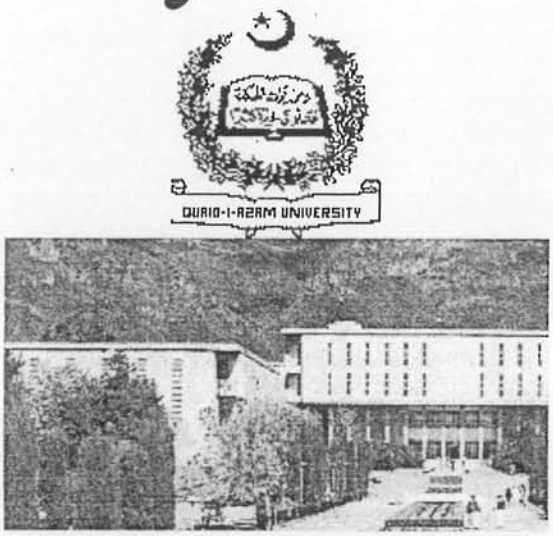
Library Information System



Developed By: Amir Ali
Supervised By:Anees-ur-Rehman
PGD (IT) Final
Computer Centre
Q.A.U Islamabad

Contents

Library

Information System

Developed By:
Amir Ali

Supervised BY:
Anees-ur-Rehman

CERTIFICATE

Report Title: Library information system

This dissertation Amir Ali is accepted in its present form by the of Computer centre, as fulfilling the requirement for the PGD (it) by the Computer centre Q.A.U university. After approved by sir Anees-ur-Rehman (project supervisor) of Session 2005– 2006.

Supervised by: **Prepared** by:

Anees-ur-Rehman **Amir Ali**

Computer centre Q.A.U Islamabad

Supervisor Signature:

Submitted On:

DEDICATED

To

- **My Mother (Late)**

  The Woman behind our (family) success

- **My Second eldest sister**

  Who have given me her support during my studies and paid the role of my mother after her death.

PROJECT BRIEF

PROJECT TITLE: Library information system

ORGANIZATION: computer centre Q.A.U Islamabad

UNDERTAKEN BY: Amir Ali

SUPERVISED BY: Anees-ur-Rehman

SESSION: 2005-2006

SOFTWARE TOOLS: C++, Micro soft word

OPERATING SYSTEM: XP

Acknowledgment

Praise to Allah, Lord of the worlds, who enabled me to complete the project and fulfill the required functionalities. I am thankful to deputy direvtor sir Nazim-ud-Din course coordinater sir Javed Hussain and sir Munawer Hussain taiwana for providing us facilities for the hardware equipment and software needed for the project. .They provided us every opportunity to work in a healthy atmosphere and this was all not possible without the guidance and moral support of my super wiser sir Anees-ur-Rehman. He was always there whenever we needed his help and ideas and helped in my personal difficulty. I am really thankful to him and all staff of computer centre for arranging lab during the course of the project. In the end I would also like to thank Director.

Library Information System

Chapter 1

System Requirement

This chapter sums up the activities about requirement of system.

Project Main points

- Book status
- Members status
- Update database
- Book search
- Member search
- Print database
- About Developer
- Exit

1.1 Book status

In book status we have all the record about book i.e I.D number, book name, writer name (author). In book status we have all the record of books how many books there in library. So this module will provide all the information regarding book.

1.2 Member status

In Member status we we have all the record of member's i.e registration number, name, address and email address. In member status we have all the record about how many members of library. This module will provide all the information regarding Library members.

1.3 Update database

In update database we can update our records. We can change version of or add new book as w ell as member .We have two options

- **M = member**
- **B = book**

This module will provide the facility either to add new Book/Member to the system or Edit any existing Book/Member record or delete any book/member from the system. In order to edit or delete any book admin should provide the relevant accession number of the book, system will locate the relevant record and then admin can edit the existing record and system will save the changes made by him.

1.3.1 Book update

In book update we an add a book delete a Book or edit a book

- **A=add a book**
- **D=delete a book**
- **E=edit a book**

This module will provide the facility either to add new Book to the system or Edit any existing

Book record or delete any book from the system. In order to edit or delete any book admin should provide the relevant accession number of the book, system will locate the relevant record and then admin can edit the existing record and system will save the changes made by him.

When more copies of the presently entered books are provided to the library, then the user/librarian clicks the Entry button and chooses the book from the popup list, which contains the list of all the books, which are already entered in the book form. The book no of the same books is the same but a separate ID will be generated automatically without the interaction of the user to the new entry of the book. The save button will confirm the entrance of the book. We can also mention the cupboard in which we have to place the newly entered book. A remarks field is also provided for extraordinary comments about the book.

The interesting and appreciating thing about this project is that both the ISSUE and RETURN of books is controlled by one button. In the same LIBRARY form one can not only enter new books but also perform the issue and return operations. The member and Issue date refers that the selected book is issued. When someone returns the book just make the

Member cell none, the issue date will be automatically erased. This will be clear from the shot on the next page.

When someone returns the book. Then search the book first and select none in the Member. The issue date will automatically disappear. This book is now again ready to be issued to some other person. Because it's Member status is none and issue date is empty, means that the book is now not issued to anybody and ready to be issued to somebody if someone desires.

When the user clicks the BOOKS ENTRY in the MAIN LIBRARY FORM, this form appears. This form displays the complete information about the books. The Create button is used to enter a new book. This form has also strong searching facility. The user can search on any field, thus a versatile searching mechanism is available. The form has also deletion and saving mechanisms. The view button is used to view the books record. The Cancel button cancels the displayed items. First, Next, Previous and Last buttons perform the general functions. Each button has given the short keystrokes to hit them directly from the keyboard without using the mouse. (Alt must be pressed with the specific alphabet in each case).

1.3.2 Member update

In book update we an add a book delete a book or edit a book

- **A=add a book**
- **D=delete a book**
- **E=edit a book**

This module will provide the facility either to add new Member to the system or Edit any existing Member record or delete any member from the system. In order to edit or delete any book admin should provide the relevant I.D number of the member, system will locate the relevant record and then admin can edit the existing record and system will save the changes made by him.

This report will be displayed when the user clicks the Students button in the main report form. This report will display those members only whose designation is student. It will display student no, name, F/Name, class no, technology, session, card no, address and phone no.

This report will be displayed when the user clicks the Teachers button in the main report form. This report will display those members only whose designation is teacher. It will display teacher no, name, F/Name, technology, session, cardio, address and phone no.

This report will be displayed when the user clicks the Staff button in the main report form. This report will display those members only whose designation is staff. It will display staff no, name, F/Name, technology, session, card no, address and phone no.

1.4 Book search

In book search we have all record in our computer so when we have to search a book, we can search it with accession number, title, author name, publisher. We have to press a key from keyboard i.e

- **N= accession number**
- **T= title**
- **A= author**
- **P=publisher**

This module will provide the facility to search a Book in system if it exist otherwise display massage Book not found.

1.1Member search

In member search we have all record in our computer so when we have to search a member; we can search it with id number, name, and email category. We have to press a key from keyboard i.e

- **I= id number**
- **N=name**
- **E=email**
- **C=category**

This module will provide the facility to search a member in system if it exist otherwise display massage Book not found.

1.6 Print database

In print database, if we have a printer then we can print any record of book or member

1.7 About Developer

In this all information about developer.

Exit

In exit we come out form main program.

Library Information system

Chapter 2

SOFTWARE MANAGEMENT

❖ In this chapter we will discuss about management

Version			
Draft	Amir Ali	Initial Draft was created for distribution and review comments.	15-03-2006
Preliminary	Same as above	Second draft incorporating initial review comments, distributed for final review.	20-03-2006
Final	Same as above	First complete draft, which is placed under change control.	30-03-2006
Revision 1	Same as above	Revised draft, according to the change control process and maintained under change control.	28-04-2006
Revision 2	Same as above	Revised draft, according to the change control process and maintained under change control.	15-05-2006

1. Introduction

2.1 Project Overview

In my library system there are so many functionalities about record of books, record of members for e0xample addition of books/member deletion of book/member. We can also edit a record of book/member. We can search a book by its accession number, Author name, Publisher and by title of book. We can also search a member by its I.D, name, Email, and Category .In library information system we can also print any record of book/member if any body required.

