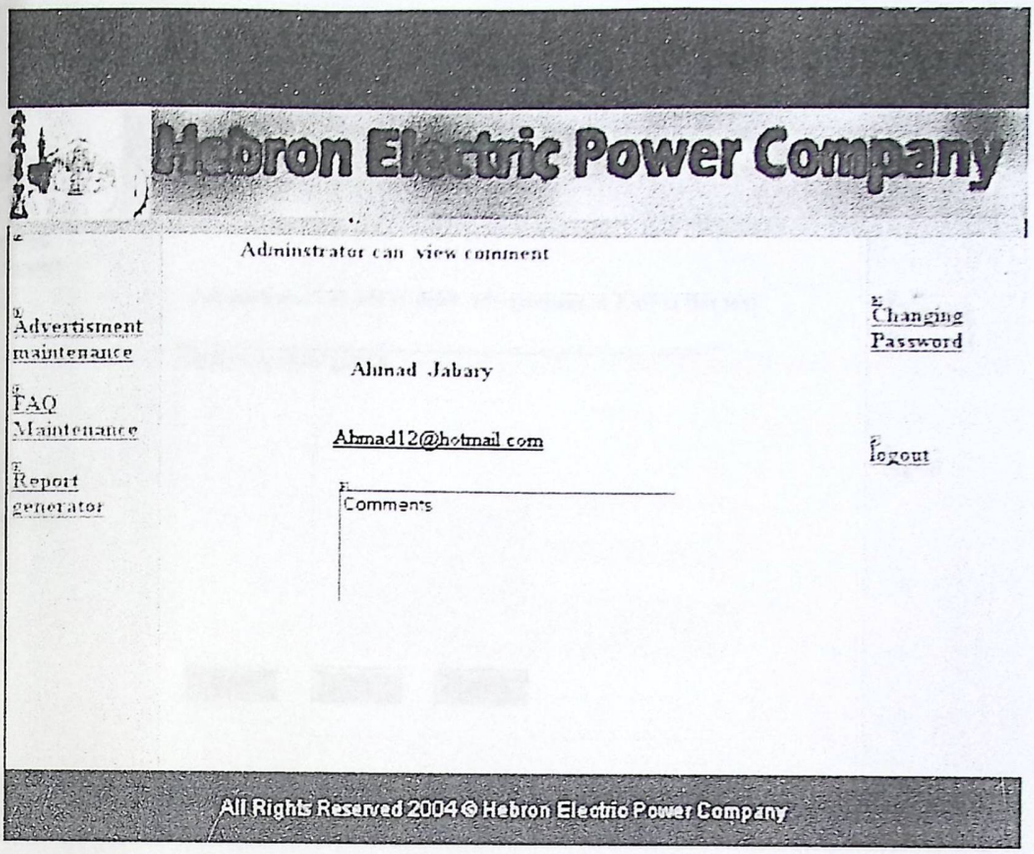


Figure 4.10 Consumption report

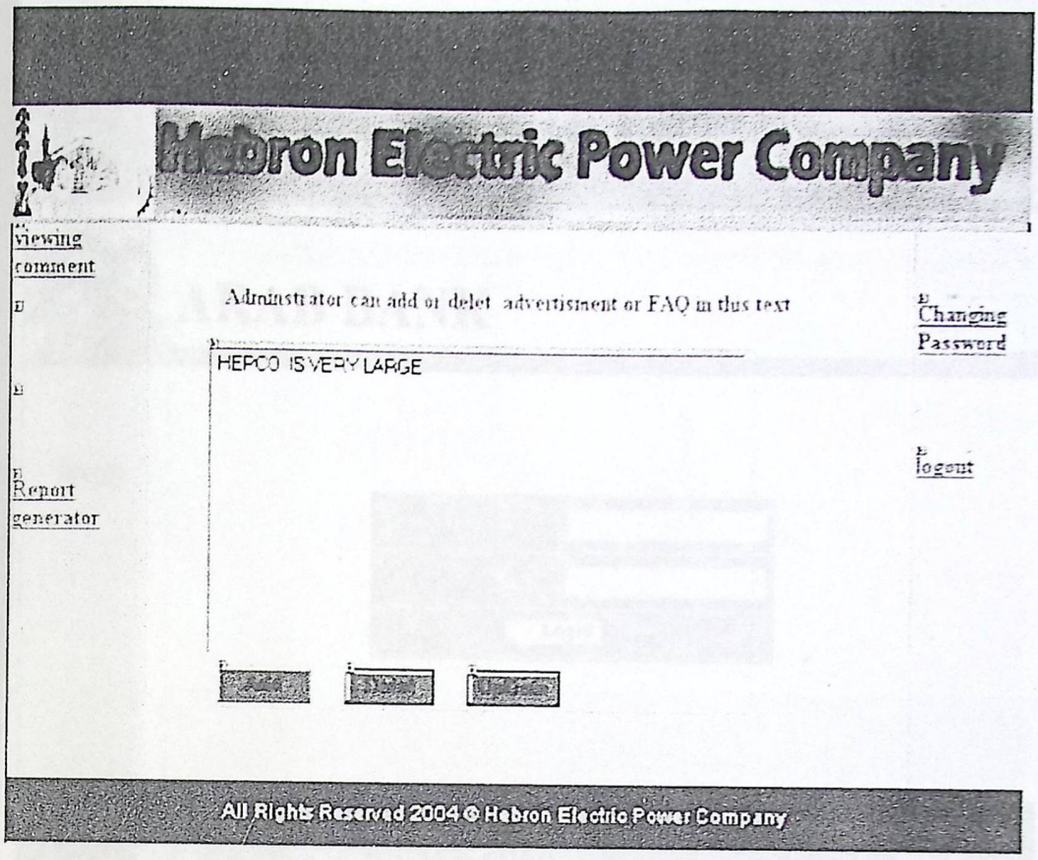


Figure 4.11 FAQ and Advertisement maintenance

4.2.2 Bank screens

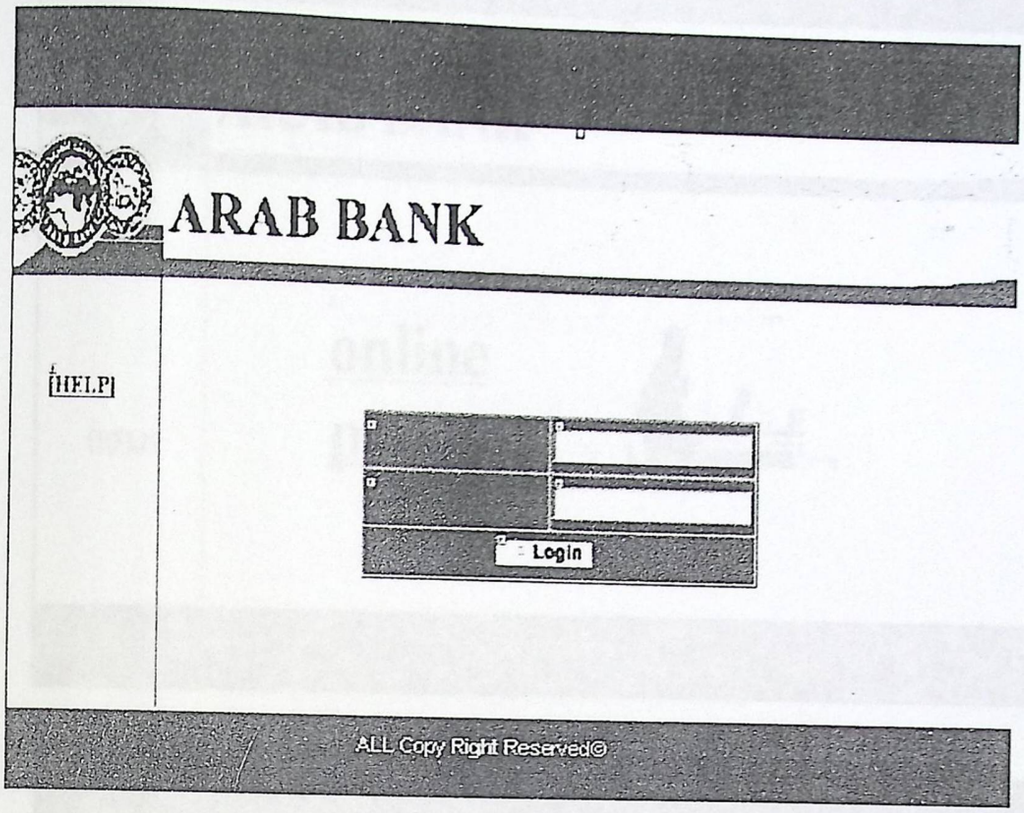


Figure 4.12 Bank login

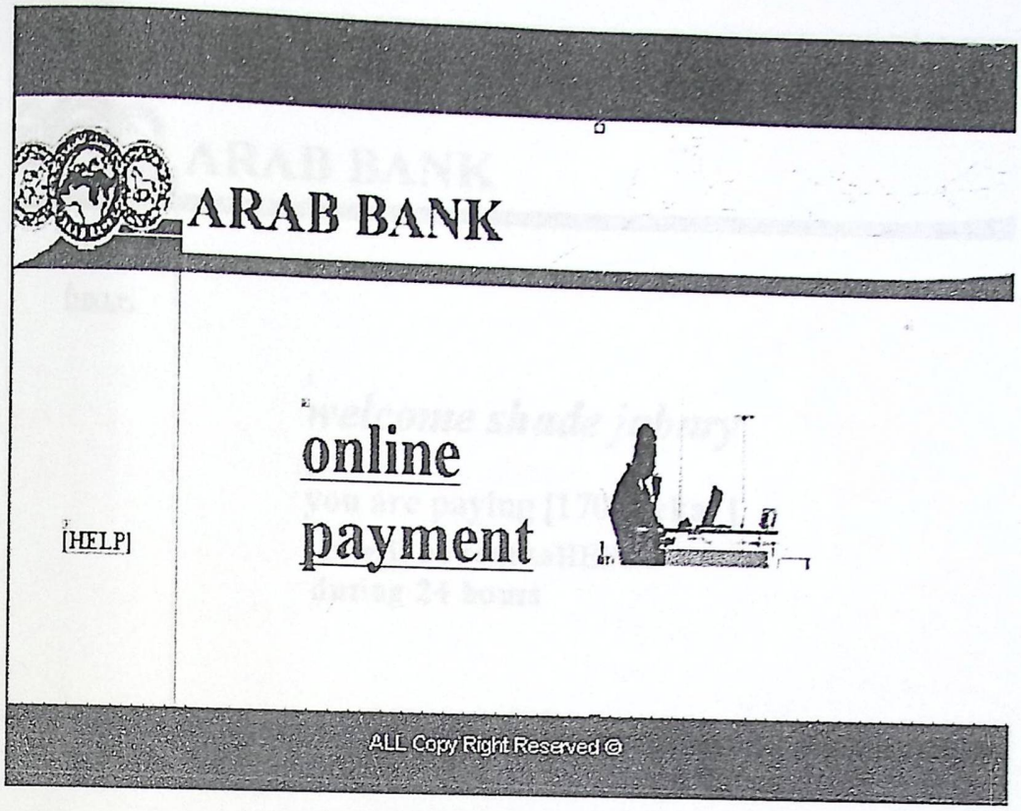


Figure 4.13 Online payment using bank

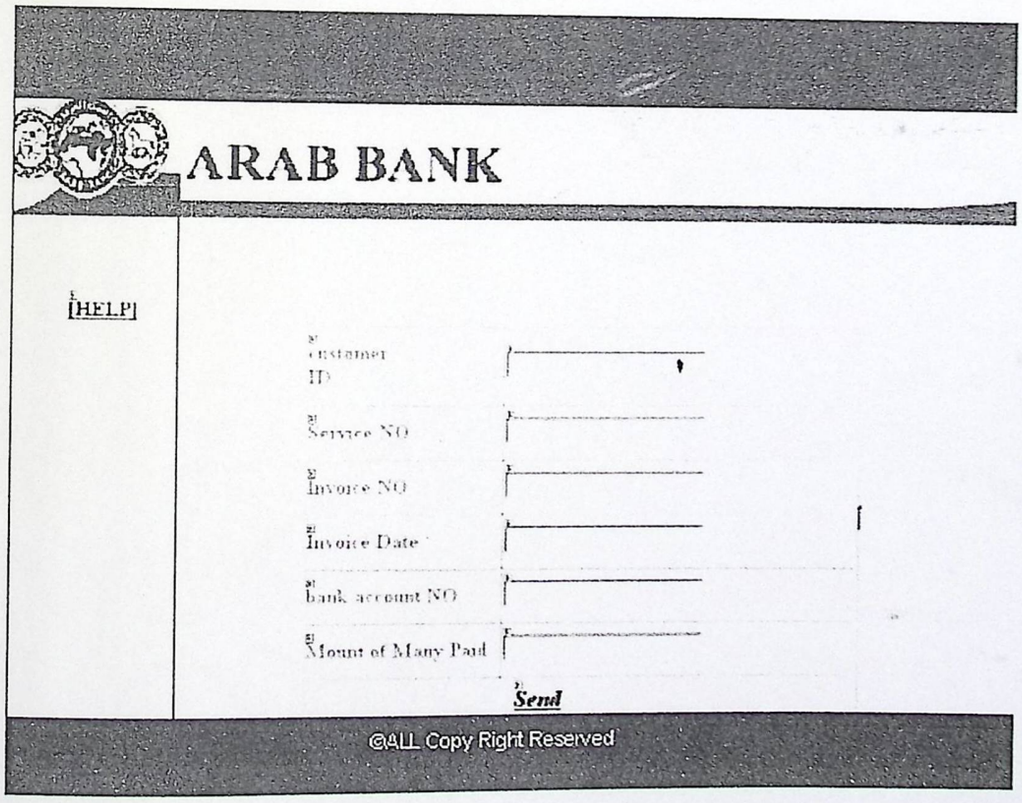


Figure 4.14 payment screen

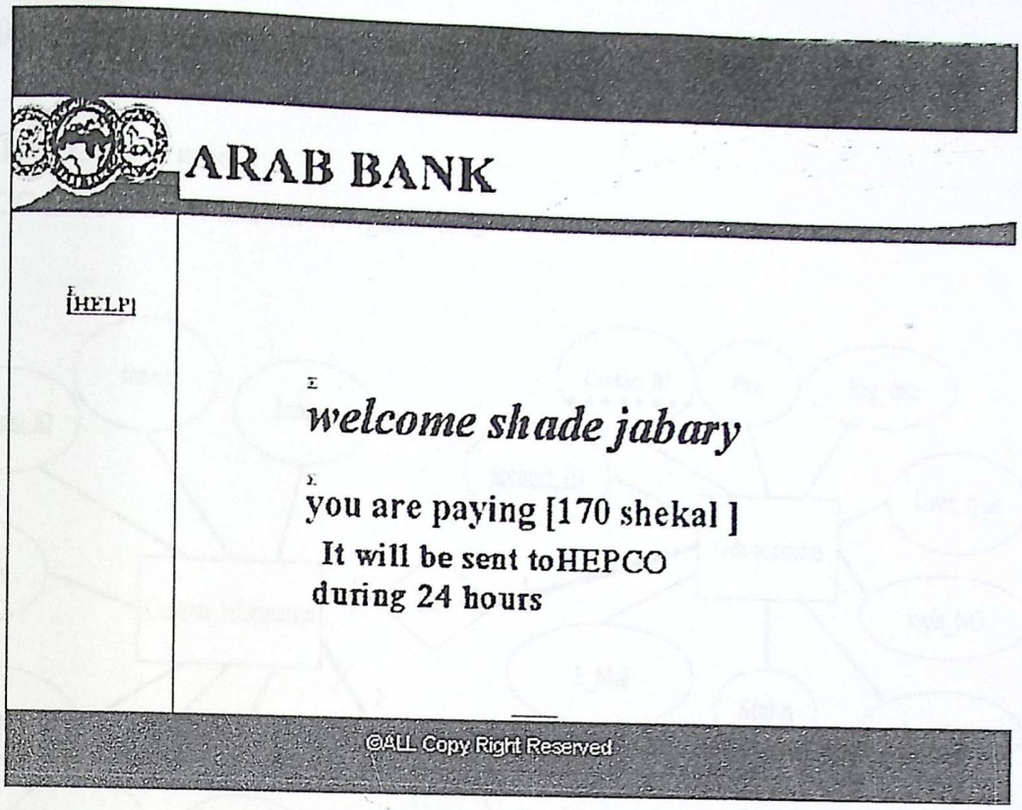


Figure 4.15 payment confirmation

4.3 Database Design

4.3.1 HEPCO ER model

HEPCO ER model is shown in figure 4.16

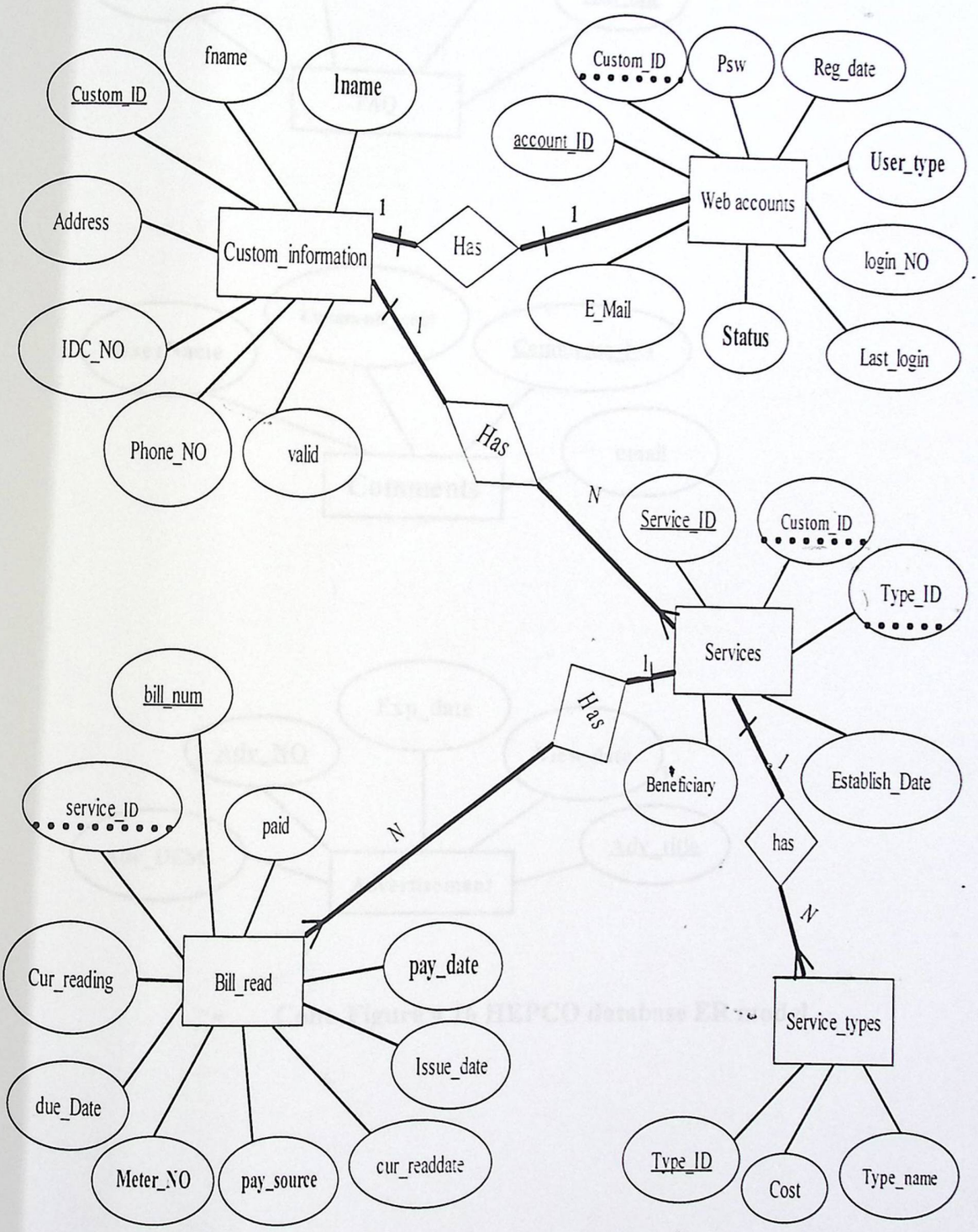
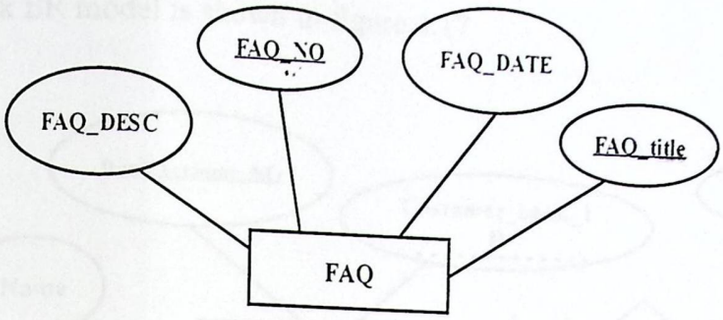
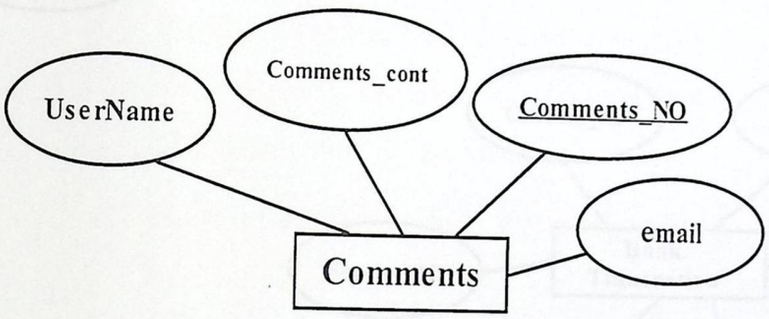
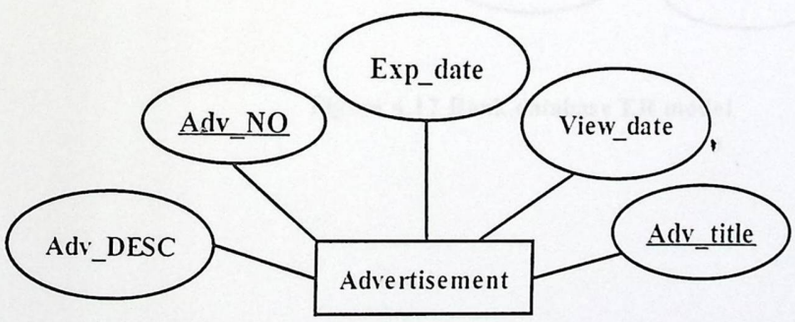


