

Online Syllabus

Prepared By

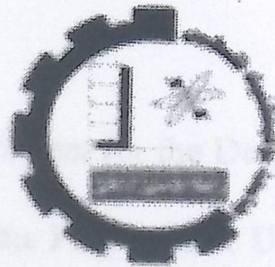
Fatten Y. Abu Arqub

Lana A. Yaghmour

Bassim A. Al-qadi

Supervised By

Dr. Mahmoud H. Alsaheb

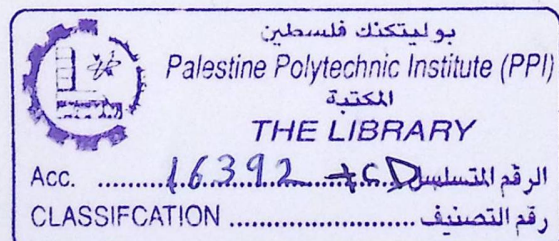


A Final Project for the Degree of B.SC

Jul-03

Supervisor Signature
Signature

Department Manager



Online syllabus

Prepared By

Fatten Y. Abu Arqub

Lana A. Yaghmour

Bassim A. Al-qadi

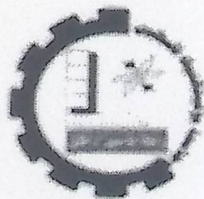
Supervised By

Dr. Mahmoud H. Alsaheb

A Final Project for the Degree of B.SC

Palestine Polytechnic University

PPU



Jul-03

Acknowledgment

Many thank to every one who helped us in our project especially our supervisor Dr. Mahmud Al-Jabir.

Dedication

And all who have helped us to reach our goals, as Yagmour on.

To our families for supporting and encouraging us all the time.

To our teachers, lecturers and friends.

Fatten Abu Arqub

Lana Yagmour

Bassim al-qadi

Acknowledgment

Many thank to every one who helped us in our project especially our supervisor Dr. Mahmud Al sahib.

And all who have helped us to reach our goals, as Yaghmour co., Arab Center for Information Technology.

We thank Mr. Abed Asalam Dwiki, and Mr. Shade Al Rjube.

TABLE OF CONTENTS

Abstract

The aim of this project is to enable the instructor to show their course syllabus on web and control it, this project help the instructor to show effective course syllabus that it's useful for student and help them to get all information about course.

On the other hand we enable the instructor to do some operation on his syllabus by adding, and updating on syllabus, this operation deal with DB so any change on content of Database is doing remotely.

Not only the instructor doing operations on course syllabus, also the administrator can add the instructor and update instructor information that is stored on Database.

TABLE OF CONTENTS

<i>Dedication</i>	<i>IV</i>
<i>Acknowledgment.....</i>	<i>V</i>
<i>Abstract</i>	<i>VI</i>
<i>List of contents</i>	<i>VII</i>
<i>List of tables.....</i>	<i>XI</i>
<i>List of figures.....</i>	<i>XII</i>
<i>Chapter One (Introduction)</i>	
<i>1-1 Overview</i>	<i>3</i>
<i>1-2 Systems importance.....</i>	<i>3</i>
<i>1-3 System objectives.....</i>	<i>4</i>
<i>1-4 Project planning</i>	<i>5</i>
<i>1-4-1 feasibility study</i>	<i>5</i>
<i>1-4-1-1 economic feasibility requirement</i>	<i>5</i>
<i>1-4-1-2 economic feasibility (development)</i>	<i>7</i>
<i>1-4-1-3 technical feasibility</i>	<i>8</i>
<i>1-4-1-4 time feasibility</i>	<i>8</i>
<i>1-4-1-5 time schedule.....</i>	<i>8</i>
<i>1-4-2 schedule</i>	<i>9</i>

Chapter two (system requirement)

2-1 functional requirement10

2-1-1 definition requirement10

2-1-2 requirement specification10

2-2 non functional requirement13

product requirement13

process requirement.....13

external requirement.....13

chapter three (system analysis)

3-1 introduction16

3-2 system feature16

3-3 system tool.....16

3-3-1 ASP active server page16

3-3-2 VBS visual basic script17

3-4 content of the course syllabus17

3-5 view method19

chapter four (system design)

4-1 introduction.....24

4-2 design models24

4-2-1 ER – model24

4-2-2 context diagram26

4-2-3 structure chart	26
4-2-4 database design.....	27
4-2-5 data dictionary	29
4-2-6 data flow diagram.....	44
4-2-7 flow chart	46
4-3 system interface	56
 chapter five (implementation)	
5-1 introduction	90
5-2 software needed	90
5-2-1 windows 2000 advanced server	90
5-2-1-1 open data base connectivity (ODBC).....	91
5-2-2 Microsoft office 2000.....	95
5-2-3 active server pages(ASP).....	95
5-2-4 VB script.....	95
5-2-5 structured query language (SQL).....	96
5-3 Implementation of the system	96
5-3-1 initial implementation of the system	96
5-3-2 implementation of system pages	99
5-3-3 implementing ASP codes	100
 chapter six (Testing and Codes)	
6-1 introduction	103
6-2 unit testing	103

<i>6-3 integration testing.....</i>	<i>104</i>
<i>6-4 validation testing</i>	<i>105</i>
<i>6-5 acceptance testing</i>	<i>105</i>
<i>chapter seven (conclusion and future work</i>	
<i>7-1 conclusion</i>	<i>109</i>
<i>7-2 future work</i>	<i>109</i>
<i>References</i>	<i>110</i>
<i>Appendix</i>	<i>111</i>

List of tables

Table 1.1 software cost table..... 5
Table 1.2 hardware cost table6
Table 1.3 software cost table(development)7
Table 1.4 hardware cost table (development).....7
Table 1.5time schedule table8
Table 1.6 schedule table..... 9
Table 6.1 show results for unit testing to login operation .. 104
**Table 6.2 show results for integration testing to adding book
.....105**

Figure 1.1 screenshot of the system content45
Figure 1.2 flow chart for add new course46
Figure 1.3 flow chart for join instructor and course47
Figure 1.4 flow chart for update course information48
Figure 1.5 flow chart for add new instructor49
Figure 1.6 flow chart for add new office50
Figure 1.7 flow chart for add office hour52
Figure 1.8 flow chart for edit course52
Figure 1.9 flow chart for edit office hour53
Figure 1.10 flow chart for edit join office with instructor ...54
Figure 1.11 flow chart for join office with instructor 55
Figure 1.12 flow chart 56
Figure 1.13 screenshot for login page 57
Figure 1.14 screenshot for register page 58
Figure 1.15 screenshot for operation page(instructor) 59
Figure 1.16 screenshot for instructor form 60
Figure 1.17 flow chart61

List of figures

<i>Figure 3.1 data control diagram (student)</i>	21
<i>Figure 4.1 ER Model of database</i>	25
<i>Figure 4.2 context diagram</i>	26
<i>Figure 4.3 structure chart</i>	26
<i>Figure 4.4 instructor validation and display the operation form</i>	44
<i>Figure 4.5 add new syllabus</i>	45
<i>Figure 4.6 update the syllabus content</i>	45
<i>Figure 4.7 flow chart for add new course</i>	46
<i>Figure 4.8 flow chart for join instructor and course</i>	47
<i>Figure 4.9 flow chart for update course information</i>	48
<i>Figure 4.10 flow chart for add new instructor</i>	49
<i>Figure 4.11 flow chart for add new office</i>	50
<i>Figure 4.12 flow chart for add office hour</i>	52
<i>Figure 4.13 flow chart for edit course</i>	52
<i>Figure 4.14 flow chart for edit office hour</i>	53
<i>Figure 4.15 flow chart for edit join office with instructor</i> ...	54
<i>Figure 4.16 flow chart for join office with instructor</i>	55
<i>Figure 4.17 home page</i>	56
<i>Figure 4.18 administrator login page</i>	57
<i>Figure 4.19 administrator operation page</i>	58
<i>Figure 4.20 administrator operation page(instructor)</i>	89
<i>Figure 4.21 add new instructor form</i>	60
<i>Figure 4.22 fined instructor</i>	61

<i>Figure 4.24 join new office form.....</i>	<i>62</i>
<i>Figure 4.25 add new course form.....</i>	<i>64</i>
<i>Figure 4.26 link course with instructor form</i>	<i>65</i>
<i>Figure 4.27 course list page.....</i>	<i>66</i>
<i>Figure 4.28 course detail</i>	<i>67</i>
<i>Figure 4.30 navigation design for administrator.....</i>	<i>68</i>
<i>Figure 4.31 instructor login page</i>	<i>69</i>
<i>Figure 4.32 welcome page</i>	<i>70</i>
<i>Figure 4.33 instructor's course list page</i>	<i>71</i>
<i>Figure 4.34 add new syllabus form.....</i>	<i>73</i>
<i>figure 4.35 instructor syllabus list</i>	<i>74</i>
<i>figure 4.36 list of syllabus content</i>	<i>75</i>
<i>figure 4.37 form to add syllabus content</i>	<i>76</i>
<i>figure 4.39 course list page</i>	<i>78</i>
<i>figure 4.40 course detail page</i>	<i>79</i>
<i>figure 4.41 syllabus list</i>	<i>80</i>
<i>figure 4.42 course syllabus page</i>	<i>81</i>
<i>figure 4.43 exam page</i>	<i>84</i>
<i>figure 4.44 home work page</i>	<i>84</i>
<i>figure 4.45 common question page</i>	<i>86</i>
<i>figure 4.46 navigation design for view course syllabus content</i>	<i>87</i>
<i>Figure 5.1.....</i>	<i>92</i>
<i>Figure 5.2</i>	<i>93</i>
<i>Figure 5.3 database web require some layers from software's to treat it</i>	<i>94</i>

Chapter One

Introduction

1-1 Overview

1-2 system importance

1-3 system Objectives

1-4 Project planning

Chapter 1

Introduction

Chapter One

Introduction

1-1 *Overview*

1-2 *system importance*

1-3 *System Objectives*

1-4 *Project planning*

1-1 Overview

Nowadays, the usage of internet in our society is increasing, because of the important services that it provides.

Over the years, the development of internet application becomes more flexible and cleverer than the previous applications, also there are a tool that enables the programmers to control many operations remotely, for example there are tools that administrator control and manage the database remotely.

In our project we will make use of the existing tools and features to enable the lecturer to show the syllabus of the courses and administrate the content of database remotely.

We use some of the internet tools and programming languages to accomplish the attended work effectively.

1-2 System Importance

In our project the main objective is to show course syllabus on web & control content of course syllabus, by using form for adding & updating on the syllabus.

So to make the updating and adding operation easy we use web page written in asp codes. We use asp in our system because it give us dynamic page not static page, and we take the data from DB remotely.

The lecture will be able to update the content of the DB remotely without any need to republish to the syllabus page;

Why we use dynamic page:

In this page we build one form for add, update operation that is used by any instructor.

This dynamic page allow us to make update on the form one time because all instructor use this form, but when we use static page for many instructor, if we want to update any thing for these instructor we want to do update operation many times for every static page that is not easy.

1-3 System Objectives

The main objective of the project is to show course syllabus on web and control the content of course syllabus.

Use some tools of e-learning to develop the traditional learning.

1-4 Project planning

1-4-1 Feasibility Study

1-4-1-1 Economic Feasibility (requirement)

The cost of this project is divided into:

Software cost

Software	Cost
Windows 2000 Advanced server	5000\$
Windows 2000 Professional	319\$
SQL Server	150\$
Office2000	150\$
Total	5619\$

Table 1.1 Software cost Table

Hardware cost

No.	Component	Specifications	Cost
1	Pentium III	1000MHz 256 MB RAM 20 GB hard disk Monitor Mouse and Keyboard Modem Network card	900\$
2	Pentium III	600 MHz 128 MB RAM 10 GB hard disk Monitor Mouse and Keyboard Modem Network card	600\$
3	Internet Account	Four months	85\$
		Total	1585\$

Table 1.2 Hardware cost Table

1-4-1-2 Economic Feasibility (development)

The cost to develop this project is divided into:
Software cost:

Software	Cost
Windows 2000 professional	319\$
Office 2000	150\$
Total	469\$

Table 1.3 software cost table development

Hardware cost:

NO.	Component	Specification	cost
1	Pentium III	600 MHz 128 MB RAM 10 GB hard disk Monitor Mouse and Keyboard Modem Network card	600\$
		Total	600\$

Table 1.4 hardware cost table development

1-4-1-3 Technical Feasibility

From hardware we need two computers and other development tools. One power full personal computer (PC) to be as server, the other (PC) as a client to show effectively that it works as client server architecture.

The needed software is windows 2000 advanced server, Microsoft Internet Information Server 5.0 (IIS), Microsoft Office 2000, Active Server Pages (ASP), Hyper Terminal Markup Language (HTML), and SQL Server.

1-4-1-4 Time Feasibility

The expected time for this project to be completed is four months, and we expect to finish work during specific time.

1-4-1-5 Time schedule

T1	Sys. Definition	1 Week
T2	Req. Analysis	2 Weeks
T3	Design	8 Weeks
T4	Coding	8 Weeks
T5	Testing	4 Weeks
T6	Documentation	16 Weeks

Table 1.5 time schedule table

1-4-2 Schedule

For this project we distributed the tasks among 16 weeks as follows:

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

Table 1.6 Schedule Table

After analyzing the requirements and understanding the idea we are able to make the preparation of the project:

- Install Windows 2000 Advanced server.
- Install IIS, Active Directory.
- Install Windows 2000 Professional.
- Install Microsoft Office 2000.

Chapter Two

System Requirements

- 2-1 Functional Requirements
 - 2-1-1 Requirements Definitions
 - 2-1-2 Requirement Specification
- 2-2 Non - Functional Requirements

Chapter two

System Requirements

Chapter Two

System Requirements

2-1 Functional Requirements

2-1-1 Requirements Definitions

2-1-2 Requirement Specification

2-2 Non - Functional Requirements

Chapter two

System Requirements

Chapter Two

System Requirements

2-1 Functional Requirements

2-1-1 Requirements Definitions

2-1-2 Requirement Specification

2-2 Non - Functional Requirements

2.2.1 System Goals

Function: connect the syllabus page & course page with DB.

Distribution: add new course syllabus data & information in DB content.

Input: course name, course no, description, topic, assignment in lab & lecturer exams

Output: student info grades references text book ID and password.

Output: offer new, old syllabus & lecturer of course.

Destination: allow the student to view the syllabus remotely.

Security: internet access.

Access: create DB user.

Port: sending: fill data in to DB.

File: access: none.

2-1 Functional Requirements

2-1-1 Requirements Definitions

1. Each syllabus page is connected with DB.
2. Allow administrator and each lecturer to update the content of the DB remotely.
3. Allow the lecturer to store the old and new syllabus of the course in DB and show in the web page.
4. Allow the student to view the syllabus page.

2-1-2 Requirements Specifications

2-1-2-1 Each syllabus page will connect with DB.

2.1.2.1 Software project

Function: connect the syllabus page & courses page with DB.

Description: add new course syllabus data & information to DB content.

Inputs: course name, course no, description, topic, assignment to lab & lecturer exams teaching method labs grades references text book ID and password.

Outputs: offer new, old syllabus & lecturer of course.

Destination: allow the student to view the syllabus remotely.

Requires: internet tools.

Pre-condition: create DB table

Post-condition: fill data in to DB.

Side-effects: none.

2-1-2-2 Allow each lecturer to update the content of the DB remotely.

2.1.2.2 Software project

Function: change the content of syllabus by the lecturer on the internet.

Description: update the content include every thing without (course name, course no, description)

Inputs: the entire data.

Outputs: updated syllabus and new syllabus.

Destination: updated form.

Requires: correct data.

Pre-condition: old syllabus.

Post-condition: new syllabus.

Side-effects: none.

2-1-2-3 Allow the lecturer to store the old and new syllabus of the course in DB and show in the web page.

2.1.2.3 Software project

Function: store & show the new old syllabus of the course in the internet.

Description: store the new, old syllabus & view it on the web.

Inputs: input data to new syllabus.

Outputs: old, new syllabus of the course.

Source: DB.

Destination: data page.

Requires: data for new syllabus.

Pre-condition: store old syllabus and create new syllabus.

Post-condition: show the old, new syllabus

Side-effects: none.

2-1-2-4 Allow the student to view the syllabus page.

2.1.2.4 Software project

Function: offer the syllabus on the internet.

Description: offer all the syllabus of any syllabus on the internet.

Inputs: select from page.

Outputs: syllabus of selected course.

Source: data and information about syllabus from DB.

Destination: new page.

Requires: correct domain name.

Pre-condition: none

Post-condition: none.

Side-effects: none.

2-2 Non-functional requirements:

We can classify the Non-functional requirements into three types, Product Requirements, Process Requirements, and the External Requirements.

❖ Product Requirements

1. Easy to access to DB by lecturer.
2. High powerful and speed display.
3. Easy to update by lecturer remotely.
4. The system must be strong against unexpected errors (recovery and periodic reviews).
5. Easy to view by the student.

❖ Process Requirements

1. Any change in the DB content will take place at the same time.
2. When the lecturer deletes any data stored on the DB this operation is not visual for him.

❖ External Requirements

1. We use password and ID as a security feature to protect the data from authorize user and we the feature of Windows NT.
2. Only instructor can update on his course syllabus.

Chapter Three
System Analysis

3-1 Introduction

3-2 System Feature

3-3 System Tool

3-4 Content of the Course Syllabus

3-5 View Method

3-6 How the instructor access

Chapter 3

System analysis

content remotely

3-1 Introduction

In this chapter we will talk about the content of the course syllabus that we use to operate this system successfully. This will be done by analyzing the (AD, ASP, VPS and MS SQL Server).

Chapter Three System Analysis

Also we will talk about the content of the course syllabus that we use to operate this system successfully. This will be done by analyzing the (AD, ASP, VPS and MS SQL Server).

3-1 Introduction

3-2 System Feature

3-3 System Tool

3-4 Content of the Course Syllabus

3-5 View Method

3-6 How the instructor accesses database content remotely

we will help to improve the performance of the system so we use windows 2000 advance server. The security in our system is not a basic or important thing. It just prevents unauthorized user to access database content and change any thing in it except the instructor.

3-2 System tool:

3-2-1 ASP active server page

"ASP stands for Active Server Pages. ASP is a program that runs inside IIS. ASP is a Microsoft Technology. An ASP file is just the same as an HTML file. An ASP file can contain text, HTML, XML, and

3-1 Introduction

In this chapter we talk about the tools we use to operate this system successfully, this will be done by anglicizing the (AD, ASP, VPS and MS SQL Server).

Also we will talk about the content of the course syllabus then analyze this content, how the student can access the course syllabus page, how the instructor can access to the database, update its content and how he can create a new account.

3-2 System feature:

This system needs to operate and build successfully a (512 MB) memory to accept a large amount of data; it needs a hard disk (scasi 10000 GB) because it's affected the performance of the system and increase the speed to perform the main operation by many users. The operating system that we use will help to improve the performance of the system so we use windows 2000 advance server. The security in our system is not a basic or important thing; it just prevents unauthorized user to access database content and change any thing in it except the instructor.

3-3 System tool:

3-3-1 ASP active server page

"ASP stands for Active Server Pages; ASP is a program that runs inside IIS, ASP is a Microsoft Technology, An ASP file is just the same as an HTML file, An ASP file can contain text, HTML, XML, and

scripts. Scripts in an ASP file are executed on the server. An ASP file has the file extension ".asp".¹

3-3-2 VBS visual basic script

"VBScript is a member of Microsoft's Visual Basic family of development products. VBScript is a scripting language for HTML pages on the World Wide Web and corporate intranets."²

3-4 Content of the Course Syllabus:

Course syllabus:

"A Course syllabus is a plan of the course, it contains all data and information that the instructor wants to clarify through the course to achieve all the goals of the course, the syllabus content is put and controlled by the instructor.

course syllabus can be a valuable organization communication and learning tool for that the course syllabus must be clear, brief and complete to help the student to complete the course requirement in a first few days of the course and develop the "individual learning contracts" with the faculty member to complete those sections of the syllabus. The nature and the content of the syllabus will vary by factor such as the instructor philosophy, the discipline and characteristics of the course, level of students, instructor norms or policies."

Below is a check list of items that might be included in a course syllabus:-

Course information: all information about the course like title, number, section, time, place, catalog description, prerequisites. All this

¹ David Brebner, Asp_tutorial\aspquiz.htm.

² Brian Jonnson, The VB Script Language, Ch 2.

information is controlled and put by the registration and we can get it from registration.

Instructor information: all information about instructor like name, all office location if he has more office like one in Abu ktela, wade alhariah and ainsara, all phone number for the offices, home and mobile, and all office hours that allow the student to connect with there instructor.

Any assumption: about needed background or preparation or current skill level for the course.

Course policies: some of these policies are put by college or university and the other by instructor and some of these policies are depend to the nature of the course like extra credit, make up work, late work, attendance, participation academic dishonesty, changes in the syllabus, cooperative or teamwork.

Description of the course: there are two descriptions from instructor and registration.

Behavioral course objectives: clarify all the objectives that the instructor want to achieve throw the course and the way to achieve this objectives.

Required material for the course: the instructor clarifies the text books he use through the course and some courses needed more than one book may for lab. Provide all information about the book like title, author, what and where to get them, what they should get and of thee reading or other materials, why they should read them.

Optional material for the course: provide information about references and useful link that provide more information and skills to the students.

information is controlled and put by the registration and we can get it from registration.

Instructor information: all information about instructor like name, all office location if he has more office like one in Abu ktela, wade alhariah and ainsara, all phone number for the offices, home and mobile, and all office hours that allow the student to connect with there instructor.

Any assumption: about needed background or preparation or current skill level for the course.

Course policies: some of these policies are put by college or university and the other by instructor and some of these policies are depend to the nature of the course like extra credit, make up work, late work, attendance, participation academic dishonesty, changes in the syllabus, cooperative or teamwork.

Description of the course: there are two descriptions from instructor and registration.

Behavioral course objectives: clarify all the objectives that the instructor want to achieve throw the course and the way to achieve this objectives.

Required material for the course: the instructor clarifies the text books he use through the course and some courses needed more than one book may for lab. Provide all information about the book like title, author, what and where to get them, what they should get and of thee reading or other materials, why they should read them.

Optional material for the course: provide information about references and useful link that provide more information and skills to the students.

Provide all course requirements that asked from the students to complete the course like exams, home works, projects and common questions with dates, grading criteria, objectives and descriptions of these requirements.

Information about grading and evaluation policies: how will course grade be computed, what counts, how much, in what way?

Clarify all the teaching method: that the instructor want to use throw the course.

Course outline and topic that teach throw the course and distribute this topic to the weeks throw the semester.

As you know some courses have topic to class and topic to lab, for that they clarify this topic and distributed it to lab time and class time.

Finally, provide the instructor expectation for the student, about amount of time spend outside of class, about behavior in class, about effort or participation and provide more information and news about the course.

3-5 View method:

To simplify the access to the database for the instructor and student we add a new category on the PPU home page. The student can see pages contain the content of syllabus like (course name, numbers, sections, and credits hours, and brief description of the course and the name of instructors who teach the course.

As we know the syllabus content is controlled by the instructor so they need a form to update and add from the PPU home page, this form contains all fields that instructors need to control course syllabus content. Any instructor has an account (ID, password) to allow him to get an authorization to update and add to database.

How the students access the course syllabus?

First, the student open the PPU web site www.ppu.edu, next select the course from the menu, then the course page is open which contain general information about the courses and instructors names who teach the course, select the instructor name from it. Before the student select the instructor name, all the course syllabus are showing by number of semester and year(example: 2001-1, 2001-2).

Finally, the course syllabus is show and the student can show all the content.

How the instructor access to database content remotely:

To enable the instructor to access the database designs suitable forms to add and update the content of the database. The instructors open the ppu web site www.ppu.edu, from the categories they select instructors form, the form is loaded and ask to ID and password. This ID and password is checked, if it's true the member page is loaded, so the instructor can add and update from the data base, after they update and add, they press the sign out button to get out.

Any change like (add, update) on the database will take place remotely.

Data control diagram: (student)

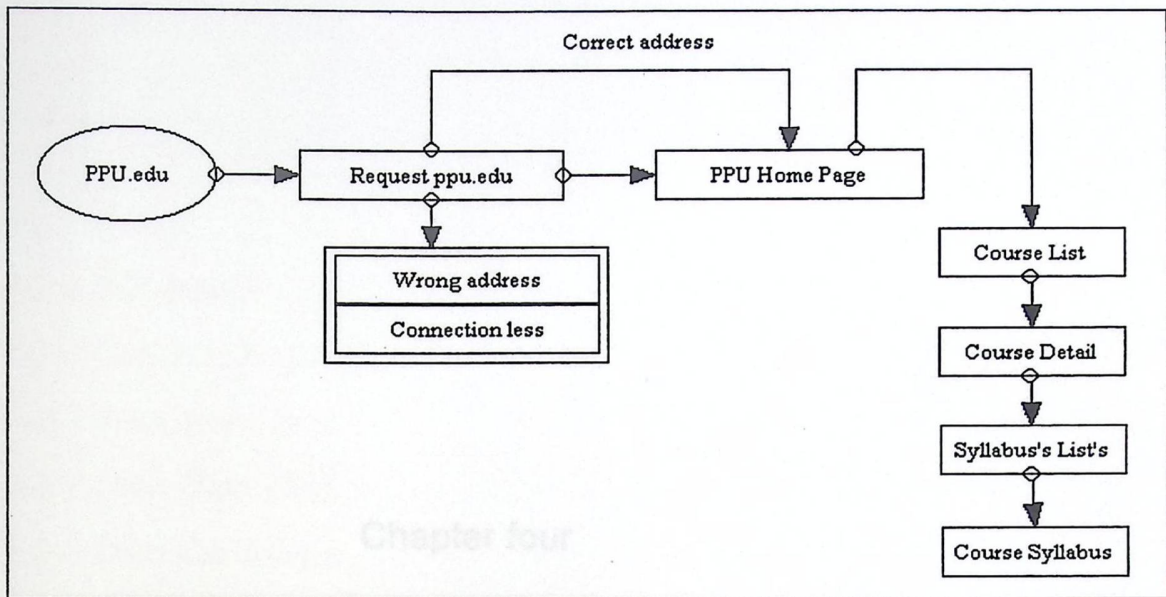


Figure 4.3 data control diagram: (student)

Data control diagram: (student)

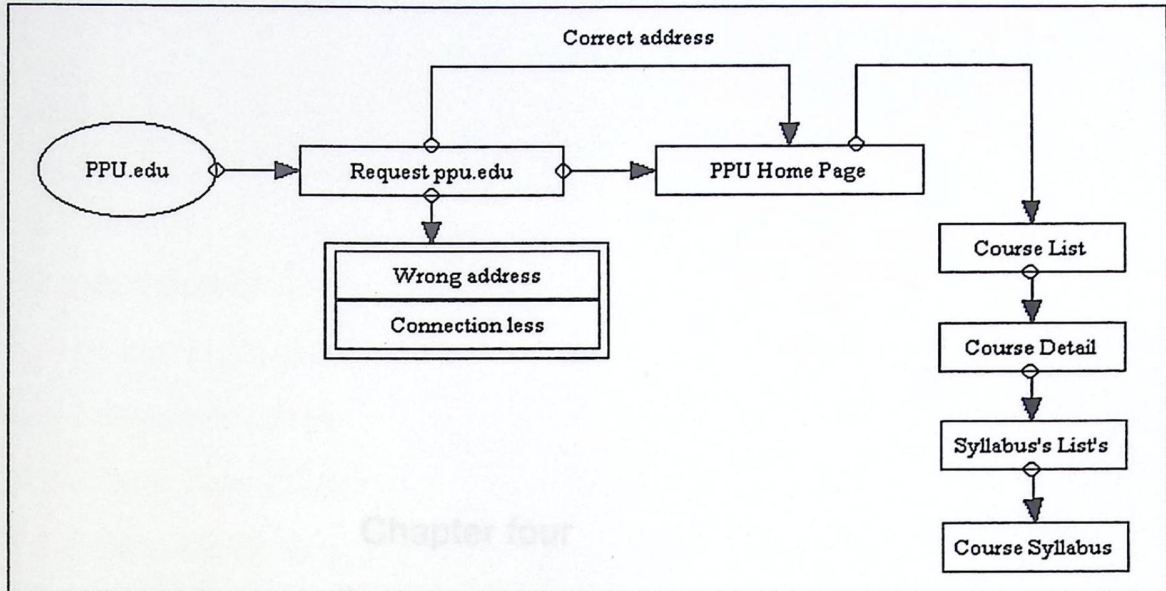


Figure 4.3 data control diagram: (student)

Data control diagram: (student)

