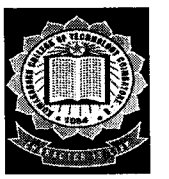

PROJECT ENTERPRISE TRACKING SYSTEM

By

C.RAMACHANDRAN
(Registration Number: 71205621034)

Of

KUMARAGURU COLLEGE OF TECHNOLOGY
COIMBATORE - 641 006

A PROJECT REPORT
Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements
For the award of the degree

Of

MASTER OF COMPUTER APPLICATIONS

ANNA UNIVERSITY
Chennai 600 025

JULY 2008

BONAFIDE CERTIFICATE

Certified that this project report titled **PROJECT ENTERPRISE TRACKING SYSTEM** is the bonafide work of **Mr.C.RAMACHANDRAN**(Registration Number: 71205621034) 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.

Supervisor

Head of the Department

Submitted to Project and Viva Examination held on 01.07.2008

1/7/08
INTERNALEXAMINER

1/7/08
EXTERNAL EXAMINER

B-SQUARE
Software Solutions

30th May 08

PROJECT COMPLETION CERTIFICATE

This is to certify that **Mr. C.Ramachandran (Reg. no 71205621034)** of Kumraguru College of Technology, has done his project "**PROJECT ENTERPRISE TRACKING SYSTEM"** at our organization from 20th December 07 to 30th May 08. During the project, he has successfully completed the project to our satisfaction.

We wish him all the very best.

For B-SQUARE Software Solutions,

Manager - Human Resources

New No: 15, Old No: 13, Subramanium Salai, West Saidapet, Chennai – 15
Website: www.bsquareinfo.com Email: balu.k@bsquareinfo.com
Contact No: 044 - 64186705 Mobile No: +91 - 98417 72385

ABSTRACT

"**Project Enterprise Tracking System**" is a tool required to keep in track the whole process of the project and the programmers involved. This helps the project manager to access the time taken for the project with module wise break up and nature of the job, which needs attention. This also helps the project manager to identify the assets and weakness of the Programmer's to give them training by using knowledge sharing module where they lack experience.

In order to do any project to be efficiently and successfully developed it's always a right mechanism to divide the project into number of modules. This also helps in a clear and better understanding of the flow of the project. Knowledge sharing in this system is used to help the employees. Sharing information in organization wide can lead to effective time saving for other members of the organization when they are in need of the information. Such sharing of acquired knowledge will also keep the overall staff equipped with knowledge above an average user.

The usage of Internet is growing day-by-day and so are the vulnerabilities in it.

- This project deals with the status of the day to day important happenings of all employees.

- This project keeps track of all activities, which helps in managing the Organization effectively.

- It provides different subsystem for different kinds of activities.

- The main objective of the PETS shows the actual flow of project development process and total utilization of employees and their effort towards the completion of the project

ACKNOWLEDGEMENT

I would like to take this opportunity to say thank you to the people who have helped to make this project.

First I would like to express my heartfelt thanks to our principal **Dr.Joseph V.Thanikal**, B.E., M.E., Ph.D., PDF., CEPIT., for having given me the opportunity to do this project.

I express gratitude and thanks to **Dr.S.Thangasamy**, Ph.D., Dean, Computer Science and Engineering, for providing moral support towards this project work.

I express my deep sense of gratitude and profound thanks to **Dr.M.Gururajan**, Ph.D., HOD, Computer Application, for providing moral support towards this project work.

I tender my special thanks to **Ms. V.Geetha,** MCA.,MPhil Assistant Professor and Project Coordinator, Department of Computer Applications for her constant encouragement.

I sincerely express my humble gratitude to **Ms.R.Vanitha**, M.C.A., MPhil, Lecturer, Department of Computer Application, who has been my guide with valuable and holistic suggestions and extended kind of operation and encouragement to make this project.

I also express my deep sense of gratitude and profound thanks to **Mr. R.Sathiesh** Project Manager, B-SQUARE Software Solutions, who has been my guide and gave valuable suggestions and encouragement.

I acknowledge my hearty thanks to all my beloved friends on their valuable co-operation in the proceeding of my work.

I wish to credit my special accordance to my parents for their encouragement and prayers to have a successful project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 ORGANIZATION PROFILE

Bsquare Software Solution is the Software and IT services providing company which gathers the industry needs by representing major Technologies in the development and services both in the National and International level as well as software, hardware, voice and non-voice support.

Bsquare deliver software products, ranging from enterprise project to Internet applications with enabled security.

The company offer wide range of significant solutions for our clients required according to their need:

- Software design and development
- Web Services
- Project management
- Quality assurance and quality control
- System Integration
- Testing Services

Skill set:

We can proudly claim that our business is everything from software development and customer support. Our company successfully delivered products using this cumulative skill set:

Category	Item	Year of most recent use	Comment
Os	NT//2000/XP	2007	
Languages & technologies	JAVA, J2EE	2007	Used mostly server side oriented. JDBC, Hibernate, JSP, JMS. Preferred platform.
	JAVA SCRIPT/DHTML, PHP	2007	Web designing in both static & dynamic criteria
	VB,C#/ ASP.NET/ASPOSE	2007	Preferred platform
	AJAX	2007	
Databases	MY SQL	2007	
	SQL SERVER	2007	
	ORACLE	2006	
Application Servers			
	IIS	2007	
	APACHE	2006	

Table 1.1.1 Skill Set

There is a trend among software engineers to move an organization to another, never staying for the lifetime of a product, never caring about what happens after they leave. Then software projects fail and the money invested disappear in a pit.

This is not the case with Bsquare Software Solution. The company is here to stay, and the employee turnover is zero.

The company believe that excellent products are delivered by people who care about their work. The company aim to deliver the best possible software and be proud of it. The company favor technical decisions as opposed to political ones. Our relationship with the customer grows naturally with the successful delivery of each new product.

The company engineers worked in the software industry's best practices of managed software development. With each project our internal coherence is growing. We are developing and introducing best practices, all the time, in a controlled way.

The appropriate tools must support every development practice. We use:

- A centralized bug database (Test Track) to keep track of tasks and defects.
- Subversion Source control repository, hosted on a RAID server
- Standard internal code style, consistent across languages
- Standard internal naming style specialized for programming constructs, file names and path.
- Standard projects organization, deliverables, version naming.
- Standard internal code versioning management, ranging from patch management, code branches management, database structure versioning, releases organization.
- Standard software tools, available on every machine.
- Standard backup practice

1.2 PROBLEM DESCRIPTION

The **"Project Enterprise Tracking System"** provides solution by making available the details of the activities carried by the organization on time to take decision effectively, easily and timely. This system deals with the storage of the complete project details and the details of all employees.

This system keeps account of all activities, which helps in managing the organizations effectively. It provides different subsystems for different kinds of activities. It has some security features to safeguard the details stored in this system.

This system provides a login for the client also using which the client can login into this web application and view their corresponding project status. Also this system includes the knowledge management to share the information.

Sharing this organization wide can lead to effective time saving for other members of the organization when they are in need of the information. Such sharing of acquired knowledge will also keep the overall employee equipped with knowledge above an average user.

The system is an Internet based application that can be accessed throughout the organization or a specified group/Dept. This system can be used as a knowledge /information mgmt system for the organization. Employees logging should be able to upload any kind of technical information. Employees logging in may also access/search any information put up by others. It should facilitate knowledge sharing from the grass root level like project teams to the entire organization.

CHAPTER 2
SYSTEM ANALYSIS

2.1 EXISTING SYSTEM ARCHITECTURE

The existing system is very large to maintain manually. Developers or Project Leaders of different knowledge maintain it in written documents. There are a variety of problems involved in the existing system while performing a detailed study.

The Problem in the existing system is that it is not able to track the total time spent by the employees of the organizations. There is an informal communication happens between hierarchies inside the organization in this regard. Even though it has been kept in written form it is very large and it is very difficult to manage. A utility that comprises of the following advantages will be an excellent answer for the above issues:

- Significant ROI is possible through deploying clock wares time and work Flowing systems.
- Human Resources can track and monitor all levels of categories with an in limited number of balances.
- Line Managers can use extensive workflow features to ensure that time taken to complete the projects. Submitted, approved and processed on time.
- Senior Management can benefit from reporting tools that analyze major projects, clients and Employees.
- Security from role-based and other key security features ensures the system supports an organization structure and policies.
- Ease of Use ensures simple system administration and minimal training requirements.
- A qualitative work cannot be achieved.

2.2 PROPOSED SYSTEM ARCHITECTURE

To overcome the problem faced in the existing system, development of the new **"Project Enterprise Tracking System"** in the latest application software is decided. The "**Project Enterprise Tracking System"** developed in .NET with HTML, DHTML, besides being user friendly, as nearly been successful in overcoming the drawbacks of the current system. Accepting passwords provides security features. Necessary validations are performed on the data entered. Thus the proposed ensures that it will meet the requirements of the user.

By changing the manual management activities to an automated **"Venture Enterprise Tracking System"** to an automated system, the following advantages were identified.

