

P-1748



INTEGRATED OFFICE MANAGER

By

**S.Senthamil Kannan
(Reg.no.71203621047)**

Of

**KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE**

(Affiliated to Anna University)

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

*In the partial fulfillment of the requirements
for the award of the degree
of*

MASTER OF COMPUTER APPLICATIONS

June, 2006

Kumaraguru College of Technology

Coimbatore

Department of Computer Applications

BONAFIDE CERTIFICATE

Certified that the project report titled "**Integrated Office Manager**" is the bonafide work of **Mr. S. Senthamil Kannan** who carried out the project under my supervision. Certified further that to the best of my knowledge the work reported here 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.

C. Anand
13/6/06
PROJECT GUIDE

[Signature]
HEAD OF THE DEPARTMENT

Submitted for the University Examination held on

29-06-06

C. Anand
29/6
INTERNAL EXAMINER

[Signature]
29/6/06
EXTERNAL EXAMINER



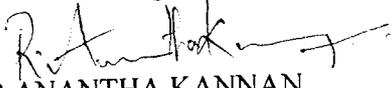
09/06/2006

TO WHOM SOEVER ITMAY CONCERN

This is to certify that Mr.S.SENTHAMIL KANNAN, Roll No. 03MCA49 Master of Computer Application, final year student of KUMARAGURU COLLEGE OF TECHNOLOGY has completed his project entitled "Integrated Office Manager" In our concern from January '06 till June'06.

During the period the conduct and attendance found to be good.

Authorized Signatory


R.ANANTHA KANNAN
(DIRECTOR)

ABSTRACT

The main objective of Integrated Office Manager is to control all the activities in Software Company. The existing system prevailing in company is manual paper work. The system also improves communication between employees and management.

The core concept of this application is to maintain the client information which includes project details, payment details and there is provision for administrator for sending daily status to corresponding client. This system facilitates to store information about recruited persons (Freshers) and their marks obtained in tests conducted during their probation period. Based on the marks obtained in tests, freshers are designated as Permanent Employee.

This system also allows the administrator to grant/reject permission for the employee who applies for leave or resource, the status about their request being sent through mail to the concerned employees.

Above all, this system aims at automating the tasks associated with the processing and tracking of software defect reports. The module includes methods for generation of software problem reports, reporting the problem to the concerned person, and tracking the status of each software problem report. This system aims to provide a centralized control over problem report and communication of problem between the project team mates.

The potential objective is to make the implementation of the system in a generalized manner, which means that with some minimal modification it could be adapted into any organization.

The system was developed in VB.NET and SQL Server (Backend).

ACKNOWLEDGEMENT

I would like to express my gratitude and humble thanks to our beloved principal **Dr. Joseph.V.Thanikal**, for having given me the adequate support and opportunity for completing this project work successfully.

I would like to express my deep sense of gratitude to **Dr. M. Gururajan**, HOD, Department of Computer Applications for providing moral support towards this project work.

I am deeply indebted to **Ms. V. Geetha**, Assistant Professor, Department of Computer Applications, my internal project guide for offering her guidance, timely encouragement and support to me for the completion of this project.

I thank **Mr. A. Muthukumar**, Assistant Professor, Department of Computer Applications for his encouragement to do this project.

I would like to express my sincere thanks to **Mr. R. Anantha Kannan**, Project Leader, Honeybee Technologies, Coimbatore, my external guide for his support, and experience from which I have greatly benefited.

Finally I thank **my lovable parents and friends** who helped me in many ways during the course of project and have made it great success.

TABLE OF CONTENTS

	PAGE NO
ABSTRACT	iii
ACKNOWLEDGEMENT	iv
LIST OF TABLES	vii
LIST OF FIGURES	viii
LIST OF SYMBOLS, ABBREVIATIONS AND NOMANCLATURE	viii
1. INTRODUCTION	1
1.1 OVERVIEW OF THE PROJECT	1
1.2 ORGANIZATION PROFILE	2
2. SYSTEM REQUIREMENT AND SPECIFICATION	3
2.1 HARDWARE REQUIREMENTS	3
2.2 SOFTWARE REQUIREMENTS	3
2.3 SOFTWARE OVERVIEW	4
3. SYSTEM ANALYSIS	7
3.1 PROBLEM STATEMENT	7
3.2 EXISTING SYSTEM	7
3.2.1 DRAWBACK OF EXISTING SYSTEM	8
3.3 PROPOSED SYSTEM	8
3.3.1 BENEFITS OF PROPOSED SYSTEM	9

4. SYSTEM DESIGN	10
4.1 ELEMENT OF DESIGN	10
4.1.1 INPUT DESIGN	10
4.1.2 OUTPUT DESIGN	12
4.1.3 ARCHITECTURE DESIGN	13
4.1.4 DATABASE DESIGN	15
4.2 TABLE DESIGN	16
4.3 DATA FLOW DIAGRAM	24
5. SYSTEM TESTING AND IMPLEMENTATION	30
5.1 SYSTEM TESTING	30
5.1.1 UNIT TESTING	30
5.1.2 INTEGRATION TESTING	30
5.1.3 VALIDATION TESTING	31
5.1.4 ALPHA TESTING	31
5.1.5 BETATESTING	31
5.2 SYSTEM IMPLEMENTATION	32
6. CONCLUSION AND FUTURE ENHANCEMENT	33
7. APPENDICES	34
8. REFERENCES	46

LIST OF TABLES

TABLE DESCRIPTION		PAGE NO
Table 4.2.1	User	16
Table 4.2.2	ProjectMaster	16
Table 4.2.3	FresherMaster	17
Table 4.2.4	FresherMark	17
Table 4.2.5	ClientMaster	18
Table 4.2.6	ClientPayment	18
Table 4.2.7	EmployeeProject	19
Table 4.2.8	ClientStatus	19
Table 4.2.9	ProjectStatus	20
Table 4.2.10	SPR	20
Table 4.2.11	SDCR	21
Table 4.2.12	EmployeeMaster	22
Table 4.2.13	Resource	22
Table 4.2.14	Resourcestatus	23
Table 4.2.15	LeaveMaster	23
Table 4.2.16	LeaveStatus	23

LIST OF FIGURES

FIGURE DESCRIPTION	PAGE NO
Figure 4.3.1 LOGIN PROCESSING	24
Figure 4.3.2 FRESHER INFORMATION MAINTENANCE	25
Figure 4.3.3 LEAVE MANAGEMENT	26
Figure 4.3.4 RESOURCE MANAGEMENT	27
Figure 4.3.5 DEFECT TRACKING SYSTEM	28
Figure 4.3.6 CLIENT INFORMATION MAINTENANCE	29

LIST OF SYMBOLS, ABBREVIATIONS AND NOMANCLATURE

SPR	-	Software Problem Report
SDCR	-	Software Design/Code Change Report
RFA	-	Request For Action
SQL	-	Structured Query Language
DFD	-	Data Flow Diagram
SMTP	-	Simple Mail Transfer Protocol

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW OF PROJECT

1.1.1 INTEGRATED OFFICE MANAGER

Integrated Office Manager is centralized software, controlling all the activities in the Software Company, the project aims at automating all the activities in the Software Company. The system also improves communication between employees and management.

The Modules in the Integrated Office Manager are,

- ❖ Profile Management(Login Processing)
- ❖ Fresher Information Module
- ❖ Leave Management
- ❖ Resource Management
- ❖ Defect Tracking System
- ❖ Client Information Maintenance

1.1.2 Project Scope

The scope of the Project is as follows:

- ❖ Access all the details from a centralized location.
- ❖ Quick and easy search for variety of information.
- ❖ Enhance efficiency by eliminating redundant data.
- ❖ Send Mails within company through Mail System to intimate about resource and leave status.