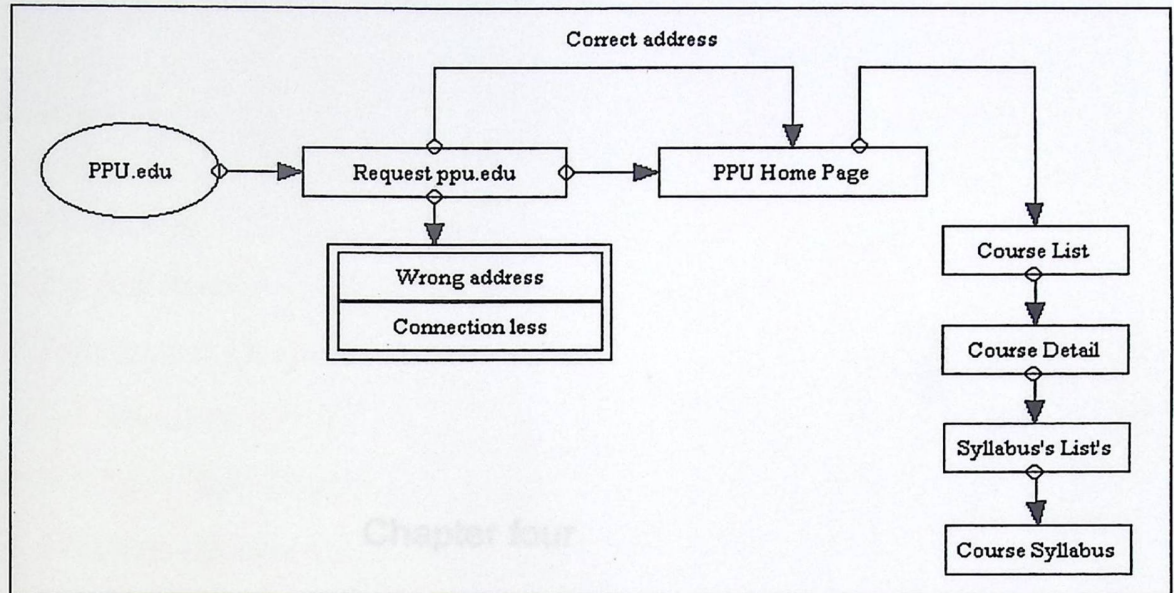


Figure 4.3 data control diagram: (student)

Chapter Four

System Design

4-1 Introduction

4-2 Design Models

4-2-1 E/R Model

4-2-2 Context Diagram

4-2-3 Structure Chart

4-2-4 Data Base Design

4-2-5 Data Dictionary

4-2-6 Data Flow Diagram

4-2-7 Flow Chart

4-3 System Interface

Chapter four

System Design

Chapter Four

System Design

4-1 Introduction:

4-1 Introduction

4-2 Design Models

4-2-1 E/R Model

4-2-2 Context Diagram

4-2-3 Structure Chart

4-2-4 Data Base Design

4-2-5 Data Dictionary

4-2-6 Data Flow Diagram

4-2-7 Flow Chart

4-3 System Interface

ER model of database content

4.3.1 ER Model of Database Content:

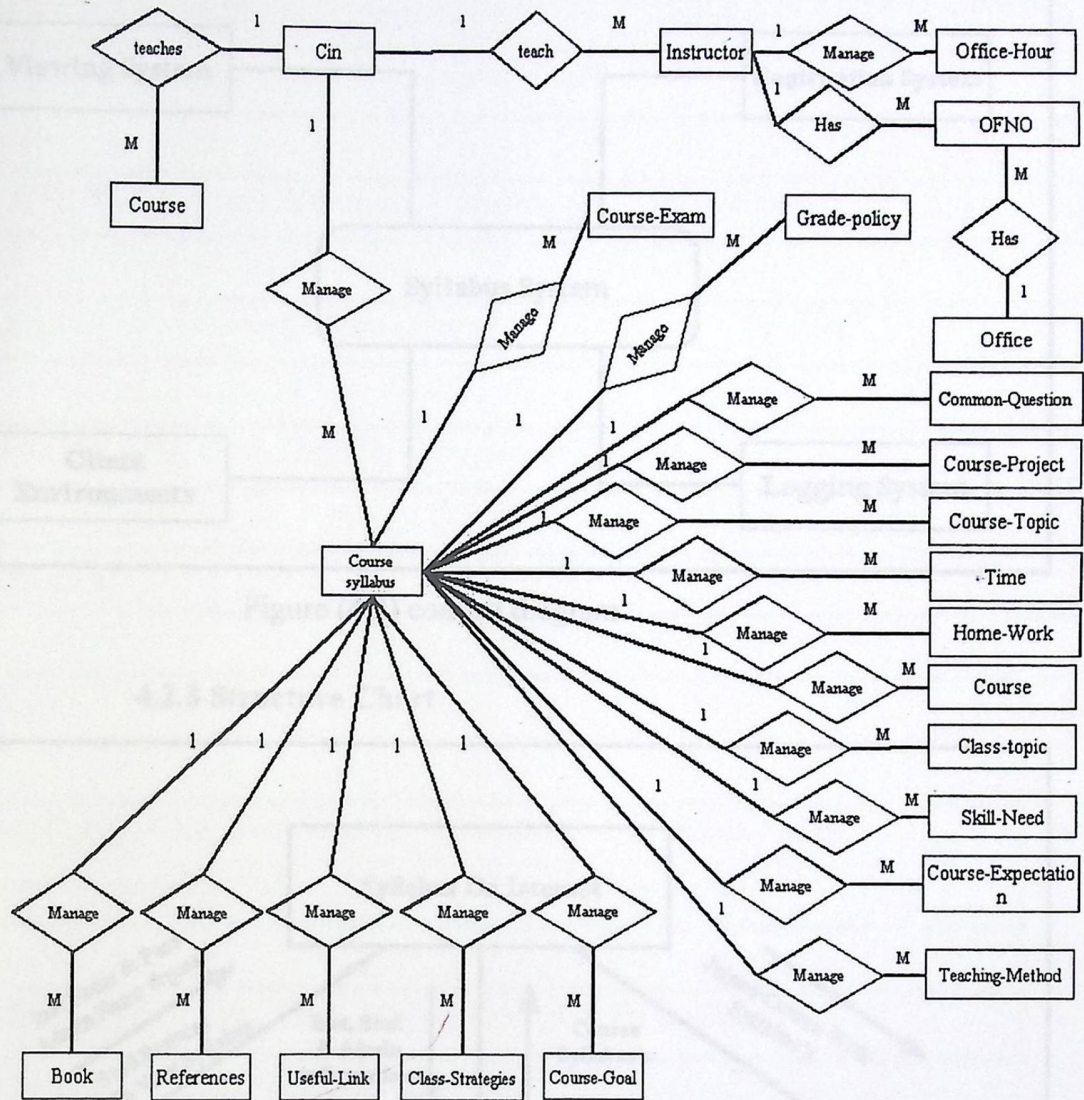


Figure (4.1) ER model of database

4.2.2 Context diagram

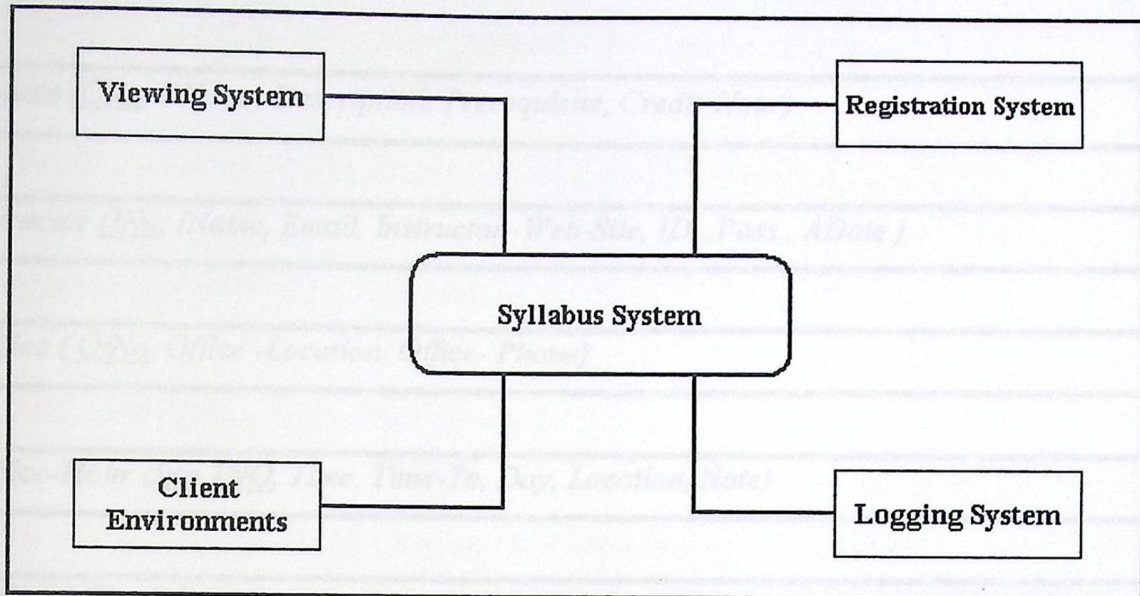


Figure (4.2) context diagram

4.2.3 Structure Chart

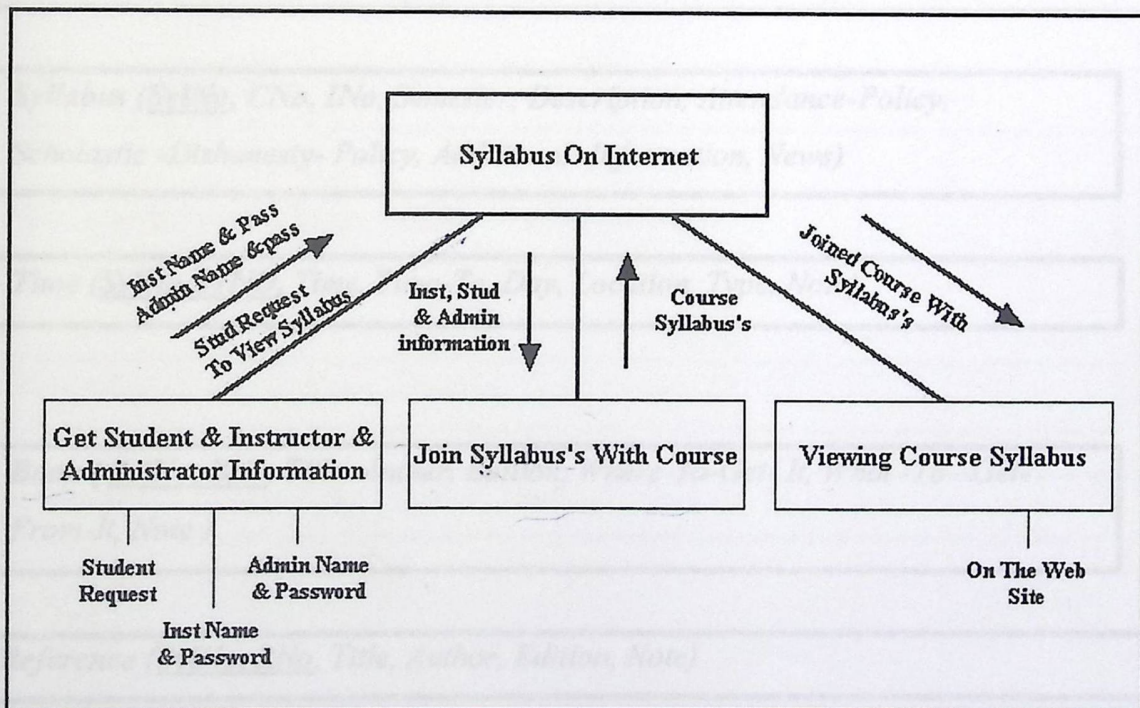


Figure (4.3) structure chart

4.2.2 Context diagram

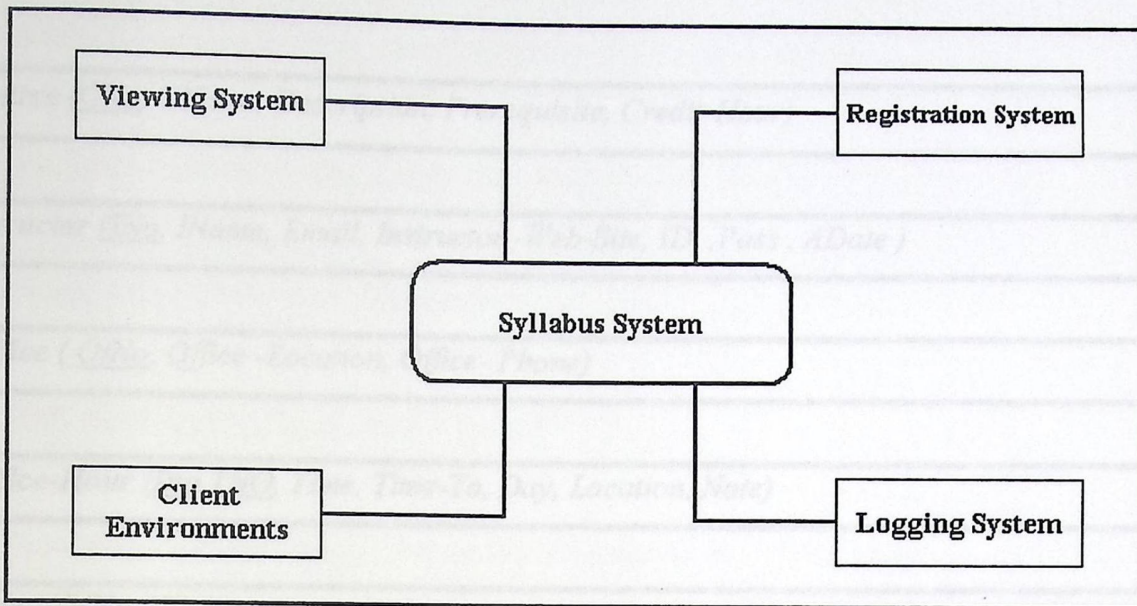


Figure (4.2) context diagram

4.2.3 Structure Chart

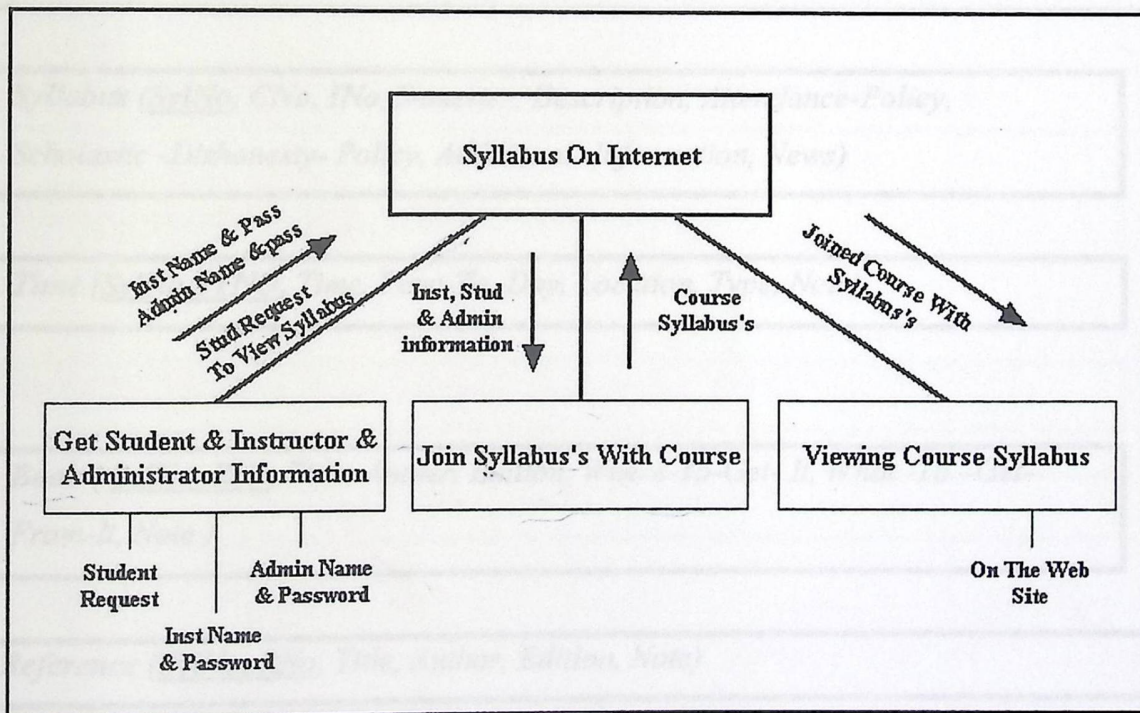


Figure (4.3) structure chart

4.2.2 Context diagram

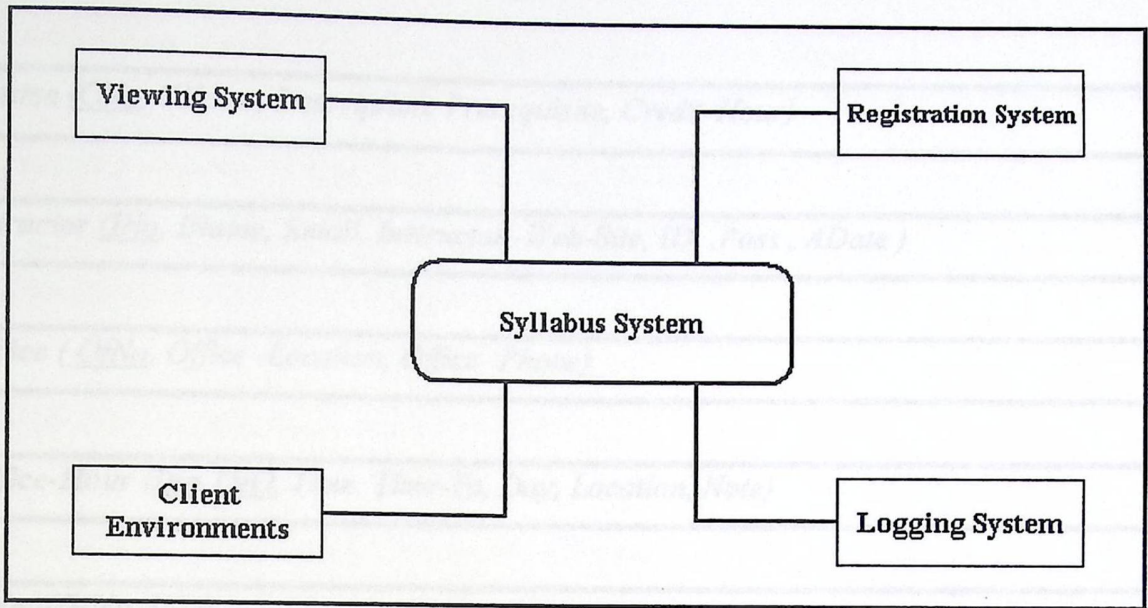


Figure (4.2) context diagram

4.2.3 Structure Chart

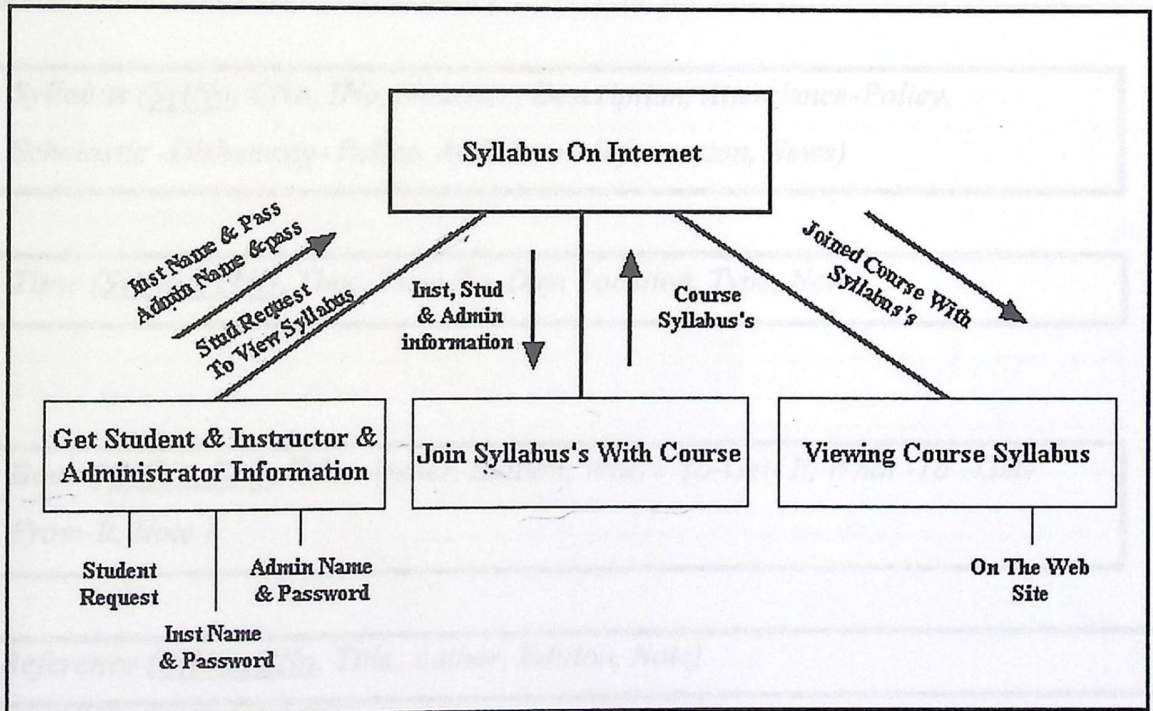


Figure (4.3) structure chart

4.2.4 Database

design:

Course (CNo, CName, Description, Prerequisite, Credit-Hour)

Instructor (INo, IName, Email, Instructor -Web-Site, ID ,Pass , ADate)

Office (OfNo, Office -Location, Office- Phone)

Office-Hour (INo,TNO, Time, Time-To, Day, Location, Note)

Ofno(OfNo, INo)

Cin (INo, CNo)

Syllabus (SylNo, CNo, INo, Semester, Description, Attendance-Policy,
Scholastic -Dishonesty- Policy, Additional -Information, News)

Time (SylNo,STNO, Time, Time-To, Day, Location, Type, Note)

Book (SylNo, BNo, Title, Anther, Edition, Where-To-Get- It, What -To -Get-
From-It, Note)

Reference (SylNo, RNo, Title, Author, Edition, Note)

Useful- Link (SylNo, UNo, Link)

Class- Strategies (SylNo, CSNo, Strategies)

Course- Goal (SylNo, GNo ,Goal ,Activities- To- Achieve- Goal)

Teaching- Method (SylNo, TMNo ,Teaching Method ,Goal)

Expectation (SylNo, Eon, Expectation)

Skill- Need (SylNo, SkNo ,Back -Ground)

Class -Topic (SylNo, Week, Chapter-No, Chapter Name, TYPE, Note)

Home-Work (SylNo, CHNo, Home-Work, Description, Date ,TYPE, Note)

COURSE- Project (SylNo, CPNo, Project, Description, Date, TYPE, Note)

Common-question (SylNo, CQNo, common-question)

Exam (SylNo, EXNo, Exam, Material, Date, Grade, TYPE)

Grade- Policy (SylNo, GNo, GName, Mark)

4.2.5 Data dictionary:

Table name	Attribute	Description	Table Keys			Type	Constraint
			Primary	Foreign	reference		
Course	CNo	Course number	Primary	-	-	Integer	Not null
	CName	Course name	-	-	-	Character	Not null
	Description	A brief descriptions of the course	-	-	-	Character	-
	Prerequisite		-	-	-	Character	Not null
	Credit Hour	Number of credit hour of the course	-	-	-	Integer	Not null

instructor	<i>INo</i>	Job number of the instructor and it must be unique	Primary	-	-	Integer	Not null	
	<i>IName</i>	Full instructor name	-	-	-	Character	Not null	
	<i>E-Mail</i>	Instructor email	-	-	-	Character	Not null	
	<i>Instructor Web Site</i>	The instructor web site	-	-	-	Character	-	
	<i>ID</i>	User name of instructor	Primary	-	-	Character	Not null	
	<i>Pass</i>	Password of the instructor	-	-	-	Character	Not null	
	<i>ADate</i>	Time of operate the account	-	-	-	Date	Not null	
	<i>Office</i>	<i>OfNo</i>	Office number	Primary	-	-	Integer	Not null
		<i>Office Location</i>	The location of the instructor office	-	-	-	Character	Not null

	Office Phone	Phone number of the office	-	-	-	Instructor	Integer	Not null
Office-hour	<i>INo</i>	Instructor number	Primary	Foreign	Instructor table	Integer	Integer	Not null
	<i>Time</i>	Start time of the instructor office hours	Primary	-	-	DateTime	DateTime	Not null
	<i>Time-To</i>	End time of the instructor office hours	-	-	-	DateTime	DateTime	Not null
	<i>OfNo</i>	Office number	-	Foreign	Office table	Integer	Integer	Not null
	<i>Day</i>	Day that the instructor has office hour	-	-	-	Character	Character	-
	<i>Note</i>	Any note the instructor want to add in this office hours	-	-	-	Character	Character	-

OFNO	INo	Instructor number	Primary	Foreign	Instructor table	Number	NotNull
	OFNO	User name of instructor	Primary	Foreign	Office	Number	NotNull
syllabus	Sy/No	Sequential number of all course syllabus in syllabus table	Primary	-	-	Integer	NotNull
	CNo	Course number	-	Foreign	Course table	Integer	NotNull
	INo	Instructor number	-	Foreign	Instructor table	Integer	NotNull

SNo	<p>A sequential number off the syllabus for each instructor (e.g. ((INo=110, SNo=1) to (INo=101, SNo=5)) ((INo=1102, SNo=1) _</p>	-	-	-	-	-	Not null
Semester	<p>Number of semester plus year (e.g. 2003-1, 2003-2)</p>	-	-	-	-	-	Integer
Description	<p>A brief description of the course, instructor description</p>	-	-	-	-	-	Character

SNo	A sequential number off the syllabus for each instructor (e.g. ((INo=110, SNo=1) to (INo=101, SNo=5)) ((INo=1102, SNo=1)_	-	-	-	-	Integer	Not null
Semester	Number of semester plus year (e.g. 2003-1, 2003-2)	-	-	-	-	Integer	Not null
Description	A brief description of the course, instructor description	-	-	-	-	Character	-

Time	Sy/No	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
			Primary	-	-	number	Not null
	StNo	Sequential number of all course time in course table	-	-	-	DateTime	Not null
	Time	Time of the lecture or lab, start time	-	-	-	DateTime	Not null
	Time-To	End time of the lecture or lab	-	-	-	Character	Not null
	Day	Day of lecture or lab	-	-	-	Character	Not null
	Location	The location of the lecture or lab in any section	-	-	-	Character	Not null

Type	Its determine this time for lab or lecture by type lab or lecture in filed	-	-	-	Character	-
Book	<i>Sy/No</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer Not null
	<i>B/No</i>	Number of the book	Primary	-	-	Integer Not null
	Title	book title	-	-	-	Character Not null
	Author	book authors	-	-	-	Character Not null
	Edition	The edition number of book	-	-	-	Character Not null
	Where-to-get-it	Its determine from where you can get book	-	-	-	Character -

	What-to-get-from-it	Its about material that teach in course	-	-	-	Character	-
	Note	Any note the instructor want to add	-	-	-	Character	-
Reference	<i>SyNo</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
	<i>RNo</i>	References number	Primary	-	-	Integer	Not null
	Title	references title	-	-	-	Character	-
	Author	references authors	-	-	-	Character	-
	Edition	The edition number of references	-	-	-	Integer	-

	Note	Any note the instructor want to add	-	-	-	Character	-
Useful-link	<i>Sy/No</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
	<i>UNo</i>	Sequential number of useful link for the instructor	Primary	-	-	Integer	Not null
Class-strategie	Link	The address of this useful link that may help the student	-	-	-	Character	-
	<i>Sy/No</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null

s	CSNo	Sequential number of the strategies	Primary	-	-	Integer	Not null
	Strategies	The strategies that the instructor use throw the course and class	-	-	-	Character	-
	SyINo	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
Course-goal	GNo	Sequential number of the goal	Primary	-	-	Integer	Not null
	Goal	The goals that the instructor want to achieve throw the course	-	-	-	Character	-

	Activities- to- achieve the goal	The activities that the instructor use to achieve the previous goal	-	-	Character	-
Teaching -method	Sy/No	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Integer	Not null
	TM/No	Sequential number of teaching method	Primary	-	Integer	Not null
	Teaching- Method	The teaching method that the instructor want to use throw the course	-	-	Character	-
	Goal	The goal that achieve throw use like this teaching method	-	-	Character	-

Expectation	<i>Sy/No</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	NotNull
	<i>ENo</i>	Sequential number of expectation	Primary	-	-	Integer	Primary
	<i>Expectation</i>	The expectation that instructor want to add	-	-	-	Character	-
Skill-need	<i>Sy/No</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	NotNull
	<i>SKNo</i>	Sequential number of skill need	Primary	-	-	Integer	NotNull

	<i>Back-Ground</i>	The back ground that instructor expect the students know about materials	-	-	-	Character	-
course topic	<i>Sy/No</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
	<i>PTNo</i>	Sequential number of the course topic	Primary	-	-	Integer	Not null
	<i>Week</i>	Number of week that the instructor want to view this topic (e.g. week 1, week 2)	-	-	-	Integer	-