- Reduction of manual work
- Flexible and easy to use
- User Friendly.
- Efficient
- Reliability
- Simple and auto format

2.3 USER INTERFACE REQUIREMENTS

The "**Project Enterprise Tracking System**" has the following modules

- **Login Module**
- **Project List Module**
- **Department Details Module**
- **Employee List Module**
- **Designation Details Module**
- **Client List Module**
- **Knowledge Management Module**

LOGIN MODULE:

This module is considered to be the most important part of the project since it deals with the authentication and security aspects. There rare three types of Login modules:

- Administrator Login
- Client Login
- Employee Login

Admin Login:

The Administrator can login into this system through his/her corresponding login id and password. When you enter into this web application in administrator login mode you can view and change the Project details, Department, team, an employee details and client details and the PETS.

Client Login:

The Client can login into this system through their corresponding login id and password. When they enter into this web application in Client login mode, the client can view the status of their project details.

Employee Login:

Employees those who are working in projects can also enter using their corresponding id & password, so that they can know their own project status.

PROJECT LIST MODULE:

This module is used to display the information about the current project. It displays project list, project list includes project id, project name, project type, project type id, department id, team id, client id, start date, end date and status of the project.

DEPARTMENT DETAILS MODULES:

This module is used to display the information about the Departments of the organization. This module displays Departments details, includes Departments id, Departments name. Here the options are given to view, modify and delete the Department details.

EMPLOYEE LIST MODULE:

This module is used to display the information about the Employee details. Here the options are given to view, modify and delete the Employee Records. Here the Privileges were provided at the right side of the page for the user's convenience.

DESIGNATION DETAILS MODULE:

This module is used to display the information about the Designation details. This module displays Designation list, includes Designation id, Designation name. It is used to know the Designation used in the organization.

CLIENT LIST MODULE:

This module is used to display the information about the Client details. It plays Client list, Client list includes Client id, Client name, Client Address. Also give feedback about the project by using the feedback form.

KNOWLEDGE MANAGEMENT MODULE:

With this module, the users would be able to share the documents with other users. The interface to create content online, upload new document/edit existing document, and to fill document information like keywords, title, etc. would be dealt here. In this knowledge management use help menu. It shall contain necessary tutorial and context-linked help files for the new users.

CHAPTER 3
DEVELOPMENT ENVIRONMENT

3.1 HARDWARE REQUIREMENTS

PROCESSOR	:	PENTIUM IV
RAM	:	1 GB
MONITOR	:	17" COLOR
HARD DISK	:	120 GB
FLOPPY DRIVE	:	1.44 MB
CDDRIVE	:	LG 52X
KEYBOARD	:	MULTIMEDIA
MOUSE	:	LOGITECH

3.2 SOFTWARE REQUIREMENTS

OPERATING SYSTEM	:	WINDOW XP
ENVIRONMENT	:	VISUAL STUDIO .NET
PLATFORM	:	ASP.NET
DATABASE	:	SQL SERVER
SCRIPTING LANGUAGE	:	C#

3.3 SOFTWARE DESCRIPTION

Visual Studio .NET

Visual Studio .NET is the rapid application development tool for C#. Visual Studio.NET offers complete integration with ASP.NET and enables to drag and drop

server controls and design Web Forms as they should appear when user views them. Some of the other advantages of creating C# applications in Visual Studio.NET are

- Visual Studio .NET is a Rapid Application (RAD) tool. Instead of adding each control to the Web Form programmatically, it helps to add these controls by using Toolbox, saving programming efforts.

Visual Studio .NET supports custom and composite controls. Can create custom controls that encapsulate a common functionality that might need to use in a number of applications.

Visual studio .NET does a wonderful job of simplifying the creation and consumption of Web Services. Much of the programmer-friendly stuff (creating all the XML-based documents) happens automatically, with out much effort on the programmer's side. Attribute-based programming is a powerful concept that enables Visual Studio .NET to automate a lot of programmer-unfriendly tasks.

.NET FRAMEWORK:

- The .NET Framework is the infrastructure for the new Microsoft .NET Platform.

- The .NET Framework is a common environment for building, deploying, and running Web applications and Web services.

- The .NET Framework contains a common language runtime and common class libraries –like ADO .NET, ASP.NET and Windows Forms to provide advanced standard services that can be integrated into a variety of computer systems.

- The .NET Framework provides a feature-rich application environment, simplified development and easy integration between a numbers of different developments languages.

- The .NET Framework is language neutral. Currently it supports C++, C#, Visual Basic, and Jscript (The modern version of JAVASCRIPT).

- Microsoft's Visual Studio.NET is a common development environment for the new .NET Framework.

ASP.NET

Typically this is accomplished by overriding the proper event in the server control class, and manually adding the extended functionality. For example, to provide cell-highlighting with the Data Grid you would override the Data Grid's Item Created event and set the Attributes property of the Data Grid Item's various Cells so that their on mouse over and on mouse out client-side events triggered the desired color highlighting. The disadvantage of this approach is that it requires a bit of coding each and every time you want to use the functionality. That is, if you have a number of ASP.NET Web pages where you want to use this enhanced Data Grid, you have to make these additions by hand on each page. While this is not terribly difficult, it can get time-consuming, and tedious, especially if after making these changes you decide to not do the highlighting after all.

A less tedious approach is to create a new ASP.NET server control, one that takes all of the functionality of the base server control, and extends it by adding on this one little bit of extra functionality. With this approach, you write the code once, compile the new server control, and then can reuse the server control on any number of ASP.NET Web pages without adding anything extra.

In this article we'll examine how to improve the functionality of the Calendar Web control by adding a property called Hover Color. If this property is set, whenever the user moves their mouse over a day in the Calendar control, that day becomes

"highlighted" with the specified color. Our discussion of adding this functionality will look at inheritance, adding properties to Web controls, and a brief look at maintaining property state in Web controls across post backs.

Adding Functionality with Inheritance

One of the more powerful features of object-oriented programming lies in inheritance.The concept of inheritance stems from the fact that complex objects are extensions of simple ones. For example, the Hummer is an extension of a standard SUV; a standard SUV is an extension of a standard pickup truck; a pickup truck is an extension of a standard automobile. Inheritance makes design and implementation much easier because it gives designers and engineers a starting point. That is, when building a Hummer, the engineers already have the base components done for them: they don't need to worry about the dynamics of the internal combustion engine, or the computer system that works the fuel injector.

Since .NET is a fully object-orientated platform, it should come as no surprise that the ASP.NET object model is ripe with possibilities for extension via inheritance. Just look at the provided ASP.NET server controls. All server controls are extended, either directly or indirectly, from the System .Web .UI.Control class. This class provides the base functionality all ASP.NET server controls need to perform in the life cycle of an ASP.NET Web page. This class is refined further with the System.Web.UI .WebControls.WebControl class, which adds style information, such as background color, fore color, font information, and such, to the base properties defined by the Control class.

In a similar vein, we can take an existing ASP.NET server control class, such as the Calendar or Data Grid or Textbox, and create a new class derived from this base class. Automatically, without having to write a single line of code, this inheritance gives our new server control all of the functionality provided in the base class. Therefore, to add new functionality we only need to write the code necessary for said new functionality

- and with that little addition, we'll have a fully robust, useable Web control similar to the base control in all ways except now that its functionality has been extended.

Step 1: Initial Request

The web client requests the main application URL, shown in **Figure**

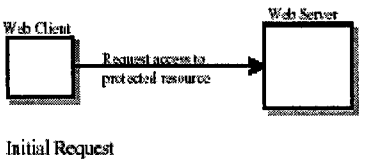

Figure 3.2.1 Initial Request

Since the client has not yet authenticated itself to the application environment, the server responsible for delivering the web portion of the application detects this and invokes the appropriate authentication mechanism for this resource.

Step 2: Initial Authentication

The web server returns a form that the web client uses to collect authentication data (for example, username and password) from the user. The web client forwards the authentication data to the web server, where it is validated by the web server, as shown.

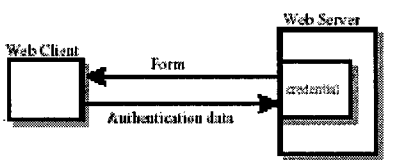

Figure 3.2.2 Initial Authentication

The validation mechanism may be local to the server, or it may leverage the underlying security services. On the basis of the validation, the web server sets a credential for the user.

Step 3: URL Authorization

The credential is used for future determinations of whether the user is authorized to access restricted resources it may request. The web server consults the security policy (derived from the deployment descriptor) associated with the web resource to determine the security roles that are permitted access to the resource. The web container then tests the user's credential against each role to determine if it can map the user to the role.

The web server's evaluation stops with an "is authorized" outcome when the web server is able to map the user to a role. A "not authorized" outcome is reached if the web server is unable to map the user to any of the permitted roles.

Step 4: Fulfilling the Original Request

If the user is authorized, the web server returns the result of the original URL request, as shown.