Figure 4.16 HEPCO database ER model







Cont. Figure 4.16 HEPCO database ER model

4.3.2 Bank ER Model

Bank ER model is shown in figure 4.17

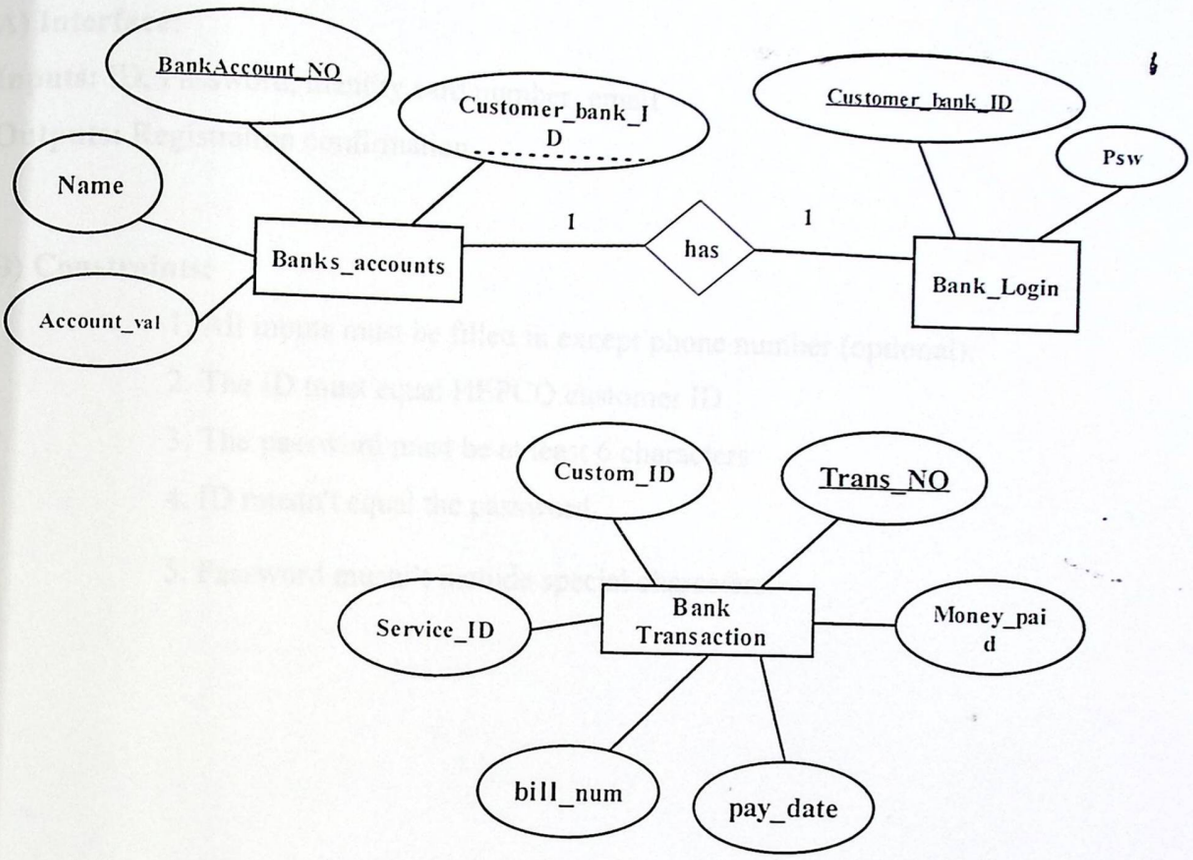


Figure 4.17 Bank database ER model

4.4 Functional Design

This section provides a brief description for each function and its flowchart.

1. Registration for online service:

The public user should get an ID and a password to get online services

A) Interface:

Inputs: ID, Password, identity card number, email

Outputs: Registration confirmation.

B) Constraints:

1. All inputs must be filled in except phone number (optional).
2. The ID must equal HEPCO customer ID
3. The password must be at least 6 characters
4. ID mustn't equal the password
5. Password mustn't include special characters.

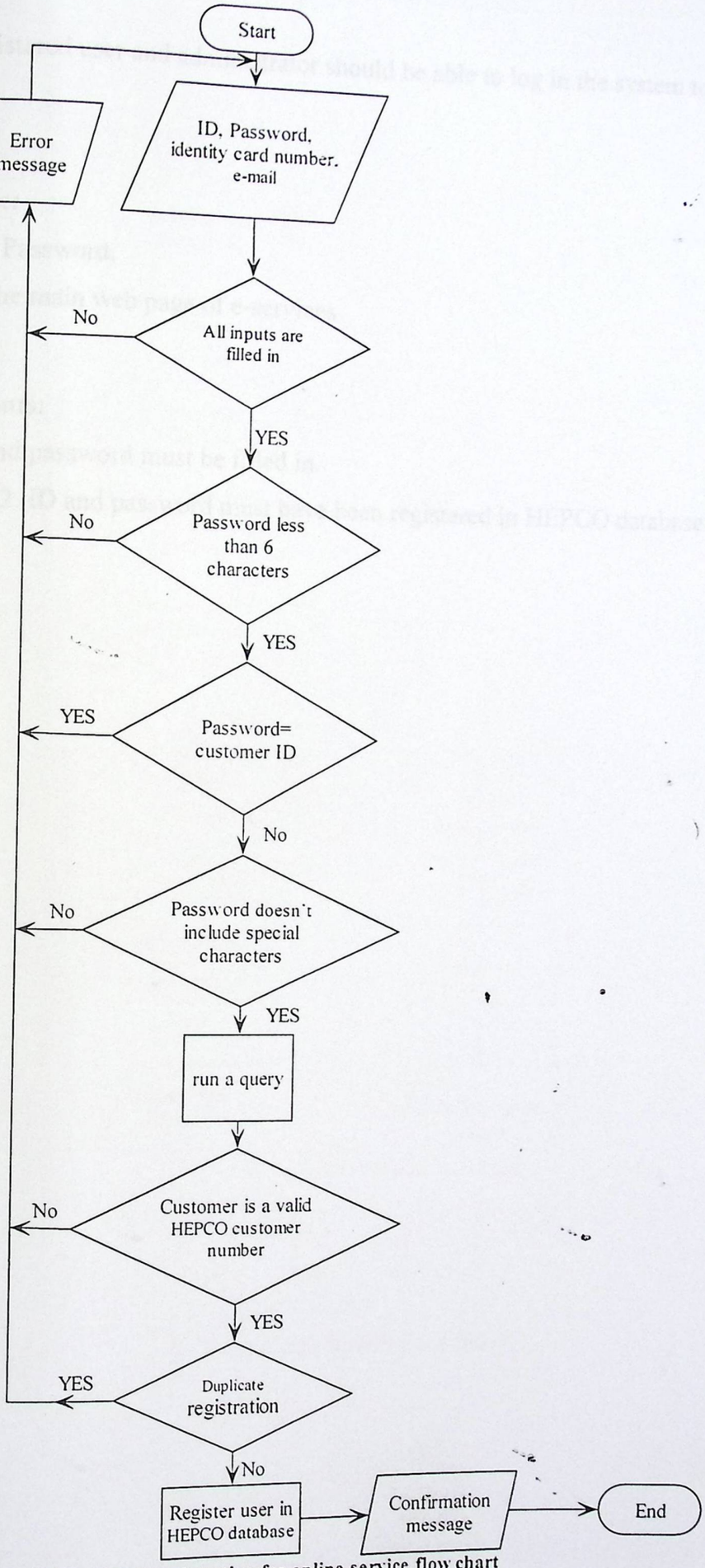


Figure 4.18 Registration for online service flow chart

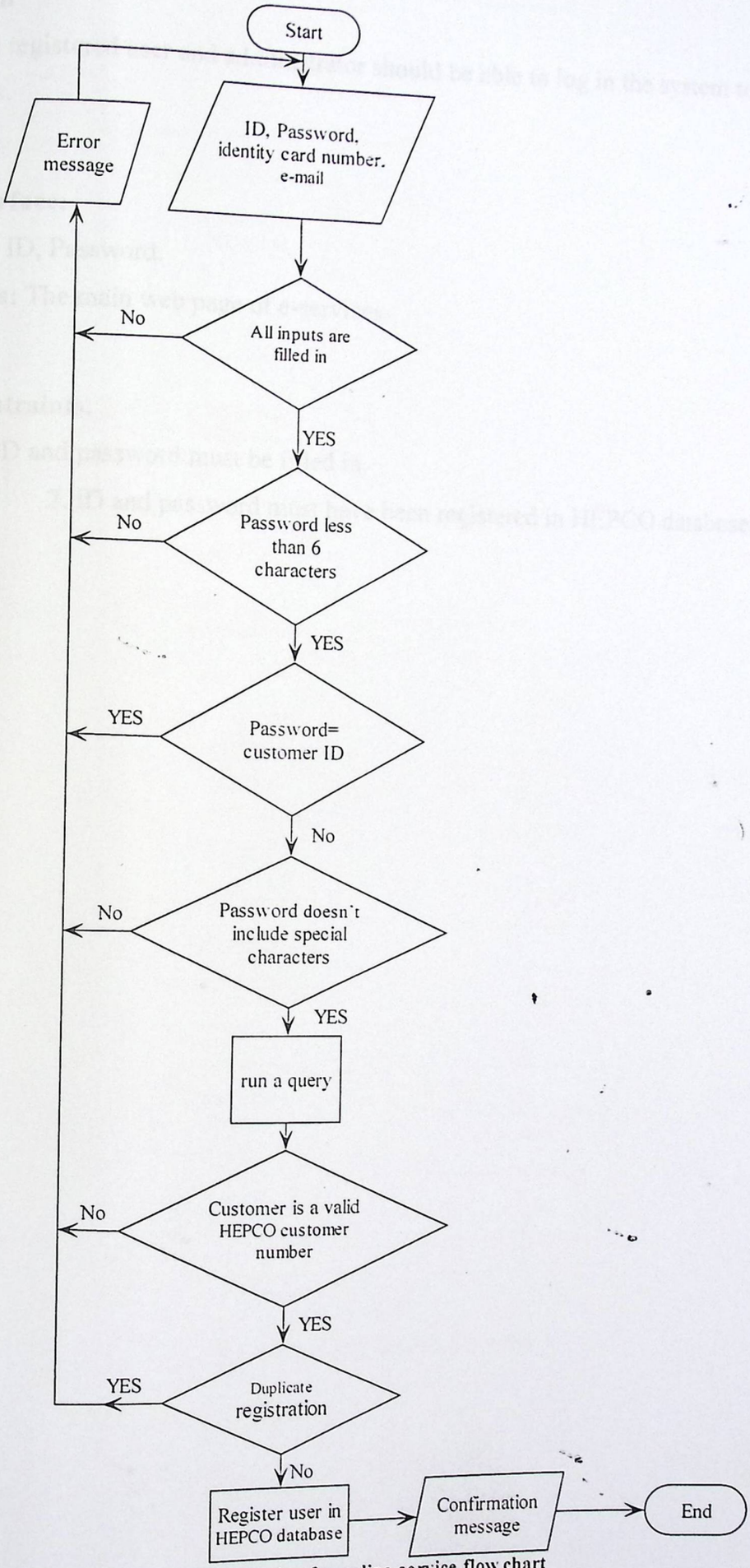


Figure 4.18 Registration for online service flow chart

2. Login

The registered user and administrator should be able to log in the system to use e-services.

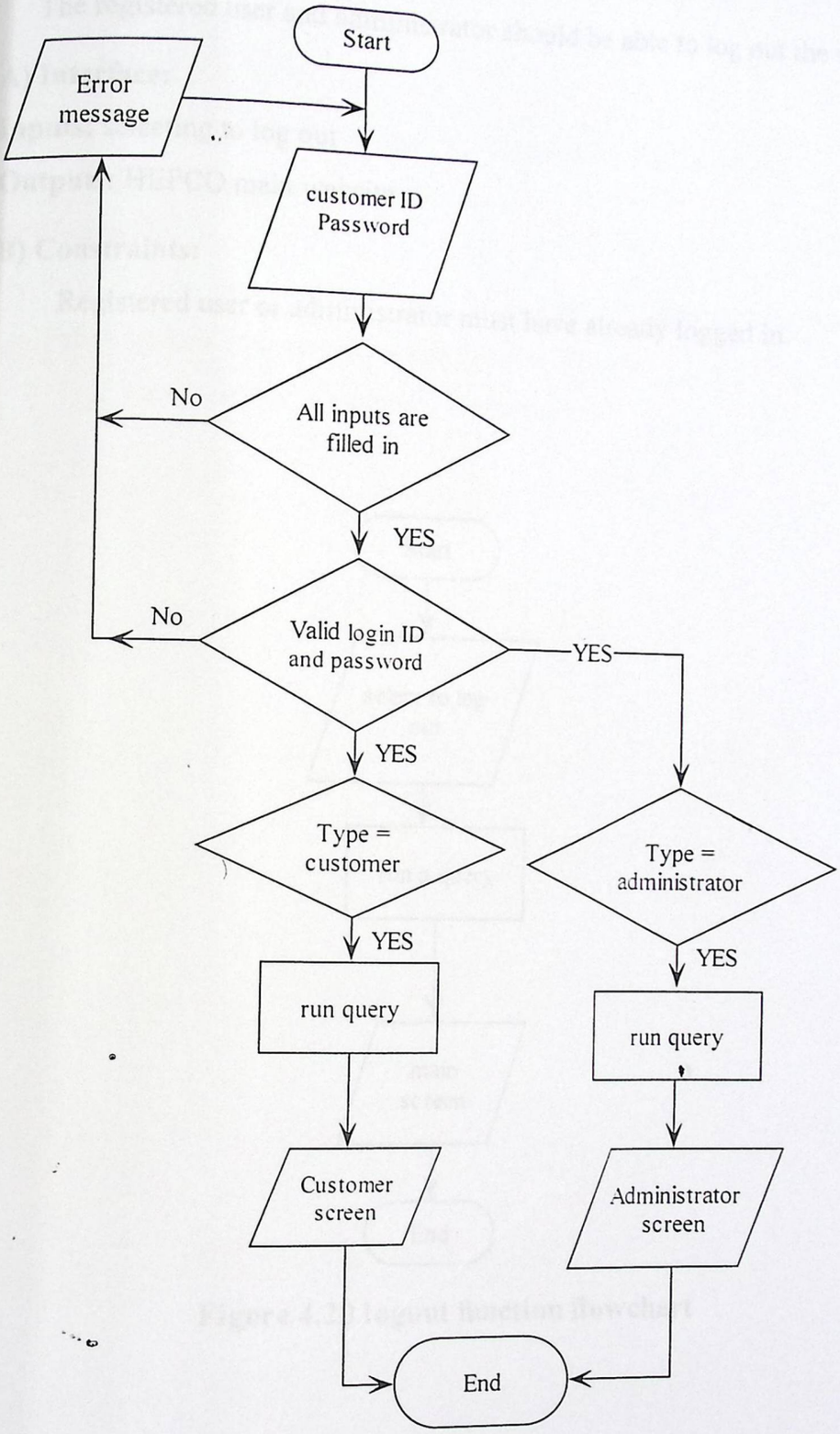
A) Interface:

Inputs: ID, Password.

Outputs: The main web page of e-services

B) Constraints:

1. ID and password must be filled in.
2. ID and password must have been registered in HEPCO database.



4.19 Login function flowchart

3. Logout

The registered user and administrator should be able to log out the system

A) Interface:

Inputs: selecting to log out

Outputs: HEPCO main website

B) Constraints:

Registered user or administrator must have already logged in.

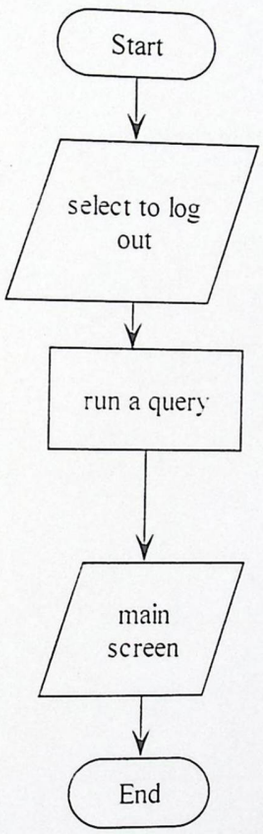


Figure 4.20 logout function flowchart

4. Changing login password

The registered user and the administrator should be able to change login password.

A) Interface:

Inputs: ID, old password, new password, password confirmation

Outputs: changing password confirmation

B) Constraints:

1. The new password must be at least 6 characters long and doesn't include special characters, and mustn't equal the user ID.

2. New password= password confirmation

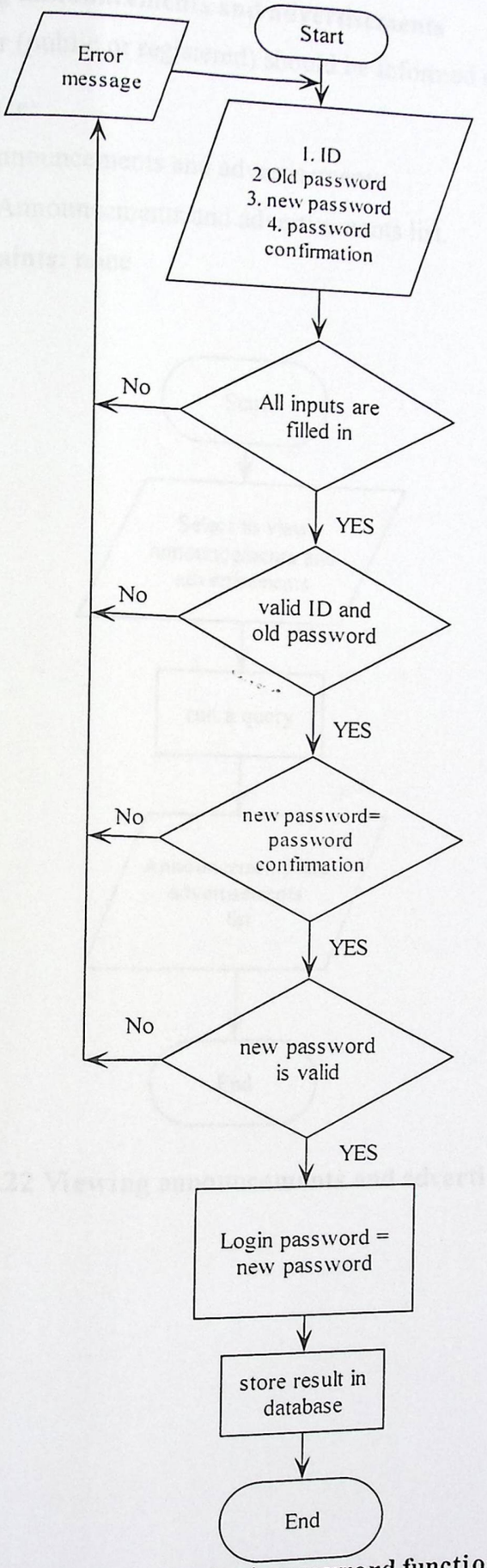


Figure 4.21 Changing login password function flowchart

5. Viewing announcements and advertisements

The user (public or registered) should be informed of the company announcements.

A) Interface:

Inputs: Announcements and advertisements

Outputs: Announcements and advertisements list.

B) Constraints: none

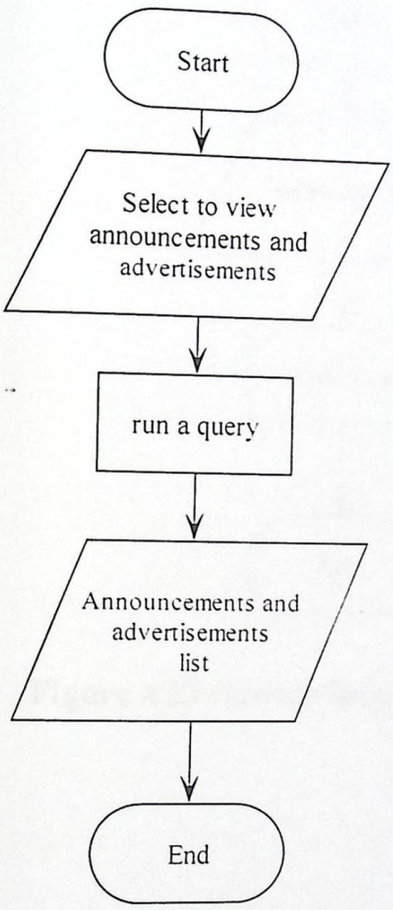


Figure 4.22 Viewing announcements and advertisements function flowchart

6. Viewing Help

The user should get help.

A) Interface:

Inputs: Help contents.

Outputs: Help information.

B) Constraints: none

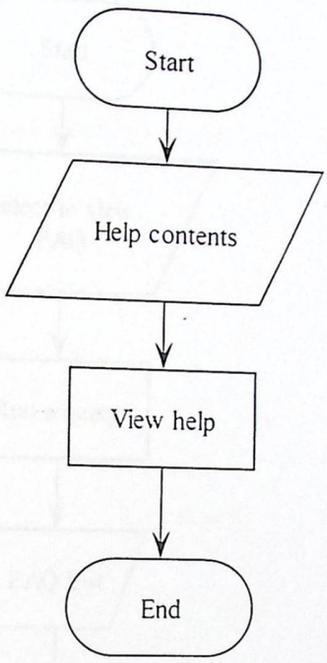


Figure 4.23 viewing help function flowchart

7. Viewing FAQ

The user should be able to view the frequently asked questions about the system.

A) Interface:

Inputs: Questions and answers

Outputs: FAQ list.