Chapter-No	Number of chapter that the instructor want to view throw the week	-	-	-	Integer	-
Chapter-Name	The title of the chapter	-	-	-	Character	-
Note	Any note that instructor want to add	-	-	-	Character	-
Type	Type of course topic lab or lecture	-	-	-	Character	-
SyINo	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
CHNo	Sequential number of the class home work.	Primary	-	-	Integer	Not null

home-work

Home-Work	The name of the home work that the instructor want	-	-	-	Character	-
Description	A brief description of the home work to explain to the student.	-	-	-	Character	-
Date	The date that determined to receive the home work	-	-	-	Date Time	-
Note	Any not the instructor want to add	-	-	-	Character	-
Type	Type of course home work lab or lecture	-	-	-	Character	-

<i>SyINo</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
<i>CPNo</i>	Sequential number of the class project.	Primary	-	-	Integer	Not null
Project	The name of the project that the instructor want	-	-	-	Character	-
Description	A brief description of the project to explain to the student.	-	-	-	Character	-
Date	The date that determined to receive the project	-	-	-	Date/Time	-

	<i>Note</i>	Any note the instructor want to add about project	-	-	-	Character	-
	Type	Type of course home work lab or lecture	-	-	-	Character	-
Common -question	<i>Ssy/No</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
	<i>CQNo</i>	Sequential number of the common question The name of the common question that the instructor want	Primary	-	-	Integer	Not null
	Common Question		-	-	-	Character	-

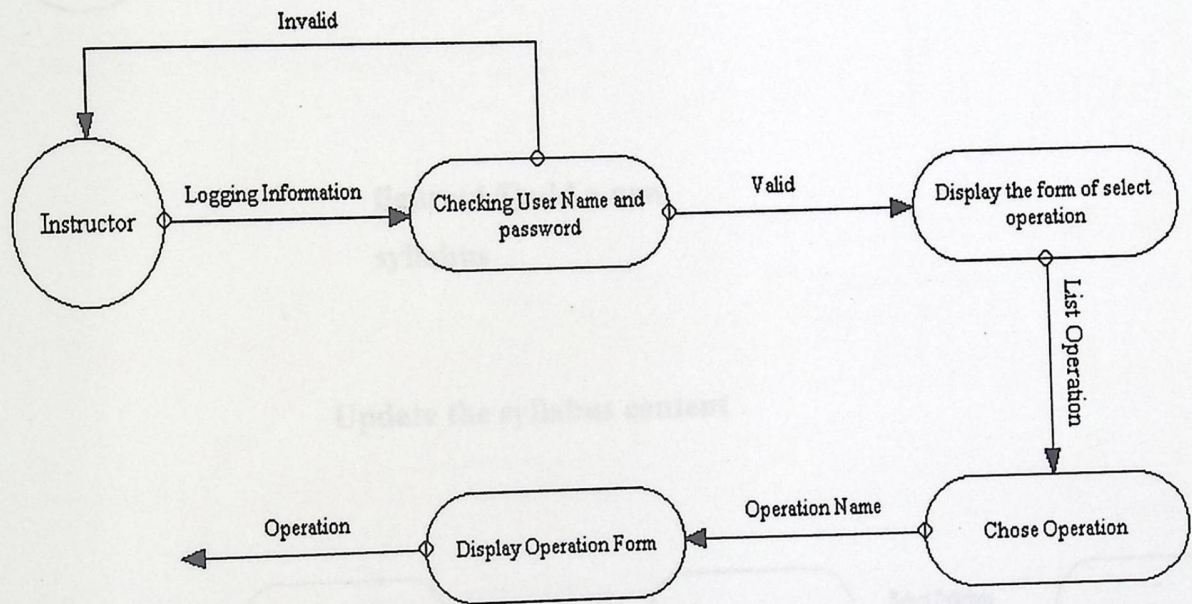
	<i>SyINo</i>	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
Exam	<i>EXNo</i>	Sequential number of the exam	Primary	-	-	Integer	Not null
	Exam	The name of the exam (e.g. first, second)	-	-	-	Character	-
	Material	The material of the exam that determined by the instructor.	-	-	-	Character	-
	Date	Date of the exam	-	-	-	DateTime	-
	Grade	The grade of the exam	-	-	-	Integer	-

Grade-policy	Sy/No	Sequential number of all course syllabus in syllabus table	Primary	Foreign	Syllabus table	Integer	Not null
	GPN0	Sequential number of the grade	Primary	-	-	Integer	Not null
	GName	Grade name (e.g. first, HW, project, participation)	-	-	-	Character	-
	Mark	Distribution mark to the GName (e.g. first 20%)	-	-	-	Integer	-

4.2.6 Data flow diagram

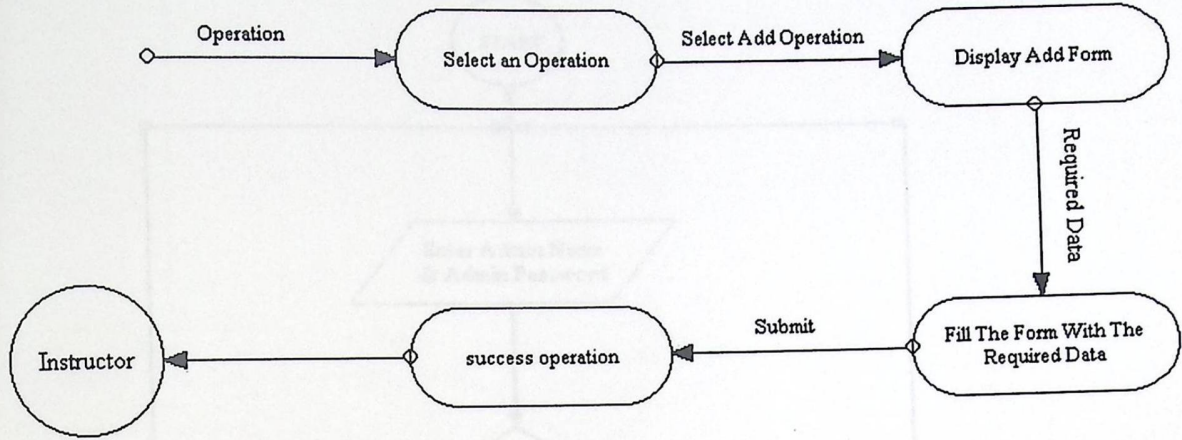
This diagram shows the flow of data in the whole system, from the beginning page then to insure that the instructor feeds the correct ID and password, then the operation list will be displayed to him

instructor validation and display the operation form



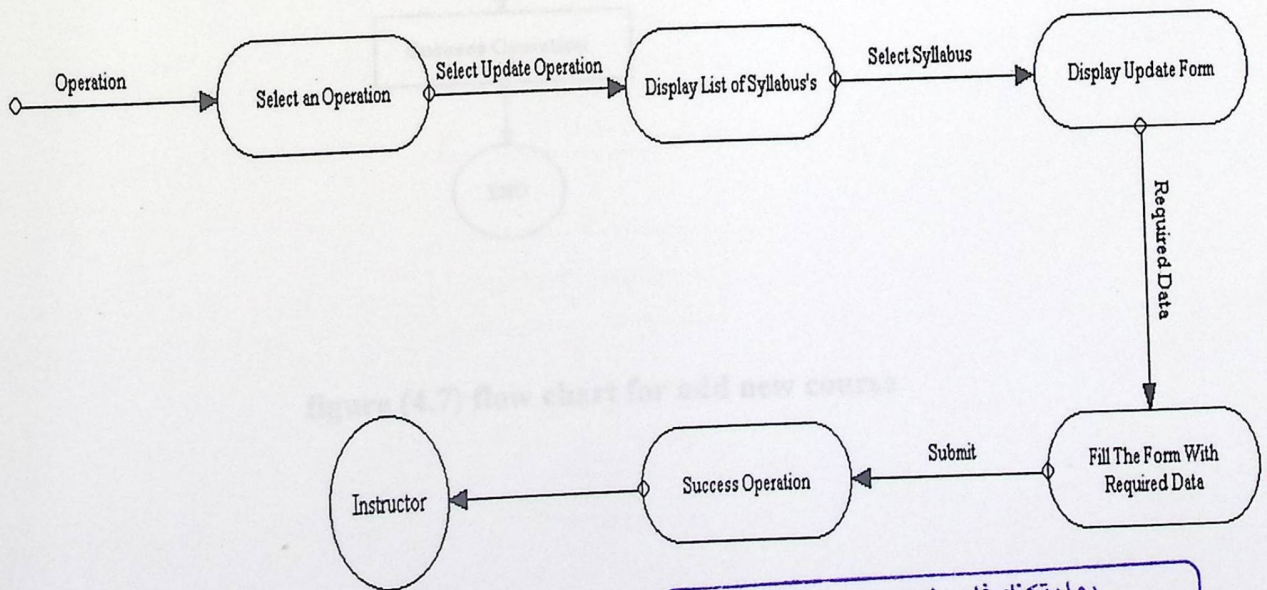
figure(4.4) instructor validation and display the operation form

add a new syllabus



figure(4.5)add a new syllabus

Update the syllabus content



figure(4.6) Update the syllabus content

4.2.7 Flow Chart

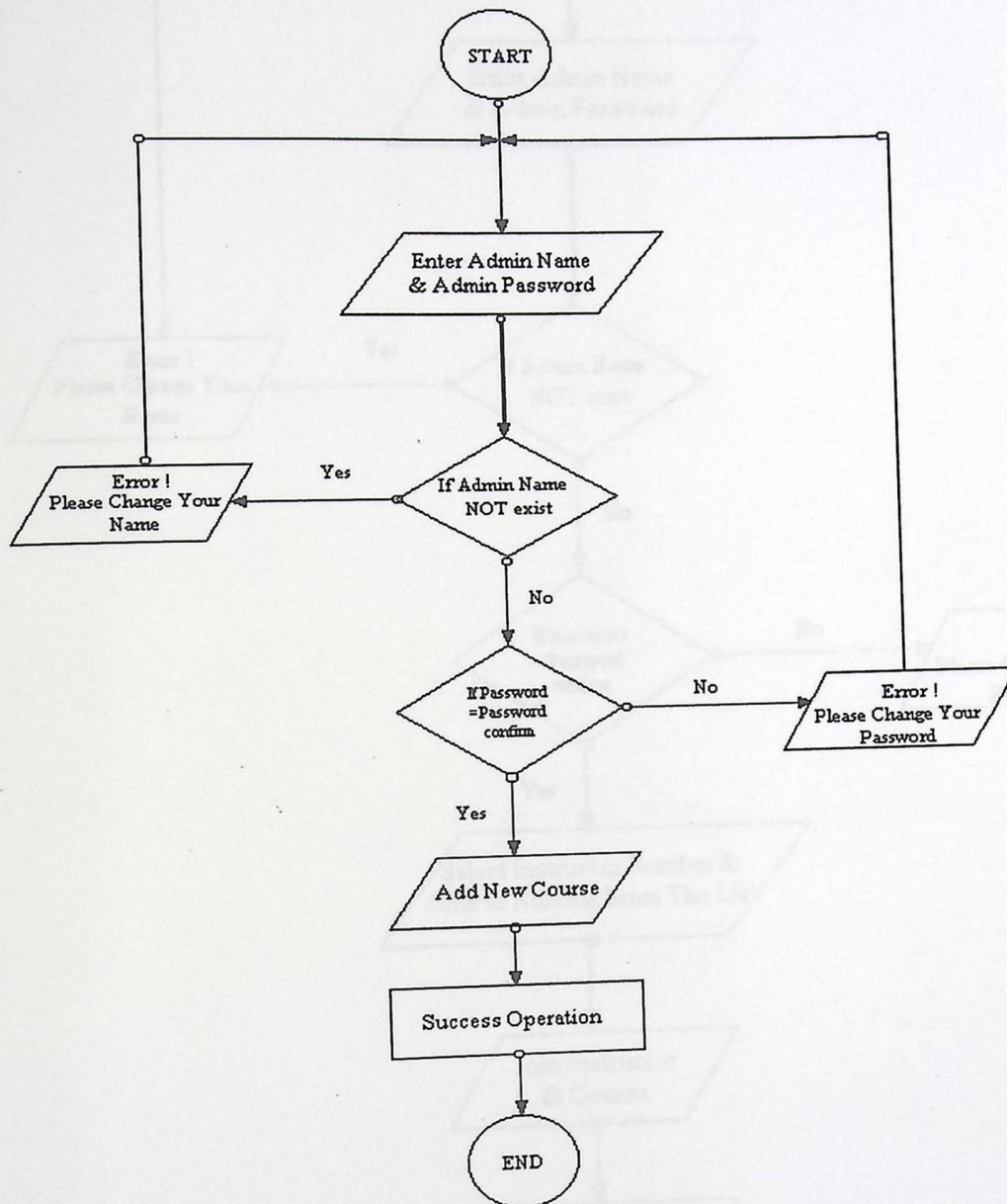


figure (4.7) flow chart for add new course

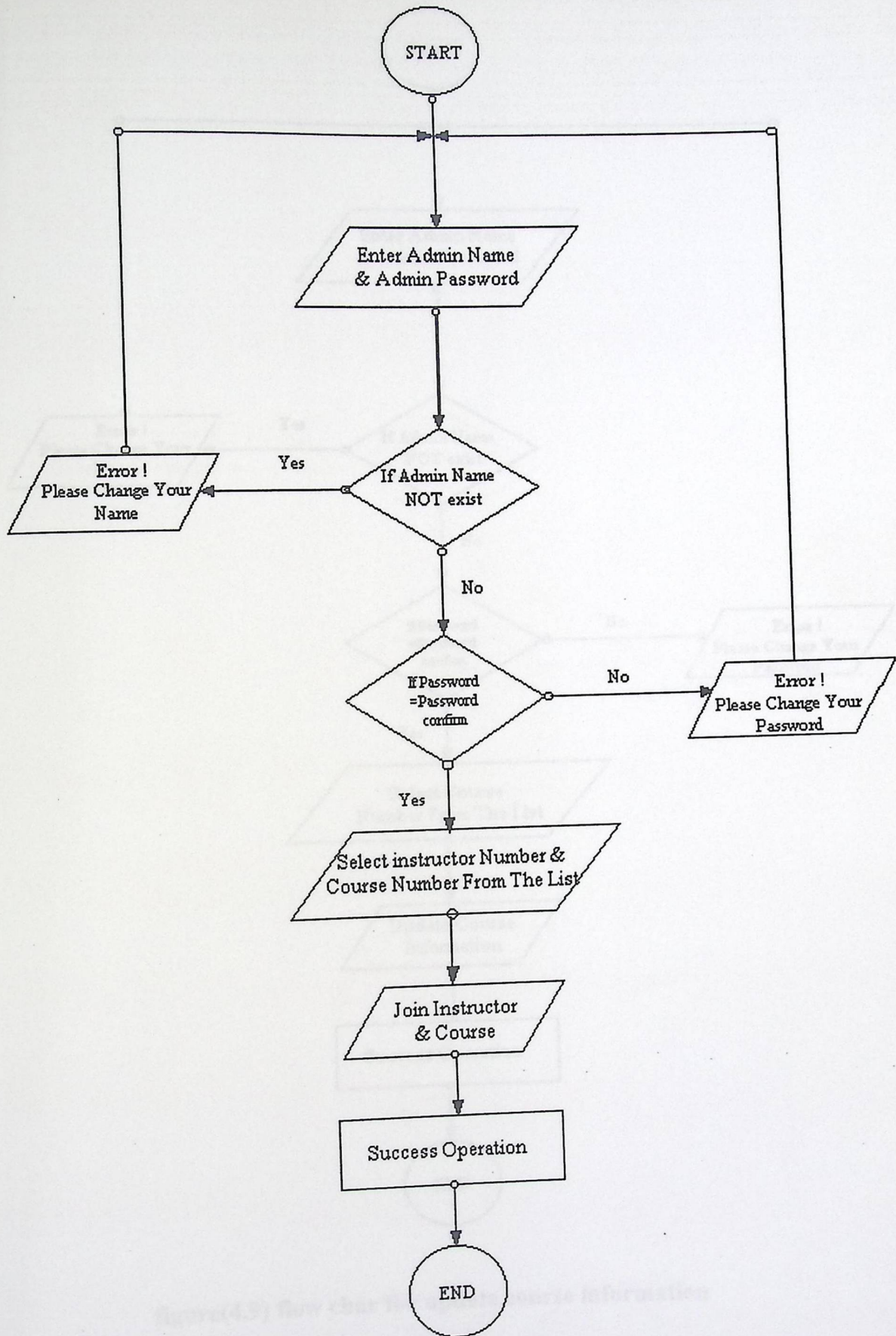
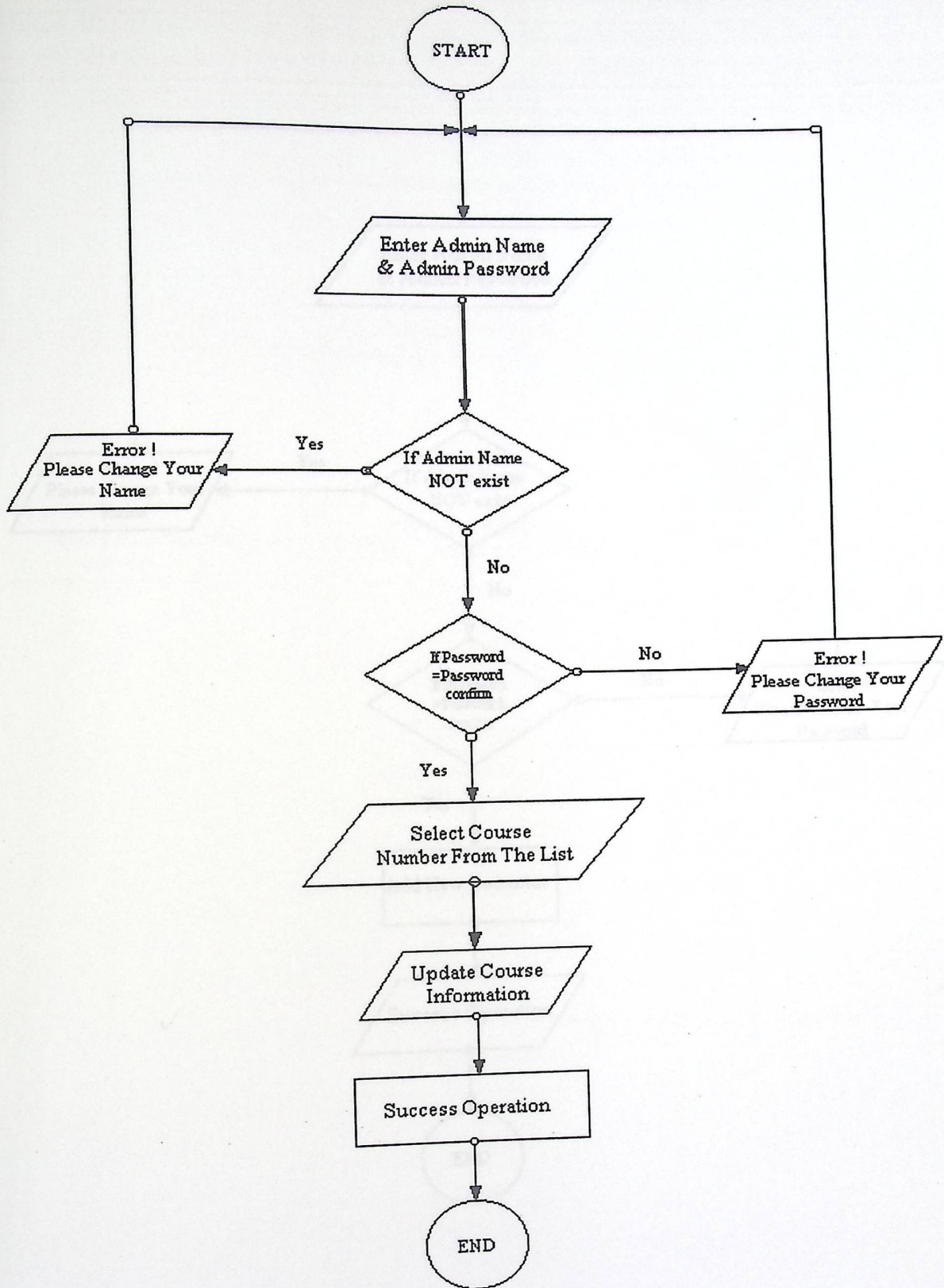