Project deliverable

les	Delivery Location	Delivery Method	Quantity	Expected Date
Library Information System	Q.A. University Islamabad	Installing Disk	1	June, 05, 2006
User manual	Q.A. University Islamabad	Book let	1	June, 06, 2006

2.2Evaluation of Software project Management Plan

Draft	Amir Ali	Initial Draft created for distribution and review documents	18-03-2006
Preliminary	Same as above	Second draft incorporating initial review comments, distributed for final review.	25-03-2006
Final	Same as above	First complete draft, which is placed under change control	30-03-2006
Revision 1	Same as above	Revised draft, according to the change control process and maintained under change control	08-04-2006
Revision 2	Same as above	Revised draft, according to the change control process and maintained under change control	15-04-2006

2.4 Reference Materials

1. IEEE Standard 1058.1-1987 for Software Management Plans.
2. Software Engineering by Roger.S.PressMan (4th Edition).

2.5 Definition, Acronyms, or abbreviations

SDK	Software Development Kit
SRS	System Requirement Specification
RSD	Requirement Specification Document
I/O	Input Output
SDS	Software Design Specification

Project Organization

2.1 Process Model

Milestones	Description	Content	Expected Date
Problem Analysis	The problems described by the user for software.	Scope Product Perspective Product functions Constraints Assumptions and Dependencies External Interface Requirements Design constraints Functional	April 15,2006
Detailed Requirement Analysis	Getting Requirements from the customer in		April 18,2006

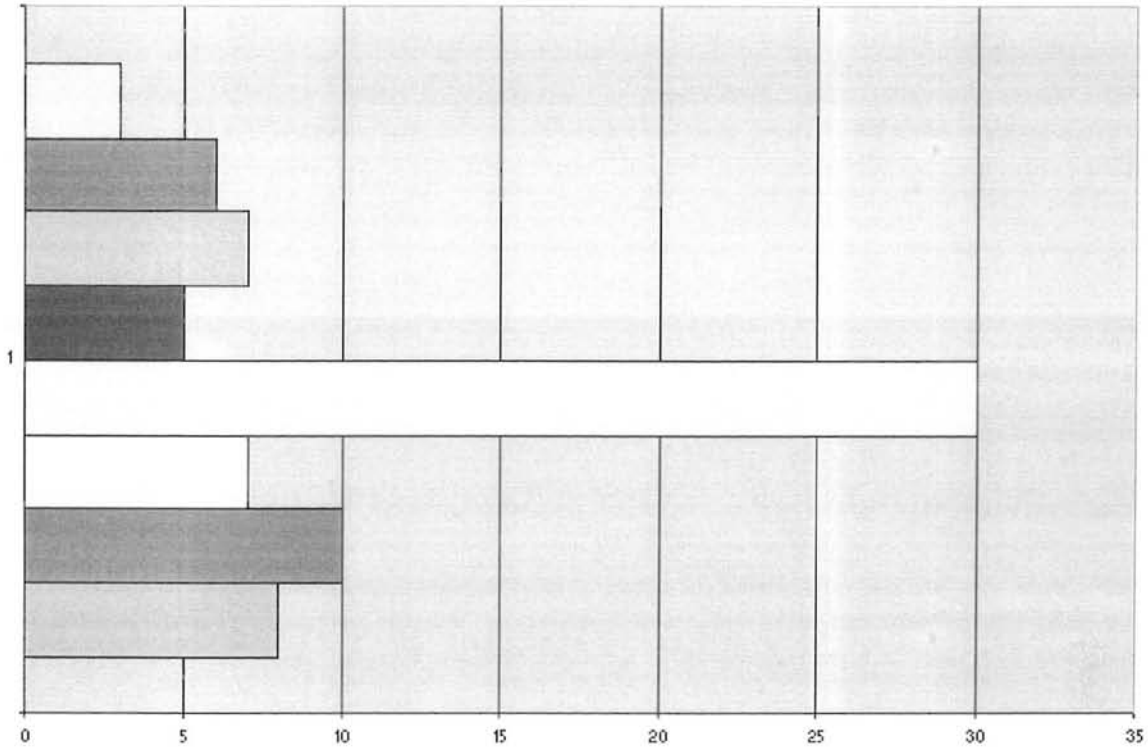
	detail		
Software Requirement Specification	This document will contain the basic requirements of the customer in detail for providing basis for the software development.	Scope Product Perspective Product functions Constraints Assumptions and Dependencies External Interface Requirements Design constraints Functional Requirements Logical Database Requirements	April 23, 2006
Software Design Specification	This document contains the design suitable for development.	Actors Use cases Main Components Functionality of each component Component interaction Component Interaction Model	May 01, 2006
Coding Plan	Tool selection for development of modules		May 06, 2006
Development	Implementation of the design	Different Modules	May 22, 2006

Integration	Integration of different components.	Software components	May 25,2006
Interface	Development of user interfaces	Interface components	May 30,2006
Testing	Black box testing Focuses on the functional requirements of the software. Is a test case design method that uses the control structure of the procedural design to derive test cases?	White Box testing Back Box Testing	June 02, 2006
Final Presentation	Final presentation of the software	Software Document	June 05,2006

Gantt chart:

Activity	Description	Duration	Dependencies
A1	Problem Analysis	3 days	None
A2	Detailed Requirement Analysis	5 days	No of requirements
A3	System Requirement Specification	5 days	Same as above
A4	Software Design Specifications	8 days	No. Of modules
A5	Coding Plan	5 days	Modules interactivity with each other
A6	Development	15 days	
A7	Interface	5 days	
A8	Testing	3 days	No. of users

A9	Integration	3 days	
A10	Final Presentation	3 days	Type of technology, team members



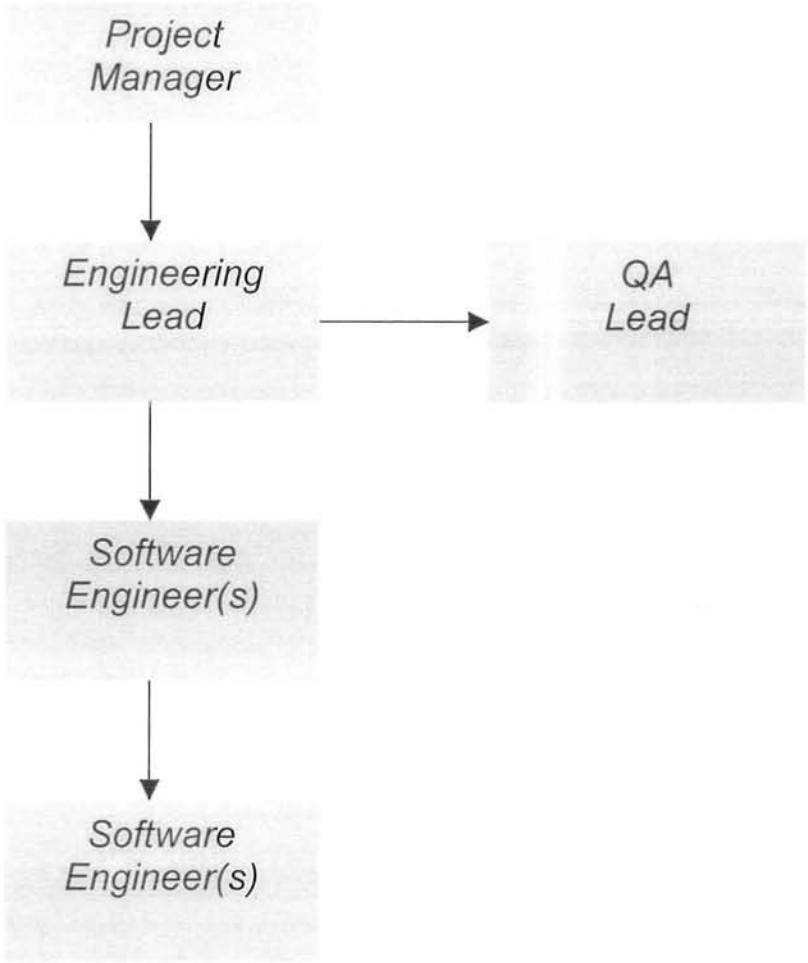
Work Products

Work Product Name	Planned Completion Date	Placed under change control?	Deliverable to customer?	People who must sign off on the Work Product
Software Project Management Plan	18-03-2006	YES	NO	Project Manager, Engineering Lead, QA Lead, Documentation Lead

Change control Plan	27-03-2006	YES	YES	Project Manager, Engineering Lead, QA Lead, Documentation Lead
Top 10 Risk List	26-03-2006	YES	NO	Same as above
Change Proposals	29-03-2006	YES	YES	Same as above
Vision Statement	29-03-2006	YES	NO	Same as above
Software Development Plan, including project cost and schedule estimates	02-04-2006	YES	YES	Same as above
User Interface Style Guide	30-04-2006	YES	YES	Same as above
User Manual / requirements specification	05-04-2006	YES	YES	Same as above
Quality Assurance Plan	21-03-2006	YES	NO	Same as above
Software Architecture	23-03-2006	YES	NO	Same as above
Software Integration Procedure	24-03-2006	YES	NO	Same as above
Staged Delivery Plan	24-03-2006	YES	YES	Same as above
Individual stage plans, including miniature milestone schedules	25-03-2006	YES	YES	Same as above