1.2 ORGANIZATION PROFILE

HONEYBEE TECHNOLOGIES was established on August 18th 2005 at Coimbatore, Tamilnadu. The firm was initially started as “**Embedded System Engineer Training Institute**”. By September 1st 2005 the “**Research and Development**” in the field of Electronics and **Real Time Industrial projects** had been launched. The institute offers a very good infrastructure with excellent training methodology, experienced engineers to train the students for the electronic industries. They also assist the students for placement. Our motto is to provide a unique solution to all our customers.

SERVICES

The Company currently provides embedded product development, research and development services. They conduct various embedded courses for students and trainees. They can offer placement for payable trainees. And they provide various value added seminars for students and trainees.

The **objectives** - though many - can easily be summarized as below:

- ❖ **CREATING** (embedded system) qualified professionals for the electronic industry.
- ❖ **PROVIDING** (the best possible) Technical service to all kinds of electronic industries.

CHAPTER 2

SYSTEM REQUIREMENT AND SPECIFICATION

The system requirement and specification provided here are used in the development of this system.

2.1 HARDWARE REQUIREMENTS

The hardware components on which this application is developed are,

Processor	:	Intel Pentium III 1.13GHZ
Primary Memory (RAM)	:	256 MB
Secondary Memory (Hard disk)	:	40 GB
Monitor	:	Samsung, COLOR, 15inch
Display card	:	SVGA
Mouse	:	HCL
Keyboard	:	Samsung Standard 101 Keys

2.2 SOFTWARE REQUIREMENTS

The software components required to develop this application are,

Operating System	:	Windows XP and above
Language	:	VB.NET
Back-End	:	MICROSOFT SQL-SERVER 2000.

2.3 SOFTWARE OVERVIEW

VB.NET which is the new version of Visual Basic. There is Visual Studio.NET, an Integrated Development Environment that hosts VB.NET, C#, and C++.NET. Underlying all this is the .NET Framework and its core execution engine, the Common Language Runtime. In the .NET model, you write applications that target the .NET Framework. This gives them automatic access to such benefits as garbage collection (which destroys objects and reclaims memory for you), debugging, security services, inheritance, and more. When you compile the code from any language that supports the .NET Framework, it compiles into something called MSIL, or Microsoft Intermediate Language. This MSIL file is binary, but it is not machine code; instead, it is a format that is platform independent and can be placed on any machine running the .NET Framework. Within the .NET Framework is a compiler called the Just-In-Time, or JIT, compiler. It compiles the MSIL down to machine code specific to that hardware and operating system.

ADO.NET

ADO.NET uses a disconnected architecture to operate. The reason for this is that traditional applications that maintained an open database connection did not scale well. If a component opened a connection to the database and held it open for the lifetime of the application, it consumed expensive database resources, when the connection probably was needed for only a small percentage of that time.

As the number of users grows, the overhead of the database connections can begin to affect the database performance negatively. Therefore, Microsoft decided to use a disconnected architecture for ADO.NET. This not only solves the problems with scalability by reducing the number of active connections, it makes it much easier to transfer data from one component to another. You do not need to have both components connected to the database, nor do you have to have them both understand some binary format of the data, as you will see in a moment.

Additionally, Microsoft recognized that in much programming today, you basically have disconnected application architecture, thanks to the Web. Think of a typical Web application: The user requests a page and the page is generated on the server, with a mix of HTML and data from the database. By the time the page is sent back to the user and rendered in his browser, the connection to the database is no longer needed. Therefore, if it takes one second to render a page but the user views it for twenty-nine seconds, a database connection is needed for only one-thirtieth of the total time the page is used. If you still need to use a connected architecture, Microsoft recommends you use ADO. ADO.NET is inherently disconnected, so ADO is still a better approach if you need a continuous connection to the underlying database.

Overview of SQL SERVER 2000

A database is similar to a data file where data gets stored. Like a data file, a database does not present information directly to a user; the user runs an application that accesses data from the database and presents it to the user in an understandable format.

Database systems are more powerful than data files in that data is more highly organized. In a well-designed database, there are no duplicate pieces of data that the user or application must update at the same time. Related pieces of data are grouped together in a single structure or record, and relationships can be defined between these structures and records.

When working with data files, an application must be coded to work with the specific structure of each data file. In contrast, a database contains a catalog that applications use to determine how data is organized. Generic database applications can use the catalog to present users with data from different databases dynamically, without being tied to a specific data format.

Although there are different ways to organize data in a database, relational databases are one of the most effective. Relational database systems are an application of mathematical set theory to the problem of effectively organizing data. In a relational database, data is collected into tables (called relations in relational theory).

When organizing data into tables, you can usually find many different ways to define tables. Relational database theory defines a process called normalization, which ensures that the set of tables you define will organize your data effectively.

SQL Server 2000 supports having a wide range of users access it at the same time. An instance of SQL Server 2000 includes the files that make up a set of databases and a copy of the DBMS software. Applications running on separate computers use a SQL Server 2000 communications component to transmit commands over a network to the SQL Server 2000 instance.

CHAPTER 3

SYSTEM ANALYSIS

System Analysis is the process of understanding a problem domain and the user requirements for the purpose of developing a computer application system to serve the needs of the users.

3.1 PROBLEM STATEMENT

The main objective of this system is to control all the activities in the software company, by automating all the tasks. The system also improves communication between employees and management. This system should be user-friendly and should be able to provide access to the user who has appropriate username and password.

3.2 EXISTING SYSTEM

The System is to automate all the activities in the software company. The existing system prevailing in the company is manual paper work. Client information is maintained manually in the company, a form is provided for counselor for filling client details including the payment and project details. Fresher information including marks in their probation period was maintained manually.

For applying a resource or leave, the employee fills the corresponding request form and submits to Counselor/Resource Manager. He/She fills the request made by employees in another form and forwards the request form to the administrator, he/she would update the status of the request and the updated status is vocally informed to the concerned employee.

In Defect Tracking System, the tester tests the module with the test cases, when tester identifies a bug in the modules he reports to the project leader by filling up a form (SPR) describing a problem. It is assigned a number and forwarded to the Project Leader, who validates that bug report to check whether it is valid or not.

The approved bug report is send to the designer/developer who actually developed the module under consideration. The designer/developer is supposed to update the code and send the details of code change to the originator of SPR through another form (SDCR). Thus, once the problem is actually solved, RFA is being sent to Software Configuration Manager who files the report forms and carry out the version change.

3.2.1 Drawbacks of the existing system

The following were the drawbacks of existing system:

- ❖ Requirement of more man power
- ❖ Difficult to maintain.
- ❖ Inefficient & not flexible
- ❖ Involves high cost & resources

3.3 Proposed System

The purpose of implementing the proposed system is to reduce the manual work in organization. It also has some potential benefits such as minimizing the cost, and increases the quality service to the clients. The cost is significantly reduced as compared to the third party tools.

The proposed system aims to provide a good user interface to fill forms, send forms to the responsible person and track status of each request at any point of time. It also aims to provide a easier communication facilities between the team members. Each user is provided with user type and password and can use the

system with those parameters. According to user type the access privileges are given to the user.

Separate form is provided for administrator to use, in which he can control the users of the system. More over the administrator can grant the desired access permissions to the users of the system.

3.3.1 Benefits of Proposed System

The following are benefits of proposed system:

- ❖ Single Core System
- ❖ Centralized Database System
- ❖ Compatible for integration of additional automated system
- ❖ Intranet System
- ❖ Users can send mail through Mail System
- ❖ Users are granted rights for use of the centralized database as per their usage

CHAPTER 4

SYSTEM DESIGN

System design is the most creative and challenging phase in the life cycle of system development. The first step to determine is what input data is needed to form the system and the database that has to be designed should meet the requirement of the proposed system. The next step is to determine how the output is produced and in what format.