B) Constraints: none

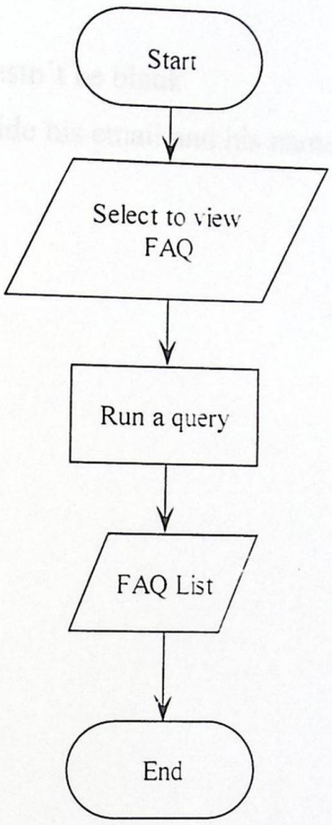


Figure 4.24 Viewing FAQ function flowchart

8. Reporting comments

The users should be able to report their comments and feedbacks

A) Interface:

Inputs: Comment, questions and user name and email.

Outputs: Confirmation screen.

B) Constraints:

1. Message mustn't be blank
2. The user must provide his email and his name.

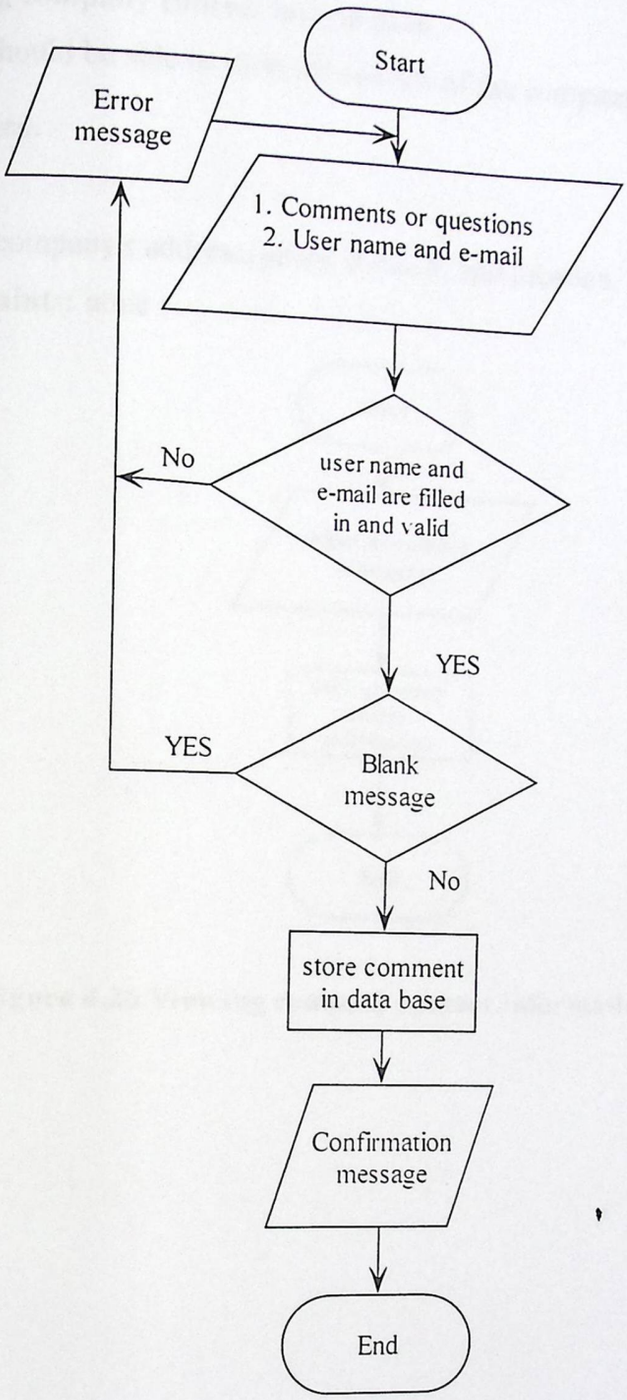


Figure 4.25 Reporting comments function flowchart

9. Viewing company contact information

The user should be able to view the address of the company to contact it.

A) Interface:

Inputs:-.

Outputs: company's address, phone #, fax #, and location.

B) Constraints: none

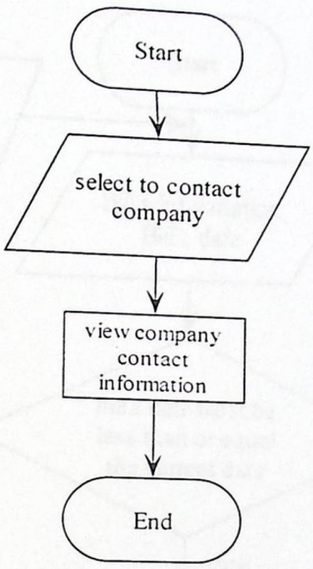


Figure 4.26 Viewing company contact information function

10. Viewing Bill

The registered user should be able to view, save and print bills.

A) Interface:

Inputs: Bills information, bill date

Outputs: Bill is displayed, printed or saved

B) Constraints: bill's date must be less than or equal the current date.

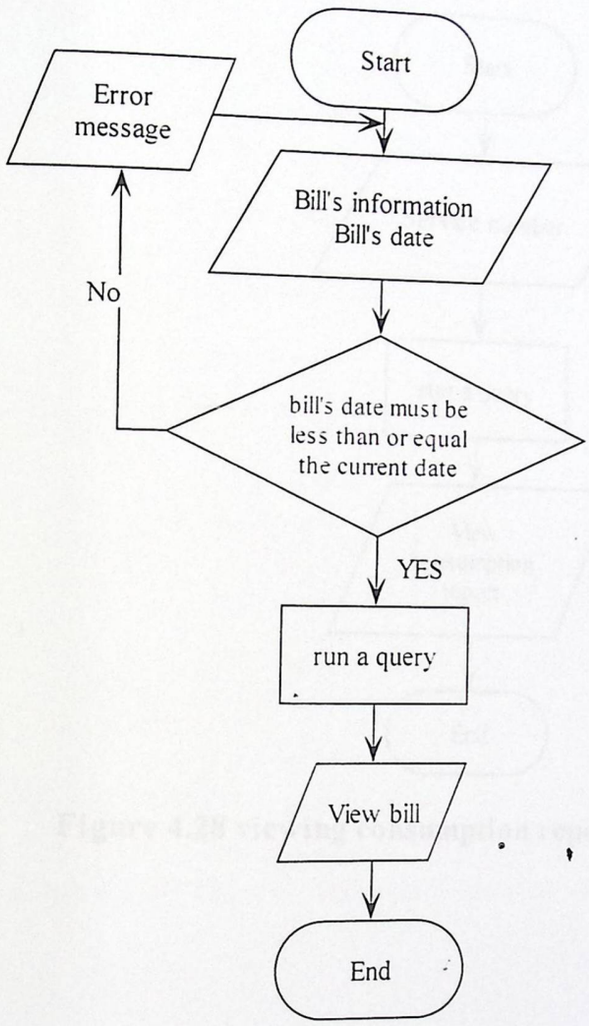


Figure 4.27 viewing bill function flowchart

11. Viewing Consumption Report

The registered user should be able to view their consumption reports.

A) Interface:

Inputs: Service number

Outputs: Consumption report for the last 6 months is viewed

B) Constraints: none.

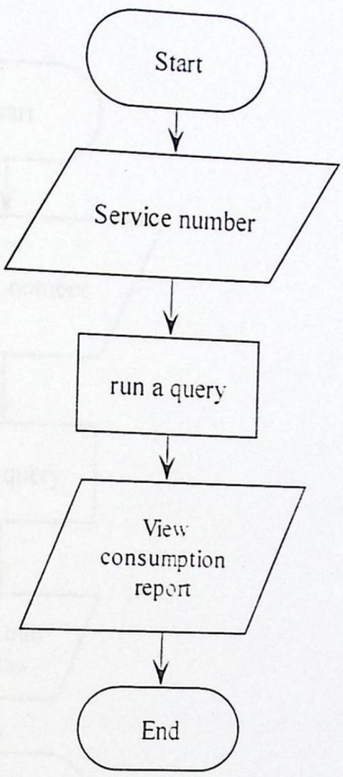


Figure 4.28 viewing consumption report flowchart

12. Viewing Bills Status

The registered user should be able to view bills status for 12 months.

A) Interface:

Inputs: bills information

Outputs: Bills Status for the last 12 months is viewed.

B) Constraints: none

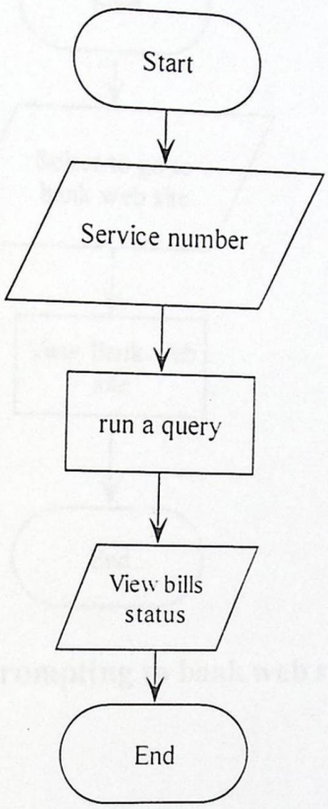


Figure 4.29 Viewing bill status function flowchart

13. Prompting to Bank Website

The registered user should be prompted to bank website to pay his bills.

A) Interface:

Inputs: Select to go to bank website

Outputs: Bank website

B) Constraints: none

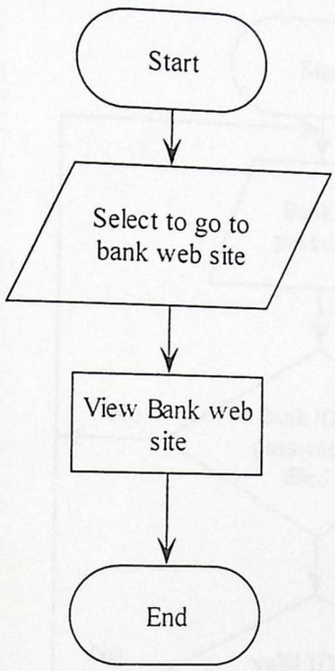


Figure 4.30 Prompting to bank web site function flowchart

14. Bank Login

The registered user should be able to log in the bank website.

A) Interface:

Inputs: Bank ID and password

Outputs: Bank e-payment screen

B) Constraints: ID and Password must be provided and must be valid.

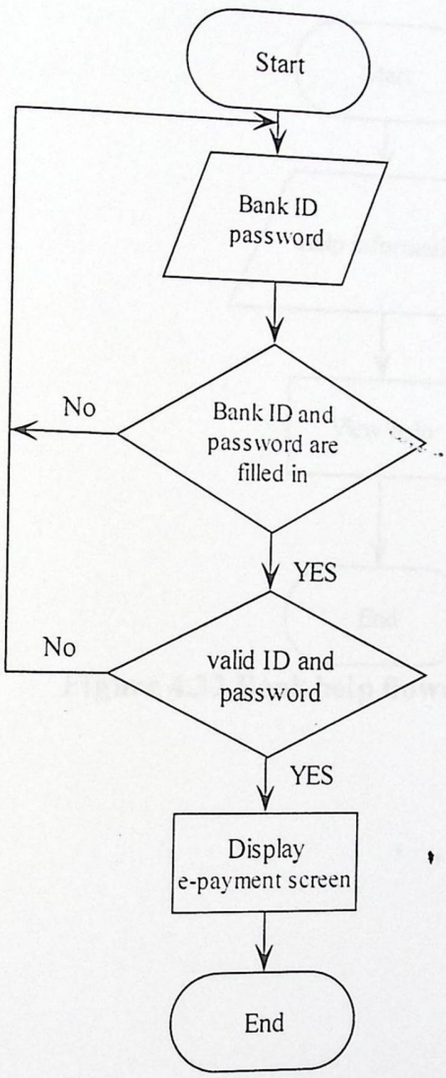


Figure 4.31 Bank login function flowchart

15. Bank Help

The registered user should get a help from the bank website.

A) Interface:

Inputs: Help information

Outputs: help viewed.

B) Constraints: none

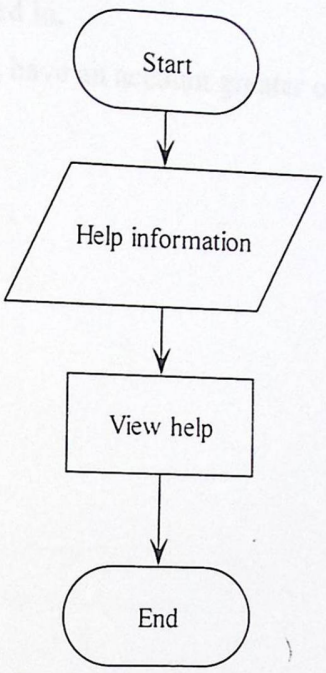


Figure 4.32 Bank help flowchart

16. Online Payment Using Bank Website

The registered user should be able to pay bills by bank website

A) Interface:

Inputs: account number, bill number, HEPCO customer ID, service ID and amount of money to be paid

Outputs: Payment confirmation.

B) Constraints: 1.all inputs must be filled in,

2. The registered user must have an account greater or equal to bill value.

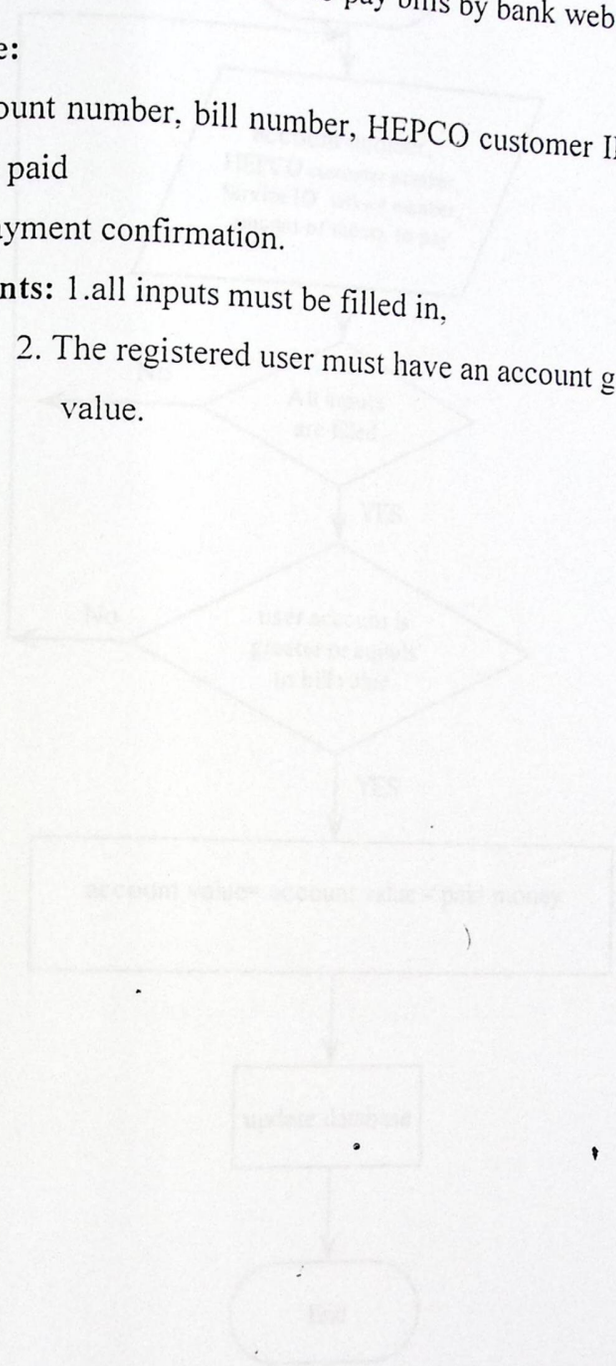


Figure 4.33 Online Payment Using Bank Website flowchart

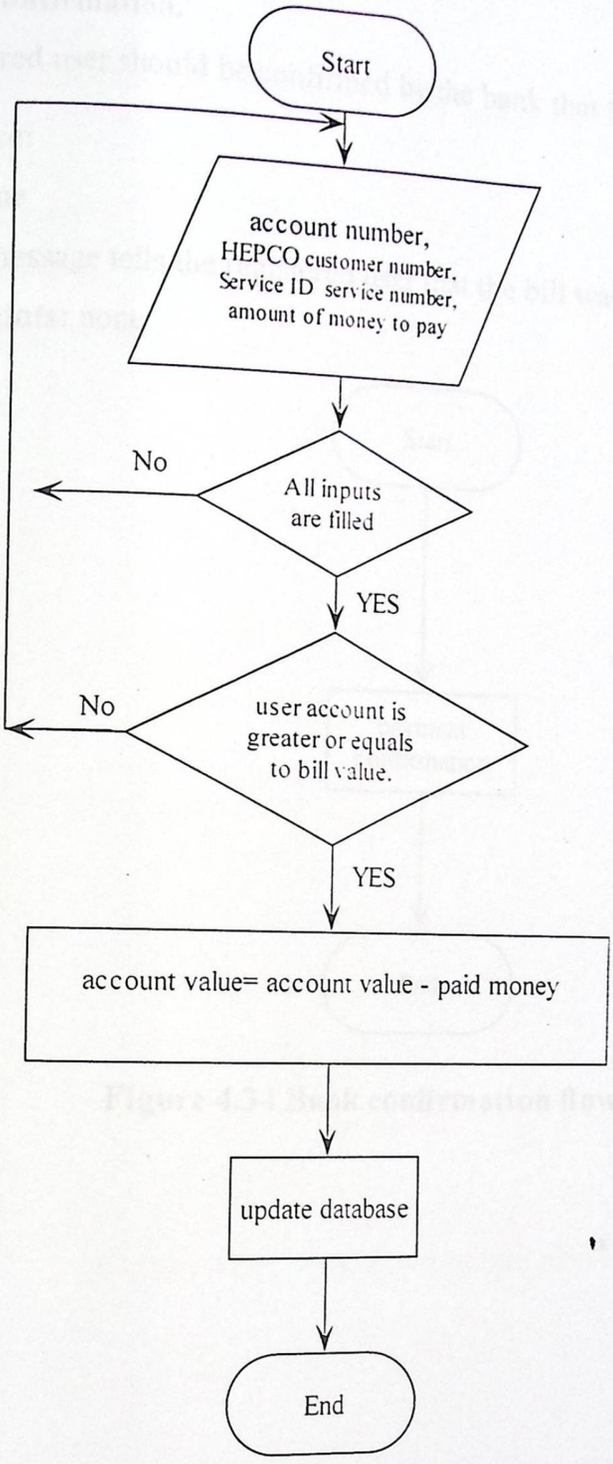


Figure 4.33 Online Payment Using Bank Website flowchart

17. Bank confirmation.

The registered user should be confirmed by the bank that the bill was paid.

A) Interface:

Inputs: none

Outputs: message tells the registered user that the bill was paid.

B) Constraints: none

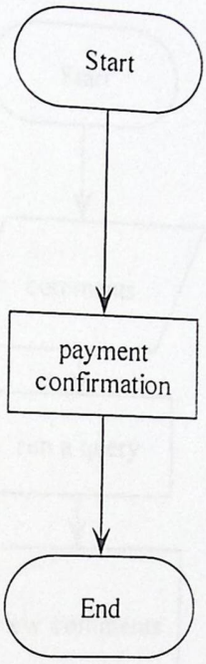


Figure 4.34 Bank confirmation flowchart.

18. Viewing Comments

The administrator should be able to view comments sent by users.

A) Interface:

Inputs: comments

Outputs: comments list and details.

B) Constraints: none

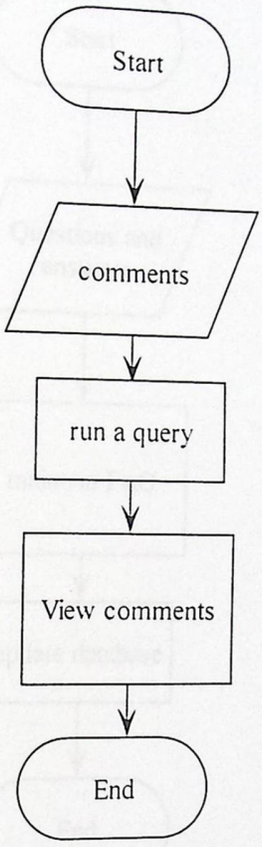


Figure 4.35 Viewing comment flowchart

19. FAQ Maintenance

The administrator should be able to add, delete and modify FAQ.

A) Interface:

Inputs: Questions and answers.

Outputs: FAQ list

B) Constraints: none

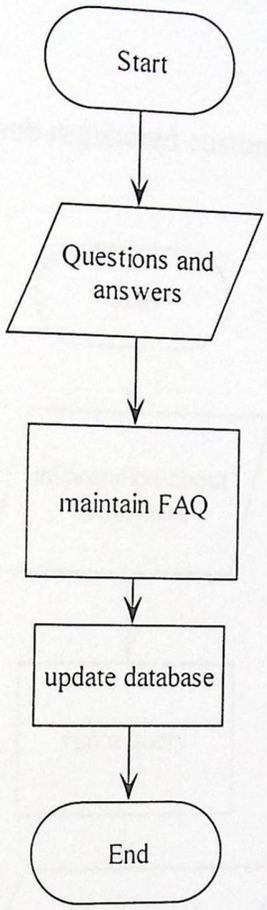


Figure 4.36 FAQ maintenance flowchart

20. Reports Generation

The administrator should be able to generate reports, like number of web registered customers, customer profile.