figure (4.8) flow chart for join instructor and course



figure(4.9) flow char for update course information

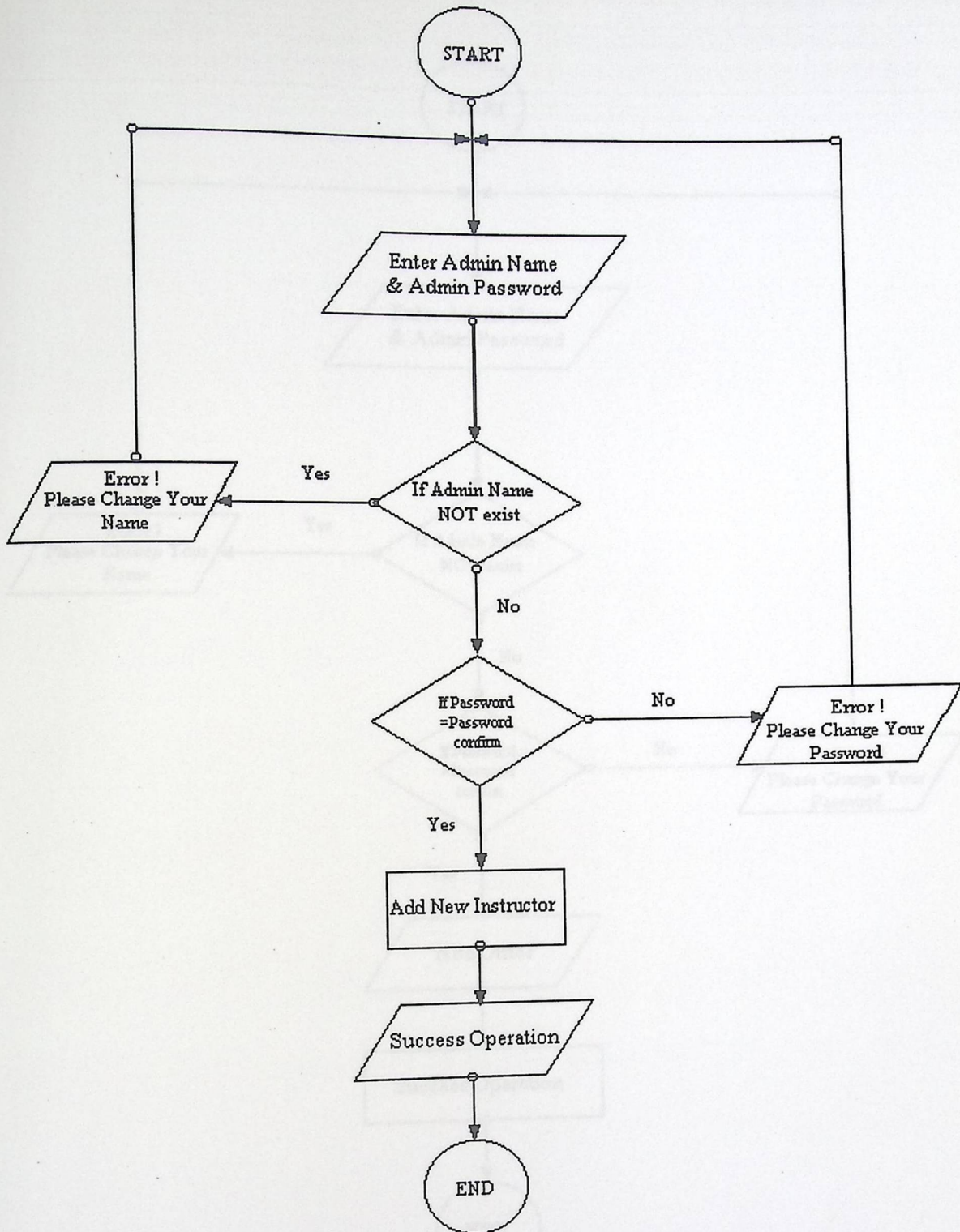


figure (4.10) flow chart for add new instructor

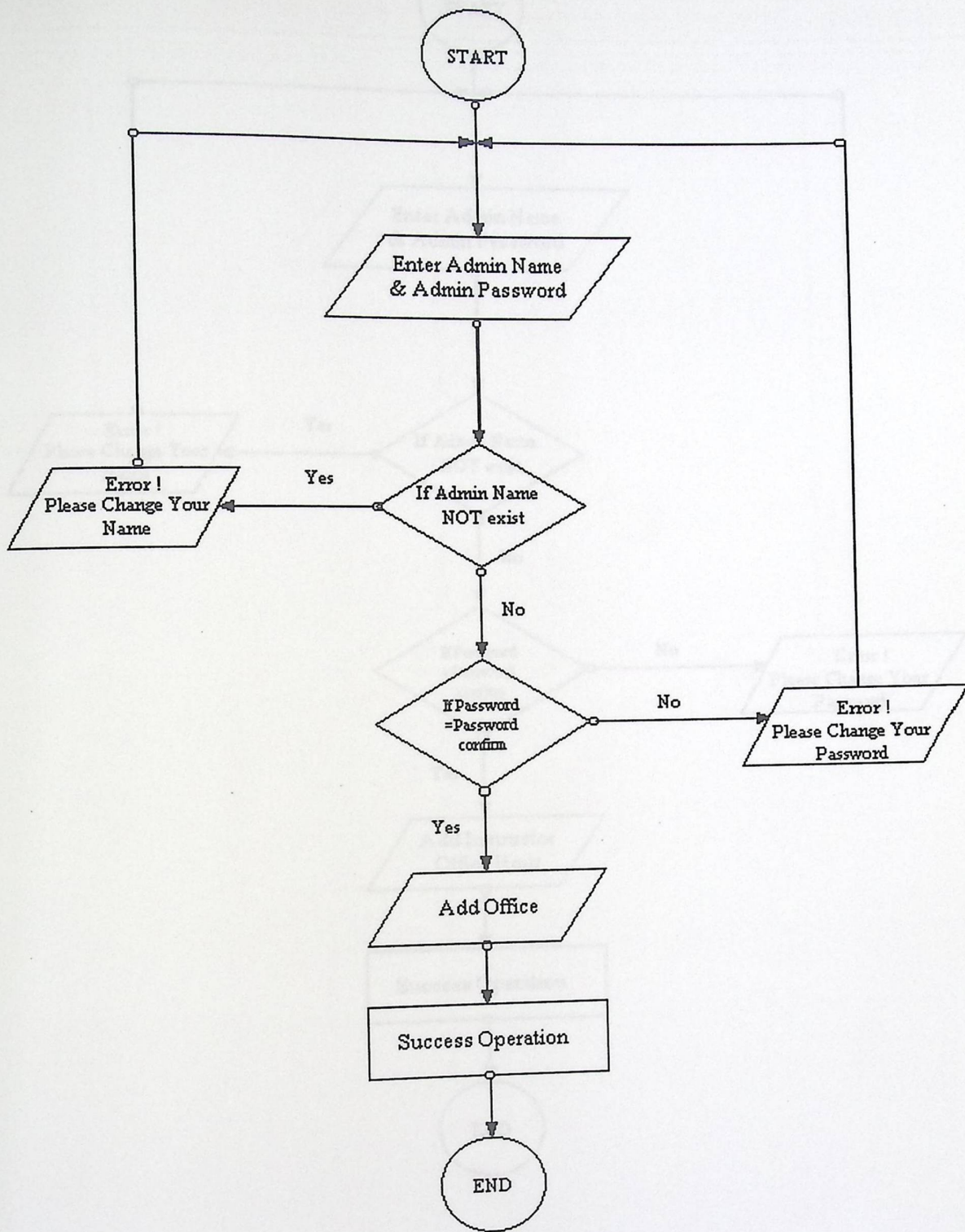
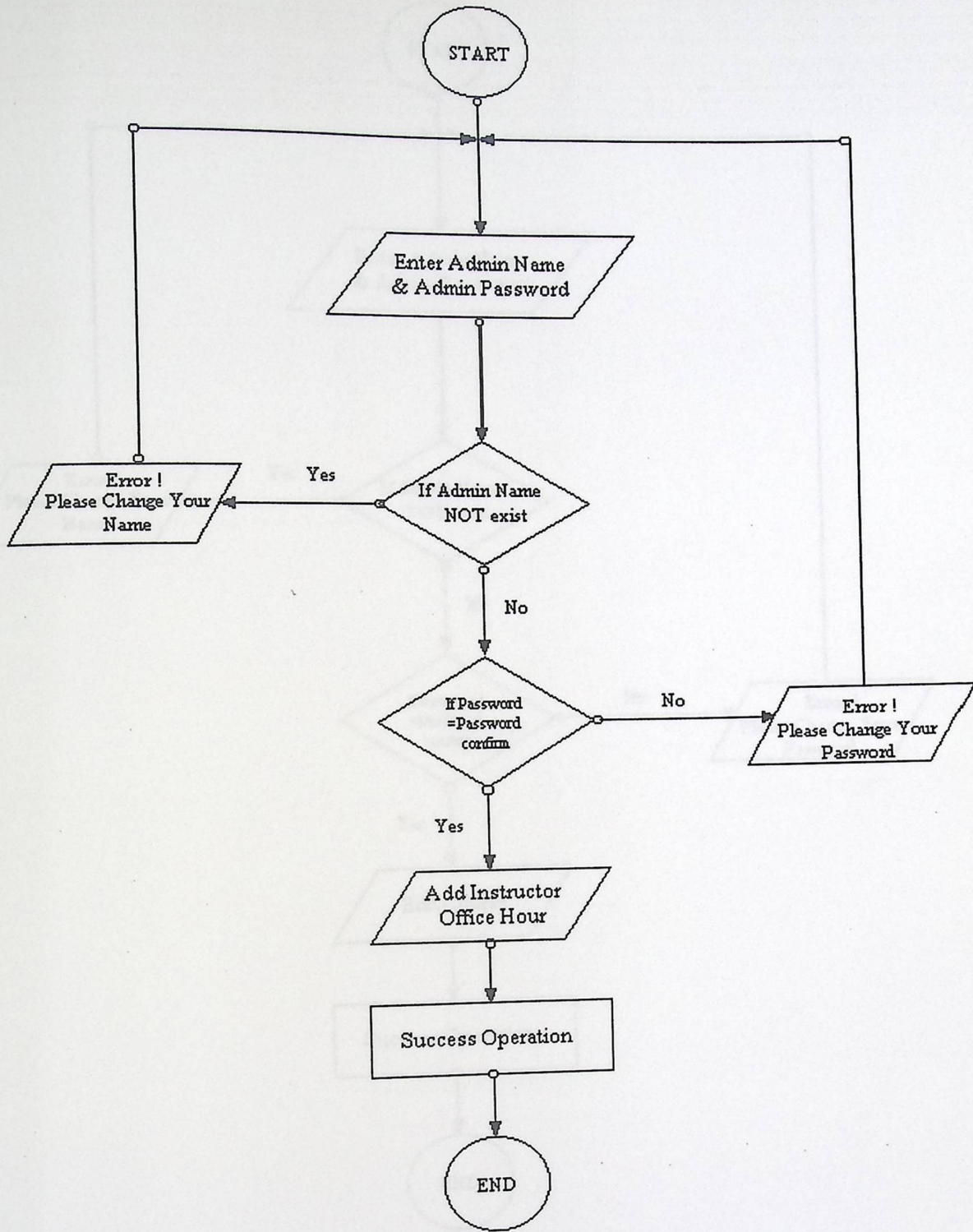
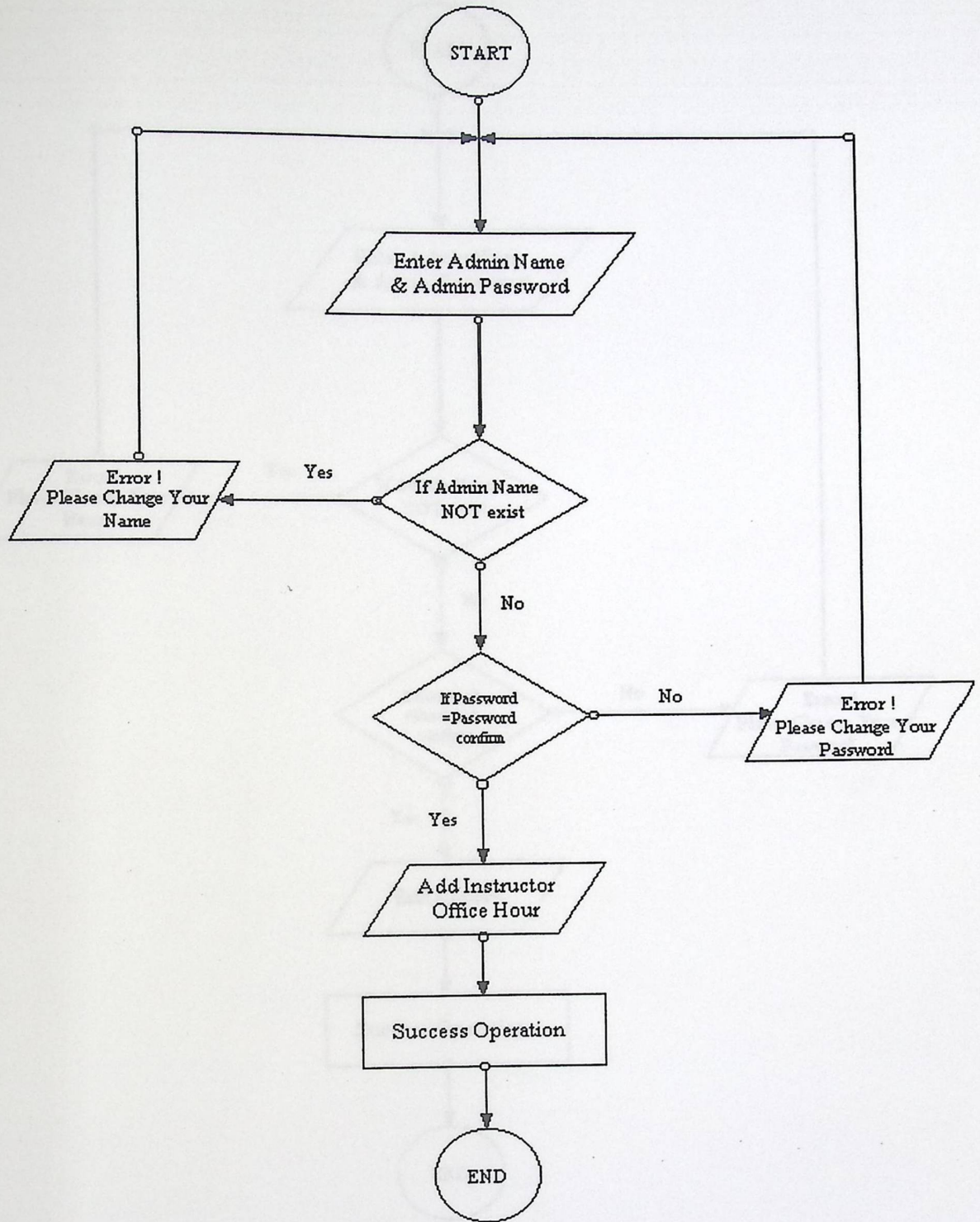


figure (4.11) flow chart for add new office



figure(4.12) add office hour



figure(4.12) add office hour

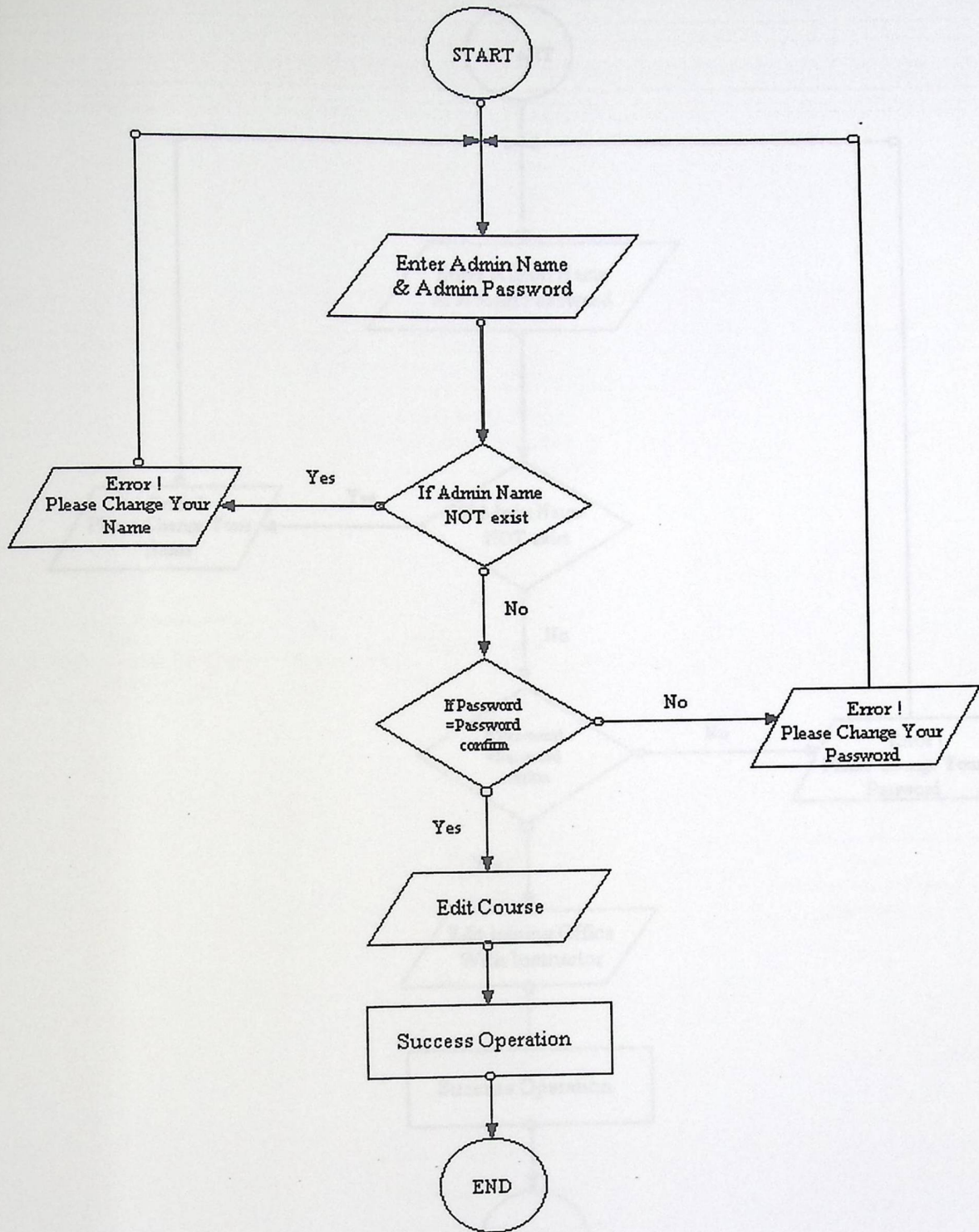


Figure (4.13) edit course

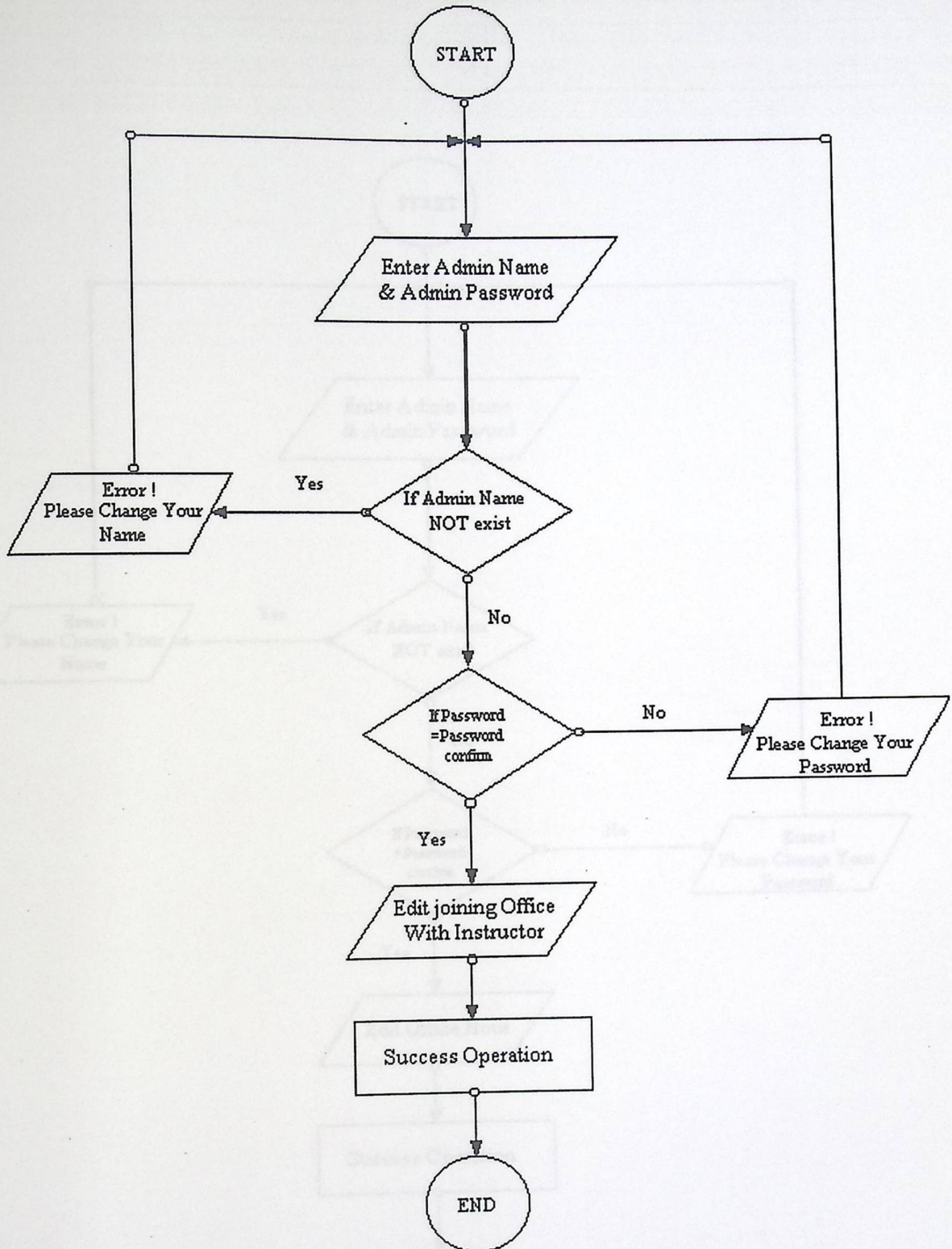


figure (4.14) flow chart for edit join office ith instructor

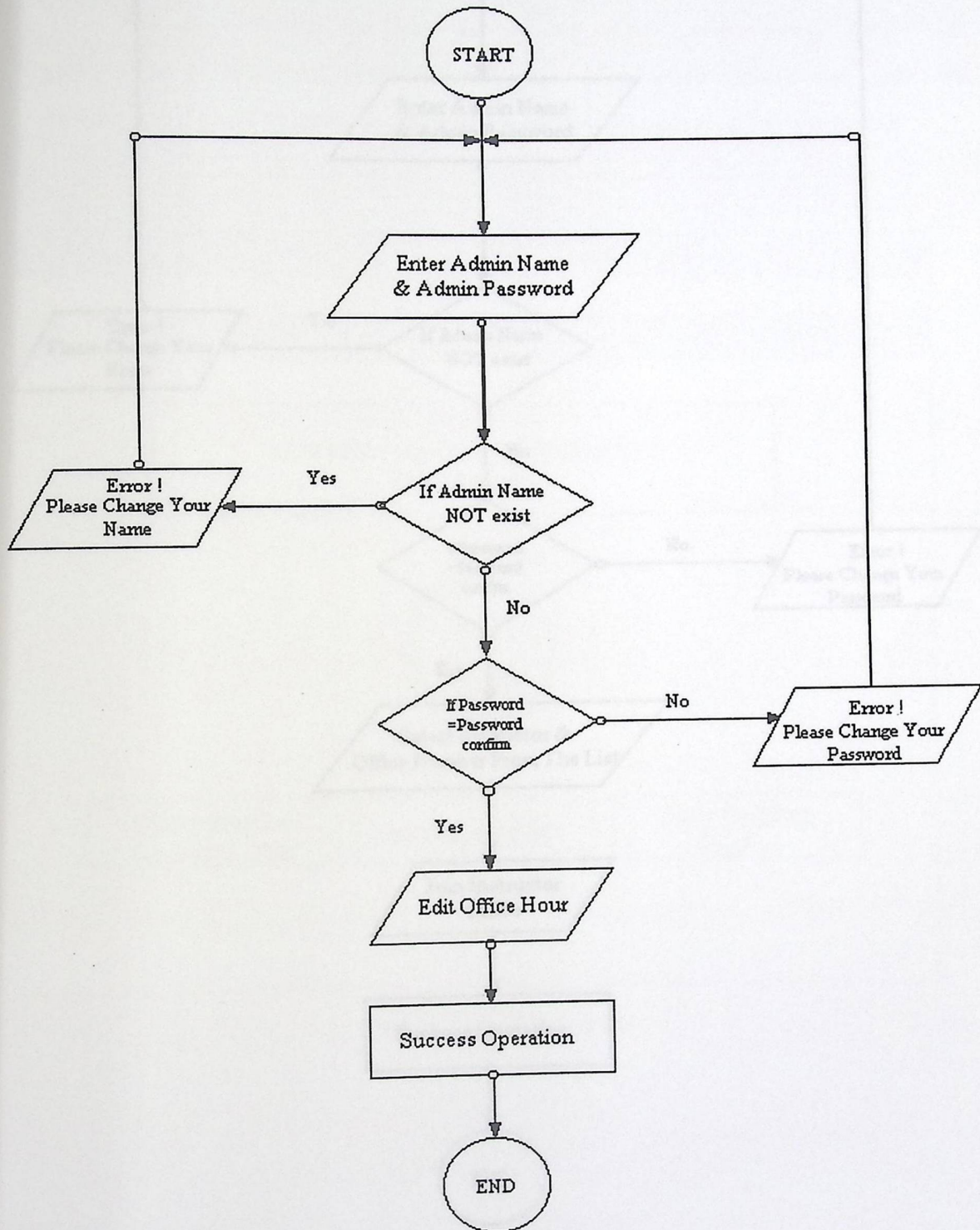


Figure (4.15) flow chart for edit office instructor

figure(4.15) flow chart for edit office hour

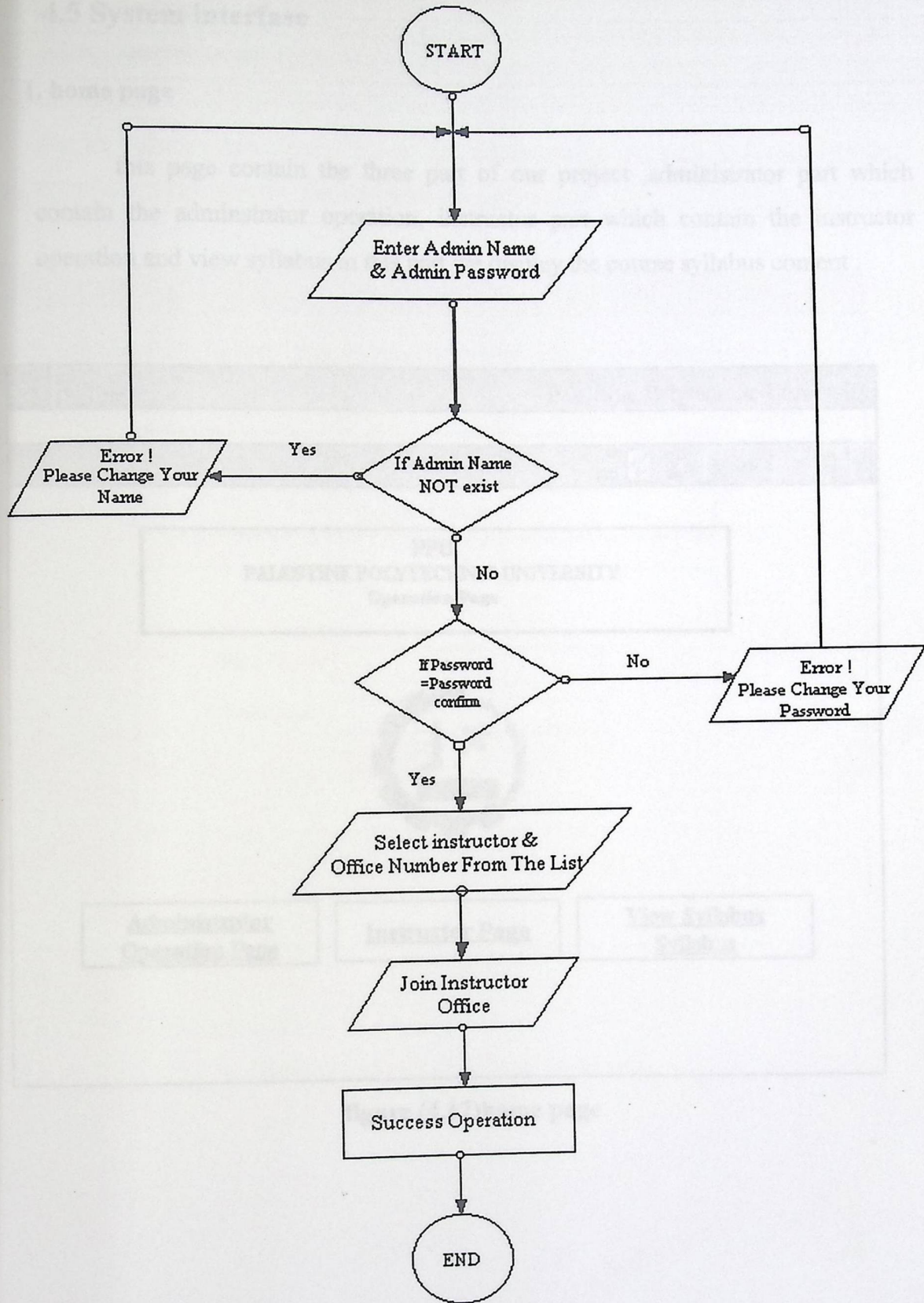


Figure (4.16) flow chart for join office instructor

4.5 System interfase

1. home page

this page contain the three part of our project ,administrator part which contain the adminstrator operation, instructor part which contain the instructor operation and view syllabus in this part we displey the course syllabus content .

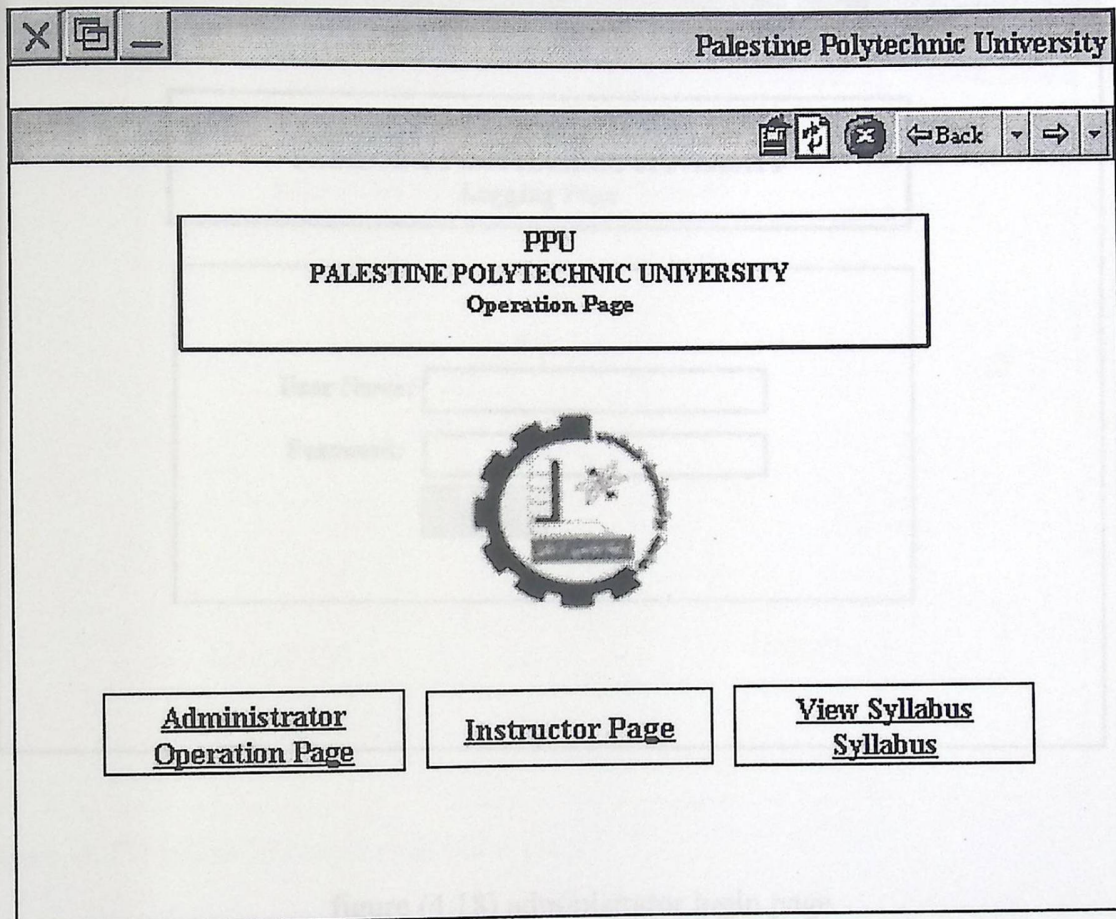


figure (4.17)home page

2. Administrator part:

2.1 Administrator Login Page

Log in page in this page we ask the administrator to feed a correct ID and password to continue to the administrator operation page

The screenshot shows a web browser window with the title "Palestine polytechnic university". The browser's address bar and navigation buttons (Back, Forward) are visible at the top. The main content area contains a header box with the text "PPU PALESTINE POLYTECHNIC UNIVERSITY Logging Page". Below this header is a form with two input fields: "User Name:" and "Password:". A "Login" button is positioned below the "Password:" field.

figure (4.18) administrator login page

2.2 Administrator Operation Page

this page contain to part instructor and course in this page we determine if the operation to instructor or course (add, update), if the administrator select the instructor the instructor operation page will display.

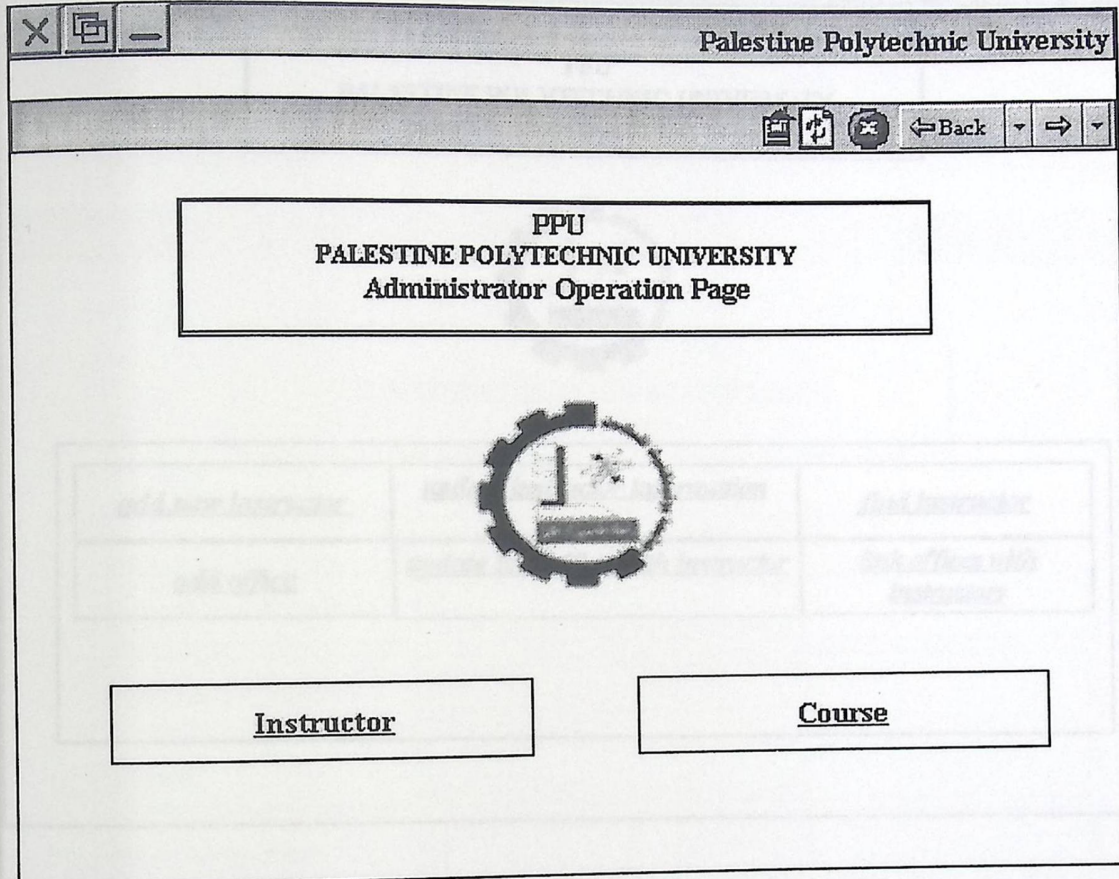


figure (4.19) administration operation page

2.3 Administrators Operation Page (Istructor)

Instructor operation page this page contain all the operation that its special to the instructor part

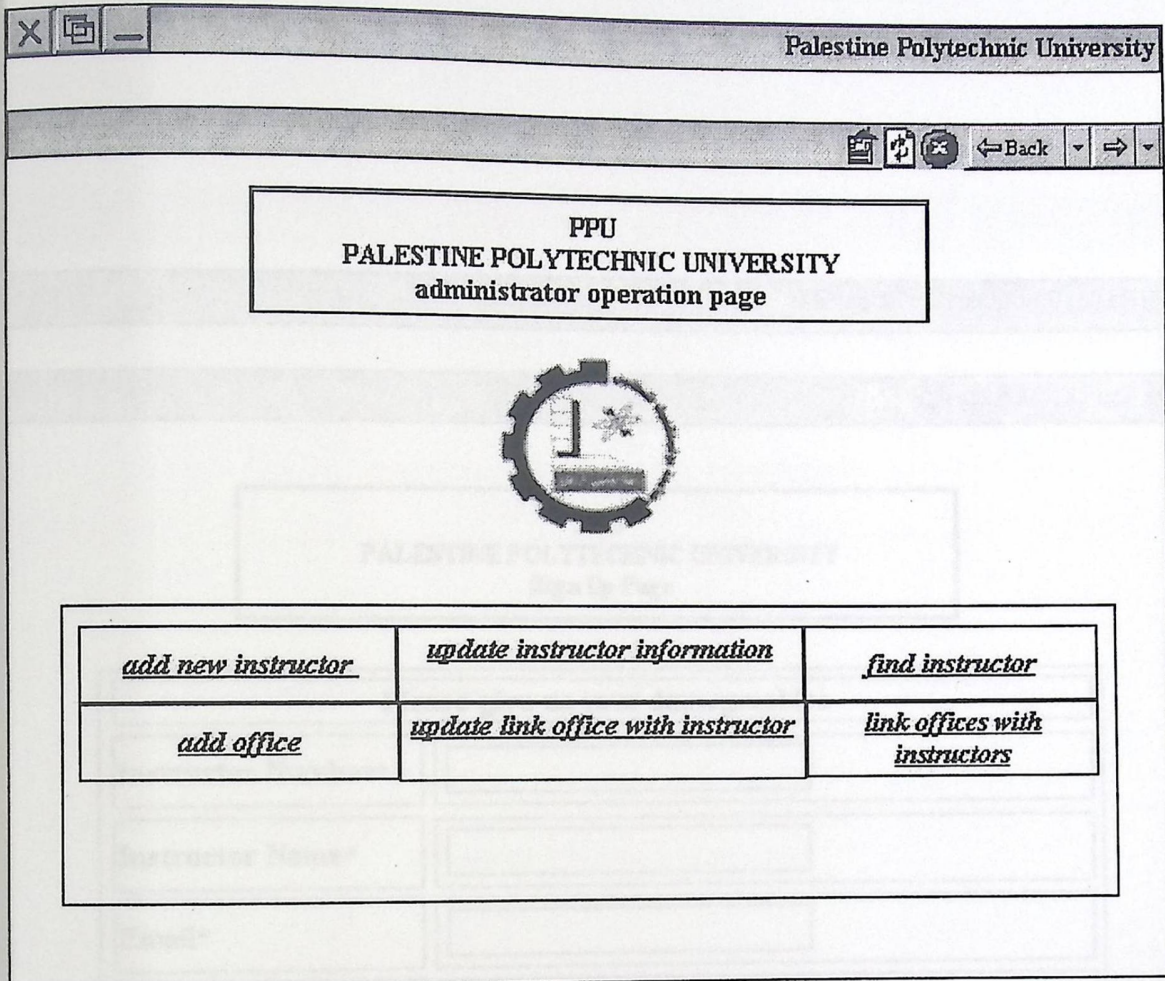


figure (4.20) administrator operation page
(instructor)

2.3.1 Add New Instructor form

if the administrator select add new instructor the add new instructor form will display. Then the confirm page will display and the success page at the end will be displayed.