4.1 ELEMENT OF DESIGN

The following are the elements of the system design:

- ❖ Input design
- ❖ Output design
- ❖ Architectural design
- ❖ Database design

4.1.1 Input design

The input design is the process of converting the user- oriented inputs into the computer-based format. The goal of designing input data is to make the automation as easy and free from errors as possible.

The input design requirement such as user friendliness, consistent format and interactive dialogues for giving the right message and help for the user at right time are also considered for the development of the project.

Throughout the system, the forms used to get input are,

User Login Form

This form reads user type and password as input and validates it for authorization.

Administrators Add User Form

This form is used only by administrator to add new user with access permission to handle the system.

Change Password Form

This form is used by the users of this software to change their password. It will read input like user id, old password, and new password.

Client Detail Entry Form

This form is used by the counselor to add new client and their project details. It will read the input like name, address, phone number, payment details, and project signed for that client.

Fresher Mark Entry Form

This form is used by the counselor to enter the marks of the particular fresher during the propagation period. It will read inputs like Fresher id, Fresher name, and marks...etc

Software Problem Report Form

This form is used by the tester to enter all the defect details, functionality in which error has occurred, error category ...etc It will read the inputs like project id, defect category, defect severity, filename....etc.

Software Design/Code Change Report Form

This form will be used by the programmer to enter the details of change of the particular product for which software problem report has been raised. It will read inputs like SDCR No, estimated start date, estimated end date, estimated effort(in days)...etc

Resource Entry Form

This form will be used by resource manager to add a new resource request for the employee, with details such as employee id, resource id, resource name, purpose...etc

Resource Updating Form

This form is used only by the administrator to update the resource details of the employee.

Leave Entry Form

This form will be used by counselor to add a new leave request for the employee, with details such as employee id, employee name, purpose...etc

Leave Updating Form

This form is used only by the administrator to update the resource details of the employee.

The feedback form also provided with the system to get the valuable suggestion from the fresher.

4.1.2 OUTPUT DESIGN

A quality output is one which meets the requirement of the end user and presents the information clearly. The application output design is customized based on user input, which will generate the data depending on user's requirement. The output of this system is generated with the intention of two types of people.

Organization staffs

These staff are mostly counselor, resource manager, and employees. They can view their related information such as client details, resource allocation details, leave allocation details, project details, software problem report status, software design/code change report details ...etc.

Administrator

Administrator can view all the reports, such as fresher mark report, client project report, resource and leave allocation report, software problem reportetc

4.1.3 ARCHITECTURE DESIGN

Architectural design is concerned with refining the conceptual design 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.

4.1.3.1 Integrated Office Manager – Module Overview

The Modules in the Integrated Office Manager are,

- ❖ Profile Management(Login Processing)
- ❖ Fresher Information Maintenance
- ❖ Leave Management
- ❖ Resource Management
- ❖ Defect Tracking System
- ❖ Client Information Maintenance

Profile Management (Login Processing)

This module accepts login and password from the user and validates authorization. If it is a valid user then the module guides them further to the next module. User has the option for changing his password. Only Administrator can assign new user for the system.

Fresher Information Module

Fresher information module facilitates to store information about recruited persons (Freshers) and their marks obtained in tests conducted during probation period. Based on the marks obtained in tests, freshers are designated as Permanent Employee.

Leave Management

Leave management module facilitates the Administrator to grant/reject permission for the employee who has applied for leave. The Administrator would update the request status and the updated status is intimated to requested employee through mail with the help of this system.

Resource Management

Resource management module facilitates the Administrator to allocate/reject permission for the employee who has applied for resource. The Administrator would update the request status and the updated status is intimated to requested employee through mail with the help of this system.

Defect Tracking System

Defect Tracking System aims at automating the tasks associated with the processing and tracking of software defect reports. The module includes methods for generation of software problem reports, reporting the problem to responsible person, communicating the solution of the problems and tracking the status of each software problem report. This module aims to provide a centralized control over problem report and communication of problem between the project team mates

Client Information Maintenance

Client Information Maintenance is used for storing the information about client, not only that project signed date, and shipping date is also stored. Payment details of the project are also maintained. This module provides the provision for the administrator to send daily status of the project to the corresponding client.

4.1.4 DATABASE DESIGN

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective of database design is to make the data access easy, inexpensive and flexible to the user. The design of the database is one of the most critical parts of design phase. An elegantly database can play as a strong foundation for the whole system. The details about the data relevant for the system are identified first.

The data types for each data item in the tables are decided. For the optimum design of the database, to have better response time, to have data integrity, to avoid redundancy and for the security of the database all the tables created are normalized. The database design is made up of two levels,

- ❖ Conceptual level
- ❖ Normalization

Normalization

The normalization simplifies the entities, removes the redundancies from the system data and finally builds a data structure, which is both flexible and adaptable to the system. Normalization offers a systematic step-by-step approach towards this goal. The different normal forms applied are given below,

- ❖ First normal form (NF)
- ❖ Second normal form (2NF)
- ❖ Third normal form (3NF)

4.2 TABLE DESIGN

The table design containing Field Name, Data Type, Size of the field, Description and information about keys are represented.

User

The table holds information about the users using this system.

Field Name	Type	Description	Key
userid	Varchar(10)	User Id	Primary Key
username	Varchar(25)	User Name	
UserType	varchar(20)	User Type	
Designation	Varchar(20)	Designation	

Table 4.2.1 User

Project Master

The table holds information about project currently going in the company.

Field Name	Type	Description	Key
ProjectId	Varchar(8)	Project Id	Composite Key
CNo	Varchar(10)	Client Id	Foreign Key + Composite Key
Projectname	Varchar(20)	Project Name	
startdate	DateTime	Project start date	
Estimatedshippingdate	DateTime	ProjectShipping Date	

Table 4.2.2 Project Master

FresherMaster

The table holds information about the freshers who are in probation period.

Field Name	Type	Description	Key
FId	Varchar(10)	Fresher Id	Primary key
FName	Varchar(25)	Fresher Name	
FQualification	Varchar(8)	Fresher Qualification	
FCollegeName	Varchar(30)	Fresher College Name	
Fgender	char(1)	Fresher Gender	
Fdob	Datetime	Fresher Date Of Birth	
Fdoj	Datetime	Fresher Date of Join	
FAge	Int	Fresher Age	
FMobileNo	Int	Fresher Mobile No	
FMailId	Varchar(25)	Fresher Mail Id	
FAddress	Varchar(50)	Fresher Address	

Table 4.2.3 FresherMaster

FresherMark

The transaction table holds information about the marks obtained by the fresher in probation period.

Field Name	Type	Description	Key
FId	Varchar(10)	Fresher ID	Foreign key
Test1	Int	1 st Test Mark	
Test2	Int	2 nd Test Mark	
Test3	Int	3 rd Test Mark	
TotalMarks	Int	Total Marks Obtained	
Average	float	Average Of Marks	
Performance	Varchar(15)	Performance Of Fresher	

Table 4.2.4 FresherMark

ClientMaster

The table holds information about the client who have signed project in the company.

Field Name	Type	Description	Key
CNo	Varchar(10)	Client Id	Primary Key
CName	Varchar(25)	Client Name	
CAddress	Varchar(30)	Client Address	
CPhoneNo	Int	Client Phone Number	
CMailId	Varchar(20)	Client Mail Id	
CFaxNo	Int	Client Fax Number	
CManufacturer	Varchar(25)	Manufacturer of	
CKnownthru	Varchar(15)	Known through	
CNegotiateddate	Datetime	Negotiated Date	
CSigneddate	Datetime	Signed Date	

Table 4.2.5 ClientMaster

ClientPayment

The table holds information about the payments made by clients for their project.

