CUSTOMER RELATION MANAGEMENT

By

V. S. JAGADEESWARAN

Reg. No. 71202702005

Of

Kumaraguru College of Technology

COIMBATORE.

(Affiliated to Anna University)

A PROJECT REPORT

Submitted to the

FACULTY OF SCIENCE AND HUMANITIES

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF SCIENCE

IN

APPLIED SCIENCE – COMPUTER TECHNOLOGY

JUNE, 2004

Kumaraguru College of Technology
Coimbatore – 641023

Department of Computer Science and Engineering

Bonafide Certificate

Certified that this project report titled **CUSTOMER RELATION MANAGEMENT** is the bonafide work of **Mr. V. S. JAGADEESWARAN (Reg No. 71202702005)** who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

GUIDE 16·6·04

HEAD OF THE DEPARTMENT

Submitted for the University Examination Held on __17/6/04__

Internal Guide 17·6·2004

(ap/cse (pg)/kct)

External Guide

CERTIFICATE

This is to certify that **V.S.JAGADEESWARAN (71202702005)** Student of **KUMARAGURU COLLEGE OF TECHNOLOGY,** Coimbatore has successfully completed his project entitled **"CUSTOMER RELATION MANAGEMENT"** in our concern for a period of four months from January 2004 to April 2004 as part of his M.Sc [APPLIED SCIENCE – COMPUTER TECHNOLOGY] Course. His conduct was found good and satisfactory during the project time. Due to our Company's policies source code cannot be provided.

We are fully satisfied with his project work and we wish him all success in their future endeavors.

For **CYBER e-SOFT INDIA**

Jeyakumar John
(Technical Director)

CYBER e-SOFT INDIA
19, Jose Complex, 10th Street,
Coimbatore - 641 012
Tamilnadu INDIA

● *Software*   ● *Consulting*   ● *Training*

ABSTRACT

This project titled **"Customer Relation Management "** is codenamed as **CRM.** CRM is web enabled application software developed for a software company for monitoring their transaction starting from enquiry and up to delivery of the product to customer. It enables the customer to monitor their project work status at any time without any personal contact. This project is developed for **Cyber e-Soft India.**

The major goal of the system is to make new relationship between the customer and the software concern through internet and to make customer-concern interaction more effective. The company should maintain all their day-to-day transaction in web. So the concern has to study their entire present system carefully and redevelop all their existing transaction to keep updating on the web.

The internet through which e-commerce thrives, promises much to business at low cost platform to reach out to global markets and one which defines geographical bounds and time zones to enable business to stay connected with the rest of the world.

This system is the integration of Client/Server architecture and the web technology. It is developed based on the Multi-Tier Client/Server architecture.

ACKNOWLEDGEMENT

I wish to express my sincere thanks to **Dr. K.K.Padmanaban Ph.D.,** principal, Kumaraguru College of Technology, Coimbatore, for permitting me to undertake this project.

I express my deep sense of gratitude to **Dr.S.Thangasamy Ph.D.,** Head of the Department,Compuer Science and Engineering, Kumaraguru College of Technology Coimbatore guide for his valuable advice, guidance, assistance and encouragement for the successful completion of this project.

I am also grateful to my my guide **Mr. Ranganathan Dinesh M.S.,** Assistant Professor, Department of Computer Science and Engineering , Kumaraguru College of Technology,for his valuable guidance and encouragement at every stage of this project work.

I express sincere thanks to **Mr. Jeyakumar John** Technical Director of Cyber e-Soft India for his support at various levels of my project work.

I also express sincere thanks to **Miss.SumathiNayaki. M.Sc.,Project** Manager for spending his time with me and helping me out during the phases of this project. I specially thank **all the staff of Cyber e-Soft India** for their co-operation and valuable advice.

I Most of all thank **My Parents and Friends** for their blessings without which we would not be able to do anything.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

CRM is developed according to the requirements of the concern to monitor all their transactions and to provide current (24 x 7 x 365) and accurate information to their customers.

Two major modules in CRM are

- **Internal Office Automation Application**
- **Business to Customer(B2C) via Internet**

Internal Office Automation Application

Deals with all the internal office processes like maintaining employee details, customer details, project details, project monitoring, order processing, invoice processing, enquiry processing and report generating.

Business to Customer via Internet

Enables the customer to interact with the concern through internet. It deals with processes like sending enquiries, online ordering and project monitoring. It provides online support for the customers.

1.2 ORGANIZATION PROFILE

Cyber e-Soft India is a worldwide provider of information technology services. Established in 1998 and situated in Coimbatore.

Vision

To endure the challenges of evergreen Global demands – *ALWAYS*

Focus

- Web Technologies
- Client - Server Technologies
- System Programming

Services

The company has the expertise to address the entire software development life cycle, starting from Feasibility Study, Requirement Analysis, Design and Development to Implementation, Maintenance and Support of the end products. With expertise of software professionals, in charge of Development, the company has the ability to comprehend the client's needs to their utmost satisfaction. The company uses the best technologies practiced in the industry today.

They are implementing software varying from real-time mission-critical systems to scalable, interoperable and high-availability implementations for the web resulting in effective and extendable solutions.

They deal with advanced web technologies using Sun Products and Microsoft Technologies, such as J2EE, Dot Net Framework, VB, ASP, COM, DCOM, VC++ with Microsoft Foundation Classes.

CHAPTER 2

SYSTEM REQUIREMENTS AND SPECIFICATION

The Software Requirements Specification is a technical specification of requirements for the software product. The goal of software requirements definition is to completely and consistently specify the technical requirements for the software product in a concise and unambiguous manner.

The Software Requirements Specification is based on the System Definition. High-level requirements specified during initial planning are elaborated and made more specific in order to characterize the features that the software product will incorporate. The requirement Specification is primarily concerned with functional and a performance aspect of a software product and emphasis is placed on specifying product characteristics without implying how the product will provide those characteristics.

Desirable properties of a Software Requirements Specification are

- Correct
- Complete
- Consistent
- Unambiguous
- Functional
- Verifiable
- Traceable

2.1 HARDWARE REQUIREMENTS

- Processor   :   Intel Pentium
- Processor speed   :   500 MHz
- Memory   :   64 MB RAM
- Hard Disk   :   10 GB

2.2 SOFTWARE REQUIREMENTS

- Operating System

    ✓ Server   : Windows 2000/NT Server
    ✓ Client   : Windows 9x

- Application Server : Web Logic 6.1

- Database Server   : Microsoft SQL Server 2000

- Languages

    ✓ JAVA
    ✓ EJB 2.0
    ✓ JSP 1.2
    ✓ HTML

- Development Tool   : J2SDK 1.4.2

- Browser   : Internet Explorer

١

2.3 SOFTWARE OVERVIEW

Java

Java is an Object Oriented Programming language developed at Sun Microsystems in June 1995. Java has built-in support for threads, networking and a vast variety of other tools. The amazing thing about Java is that it can be used to create a huge variety of applications and is noted for its 'Write once Run anywhere' characteristic.

Java is simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multithreaded and dynamic language.

The Java architecture consists of Java Virtual Machine or JVM is an abstract computer that runs compiled Java programs. The JVM supports object-oriented programming directly by including instructions for object method invocation. The java compiler generates architecture-independent byte codes.

Benefits of Java

Java allows the user to:

- ✓ Write robust and reliable programs.
- ✓ Build an application on almost any platform, and run that application on any other supported platform without having recompiling your code.
- ✓ Distribute your applications over a network in a secure fashion.

Java has an extensive use of library of routines for copying with TCP/IP protocols like HTTP and FTP. Java application can open and access objects across

net local Universal Resource Locator (URL) with the same ease as accessing the local file system. The Remote Method Invocation method enables communication between distributed objects.

The Java security features are supposed to keep Java program from doing overrunning the run-time stack, corrupting memory outside it's own process, reading or writing files, when invoked through a security conscious class loader like web browser. The multithreading concept helps to produce the better interactive responsiveness and real-time behavior.

The JDBC (Java Database Connectivity) interface allows Java applets, serve lets and applications to access data from popular database management systems.

Enterprise JavaBeans

EJBs are writing-once, run anywhere, middle-tier (server side) components. Enterprise JavaBeans are used to encapsulate business rules. A typical EJB consists of methods that encapsulate business logic. A remote client can invoke these methods, which typically result in the updating of a database.

Although EJBs are straightforward to write, they can take advantage of advanced services as concurrency, persistence and transaction support. EJB shifts the burden of implementing these services from the shoulders of the application developer to those of the container provider.

Benefits of EJB

EJBs encapsulate business logic. EJBs are reusable components, which are very much suitable for development of distributed components. Benefits of EJBs are

- Complete separation of business logic and presentation logic, data logic.
- Develops server side reusable components.
- Automatically handles client sessions.
- Support for ACID transaction.
- Support for management of multiple instances.
- Support for persistence and security.
- Develops powerful Middle-Tier architecture, results in decrease of server process.

Java Server Pages

Java Server Pages provides web developers with a framework to create dynamic content on the server. It is the combination of HTML or XML and java code. It extends the Java Servlet API. It separates the java code and design details completely and the coding style is simple and clear.

Benefits of JSP

- JSPs run on all the main web servers.
- JSPs provide better facilities for separation of page code and template data by means of JavaBeans, Enterprise Java Beans and custom tag libraries.
- JSPs are interpreted only once, to Java byte-code and re-interpreted only when the file is modified.
- JSPs can maintain state on the server between requests.
- It spawns a new thread for each request.
- Runs in a ready-loaded JVM as an extension to the web server.

Web logic

Web logic Server is a multithreaded server that listens on the network for a client request. It establishes a connection with a client, including negotiating details such as protocol, encryption, and authentication. Then it processes the client's requests by executing Java classes on behalf of the client.

On the back-end, Web logic Server can connect to virtually any network accessible service and other services like directory and naming services, messaging systems, and legacy applications that are integrated with Web logic Server applications.

Benefits of Web logic Server

1. Life cycle management
2. Transaction support
3. Persistence
4. Clustering support
5. Security
6. Resource pooling

Database Design

Customer Relation Management System uses Microsoft SQL Server 2000 as its backend. Microsoft extends the performance, reliability, quality, and ease-of-use of Microsoft SQL Server version 7.0. Microsoft SQL Server 2000 includes several new features that make it an excellent database platform for large-scale online transactional processing (OLTP), data warehousing, and e-commerce applications.

Features of SQL Server 2000

Microsoft SQL Server 2000 features include:

- Internet Integration.
- Scalability and Availability.
- Enterprise-Level Database Features.
- Ease of installation, deployment, and use.
- Data warehousing.