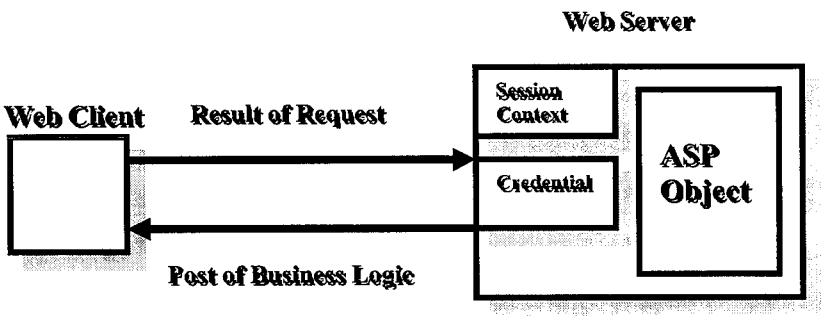

Figure 3.2.3 Fullfill the original Request

Summary

While the .NET platform provides a powerful general-purpose foundation for many types of applications, it does meet all of the needs of web development teams. Achieving productivity in web development requires a specialized development platform. Such a platform must be tailored to the skills and needs of the developers building the

applications. It must support the unique requirements of web development by providing specialized services for Web applications. And lastly, it must meet the needs of IT management, enabling the IT organization to support the business through the timely and cost-effective delivery of new solutions.

STRUCTURED QUERY LANGUAGE (SQL):

SQL (pronounced SEQUEL) is the programming language that defines and manipulates the database. SQL databases are relational databases, which mean data is stored in a set of simple relations. A database can have one or more tables. Each table has columns and rows. Oracle stores each row of a database table containing data for less than 256 columns as one or more row pieces. A table that has an employee database, for example, can have a column called employee number and each row in that column is an employee's employee number.

We can define and manipulate data in a table with SQL statements. We use SQL's data definition language (DDL) statements to set up the data. DDL statements include statements for creating and altering databases and tables.

We can update, delete, or retrieve data in a table with SQL's data manipulation language (DML). DML statements include statements to alter and fetch data. The most common SQL statement is the `SELECT` statement, which retrieves data from the database.

In addition to SQL statements, the Oracle server has a procedural language called PL/SQL. PL/SQL enables programmers to program SQL statements. It lets you control the flow of a SQL program, use variables, and write error-handling procedures.

SQL Statements:

All operations on the information in an Oracle database are performed using SQL **statements**. A SQL statement is a string of SQL text that is given to Oracle to execute. A statement must be the equivalent of a complete SQL **sentence**, as in:

SELECT ename, deptno FROM emp;

Only a complete SQL statement can be executed, whereas a **sentence fragment**, such as the following, generates an error indicating that more text is required before a SQL statement can run:

SELECT ename

A SQL statement can be thought of as a very simple, but powerful, computer program or instruction. SQL statements are divided into the following categories:

- Data definition language (DDL) statements
- Data manipulation language (DML) statements
- Transaction control statements
- Session control statements
- System control statements
- Embedded SQL statements

Data Definition Language (DDL) Statements:

Data definition language statements define, maintain, and drop schema objects when they are no longer needed. DDL statements also include statements that permit a user to grant other users the **privileges**, or rights, to access the database and specific objects within the database.

Data Manipulation Language (DML) Statements:

Data manipulation language statements manipulate the database's data. For example, querying, inserting, updating, and deleting rows of a table are all DML operations. Locking a table or view and examining the execution plan of an SQL statement are also DML operations.

Transaction Control Statements:

Transaction control statements manage the changes made by DML statements. They enable the user or application developer to group changes into logical transactions. Examples include `COMMIT`, `ROLLBACK`, and `SAVEPOINT`.

Session Control Statements:

Session control statements let a user control the properties of his current session, including enabling and disabling roles and changing language settings. The two session control statements are `ALTER SESSION` and `SET ROLE`.

System Control Statements:

System control statements change the properties of the Oracle server instance. The only system control statement is `ALTER SYSTEM`. It lets you change such settings as the minimum number of shared servers, to kill a session, and to perform other tasks.

Embedded SQL Statements:

Embedded SQL statements incorporate DDL, DML, and transaction control statements in a procedural language program (such as those used with the Oracle precompilers). Examples include `OPEN`, `CLOSE`, `FETCH`, and `EXECUTE`.

CHAPTER 4
DESIGN AND DEVELOPMENT

4.1 DATA DESIGN

The database design is a must for any application developed especially more for the data store projects. The database is normalized so as to avoid redundant data. Normalization reduces the wastage of valuable memory space. The database system must give assurance for security of information, despite the system crashes due to unauthorized access.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general theme behind a database is to integrate all the information. The general objective of database design is to make the data access easy, inexpensive and flexible to the user.

The main objectives of designing a database are:

- Data integration
- Data integrity
- Data independence

4.1.1 TABLE RELATIONSHIP

1. Login

Field Name	Data Type	Constraints	Description
Login ID	VARCHAR (20)	Primary Key	Unique ID of The User
Password	VARCHAR (10)	Not Null	Secured password for the User
Mode	VARCHAR (15)	Not Null	User Mode

Table 4.1.1.1 Login Table

2. Project List

Field Name	Data Type	Constraints	Description
Project ID	VARCHAR (10)	Primary Key	Unique ID of The Project
Project Name	VARCHAR (20)	Not Null	Name of The Project
Department ID	VARCHAR (10)	Not Null	Unique ID of The Department
Team ID	VARCHAR (10)	Not Null	Unique ID of The Team
Client ID	VARCHAR (10)	Not Null	Unique ID of The Client
Start Date	SMALLDATETIME	Not Null	Project Start Date
End Date	SMALLDATETIME	Not Null	Project End Date
Status	VARCHAR (30)	Not Null	Status of the Project

Table 4.1.1.2 Project List Table

3. Departments

Field Name	Data Type	Constraints	Description
Department ID	VARCHAR (10)	Primary Key	Unique ID of The Department
Department Name	VARCHAR (20)	Not Null	Name of The Department

Table 4.1.1.3 Departments Table

4. Teams

Field Name	Data Type	Constraints	Description
Team ID	VARCHAR (10)	Primary Key	Unique ID of The Team
Team Name	VARCHAR (20)	Not Null	Name of The Team
Department ID	VARCHAR (10)	Not Null	Unique ID of The Department

Table 4.1.1.4 Team Table

5. Employee

Field Name	Data Type	Constraints	Description
Emp ID	VARCHAR (10)	Primary Key	Unique ID of The Employee
First name	VARCHAR (15)	Not Null	First Name of The Employee
Middle Name	VARCHAR (15)	Not Null	Middle Name of The Employee
Last Name	VARCHAR (15)	Not Null	Last Name of The Employee
DOB	SMALLDATETIME	Not Null	Date of Birth of The Employee
DOJ	SMALLDATETIME	Not Null	Joining Date of The Employee
Designation ID	VARCHAR (10)	Not Null	Unique ID of The Designation
Phone No	NUMERIC (9)	Not Null	PhoneNumber of The Employee
Mobile No	NUMERIC (9)	Not Null	MobileNumberof The Employee
Mail ID	VARCHAR (30)	Not Null	Email Address of The Employee
Address	VARCHAR (50)	Not Null	Address of The Employee
Blood Group	VARCHAR (10)	Not Null	Blood Group of The Employee

Table 4.1.1.5 Employee Table

6. Employee Education

Field Name	Data Type	Constraints	Description
Emp ID	VARCHAR (10)	Not Null	Unique ID of The Employee
UG Degree	VARCHAR (30)	Not Null	UG Degree Name of The Employee
UG College	VARCHAR (30)	Not Null	UG College Name of The Employee

UG University	VARCHAR (30)	Not Null	UG University Name of The Employee
UG YOG	SMALLDATETIME	Not Null	UG Year of Graduation of The Employee
PG Degree	VARCHAR (30)	Not Null	PG Degree Name of The Employee
PG College	VARCHAR (30)	Not Null	PG College Name of The Employee
PG University	VARCHAR (30)	Not Null	PG University Name of The Employee
PG YOG	SMALLDATETIME	Not Null	PG Year of Graduation of The Employee

Table 4.1.1.6 Employee Education Table

7. Designation

Field Name	Data Type	Constraints	Description
Dest ID	VARCHAR (10)	Primary Key	Unique ID of The Designation
Dest Name	VARCHAR (20)	Not Null	Name of The Designation

Table 4.1.1.7 Designation Table

8. Clients

Field Name	Data Type	Constraints	Description
Client ID	VARCHAR (10)	Primary Key	Unique ID of The Client
Client Name	VARCHAR (20)	Not Null	Name of The Client
Address	VARCHAR (20)	Not Null	Address of The Client
Contact Person	VARCHAR (20)	Not Null	Contact Person Name of The Client
Contact Number	Numeric (9)	Not Null	Contact Number of The Client
Mail Id	VARCHAR (30)	Not Null	Email Address of The Client

Table 4.1.1.8 Client Table

9. Project Type:

Field Name	Data Type	Constraints	Description
TypeId	VARCHAR (10)	Primary Key	Unique ID of The Project Type
TypeName	VARCHAR (20)	Not Null	Name of The Project Type

Table 4.1.1.9 Project Type Table

10. POSTARTICLE TABLE:

FILED NAME	FILED TYPE	Constraints	DESCRIPTION
Aid	Numeric(9)	Primary Key	Article id
Cname	VARCHAR(10)	Not Null	Category name
Title	VARCHAR(30)	Not Null	Article title
Key	VARCHAR(10)	Not Null	keyword
Adesc	VARCHAR(50)	Not Null	Article description