Field Name	Type	Description	Key
CNo	Varchar(10)	Client ID	Foreign Key + Composite Key
ProjectId	varchar(8)	Project ID	Foreign Key + Composite Key
Mode	varchar(10)	Mode Of Payment	
TotalAmount	Int	Total Amount	
PaidAmount	Int	Amount Now Paid	
TotallyPaid	Int	Totally Paid Amount	
Balance	Int	Balance Amount	

Table 4.2.6 ClientPayment

EmployeeProject

The table holds information about the employee role in the project

Field Name	Type	Description	Key
EmpId	Varchar(10)	Employee Id	Composite Key
CNo	varchar(10)	Product ID	Foreign Key + Composite Key
ProjectId	varchar(8)	Project Id	Foreign Key + Composite Key
EmpName	varchar(25)	Employee Name	
EmpRole	varchar(15)	Role Of Employee	

Table 4.2.7 EmployeeProject

ClientStatus

The table holds information about the Daily status of the project which was sent to client.

Field Name	Type	Description	Key
CNo	Varchar(10)	Client ID	Foreignkey+ C.K
ProjectId	varchar(8)	Project Id	Foreignkey + C.K
DailystatusDescription	varchar(400)	Daily Status	
ProblemDescription	varchar(300)	Description	
ProjectStage	varchar(250)	Status Of project	

Table 4.2.8 ClientStatus

ProjectStatus

The table holds information about the status of the projects.

Field Name	Type	Description	Key
ProjectId	Varchar(8)	Project ID	Foreign Key
Status	varchar(15)	Project Status	

Table 4.2.9 ProjectStatus

SPR

The table holds information about the problems.

Field Name	Type	Description	Key
ProjectId	Varchar(8)	Project Id	Composite key + Foreign Key
SPRNumber	Int	SPR Number	Composite key
SPRDate	datetime	SPR Date	
Originator	Varchar(30)	Originator Of SPR	
PdlcPhase	Varchar(10)	Error Detected At	
ErrorPertainTo	Varchar(10)	Error in Which Phase	
DefectCategory	Varchar(4)	Category Of Defect	
DefectSeverity	Varchar(1)	Severity Of Defect	
DefectIntroducedAt	Varchar(2)	Defect Introduced At	
ProblemDescription	Varchar(250)	Description Of Problem	
FileName	Varchar(150)	Name Of File	
Functionality	Varchar(150)	Functionality in which Error Occurred	
DocumentReferred	Varchar(150)	Document Referred	
status	Varchar(10)	Status Of Project	
Designer	Varchar(30)	Designer Of Project	

Table 4.2.10 SPR

SDCR

The table has the details of change made in code/design with respect to Software Problem Report (SPR)

Field Name	Type	Description	Key
ProjectId	Varchar(8)	Project Id	Composite key + Foreign Key
SDCRNumber	Int	SDCR Number	Composite key
SPRNumber	Int	SPR Number	Foreign Key
SDCRDate	DateTime	SDCR Date	
Subject	Varchar(50)	Subject	
EstimatedEffort	Int	Estimated Effort (Days)	
Estimatedstartdate	DateTime	Estimated Start Date	
Estimatedenddate	DateTime	Estimated End Date	
ActualEffort	Int	Actual Effort In Days	
Actualstartdate	DateTime	Actual Start Date	
Actualenddate	DateTime	Actual End Date	
Category Change	Varchar(5)	Category Of Change	
DefectCause	Varchar(2)	Cause For The Defect	
DetailOfChange	Varchar(250)	Details Of Change	

Table 4.2.11 SDCR

EmployeeMaster

The table holds information about the Employees working in the company.

Field Name	Type	Description	Key
EmpId	Varchar(10)	Employee Id	Primary key
EmpName	Varchar(25)	Employee Name	
EmpAge	Varchar(16)	Employee Age	
EmpDOB	DateTime	Employee Date Of Birth	
EmpQualification	Varchar(15)	Employee Qualification	
EmpJoiningDate	DateTime	Employee Joining Date	
Empexperience	Varchar(16)	Employee Experience	
EmpContactNo	Int	Employee Contact Number	
EmpMailId	Varchar(25)	Employee Mail Id	
EmpAddress	Varchar(40)	Employee Address	
EmpCountry	Varchar(25)	Country Of Employee	

Table 4.2.12 EmployeeMaster

Resource

The table holds information about the Employee and requested resource details.

Field Name	Type	Description	Key
ResourceReqNo	Varchar(10)	Resource Request Form No	Primary Key
ResourceId	Varchar(10)	Resource Id	
EmpId	Varchar(10)	Employee Id	Foreign key
EmpMailId	Varchar(10)	Employee Mail Id	
ResourceName	Varchar(10)	Resource Name	
RequestedDate	DateTime	Requested Date	
Purpose	Varchar(15)	Purpose For Resource	
HowMany	Int	How Many Required	

Table 4.2.13 Resource

ResourceStatus

The table holds information about the status of the requested resource.

Field Name	Type	Description	Key
ResourceReqNo	Varchar(10)	Resource Request Form No	Foreign Key
ResourceStatus	Varchar(15)	Resource Status	

Table 4.2.14 ResourceStatus

LeaveMaster

The table holds information about the Employee and requested leave details.

Field Name	Type	Description	Key
LeaveReqNo	Varchar(10)	Leave Request Form No	Primary Key
EmpId	Varchar(10)	Employee Id	Foreign key
EmpMailId	Varchar(10)	Employee Mail Id	
RequestedDate	DateTime	Requested Date	
Purpose	Varchar(15)	Purpose For Leave	
HowManyDays	Int	How Many Days Required	

Table 4.2.15 LeaveMaster

LeaveStatus

The table holds information about the status of the requested leave.

Field Name	Type	Description	Key
LeaveReqNo	Varchar(10)	Leave Request Form No	Foreign Key
LeaveStatus	Varchar(15)	Leave Status	

Table 4.2.16 LeaveStatus

4.3 Data Flow Diagram (DFD)

The data flow diagram is graphical representation which depicts the information regarding the flow of control and the transformation of data from input to output. The dataflow may be used to represent the system or software at any level of abstraction. A level 0 data flow diagram is called the Context Diagram, which represents the entire software element as single bubble with input and output arrows.

LOGIN PROCESSING

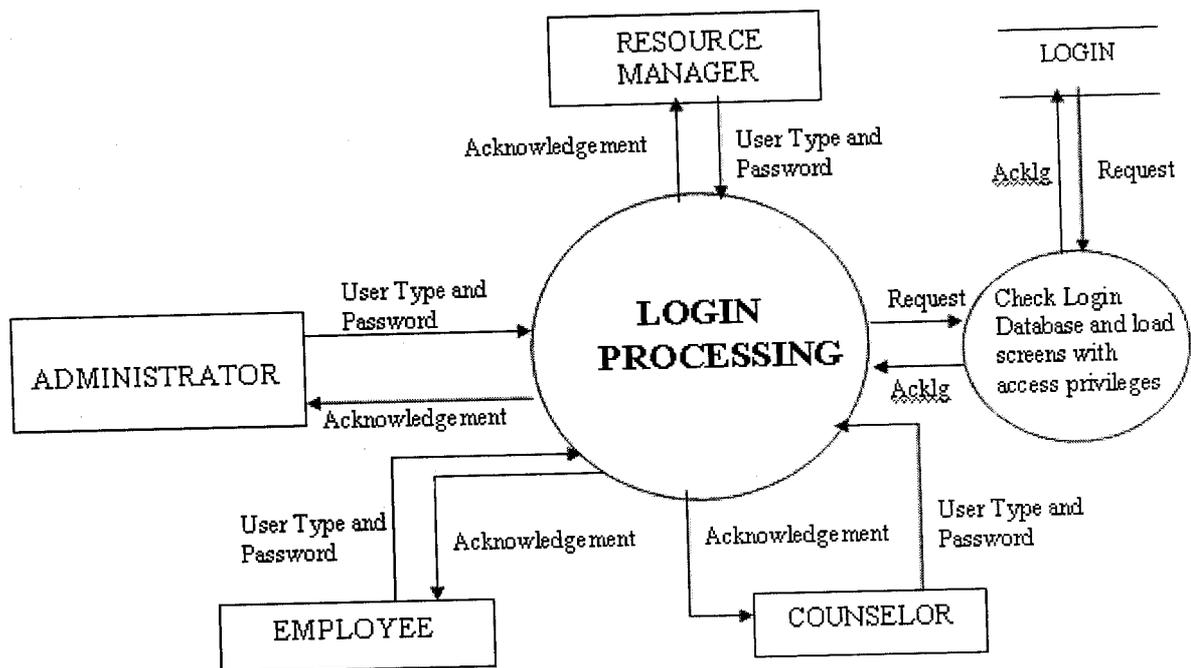


Figure 4.3.1 DFD FOR LOGIN PROCESSING

This module accepts login and password from the user and validates authorization using Login table. If it is a valid user, then the module guides them further to the next module.