Database Architecture

Microsoft SQL Server 2000 data is stored in databases. The data in a database is organized into the logical components visible to users. A database is also physically implemented as two or more files on disk.

When using a database, you work primarily with the logical components such as tables, views, procedures, and users. The physical implementation of files is largely transparent. Typically, only the database administrator needs to work with the physical implementation.

Each instance of SQL Server has four system databases (**master, model, tempdb,** and **msdb**) and one or more user databases. Some organizations have only one user database, containing all the data for their organization. Some organizations have different databases for each group in their organization, and sometimes a database used by a single application. For example, an organization could have one database for sales, one for payroll, one for a document management application, and so on. Sometimes an application uses only one database; other applications may access several databases.

It is not necessary to run multiple copies of the SQL Server database engine to allow multiple users to access the databases on a server. An instance of the SQL Server Standard or Enterprise Edition is capable of handling thousands of users working in multiple databases at the same time. Each instance of SQL Server makes all databases in the instance available to all users that connect to the instance, subject to the defined security permissions.

When connecting to an instance of SQL Server, your connection is associated with a particular database on the server. This database is called the current database. You are usually connected to a database defined as your default database by the system administrator, although you can use connection options in the database APIs to specify another database. You can switch from one database to another using either the Transact-SQL USE database name statement, or an API function that changes your current database context.

SQL Server 2000 allows you to detach databases from an instance of SQL Server, then reattach them to another instance, or even attach the database back to the same instance. If you have a SQL Server database file, you can tell SQL Server when you connect to attach that database file with a specific database name.

Tables can be created in two ways.

1. SQL Query Analyzer.

2. SQL Enterprise Manager

Microsoft SQL Server 2000 SQL Query Analyzer is a graphical tool that allows you to:

- Create queries and other SQL scripts and execute them against SQL Server databases.
- Quickly create commonly used database objects from predefined scripts.
- Quickly copy existing database objects.
- Debug and Execute stored procedures.
- Debug query performance problems.
- Quickly insert, update, or delete rows in a table. (Open Table window)

Microsoft SQL Server Enterprise Manager is the primary administrative tool for Microsoft SQL Server™ 2000 and provides a Microsoft Management Console (MMC)–compliant user interface that allows users to:

- Define groups of servers running SQL Server.
- Register individual servers in a group.
- Configure all SQL Server options for each registered server.
- Create and administer all SQL Server databases, objects, logins, users, and permissions in each registered server.
- Define and execute all SQL Server administrative tasks on each registered server.
- Design and test SQL statements, batches, and scripts interactively by invoking SQL Query Analyzer.

2.4 SPECIFICATION

Platform Specification

Operating System

An Operating System is a collection of computer programs that control, how the computer works. It can also be defined as the software, which acts as a traffic cop, directing requests and information to add from the various devices within a single PC. The OS handles disk requests (read and write) keyboard translations, memory accesses, peripheral accesses and many other functions.

Windows 9x

Windows 9x is a desktop operating system. It runs on Intel/Cyrex/AMD processors. It supports file systems FAT32 (NRWF), VFAT (NRWF).Also integrated with DOS for command line interface and explorer for graphical user interface. It supports multimedia applications and internet applications. Generally it is used as the client operating system.

Windows NT

This is a network operating system. This is based on the Client-Server architecture. Benefits of Windows NT are:

- More intuitive interface.
- Better multitasking and multithreading.
- Clients can be attached to workstations.
- Plug-and-Play technology.
- Higher level of security.
- NTFS-a powerful NT File System.

Windows 2000/NT Server

Microsoft Windows 2000/NT Server 4.0 is a multipurpose server operating system. A multipurpose operating system integrates a variety of network services.

The services it provides are designed to address customer requirements and are designed to address customer requirements and are managed in a single way. It offers a complete end to end solution. It gives the ability to exploit the popularity of compatible third party products and such applications as Active Server Pages, for dynamic content Web pages or SQL Server for specialized database tasks and web servers for web services.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The system study and data collection were done in Cyber e-Soft India, Coimbatore. The present system is developed in Visual Basic 6 and ASP using Microsoft-Access as the database.

Disadvantages of existing system

- The users' categories are not separated.
- All users have same access privileges.
- No support for project monitoring.
- More manual work is done.
- It is not portable.
- The report generated was not all user-friendly.
- It won't support more than a certain number of client interaction concurrently.
- Session handling mechanism was not good.
- No clear separation of business logic, presentation logic and data logic.
- Data transactions are not secure and efficient.
- Modification and Updating is difficult and complex.

3.2 PROPOSED SYSTEM

The proposed system CRM is developed using JAVA, JSP, EJB and Microsoft SQLSERVER 2000 as database.

Advantages of the proposed system

- The users' are separated into three major categories

  - ✓ Administrator
  - ✓ Project Leaders
  - ✓ Junior Programmers

- Each and every user has separate access privileges depending on his category.
- Clients can monitor their project status at any time.
- Clients can send queries at any time.
- Almost all the processes are automated to implement the concept of paperless office.
- This system is portable.
- Report generation process is more efficient, powerful and clear.
- Business logic is completely separated from the presentation and data logic.
- Middle-Tier is made powerful and the processing of database server is decreased.
- The sessions are handled automatically and efficiently.
- This system can support any number of client-interaction concurrently and efficiently.
- Data transactions are made to meet the ACID properties of client/server computing.

3.3 PROTOTYPE

Planning the software development process involves several important considerations. The most important consideration is to define a product life-cycle. The software life cycle encompasses all activities required to define, develop, test, deliver, operate, and maintain a software product. Different models emphasize different aspects of the life-cycle, and no single life-cycle model is appropriate for all software products. Life-cycle models used are the phased model, the cost model, the prototype model and the successive versions model.

The Prototype Life-Cycle Model

Prototype is a mock-up or model of a software product. A prototype incorporates components of the actual product. Typically, a prototype exhibits limited functional capabilities, low reliability, and inefficient performance.

There are several reasons for developing a prototype:

- To illustrate input data formats, messages, reports, and interactive dialogues for the customer. This is used to explain various processing options to the customer and to gain better understanding of the customer's needs.
- To explore technical issues in the proposed product.

The nature and extent of prototyping to be performed on particular software is dependent on the nature of the product. New versions of existing products can most likely be developed using the phased life-cycle model with little or no prototyping. Development of a totally new product will probably involve some prototyping during the planning and analysis phase or the product may be developed by iterating through a series of successive designs and implementations.

CHAPTER 4

SYSTEM DESIGN

4.1 DESIGN PRINCIPLES

The process of design involves "conceiving and planning out in mind" and "making a drawing, pattern, or sketch of". In software design, there are three distinct types of activities: external design, architectural design and detailed design. Architectural design and detailed design are collectively called internal design.

External design of software involves conceiving, planning out, and specifying the externally observable characteristics of a software product. These characteristics include user displays and report formats, external data sources and data sinks, and the functional characteristics, performance requirements and high level process structure for the product. External design begins during the analysis phase and continues into the design phase. Requirements definition is concerned with specifying the external, functional and performance requirements for a system. External design is concerned with refining those requirements and establishing the high level structural view of the system.

Internal design involves conceiving, planning out and specifying the internal structure and processing details of the software product. The goals of internal design are to specify internal structure and processing details, to record design decisions and indicate why certain alternatives and trade-offs were chosen, to elaborate the test plan, and to provide a blueprint for implementation, testing and

maintenance activities. The work products of internal design include a specification of architectural structure, the details of algorithms and data structures, and the test plan.

Design of Customer Relation Management System

CRM is developed for a software concern to maintain their transaction starting from enquiry and up to the delivery of the product. It builds the good customer-concern relationship. It enables the customer to interact with concern more easily and effectively.

This tool automates all the process inside the concern and enables the customer to send enquiry, view reply and monitor their project status at any stage through internet.

Requirements Definition

- The software should develop solution for two major problems.
    - ✓ Tool to automate the internal office activities and data.
    - ✓ Tool to enable the customer to interact with concern through internet.
- To automate the internal office activities the software should maintain a database related to all the data and the workflows inside the concern.
    - ✓ It should store the employee details, customer details, project details, invoice details and the enquiry details.
    - ✓ It should process the invoice and enquiries.
    - ✓ It should generate the reports.
    - ✓ It should define the access level to the users.
- To enable the customer interaction the software should maintain a database related to all the data and transactions of the customers.

✓ It should encapsulate the presentation, business and data logic separately.

✓ It should store the enquiries of the customers.

✓ It should process the customer enquiries.

✓ It should store the project status to give the customers current and accurate information.

✓ It should support multi-client interaction.

✓ It should provide a secured transaction.

4.2 ARCHITECTURAL DESIGN

Architectural design is concerned with refining the conceptual view of the system, identifying internal processing functions, decomposing high-level functions into sub functions, defining internal data streams and data stores and establishing relationships and interconnections among functions, data streams and data stores.

4.2.1 Customer Relation Management System -Module Overview

CRM consists of two major modules.

- Internal Office Automation Application.
- Business to Customer (B2C) via internet.

4.2.2 Internal Office Automation Application

This module is divided into four sub modules.

- New User Login.
- Administrator.
- Project Leader.
- Junior Programmer.

New User Login

New user login interface is designed to enable the user to register **UserID** and **Pass word**. The User ID will be the Employee ID of that user. Before registration the User ID will be checked with the stored data in the database. The user can only register only if User Id is valid.

Administrator

Administrator interface is designed to enable the administrator to do the processes related to his access privileges. This interface is designed as menu-driven GUI. Administrator has full access privilege that is administrator acts as a **Super-User**.

Data's related to administrator are

- Employee details
- Customer details
- Project details
- Product details
- Invoice details
- Enquiry details
- Project status

Data Process related to administrator are

- Invoice process
- Enquiry process
- Report generation
- Customize database

Project Leader

Project leader interface is designed to enable the user to do the processes related to his access privileges. This interface if designed as tapped panels. The user can access the data related to his projects only. The validation is done using the users' User ID.

Data's related are

- Project details assigned to the user.
- Programmers belong to the user's team.

Data Process related are

- Customize project status
- Customize user's team

Junior Programmer

Junior programmer interface is designed to enable the user to do the processes related to his access privileges. This interface if designed as tapped panels. The user can access the data related to his modules only.

Data's related are

- Modules details assigned to the user.
- A programmer belongs to the user's team.