Please give us your demographics	
Instructor Number*	<input type="text"/>
Instructor Name*	<input type="text"/>
Email*	<input type="text"/>
Your Web Site*	<input type="text"/>
User Name*	<input type="text"/>
Please provide your Password	
Password*	<input type="text"/>
confirm password *	<input type="text"/>
Member Login Here!	<input type="button" value="Sign Up Now!"/> <input type="button" value="Cancel"/>

figure (4.21) add new instructor form

2.3.1 Add New Instructor form

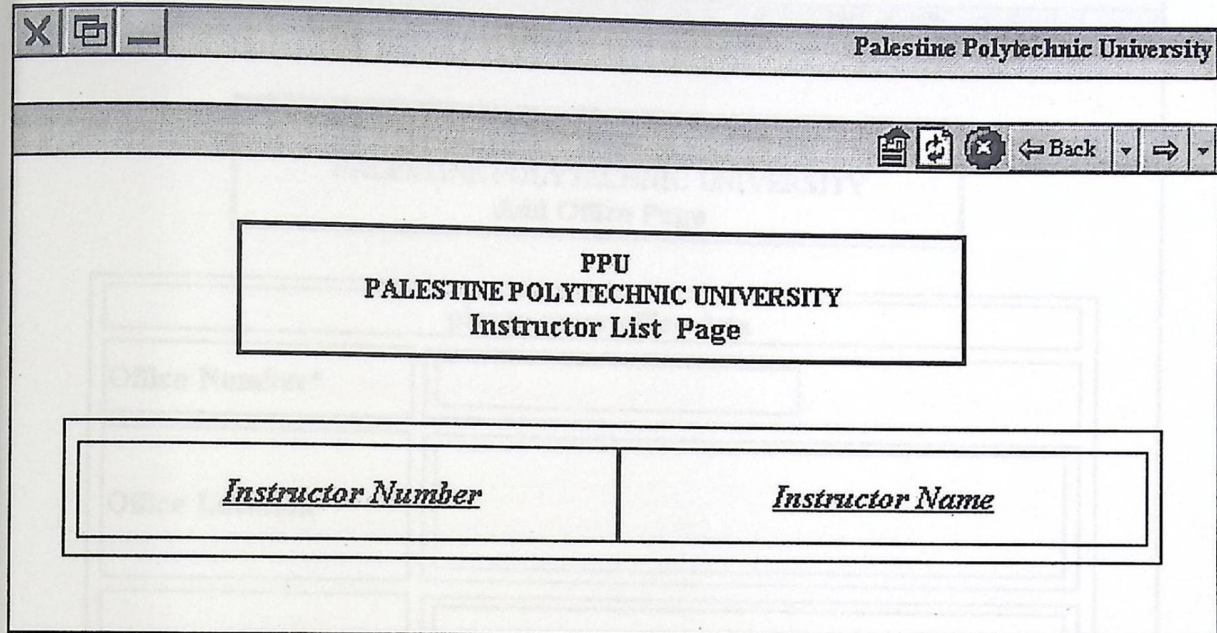
if the administrator select add new instructor the add new instructor form will display. Then the confirm page will display and the success page at the end will be displayed.

Please give us your demographics	
Instructor Number*	<input type="text"/>
Instructor Name*	<input type="text"/>
Email*	<input type="text"/>
Your Web Site*	<input type="text"/>
User Name*	<input type="text"/>
Please provide your Password	
Password*	<input type="text"/>
confirm password *	<input type="text"/>
Member Login Here!	<input type="button" value="Sign Up Now!"/> <input type="button" value="Cancel"/>

figure (4.21) add new instructor form

2.3.2 Find Instructor Page

if the administrator select fined instructor from the instructor operation page this page will be displayed and show all the instructor number and name in the database.



figuer (4.22) fined instructor

4.3.4 Add New Office form

If the instructor select from the instructor operation page add new office this page will display to feed the office data.

The screenshot shows a web browser window titled "Palestine Polytechnic University". The browser's address bar and navigation buttons (Back, Forward) are visible. The main content area displays a form titled "PPU PALESTINE POLYTECHNIC UNIVERSITY Add Office Page". The form contains the following elements:

- A header box with the text: "PPU PALESTINE POLYTECHNIC UNIVERSITY Add Office Page".
- A section titled "please enter office data" containing three input fields:
 - Office Number***: A text input field.
 - Office Location***: A text input field.
 - Office Phone***: A text input field.
- At the bottom of the form, there are two buttons: "Add To Database" and "Cancel".

figure (4.22)add new office form

2.3.5 Joining Office With Instructor

If the administrator select link office with instructor operation this form will display with list of offices and instructors.

The screenshot shows a web browser window titled "Palestine Polytechnic University". The browser's address bar and navigation buttons (Back, Forward) are visible. The main content area displays a form titled "PPU PALESTINE POLYTECHNIC UNIVERSITY Join Offices With Instructors Page". Below the title, there is a text prompt: "enter office number and instructor number you can get it from the list". The form itself contains a header box with the text "please enter course number & instructor number". Below this, there are two input fields: "Office number" and "Instructor Number*". At the bottom of the form, there are two buttons: "Add To Database" and "Cancel".

figure (4.23) join office with instructor form

2.4 Administrators Operation Page (Course)

2.4.1 Add new course form

If the administrator select ad new course from the course operation pages this from will be display.

The screenshot shows a web browser window titled "Palestine Polytechnic University". The browser's address bar and navigation buttons (Back, Forward) are visible. The main content area displays a form titled "PPU PALESTINE POLYTECHNIC UNIVERSITY Add Course Page". The form is titled "Please Enter Course Data" and contains the following fields:

Please Enter Course Data	
Course Number*	<input type="text"/>
Course Name*	<input type="text"/>
Description*	<input type="text"/>
Prerequisites*	<input type="text"/>
Credit hour*	<input type="text"/>
	<input type="button" value="Add To Database"/> <input type="button" value="Cancel"/>

figure (4.24) add new course form

2.4.2 Link course with instructor form

if the instructor select link course with instructor from the course operation page this form is display with list of courses and instructors.

The screenshot shows a web browser window with the following elements:

- Browser title bar: Palestine Polytechnic University
- Navigation bar: Home, Refresh, Stop, Back, Forward buttons.
- Page header: PPU PALESTINE POLYTECHNIC UNIVERSITY Join Course With Instructors Page
- Text: enter office number and instructor number you can get it from the list
- Form area:
 - Header: please enter course number & instructor number
 - Field 1: Course Number* (input field)
 - Field 2: Instructor Number* (input field)
 - Buttons: Add To Database, Cancel

figure (4.25)link course with instructor from

2.4.3 Update course information

If the administrator select update course information from the course operation page first the course list page will display to show the entire course in the database.

1. Course list page

This page contain list of course number and course name.

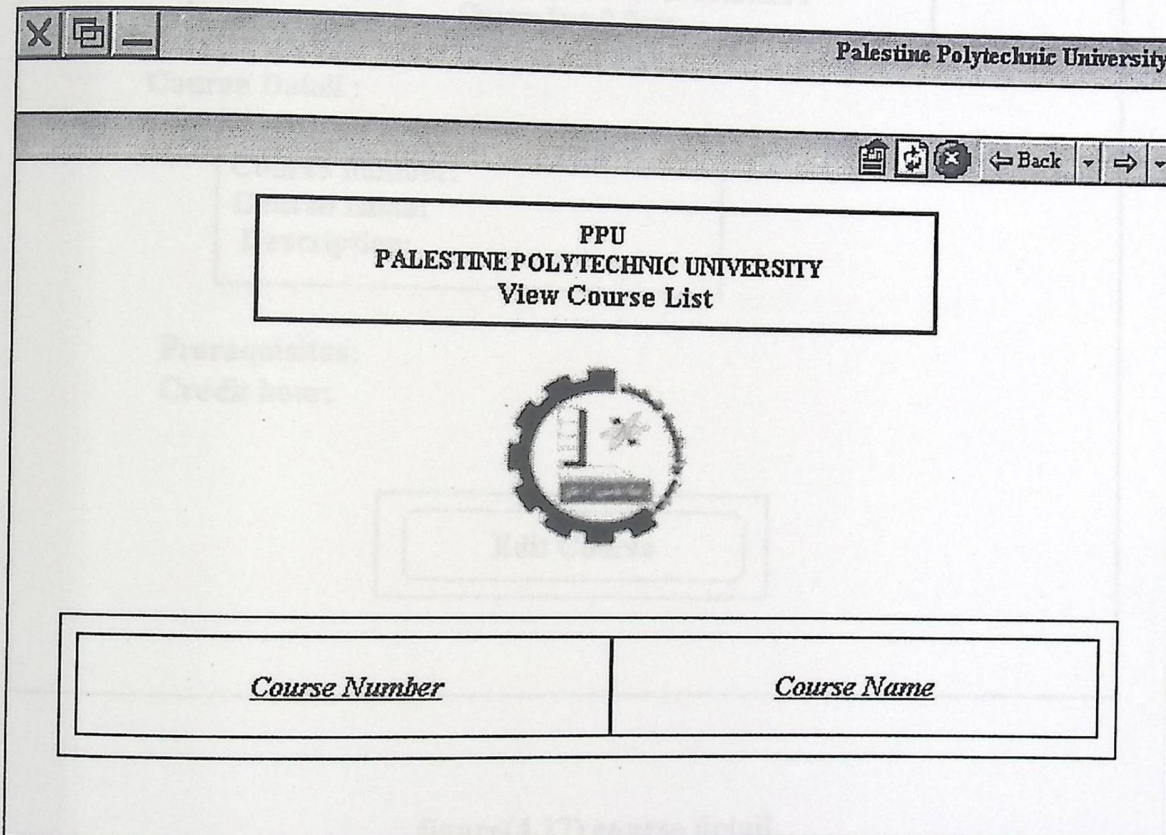
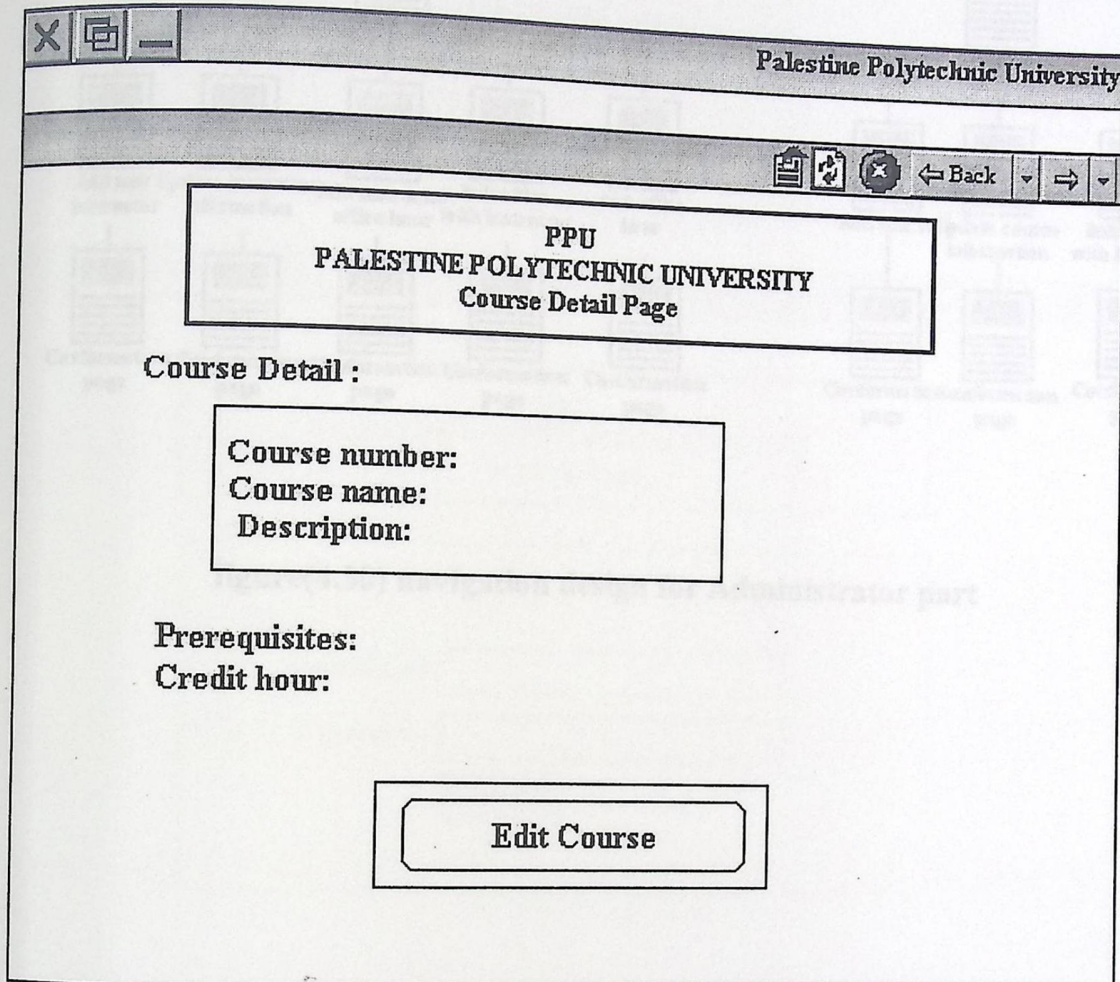


figure (4.26) course list page

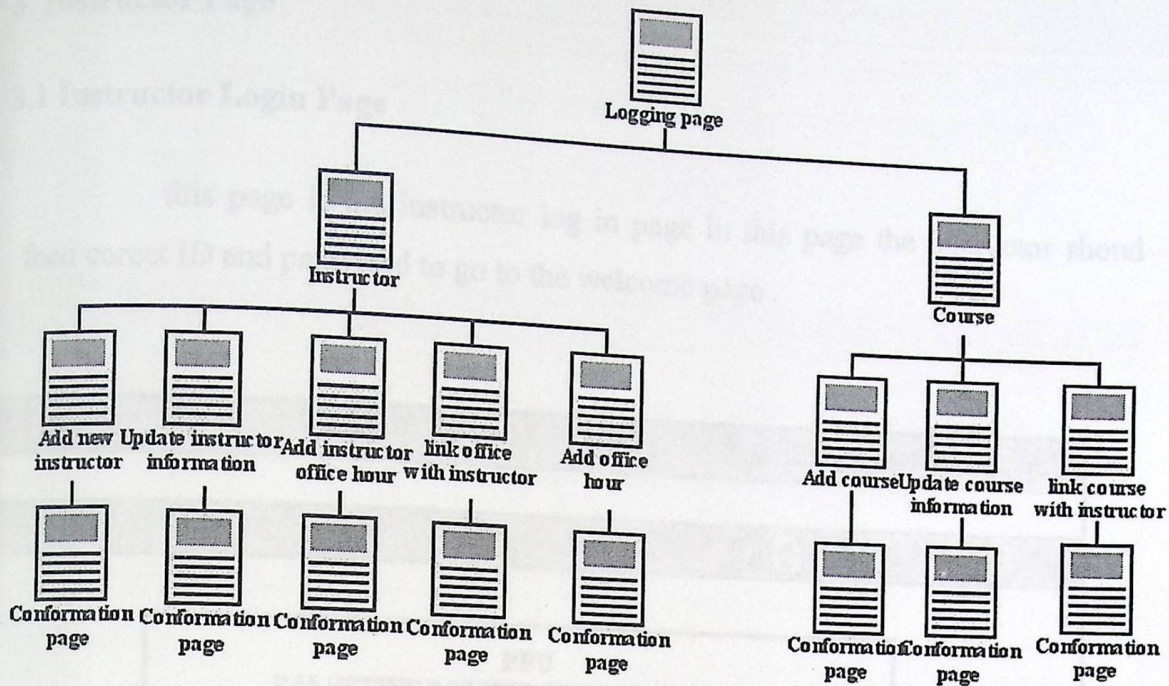
2. Course detail page

this page contain the course detailed like course number name description prerequisites and credit hour ,when the administrator click edit course this go to the course from to edit the course information.



figure(4.27) course detail

navigation design for Administrator part



figure(4.30) navigation design for Administrator part

3. Instructor Page

3.1 Instructor Login Page

this page is the instructor log in page in this page the instructor should feed correct ID and password to go to the welcome page .

The image shows a web browser window displaying the instructor login page for Palestine Polytechnic University. The browser's title bar reads "Palestine polytechnic university". The address bar contains navigation icons and a "Back" button. The main content area is enclosed in a rectangular frame and contains the following elements:

- A header box with the text: **PPU**
PALESTINE POLYTECHNIC UNIVERSITY
Logging Page
- A "User Name:" label followed by a text input field.
- A "Password:" label followed by a text input field.
- A "Login" button positioned below the password field.

figure(4.31) instructor login page

3.2 Wellcome Page

this is welcome page that's mean the instructor feed corect ID and password.

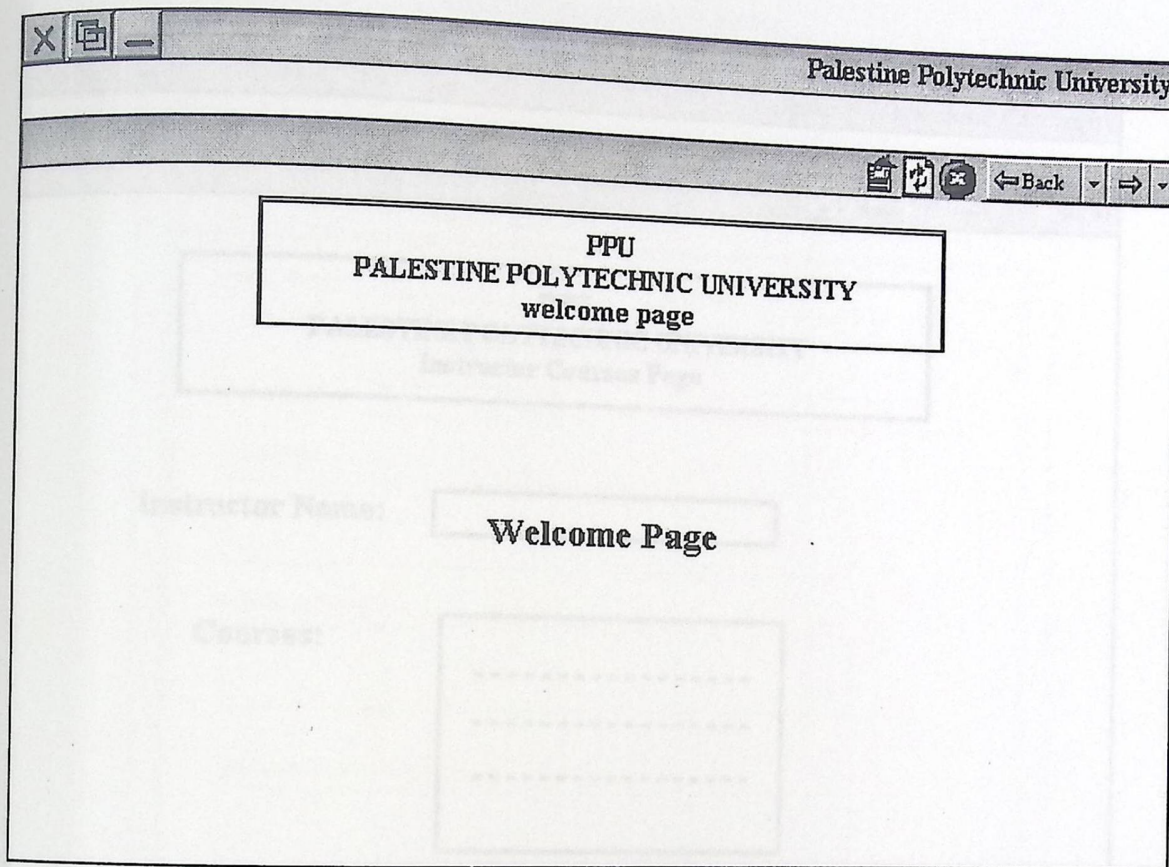


figure (4.32) welcome page

3.4 add new syllabus form

add new syllabus from which allow the instructor to add new course syllabus .

The screenshot shows a web browser window titled "Palestine Polytechnic University". The page content is as follows:

- Header: PPU PALESTINE POLYTECHNIC UNIVERSITY Add New Syllabus Page
- Form fields:
 - Instructor Name:
 - Courses:
 - Syllabus Number: Semester:
 - Enter Course & Instructor Number
 - Course Number: Instructor Number:
 - Description:
 - Attendance Policy:
 - Scholastic Dishonesties Policy:
 - Additional Information:
 - News:
- Submit button:

figure (4.34) add new syllabus form

3.5 add syllabus contents

1. Instructor syllabus list

When the instructor select update syllabus list from the operation page the instructor syllabus list will display this page contain instructor name course name and list of syllabus.

The screenshot shows a web browser window with the title bar 'Palestine Polytechnic University'. The browser's address bar contains navigation icons and a 'Back' button. The main content area displays the following text:

PPU
PALESTINE POLYTECHNIC UNIVERSITY
Instructor Syllabus List

Syllabus's List :

Instructor: _____

Course: _____

Semester: _____

figure (4.35) instructor syllabus list

2. list of syllabus contents

When the instructor selects update of syllabus content this page will display and contain list of syllabus content to add any content click to it to go to add form.

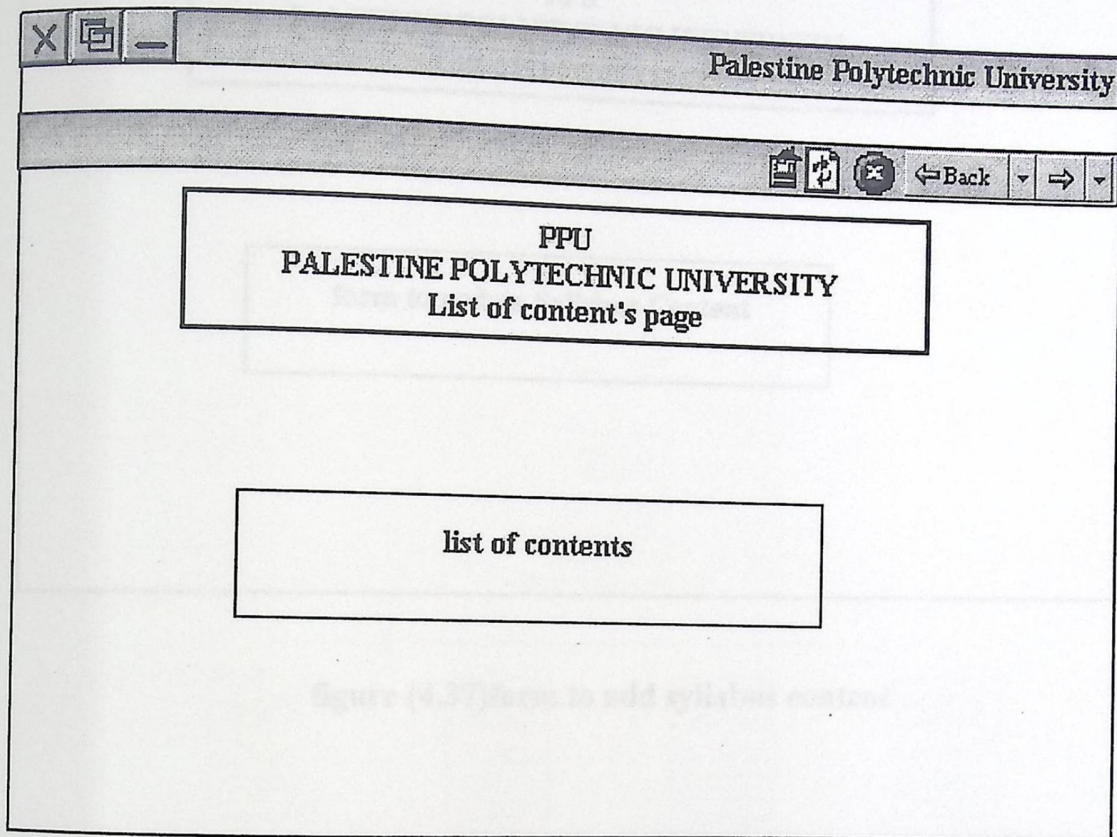


figure (4.36) list of syllabus content

3. Form to add syllabus contents

This from is display to any content selected from the content list

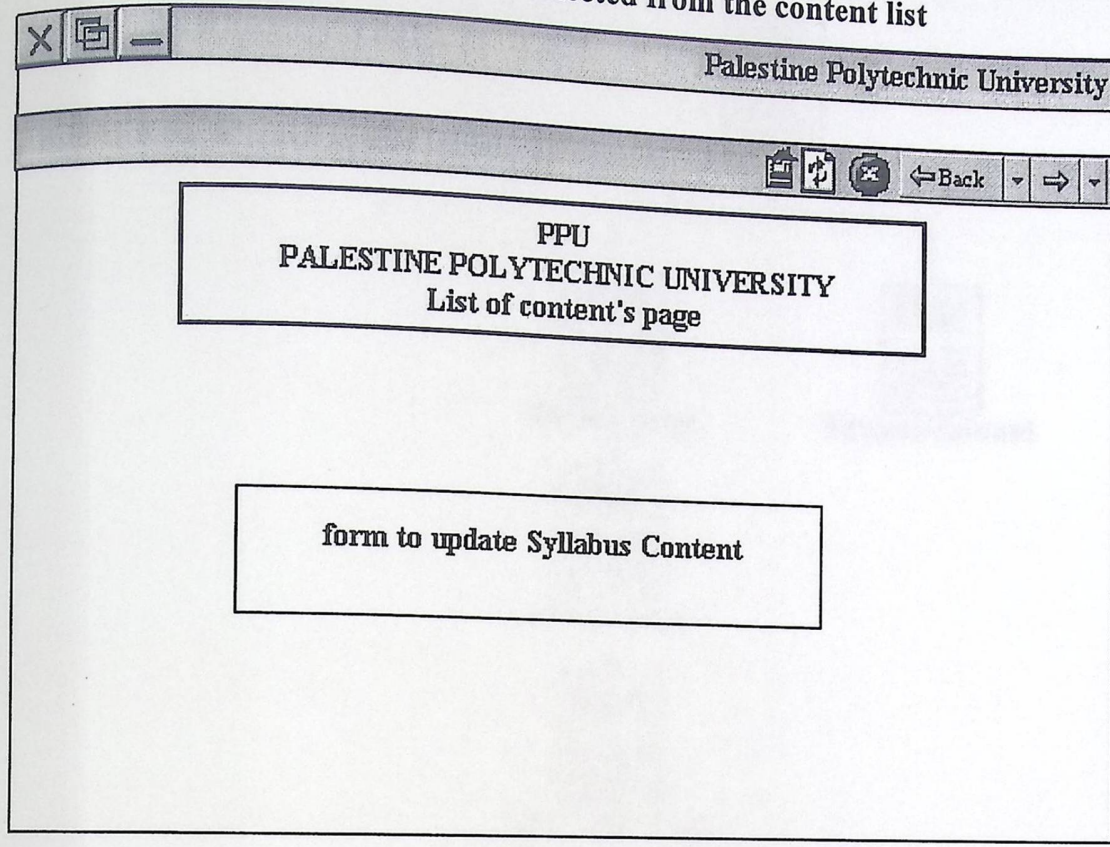


figure (4.37)form to add syllabus content

Navigation design for Instructor part:

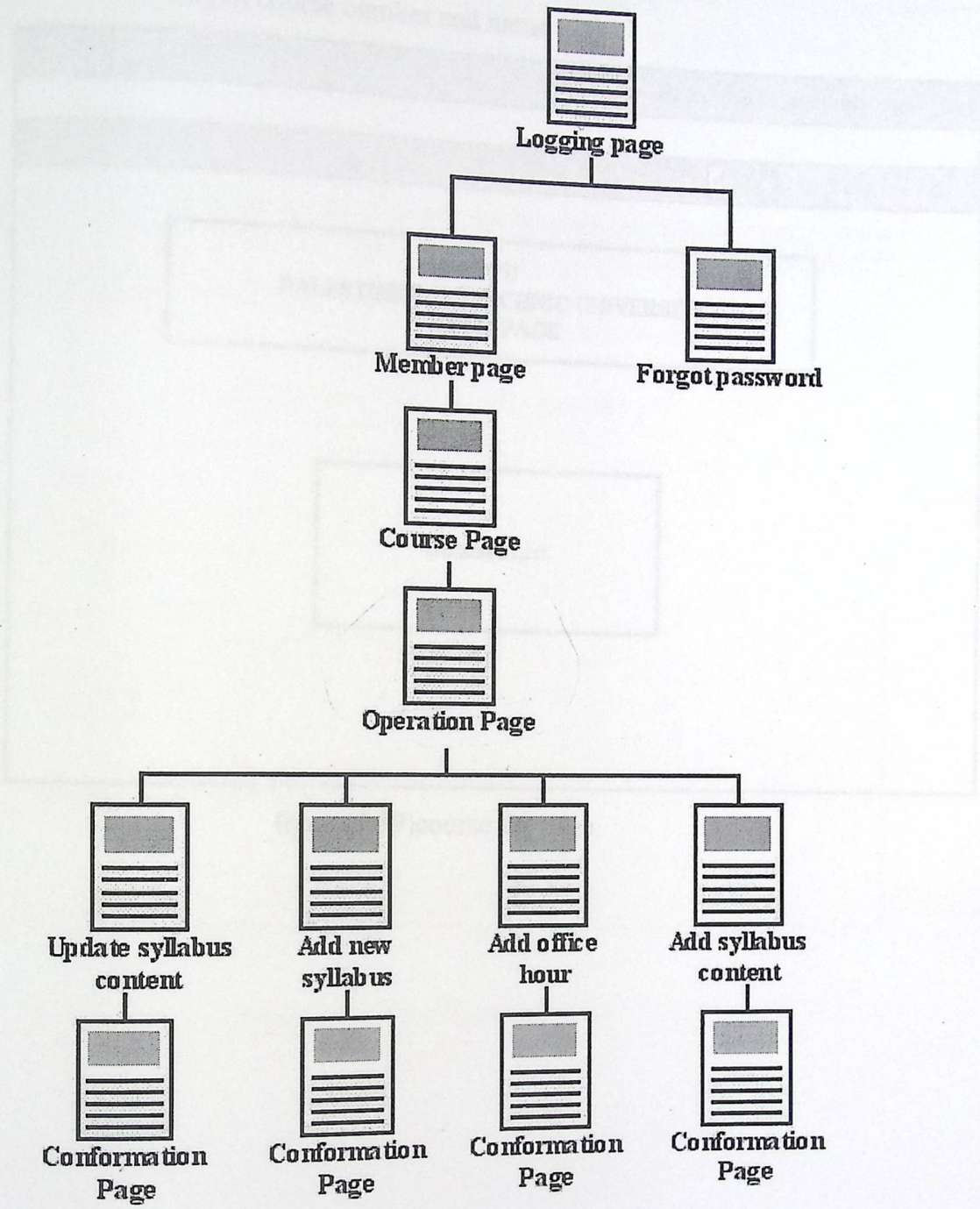
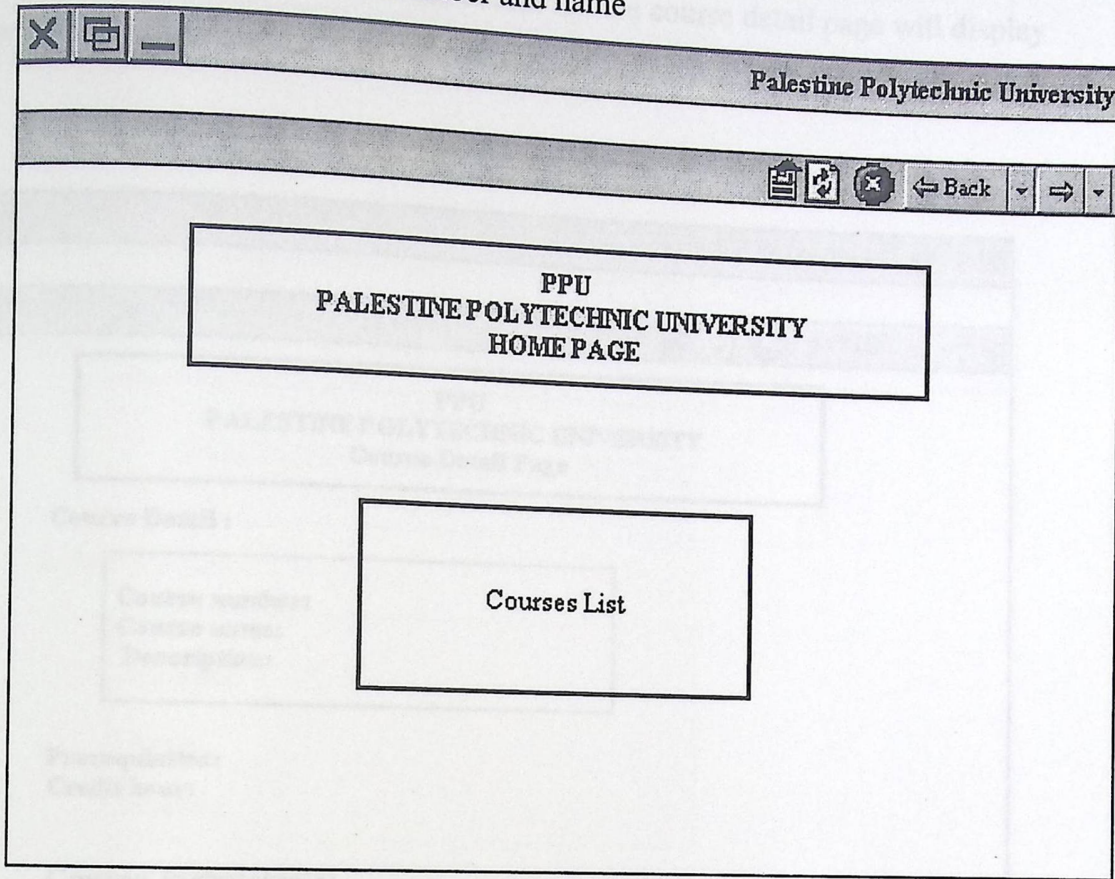


Figure (4.38) Navigation design for Instructor part

3. View Syllabus Content

3.1 Course List Page

list of course number and name



figure(4.39)course list page

3.2 Course Detail page

when the viewer select course from the course list the course detail page will display to show course information and detail. with list of instructors name who teach this course

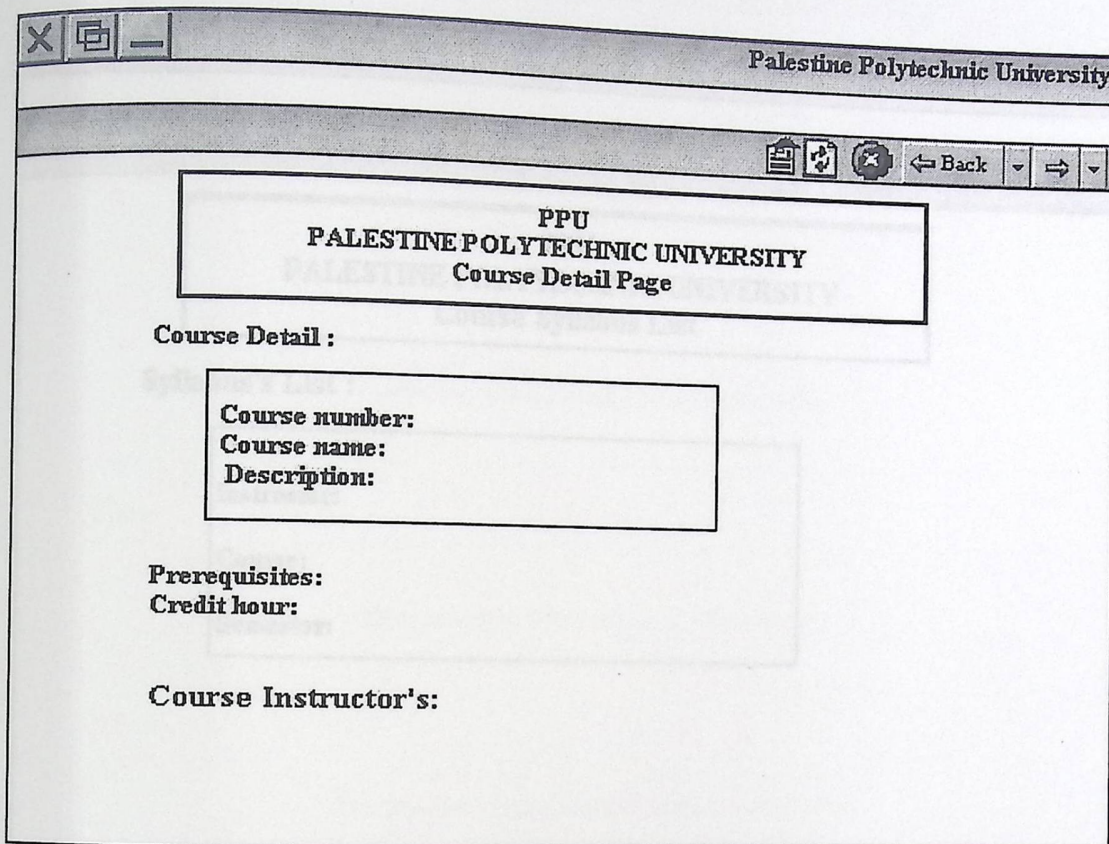
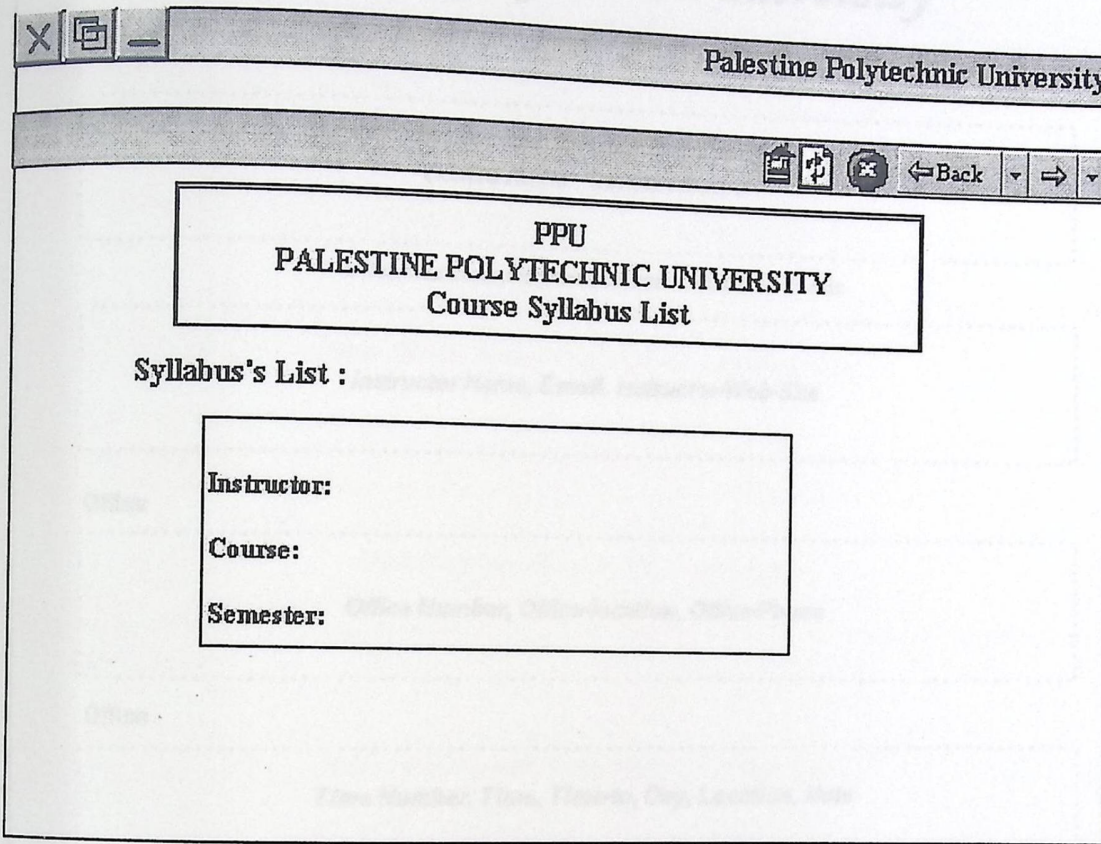


figure (4.40)course detail page

3.3 Course Syllabus List

This page contains course name and the instructor name with all syllabus and semester number.



figure(4.41) course syllabus list

3.4 Viewing syllabus page

When the viewer select any semester this page will display to show course syllabus

Palestine Polytechnic University

Palestine polytechnic university

Course Name , Course Number

Instructor Name, Email, Instructor-Web-Site

Office

Office Number, Office-location, Office-Phone

Office

Time Number, Time, Time-to, Day, Location, Note

Course Description

Description

Course Goal

Goal Number, Goal

Activates To Achieve The Goals:

Office Number, Office-location, Office-Phone

Course Expectation

Expectation Number, Expectation

Needed Skills:

Skill Number, Back-Ground

Class Strategy:

Class Strategy Number, Strategy

Time:

Type, STNO, Time, Time-To, Day, Location, Note

Text Books:

Book Number, Title, Author, Edition, Note

References:

Reference Number, Title, Author, Edition, Note

Useful Links:

Useful Link Number, Link, Office-Phone

Type

Note

Course Homeworks:

Home Works Number, Syllabus number, Office-Phone

Course Projects:

Syllabus Number, Projects Number

Common Questions:

Syllabus Number, Common Question Number, Office-Phone

Course Exam:

Syllabus Number, Exam Number, Office-Phone

Grading Policy:

Grading Number, Grading Name, Mark

Attendance Policy:

Attendance Policy

Scholastic Dishonesty Policy:

scholastic-Dishonesty-Policy, Office-location, Office-Phone

News:

News

figure (4.42) course syllabus page

3.5 Exam Page

This page contains the detail of exam that selected from the course syllabus page.

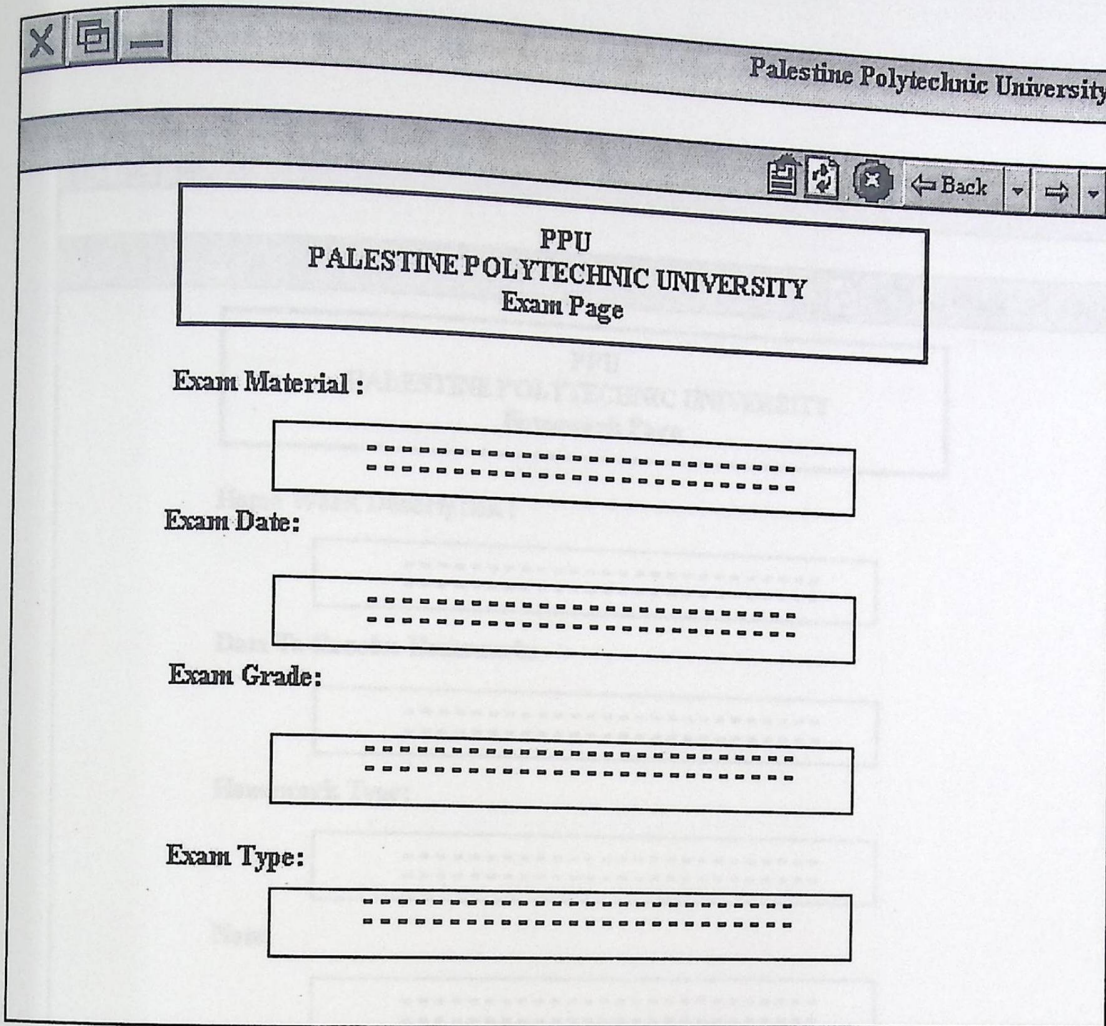


figure (4.43) exam page

3.6 Home Work Page

This page contains the detail of home work that selected from the course syllabus page.

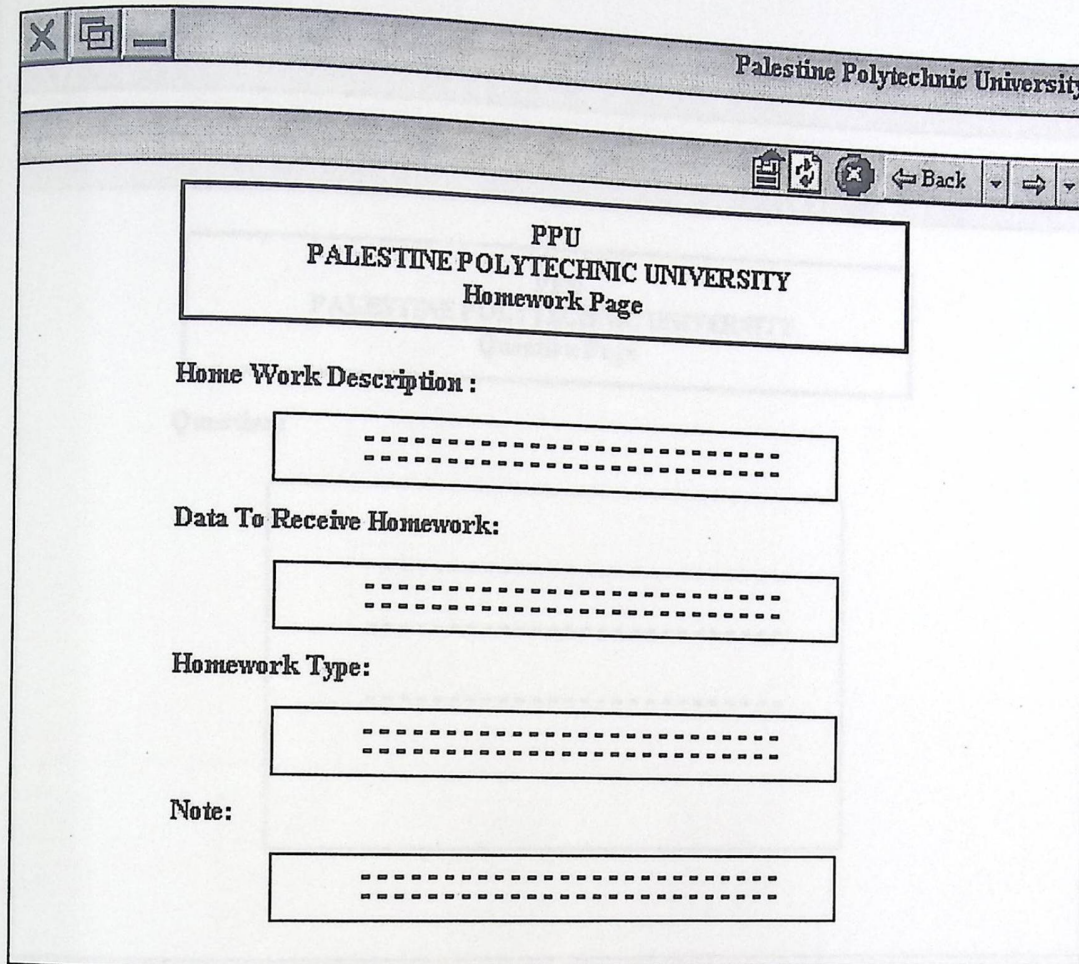


figure (4.44) home work page

3.6 common Question Page

This page contains the detail of common question that selected from the course syllabus page.

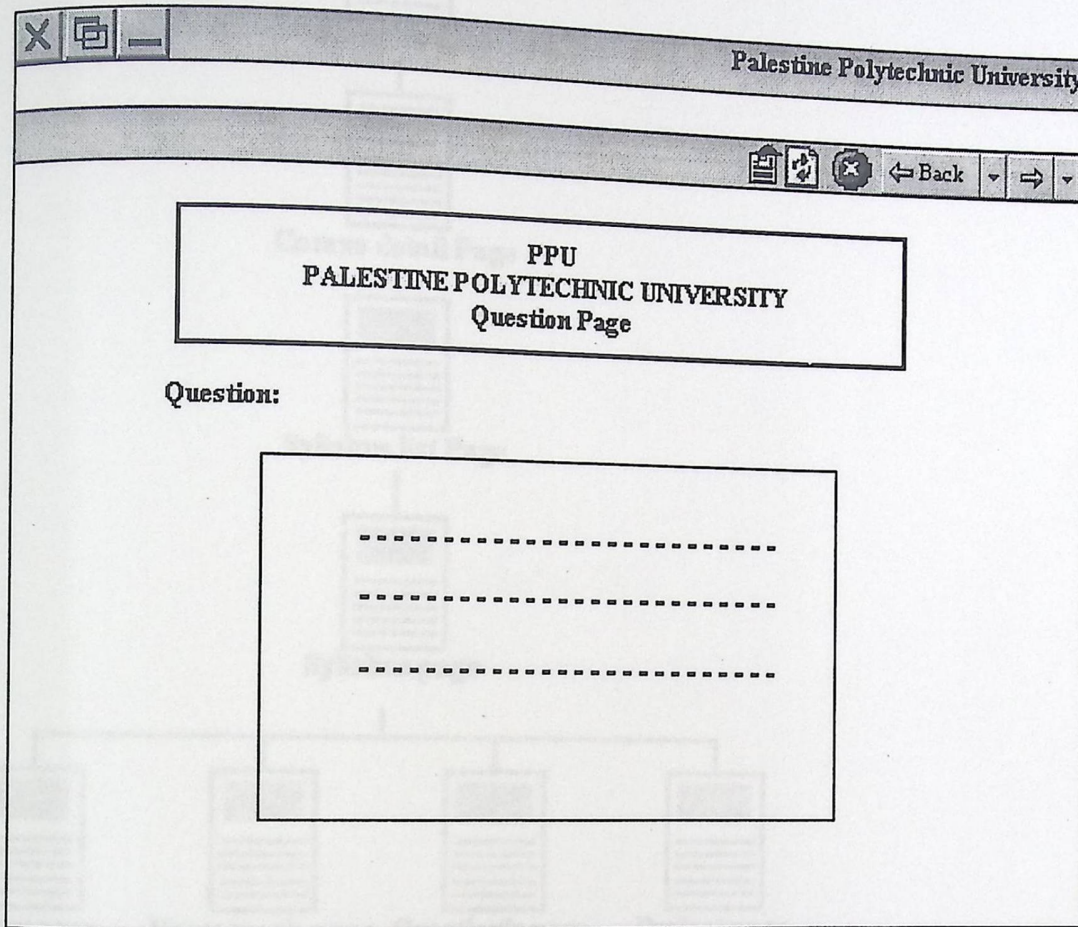


figure (4.45) common question page

Navigation design for view course syllabus content

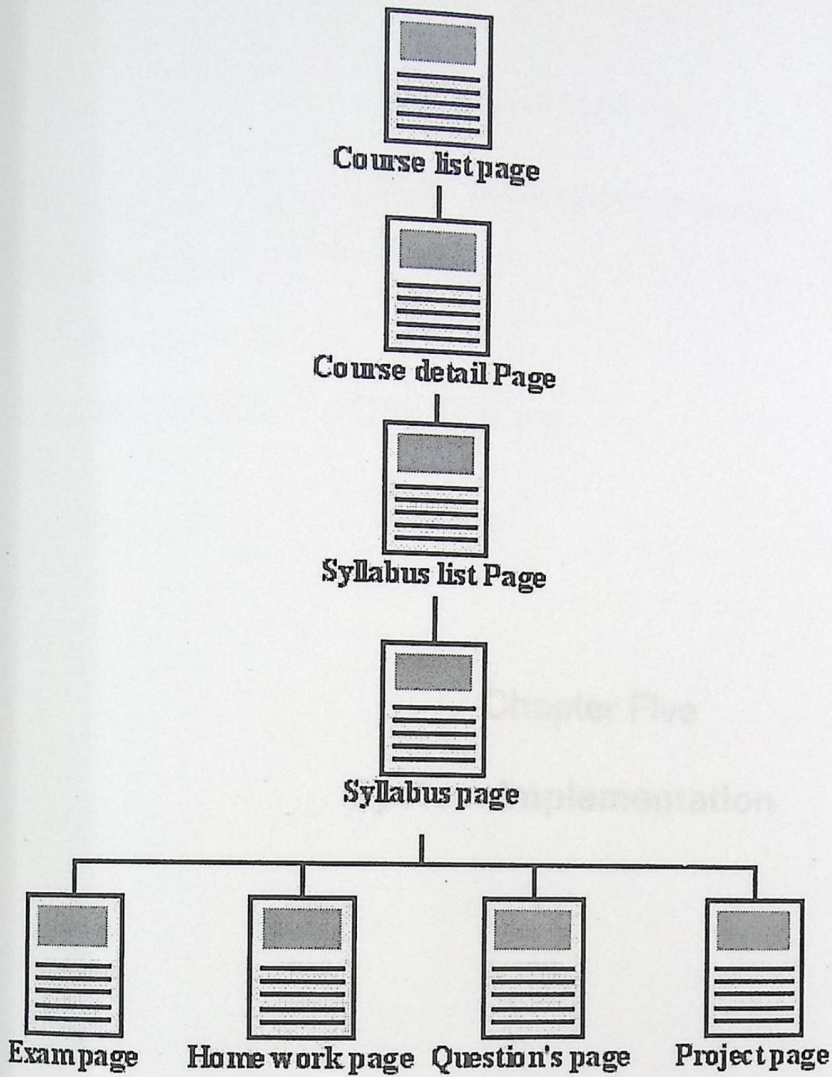


Figure (4.46) Navigation design for view course syllabus content

Chapter Five

System Implementation

- 5.1 Introduction
- 5.2 Software Needed
- 5.3 Implementation of the System

Chapter Five

System Implementation

Chapter Five

System implementation

5-1 Introduction

5-2 Software Needed

5-3 Implementation of the System

5-2 Software Needed

This section describes the software needed for this system and how we use it to implement this system.