Coding Standard	25-03-2006	YES	YES	Same as above
Detailed design documents	01-04-2006	YES	YES	Same as above
Software construction plans	06-04-2006	YES	YES	Same as above
Deployment document	25-04-2006	YES	NO	Same as previous
				Same as previous
Release Checklist	31-04-2006	YES	NO	Same as previous
Release Sign-off Form	03-05-2006 05-05-2006			
Document		YES	NO	Same as previous

Project Responsibilities



Responsibility	Persons Responsible
Overall Project Manager	Anees-ur-Rehman
Engineering Manager	Amir Ali
Quality Assurance	Anees-ur-Rehman
End-user Documentation	Amir Ali
Requirements Development	Amir Ali
Software Architecture	Amir Ali
Technical Self-Reviews	Amir Ali

Managerial Process

Management Objectives and Priorities

- Risk Management procedure used is proactive risk strategy.
- Relative priorities are functionality, schedule and resources (budget, time, and technical people).
- PERT charts will be available for quick management assessment.
- MS Project will be used for management purposes.

Assumption, Dependencies and Constrains

Assumptions

- This product will be used only by this enterprise internationally for their Testing Department and by the management staff to track the activities of the product.

Dependencies

- The database depends upon MS-Access, with maximum size (10 MB approx)
- The number of concurrent users can be limited by MS-Access.

Hardware Constraints

- Monitors: 800*600 minimum resolutions at 256 colors minimum.
- Memory: Approximately 64 megabytes.
- I/O: One or two button mouse and standard 101-key keyboard.
- CPU: At least 600 MHz should be on the computer.

Risk Management

- Technology being used is new to the organization.
- Specialized user interface is required for the project.

Monitoring and Controlling Mechanics

- Must work on network because it is a web base application.
- Must be Browser independent
- Must have clear help/error messages.
- Text should be kept minimum to facilitate the user.
- Color choices should be appropriate to accommodate users of all kinds.

Staff Plan

Staffing Factor	Required
Number of Personnel	3
Software Engineer	1 full time, 1 part time
Senior Software Engineer	1
Engineering Lead	1
Quality Assurance Lead	1
Duration of the Project	49 days for the first release
Training Days	1 week

Technical process

Methods, Tool and Techniques

Hardware Environment

- Monitors: 800x600 minimum resolutions at 256 colors minimum.
- Memory: Approximately 64 mega bytes.
- I/O: One or two button mouse standard 101-key keyboard.

Operating System

Microsoft Windows (95, 98, 2000, XP, NT Workstation, NT Server) platform preferred.

Software Tools Methods and Techniques

- Microsoft Word
- C++
- TC-300 IDE
- Rational Rose
- Object Oriented methodology for analysis, design and testing
- Unit Testing
- Integration testing

Software Documentation

Software Development plans, including project cost and schedule estimates.

Project Support Functions

- System Requirement Specification
- Software Design Document

Work Packages, Schedule and Budget

Work Packages

Work Products:

Work Package Identification	Work Packages
W1	Software Project Plan
W2	Change Control Pan
W3	Change Proposals
W4	Vision Statement
W5	Top 10 Risks List
W6	Software Development Plan, including project cost and schedule estimates
W7	User Interface Style Guide
W8	User Manual / Requirement Specification
W9	Quality Assurance Plan
W10	Software Architecture
W11	Software Integration Procedure
W13	Individual stage plans, including milestones.
W14	Coding Standard
W15	Detailed design documents
W16	Software construction plans

W17	Deployment Document
W18	Release Checklist
W19	Release Sign-off log
W20	Software Project Log
W21	Software Project History Document

Resource Requirements:

Resources Required	Duration
Personnel	7 Weeks
Software Engineers	7 Weeks
Senior Software Engineer	7Weeks
Engineering Leader	7 Weeks
Quality Assurance Lead	4 Weeks
Training Leader	1 Week
Computers	2
Software used	8
Budget Required	Rs. 1,20,000

Budget and Resource Allocation:

Project Functions	Budget Allocation
Engineering	Rs. 40000
Quality Assurance	Rs. 30000
Documentation	Rs. 15000
Management	Rs. 35000

Schedule:

Already described in Gantt chart

Library Information System

Chapter3

RISK MANAGEMENT PLAN

This chapter sums up the activities

Of the Risk Management plan, Roles and

Responsibilities, tools and Risk Budget.

1. Introduction (Risk Management System)

Despite much research and progress in the area of Software Project Management, software development projects still are not achieving the target of delivering desired systems on time, within the available financial resources and desired quality. Much of the failure in achieving those targets could be avoided by managers proactive planning for dealing with risk factors rather than waiting for problems to occur and then trying to react on the time of occurrence. Usually this reaction is too little and too late, because by the time the problem is fully recognized, the schedule has already been disturbed, a considerable amount of resources has been utilized, and the product quality has suffered due to introduction of errors. Risk management has been proposed as a solution to for overcoming errors appeared insight into potential problem areas and to identify these problems, address and eliminate them before they can create any problems in the project.

In order to implement a successful risk management program, project managers need tools to help them reduce risks. Risk Management helps project managers in identifying risks in earlier phases of the project cycle, defining risks in earlier phases of the project cycle and defining risk containment actions. The system should support Risk Assessment during the initial phase of the development as well as during project delivery phase.

A good measurement program helps managers:

- Communicate unambiguously throughout the organization.
- Identify and correct technical and management problems by focusing on early discovery of errors.
- Make key tradeoffs by assessing the impact of decision.
- Defend and justify decisions by providing data to explain how issues are prioritized and managed.

Using these as the evaluation criteria a detailed search and evaluation of the Risk Management System available in the industry was made.

2. Purpose

The purpose of this document is to describe how we can perform the job of managing risks for online testing. It identifies risks which may occur in the project, defines roles and responsibilities for participants in the risk management process, the risk management activities that will be carried

out, the schedule and budget for risk management activities and tools and techniques that will be used during this process.

3. Roles and Responsibilities

3.1 Project manager

The project manager will assign a Risk Officer to the project, and identify this individual on the project's organization chart. The Project Manager and other members of the Project Management team will meet every week to review the status of all risk resolving efforts, review the exposure assessments for any new risk items, and redefine the project's Top Ten Risk List.

3.2 Software Quality Assurance involvements

The Project Manager and other members of the project will check about the quality of the project and will assign role for each member of the team for making quality assured software

3. 3 Risk Officer

The Risk officer has the following responsibilities and authorities:

- ✓ Coordinating between risk identification and analysis activities
- ✓ Maintaining the project's risk list
- ✓ Notifying project management of the new risk items discovered
- ✓ Reporting risk resolution status to management
- ✓ The Risk Officer should normally not be the project Manager.

3.4 Project Member Assigned a Risk

The Risk Officer will assign each newly identified risk to any member of the project, who will assess the exposure and probability for the risk factor and report the results of that analysis back to the Risk Officer. Project members who have assigned the responsibilities for performing the steps of the mitigation will report progress about the risk mitigation to the Risk Officer biweekly.

4. Risk Documentation

4.1 Risk List

The risk factors identified and managed for this project will be accumulated in a risk list. The Risk list contains the following items:

1. Personal Risk
2. Unrealistic schedules and budgets.
3. Developing wrong software solution.
4. Developing wrong user interface.
5. Continuing streams of requirement changes.
6. Shortfall in extremely furnished components.
7. Shortfall in externally performed tasks.
8. Real time performance shortfall.
9. Wrong assessment of requirements.

The ten risk items that currently have the highest estimated risk exposure are referred to as the project's Top Ten Risk List.

4.2 Risk Data Items

The following information will be stored for each project risk:

- Risk ID
- Classification
- Description
- Probability
- Impact
- Risk Exposure
- First Indicator
- That risk is becoming a problem
- Mitigation approaches
- Owner
- Date due
- Contingency plan
- Contingency plan trigger

3.3 Closing Risk

A risk item can be considered closed when it meets the following criteria:

The planned lessening actions have been completed and the estimated risk exposure of probability time's impact is less than 2.