Data Process related are

- Customize module status

4.3 DATA FLOW DIAGRAMS

Data flow diagrams are commonly used during problem analysis and design. A DFD shows the flow of data through a system. It views the system as a function that transforms the inputs into desired outputs. A DFD aims to capture the transformation that takes place within a system into output data so that eventually the output data is produced. The agent that performs the transformation of data from one state to another is called a process (Bubble).The processes are shown by named circles and dataflow are represented by named arrows. A square defines a source or destination of system data. An open rectangle is a data source

DFD - Internal Office Automation Tool

Figure 4.3.1
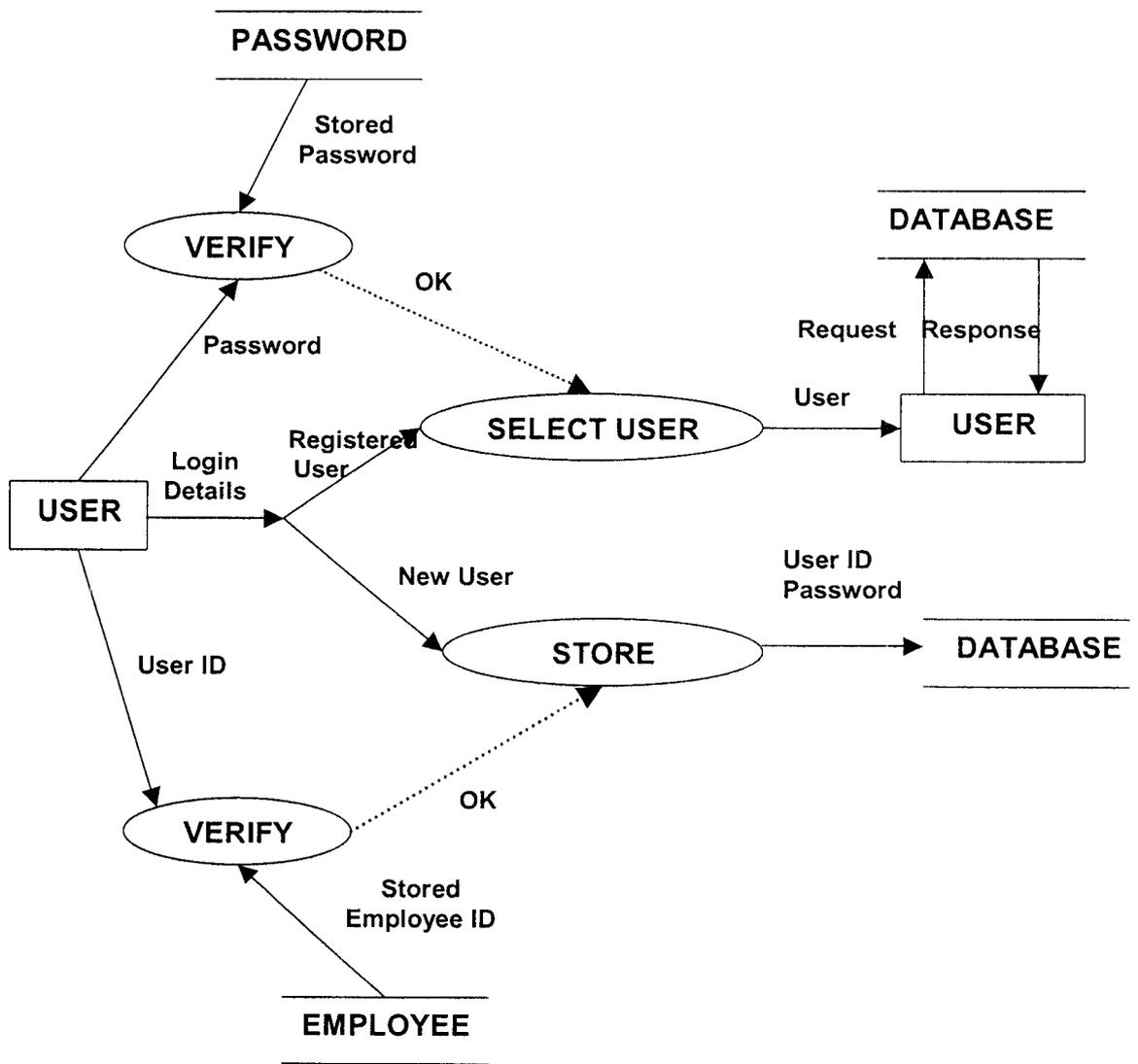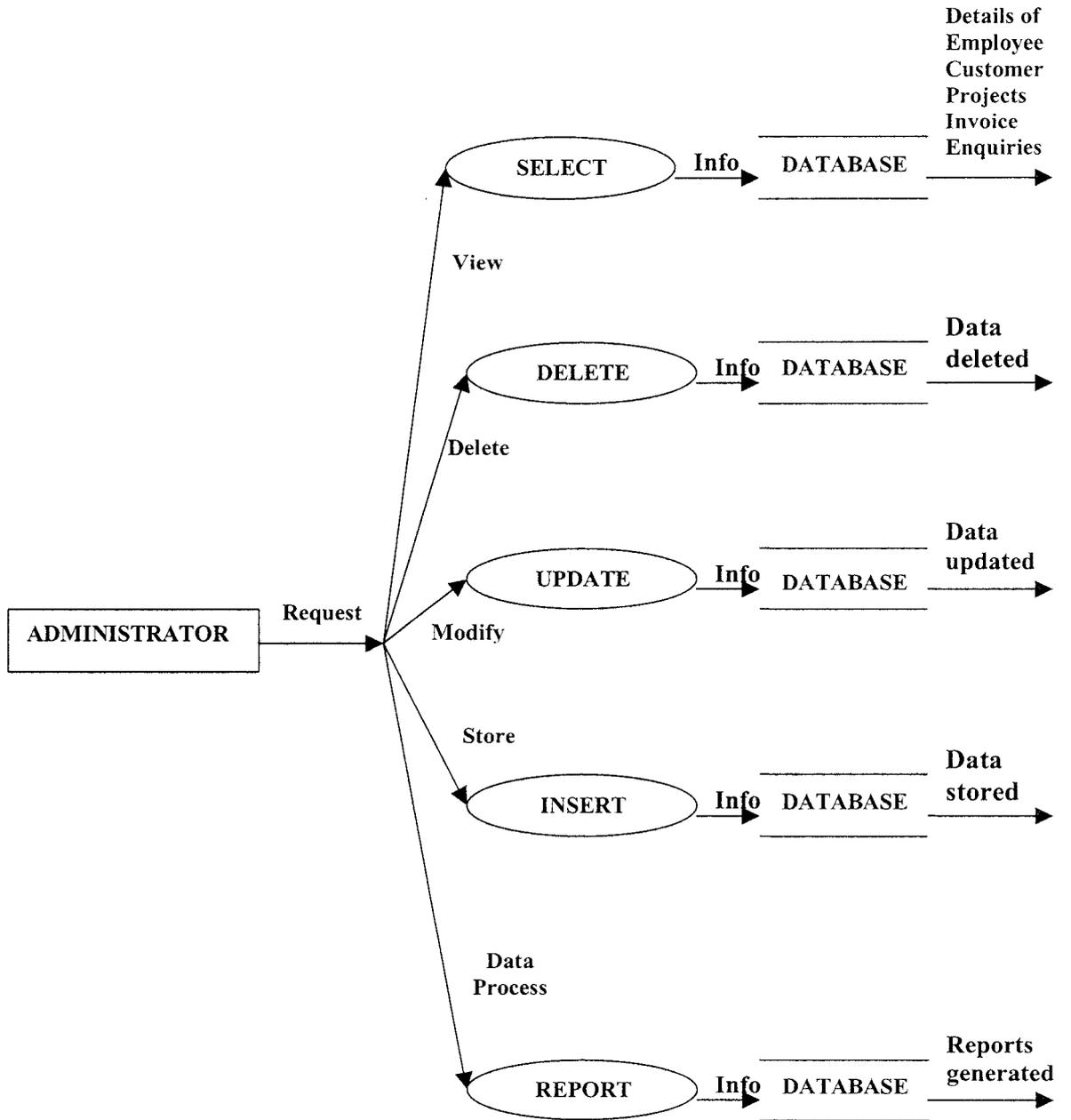
DFD - Business to Customer (B2C) Via Internet