A) Interface:

Inputs: Reports information

Outputs: Generated reports

B) Constraints: none

20.1 reporting number of web registered customers:

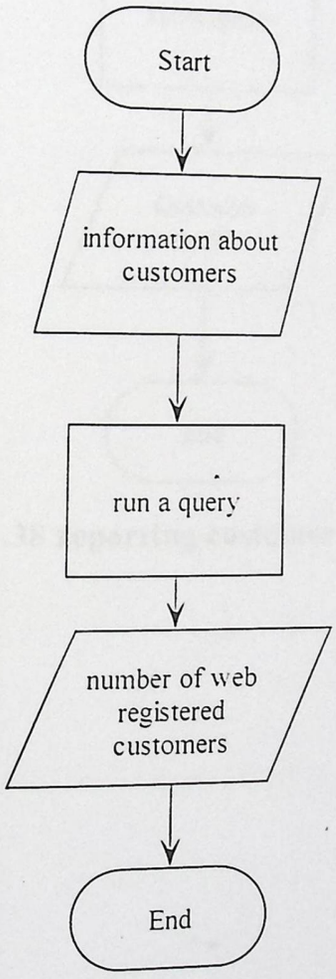


Figure 4.37 Reporting number of web registered customers flowchart

21.2 reporting customer profile:

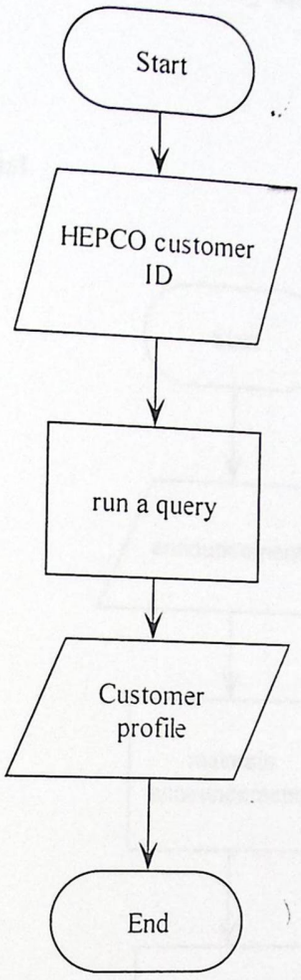


Figure 4.38 reporting customer profile flowchart:

21. Announcements Maintenance

The administrator should be able to add, modify and delete announcements

A) Interface:

Inputs: announcements

Outputs: announcements list

B) Constraints: none

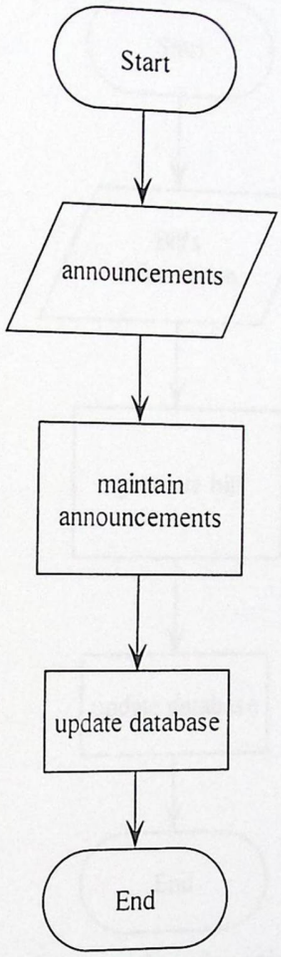


Figure 4.40 Announcements Maintenance flowchart

22. Bill's Generation:

The administrator should be able generate bills.

A) Interface:

Inputs: bill's information

Outputs: generated bill

B) Constraints: none

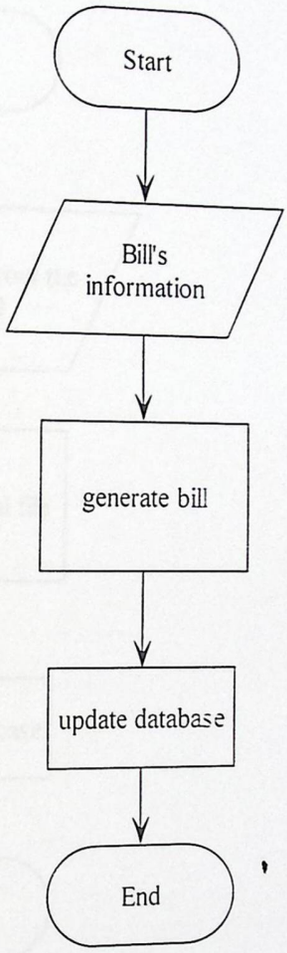


Figure 4.41 Bill's Generation function flowchart

23. Generating XML transactions:

The bank should generate XML transactions to HEPCO

A) Interface:

Inputs: information about the paid bill

Outputs: XML file

B) Constraints: none

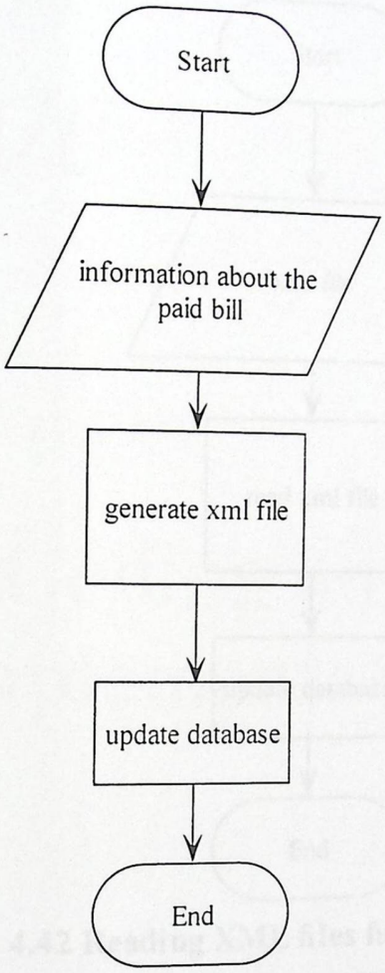


Figure 4.42 Generating XML transactions flowchart

24. Reading XML files:

HEPCO should read the XML files and update its database

A) Interface:

Inputs: XML files

Outputs: Confirmation.

B) Constraints: none

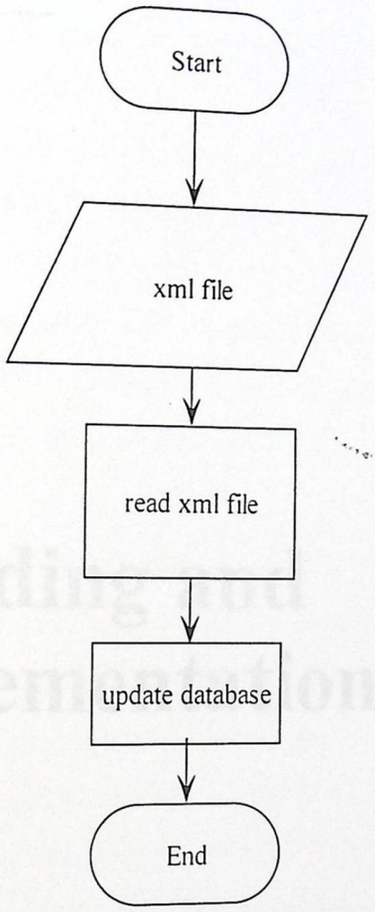


Figure 4.42 Reading XML files function flowchart

Chapter

Five

Coding and Implementation

5.1 Introduction

This chapter describes the main steps must be followed to start in coding and programming to reach to the design that is described in the previous chapters, and mention the programming language is used for this purpose. This chapter focuses on the coding and implementation of HEPCO project.

Coding refers to the process of writing the necessary program, which implements the main procedures and functions of the project. The code of the project is to be written from the scratch using VB.NET language and asp.net.

The project is to be implemented as a set of programmed pages; each page carries out a specific functions related to the project, the project is to be programmed under windows XP operating system.

5.2 Coding Programming Language

Before starting in coding, it is desirable to study the scopes of languages and to match between the project requirements and what a specific language offers. There are different programming languages and different DB systems to implement the project, but after studying and analyzing all these options, the development team has selected ASP.NET for coding the web-based application, SQL server 2000 for DB and Visual Basic.NET for coding.

- *Relationship between ASP.NET, VB.NET, SQL server2000:*

After creating the design specifications that will be used while creating the web application, create a new project in the Visual Studio.NET and create the interface for HEPCO web application by setting the initial properties for the asp.net web forms. Visual Basic.NET is used to write the events procedures that will run when different actions are performed on a control or object. Then HEPCO database is built using SQL server 2000.

From the visual studio.net a connection to HEPCO is created, this

connection provides the facilities to access the database (input, output, access) from the web application. Since the desirable system is to be run on the internet and need a database handle its data, ASP.NET is chosen which is good to accomplish the project. It is built on .NET platform.

- *What is .NET platform?*

.NET is a platform that can be used for building and running the next generation of Microsoft Windows and web application .The .NET platform consists of the following core technologies:

1. The .NET Framework
2. The .NET enterprise Servers
3. Building block services
4. Visual studio.NET

- *Why Visual Studio.NET?*

Visual Studio.NET simplifies the development of powerful, reliable enterprise Web solutions and increases developer efficiency by providing a familiar, shared development environment. Pre-build components, programming wizards and the ability to reuse components that are written in any language can reduce development time significantly. It provides Integrated Development Environment (IDE) that provides a consistent look and feel, regardless of programming language being used. IDE provides the tools that are used in programming as shown in figure5.1. [6]

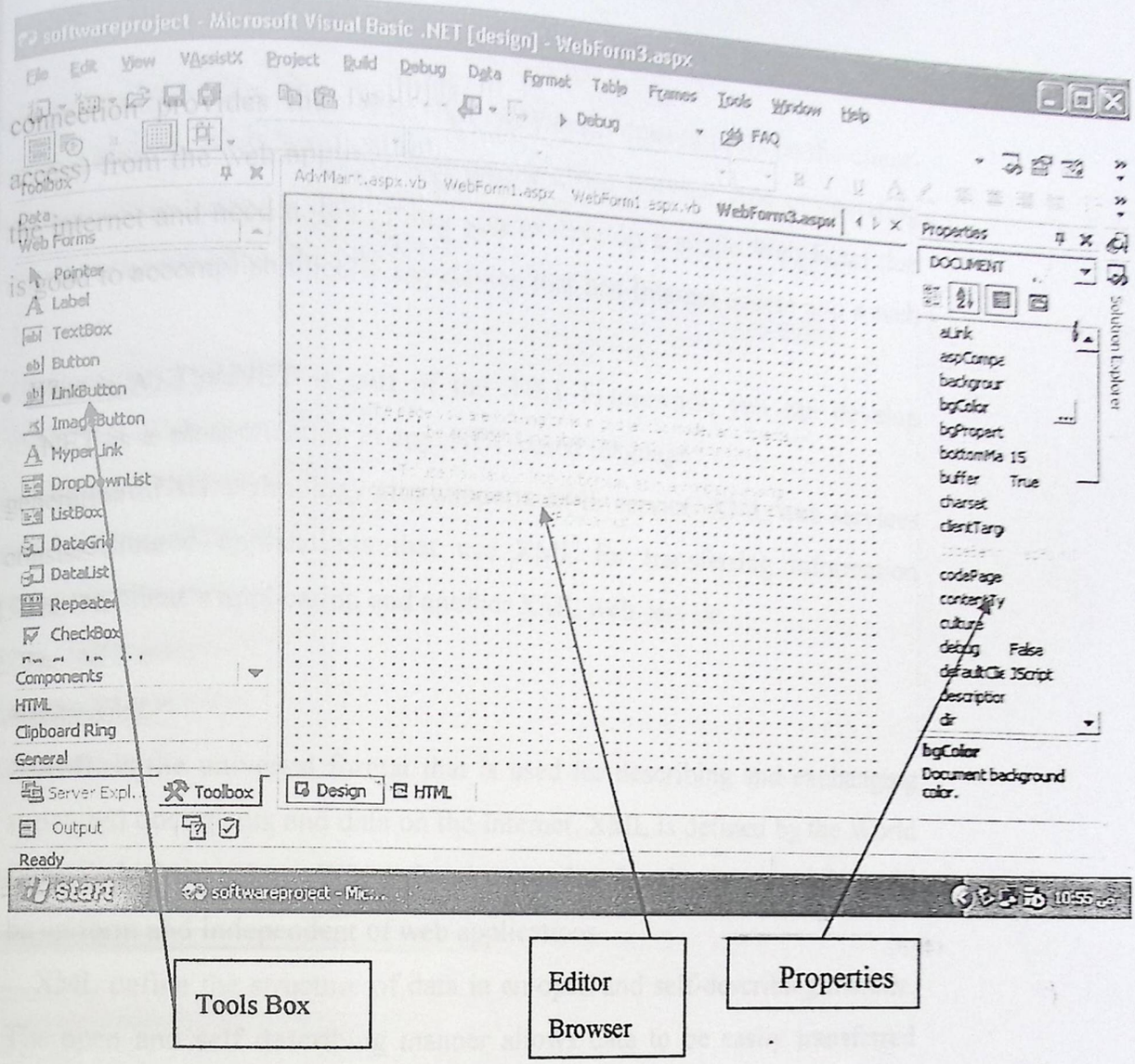


Figure 5.1 Integrated Development Environment in visual studio.NET

- ***What is ASP.NET?***

Developing ASP.NET web applications in the .NET framework are similar to developing windows applications. The fundamental component of ASP.NET is the web form. The web form is the web page that users view in browsers. An ASP.NET web application comprises of one or more web forms. A web form is dynamic page that can access server resources.

For example, a traditional web page can run script on the client to perform basic tasks. An ASP.NET web form conversely, can also run server side code to access a database, to generate additional Web Forms or to take advantage of built-in security in the server.

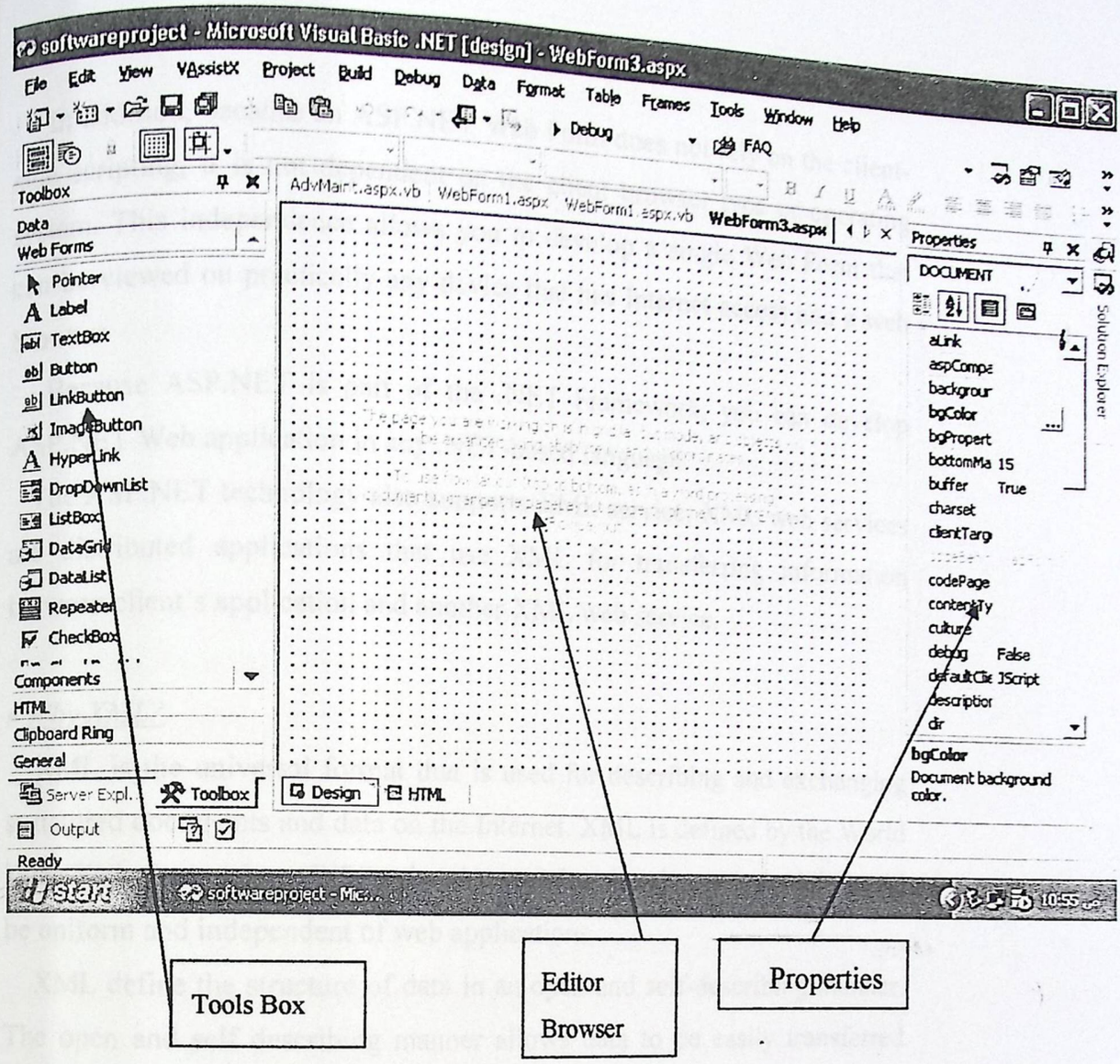


Figure 5.1 Integrated Development Environment in visual studio.NET

- ***What is ASP.NET?***

Developing ASP.NET web applications in the .NET framework are similar to developing windows applications. The fundamental component of ASP.NET is the web form. The web form is the web page that users view in browsers. An ASP.NET web application comprises of one or more web forms. A web form is dynamic page that can access server resources.

For example, a traditional web page can run script on the client to perform basic tasks. An ASP.NET web form conversely, can also run server side code to access a database, to generate additional Web Forms or to take advantage of built-in security in the server.

In addition, because an ASP.NET Web Form does not rely on the client-side scripting, it is not dependent on the client browser type or operating system. This independence allows you to develop a single Web Form that can be viewed on practically any device that has Internet access and a web browser

Because ASP.NET is part of the .NET Framework, you can develop ASP.NET Web application in any .NET-based language

The ASP.NET technology also supports XML service. XML web services are distributed applications that use XML for transferring information between client's application and another XML web service.

- *Why XML?*

XML is the universal format that is used for describing and exchanging structured documents and data on the Internet. XML is defined by the World Wide Web Consortium (W3C), thereby ensuring that the structured data will be uniform and independent of web applications

XML define the structure of data in an open and self-describing manner. The open and self describing manner allows data to be easily transferred over a network and to be consistently processed by the receiver. XML document describes how the data is structured, not how it should be displayed or used, it contains tags that assign meaning to the content of the document. The stages allows programmers to find the data that they need in the XML document

- *Why SQL server 2000?*

SQL Server 2000 is a powerful tool for turning information into opportunity. Industry-leading support for XML, enhanced tools for system management and tuning, and exceptional scalability and reliability make SQL Server 2000 the best choice for the agile enterprise.

It has the following features:

- High Availability: Maximize the availability of your business applications with log shipping, online backups, and failover clusters.
- Scalability: Scale your applications up to 32 CPUs and 64 gigabytes (GB) of RAM. SQL Server 2000 has demonstrated record-breaking performance that you can leverage
- Security: Ensure your applications are secure in any networked environment, with role-based security and file and network encryption.
- Simplified Database Administration: Automatic tuning and maintenance features enable administrators to focus on other critical tasks.
- Data Transformation Services: Automate routines that extract, transform, and load data from heterogeneous sources.
- Rich XML Support: Simplify the integration of your back-end systems and data transfer across firewalls using XML.

5.3 Database System

The database system is created according to the design that is descried in the previous chapter which achieves the system requirements.

The tables of the databases are shown in the table 5.1.

Table 5.1 Database Tables Description

Table name	Description
Customer_Information	This is the table of the all customers of HEPCO
Web_accounts	This is the table of the all customers who are registered for online services
Services	This table of all services for customers in HEPCO
Services_types	This table contains all types of the provided service and its related cost.
Bill_read	This table is created to provide information bills.
Advertisement	This table provides title and description of advertisement and announcements.
FAQ	This table provides contents and dates of frequently and ask questions
Comment	This table provides title and description of comment and email for customer who sent comment.
Bank_transaction	This table contains all the transactions of paying bills by bank
Bank_Login	This table contains customers' bank IDs and passwords
Bank_accounts	This table contains all users accounts information

- **Databases' tables description :**

1. HEPCO database

Customer Information: This is the table of the all customers of HEPCO

Field Name	Data Type	Key	Length	Description
Custom_ID	Varchar	PK	20	Unique number identifies the customer
fname	Varchar		20	Customer's first name
lname	Varchar		20	Customer's last name
IDC_NO	Varchar		9	The number of the identity card
Address	varchar		100	Address of the customer
Phone_NO	Varhcar		7	Customer's telephone number
valid	boolean		1	Determines whether the customer still get services from the company or no

Web accounts table: This is the table of the all customers who are registered for online services

Field Name	Data Type	Key	Length	Description
account_ID	Integer	PK	4	Identifies the web registered customer
Custom_ID	Varchar	FK	20	The number that identifies the customer
PSW	Varchar		20	Login password
Reg_date	Date		8	The date of customer's web registration
Status	Boolean		1	Determines weather the customer has logged in or not
Login_NO	Integer		4	The number of customer's login times
Last_Login	Date		8	Determines whether the customer still getting services from the company or not
User_type	Char		1	Determines whether the user is a

E_mail	Varchar		50	customer or Administrator Customer's e-mail address

Services Table: This table of all services for customers in HEPCO

Description	Length	Key	Data Type	Field Name
Service_ID	Varchar	PK	10	The unique number identifies each service
Custom_ID	Varchar	FK	20	The number of the customer who owns this service
Type_ID	Integer	FK	4	Identifies service's type
Establish_Date	Date		10	Date of establishing this service.
Beneficiary	Varchar		20	The name of the person who gets benefits of this service

Services types table: This table contains all types of the provided service and its related cost.

Field Name	Data Type	Key	Length	Description
Type_ID	Integer	PK	4	The unique number for each type
Type_name	Varchar		20	The name of each service
Cost	Float		8	The cost for each KW

Bill_read table: This table is created to provide information bills.

Field Name	Data Type	Key	Length	Description
bill_num	Decimal	PK	9	The number of the generated bill(each bill has a unique number)
Service_ID	Varchar	FK	10	the number of the service that the bill is generated to(unique for each service)
Issue_date	Date		8	The date of bill generation
due_Date	Date		8	The date that the bill should be paid before
Meter_NO	Integer		4	The meter's number
Cur_reading	Decimal		9	Current reading of meter
Cur_readdate	Date		8	The date of current reading
paid	Boolean		1	This field determine whether this bill is paid or not
pay_source	Varchar		50	The source by which the bill was paid
pay_date	Date		8	The date when the bill was paid

Advertisement table: This table provides title and description of advertisement and announcements.

Field Name	Data Type	Key	length	Description
Adv_NO	Integer	PK	4	A unique number for each advertisement
Adv_title	Varchar		50	The title of the advertisement
Adv_DESC	Varchar		500	Contents of the advertisement
Exp_date	Date		8	The date after which the advertisement mustn't be viewed
View_date	Date		8	The date when the advertisement will be viewed in the screen

FAQ table: This table provides contents and dates of the frequently and ask question

Field Name	Data Type	Key	Length	Description
FAQ _NO	Integer	PK	4	FAQ number
FAQ_title	Varchar		50	The question
FAQ_DESC	Varchar		500	The answer of the question
FAQ _ Date	Date		10	The last modification date

Comment table: This table provides title and description of comment and email for customer who sent comment.