Table 4.1.1.10 Post article Table

11. SUGGESTION TABLE:

FILED NAME	FILED TYPE	Constraints	DESCRIPTION
Sid	Numeric(9)	Primary Key	Suggestion id
Aid	Numeric(9)	Foreign Key	Article id
Title	VARCHAR(30)	Not Null	Article title

Table 4.1.1.11 Suggestion Table

4.2 PROCESS MODEL

4.2.1 USECASE DIAGRAM

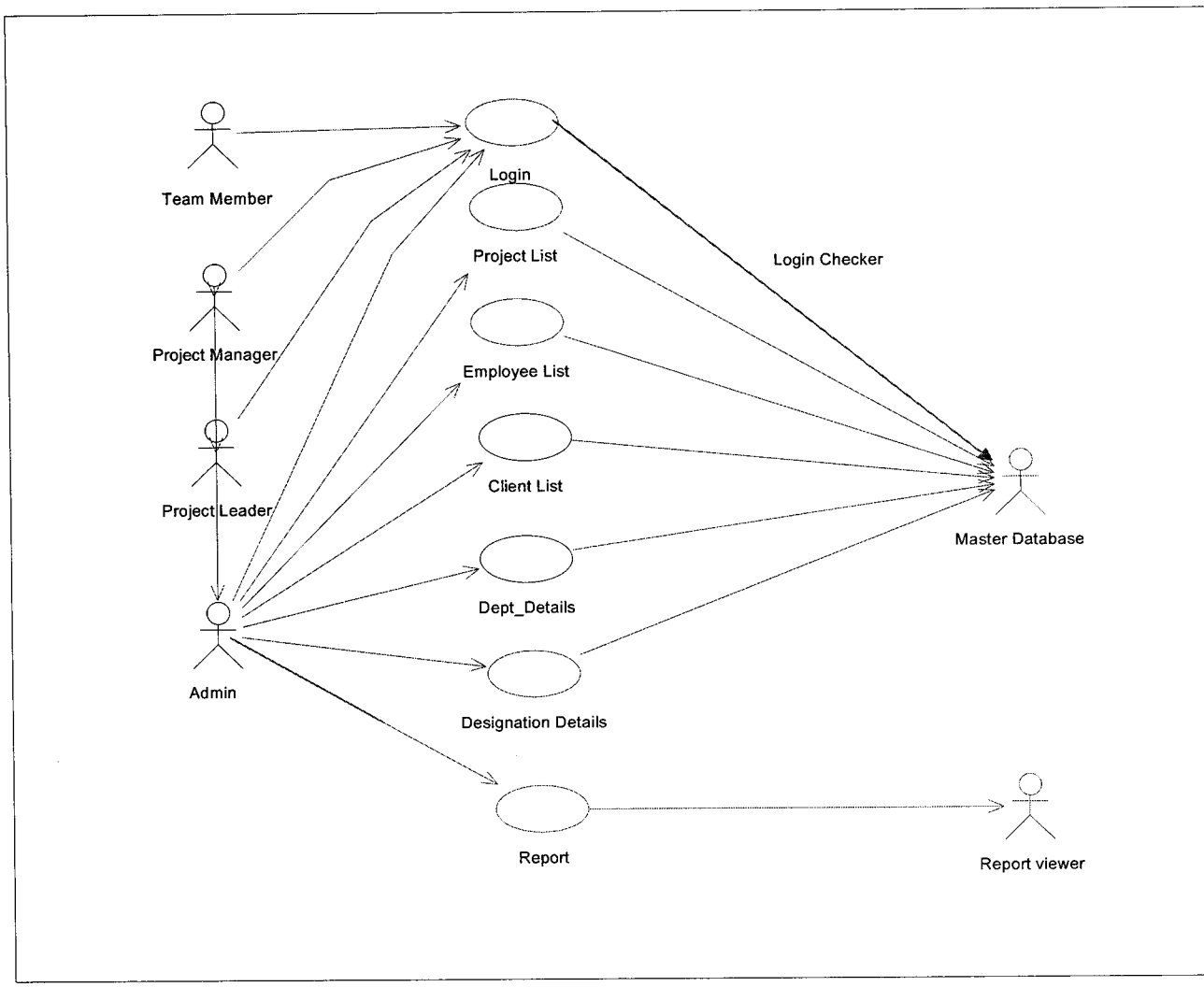

Fig 4.2.1.1 Use-Case Diagram

4.2.2. DATA FLOW DIAGRAM

Login Level 0:

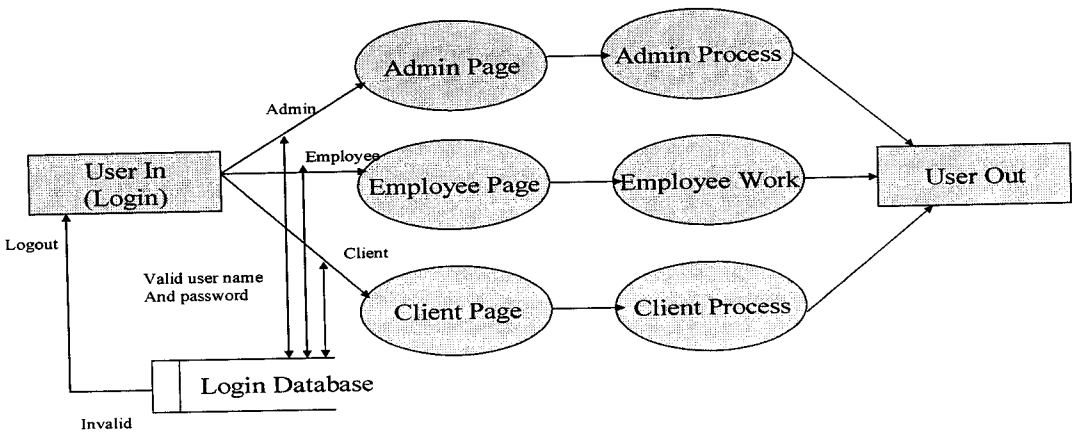

Fig 4.2.2.1 Level 1 DFD for User Page

Admin Login Level 1

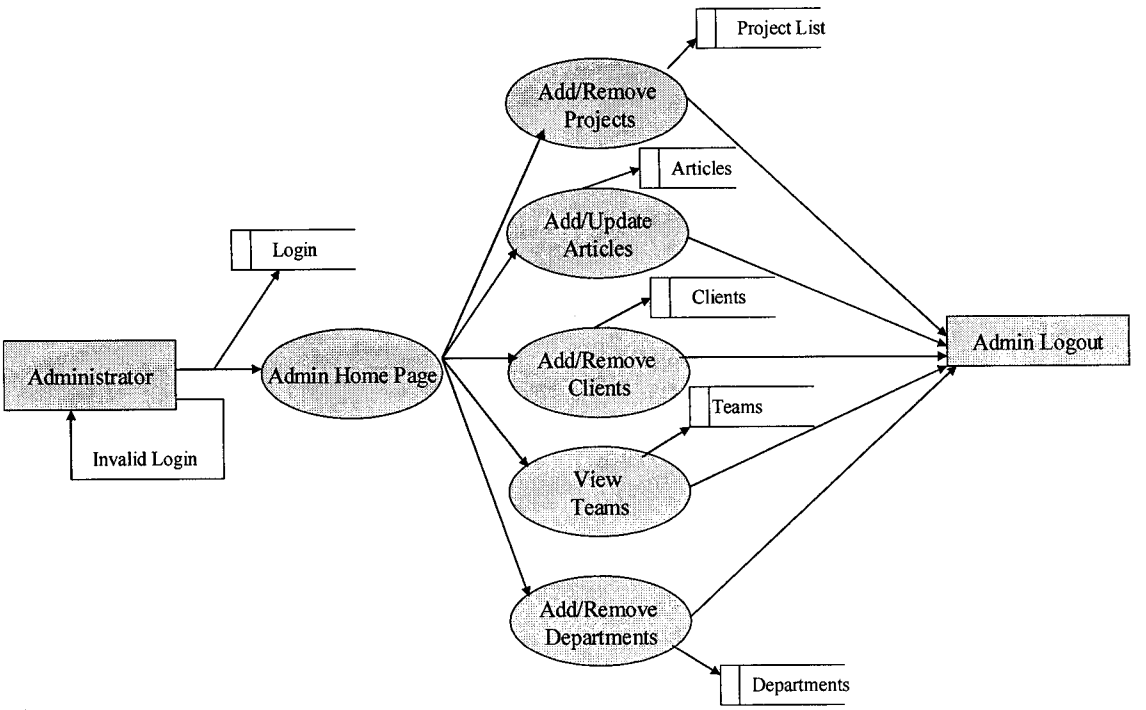

Fig 4.2.2.2 Level 2 DFD for Admin Page

Employee Home Page Level2

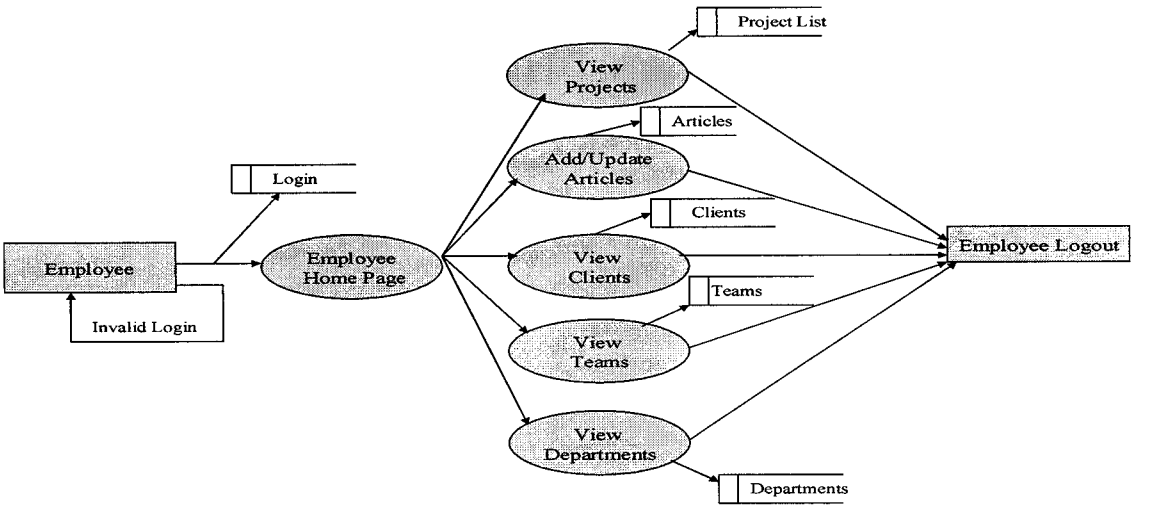

Fig 4.2.2.3 Level 2 DFD for User Page

Client Home Page Level2

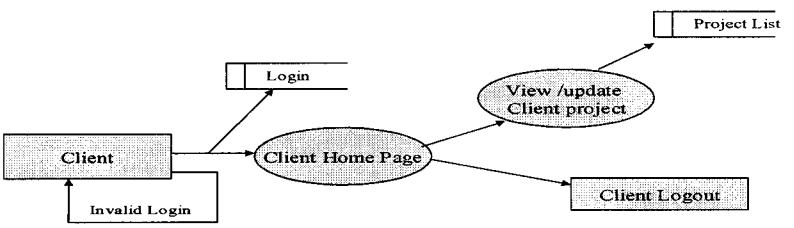

Fig.4.2.2.4 Level 2 DFD for Client Page

CHAPTER 5
TESTING

5.1 TEST CASES

Anything may be the system; testing phase is the final and important phase for it to be success. It is the stage of implementation, which is aimed at ensuring that the system works accurately and effectively before live operation commences. System testing makes a logical assumption that if all parts of the system are correct the goal will be successful. The user tests the developed system and changes are made according to their needs.

UNIT TESTING

This kind of testing is to verify the smallest unit of the software module. This is also known as "Module Testing". This test is carried out during the programming stage. This test ensures the expected output from each of the module. Exceptions have been handled and appropriate Error messages have been given in each module so as to avoid abnormal termination of the program.

.

Unit testing in the context of the developed system:

Here in this project, every module and the sub modules has been tested separately with the given test cases. For example in the Resource Management module, when any negative value is given for leverage or realizations, it should not calculate the optimal effort and should show an error message to the user.

TEST CASES

Test cases for Login Screen:

Test Case ID	Test Case Name	Test Case Description	Expected Result	Actual Result	Test Script Status
L01	User Name	Enter null string in "User Name" TextBox	"User Name" cannot be empty. Should display message to user	"User Name" cannot be empty. Should display message to user	Pass
L02	Password	Enter null string in "Password" TextBox	"Password" cannot be empty. Should display message to user	"Password" cannot be empty. Should display message to user	Pass
L03	User Name	Check maximum number of characters allowed in "User Name" TextBox	Should allow 10 characters in "User Name"	Should allow 10 characters in "User Name"	Pass
L04	User Name	Check Shift+Tab functionality in "User Name" TextBox	On hitting Shift+Tab key the focus should be taken to the screen	On hitting Shift+Tab key the focus should be taken to the screen	Pass
L05	Password	Check Tab functionality in "Password" TextBox	On hitting Tab key the focus should be taken to the "Login" Button	On hitting tab key the focus should be taken to the "Login" Button	Pass

L06	User Name	Enter invalid User name in "User Name" TextBox	Should display message to user	Should display message to user	Pass
L07	Password	Enter invalid User name in "Password" TextBox	Should display message to user	Should display message to user	Pass
L08	User Name and Password	Enter valid User Name and Password	Should navigate to Home page	Should navigate to Home page	Pass
L09	Password	Check Shift+Tab functionality in "Password"	On hitting Shift+Tab key the focus should be taken to the "User Name" TextBox	On hitting Shift+Tab key the focus should be taken to the "User Name" TextBox	Pass

Table: Test cases for Login Screen

Test cases for Personal details:

Test Case ID	**Test Case Name**	**Test Case Description**	**Expected Result**	**Actual Result**	**Test Script Status**
L01	First Name	Enter null string in "First Name" TextBox	"First Name" cannot be empty. Should display message to user	"First Name" cannot be empty. Should display message to user	Pass
L03	First Name	Check maximum number of characters allowed in "User Name" TextBox	Should allow 20 characters in "First Name"	Should allow 20 characters in "First Name"	Pass

L01	Last Name	Enter null string in "Last Name" TextBox	"Last Name" cannot be empty. Should display message to user	"Last Name" cannot be empty. Should display message to user	Pass
L03	Last Name	Check maximum number of characters allowed in "Last Name" TextBox	Should allow 10 characters in "Last Name"	Should allow 10 characters in "Last Name"	Pass
L01	User Name	Enter null string in "User Name" TextBox	"User Name" cannot be empty. Should display message to user	"User Name" cannot be empty. Should display message to user	Pass
L03	User Name	Check maximum number of characters allowed in "User Name" TextBox	Should allow 20 characters in "User Name"	Should allow 20 characters in "User Name"	Pass
L07	Password	Check Tab functionality in "Password" TextBox	On hitting Tab key the focus should be taken to the "Login" Button	On hitting tab key the focus should be taken to the "Login" Button	Pass
L02	Password	Enter null string in "Password" TextBox	"Password" cannot be empty. Should display message to user	"Password" cannot be empty. Should display message to user	Pass

Table: Test cases for Personal details Screen.

Output Testing:

After performing the validation testing, the next step is output testing of the proposed system since no system would be termed as useful until it does produce the required output in the specified format. Output format is considered in two ways, the screen format and the printer format.

INTEGRATION TESTING

This kind of testing is a systematic testing for constructing tests to uncover errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole.The system underwent a series of Integration tests that recorded smooth transmission of data from one module to the other. In this project the developed system is tested after integrating various modules together, and the detected errors were corrected.

DATA TESTING

Various kinds of test data are used to verify the system, and the output is verified with the required one. The data available should be appropriate on whose basis testing is performed.

RESULT

Thus various kinds of tests like unit testing, integration testing and data testing are done. Thus it is tested that the system will work properly and effectively.

CHAPTER 6
PERFORMANCE AND LIMTIATIONS

6.1 MERITS OF THE SYSTEM

- Significant ROI is possible through deploying clock wares time and work Flowing systems.
- Human Resources can track and monitor all leave categories with an in limited number of balances.
- Line Managers can use extensive workflow features to ensure that time taken to complete the projects. Submitted, approved and processed on time.
- Senior Management can benefit from reporting tools that analyze major projects, clients and billing.
- Components provide highly configurable business rules and policies that are easily maintained.
- Security from role-based and other key security features ensures the system supports an organization structure and policies.
- Productivity flowing features is included in one system.
- Ease of Use ensures simple system administration and minimal training requirements.

6.2 LIMITATIONS

Following are the limitations of the system namely

- It is possible with Internet access only
- Modification is possible
- Since auto format it change automatically
- Only admin can have the rights to monitor and make changes in the proposed system.
- The client can view the status of the project but not the partial implementation of the project.
- This project status can't track the changes of Time to Time project development process.

6.3 FUTURE ENCHANCEMENTS

The system is developed keeping in mind that it should adapt to the further requirements to the greatest extent. The newly developed system is found to be working efficiently and effectively than existing one.

- The clients can use the voice mail for their registration process and feedback process.
- The client can view the existing projects through articles.
- The partial implementation of the project can be viewed by the client.
- Securities of the system for using hand print through password entries.