FRESHER INFORMATION MAINTENANCE

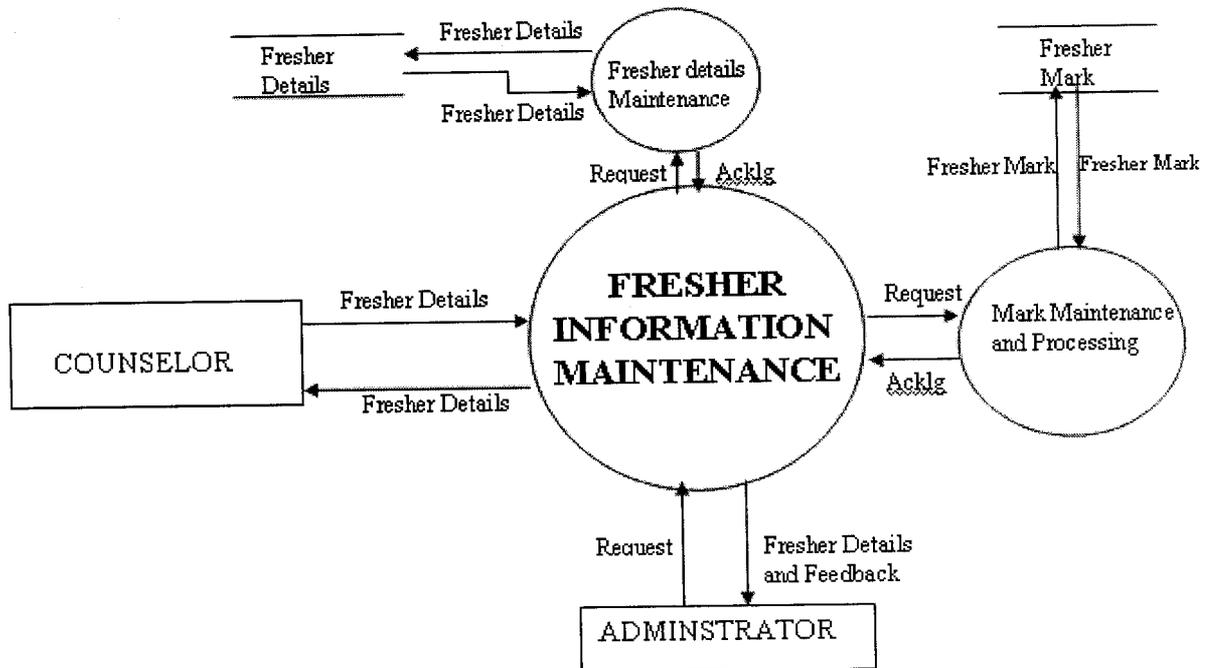


Figure 4.3.2 DFD FOR FRESHER INFORMATION MAINTENANCE

Fresher information module facilitates to store information about recruited persons (Freshers) and their marks obtained in tests conducted during probation period. Based on the marks obtained in tests, freshers are designated as Permanent Employee. Administrator has the provision of seeing the fresher details and their performance.



LEAVE MANAGEMENT

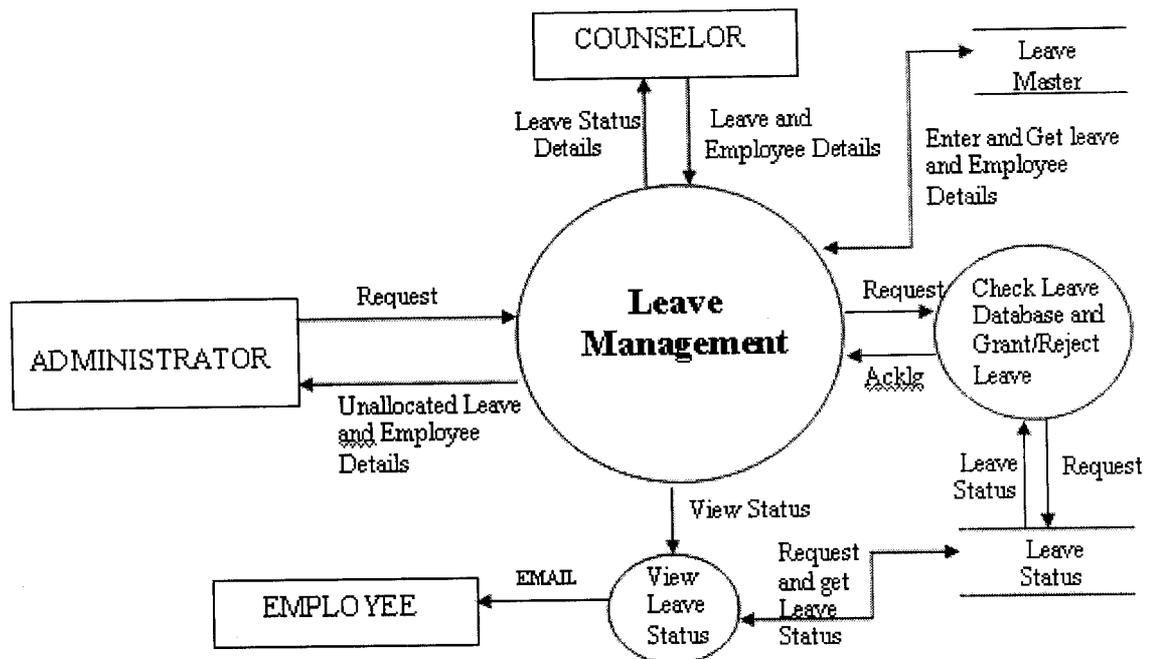


Figure 4.3.3 DFD FOR LEAVE MANAGEMENT

Leave management module facilitates the Resource Manager to enter the employee request for leave and the administrator to grant/reject permission for the employee who has applied for leave. The Administrator would update the request status and the updated status is intimated to requested employee by resource manager through mail with the help of this system.