5-2-1 Windows 2000 Advanced Server

Windows 2000 Advanced Server is a powerful operating system. It supports many features needed in this system in order to support the clients with many services, like the web server which supports clients of browsing web pages.

Windows 2000 Advanced Server includes many components that support the clients with many services. One of those components is Domain Name System (DNS) which contains the users, computers in domain. Another component is the Internet Information Services (IIS), which allows us to publish our page through

5-1 Introduction

In recent years the Internet became the most widely used form of communication, because of the important services that it provides, one of the main service that the internet provides us is administrating the content of database remotely. This system takes some of the Internet advantages and applies it to making the lecturer show the syllabus of the courses and administrate the content of database remotely.

This chapter describes the implementation of the functions applied in this system. To implement the required functions in this system, some of software tools must be installed on the server such as the Windows 2000 Advanced Server, Microsoft SQL Server 2000, Microsoft Office 2000, Microsoft Internet Information Server 5.0 (IIS) and the Internet Explorer 6.0; on the client machine we need Windows 2000 professional.

5-2 Software Needed

This section describes the software needed for this system and how we use it to implement this system.

5-2-1 Windows 2000 Advanced Server

Windows 2000 Advanced Server is a powerful operating system; it supports many features needed in this system in order to support the clients with many services, like the web server which supports clients of browsing web pages.

Windows 2000 Advanced Server includes many components that support the clients with many services. One of these components is Domain Name services which contain the users, computers in domain. Another component is the Internet Information Services (IIS), which allows us to publish our page through

the internet, and take our data from the database through connector known as ODBC.

5.2.1.1 Open Database Connectivity

(ODBC)

Is an open standard application programming interface (API) for accessing a database. By using ODBC statements in a program, you can access files in a number of different databases, including Access, dBase, DB2, Excel, and Text. In addition to the ODBC software, a separate module or driver is needed for each database to be accessed. The main proponent and supplier of ODBC programming support is Microsoft.

ODBC is based on and closely aligned with The Open Group standard Structured Query Language (SQL) Call-Level Interface. It allows programs to use SQL requests that will access databases without having to know the proprietary interfaces to the databases. ODBC handles the SQL request and converts it into a request the individual database system understands.

How we can open ODBC: (Steps)

1. Open control panel, double click on administrative tools, then double click on data sources (ODBC).
2. will appear the dialog box that show in figure (5-1), first we choose the system DSN then we do the next options:
 - Click on Add establish a new name DSN System.
 - Click on remove will remove an exist DSN System name.First we determine the DSN System name that wants to delete then click on remove button.

- Click on configure will show or update an exist DSN System name. Determine the DSN System name that want to update on then click on configure button.
3. Click on Add button to appear another dialog box that show on figure (5-2).
 4. Determine database type you want to use then click on finish button.

From this point the dialog box and procedure will change according to the type of database that will be used. In user manual we explain the procedure uses and steps to define the DSN System for two databases like Microsoft Access Driver, and SQL Server.

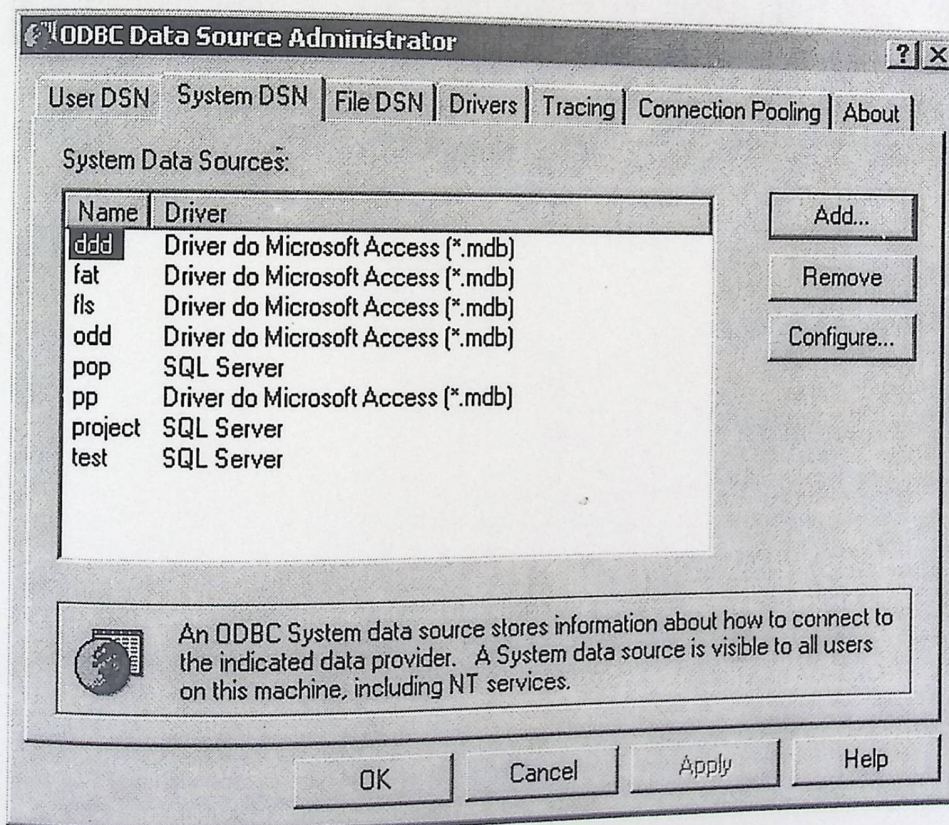


Figure (5-1).

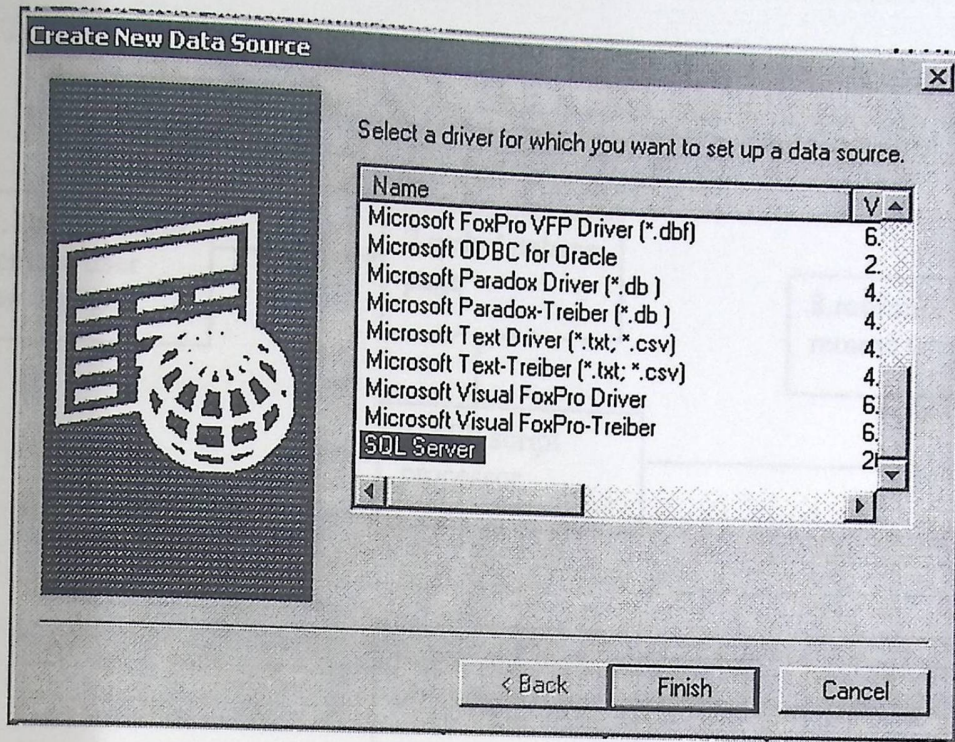


Figure (5-2).

To know how the information flow when we use web database see figure (5-3):¹

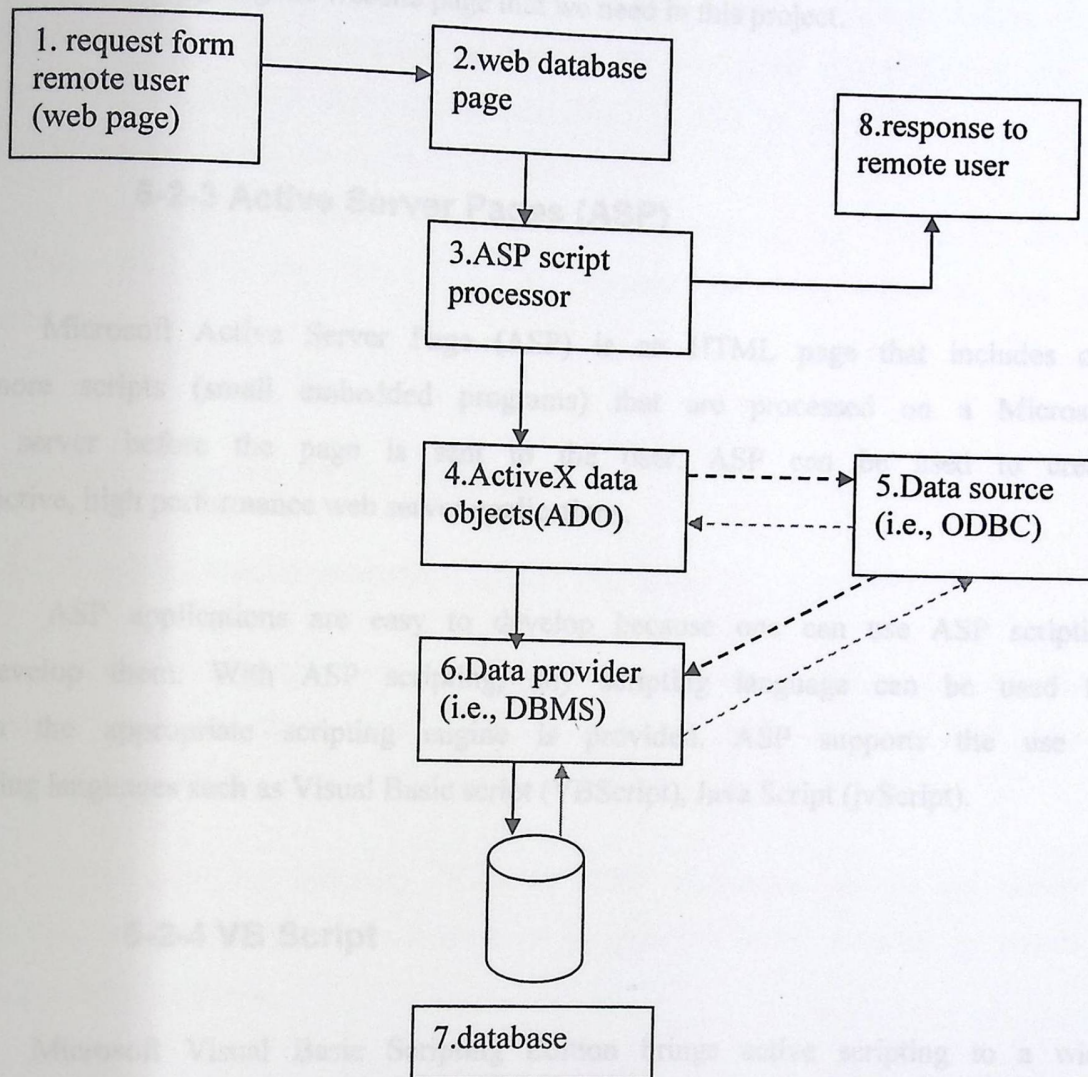


Figure (5-3) Database web require some layers from software's to treat it.

¹ Jim Buyens, Web Database Development Web Site Step By Step, Arabic Scientific Publishers, 2000.

5-2-2 Microsoft Office 2000

Microsoft office 2000 includes, Microsoft FrontPage 2000 which is a special tool for designing the website page that we need in this project.

5-2-3 Active Server Pages (ASP)

Microsoft Active Server Page (ASP) is an HTML page that includes one or more scripts (small embedded programs) that are processed on a Microsoft Web server before the page is sent to the user. ASP can be used to create interactive, high performance web server applications.

ASP applications are easy to develop because one can use ASP scripting to develop them. With ASP scripting, any scripting language can be used for which the appropriate scripting engine is provided. ASP supports the use of scripting languages such as Visual Basic script (VBScript), Java Script (JavaScript).

5-2-4 VB Script

Microsoft Visual Basic Scripting Edition brings active scripting to a wide variety of environments, including Web client scripting in Microsoft Internet Explorer and Web server scripting in Microsoft Internet Information Service.

5-2-5 Structured Query Language (SQL)

Structured Query Language (SQL), in computer science, a database sublanguage used in querying, updating, and managing relational databases. Although it is not a programming language in the same sense as C or Pascal, SQL can either be used in formulating interactive queries or be embedded in an application as instructions for handling data.

The SQL standard also contains components for defining, altering, controlling, and securing data. SQL is designed for both technical and non technical users.²

5-3 Implementation of the system

5-3-1 Initial implementation of the system

In order to implement our system we need two machines, one acts as a server and the other as a client, in order to build a LAN(Local Area Network) network, in which we will use to implement our system.

We start by installing Windows 2000 Advanced Server on server machine because it has the capabilities to provide many services to clients that we name as server.

Now we talk about the steps for installing windows 2000 advanced server from a CD-ROM onto a clean hard disk:

1. run the set up program.
2. run the setup wizard.
3. install windows 2000 networking.
4. complete the setup program.

² Encarta encyclopedia 2002.

After installing Windows 2000 Advanced Server we configure the TCP/IP protocol.

Steps for configure TCP/IP protocol:

1. right click my network places then choose proprieties.
2. now choose local area connection then right click to choose proprieties.
3. the dialog box will appear, from this dialog box will choose protocol TCP/IP.
4. we fill the fields by assigning IP address (192.186.1.2), subnet mask (255.255.255.0), default gateway (192.186.1.2) and preferred server (192.186.1.2).

After that we installing the require system components that needs in our work, one of these components are the Networking Services, it includes Domain Name Services (DNS) that contains all the computers in our network. How we installing the Networking Services through these steps:

1. click on start then from setting choose control panel.
2. double-click add/remove programs.
The Add/Remove programs window appears.
3. Click Add/Remove Windows Components.
The Windows Components wizard appears.
4. Check Networking Services then click next.
5. the system will install the Networking Services.

To install DNS do the next steps:

1. Open Control Panel.
2. double-click add/remove programs.

- The Add/Remove programs window appears.
3. Click Add/Remove Windows Components.
The Windows Components wizard appears.
 4. Click Networking Services then click details.
 5. In subcomponents of networking services list box, select the check box to the left of Domain Name System (DNS).
 6. Click ok. You are returned to the Windows Components page.
 7. Click Next. Setup copies the required files to the hard disk.
 8. Click finish to close the Windows Components wizard.
 9. Close the Add/Remove Programs window.
 10. Close control panel.

We install the Internet Information System (IIS), that enables us to publish our web page on the Internet.

Steps for install IIS:

1. Open Control Panel.
2. double-click add/remove programs.
The Add/Remove programs window appears.
3. Click Add/Remove Windows Components.
The Windows Components wizard appears.
4. Check the Internet Information System (IIS).
5. Click Next. Setup copies the required files to the hard disk.
6. Click finish to close the Windows Components wizard.
7. Close the Add/Remove Programs window.

After that we configure the DNS services by creating our zone, through this steps:

1. Click start, point to Programs point Administrative Tools, and then click DNS.
2. Select create forward lookup zone, and click next.
3. verify that standard primary option is selected, then click next.
4. Type domain.com and click next.
5. Ensure that the create A New File With This File Name option is selected and that the name of the file to be created is domain.com.dns.
6. Finish configure the DNS Server Wizard.

We build our web site that we called 'www.syllabus.edu' to represent the domain, from the default web site that found in the IIS. This setup is necessary to implement this system and to be able to control the contents of the Web site more efficiently.

In our system IIS provides security features that enables us to have secure web page because the NT Windows support that, and the database give us another level of security by using user name (ID), password.

On the other side we installed Windows 98 on the Client Machine, and then we assign IP address (192.186.1.3) to it, in order to be discovered by the server. The need of this machine is to test our work, we mean to reach server through the Internet Explorer by putting the IP, which is given to the page that we have published through the IIS or the name of the page.

5.3.2 Implementation of System Pages

In our system we have the some pages as: Operation page, Conform page, Sign up page, Success page, Administrator operation page, Logging page, View course list, Add new syllabus page, Add office page, etc...

1. Implementing Logging page(For example)

We construct this page by using ASP code, HTML through Microsoft Front Page 2000. This page in our system show how the instructor login to the system through entering user name and password. We build this page on front page 2000 by adding first form then two text box one for user name, and the other for password to submit this page click on submit button on the HTML code for this page the action is send this data to verifying page that check if user name, password is valid or not, by using post method.

5.3.3 Implementing ASP codes

In this section we discuss the process of implementing ASP codes that are used in design this pages. for example we talk about book1.asp page.

In this page we use the ASP code to we insert form in this form we have some of text boxes that we are used to insert data about books that teaches on semester on ASP code the form method is post, and the action is (confirm.asp) page.

We open the connection on this page by using the ODBC connection that we named project, this connection link to the SQL server.

Chapter Six
Testing and Comments

- 6.1 Introduction
- 6.2 Unit Testing
- 6.3 Integration Testing
- 6.4 Verification Testing
- 6.5 Acceptance Testing

Chapter Six

Testing and Comments

Chapter Six Testing and Comment

- 6-1 Introduction
- 6-2 Unit Testing
- 6-3 Integration Testing
- 6-4 Validation Testing
- 6-5 Acceptance Testing

Inputs	Actual	Expected Results	Actual Results	Comments
1. Login name	John Smith	Access granted	Access granted	
2. Password	123456	Access denied	Access denied	
3. Login name	John Smith	Access granted	Access granted	
4. Password	654321	Access denied	Access denied	

6-1 Introduction

After finishing the implementation we have to check that our system works correctly or not.

There are many testing techniques that we are going to use to discover system faults, these techniques are the Unit testing, Integration testing, Validation testing, Acceptance testing.

6-2 Unit Testing

In this technique we test each unit of the system separately from other units, this unit is examined to ensure that it works correctly or not, for example if we want to test the login operation we enter the required values, then after clicking on submit we see the result and this result appears in the web page that tells us if the user name, and password is valid or not valid.

We also check the correctness of the subroutine by entering wrong values to check that our subroutine is working correctly or not, and noticing the results if our subroutine takes the wrong values we have entered so we need to make changes of our subroutine code.

In this table we check the unit testing

Process case	Input	Expected Result	Actual Result	Comments
Login case	User name: Alsaheb. Password: Alsaheb.	User name, and password is true	Appears member page	In this case we enter the correct data
Login case	User name: Ahmed	User name, and password is false	Return to login page	In this case we enter wrong

	Password: Ahmed			data and our code work correct because it did not appear the member page
--	--------------------	--	--	---

Table (6.1) Show results for unit testing to login operation.

6-3 Integration Testing

In this technique of testing we integrate all the components of our system to test the interactions between the components in our system and component interfaces.

We integrate the subroutine that we have worked, updating (Syllabus, Course, Instructor), adding a new (Syllabus, Course, Instructor, book), etc...all these subroutines are tested as a whole and results are notice for example to adding book to database in this table

Process case	Input	Expected result	Actual result	Comments
Adding book	-Syllabus No. :10 -Book No. :1 -Book Title: principle of economic -Author : -Edition :	All data that we are enter is correct	The confirm page will appears	In this case we enter the correct data so the Confirm page will appears

	-Where to get from it : -Where you are get the book : -Note :			
--	---	--	--	--

Table (6.2) Show result for integration testing to adding book.

6-4 Validation Testing

This type of testing determines if data is valid or not, we mean that data in a Database is valid or not.

Here we test the compatibility of the entered data with the system database, for example in adding course time we have fields to enter time in it if you have enter valid time as (6:00) the data will enter to database because we make the properties for this field like previous example and we format the time in database as time , but if we enter for example (six, or any another format to time)the data will not enter to the database because it not compatible with the format of fields and format of time on database.

6-7 Acceptance Testing

Our system should be consistent with the functional requirements which we have describe in chapter two, we test if the system achieve the functional requirements like allow the instructor to show syllabus online, update on it, add new syllabus.

Chapter seven

Conclusion and future work

Chapter Seven

Conclusion and Future Work

7-1 Conclusion

7-2 Future Work

Chapter seven

Conclusion and future work

Chapter Seven

Conclusion and Future Work

7-1 Conclusion

7-2 Future Work

7-1 Conclusion

In our system we enable the instructor to do some operations on the syllabus like (Add, and Update) on his syllabus. On the other side we enable administrator to add the instructor and the registration can add, and update on the courses details. Our system view the courses, instructor for student to see the syllabus page for his instructor and see contains for this syllabus.

In chapter one we introduced an introduction about our project and it's importance.

Chapter two discusses the functional requirement which includes the requirement definitions, requirement specifications, and non functional requirements but it includes the product requirement, process requirement, and the external requirement.

In chapter three we analyzing the system, we talk about system features, and tools. Then we show the content of course syllabuses.

In chapter four discussed the design for our system and interfaces for it.

Chapter five discusses the implementation of our system, and the steps that are required to build the system.

In Chapter six we talked about the testing techniques we used in our systems.

7-2 Future Work

Our system security is supplied by NT Windows, and database through user name, and password. But we want to promote the level of security by using SSL (Secure Sockets Layer), and build an active directory to support the security by integrating with IIS.

References

Book:

- 1.1 JIM buyens, web database development website .step by step Arabic scientific publishers, 2000.

Web:

- 1.2 www.aspx.com
2. www.asp101.com
3. www.alnoor-world.com/asp/
4. <http://www.absoft2000.com/aspcom.htm>
5. www.CAT.edu

Appendix

- **Course syllabus**

- 1.3 Database Systems**

4125 :Computer Number

ECE461 : Course number

Database system is basically a computerized record keeping " : **Description**
it is a computerized system whose overall purpose is to maintain a system; that is
make that information available on demand. the information concerned information and
the anything that is deemed to be of significance to the individual or organization can be
assist in the system is intended to serve -anything, in other words, that is needed to
[organization". [C.J. Date general purpose of running the business of that individual or
Structure, Software Engineering Data : **Prerequisite**

4 : **Hours Credit**

Location Meeting Time and

Lecture

Days: Sunday, Tuesday

Time: 8:00 - 9:15

Room 106 Place: Building A

Lab

Days: Wed.

Time: 11:00 - 02:00

Place: Building A,

Teaching Methods

Project Lectures, Labs, Group

Course Topics

Class

1.Introduction to Databases

File-Base Systems Traditional

Approach Database

Database Environment Roles in the

DBMS History of

Disadvantages of DBMSs Advantages and

2.Database Environment

ANSI-SPARK Architecture The Three-level

Languages Database

Data Models
Components of DBMSs Functions and Architectures Multi-User DBMSs
3.The Relational Model
Terminology
Integrity Relational
Languages Relational
Views
4.Database Planning, Design, and Administration
Application Lifecycle The Database
Database Design Overview of
5.Entity-Relational Modeling (ER-Model)
Concepts
Constraints Structural
ER-Model The Enhanced
6.Normalization
normalization The Purpose of
dependency Functional
(Normalization (1NF, 2NF, 3NF The process of
7.Introduction to SQL
Introduction
Manipulation Data
Definition Data
8.Query-By-Example(QBE)
Introduction
Queries Building Select
9.Security
Security Database
Controls Computer-Based
Controls Non-Computer-Base
PC Security
Security DBMSs and Web
Risk Analysis
and Privacy Laws Data Protection
10. Transaction Management
Support Transaction
Control Concurrency
Recovery Database
11.Distributed DBMSs
Transaction Management Distributed
Concurrency Control Distributed
Deadlock Management and Recovery Distributed
12.Introduction to Object DBMSs

Applications Advanced Database
Relational DBMSs Weaknesses of
Concepts Object-Oriented

Lab

Part1: SQL

Selecting Data

Selection Advanced Data

DATABASE Tables Creating the

(database Objects (Sequences, views, Indexes Creating Other
PL/SQL Introduction To

Part2: Using Oracle Developer2000

Form Builder Environment Working in the

Blocks and Frames Working with Data

Items Working with Text

LOV's Creating

(Additional Input Items (Check Box, List Box, Radio Groups Creating

Triggers Working with

Alerts Messages and

Modules Working with Menu

part3: Charts and Reports

and reports in Form Including Charts

Wizard Using Chart

Running Reports Designing and

Report Builder Introduction to

Lab Assignments

Lab 1

Lab 2

Lab 3

Lab 4

Required Text

DATABASE SYSTEMS, TOMAS CONNOLLY AND CAROLYN BEGG, 2nd Edition
ADDISON-WESLEY

References

Exams

Page 86 First Exam: To

221 Second Exam: To Page

Part1 : st Lab1
:Final lab
Management System Hotel :Project
:Final Exam

Grading System

%15 :First Exam
Second Exam: 15%
%10 : st Lab1
%10 :Final lab
%15 :Project
%35 :Final Exam

Attendance Policy

Regular attendance is required at all class and Lab meetings.

Scholastic Dishonesty Policy

Scholastic dishonesty will not be tolerated and will be prosecuted to the fullest extent. You are expected to have read and understood this issue, published by the Registrar's Office, for information about procedures and about what constitutes scholastic dishonesty.

2. Music Appreciation (HUM/MUSC 2573)

Music Appreciation (MUSC/HUM-2573I)

Fall 2002 MWF (Noon 12:50 p) 3 Credit Hours

Rogers State University Health Sciences 131 or KRSC-TV

Professor: Dr. Hugh Foley Office Phone: 918-343-7566 e-mail: hfoley@rsu.edu

REQUIRED TEXT AND CD OR CASSETTE:

Willoughby, David. *The World of Music*. 4th ed. Boston: McGraw-Hill, 1999.

The book should be packaged with either a cassette or cd with the same title as the book.

COURSE DESCRIPTION AND OBJECTIVES: Music Appreciation seeks to inform the student about the indigenous beginnings of North American music, to explain 20th Century music in terms of the origins of the popular styles of blues, jazz, country, rock, and pop, and survey the Western "Classical" tradition from the middle ages to the 20th century. The class concludes with lectures on the significant music produced by Oklahomans. **Goals** for the class range from raising awareness of the various elements, origins, and developments in music, as well as enhancing critical thinking skills of evaluating music and then putting a cognitive critique in writing. By the end of the course, the student should have the background for appreciating the major genres of music with which they are most likely to come into contact in the United States.

COURSE REQUIREMENTS:

1. Listening Journal: students will have listening assignments in which they will be required to listen to the music provided on the student cd/cassettes and respond to the music critically. Students will use the terminology we have developed over the course of the semester in their critiques.
2. Each student will write two essays. **The first essay** will be a 750 word essay about a musical experience the student will attend. The student will describe the music in detail with regard to instrumentation, setting and staging, number of performers on what instruments, style of music played, musical arrangement, ponse. Students should incorporate as many musical terms as possible from Part 1 of *The World of Music* and from "Appendix A" on pages 345-354. The first essay is due **Monday, October 7th**. The **second essay** should be a 750 word essay taken from an interview with a

musician, or someone involved in music. The interview may be a transcription of a recorded interview, or written in prose from notes taken during the interview.

of birth

experience, whether or not family members played music, what started the person in music, what styles of music they prefer, what they do contemporarily in music, and how they feel about music being produced today. The second essay is due **Monday, November 18**. If students wish for the essay to become part of a permanent archive of oral history about music in Oklahoma, please provide a separate sheet of paper that authorizes the interview to be used for educational purposes. See me for further details. Students may opt to switch the order of these essays but must adhere to the due dates.

3. Each student will take a mid-term and final exam. The mid-term a take home exam and the final must be take

Bartlesville, or Pryor. The exams will cover material discussed in class from the

mid-term will be distributed October 14th and is due back no later than October 25th. The final must be taken by December 20th in an RSU testing center.