CHAPTER 7
APPENDICES

7.1 SAMPLE SCREENS

LOGIN FORM

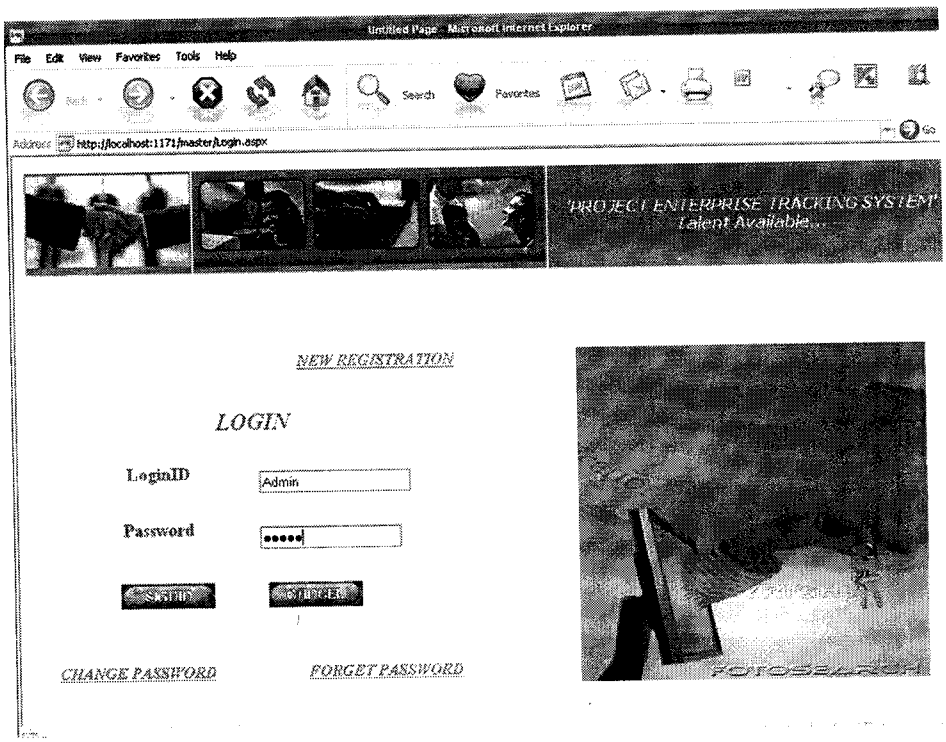

Figure A.1 User Login Form

CLIENT DETAILS

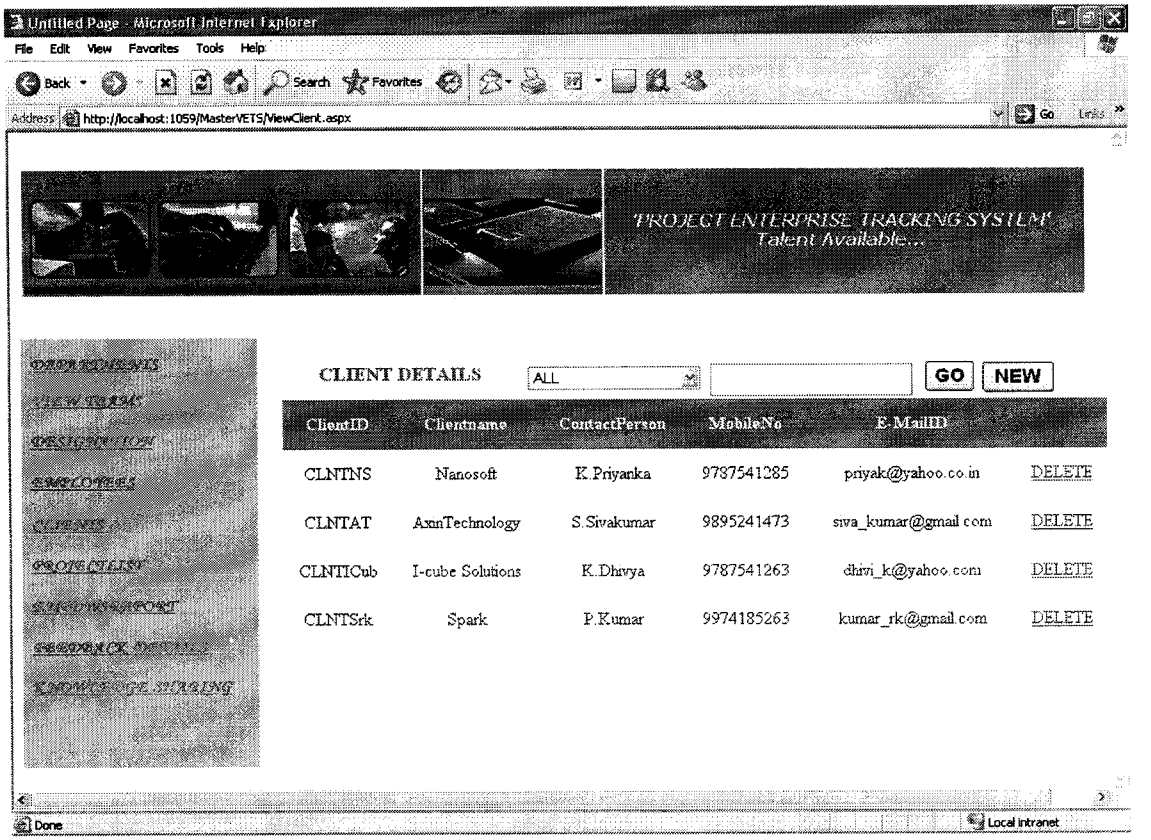

Figure A.2 Client Details

EMPLOYEE DETAILS

Figure A.3 Employee Details

PERSONAL DETAILS