RESOURCE MANAGEMENT

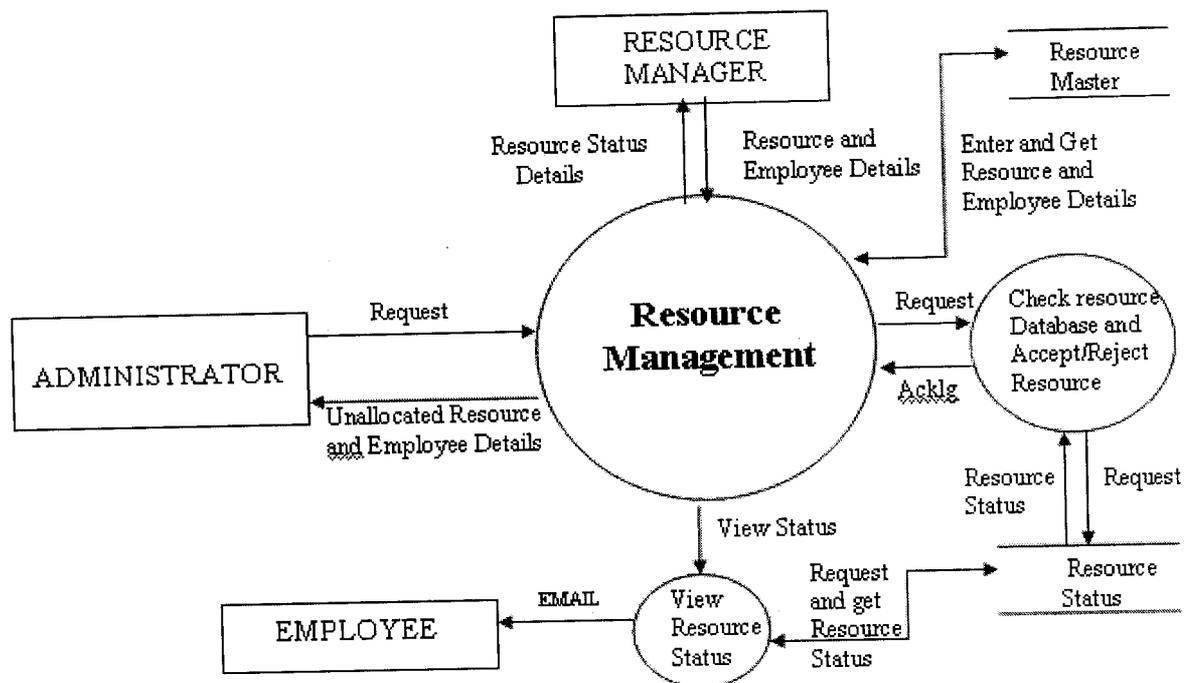


Figure 4.3.4 DFD FOR RESOURCE MANAGEMENT

Resource management module facilitates the Resource Manager to enter the employee request for resource and the administrator to allocate/reject permission for the employee who has applied for resource. The Administrator would update the request status and the updated status is intimated to requested employee by resource manager through mail with the help of this system.

DEFECT TRACKING SYSTEM

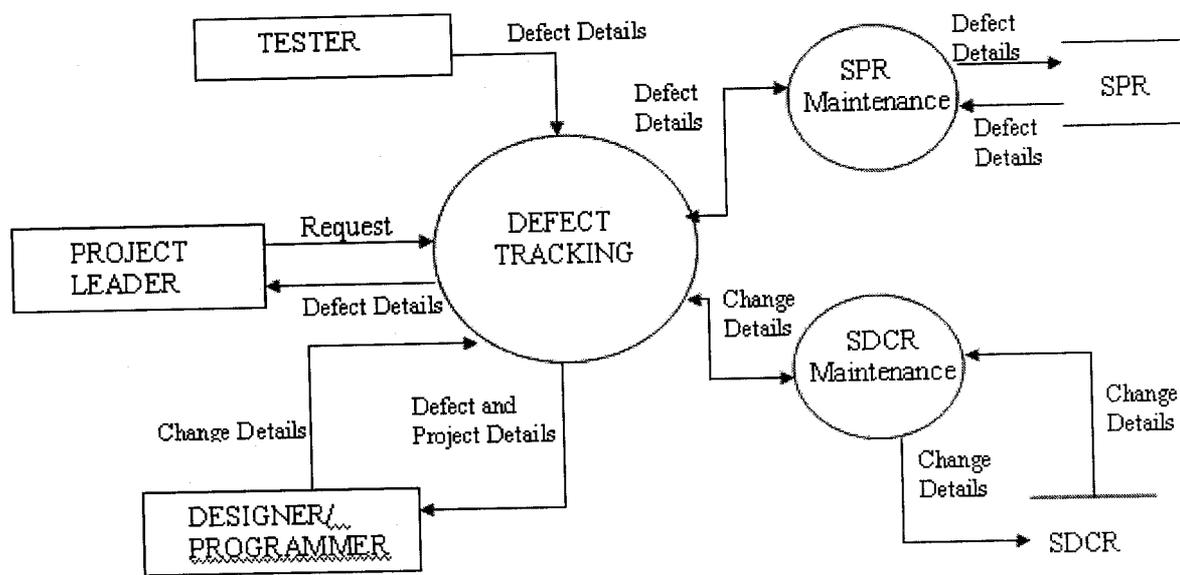


Figure 4.3.5 DFD FOR DEFECT TRACKING SYSTEM

Defect tracking module includes methods for generation of software problem reports, reporting the problem to responsible person, communicating the solution of the problems and tracking the status of each software problem report. This module aims to provide a centralized control over problem report and communication of problem between the project team mates

CLIENT INFORMATION MAINTENANCE

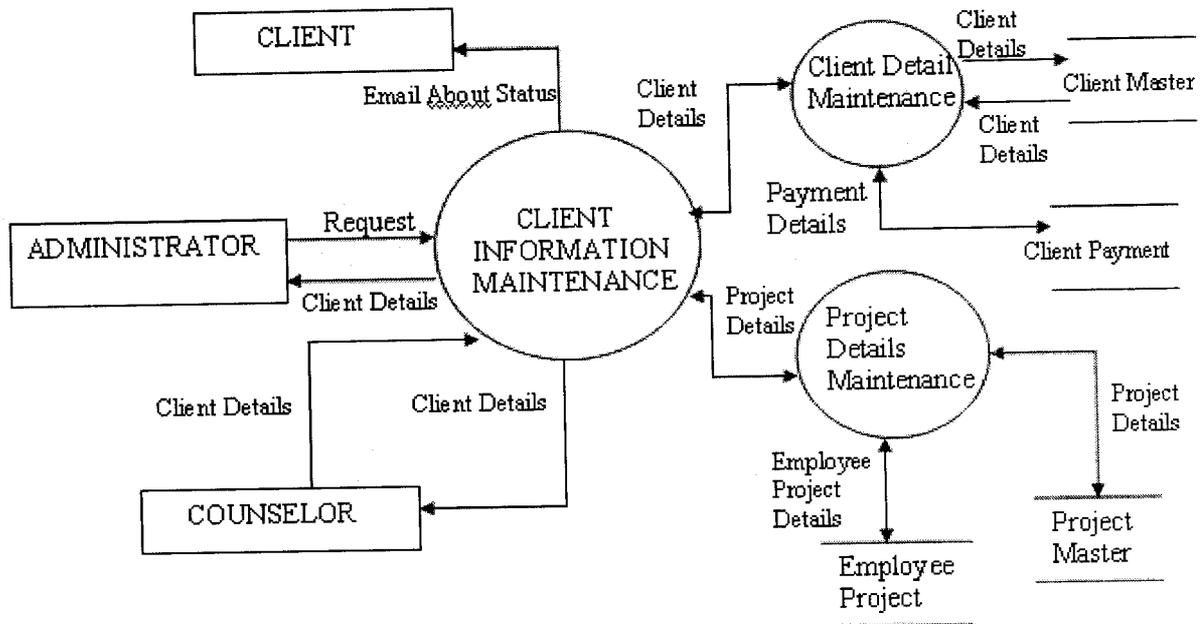


Figure 4.3.6 DFD FOR CLIENT INFORMATION MAINTENANCE

Client Information Maintenance is used for storing the information about client, project details of the particular client were also stored. Payment details of the client are also maintained. This module provides the provision for the administrator to send daily status of the project to the corresponding client.

CHAPTER 5

SYSTEM TESTING AND IMPLEMENTATION

5.1 SYSTEM TESTING

System testing is the most vital activity that has to be enforced in any system development. This could be run parallel during the development phase and after the implementation. The feedback received from this testing was examined carefully for further enhancements. It is the part of testing where the entire application has been tested. This testing is performed with the requirement document as the reference and the goal is to see whether the application meets the requirement.

5.1.1 Unit Testing

Unit testing is the process of testing the system, module by module. It checks for the various inputs and outputs and also checks whether they are required. Using this method gives a clear idea of the bugs occurred.