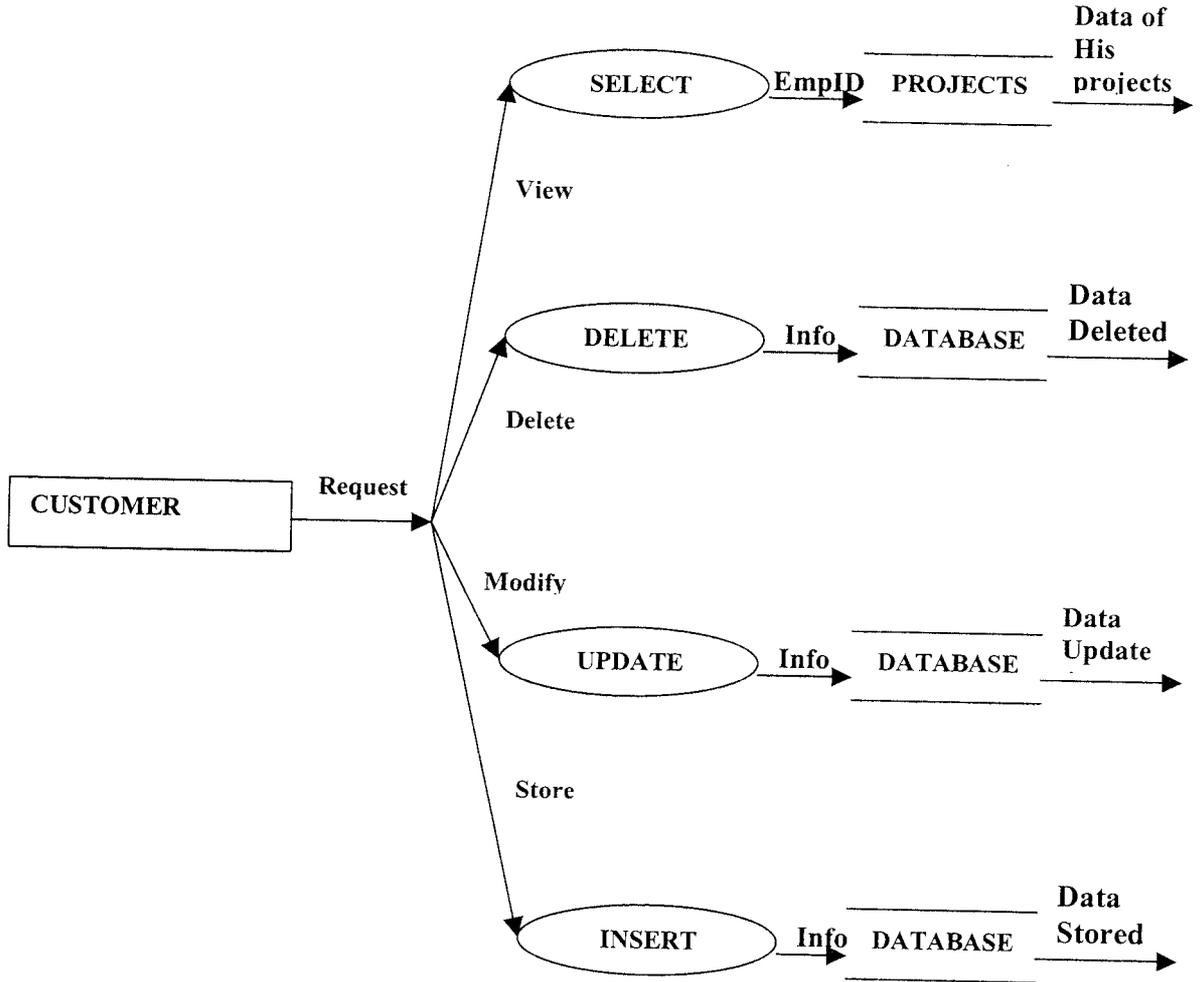**Figure 4.3.2**
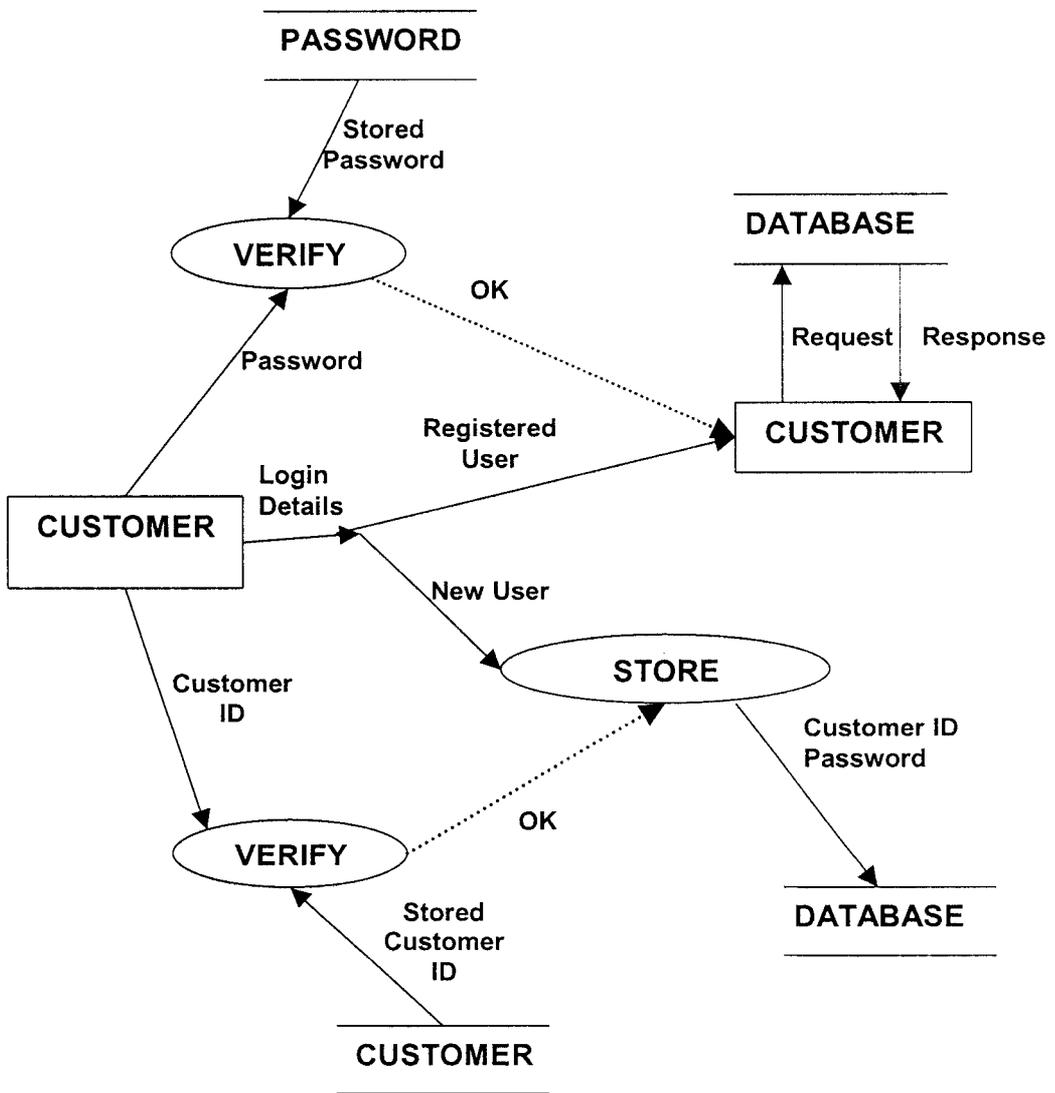
DFD – Process Flow