Field Name	Data Type	Key	Length	Description
Comments_NO	Integer	PK	4	Unique number for each comment
Comments_cont	Varchar		200	The contents of the comment.
UserName	Varchar		50	The name of the user who send this comment
email	varchar		50	The e-mail of the user who sent the message

2-Bank database

Bank Login table: This table contains customers' bank IDs and passwords

Field Name	Data Type	Key	Length	Description
Customer_bank_ID	Varchar	PK	20	The unique number for each bank agent which is given by bank
Psw	Varchar		20	Bank login password

Bank_accounts table: the table that contains all users accounts information

Field Name	Data Type	Key	Length	Description
Customer_bank_ID	Varchar	FK	20	The unique number for each bank agent which is given by bank
BankAccount_NO	Varchar	PK	10	The number of the bank account for the customer
Name	Varchar		50	The name of the customer
Account_val	float		8	The amount of money that the customer has in his account.

Bank_transaction: This table contains all transactions of bill's payments'.

Field Name	Data Type	Key	Length	Description
Trans_NO	Integer	PK	4	A unique number that identifies the transaction
Custom_ID	Varchar		20	The ID of HEPCO customer
Service_ID	Varchar		10	The ID of the service
bill_num	Decimal		9	The number of the invoice that is paid
Pay_date	Date		8	The date of payment
Money_paid	Float		8	The amount of the money paid

5.4 Establishment of Development Environment

- Purchase the computers and the software required for developing the system.
- Install windows XP
- Install the required utilities
- Install the IIS; from the control panel, choose add remove programs then choose windows components then add the IIS component by choosing its check box then click Next as shown in figure 5.2.

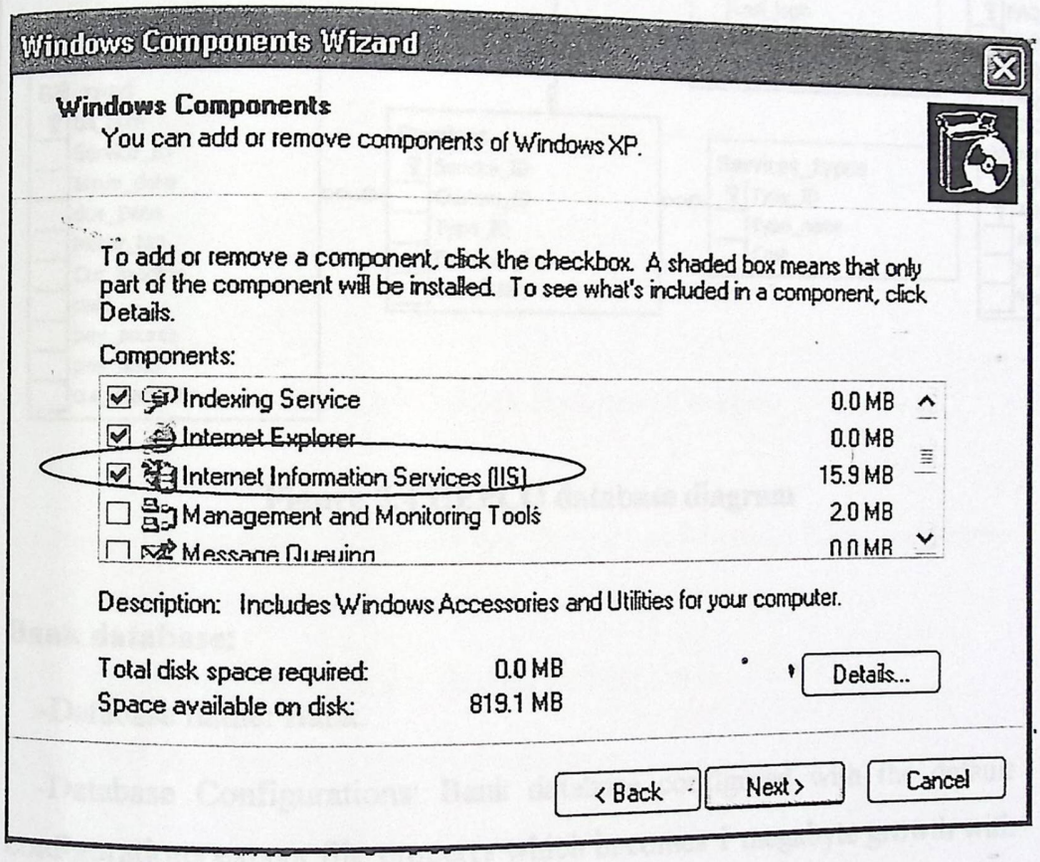


Figure 5.2 Installing IIS

- Install visual studio.NET : to achieve this first a perquisite CD is required then first VS.NET CD then the second VS.NET CD
- Install SQL server 2000 software.

- Create the database, Relationships, Primary and Foreign Keys: all primary and foreign keys are created to ensure database consistency.
- Create indexes to facilitate dealing with database
- HEPCO database diagram for the final design is shown in figure 5.4

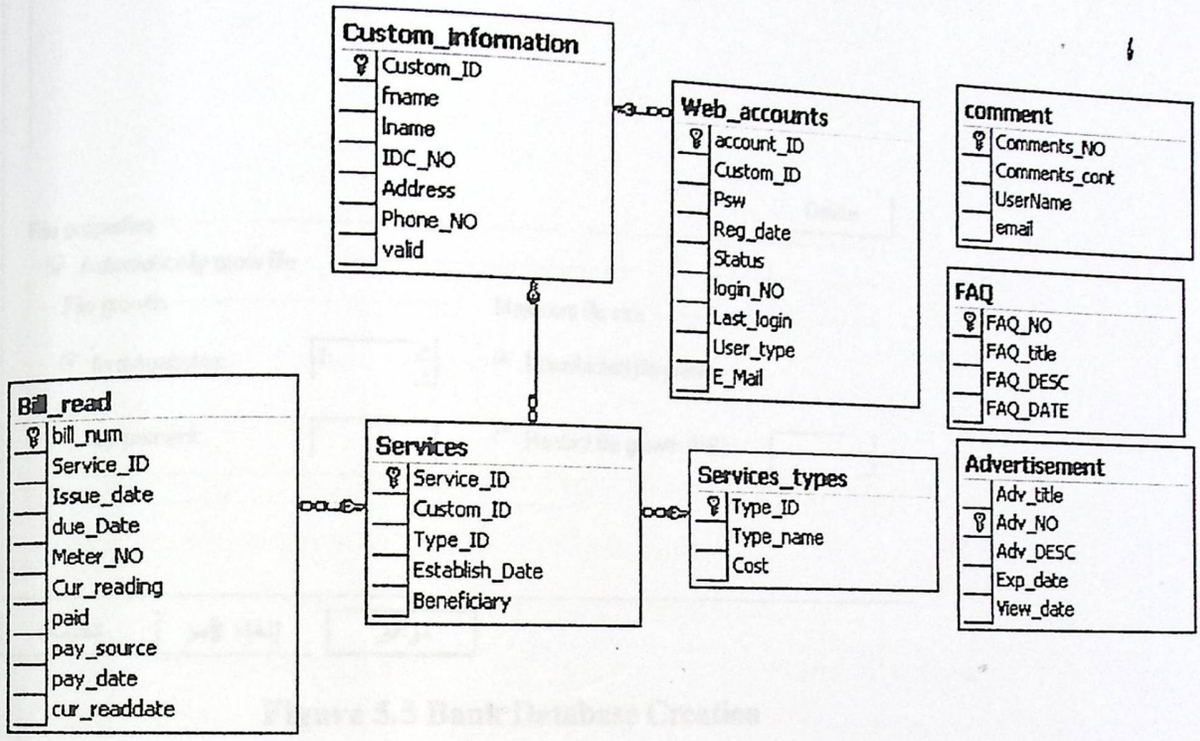


Figure 5.4 HEPCO database diagram

Bank database:

-Database name: Bank.

-Database Configurations: Bank database configured with the default configurations except file property which becomes 1 megabyte growth with unrestricted file size.

Figure5.5 shows that bank database is created.

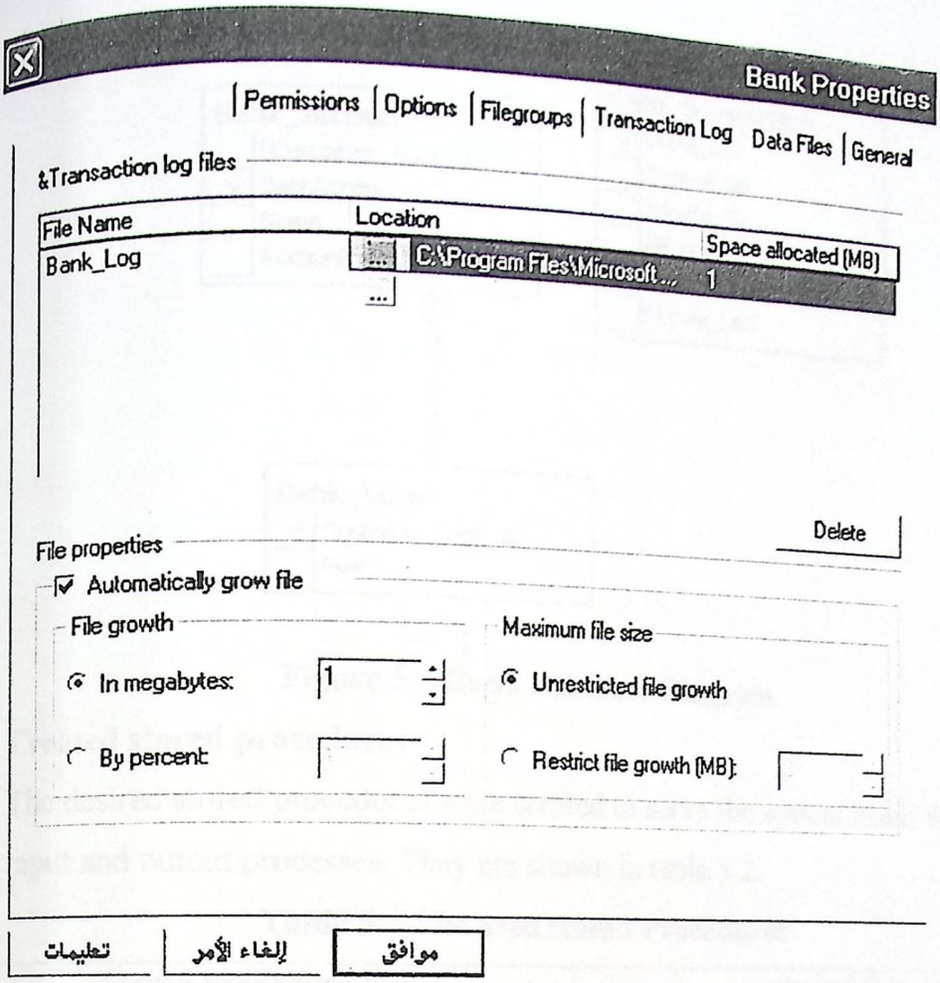


Figure 5.5 Bank Database Creation

- Create the database, Relationships, Primary and Foreign Keys: all primary and foreign keys are created to ensure database consistency.
- Create indexes to facilitate dealing with database

Figure 5.6 shows Bank database diagram

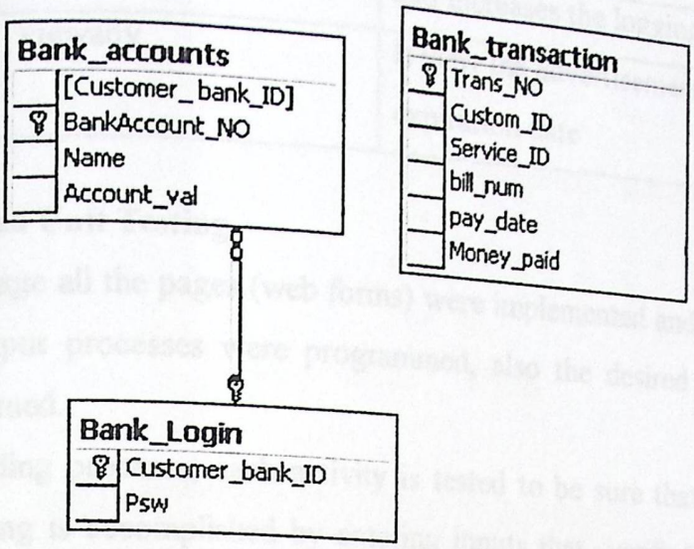


Figure 5.6 Bank Database Diagram

Created stored procedures:

The desired stored procedures were created to serve the system needs for input and output processes. They are shown in table 5.2:

Table 5.2 The used stored Procedures

Stored Procedure Name	Stored Procedure Description
genbill	It stores the bill information in the database.
getbill	It retrieves the bill information from the database to be viewed on the browser to the ser
getcon	It retrieve consumption report information from the database
login	It get the users' password and ID and check whether they are valid to log into the system
reg	It get the customer's ID and IDC and check whether he is a valid HEPCO customer currently
reg1	It register the user as a HEPCO e-service valid user
updateweb	It makes the status of the logged in user as true

	and increases the logging in times.
viewadv	It view the advertisement with according to the expiration date

5.6 Coding and Unit Testing

In coding stage all the pages (web forms) were implemented and all of the input and output processes were programmed, also the desired functions were programmed.

During coding process , each activity is tested to be sure that it works properly, testing is accomplished by entering inputs that verify the output accuracy, this helps in capturing errors from beginning of coding like logic and syntax errors, the final testing is done for the whole project . Figures 5.7 and 5.8 show an example of the programmed input and output screens.

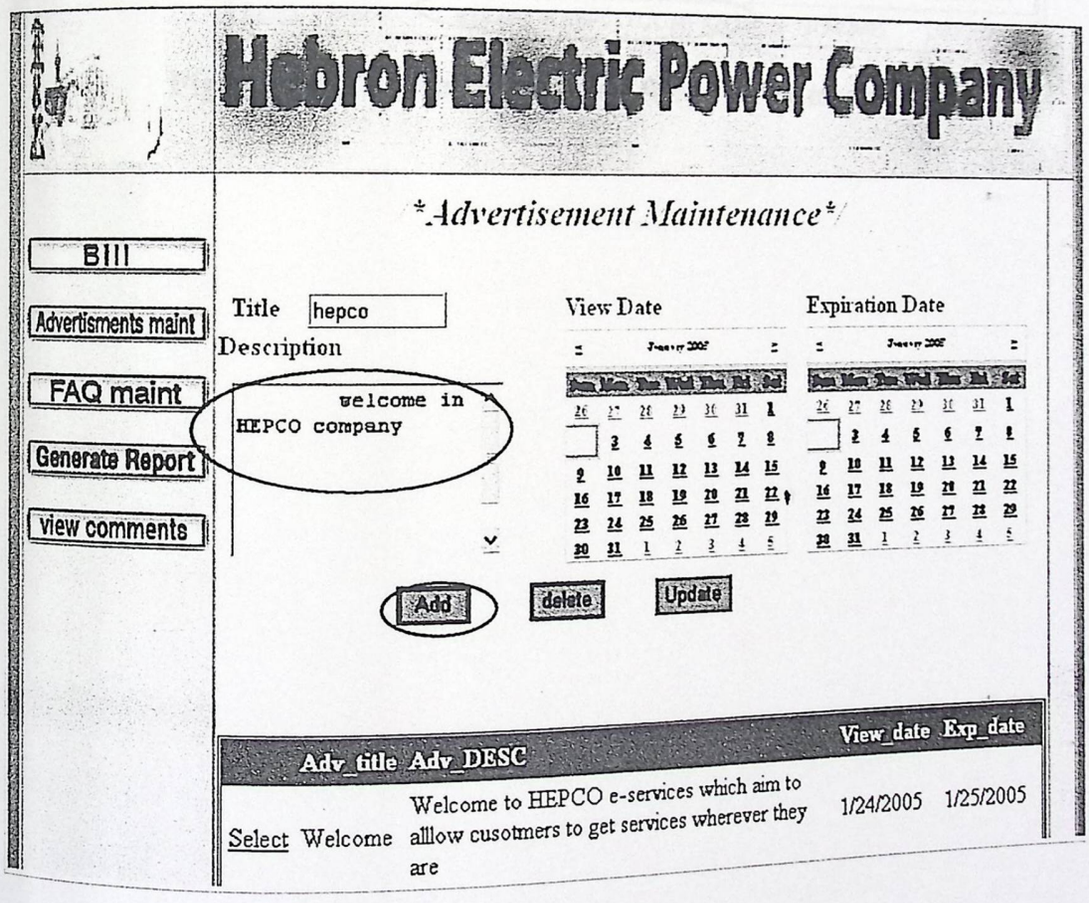


Figure 5.7 Add advertisement

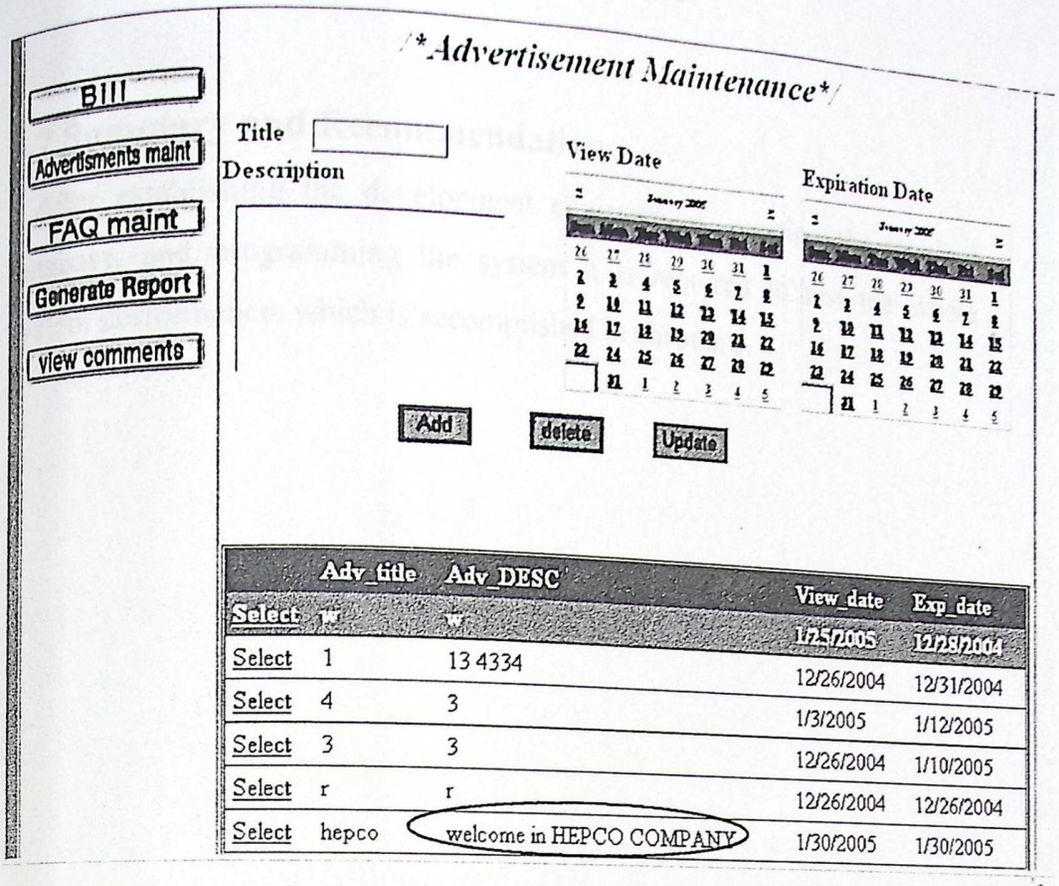


Figure 5.8 The result of adding the advertisement

5.7 Summary and Recommendation

After establishing the development environment, creating the required database, and programming the system it is required to test the whole system performance, which is accomplished in chapter 6.

Chapter

Six

Testing

6.1 Introduction

After coding and implementation stage, the system must be tested, the testing process of the system is the process of validation and verification of the system's components separately and for the system as one unit.

This section is concerned with testing the system to ensure that it performs as expected to be.

This process is done in four successive levels, these levels are:

1-Module and Unit Code Testing: where each operation was tested individually to ensure that it operates as expected, here thread testing was used.

2-System Integration Testing: Testing the project when all SW components are combined.

3-System Testing: Testing the project when all HW, SW, Procedures, and Manual components are combined together.

4-Acceptance Testing: called Validation or verification ، it is to test the project against the original project requirement.[7]

6.1.1 Testing Schedule

During programming the required testing for each accomplished process is done, and at the end of the programming the testing is done for the whole system as one unit, table 6.1 shows the schedule for testing.

Table 6.1 Testing Scheduling

Estimation time / Testing process	1st week	2nd week	3rd week	4th week	5th week	6th week	7th week
Module and Unit Code Testing							
System Integration Testing							
System Testing							
Acceptance Testing							

Virtual work

Actual work

6.2 Module and unit code testing

All system units and modules were tested against its specifications using thread testing, the test ensured that the units and modules performed as expected.

The following are some samples for module testing and its associated results using thread testing:

1-The testing for "Login" module.

Testing Method: Path testing

Conformance to specifications: testing results has showed that the module performance as expected, this is explained in table 6.1 and the flowchart for testing process is shown in figure6.1.