In Integrated Office Manager, all modules are appropriately validated with the suitable input. Each module is separately verified with their control program in bottom up approach fashion.

The following units were tested independently:

- ❖ Client Information Management Module is checked with all possible input cases.
- ❖ During Resource Allocation, Resource Management module is tested for its reliability with more than one user at a time.

5.1.2 Integration Testing

Integration testing is a systematic technique for constructing the program structure while condition tests to uncover errors associated with interfacing. The following are the various types of integration test done.

Top-Down-Integration

Top-Down-Integration is an incremental approach to construction of program structure. In Integrated Office Manager, modules are integrated by moving downward through control hierarchy, beginning with the main control module (main program).

Bottom-Up-Integration

Bottom-Up-Integration testing as its name implies begins construction and testing with atomic values. In Integrated Office Manager, Bottom-Up-Integration is done in two ways

- ❖ Low-Level modules are combined into clusters and tested
- ❖ Clusters are combined whole moving upward in the program structure.

5.1.3 Validation Testing

Validation testing is that validation succeeds when software functions in a manner that can be reasonably expected by the customer. The most popular validation testing is alpha and the beta testing.

5.1.4 Alpha Testing

Alpha test is done for this project by simulating a controlled environment with a third member not involved in the project precisely not a customer but treated as a customer.

5.1.5 Beta Testing

This is the test conducted at the user environment where the system will survive. It was done by the actual user of the system. This system also examined with this test, by installing it into the organization and user feed back are gathered, according to that this system has been refined.

5.2 SYSTEM IMPLEMENTATION

Implementation is the stage where the theoretical designs are turned into working system. The most crucial stage in achieving success of the new system is in giving confidence to the users that will work efficiently and effectively.

The primary goal of implementation is to write source code to its specification which can be easily verified, and so that debugging, testing, and modification can be eased. The goal can be achieved by making the source code as clear and straight forward as possible. The implementation is the process of converting a new or revised system into operational one. It is the key stage in achieving a new system because it involves a lot of upheaval in the user environment.

During the course of the project various risks like computer failure leading to delay and data loss were identified. Timely measures were taken to minimize them. Similarly, risk of data loss was minimized by taking regular backups. Thus, by eliminating the risk of delay and data loss, the schedule was kept on track. By keeping to schedule, the risks of missing deadlines were reduced.

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

An attempt has been made to computerize the **INTEGRATED OFFICE MANAGER** and is implemented in Honey Bee Technologies, Coimbatore to the satisfaction of the company. Validation has been done when and where it is been required. It produces timely reports from the most compatible centralized database.

The new system is effective in speed of collection of information. Efforts have been put to make sure that the system takes care of almost all the requirements of the company. The system can also be modified and expanded to a greater extent by introducing new entities.

In conclusion, it is note worthy to mention the performance of the computer system, against the manual system. Comparative analysis was carried out and it was found that the new system was successfully working for the test data provided by the authors and hope that the software will be extremely helpful to the company.

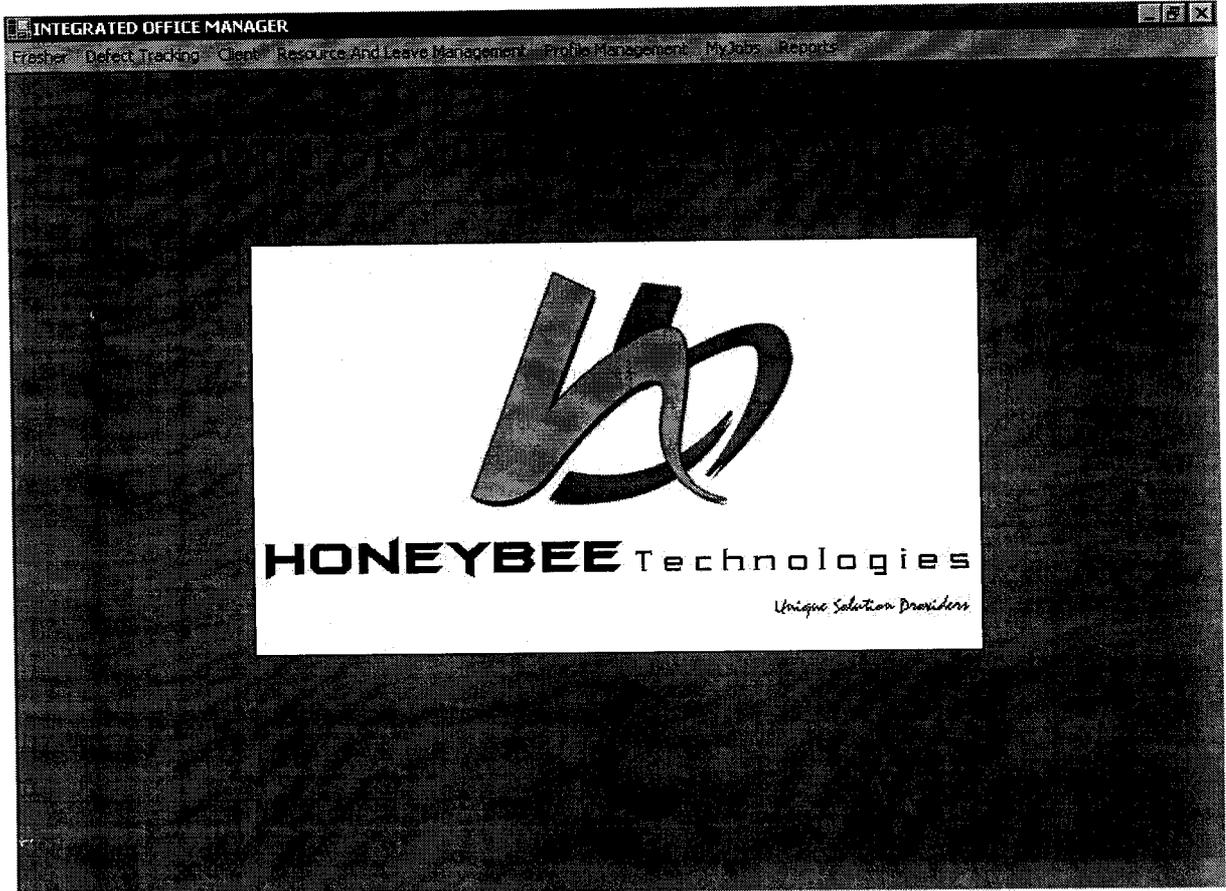
6.2 FUTURE ENHANCEMENT

Definitely, all Software will have a higher version. The system can be adapted for any further development. The system is so flexible to allow any modification need for the further functioning of programs. With a foresight, the feature that could be added for future enhancement is

- ❖ The System could made as web enabled to allow employees to work offsite.

APPENDICES

APPENDIX 1



Integrated Office Manager - Front Screen

INTEGRATED OFFICE MANAGER - [ResourceEntry]
Refresh Defect Tracking Client Resource And Leave Management Profile Management My Jobs Reports

Resource Entry

Requested Date : Resource Req No :

Resource Id :

Resource Name :

Employee Id :

Employee Mail Id :

How Many :

Purpose :

Resource Management – Resource Request Entry

INTEGRATED OFFICE MANAGER - [ViewResourceStatus]

Home | Direct Trading | Clerk | Resource And Leave Management | Profile Management | My Jobs | Reports

From : Saturday, April 15, 2006 To : Tuesday, April 25, 2006 Go

Date	ResourceReq No	ResourceId	ResourceName	EmployeeId	Employee Mail Id	HowMany	Status
4/24/2006	1221	Ram01	Ram	1002	hfhf@honeybee.com	4	Allocated
4/23/2006	200	hdb02	disk	555	fa@honeybee.com	3	Rejected
4/23/2006	555	hda01	SendingMail			2	Not Allocated
4/21/2006	123	hd1				1	Allocated
4/17/2006	143	Ram01				2	Rejected
4/16/2006	1111	Ehernet01				1	Allocated

MAIL SENDING

From :

To :

Subject : your request for resource has been processed and resource have been allocated.