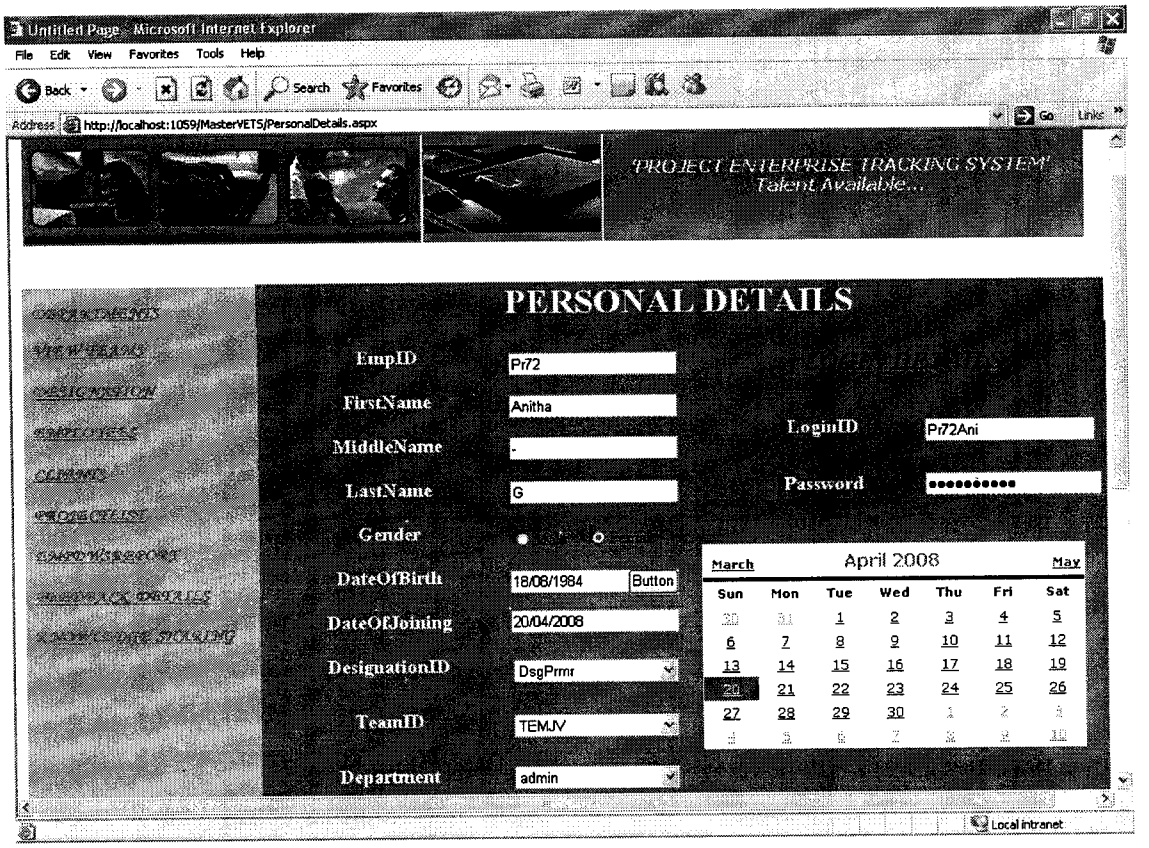

Figure A.4 Personal Details

FEEDBACK FORM

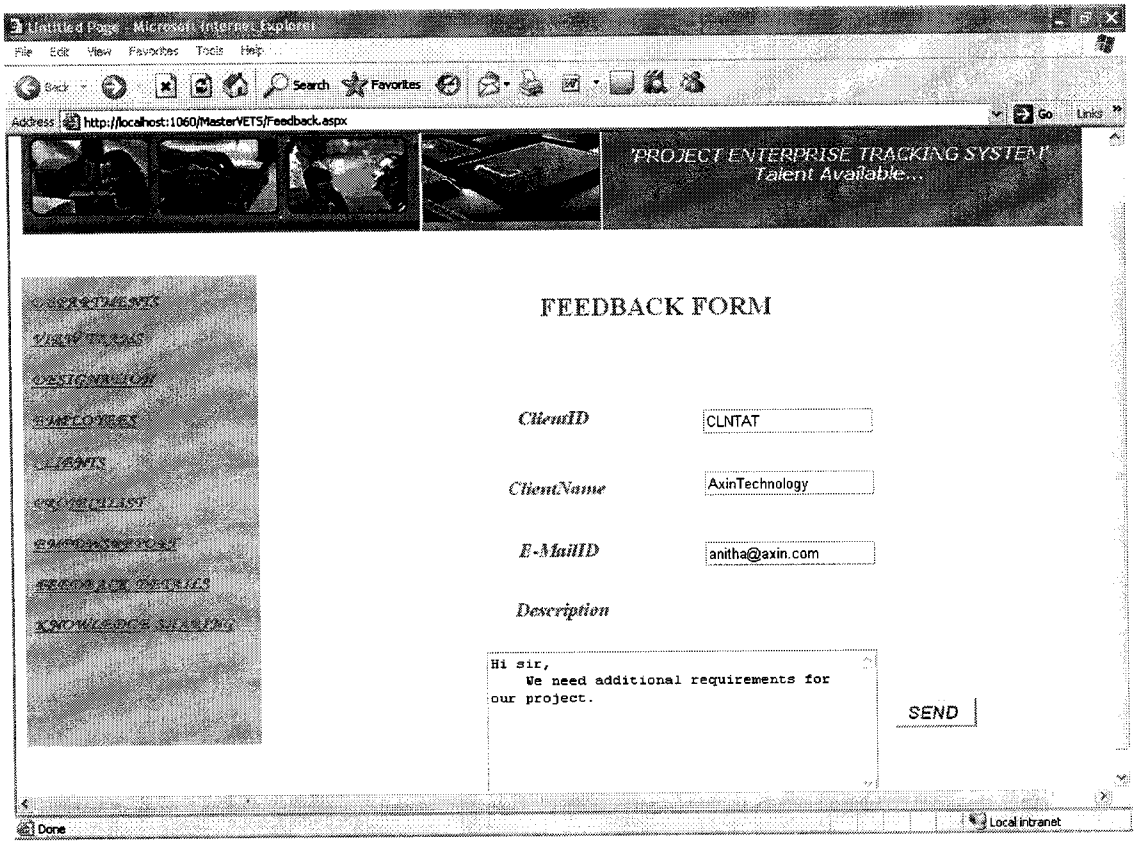

Figure A.5 Feedback Form

VIEW ARTICLE

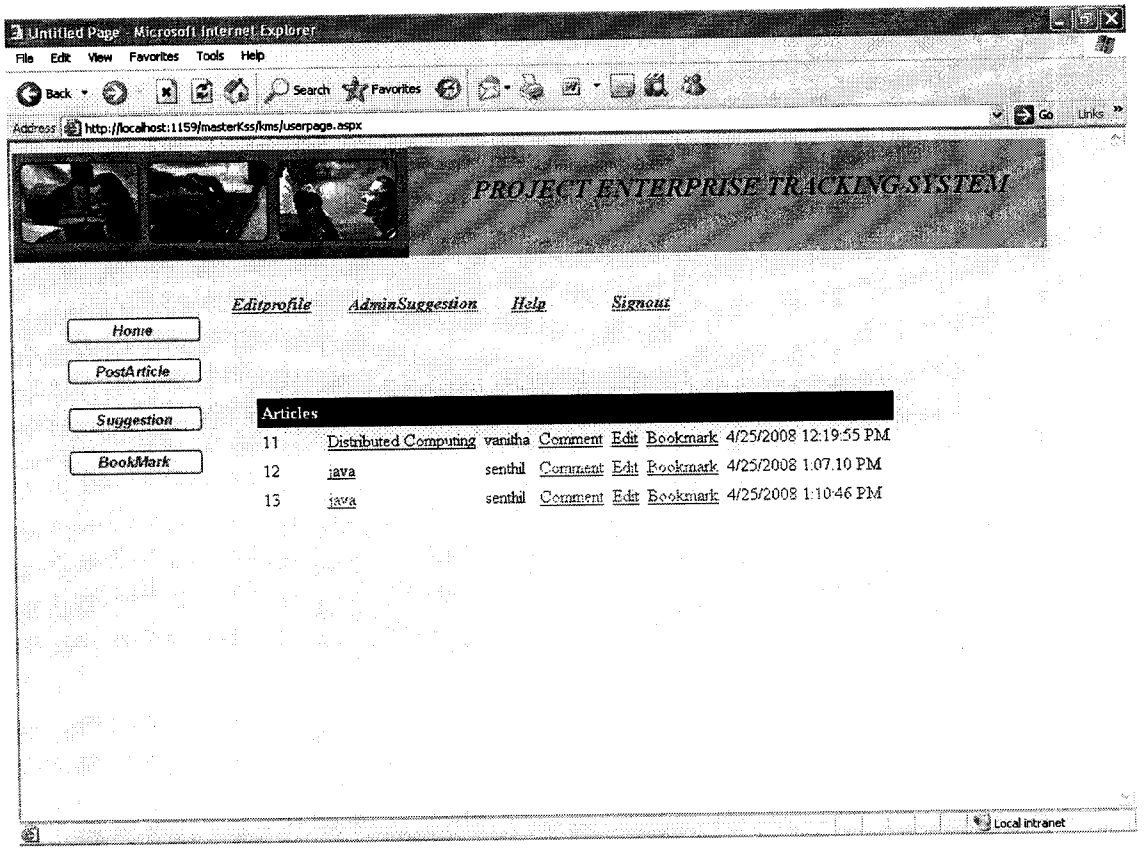

Figure A.6 View Article

NEW FILES

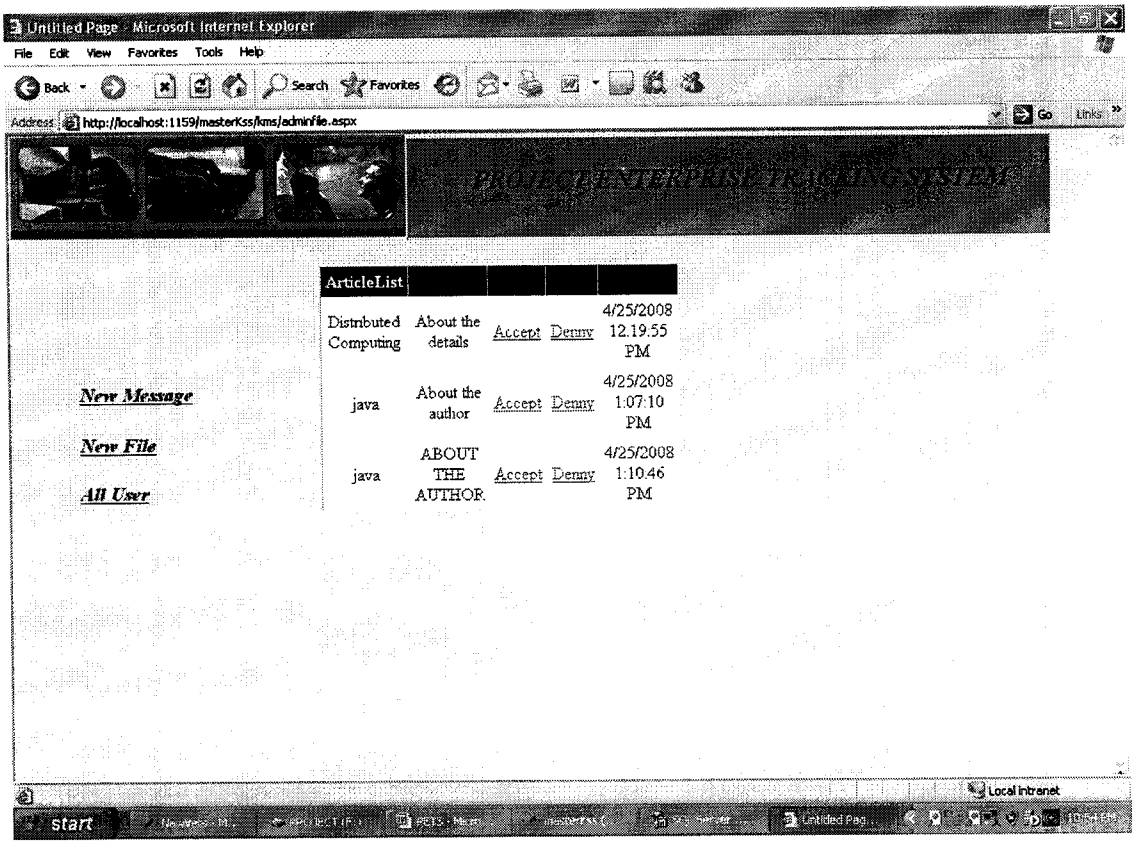