TABLE 6.2 Login Test Cases

Test cases	Test data		Expected output	Actual output
	Login ID	password		
b-c-d-e-f-g-h	123456	qwerty	accepted user as adminstrater type	accepted user as adminstrater type
b-c-d-k-l-m-n	2345	234567	accepted user to get services	accepted user to get services
b-c-d-i-b	2468	234567	Invalid username or password	Invalid username or password
b-c-d-j-b	–	–	username or password is not filled in	username or password is not filled in

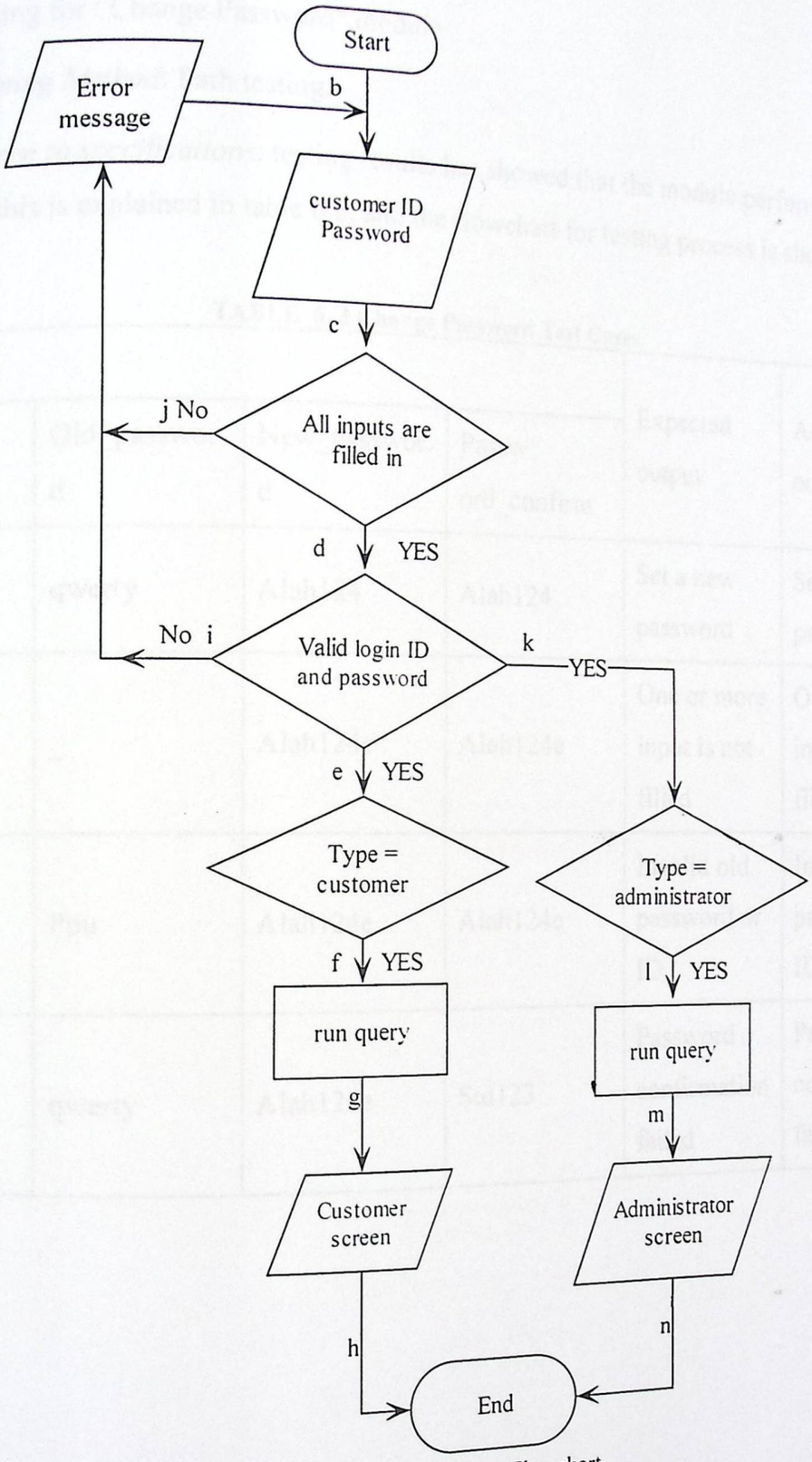


Figure 6.1: Login Testing Flowchart

2- The testing for "Change Password" module.

Testing Method: Path testing.

Conformance to specifications: testing results has showed that the module performance as expected, this is explained in table 6.2, and the flowchart for testing process is shown in figure6.2.

TABLE 6. 3 Change Password Test Cases

Test cases	Test data				Expected output	Actual output
	ID	Old_password	New_password	Password_confirm		
-c-d-e-f-g-h-i	123456	qwerty	Alah124	Alah124	Set a new password	Set a new password
b-c-m-n-o	123456	_	Alah124e	Alah124e	One or more input is not filled	One or more input is not filled
bcdlno	123456	Ppu	Alah124e	Alah124e	Invalid old password or ID	Invalid old password or ID
bcdefi	123456	qwerty	Alah124e	Std123	Password confirmation failed	Password confirmation failed

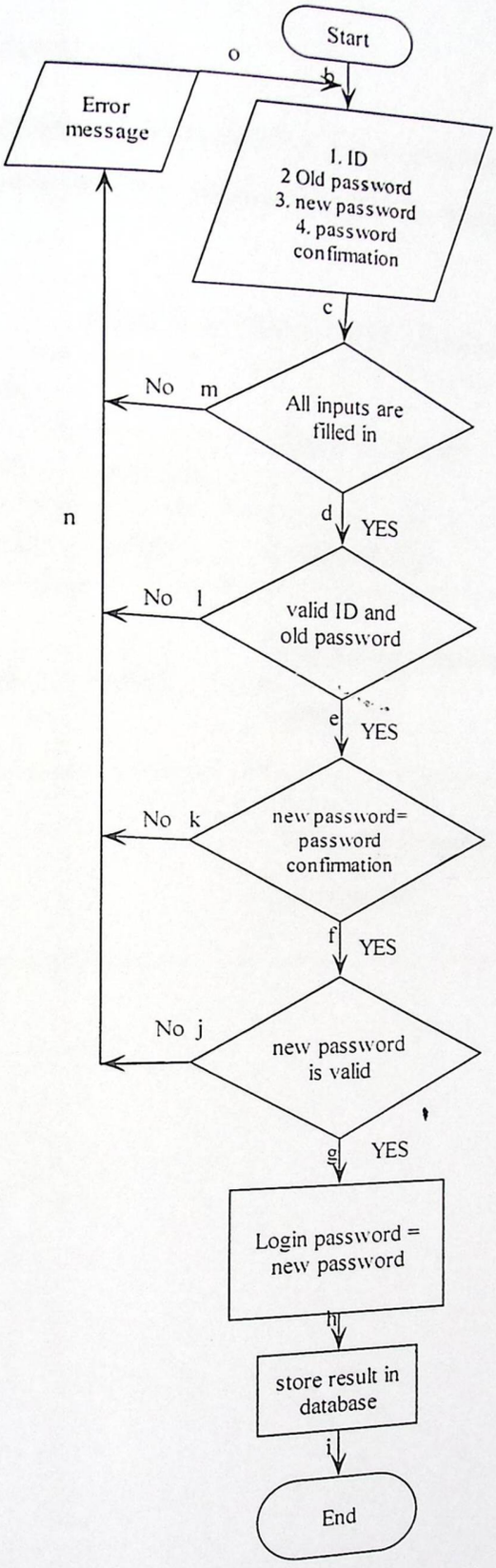


Figure6.1: Change Password Flowchart

1- The testing for "Bank Login" module.

Testing Method: thread testing

Conformance to specifications: testing results has showed that the module performance as expected, this is explained in table 6.3, and the flowchart for testing process is shown in figure6.3.

TABLE 6. 4 Bank Login Test Cases

Test cases	Test data		Expected output	Actual output
	Login ID	password		
b-c-d-e-f	12345678	asdfgh	accepted user	accepted user
b-c-d-g	1234586	asdfgh	Invalid username or password	Invalid username or password
b-c-h-g	–	–	username or password is not filled in	username or password is not filled in

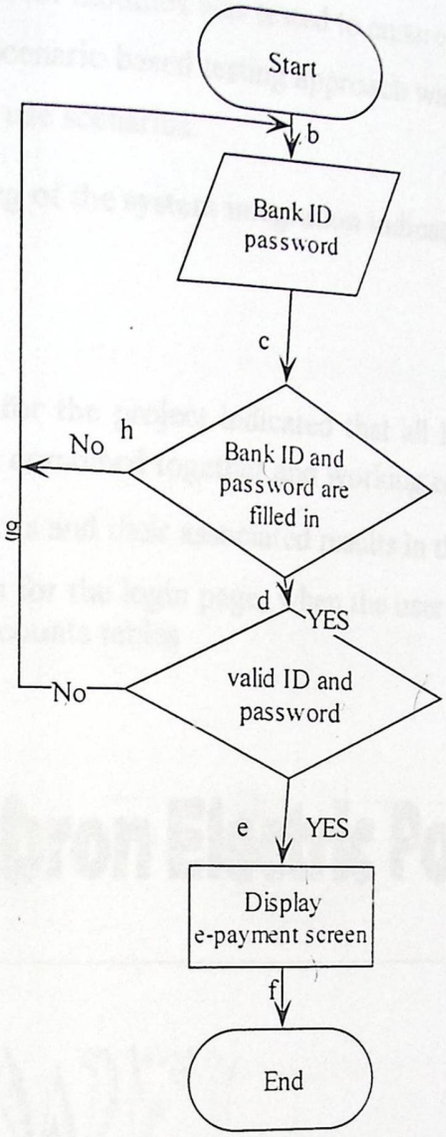


Figure 6.3: bank login Flowchart

6.3 System Integration Test

Here the integration of all modules was tested to ensure that the whole system performs as expected, scenario based testing approach was used so that the system integrated based on the use scenarios.

Test results: the testing of the system integration indicated that the system performs as expected.

6.4 System Testing

The system testing for the project indicated that all HW, SW, Procedures, and manual components are combined together and working consistently.

Here are some snapshots and their associated results in the database:

Figure 5.4 shows the inputs for the login page, when the user login, user's password and ID is checked through web_accounts tables

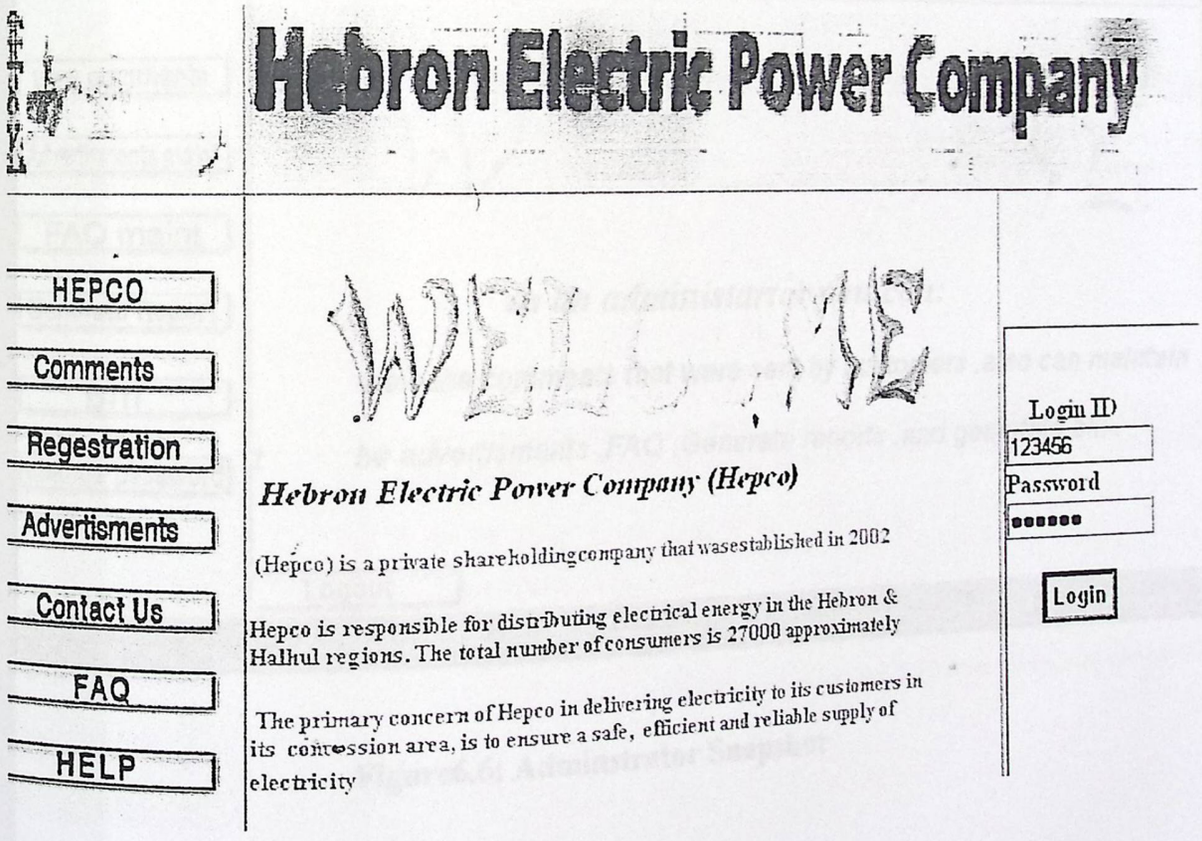


Figure 6.4 Login snapshot

Figure 5.6 shows that the password is encrypted

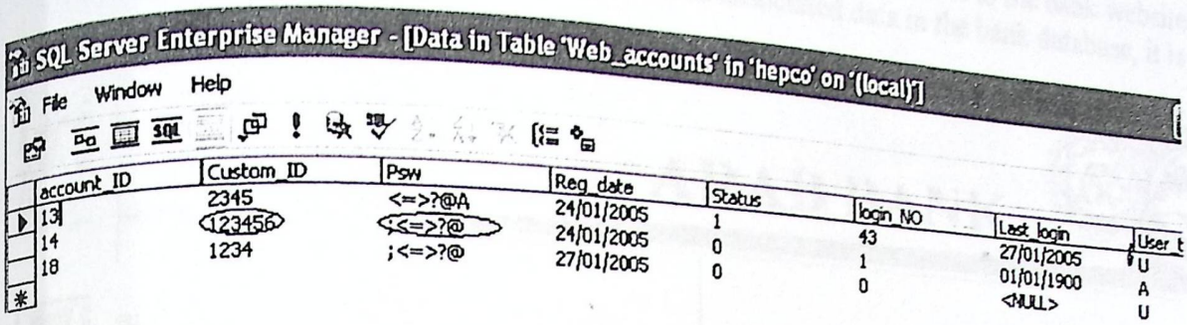


Figure 6.5: Web_accounts table

Figure 6.6 shows the page after logging in as an administrator; the administrator screen describes briefly what the administrator can do.

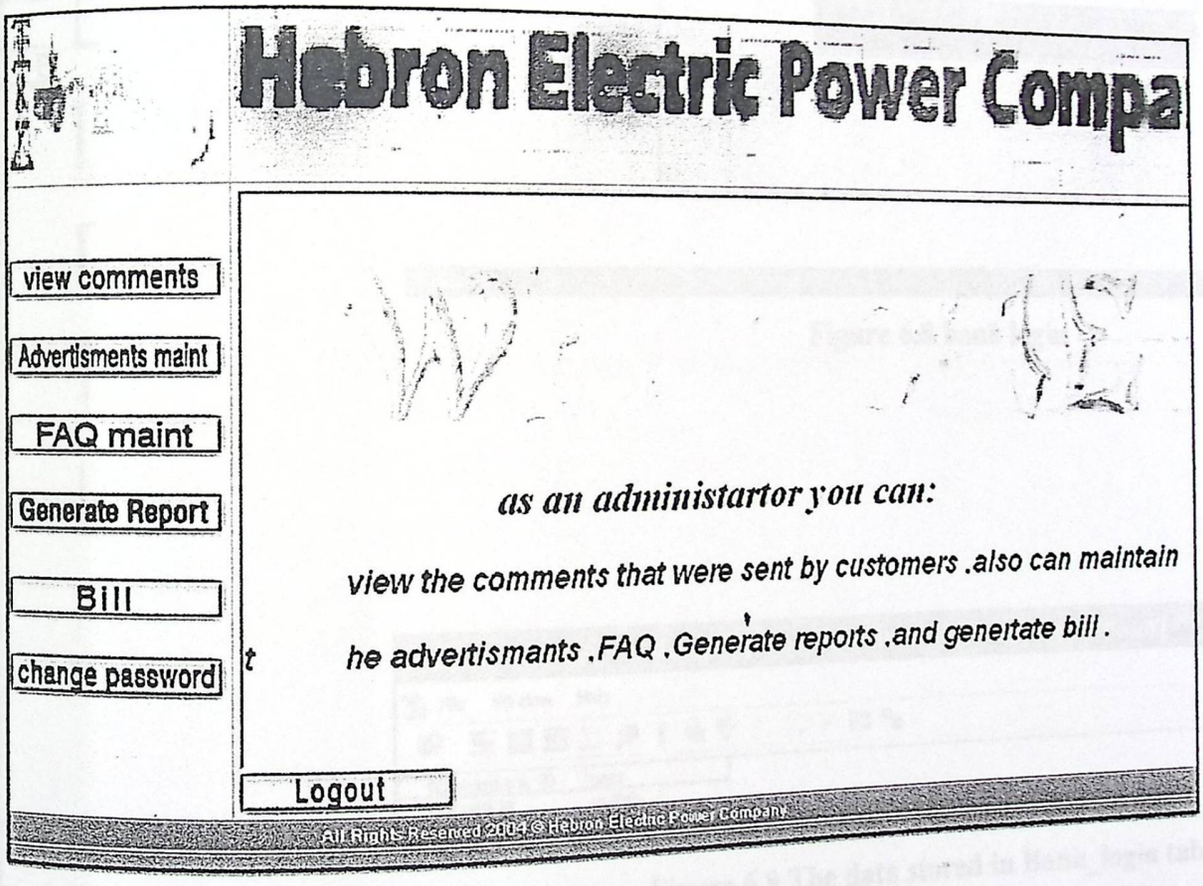


Figure6.6: Adminstrator Snapshot

Figure 6.8 shows the logging in process to the bank website, and the input data with its associated data in the bank database, it is shov

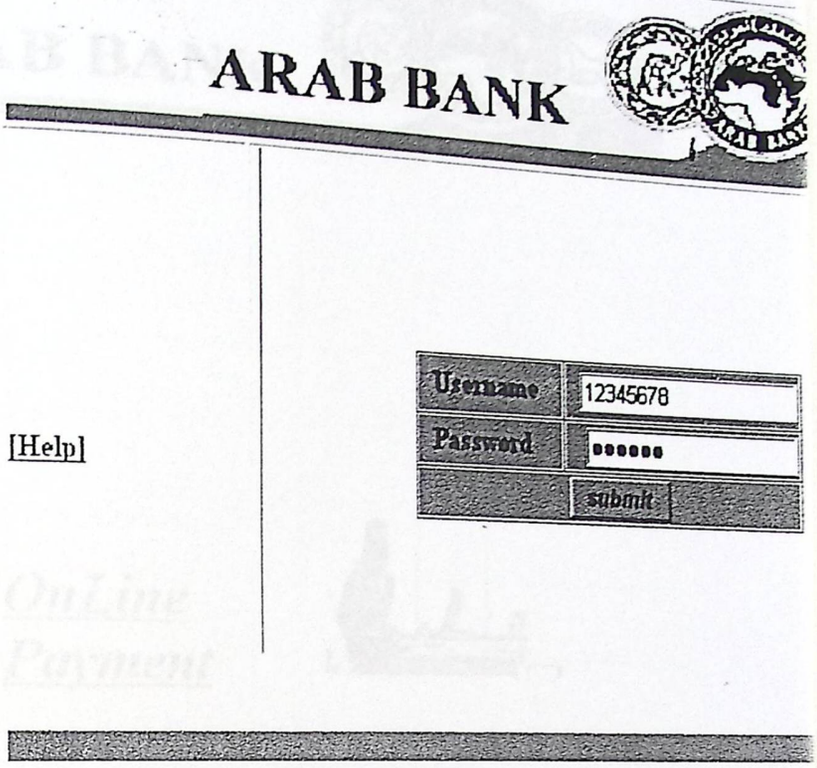


Figure 6.8 bank login

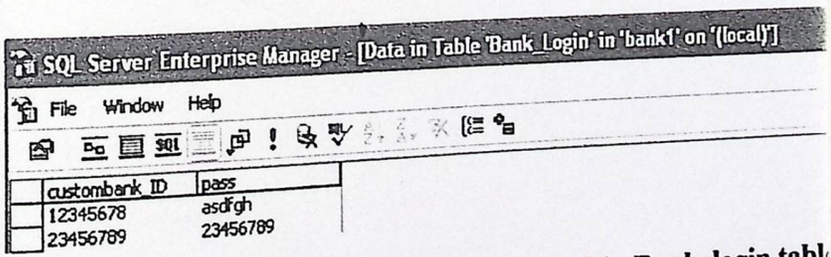


Figure 6.9 The data stored in Bank_login tabl

After the user login to the bank website he can pay his bill and figure 6.10 shows the screen which the user can see.

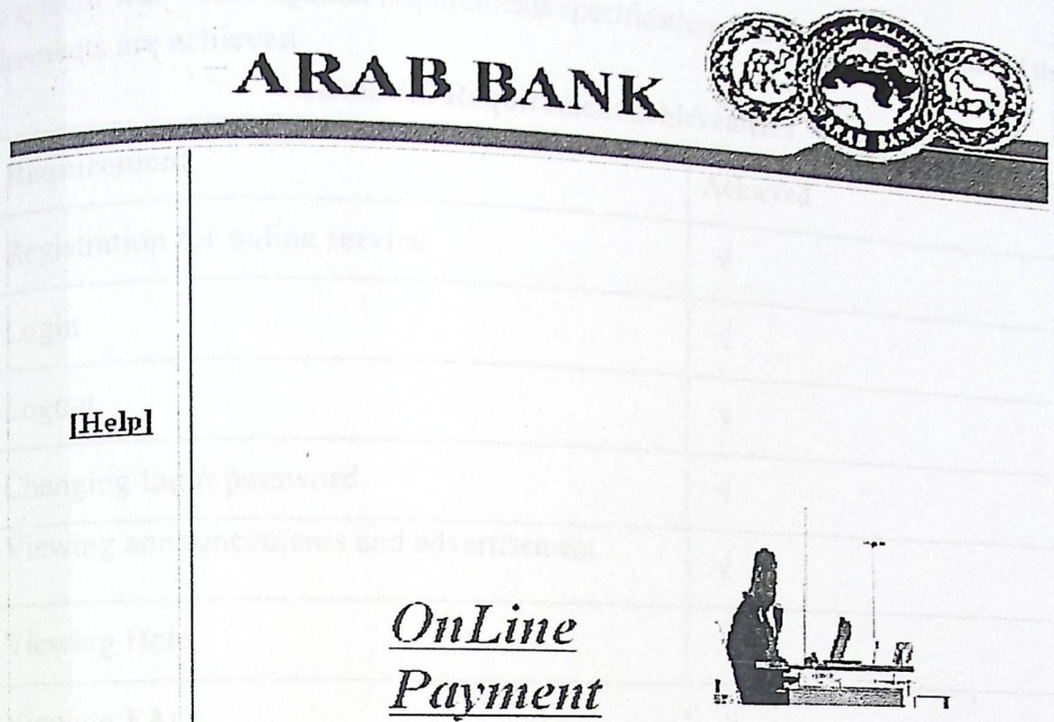


Figure 6.10 logging in to bank website

6.6 Acceptance Testing

The system was tested against requirements specifications and table 6.4 shows that all the requirements are achieved.