Figure 4.3.3
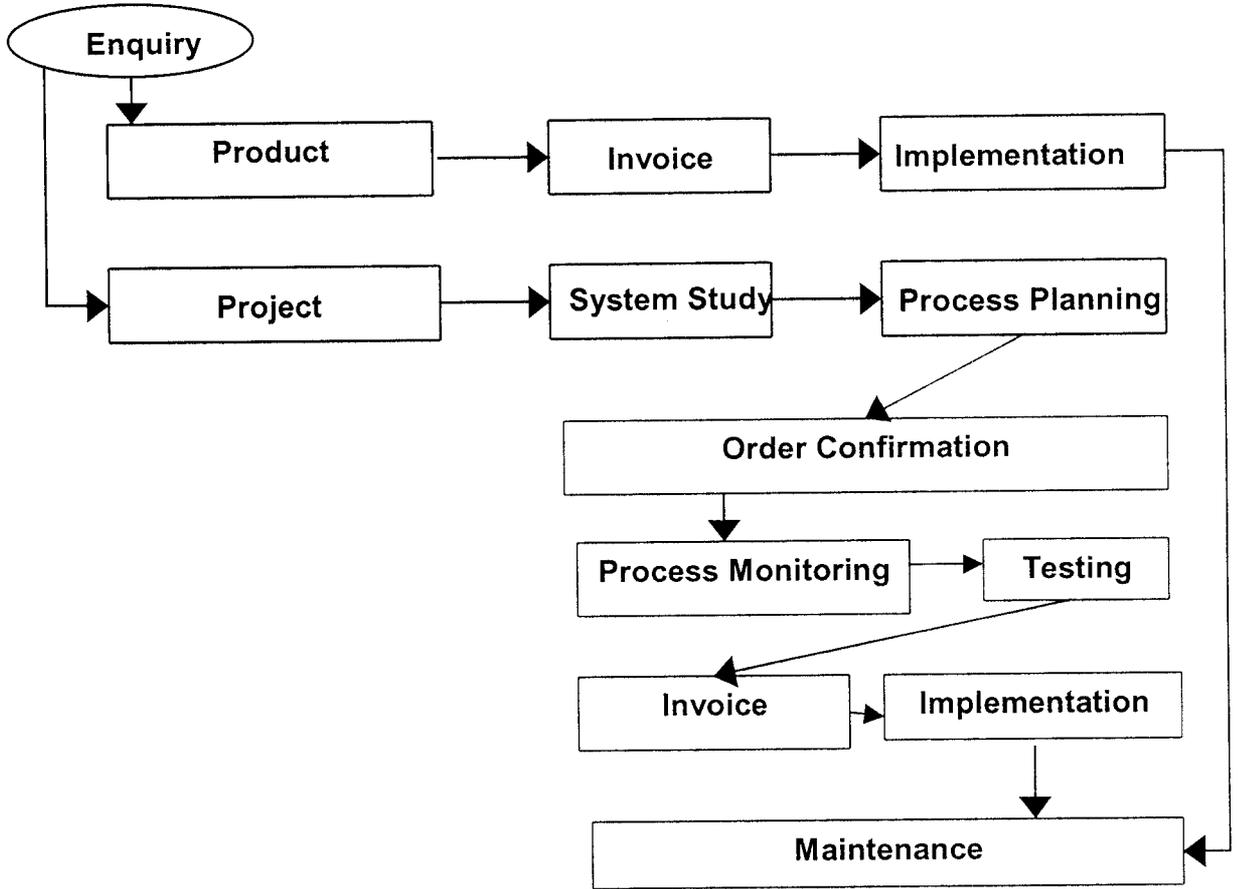
DFD - Administrator

Figure 4.3.4
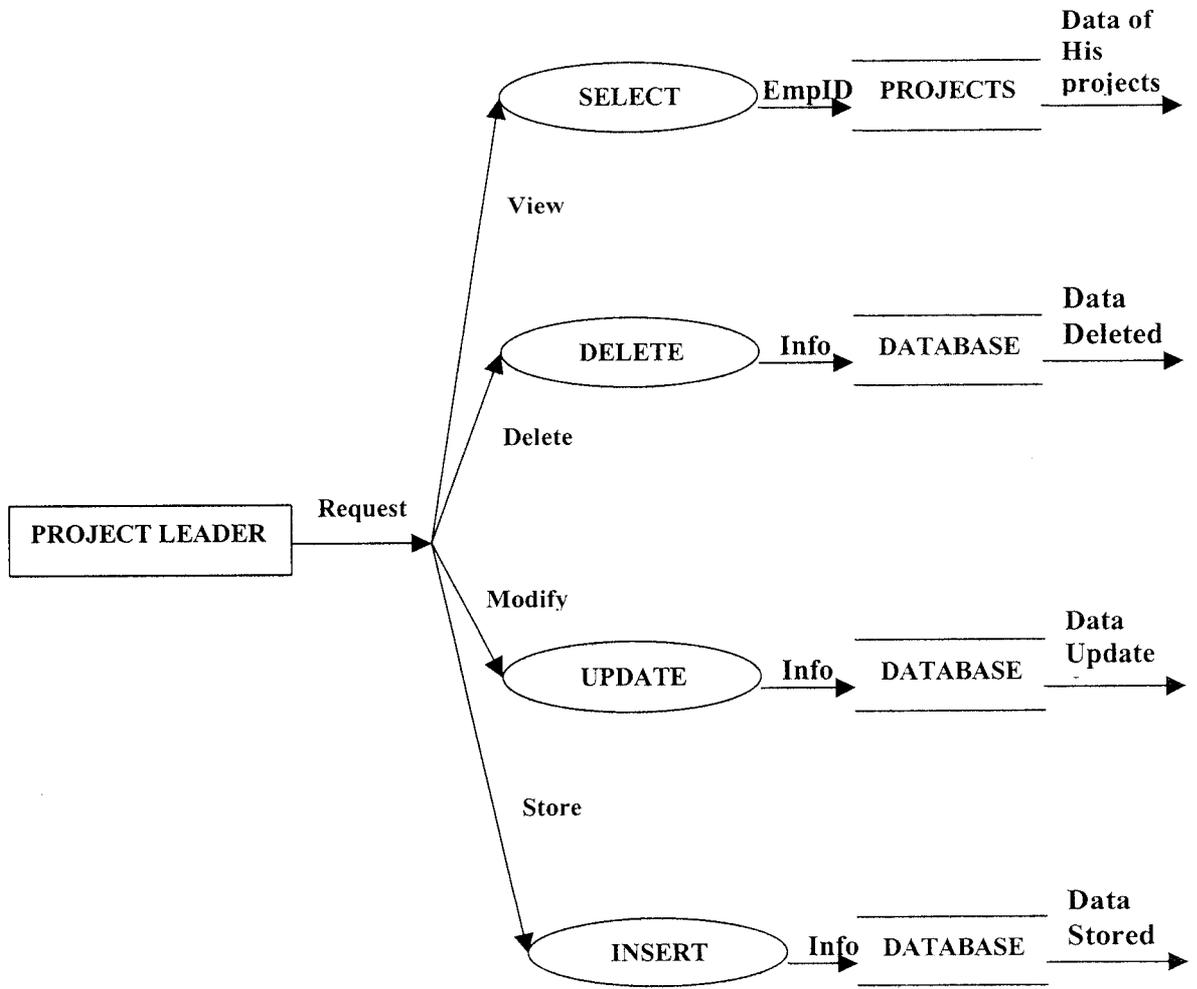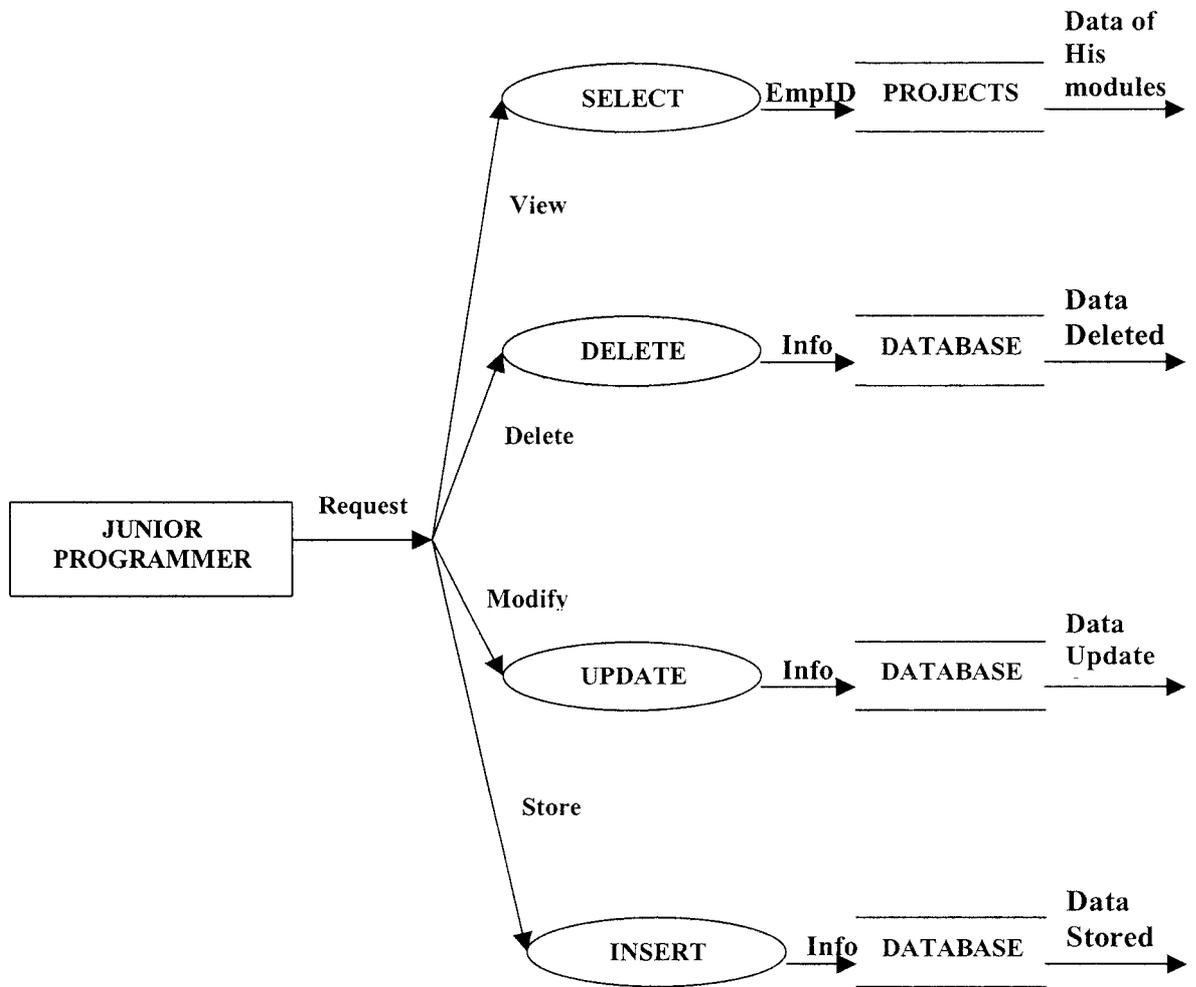
DFD - Customer

Figure 4.3.5

DFD - Project Leader

Figure 4.3.5

DFD - Junior Programmer

Figure 4.3.6

4.4 TABLE DESIGN

4.4.1 Table Name: Employee

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Decode	Varchar	20	Department Code
Ecode (PK)	Varchar	20	Employee Code
Ename	Varchar	30	Employee Name
Status	Varchar	25	Status
Skillset	Varchar	10	Skillet
Qua	Varchar	10	Qualification
Jdate	Date/Time	8	Join date
Exp	Number	2	Experience
Address	Varchar	50	Address
Zipcode	Number	8	Zip code
Contno	Number	8	Contact Number
Email	Varchar	30	E-mail

4.2.2 Table Name: Customer

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Code (PK)	Varchar	10	Employee Code
Name	Varchar	30	Employee Name
Sarea	Varchar	30	Specification Area
Website	Varchar	25	Website
Email	Varchar	30	E-mail
Contactno	Number	10	Contact no
Address	Varchar	50	Address
Zipcode	Number	8	Zip code

4.3.3 Table Name: Projects

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Pname	Varchar	30	Project Name
Pcode (PK)	Varchar	10	Project Code
Ccode	Varchar	10	Client Code
Cname	Varchar	30	Client Name
Sdate	Date/Time	8	starting date
Cdate	Date/Time	8	completed date
Cost	Number	7	Total Coast
Modules	Varchar	50	Number of modules
Swreq	Varchar	50	s/w requirements
Hwreq	Varchar	50	h/w requirements

.4.4.4 Table Name: Product Group

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Pgcode (PK)	Varchar	10	product group code
Pgname	Varchar	30	product group name
Products	Varchar	30	products

4.4.5. Table Name: Products

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Pgcode	Varchar	10	product group code
Pcode	Varchar	30	product code
Pname	Varchar	30	Product name
Tperiod	Number	3	Taining period
Tcost	Currency	10	Ttal cost
Modules	Memo	50	Mdules
Swreq	Memo	50	S/W requirements
Hwreq	Memo	50	H/W requirements

4.4.6 Table Name: New Enquiries

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Enqno (PK)	Varchar	10	Enquiry Number
Rdate	Date/Time	8	Received Date
Cname	Varchar	30	Client Name
Website	Varchar	30	Client Website
Email	Varchar	30	Client Email
Contno	Number	10	Phone Number
Address	Varchar	50	Adddress
Zipcode	Number	10	Zipcode

4.4.7 Table Name: Client Enquiries

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Enqno (PK)	Varchar	10	Enquiry Number
Rdate	Date/Time	8	Received Date
Ccode	Varchar	30	Client code
Cname	Varchar	30	Client Name
Query	Varchar	30	Queries
Aby	Varchar	30	Enquired person.

4.4.8 Table Name: Enquiry process

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Enqno	Varchar	10	Enquiry Number
Rdate	Date/Time	8	Received Date
Status	Varchar	30	Reply status
Rpdate	Date/Time	8	Replied date
Rpby	Varchar	30	Reply Person
Replay	Varchar	50	Reply

4.4.9 Table Name: Invoice

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Invno (PK)	Varchar	10	Invoice Number
Date	Date/Time	8	Received Date
Ccode	Varchar	10	Client Code
Cname	Varchar	30	Client Name
Amt	Date/Time	5	Received Amount
Recby	Varchar	30	Received Person

4.4.10 Table Name: SCom

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Pcode	Varchar	10	Project Code
Cdate	Date/Time	8	Completed Date
Modules	Varchar	50	Modules Cmpleted
Descrip	Varchar	50	Comments

4.4.11 Table Name: SProg

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Ecode	Varchar	10	Employee Code
Ename	Varchar	30	Employee Name
Sdate	Date/Time	8	Starting Date
Cdate	Date/Time	8	Completed Date
Modules	Varchar	50	Modules
Descrip	Varchar	50	Description

4.4.12 Table Name: JCom

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
Pcode	Varchar	10	Project Code
Cdate	Date/Time	30	Completed Date
Modules	Varchar	8	Modules
Descrip	Varchar	50	Description

CHAPTER 5

SYSTEM TESTING AND IMPLEMENTATION

The system testing deals with the process of testing the system as a whole. This is done after the integration process. The entire system is tested by moving through each module from top to bottom. The verification and validation processes are then carried out. The errors that occur at testing phase are eliminated and a well functioning system is developed.

Test case design focuses on a set of techniques, which meets all testing objectives, which are mentioned below.

1. Testing is a process of executing a program with the intent of finding an error.
2. A successful test is one that uncovers an as yet undiscovered error.

Testing demonstrates hat software functions work according to specifications. In addition data collected from testing provides a good indication of software reliability and some indication of software quality as a whole.