5. Activities

	Task	Participants
Risk Identification	State the techniques that will be used to identify risk factors at the beginning of the project and on an on-going basis. This may involve a formal risk assessment workshop, a brainstorming session, and interviews at the beginning of each life cycle phase. Describe any consolidated lists of risk items that will be used to identify candidate risks for this project.	Risk Officer
	The Risk Officer will assign each risk factor to an individual project member, who will estimate the probability the risk could become a problem and the impact this risk on either scale of units of dollars or schedule days, as indicated by the Risk Officer)	Assigned Project Member
	The individual analyzed risk factors are collected, reviewed, and adjusted if necessary. The list of risk Factors are sorted by descending risk exposure.	Risk Officer
	The top ten risks, or those risk factors having an estimated exposure greater than <state exposure. Threshold> are assigned to individual project members for development and execution of a risk mitigation plan.	Risk Officer

	For each assigned risk factor, recommend actions that will reduce either the probability of the risk materializing into a problem, or the severity of the exposure if it does. Return the mitigation plan to the Risk Officer.	Project Members
	The mitigation plans for assigned risk items are collected into a single list. The completed Top Ten Risk List is created and made available for the management.	Risk Officer
	Each individual who is responsible for executing a risk mitigation plan carries out the mitigation activities	Assigned Individual
	Constructive Cost Model (COCOMO)	Risk Officer
	The status and effectiveness of each mitigation action is reported to the Risk Officer every two weeks.	Assigned Individual
	The probability and impact for each risk item is reevaluated and modified if appropriate for risk management.	Risk Officer
	If any new risk items have been identified, they are analyzed as were the items on the original risk list and added to the risk list.	Risk Officer
	The Top Ten Risk List is regenerated based on the updated probability and impact for each remaining risk.	Risk Officer
	Any risk factors for which mitigation actions are not being effectively carried out, or whose risk exposure is rising, may be escalated to an appropriate level of management for visibility and action.	Risk Officer

	If the project will be storing lessons learned about mitigation of specific risks in a database, describe that database and process here and indicate the timing of entering risk-related lessons into the database.	Risk Officer

5.1 Schedules for Risk Management Activities

Risk Identification	*A risk workshop will be held on approximately 06 march 2006.*
Risk List	*The prioritized risk list will be completed and made available to the project team by approximately 06 March 2006*
Risk Management Plan	*The risk management plan, with mitigation, avoidance, or prevention strategies for the top ten risk items, will be completed by Approximately 20 December 2001.*

Risk Track is a risk management tool from Risk Services and Technology. It allows the identification of different kinds of risks that may occur during the different phases of software project development. It also allows the specification of the probability of these risks. The interface is more attractive and easy to use. It does not use the rather outdated spreadsheet like interface which the other risk management and management software use. To start a new project, first it provides a screen for project definition where we can specify the project title, creation date, project description, project manager, project leader, risk, mitigation parent and mitigation. Project ID is generated automatically.

We can also add, modify and delete all possible users, phases, risk class, risk cause, attributes, objectives, risk status, and risk types that can occur during the development of the project. After specifying all these, we can add a new risk through the Add Risk function.

In the add new risk screen you can give the Risk name, risk ID, and select the risk status, class, cause, type and phase. You can also provide a risk statement and its consequences, the risk probability, At Risk Cost, Risk exposure, mitigation exposure, mitigation exposure, cost allocation,

assigned to, date assigned on, assigned by, and action date. You can also add a mitigation using the add mitigation title screen. Here you can give the mitigation title, the effectiveness, risk exposure, cost of mitigation, mitigated exposure, cost allocation, assignee, assigned by and action date while the mitigation ID, creation date, created by, modification date and date assigned on are automatically generated. There is also a mitigation screen where we can see the cost, slip and effect on performance.

1.2 Usability

Risk Track is very easy to use software. It does not use the spreadsheet like interface rather it uses a simple interface where you provide input through input boxes and dropdown lists. It generates easily comprehendible reports, which are also a plus point of the software.

1. 2 Strength

Its strength lies in its ease of use and straight forwardness. It also covers all the phases of Risk Management Process.

Chapter4

SOFTWARE DESIGN SPECIFICATION

In this chapter we will discuss about
faction used in project

USING STRUCTURED APPROACH

1. Introduction

1.1 Product Name	LIS
Library information system	**1.0**

1.2 Developed by

No	Name	Contribution
1.	Amir Ali	Book status
2.	Amir Ali	Members status
3.	Amir Ali	Update database
4.	Amir Ali	Book search
5.	Amir Ali	Member search
6.	Amir Ali	Print database
7.	Amir Ali	About Developer

1.3 Document Generated by	Signature
Amir Ali	

1.4 Date

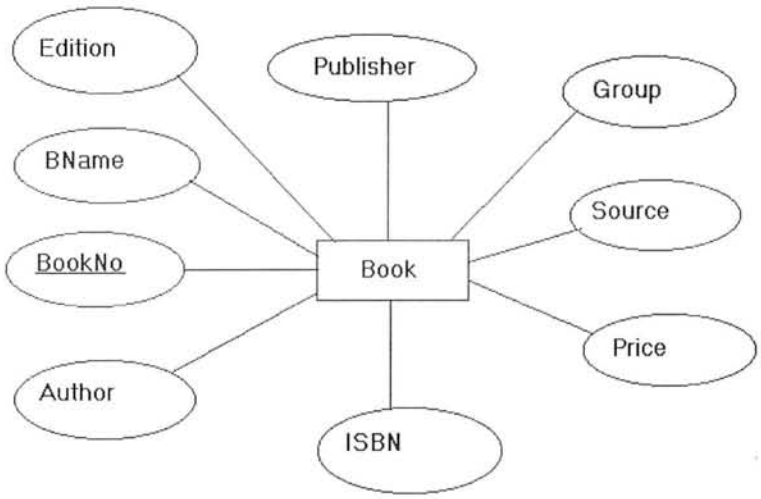
19	June	2006
Day	Month	Year

2. General Descriptions

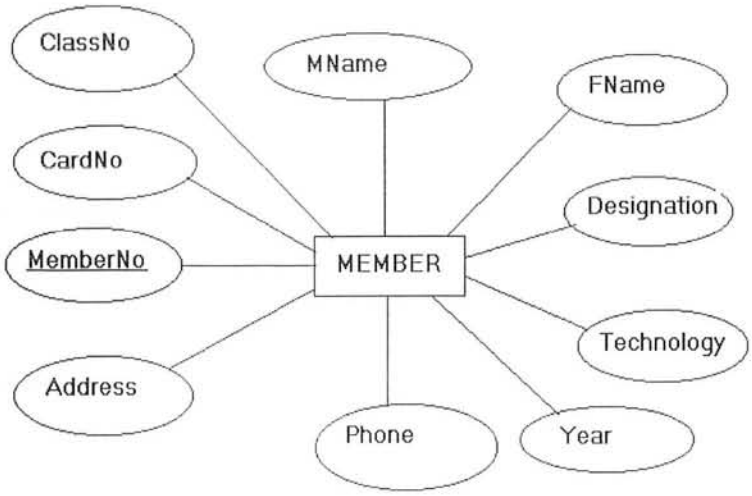
Development Environment	
Programming Language	C++
Development Operating System	Windows XP
Can be use	Window xp,98,2000

DIAGRAMATIC ENTITY REPRESENTATION WITH ATTRIBUTES:

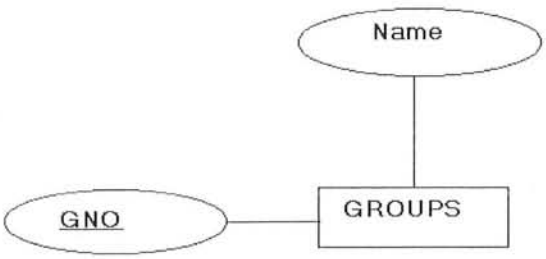
- **BOOK ENTITY**



MEMBER ENTITY



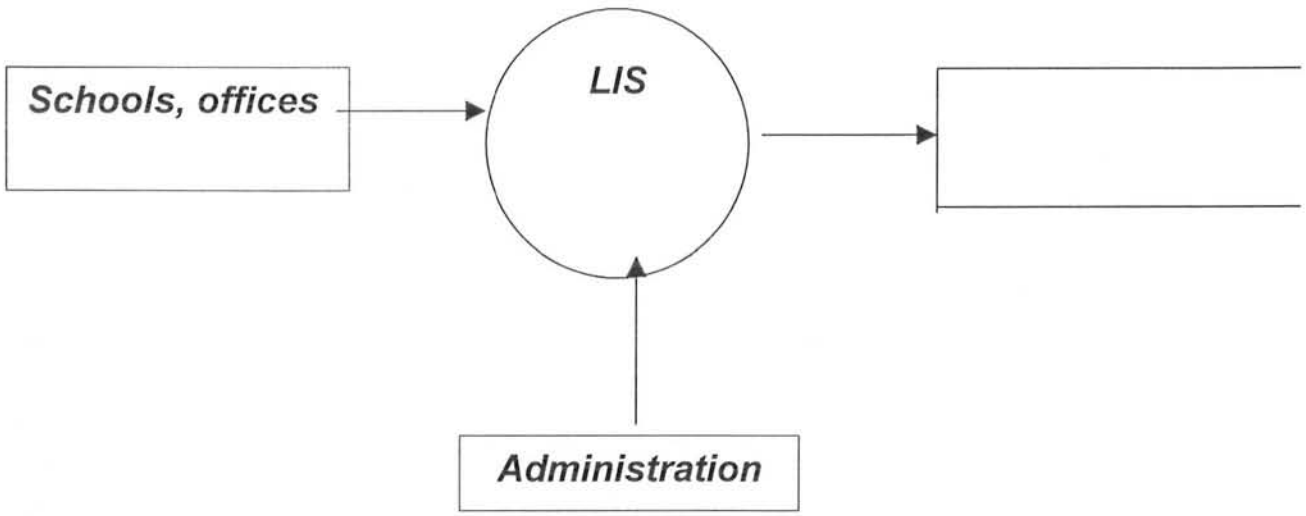
GROUPS ENTITY



- **CUPBOARD ENTITY**

DATAFLOW DIAGRAM (DFD):