Table 6.5 Requirement achievements

Requirement	Achieved
Registration for online service	√
Login	√
Logout	√
Changing login password	√
Viewing announcements and advertisement	√
Viewing Help	√
Viewing FAQ	√
Reporting comments	√
Viewing company contact information	√
Viewing Bill	√
Viewing Consumption Report	√
Viewing Bills Status	√
Prompting to Bank Website	√
Bank Login	√
Bank Help	√
Online Payment Using Bank Website	√
Bank confirmation.	√
Viewing Comments	√
. FAQ Maintainace	√
Reports Generation	√

Announcements Maintenance	√
Bill's Generation	√
Generating XML transactions	√

Test results: the testing of the system indicated that the system performs as expected

6.7 Summary and recommendation

After testing the system using the four steps descriped above , and ensuring that its performance matches the expected results, its determined that the system is ready now . for demployment.

Chapter

Seven

Maintenance

7.1 Introduction

This chapter describes how to start working with the system; the establishment of the environment that the system will work in, what is the process of deployment, and the maintenance plan.

7.2 Establishment of production Environment

The main goal of HEPCO new system is to add an application that works over Internet to work with their local computerized system.

In order to install this system (product) a certain environment for the client and server is to be established. To operate the system many things that are necessary like the operating system, the PC's and the UPS; this is a technology that helps in case of disconnecting the power, it gives the user more enough time to save the data, and prevent the system from being scratched.

To start working with the system:

- The system needs a server that is connected to the Internet to host it.
- And a client is needed which will interact with the system; the client PC requires windows operating system and internet explorer (IE 6).
- Communication lines, which connect between the client and the server.
- Data backup, making another copy of data is important in case of any lose of it.

The user needs to do the following steps

- Turn on the PC, and on the internet explorer browser write the address that will connect him to the system
- Wait to load the system
- Get the message that insures the system is connected
- Start interacting with the system

7.4 Maintenance Plan

In case of the total failure, the system has a backup data, also in case of errors occurrence the programmer team can follow the code and maintain it, a single error or multiple errors may occur will be solved and they go to a specific page . For error handling if an error occurs during the implementation, the error message and a description of that error will be displayed on a web page, then the customer must call the vendor and tell him about the error.

When the programmer fix (handle) the error, he must make unit testing and integrating testing to make sure that the last modifications do not affect to the whole system performance.

Finally, the corrected pages or components must be deployed and publish it to the web server.

The program purchaser can call the programmer using a special form Fig 7.1, which contains the error information and description, and then the programmer must record each step in another form (Fig 7.2).

Software Change Request Form

Software Change Request (SCR) **Requirement No**_____**Date:** _____

Type:

( ) New Requirement

( ) Requirement Change

( ) Design Change

( ) Suggestion for Improvement

( ) other:_____

( ) System Problem

( ) User Interface Problem

( ) Documentation Correction

Description: Please attach supporting documentation for the requested change (Screen/report printouts, document pages affected, etc.)

Figure7.1 Software Change Request Form

How To Complete The MAINTENANCE LOG

This change control log form is included as a suggested format for recording and maintaining software change request data, including changes to documentation. A Detailed Status Information form is available to record supplementary details. The log and software change requests should be maintained in the Systems Project Notebook.

Definition of the LOG fields

Page No: Enter the appropriate page number of the log sheet.

Log Date: Enter the date control log was started.

System Name: Enter the name and acronym of the system to be managed.

Request No: Enter the unique sequential number assigned to each request on the request form (i.e., software change request form, etc.)

Reqmnt No: Enter the unique number of the requirement to be changed (if known) on the request form.

Date Submitted: Enter the date the request was submitted to the maintenance team.

Approval: This area is for recording request approval information obtained from the request form.

Change Approved: Enter the date the request was approved.

Change Not Approved: Enter the date the request was disapproved.

Hold (Future Enhancement): Enter the date the request was placed on "Hold."

Status: This area is for recording basic information about the status of a request.

Technical Evaluation Phase: Enter the date the technical evaluation of the request commenced.

Change In-Progress: Enter the date work began on the request. Usually, the areas "Technical Evaluation Phase" (if applicable) and "Change Approved" should be entered prior to posting the "Change In-Progress" date. Work on most requests should not be initiated without a technical evaluation and formal approval in the request form.

Canceled: Enter the date the request was canceled.

Target Date: Enter the estimated date that the request will be completed and ready for release/implementation.

Date Complete: Enter the actual date the request was implemented.

System Name:

Request No	Reqmnt No	Date Submitted	Approval			Status				
			Change Approved	Change Not Approved	Hold (Future Enhancement).	Technical Evaluation Phase	Change In-Progress	Canceled	Target Date	Date Complete

Figure7.1 Maintenance log

References

[1] www.infozech.com/product-ebill.asp

[2] www.lbwl.com/eBillFAQ.asp

[3] www.02.sbc.com/Billing_Accounts/OnLineBilling/0,,304,00.html

[4] www.hepco-pal.com

[5] http://my.dteenergy.com/paymentOptions/eBill.do

[6] Developing Microsoft ® ASP.NET web application vitual stdio.NET ®, 2310B

[7]Sammerville,Ian.software enginering-6th edition-Adison Wesley-united states of America

Palestine Polytechnic University

College of Engineering & Technology
Electrical & Computer Engineering Department

HEPCO E-Services System's User Manual

By

Hiyam Badr Tahani Darabee'
Abeer Aljabary

Supervisor
Dr. Salman Talahmeh

Hebron-Palestine

January, 2005

1. Administrator manual

System overview

HEPCO e-services is a web based database system that enables users to get over the web the services that are provided by HEPCO, to know more about these services and how to it see user manual.

As an administrator you can generate bills, maintain advertisements and frequently asked questions (FAQ), view users' comments, and generate reports (number of customers, customer profile).

System requirements

In order to install the system you need a pc (server) that is connected to the internet to host the system

Then you must attach the database to the SQL server 2000 and copy the system to the pc.

And start working with the system

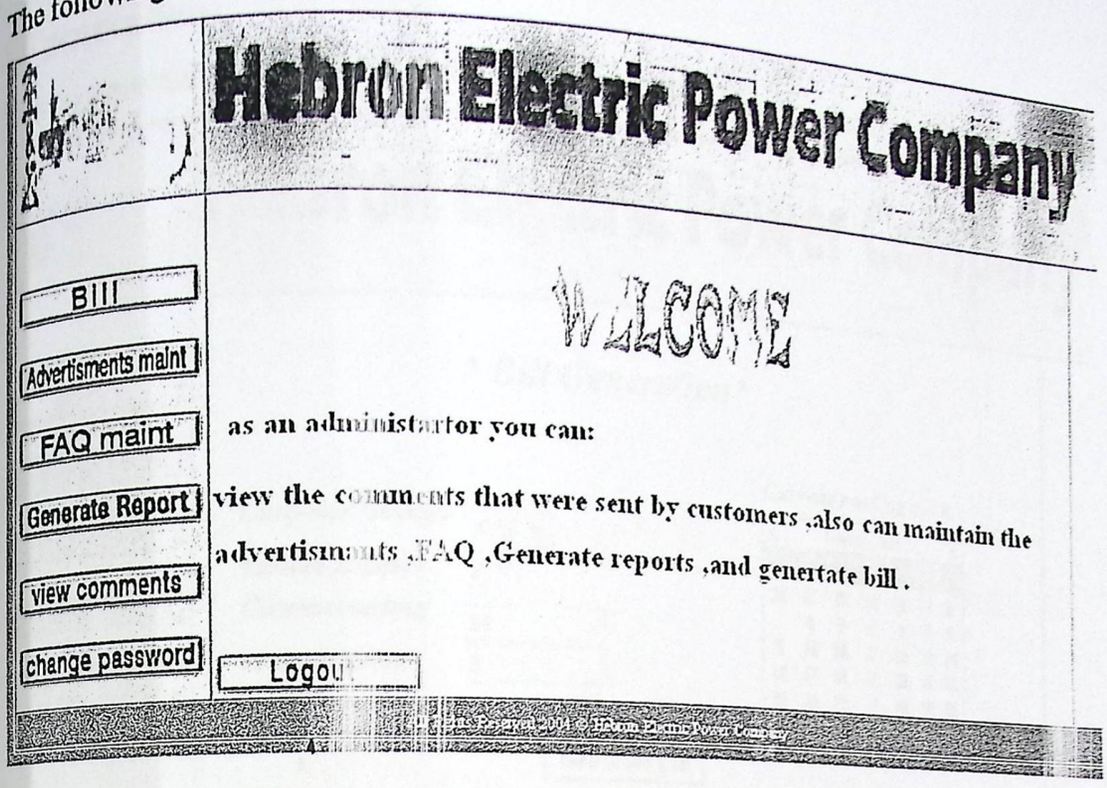
How to use the system as an administrator

Add your account to Web_accounts table in HEPCO database; don't forget to set User_type to 'A'

Then make sure that you are connected to the internet, visiting HEPCO e-services website.

Login with your ID and password, which you have already created in the SQL.

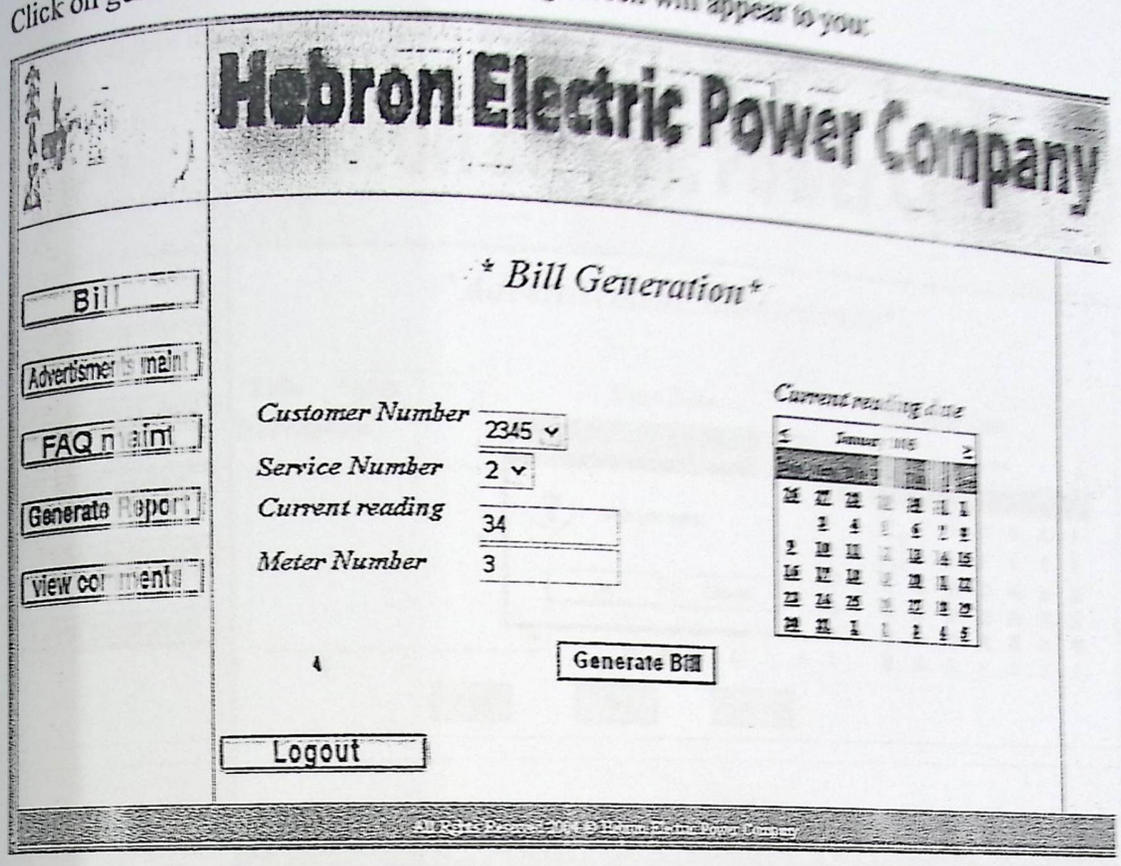
The following screen will appear:



Click on the button that performs the task that you want to perform, here are a description of how to perform these tasks:

Bill generation:

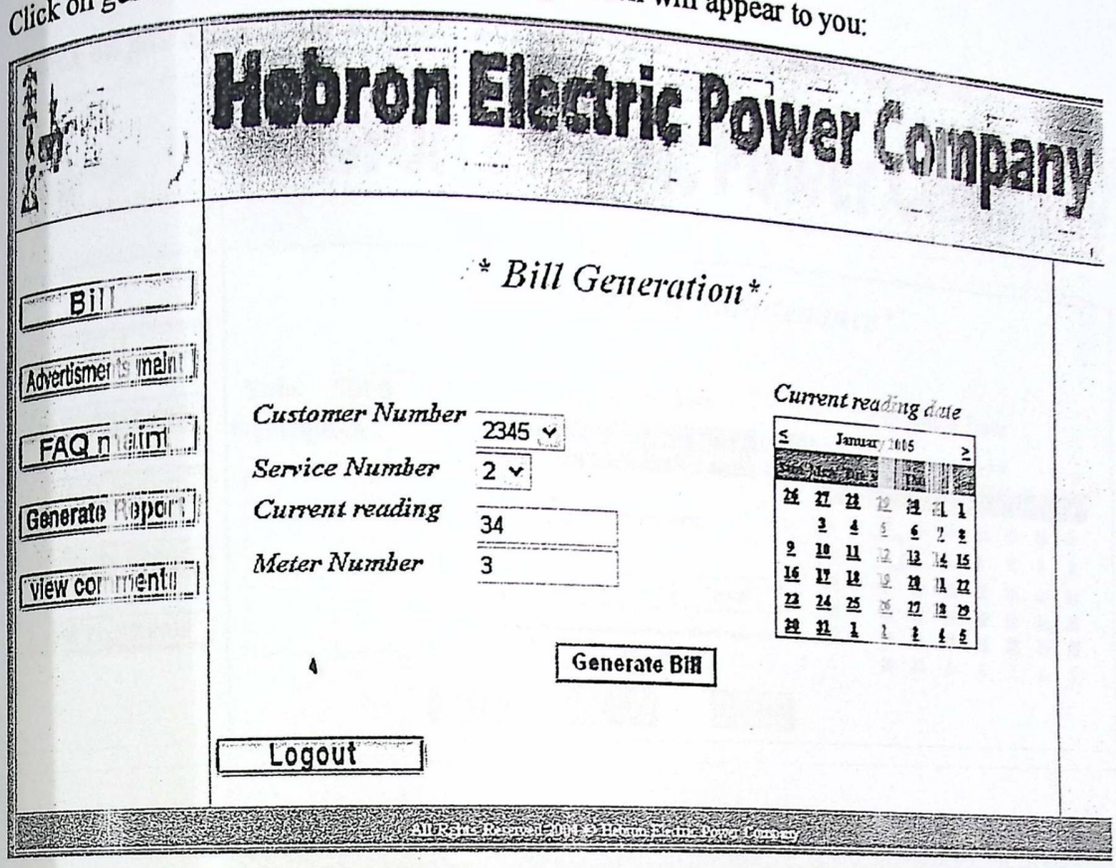
Click on generate bill button the following screen will appear to you:



Choose customer ID from the dropdown list, service ID, enter the current reading to the text box, current reading date from the calendar then click generate bill button You must be confirmed that the bill is generated.

Bill generation:

Click on generate bill button the following screen will appear to you:

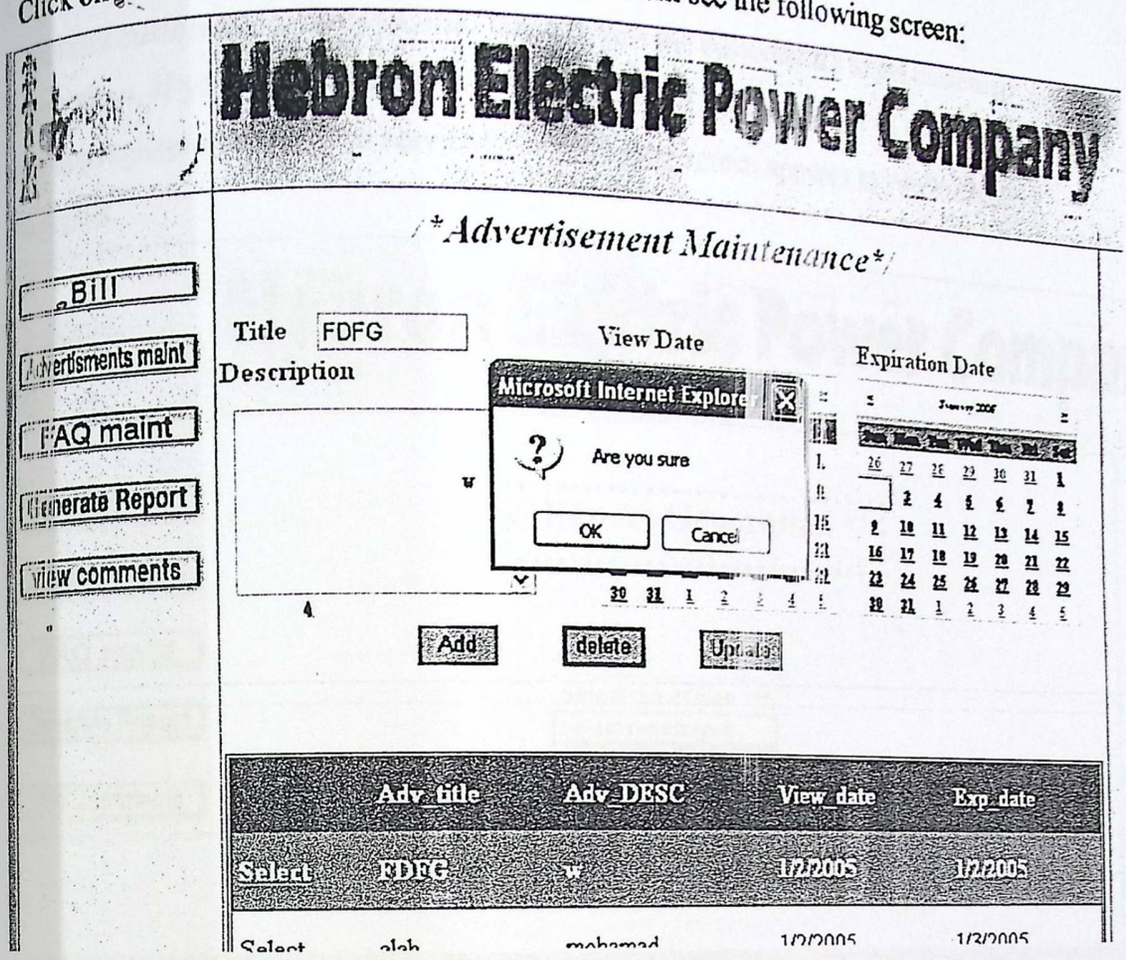


Choose customer ID from the dropdown list, service ID, enter the current reading to the text box, current reading date from the calendar then click generate bill button You must be confirmed that the bill is generated.

Advertisement maintenance

You can add, delete, and update advertisements

Click on adv maintenance button, and then you will see the following screen:



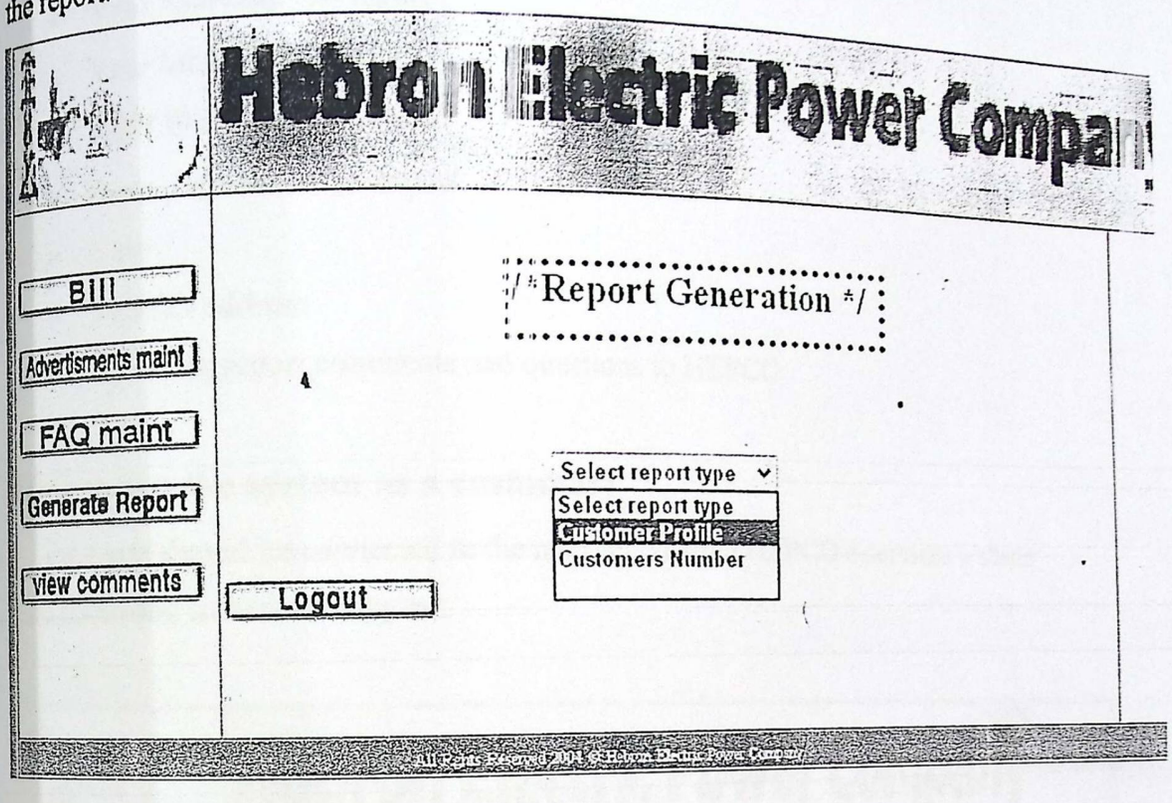
- Add advertisement: enter title, description, view date and expiration date then click add button
- Delete advertisement: select the advertisement you want to delete then click delete.
- Update advertisement: select the advertisement you want to update change its contents then click update

Generate reports

You can generate two types of reports: customers' number and customer's profile

Click on generate report, then choose report type

If u chose customer profile you must then provide the customer ID to get customer information, if you chose customers' number you will get the cumber of customers who are registered for online services, the following screen appears to you to choose the report.