COURSE OUTLINE:

Week 1 Chapter 1 Introduction to The World of Music

Week 2 Chapters 2 & 3 Listening and Understanding: The Nature and Elements of Music

Week 3 Chapter 8 Music of the Americas

Week 4 Chapters 4 & 5 Folk Music, the Blues, and American Religious Music

Week 5 Chapter 7 Country and Western Music

Week 6 Chapter 6 Jazz in America

Week 7 Chapter 7 American Popular Music (Pre-20th Century Music and Tin Pan Alley)

Week 8 Chapter 7 The Rock Era

Week 9 MID-TERM EXAM

Week 10 Chapter 10 Music to 1600

Week 11 Chapter 11 Music of the Baroque (1600-1750)

Week 12 Chapter 12 Music of the Classical Period (1750-1820)

Week 13 Chapter 13 Music of the Romantic Period (19th Century)

Week 14 Chapters 14&15 Music of the 20th Century I & II

Week 15 Music of Oklahoma

Week 16 Review for Final Exam

GRADING PROCEDURES:

Mid-term Exam: 25%

Final Exam: 25%

Essay 1 and 2 Average: 25%

Listening Journal: 25%

A=90-100 B=80-89 C=70-79 D=60-69 F=Below 60

Absentee Policy: I do not have an absentee policy. Since this some of you will watch this class at home, taking attendance will be difficult. However, I am not responsible from withdrawing you from the class should you decide to drop the course. The student is responsible for withdrawing themselves from the class.

Academic Misconduct: Students are expected to follow university policies Student Code of Responsibilities and Conduct. In Accordance with Title 12 of The Student Code (page 11), instances of alleged academic misconduct will follow the policies and procedures as described in Title 12. As a general rule, Faculty at Rogers State University have the responsibility of enforcing the academic code. Therefore, if academic misconduct is suspected I will submit a letter of alleged academic misconduct to the Office of Student Affairs.

as your own. To avoid plagiarism when you arguments, designs, words, ideas, project, etc., you must make it clear that the work originated with someone else by citing the source. Suspected plagiarism cases enact a nasty administrative process that no one will enjoy. Please avoid this hassle and do your own work.

Non-Academic Misconduct: In order to a maintain an effective learning environment, students are expected to fully comply with The Student Code.

Disruptive behavior will not be tolerated, especially in the broadcast classroom where the public may hear or view disruptive behavior. Respectful conduct is an absolute must.

ADA Statement: If you have special physical, psychiatric, or learning disabilities, please let me know as soon as possible so that your learning needs may be met. You will need to provide documentation of your disability to the Office of Student Affairs in Meyer Hall.

Course Description: Study of costs and cost relationships, job order and process cost control, management reports, and special decisions. It introduces the field of managerial accounting, illustrating the internal uses of accounting information and its usefulness to managers, mainly for decision.

Material to be covered

- Chapter 1 The Scope of Management Accounting
- Chapter 2 Cost Terms, Concepts, and Classification
- Chapter 3 Accounting for Materials and Labor
- Chapter 4 Accounting for Overhead Expenditure
- Chapter 5 Cost Accounting Methods: Overview
- Chapter 6 Job Order and Batch Costing
- Chapter 7 Contract Costing
- Chapter 8 Process Costing
- Chapter 9 Joint Products and By-Products
- Chapter 10 Service Costing
- Chapter 11 Activity-based Costing
- Chapter 12 Cost-Volume-Profit Analysis
- Chapter 13 Controlling Costs
- Chapter 14 Budgets
- Chapter 15 Investment Appraisal Techniques
- Chapter 16 Performance Appraisal

PALESTINE POLYTECHNIC INSTITUTE
Department of Administrative Sciences & Information Systems

COURSE OUTLINE

MANAGERIAL ACCOUNTING

MIS 271

Instructor

Moh'd F. Hassouneh
Room # 404

Text Book

Introduction to Cost & Management Accounting
Prepared by: Moh'd Hassouneh

Class hours

Sunday	09:30 - 10:45	Room 207
Tuesday	09:30 - 10:45	Room 207

Course Description Study of costs and cost relationships, job order and process cost control, management reports, and special decisions. It introduces students to managerial accounting illustrating the internal uses of accounting information and its usefulness to managers, mainly for decision.

Material to be covered

- Chapter 1 The Scope of Management Accounting
- Chapter 2 Cost Terms, Concepts, and Classification
- Chapter 3 Accounting for Materials and Labor
- Chapter 4 Accounting for Overhead Expenditure
- Chapter 5 Cost Accounting Methods: Overview
- Chapter 6 Job Order and Batch Costing
- Chapter 7 Contract Costing
- Chapter 8 Process Costing
- Chapter 9 Joints Products and By-Products
- Chapter 10 Service Costing
- Chapter 11 Activity-based Costing
- Chapter 12 Cost-Volume-Profit Analysis
- Chapter 13 Controlling Costs
- Chapter 14 Budgets
- Chapter 15 Investment Appraisal Techniques
- Chapter 16 Performance Appraisal

Grading:

First Hour Exam	25%
Second Hour Exam	25%
Participation and Assignments	05%
Quizzes	05%
Final Exam	40%
Total	100%

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Palestine Polytechnic University
College Of Administrative Science & Informatics
Course Outline

Course : Human Resource Management
Instructor : Dr. Marwan S. Jaloud

(HRM)

MIS 385

Office: C 107

Course Description :

The course seeks to provide students with the principles of modern HRM, beside the information on the new strategic approach of managing personnel. Topics include: philosophy, role, and policy of personnel in organization design, job analysis, recruitment strategy, selection, interviewing and decision making, training, work design, management development, incentives and reward, job evaluation, performance , appraisal .

Topics to be covered :

- 1- Human resource management : an overview
- 2- Job analysis
- 3- Human resource planning
- 4- Recruitment and selection
- 5- Training
- 6- Management development
- 7- Pay plans
- 8- Job evaluation
- 9- Performance appraisal

Text books :

-Gary Dessler, personnel \ Human Resource Management , fifth Edition, by prentice. Hall, Inc. 1991.

-Hand outs .

- De Cenzo\ Robbins, Human Resource Management, fifth Edition, by John Wiley & sons, Inc .1996

Performance Evaluation:

First exam	15%
Second exam	20%
Participation & Quizzes	15%
Final exam	50 %
Total	100%

With my best wishes ...

```

<HTML>
<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=window
s-1252">
<TITLE>exam1</TITLE>
</HEAD>
<BODY>
<%
If IsObject(Session("project_conn")) Then
    Set conn = Session("project_conn")
Else
    Set conn = Server.CreateObject("ADODB.Connection")
    conn.open "project","admin",""
    Set Session("??project_conn") = conn
End If
%>
<%
If IsObject(Session("exam1_rs")) Then
    Set rs = Session("exam1_rs")
Else
    sql = "SELECT [COURSE-EXAM].EXAM, [COURSE-EXAM].MATERIAL, [CO
URSE-EXAM].EDATE, [COURSE-EXAM].GRADE, [COURSE-EXAM].TYPE FROM [
COURSE-EXAM] WHERE ((([COURSE-EXAM].SYLNO)=" & Request.QueryStri
ng("[syllabus number]") & ") AND (([COURSE-EXAM].EXNO)=" & Reques
t.QueryString("[exam number]") & ")) "
    Set rs = Server.CreateObject("ADODB.Recordset")
    rs.Open sql, conn, 3, 3
End If
%>
<%
On Error Resume Next
rs.MoveFirst
do while Not rs.eof
%>
<TR VALIGN=TOP>
<form><center>
<TD BORDERCOLOR=#c0c0c0 ><FONT style=FONT-SIZE:18pt FACE="Verdana
" COLOR=#336699><%=Server.HTMLEncode(rs.Fields("EXAM").Value)%><b
><BR>
</b></FONT><b></TD>
</b>
</form></center>
<blockquote>
<font color="#000080">
-----
-----
-----</font><p>
<span style="text-transform: capitalize"><b>
<font face="Verdana" color="#000080" size="2">exam material: </
font></b></span>
</p>
<form>

```



```

add code
<html>

<HEAD>

<body bgcolor="#FFFFFF">

<form method="POST" action="confirm.asp" name="globe">

<div align="center">
  <center>
    <table border="1" cellpadding="2" cellspacing="3" width="460"
  >
    <tr>
      <td width="436" colspan="2">
        <p align="center"><font face="Verdana" size="1" color="
#0000FF">Please
          give us your demographics.</font></td>
        </tr>
        <tr>
          <td width="122"><font face="Verdana" size="2">instructor
number</font><font face="Verdana" size="2" color="#FF0000">*</fon
t></td>
          <td width="314"><font face="Verdana" size="2"><input type
="text" name="number" size="30"></font></td>
        </tr>
        <tr>
          <td width="122"><font face="Verdana" size="2">instructor
name<font color="#FF0000">*</font></font></td>
          <td width="314"><font face="Verdana" size="2">
            <input type="text" name="name" size="30"></font></td>
        </tr>

        <tr>
          <td width="122"><font face="Verdana" size="2">email</font
><font color="#FF0000" face="Verdana" size="2">*</font></td>
          <td width="314"><font face="Verdana" size="2">
            <input type="text" name="email" size="30"></font></td>
        </tr>

        <tr>
          <td width="122"> <font face="Verdana" size="2">your web s
ite</font><font color="#FF0000" face="Verdana" size="2">*</font><
/td>
          <td width="314"><font face="Verdana" size="2">
            <input type="text" name="yourweb" size="30"></font></td>
        </tr>

        <tr>
          <td width="122"> <font face="Verdana" size="2">user name<
/td>
          <td width="314"><font face="Verdana" size="2">*</font></td>
        </tr>
  </table>
</div>
</form>

```

```

<td width="314"><font face="Verdana" size="2">
<input type="text" name="username" size="30"></font></td>
</tr>

<tr>
<td width="436" colspan="2">
<p align="center"><font face="Verdana" size="1" color="
#0000FF">Please
provide your Password</font></td>
</tr>

<tr>
<td width="122"><font face="Verdana" size="2">Password<fo
nt color="#FF0000">*</font></font></td>
<td width="314"><font face="Verdana" size="2"><input type
="password" name="password" size="20"></font></td>
</tr>
<tr>
<td width="122">confirm password <font color="#FF0000">*<
/font></td>
<td width="314"><input type="password" name="confirmpassw
ord" size="20"></td>
</tr>
<tr>
<td width="122"><font face="Verdana" size="1"><a href="lo
gin.asp">Member Login Here!</a></font></td>
<td width="314"><font face="Verdana" size="2"><input type
="submit" value="Signup Now! " name="B1"><input type="reset" valu
e="Cancel" name="Cancel"></font></td>
</tr>
</table>
</center>
</div>
<p></p>
</form>

</body>
</html>

```

```
<html>
<head>
<meta http-equiv="Content-Language" content="en-us">
<meta name="GENERATOR" content="Microsoft FrontPage 5.0">
<meta http-equiv="Content-Type" content="text/html; charset=windo
ws-1256">
<title>New Page 1</title>
</head>

<body bgcolor="fixed" bgcolor="#FFFFFF">

<%
Function ChkString(str)
    str = replace(str, "<", "&lt;", 1, -1, 1)
    str = replace(str, ">", "&gt;", 1, -1, 1)

    ChkString = str
End Function

number = trim(ChkString(Request.form("number")))
name = trim(ChkString(Request.form("name")))
email = trim(ChkString(Request.form("email")))
yourweb= trim(ChkString(Request.form("yourweb")))
username= trim(ChkString(Request.form("username")))
Password = trim(ChkString(Request.form("password")))
confirmpassword = trim(ChkString(request.form("confirmpassword")))
)

if number = "" then
Response.Write "Please Enter Your number<br>"
else

if name = "" then
Response.Write "Please Enter Your name<br>"
else

if email = "" then
Response.Write "Please Enter valid email<br>"
else

if yourweb = "" then
Response.Write "Please Enter your web site<br>"
else

if username = "" then
Response.Write "Please Enter user name<br>"
else

if password = "" then
```

```

Response.Write "Please Enter password<br>"
else
if password <> confirmpassword then
Response.Write "Password does not match"
else
%>

```

```

&nbsp;<P align=center><i><b>Please verify if all your info you su
ppplied is correct! If
not hit back and correct.</b></i></P>

```

```

<div align="center">
  <center>
    <table border="1" cellpadding="2" cellspacing="3" width="400"
  >
    <tr>
      <td width="118" ><font face="Arial">number:</font></td>
      <td width="259" ><%=number%>&nbsp;</td>
    </tr>
    <tr>
      <td width="118" ><font face="Arial">name:</font></td>
      <td width="259" ><%=name%>&nbsp;</td>
    </tr>
    <tr>
      <td width="118" ><font face="Arial">E-mail:</font></td>
      <td width="259" ><%=email%>&nbsp;</td>
    </tr>
    <tr>
      <td width="118" ><font face="Arial">your web site:</font>
</td>
      <td width="259" ><%=yourweb%>&nbsp;</td>
    </tr>
    <tr>
      <td width="118" ><font face="Arial">user name:</font></td>
      <td width="259" ><%=username%>&nbsp;</td>
    </tr>
    <tr>
      <td width="118" ><font face="Arial">password :</font></td>
      <td width="259" ><%=password%>&nbsp;</td>
    </tr>
    <tr>
      <td width="118" ></td>
    </tr>

```

```

<td width="259" >

<form action=addnewmember.asp method="post">
<font face="Arial">
<input type=hidden name="number" value="<%=number%>">
<input type=hidden name="name" value="<%=name%>">
<input type=hidden name="email" value="<%=email%>">
<input type=hidden name="yourweb" value="<%=yourweb%>">
<input type=hidden name="username" value="<%=username%
>">
<input type=hidden name="password" value="<%=password
%>">

</font>
<p>
<font face="Arial">
<input type=submit value="Confirm"> </font> </p>
</form>

</td>
</tr>

</table>
</center>
</div>

</body>

<%
end if
end if
end if
end if
end if
end if
end if
end if
%>
</html>

```

```

<%
Function ChkString(str)
    str = replace(str, "<", "&lt;", 1, -1, 1)
    str = replace(str, ">", "&gt;", 1, -1, 1)
    ChkString = str
End Function

number = trim(ChkString(Request.form("number")))
name = trim(ChkString(Request.form("name")))
email = trim(ChkString(Request.form("email")))
yourweb= trim(ChkString(Request.form("yourweb")))
username= trim(ChkString(Request.form("username")))
Password = trim(ChkString(Request.form("password")))

set my_conn= Server.CreateObject("ADODB.Connection")
my_conn.Open "DSN=project","admin",""
set rscheck = my_conn.Execute ("Select ino from instructor where
ino=' ' & number & ' '")

if rscheck.eof then
been_here_before="No"
end if

RScheck.close
set rs = server.CreateObject("adodb.recordset")
if been_here_before="No" then
sql ="select * from instructor"

RS.open Sql,my_conn,1,3
RS.AddNew
RS("ino")=number
RS("iName")=name
RS("email")=email
RS("instructor-web-site")=yourweb
RS("id")=username
RS("pass")= password
RS("adate")= now()

RS.update
    %>
<body bgcolor="#FFFFFF">
    <center><font face="Verdana" size="2">Your Account
t Has Been Processed!! <br>
    Your User ID is <b><font color="#0000FF">: <%=use

```

```

name%> , </font> </b>and your password is
<b><font color="#0000FF">: <%=password%> </font></b>
<p><a href="operation1.asp"><font face="Verdana" size="2">Go to o
peration Page</font>
</a>

</p></center>

<%
else
%>

<center><font face="Verdana" size="2">You
r have signup already with this email<br>
Please signup with another your email add
ress.

</font>
<p><font face="Verdana" size="2"><a href="signup.
asp">Sign up Again! </a> </font> </center>

<%
end if

rs.close
my_conn.Close
set my_conn = nothing
set rs = nothing
%>
</BODY>
</HTML>

```

```

<HTML xmlns:v="urn:schemas-microsoft-com:vml" xmlns:o="urn:schema
s-microsoft-com:office:office" xmlns="http://www.w3.org/TR/REC-ht
ml40">
<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=window
s-1252">
<link rel="File-List" href="editc1_files/filelist.xml">

<TITLE>Query2</TITLE>
</HEAD>
<!--[if !mso]>
<style>
v\:*          { behavior: url(#default#VML) }
o\:*          { behavior: url(#default#VML) }
.shape       { behavior: url(#default#VML) }
</style>
<![endif]--><!--[if gte mso 9]>
<xml><o:shapedefaults v:ext="edit" spidmax="1027"/>
</xml><![endif]-->
<p align="center">&nbsp;</p>
<p align="center"><b><em>
<span style="text-transform: capitalize"><font face="Haettenschwe
iler" size="7" color="#000080">
courses list<BODY link="#006699" vlink="#006699" alink="#006699">
</font><font face="Haettenschweiler" color="#000080">
<p align="center">&nbsp;</p>

```

```

<p align="center"><i><font face="Verdana" color="#000080" size="2
">select the course
number you want to update </font></i></p>

```

```

</font><font face="Haettenschweiler" size="7" color="#000080">

```

```

<%
session.Timeout = 200
%>
<%
If IsObject(Session("project_conn")) Then
    Set conn = Session("project_conn")
Else
    Set conn = Server.CreateObject("ADODB.Connection")
    conn.open "DSN=project"

```


</table>

</div>

<%

rs.MoveNext

loop%>

</TBODY>

</TFOOT></TFOOT>

</TABLE>

</p>

</form>

</BODY>

</HTML>

```
<html xmlns:v="urn:schemas-microsoft-com:vml" xmlns:o="urn:schema
s-microsoft-com:office:office" xmlns="http://www.w3.org/TR/REC-ht
ml40">
```

```
<%
session.Timeout = 200
%>
```

```
<%
If IsObject(Session("project_conn")) Then
    Set conn = Session("project_conn")
Else
    Set conn = Server.CreateObject("ADODB.Connection")
    conn.open "DSN=project"
    Set Session("??project_conn") = conn
End If
```

```
<%
If IsObject(Session("gggg_rs")) Then
    Set rs = Session("gggg_rs")
Else
    sql = "SELECT COURSE.CNO, COURSE.CNAME, COURSE.DESCRPTION, C
OURSE.PREREQUISITE, COURSE.CREDITHOUR FROM COURSE WHERE (((COUR
SE.CNO)=" & Request.QueryString("[course number]") & ")) "
    Set rs = Server.CreateObject("ADODB.Recordset")
    rs.Open sql, conn, 3, 3
End If
```

```
<%
val=Server.HTMLEncode(rs.Fields("CNO").Value)
%>
<link rel="File-List" href="editc2_files/filelist.xml">
<!--[if !mso]>
<style>
v\:*          { behavior: url(#default#VML) }
o\:*          { behavior: url(#default#VML) }
.shape       { behavior: url(#default#VML) }
</style>
<![endif]-->
```

```
<!--[if gte mso 9]>
<xml><o:shapedefaults v:ext="edit" spidmax="1027"/>
</xml><![endif]-->
```

```
<body>
```

```
<p align="center">&nbsp;</p>
```

```
<p align="center"><b><em><font face="Haettenschweiler" size="7" c
olor="#000080">
```

```
<span style="text-transform: capitalize">course detail </span></f
ont></em></b>
```

```
</p>
```

```
<p align="center"><!--[if gte vml 1]><v:line id="_x0000_s1025"
style='position:absolute;left:0;text-align:left;top:0;z-index:1'
from="50.25pt,90.75pt"
```

```
to="507pt,90.75pt" strokecolor="#369" strokeweight="6pt">
```

```
<v:stroke linestyle="thickBetweenThin"/>
```

```
</v:line><![endif]--><![if !vml]><span style='mso-ignore:vglayout
;position:
```

```
absolute;z-index:1;left:63px;top:117px;width:617px;height:8px'><i
mg width=617
```

```
height=8 src="editc2_files/image001.gif" v:shapes="_x0000_s1025">
```

```
</span><![endif]></p>
```

```
<p>&nbsp;</p>
```

```
<blockquote>
```

```
<blockquote>
```

```
<p><span style="text-transform: capitalize"><b>
```

```
<font face="Verdana" color="#000080">course detail:</font></b></s
pan><P>
```

```
<span style="text-transform: capitalize"><b><font face="Verdana">
```

```
<TR VALIGN=TOP>
```

```
<TD BORDERCOLOR=#c0c0c0 ALIGN=RIGHT></font><FONT FACE="Verdana"
```

```
COLOR=#000000>&nbsp;&nbsp;&nbsp;</FONT>
```

```
<FONT FACE="Verdana" COLOR=#006699>Course number:</FONT><FONT FAC
```

```
E="Verdana" COLOR=#0099CC>&nbsp;&nbsp;&nbsp;<%=Server.HTMLEncode(rs.Fi
```

```
elds("CNO").Value)%></FONT><FONT FACE="Verdana" COLOR=#000000><BR
```

```
></FONT>
```

```
<font face="Verdana"></TD>
```

```
<TD BORDERCOLOR=#c0c0c0 ></font><FONT FACE="Verdana" COLOR=#00000
```

```
0>&nbsp;&nbsp;&nbsp;</FONT>
```

```
<FONT FACE="Verdana" COLOR=#006699>Course name:</FONT><FONT FACE=
```

```
"Verdana" COLOR=#0099CC>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<%=Server.
```

```
HTMLEncode(rs.Fields("CNAME").Value)%></FONT><FONT FACE="Verdana"
```

```
COLOR=#000000><BR></FONT>
```

```
<font face="Verdana"></TD>
```

```
<TD BORDERCOLOR=#c0c0c0 ></font></b><FONT FACE="Verdana" COLOR=#0
```



```

<html dir="rtl">

<body>
<%
session.Timeout = 200
%>

<%
If IsObject(Session("project_conn")) Then
    Set conn = Session("project_conn")
Else
    Set conn = Server.CreateObject("ADODB.Connection")
    conn.open "DSN=project"
    Set Session("??project_conn") = conn
End If
%>
<%

v= Request.form("co")
If IsObject(Session("gggg_rs")) Then
    Set rs = Session("gggg_rs")
Else
    sql = "SELECT * FROM COURSE WHERE (((COURSE.CNO)=" & v & ")
    )
    Set rs = Server.CreateObject("ADODB.Recordset")
    rs.Open sql, conn, 3, 3
End If
%>

    <%val=Server.HTMLEncode(rs.Fields("CNO").Value)%>

<p align="center" dir="ltr"><i><b>
<font face="Haettenschweiler" size="7" color="#000080">edit cours
e</font></b></i></p>

<form method="POST" action="editc4.asp" onsubmit="return FrontPag
e_Form1_Validator(this)" language="JavaScript" name="FrontPage_Fo
rm1" >

<div align="center" dir="ltr">
    <center dir="ltr">
        <table border="1" cellpadding="2" cellspacing="3" width="460"
dir="ltr">
            <tr>
                <td width="436" colspan="2" dir="ltr">
                    <p align="center" dir="ltr">
                        <font face="Verdana" size="1" color="#0000FF">please en
ter course data&nbsp;
                    </font></td>
                </tr>
            <tr>

```

```

    <td width="122" dir="ltr">
      <p dir="ltr"><font face="Verdana" size="2">course number<
/font><font face="Verdana" size="2" color="#FF0000">*</font></td>
      <td width="314" dir="ltr">
        <p dir="ltr"><font face="Verdana" size="2"><%=Server.HTM
LEncode(rs.Fields("CNO").Value)%></font>
        </td>
      </tr>
    <tr>
      <td width="122" dir="ltr">
        <p dir="ltr"><font face="Verdana" size="2">course name<fo
nt color="#FF0000">*</font></td>
      <td width="314" dir="ltr">
        <p dir="ltr"><font face="Verdana" size="2"></font>
        <input type="text" name="cn" size="30" value=<%=Server.HT
MLEncode(rs.Fields("Cname").Value)%>></td>
      </tr>

    <tr>
      <td width="122" dir="ltr">
        <p dir="ltr"><font face="Verdana" size="2">description</f
ont><font color="#FF0000" face="Verdana" size="2">*</font></td>
      <td width="314" dir="ltr">
        <p dir="ltr">
          <textarea rows="5" name="des" cols="36" > <%=Server.HTM
LEncode(rs.Fields("description").Value)%></textarea></td>
      </tr>

    <tr>
      <td width="122" dir="ltr">
        <p dir="ltr"><font face="Verdana" size="2">Prerequisites<
font color="#FF0000">*</font></font></td>
      <td width="314" dir="ltr">
        <p dir="ltr"><font face="Verdana" size="2">
        <input type="text" name="p" size="30" value=<%=Server.HTM
LEncode(rs.Fields("PREREQUISITE").Value)%>></font></td>
      </tr>

    <tr>
      <td width="122" dir="ltr">
        <p dir="ltr"><font face="Verdana" size="2">Credit hour<fo
nt color="#FF0000">*</font></font></td>
      <td width="314" dir="ltr">
        <p dir="ltr"><font face="Verdana" size="2">
        <input type="text" name="ch" size="30" value="<%=Server.
HTMLEncode(rs.Fields("CREDITHOUR").Value)%>"></td>
      </tr>

    <tr>
      <td width="122" dir="ltr">&nbsp;</td>
      <td width="314" dir="ltr">

```

```
<p dir="ltr"><font face="Verdana" size="2">
  <input type="hidden" name="co" value= <%=val%>>
  <input type="submit" value="add to database" name="B1"><i
nput type="reset" value="Cancel" name="Cancel"></font></td>
  </tr>
</table>
</center>
</div>
<p dir="ltr"></p>
</form>

<p align="center" dir="ltr">&nbsp;</p>
</body>
</html>
```

```

<%
Function ChkString(str)
    str = replace(str, "<", "&lt;", 1, -1, 1)
    str = replace(str, ">", "&gt;", 1, -1, 1)
    ChkString = str
End Function

co = trim(ChkString(Request.form("co")))
cn = trim(ChkString(Request.form("cn")))
des = trim(ChkString(Request.form("des")))
p = trim(ChkString(Request.form("p")))
ch = trim(ChkString(Request.form("ch")))

if co = "" then
Response.Write "Please Enter course number<br>"
else

if cn = "" then
Response.Write "Please Enter course name<br>"
else

if des = "" then
Response.Write "Please course description<br>"
else

if p = "" then
Response.Write "Please course Prerequisites<br>"
else

if ch = "" then
Response.Write "Please Enter course cridet hours<br>"
else

%>

 <P align=center><i><b>Please verify if all &nbsp;&nbsp; info you s
upplied is correct! If
not hit back and correct.</b></i></P>

<div align="center">
    <center>
    <table border="1" cellpadding="2" cellspacing="3" width="426"
>

    <tr>
        <td width="140" ><font face="Verdana">course number:</fon
t></td>

```

```

        <td width="263" ><%=co%>&nbsp;</td>
</tr>
<tr>
    <td width="140" ><font face="Verdana">course name:</font>
</td>
    <td width="263" ><%=cn%>&nbsp;</td>
</tr>
<tr>
    <td width="140" ><font face="Verdana">description:</font>
</td>
    <td width="263" ><%=des%>&nbsp;</td>
</tr>
<tr>
    <td width="140" ><font face="Verdana">Prerequisites:</fon
t></td>
    <td width="263" ><%=p%>&nbsp;</td>
</tr>

    <tr>
    <td width="140" ><font face="Verdana">Credit hour:</font>
</td>
    <td width="263" ><%=ch%>&nbsp;</td>
</tr>

    <tr>
    <td width="140" ></td>
    <td width="263" >

<form action=editc5.asp method="post">
<font face="Arial">
<input type=hidden name="co" value="<%=co%>">
<input type=hidden name="cn" value="<%=cn%>">
    <input type=hidden name="des" value="<%=des%>">
        <input type=hidden name="p" value="<%=p%>">
            <input type=hidden name="ch" value="<%=ch%>">

    </font>
<p>
<font face="Arial">
<input type=submit value="Confirm"> </font> </p>
</form>

</td>
</tr>
</table>
</center>

```



```

<%
Function ChkString(str)

    str = replace(str, "<", "&lt;", 1, -1, 1)
    str = replace(str, ">", "&gt;", 1, -1, 1)

    ChkString = str
End Function

co = trim(ChkString(Request.form("co")))
cn = trim(ChkString(Request.form("cn")))
des = trim(ChkString(Request.form("des")))
p= trim(ChkString(Request.form("p")))
ch= trim(ChkString(Request.form("ch")))

set my_conn= Server.CreateObject("ADODB.Connection")
my_Conn.Open "DSN=project","admin",""
set rscheck = my_conn.Execute ("Select cno from course where cno=
'" & co & "'")

if rscheck.eof then
been_here_before="No"
end if

RScheck.close
set rs = server.CreateObject("adodb.recordset")
sql ="select * from course"
RS.open Sql,my_conn,1,3
RS.update
RS.addnew
'RS("cno")=co
RS("cname")=cn
RS("DESCRIPTION")=des
RS("PREREQUISITE")=p
RS("CREDITHOUR")=ch

'response.write Request.form("co")

    %>
<body bgcolor="#FFFFFF">

    <center><font face="Verdana" size="2">Your course
Has Been Processed!! <br>
    Your course number is <b><font color="#0000FF">:
<%=co%> ,</font> </b>and your course name is
    <b><font color="#0000FF">: <%=cn%> </font></b>
    </font>

```

```
<p><a href="operation2.asp"><font face="verdana" size="2">Go to  
operation page</font> </a>  
</p></center>
```

<%

```
rs.close  
my_conn.Close  
set my_conn = nothing  
set rs = nothing  
>
```