Testing results in the deduction in the numbers of errors. Critical modules are tested as early as possible. The following tests are carried out.

5.1 TESTING METHODS

5.1.1 Unit Testing

It focuses verification efforts on the smallest unit of software design, the module. This is also known as **Module Testing**. The modules are tested separately. This testing is carried out during programming stage itself.

5.1.2 Validation Testing

Validation Testing can be defined in many ways but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably expected by the users.

After validation test has been conducted one of the two possible conditions exist

1. The function or the performance characteristics confirm to specification and are accepted
2. A deviation from specification is uncovered and a deficiency list is create

5.1.3 Output Testing

After performing the validation testing the next step is output testing of the proposed system since no system is useful if it does not produce the required output in the specific format. The outputs generated or displayed by the system under consideration are tested by asking the users about the formats required by them.

5.1.4 User Acceptance Testing

User acceptance of a system is a key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system users at the time development and making changes whenever required.

5.2 IMPLEMENTATION

The implementation phase of software development is concerned with translating design specification into source code. The primary goal of implementation is to raise source code and internal documentation so that conformance of the code to its specification can be easily verified, and so that debugging, testing and modification are eased. This goal can be achieved by making the source code as clear and straightforward as possible. Simplicity, clarity and elegance are the hallmarks of good programs; obscurity, cleverness and complexity are indication inadequate design and misdirected thinking.

Source code clarity is enhanced by structured coding techniques, by good coding style, by appropriate supporting documents, by good internal comments and by the features provided in modern programming languages.

The goal structured coding is to liberalize control flow through a computed program so that the execution sequence follows the sequence in which the code is written. The dynamic structure of a program as it executes then resembles the static structure of the written text. This enhances readability of code, which eases understanding, debugging, testing, documentation and modification of programs. It also facilitates formal verification of programs. The structure coding techniques are as follows

- Single entry, Single exit constructs
- Efficiency considerations
- Data Encapsulation
- Recursion

5.3 MAINTENANCE

Maintenance is the enigma of system development. It holds the software industry captive typing up programming resources. It could be described as the symmetric process of changing the software that is already in operation iq. order to prevent system failures and to improve the performance. Software maintenance involves keeping software interfaces simple and standard, paying particular attention to troublesome modules, replacing faulty components and generally planning to replace components that are old, obsolete, faulty, or at risk for imminent failure.

There are several factors that require to be maintained. They are

- Hardware platforms change or become obsolete.
- Operating system change.
- Compiler change
- Language standard's change.
- Language standard's change.
- Communication standard's change.
- Graphical user interface change.
- Related application software package change.

Maintenance can be classified into

- Adaptive maintenance
- Perceptive maintenance
- Preventive maintenance
- Corrective maintenance

Adaptive Maintenance

It deals with adapting software change in the environment. It does not lead to changes in the system functionality.

Perceptive Maintenance

It mainly deals with accommodating new or changed users requirements. It also includes activities to increase the system performance or to enhance its user interface. The objective of perceptive maintenance should be to prevent failures and optimize the software.

Preventive Maintenance

Preventive maintenance concerns activities aimed at increasing the system's maintainability such as updating documentation adding comments, improving modular structure of the system.

Corrective Maintenance

This deals with the repair of faults found. Some of the major causes of maintenance problems are

- Unstructured code.
- Maintenance programmers having insufficient knowledge of the system and on application domain.

CHAPTER 6

CONCLUSION

Customer Relation Management System is successful software with all the requirements being satisfied. The software has successfully achieved the functionality that was expected.

It contains two major packages

- Internal Office Automation application
- Business to Customer (B2C) via internet

It provides a user friendly environment and reduces the customer difficulties. It upgrades the system to the **Java Enterprise Platform**.

The conclusion leads to some ideas for further enhancements for this software.

Further Enhancements

- Provide facilities for online money transaction.
- Provide facilities for online delivery of the product.
- Provide facilities for BPO.
- Can be integrated over HR division.
- Remote administration can be enabled.
- Can be extended over to all branches of the industry.

APPENDIX- SCREEN LAYOUT

CRM Main Interface
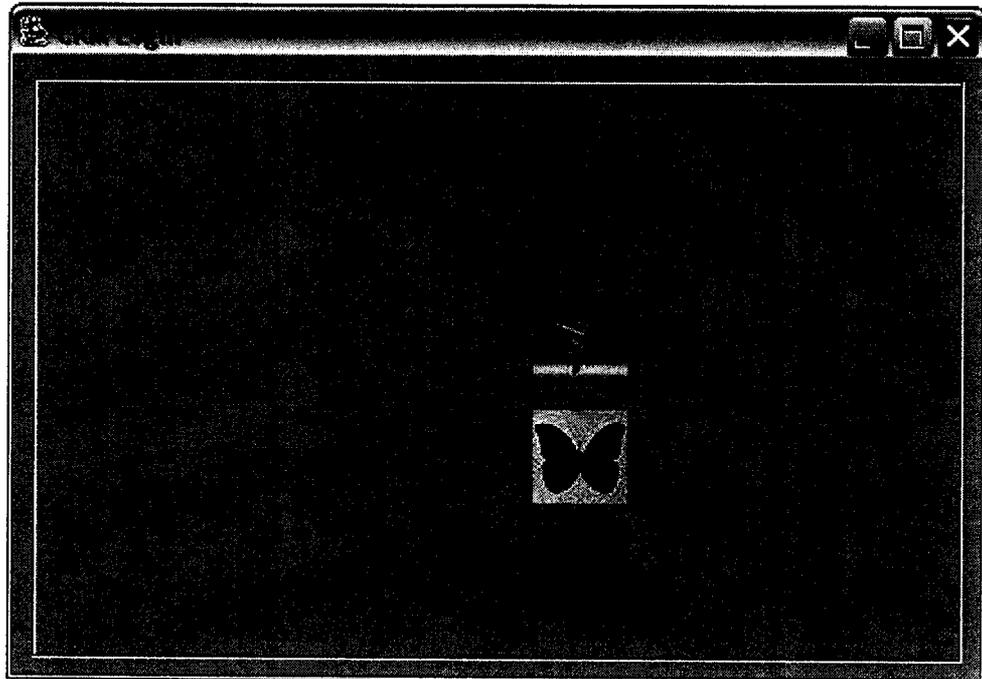
Admin Login
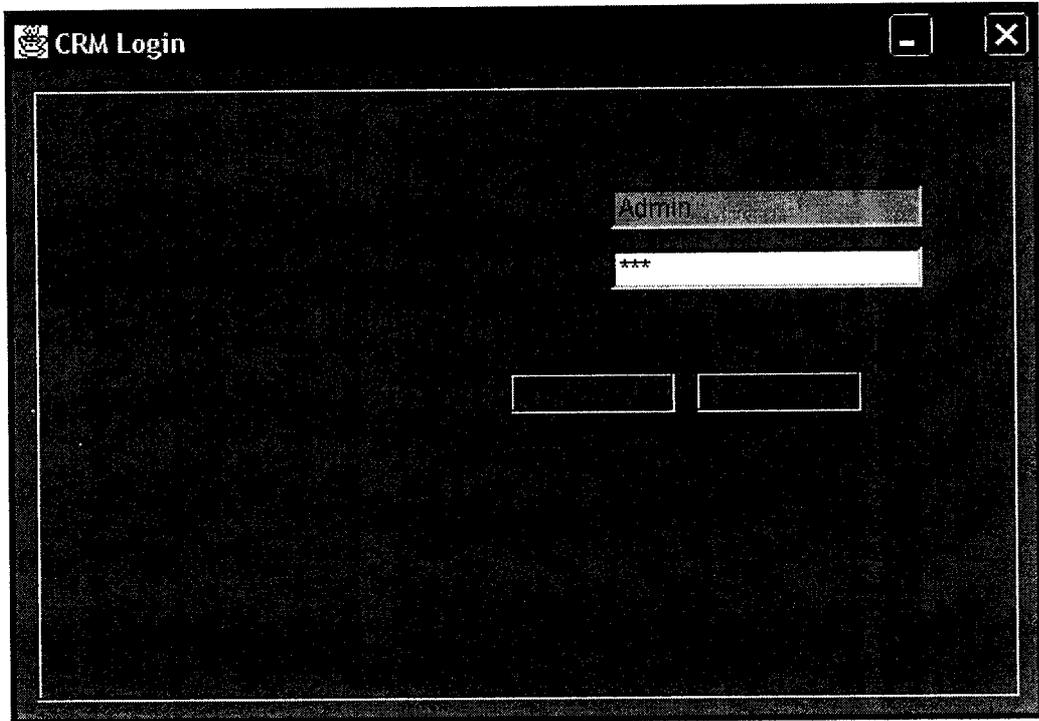
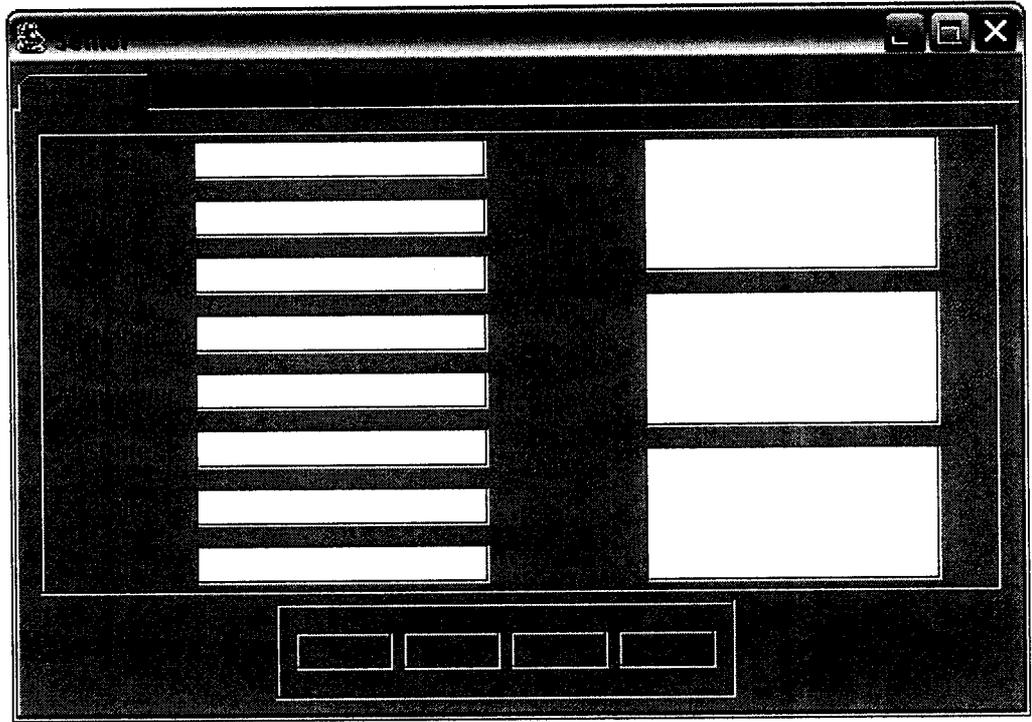## Administrator Interface