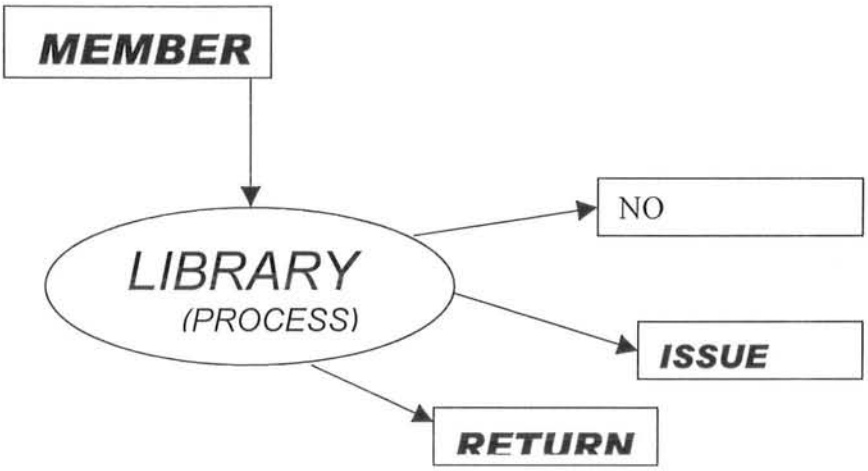
Data flow diagram (DFD) is the diagrammatic representation of the activities and processes in the system. It is used to show the flow of data in an organization. They are easy to understand for new users. They are more precise and unambiguous. They show the flow of data from one activity to another.



DATAFLOW DIAGRAM (MIS)

DATAFLOW DIAGRAM (DFD):

Data flow diagram (DFD) is the diagrammatic representation of the activities and processes in the system. It is used to show the flow of data in an organization. They are easy to understand for new users. They are more precise and unambiguous. They show the flow of data from one activity to another.



DATAFLOW DIAGRAM (GCT)

Built –in- function

void (type)
Empty data type

When used as a function return type, void means that the function does not return a value.

clrscr <CONIO.H>

Clears text mode window

Declaration: void clrscr(void);

Remarks:

clrscr clears the current text window and places the cursor in the upper left-hand corner (at position 1,1).

getch() and getche() <CONIO.H>

getch gets a character from console but does not echo to the screen
getche gets a character from console, and echoes to the screen

Declaration:

int getch(void);
int getche(void);

Remarks:

getch reads a single character directly from the keyboard, without echoing to the screen.

getche reads a single character from the keyboard and echoes it to the current text

window, using direct video or BIOS.

Return Value:

Both functions return the character read from the keyboard.

cputs <CONIO.H>

Writes a string to the text window on the screen

Declaration: int cputs(const char *str);

Remarks:

cputs writes the null-terminated string str to the current text window. It does not append a newline character.

The string is written either directly to screen memory or by way of a BIOS call, depending on the value of directvideo.

cputs does not translate linefeed characters (\n) into carriage-return/linefeed character pairs (\r\n).

gotoxy <CONIO.H>

Positions cursor in text window

Declaration: void gotoxy(int x, int y);

Remarks:
gotoxy moves the cursor to the given position in the current text window.

If the coordinates are invalid, the call to gotoxy is ignored.

Toupper
Translate characters to uppercase

Declaration:
int toupper(int ch);

int _toupper(int ch);

Remarks

toupper is a function that converts an integer ch (in the range EOF to 255) to its uppercase value (A to Z; if it was lowercase, a to z). All others are left unchanged.

_toupper is a macro that does the same conversion as toupper, except that it should be used only when ch is known to be lowercase (a to z).

sizeof (keyword)
Returns the size, in bytes, of the given expression or type (as type size_t).

Syntax:
sizeof <expression>
sizeof (<type>)

windows <conIO.H>

Defines active text-mode window

Declaration: void window(int left, int top, int right, int bottom);

The top left corner of the screen is (1,1).

Remarks:

window defines a text window onscreen. If the coordinates are in any way invalid, the call to window is ignored.

(left, top) is the (x, y) position of the window's upper left corner.

(right, bottom) is the (x, y) position of the window's lower right corner.

The minimum size of the text window is one column by one line.

The default window is full screen, with these coordinates:

80-column mode: (1, 1, 80, 25)
40-column mode: (1, 1, 40, 25)

Some other built in function

isalnum, isalpha, isascii,

iscntrl, isdigit, isgraph

islower, isprint, ispunct <all in CTYPE.H>

isspace, isupper, isxdigit

Character classification macros

Declarations:

int isalnum(int c);

int islower(int c);

int isalpha(int c);

int isprint(int c);

int isascii(int c);

int ispunct(int c);

int iscntrl(int c);

int isspace(int c);

int isdigit(int c);

int isupper(int c);

int isgraph(int c);

int isxdigit(int c);

Remarks:

The is... macros classify ASCII coded integer values by table lookup.

Each macro is a predicate that returns a non-zero value for true and 0 for false.

isascii is defined on all integer values. The other is... macros are defined only when isascii(c) is true or c is EOF.

You can make each macro available as a function by undefining it (with #undef).

Return Value:

The is... macros return a non-zero value on success. For each macro, success is defined as follows:

isalpha: c is a letter (A to Z or a to z)

isascii: the low order byte of c is in the range 0 to 127 (0x00--0x7F)

iscntrl: c is a delete character or ordinary control character
(0x7F or 0x00 to 0x1F)

isdigit: c is a digit (0 to 9)

isgraph: c is a printing character, like isprint, except that a space character is excluded

islower: c is a lowercase letter (a to z)

isprint: c is a printing character (0x20 to 0x7E)

ispunct: c is a punctuation character (iscntrl or isspace)

isspace: c is a space, tab, carriage return, new line, vertical tab, or formfeed (0x09 to 0x0D, 0x20)

isupper: c is an uppercase letter (A to Z)

isxdigit: c is a hexadecimal digit (0 to 9, A to F, a to f

Define Classes & Files

- Lis_main.cpp
- Lis_update.cpp
- Lis_mbst.cpp
- Lis_msrc.cpp
- Lis_book.cpp
- Lis_memb.cpp
- Lis_class.cpp
- Lis_prin.cpp
- Lis_abt.cpp
- Lis_src.cpp
- Lis_bkst.cpp
- Lis_book.h
- Lis_memb.h
- Lis_hdr.h