2-User manual

System overview

HEPCO e-service Allow you to get HEPCO services through the web, therefore you can view:

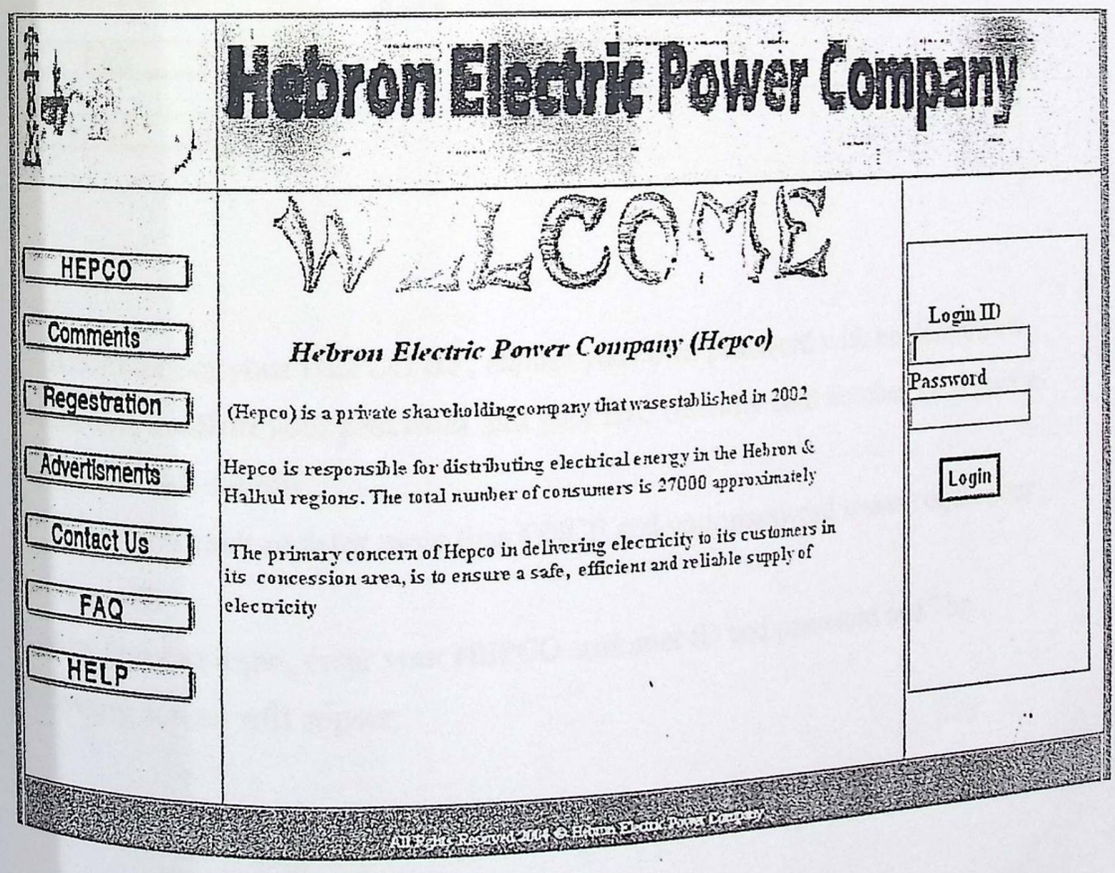
- Your bills within 6 months
- Your consumption report
- Your bills status
- HEPCO advertisements
- Frequently asked questions
- Help
- HEPCO address

Also you can report, comments and questions to HEPCO

How to use the system as a customer:

First you should be connected to the internet, visiting HEPCO e-services website

The following screen will appear:

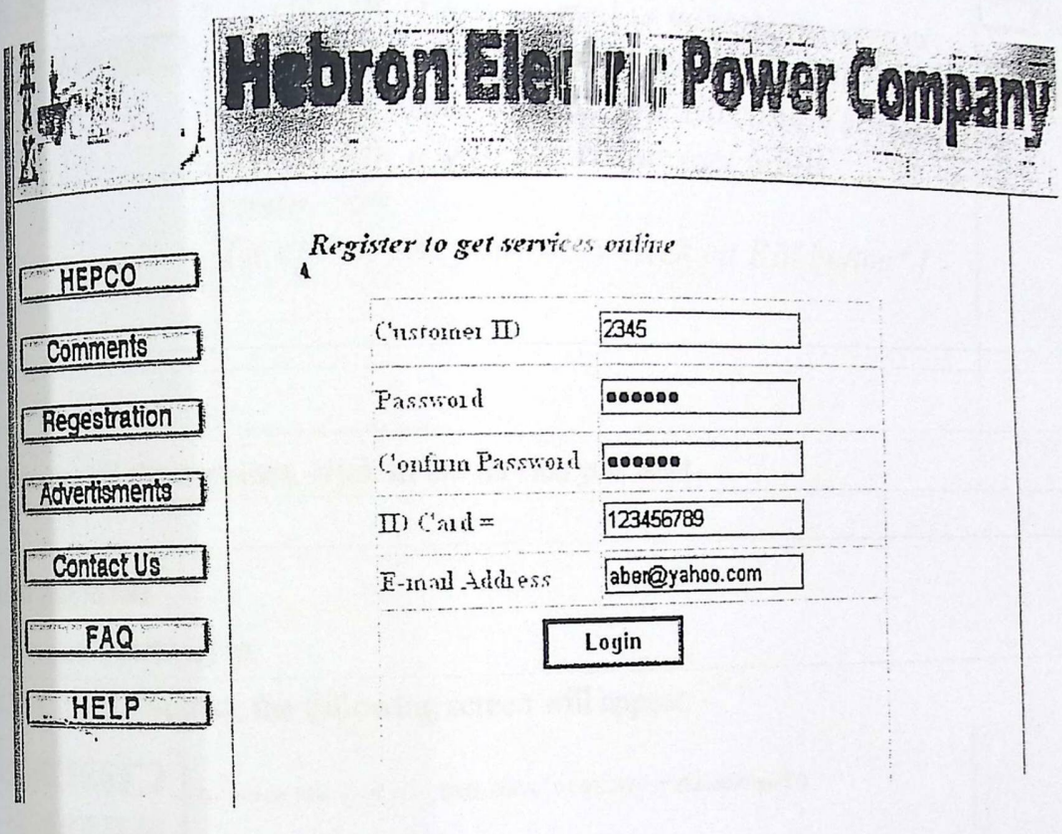


Here you can get some of the services such as FAQ, reporting your comments, reading HEPCO advertisements, viewing your bill and paying it, ... etc.

If you are a new user you must register in order to get more services, here is how to register .....

Registration:

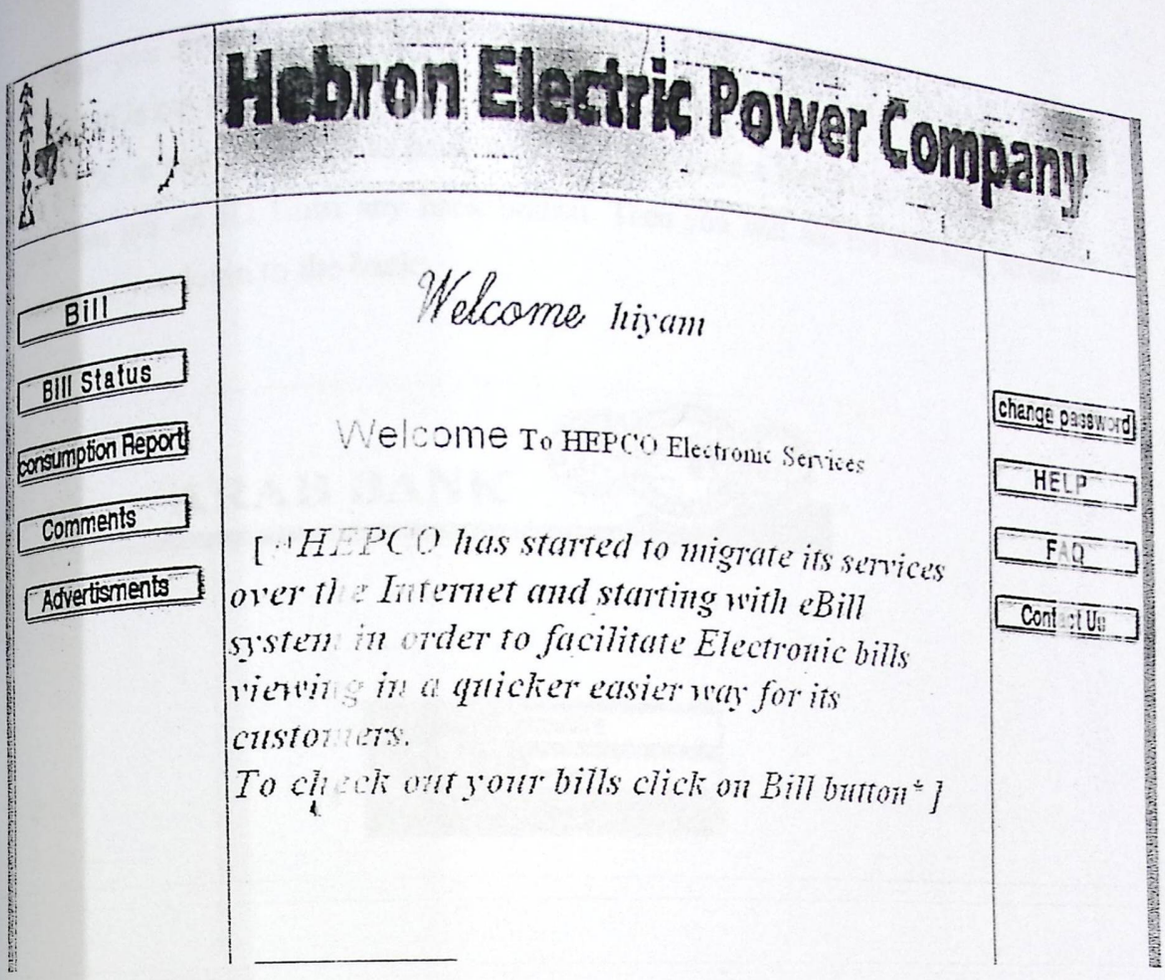
In order to register you must be a HEPCO customer. then click on the register button then the following screen will appear to you



After that put your HEPCO ID , choose your own password with not less than 6 characters, confirm your password ,put your IDC (Identity card number) and your e-mail then click submit .

Note: You can't register more than ONCE and your password mustn't equal your customer ID

Now you can login, enter your HEPCO customer ID and password and The following screen will appear:

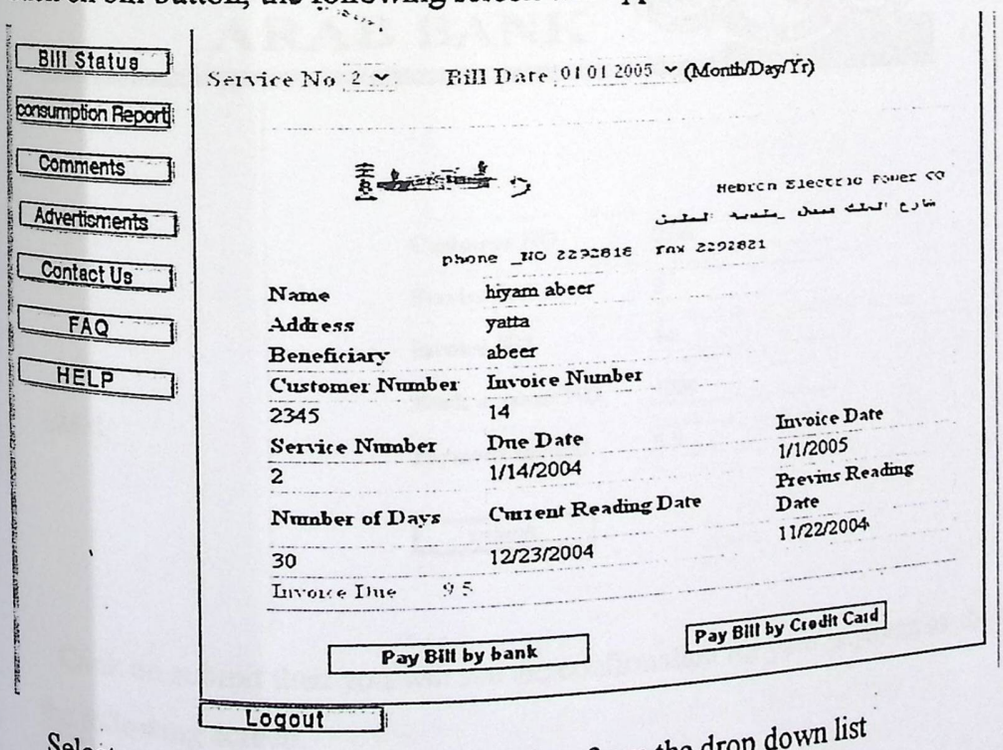


Here are all the services, click in the service you want.

Bill payment

First you must log in.

Click on bill button, the following screen will appear:

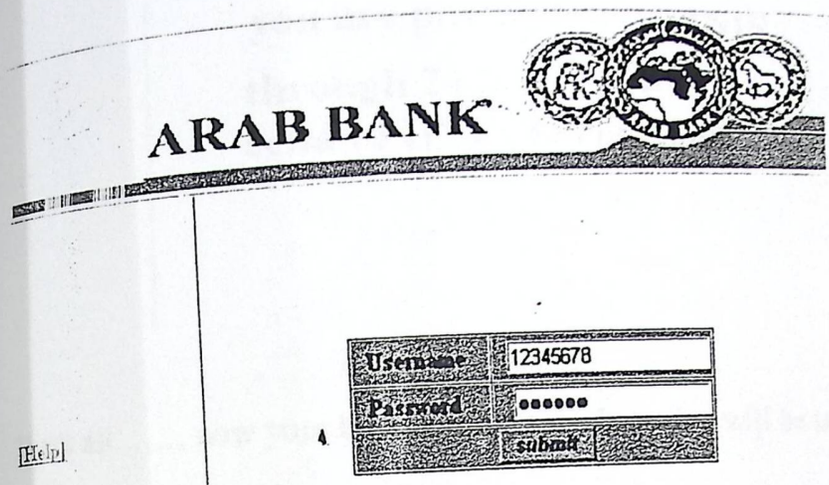


Select your service number and bill date from the drop down list

Now you can see your bill, you can print it or save it

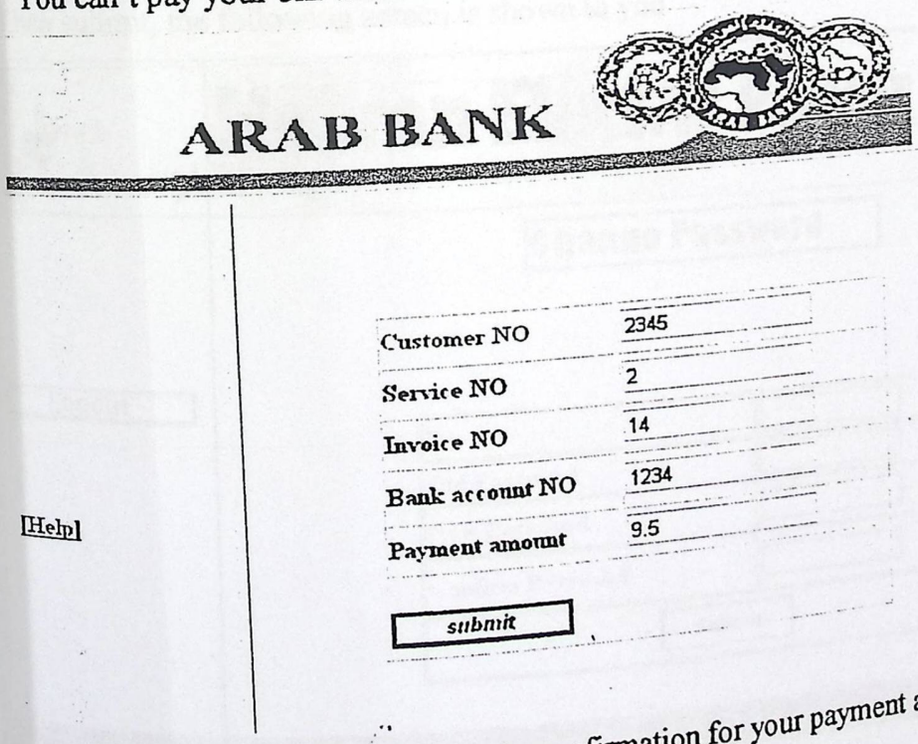
If the bill is not paid:

Click on pay bill to go to bank website, if you have a bank ID you can login else you can get an ID from any bank branch. Then you will see the following screen which let you login to the bank:

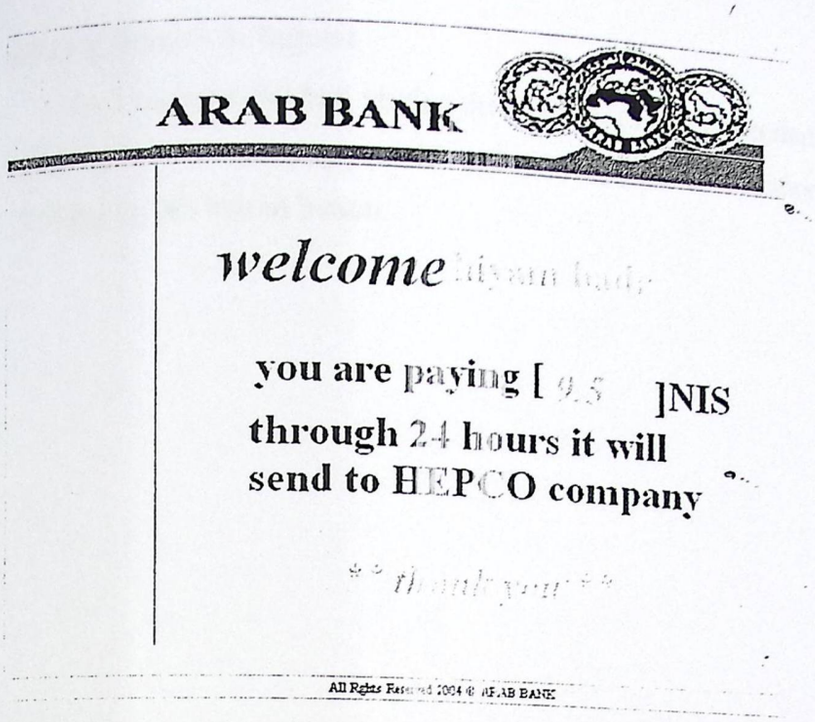


After you have been logged in you must provide your bank account number

You can't pay your bill without having a bank account!



Click on submit then you will see the confirmation for your payment as shown in the following screen.



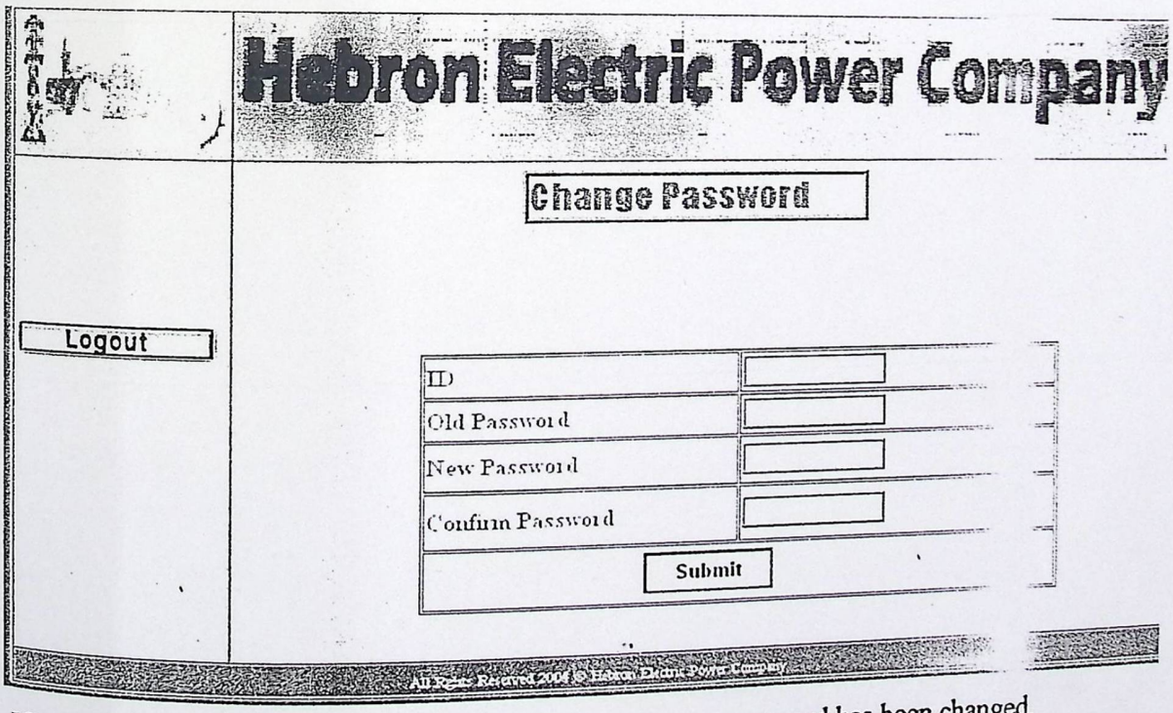
That's all……, now your bill has been paid the action will be taken within the coming 24 hours.

Change password

First you must login, click on change password

Then enter your ID and old password, new password and confirm it again

Then submit, the following screen is shown to you



If every thing is okay you should be confirmed that your password has been changed.

Don't forget to logout

After you have finished visiting the website it's recommended to logout to prevent other people from using your ID and viewing your information, you can logout by clicking on the logout button.