Admin-Master Menu
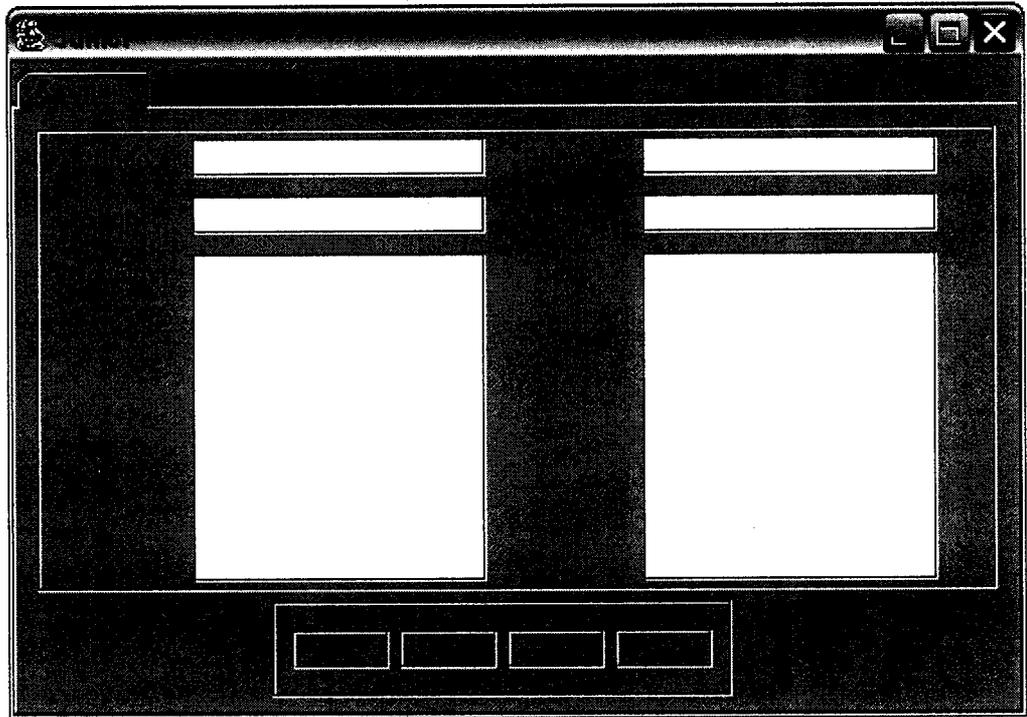
43
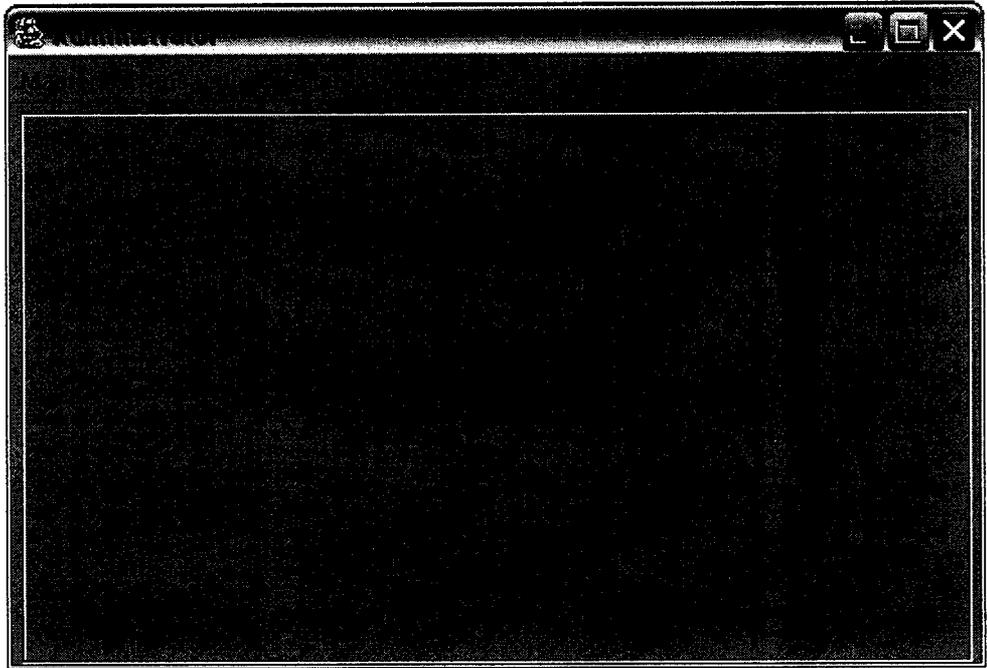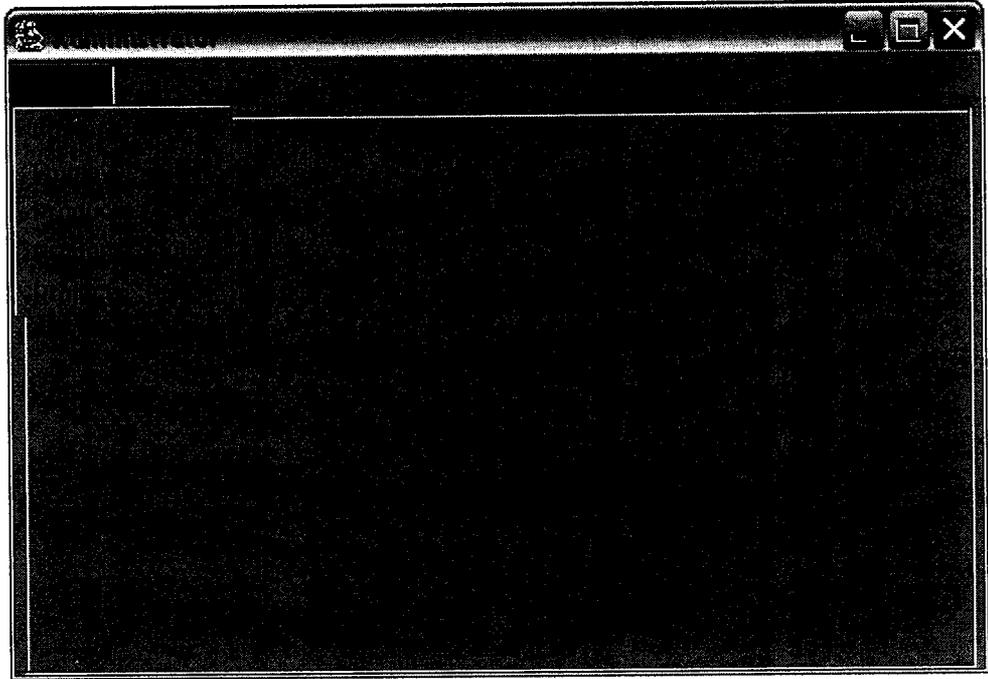
Admin-Transaction Menu
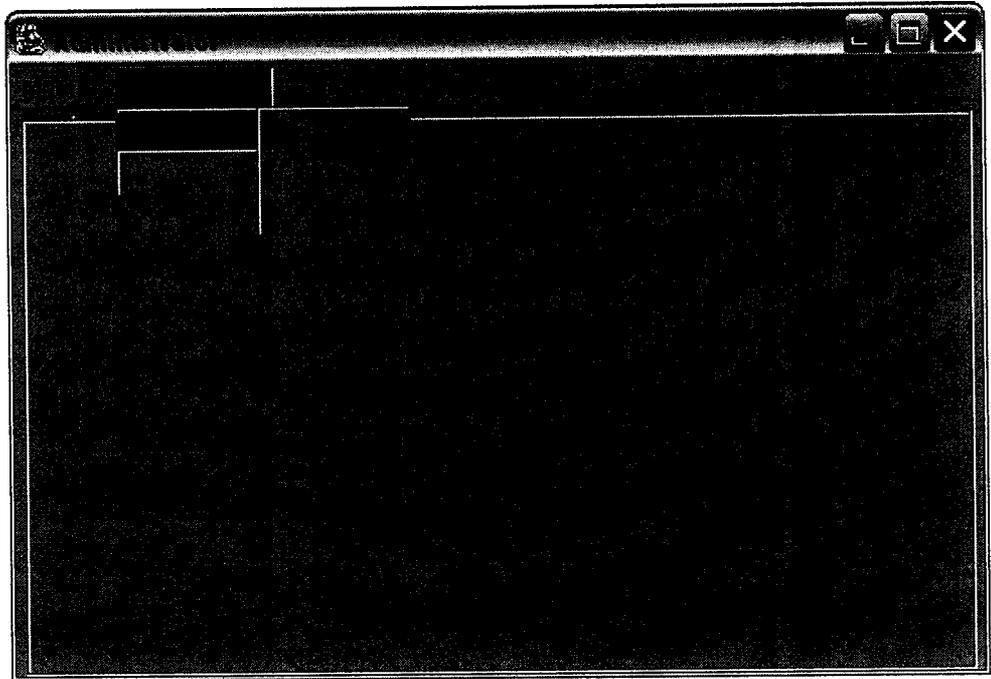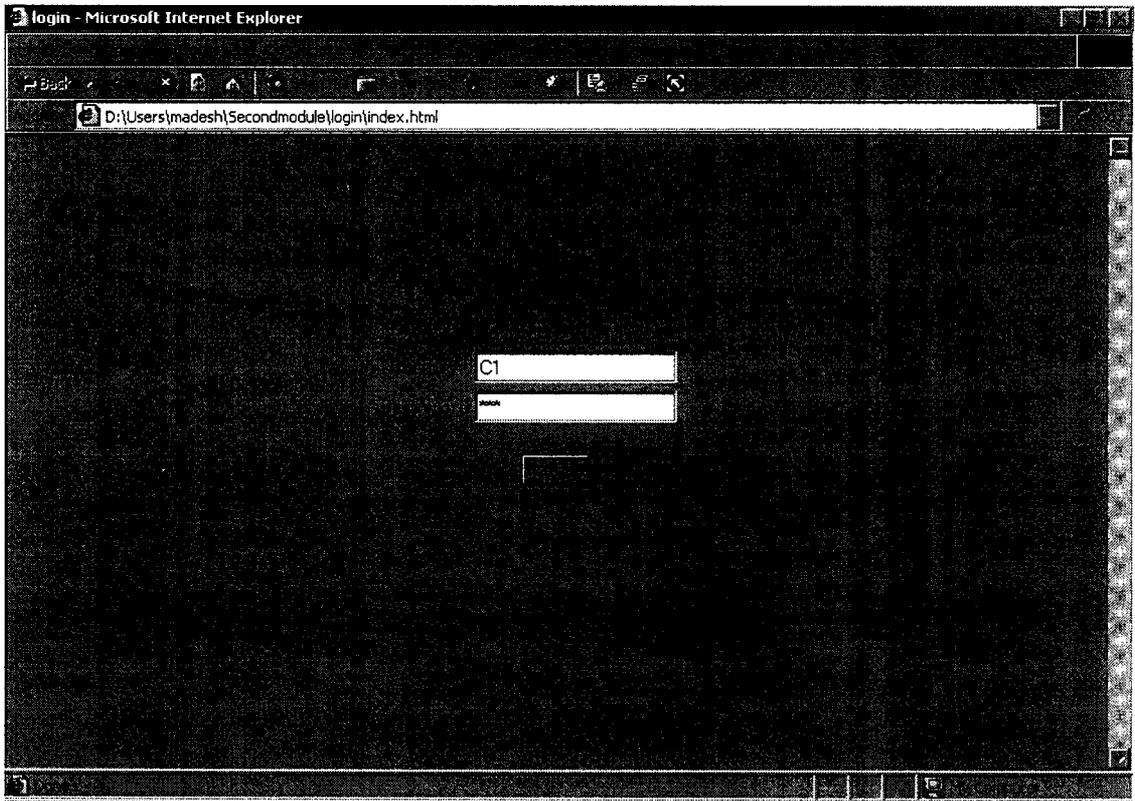
Admin-Report Menu
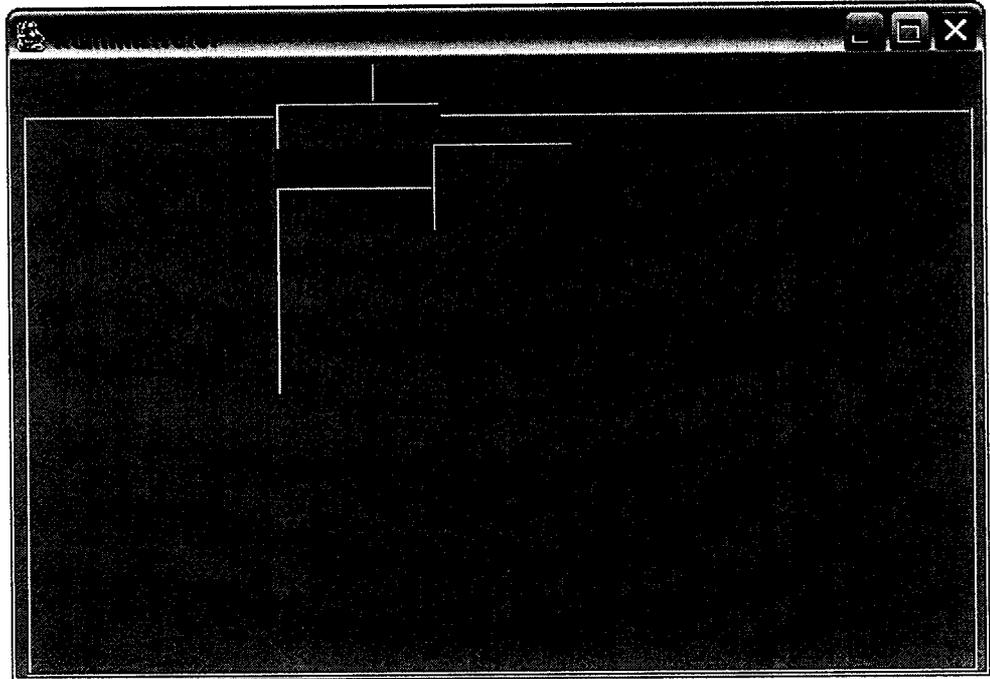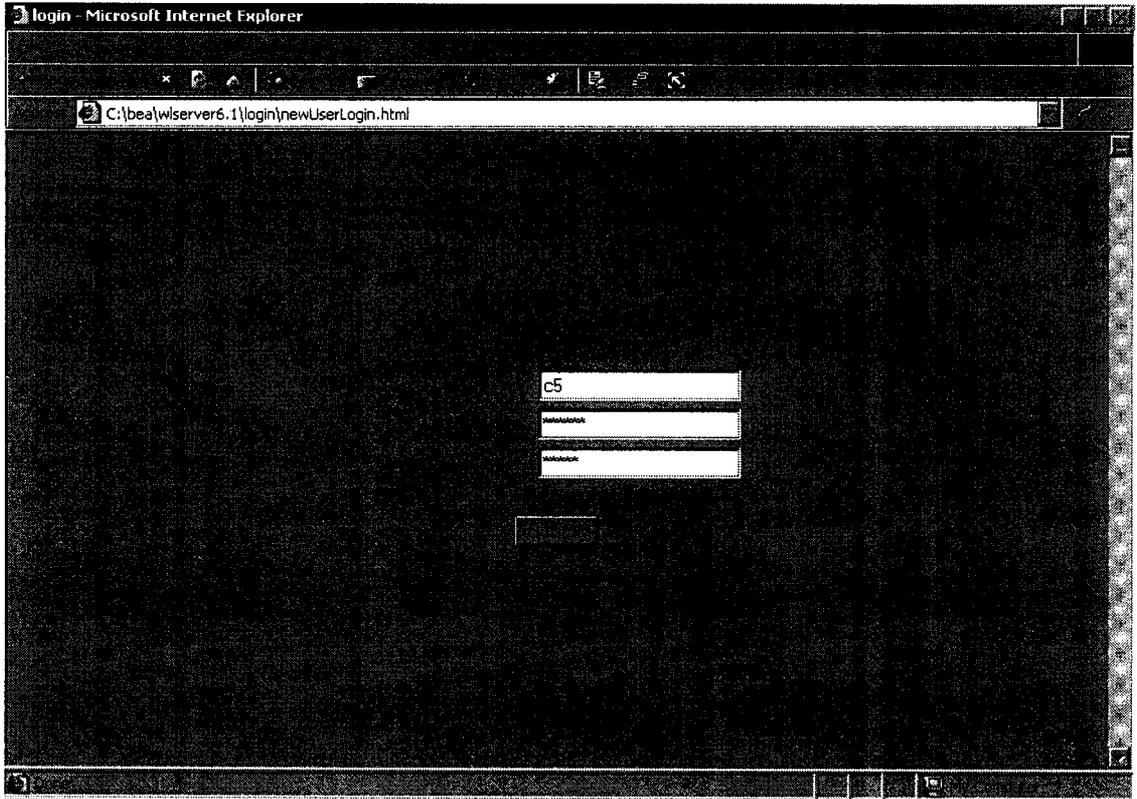
Project Leader Interface