Library Information System

Chapter 5

SOFTWARE Screen Shots

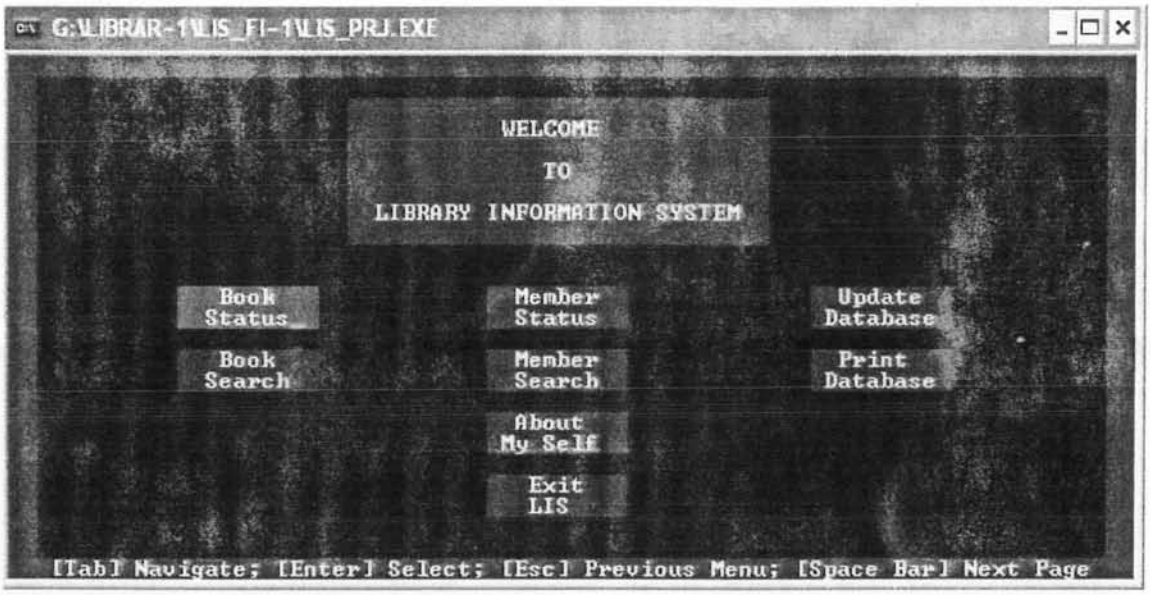
In this chapter we will discuss about

Screen shots of project

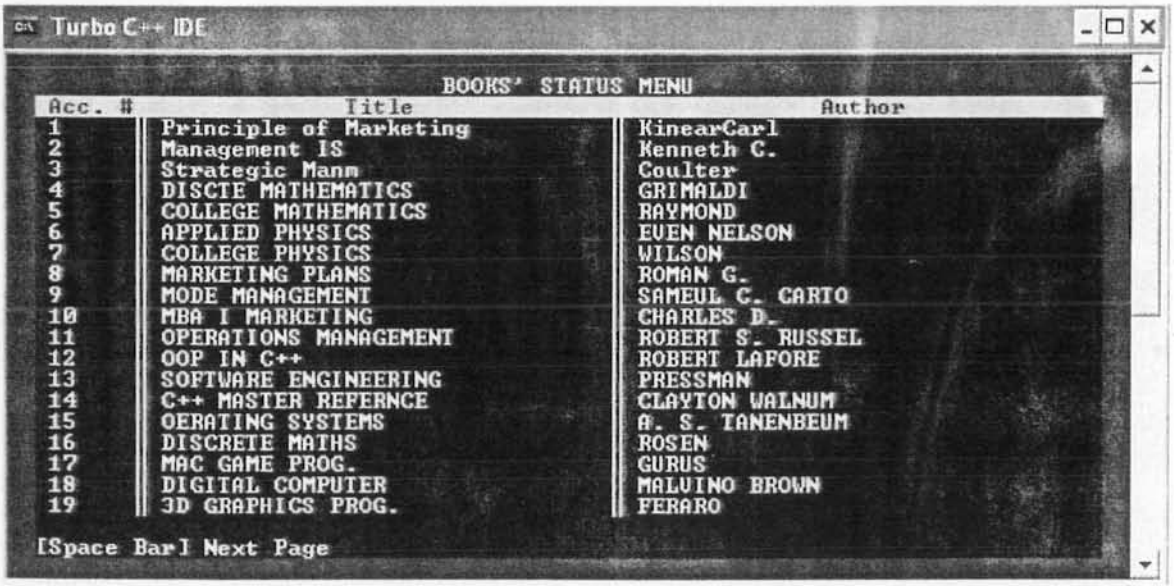
➢ SCREENSHOTS (USER MANNUAL)

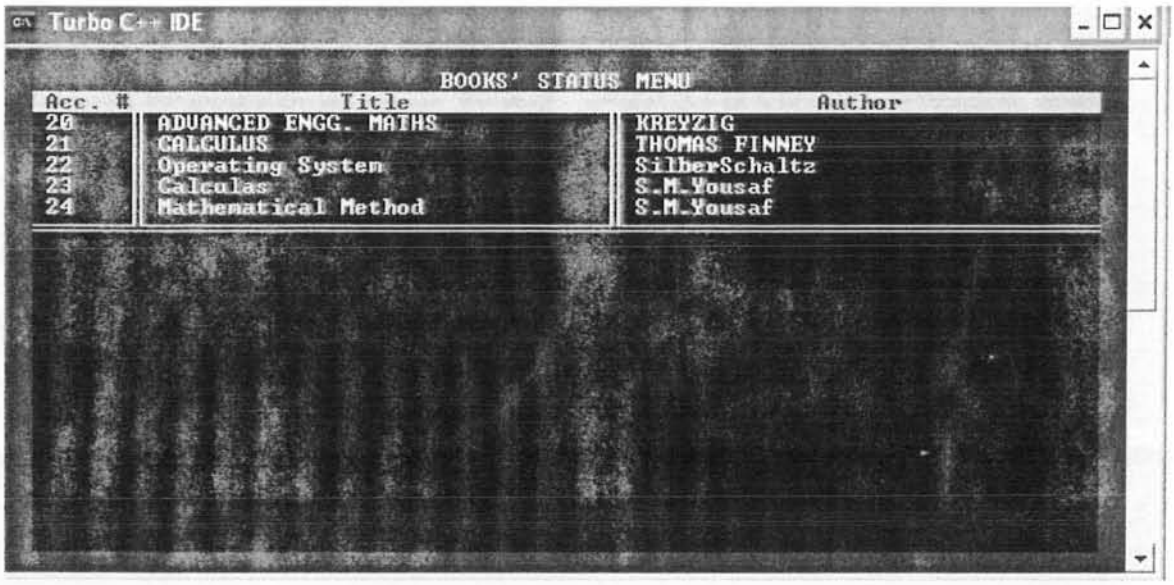
1. **Main Form**
2. **Book Status Menu**
3. **Member Status Menu**
4. **Update database**
    i. **Update database for Books**
    ii. **Update database for Members**
5. **Book search**
6. **Member search**
7. **Print Data**
8. **About Developer**

Main Form:

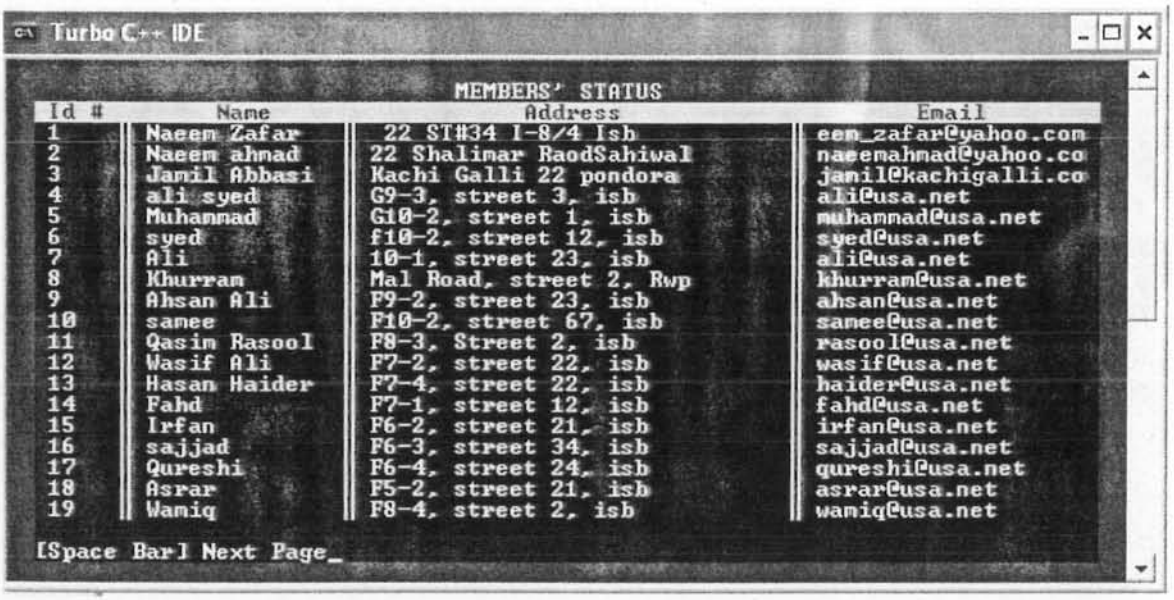


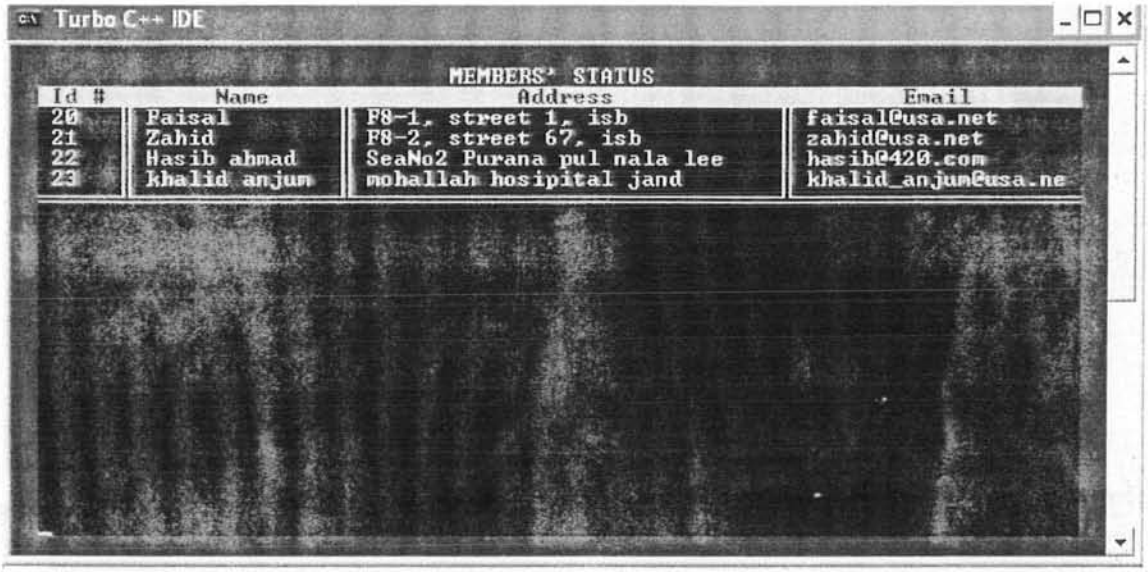
BOOKS' STATUS MENU:



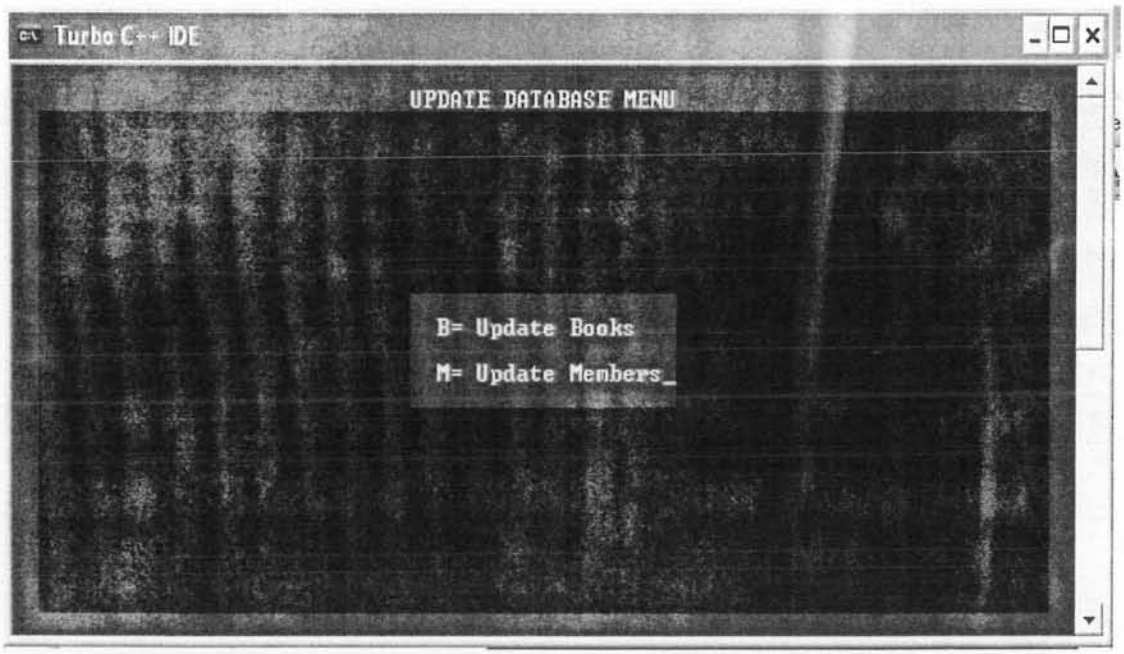


MEMBERS' STATUS MANU:

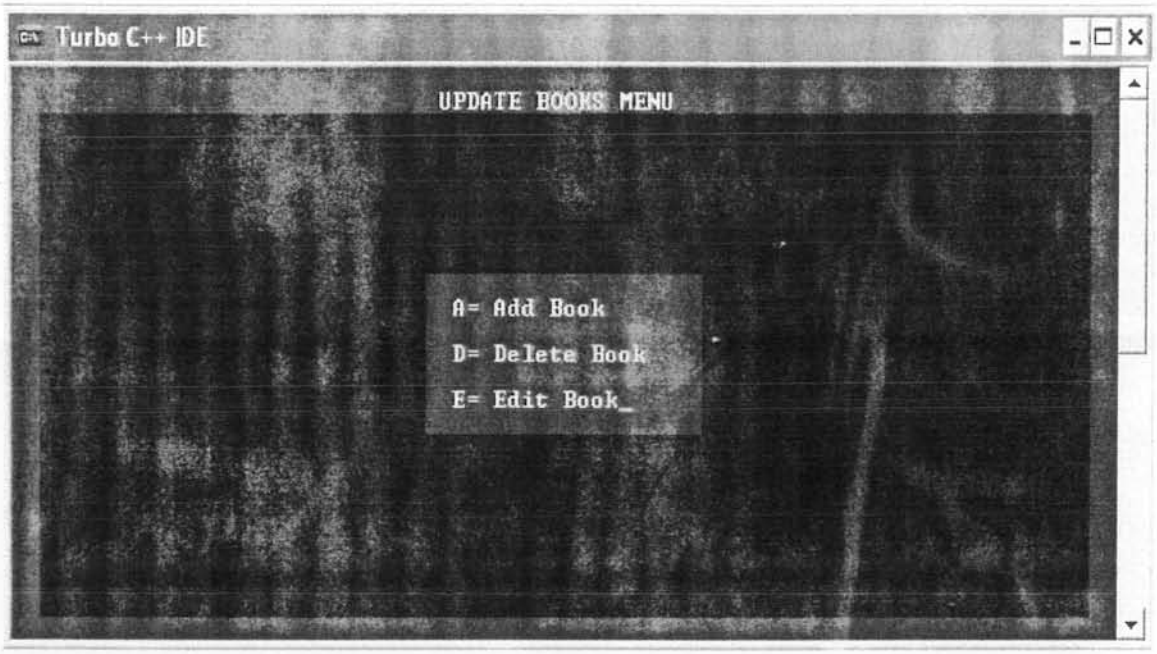




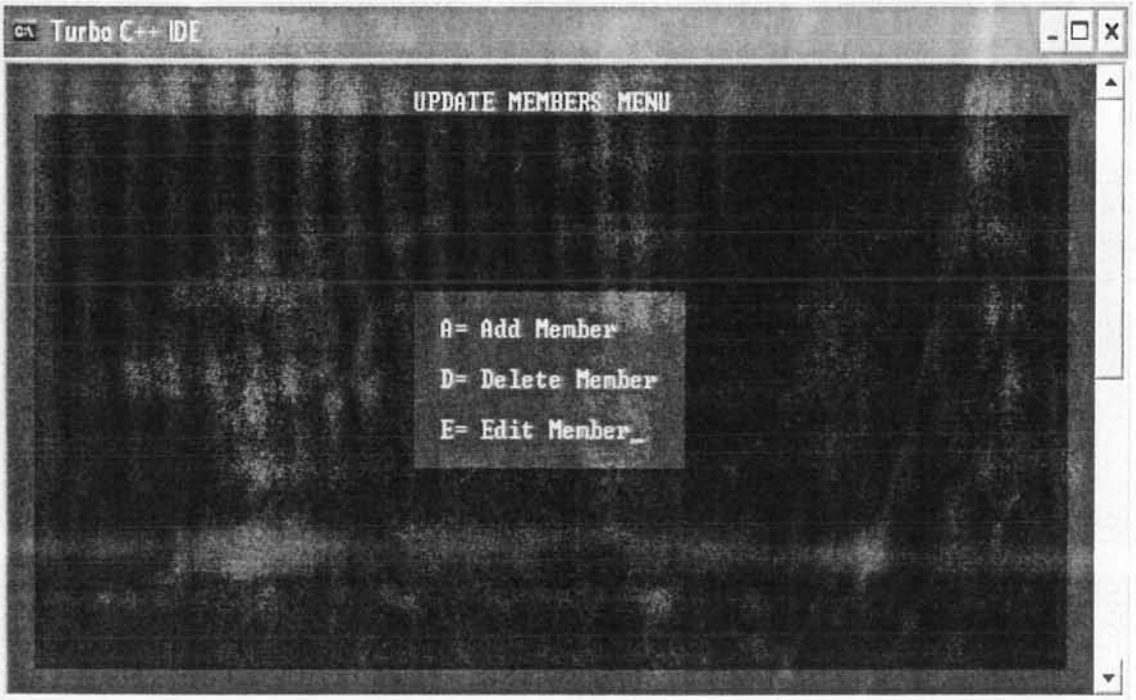
UPDATE DATABASE MENU:



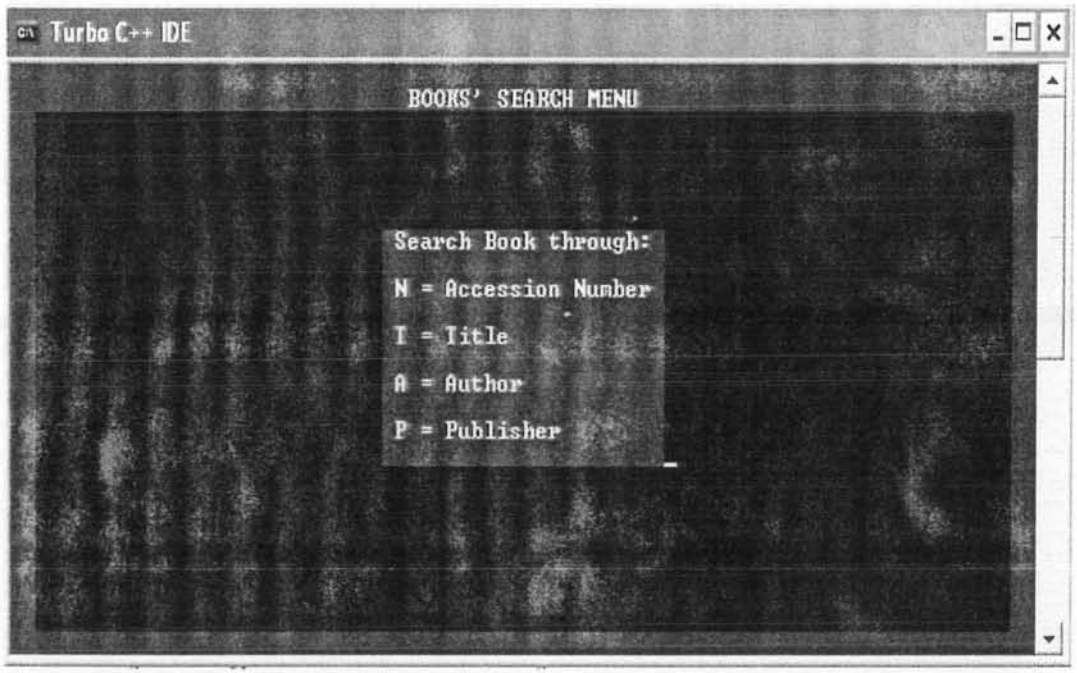
UPDATE BOOKS MENU



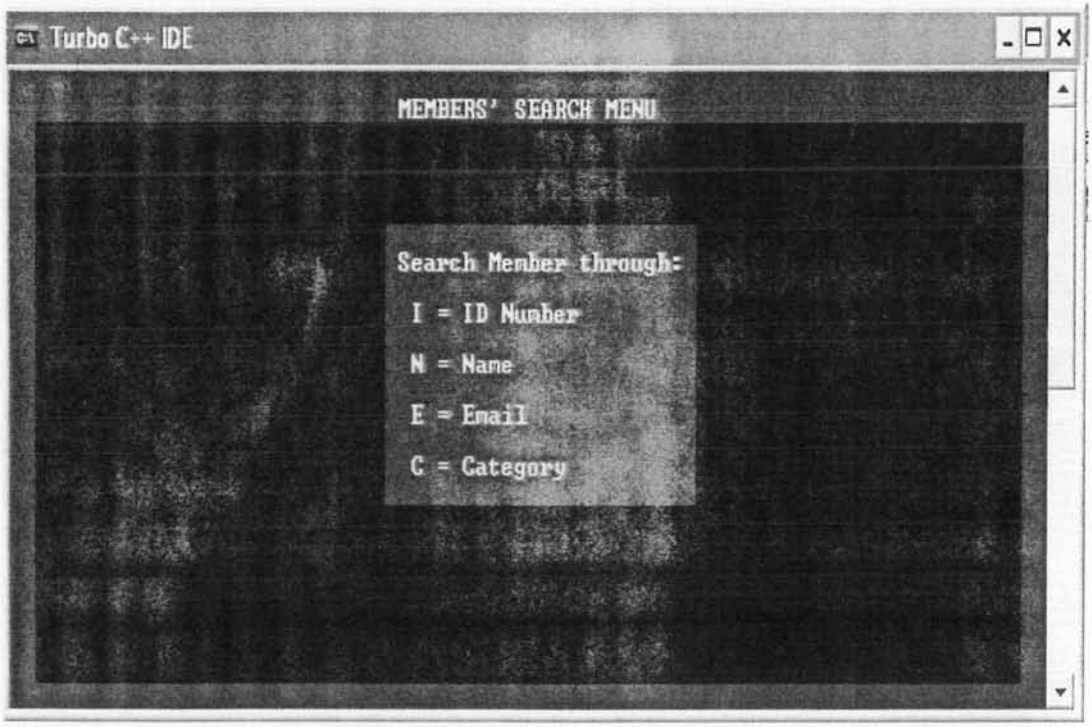
BOOKS' SEARCH MENU



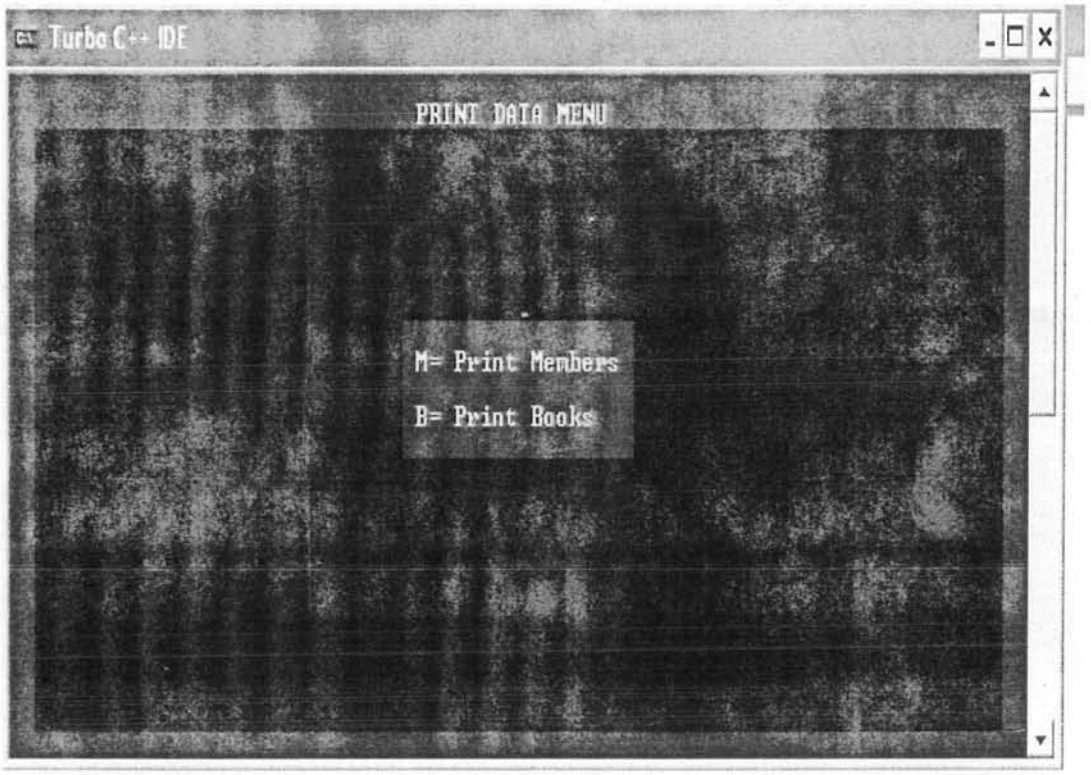
Book search menu



MEMBERS' SEARCH MENU



PRINT DATA MENU



Developer