Resource Management – Mail Sending

INTEGRATED OFFICE MANAGER - [LeaveRequestEntry]

Fresher Defect Tracking Client Resource And Leave Management Profile Management MyJobs Reports

Leave Request Entry

Requested Date :	Tuesday , April 25, 2006	Leave Requisition Form No :	143
Employee Id :	emp100		
Employee Mail Id :	kannan@honeybee.com		
How Many (Days) :	2		
Purpose :	Personal Work		

Clear OK Cancel

Leave Management – Leave Request Entry

INTEGRATED OFFICE MANAGER - [NewClientEntry]

File Edit View Help

Project Tracking Client Resource And Leave Management Profile Management My1005 Reports

New Client Entry

Client No	car001	Project ID	Inv123
Client Name	Caritor	Project Name	IntegratedOfficeManager
Address	16, 2nd kaveri street, solinganallur road, Chennai.	Start Date	Thursday , April 27, 2006
Phone Number	2645034	Estimated Shipping Date	Friday , August 25, 2006
Mail ID	abcd@caritor.com	Mode Of Payment	DD DD No 9875645
FAX Number	2435454	Total Amount	15000 Bank State Bank
Manufactures of	EmbeddedSystems	Amount Now Paid	6000 Branch TownHall
How Do U Know Us	Advertisements	Totally Paid	6000
Negotiated Date	Saturday , March 25, 2006	Balance Amount	9000
Signed Date	Tuesday , April 25, 2006		

CLEAR OK CANCEL

Client Information Module- New Client Entry

INTEGRATED OFFICE MANAGER - [SPR (Software Problem Raise)]

File Defect Tracking Client Resource And Leave Management Profile Management My Jobs Reports

SOFTWARE PROBLEM REPORT

Project Id	<input type="text" value="Inv123"/>	Project Name	<input type="text" value="IntegratedOfficeManager"/>	Date	<input type="text" value="Thursday , April 27, 2006"/>
Originator	<input type="text" value="Kannan"/>	SPR #	<input type="text" value="49"/>	SPR Date	<input type="text" value="Thursday , June 22, 2006"/>

FDLC Phase	<input type="text" value="Design"/>	Error Pertaining to	<input type="text" value="REQ"/>	Category	<input type="text" value="CAT2"/>
Defect Introduced At	<input type="text" value="PM"/>	Defect Severity	<input type="text" value="2"/>	Defect Category	<input type="text" value="DE"/>
Problem Description	<input type="text" value="Memory Model hangs in Test case 1472"/>				

Defect Details		File Name	Functionality	Document Referred
	<input type="text" value="display.cpp"/> <input type="text" value="cmecpy.cpp"/>	<input type="text" value="convert()"/> <input type="text" value="display()"/>	<input type="text" value="invi.doc"/>	

Project Leader	<input type="text" value="Thamil"/>	Status	<input type="text" value="Open"/>	Designer	<input type="text" value="Senthamil Kannan"/>
----------------	-------------------------------------	--------	-----------------------------------	----------	---

Defect Tracking System- Software Problem Report

INTEGRATED OFFICE MANAGER - [SDCR (Software Design / Change Report Raise)]

Project Id: Inv123 Project Name: IntegratedOfficeManager Date: Thursday, April 27, 2006

Originator: Kannan SPR #: 49 SPR Date: Thursday, June 22, 2006

PDLC Phase: Design Error Pertaining to: REQ Category: CAT2

Defect Introduced At: PM Defect Severity: 2 Defect Category: DE

Problem Description: Memory Model hangs in Test case 1472.

Defect Details

File Name: display.cpp cmecpy.cpp Functionality: convert() display() Document Referred: invi.doc

SDCR #: 15 SDCR Date: Tuesday, April 25, 2006

Estimated Efforts (Days): 3 Estimated Start Date: Wednesday, June 28, 2006 Estimated End Date: Friday, June 30, 2006

Actual Efforts (Days): 2 Actual Start Date: Wednesday, June 28, 2006 Actual End Date: Thursday, June 29, 2006

Subject: Memory Model Si Defect Cause: Cd Category Of Change: CAT2

Detail Of Change: Memory Model Size changed

ASSIGN CANCEL

Defect Tracking System- Software Design/Code Change Report

INTEGRATED OFFICE MANAGER - [FresherFeedbackEntry]

Fresher Defect Tracking Client Resource And Leave Management Profile Management MyJobs Reports

Fresher Feedback

FeedBack Date :	Tuesday , April 25, 2006	FeedBack For (Staff) :	R.AnanthaKannan
Subject (Class) :	OperatingSystem	Batch ID (Fresher) :	Op001

Finality Of The Staff :	Satisfied
Regularity Of The Staff :	Satisfied
Teaching Of The Staff :	Neutral
Practical Guidance :	Highly Satisfied
Doubt Clarifications :	DisSatisfied
Student-Staff Relationship :	Satisfied
Class Environment :	Highly Satisfied

Clear Assign Cancel

Defect Tracking System- Software Design/Code Change Report

INTEGRATED OFFICE MANAGER - [ViewFeedBack]

Fresher Defect Tracking Client Resource And Leave Management Profile Management My Jobs Reports

FeedBack For (Staff) : R.AnanthaKannan Month : April Year : 2006 [Go]

BatchId	Course/Joined	Punctuality	Regularity	Teaching	PracticalGuidance	DoubtClarification	Relationship	Environment
143	Embedded Sya...	Satisfied	Neutral	Neutral	Neutral	Neutral	Neutral	Neutral
143	C,C++	Neutral	Neutral	Neutral	Neutral	Neutral	Neutral	Neutral
Op001	OperatinSystem	Satisfied	Satisfied	Neutral	Highly Satisfied	DisSatisfied	Satisfied	Highly Satisfied

Fresher Information Maintenance - Administrator Viewing Feedback

The screenshot shows a web application window titled "INTEGRATED OFFICE MANAGER - [AdminLeaveStatus]". The navigation menu includes "Fresher", "Defect Tracking", "Client", "Resource And Leave Management", "Profile Management", "MyJobs", and "Reports". The date range is set from "Saturday, April 22, 2006" to "Thursday, April 27, 2006". A table displays the following data:

Requested Date	LeaveReqFormNo	EmployeeId	EmployeeMailId	Purpose	HowMany
4/26/2006	143	emp1001	thamil@honeybee.com	PersonalWork	3
4/26/2006	100	emp101	kannan@honeybee.com	PersonalWork	2
4/24/2006	1233	emp001	abc@honeybee.com	Not Well	1
4/23/2006	101	emp002	xyz@honeybee.com	PersonalWork	1

Leave Management – Administrator Viewing Leave Status

REFERENCES

Books

1. David Solomon, David Solomon's (2001), “ **Sams Teach Yourself Visual Basic Dot Net in 21 Days**”, Crimson Publications.
2. RichardHundhausen,StevenBorg,(2002), “ **Programming ADO.NET** ”, Wiley Computer Publishing
3. Elias M.Awad, (1997), “ **System Analysis and Design**”, Golgotha Publications.
4. Roger S Pressman (1997), “ **Software Engineering**”, Tata McGraw-Hill Publishing Company.
- 5.Abraham Silberschatz, Korth, Sudarchan,(2002), “ **Database System Concepts** ”, McGraw-Hill International Edition.

Websites

1. <http://www.en911.com/ebook/1-1.htm>
2. <http://samples.gotdotnet.com/QuickStart/aspplus/default.aspx?url=/quickstart/aspplus/doc/default.aspx>
3. <http://www.CrystalDecisions.com>