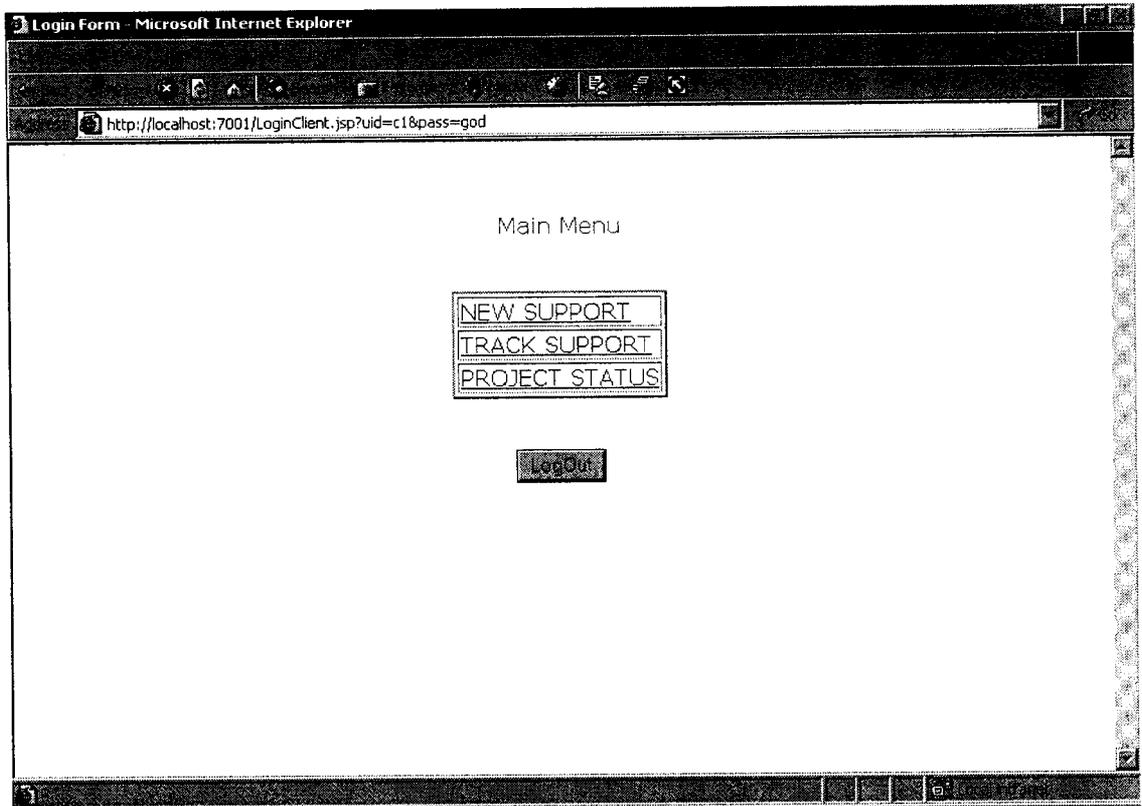**Junior Programmer Interface**

Client Login Form

New Client Entry Form

Main menu

REFERENCES

- J2SE API documentation of Sun Microsystems.
- JSP documentation of Sun Microsystems.
- EJB documentation of Sun Microsystems.
- Java complete reference, Patrick Naughton & Herbert Schmidt]
- Java programming language, Ken Arnold & James Gosling
- Software engineering concepts, Richard Fairly
- Java How to program, Deitel and Deitel
- Online reference for java www.java.sun.com