Figure A.7 New Files

ASSIGN MODULE

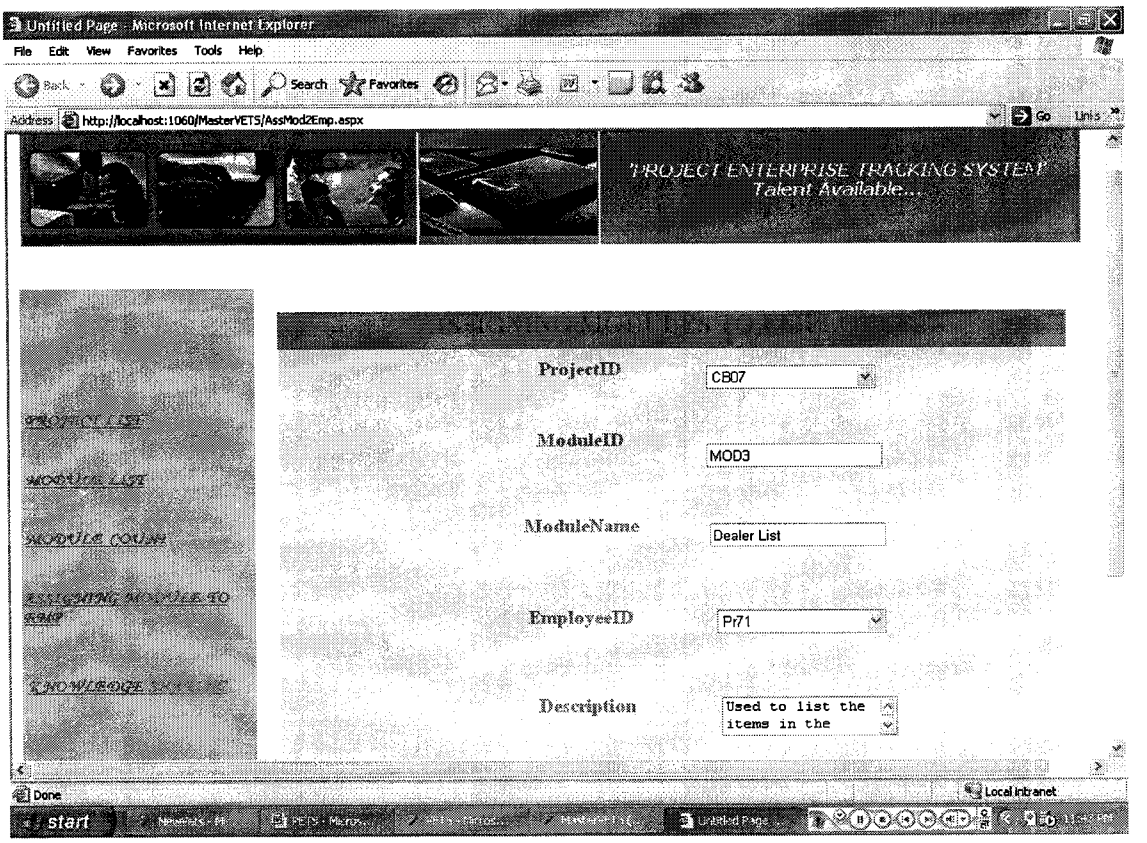

Figure A.8 Assign Module

DEPARTMENT DETAILS

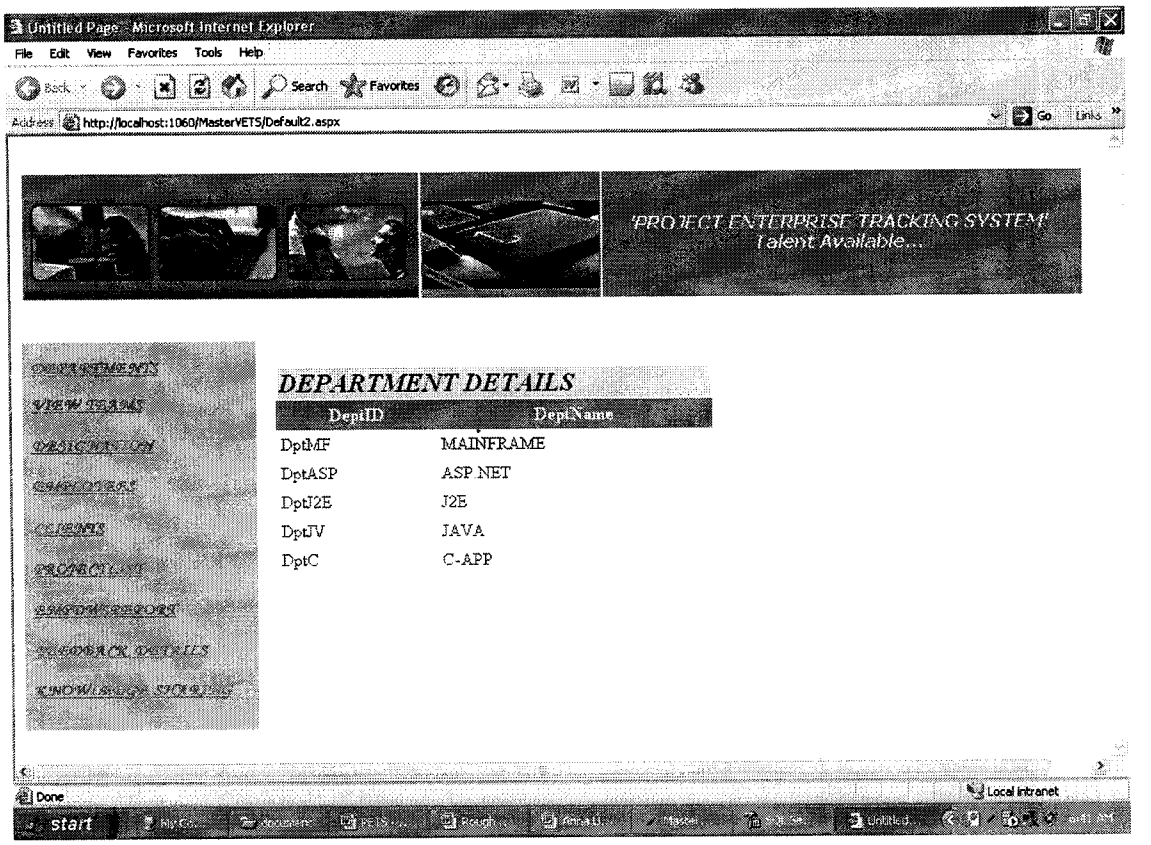

Figure A.9 Department Details

POST ARTICLE

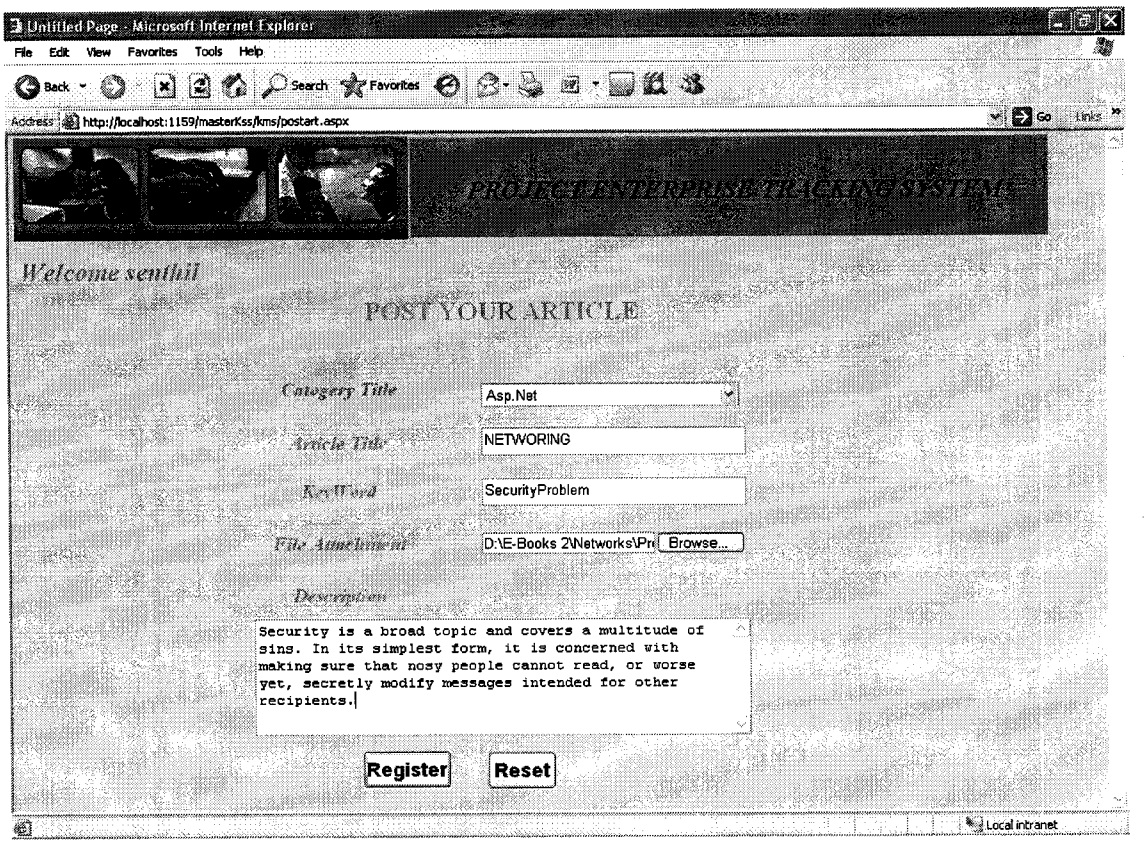

Figure A.10 Post Article

EMPLOYEES:

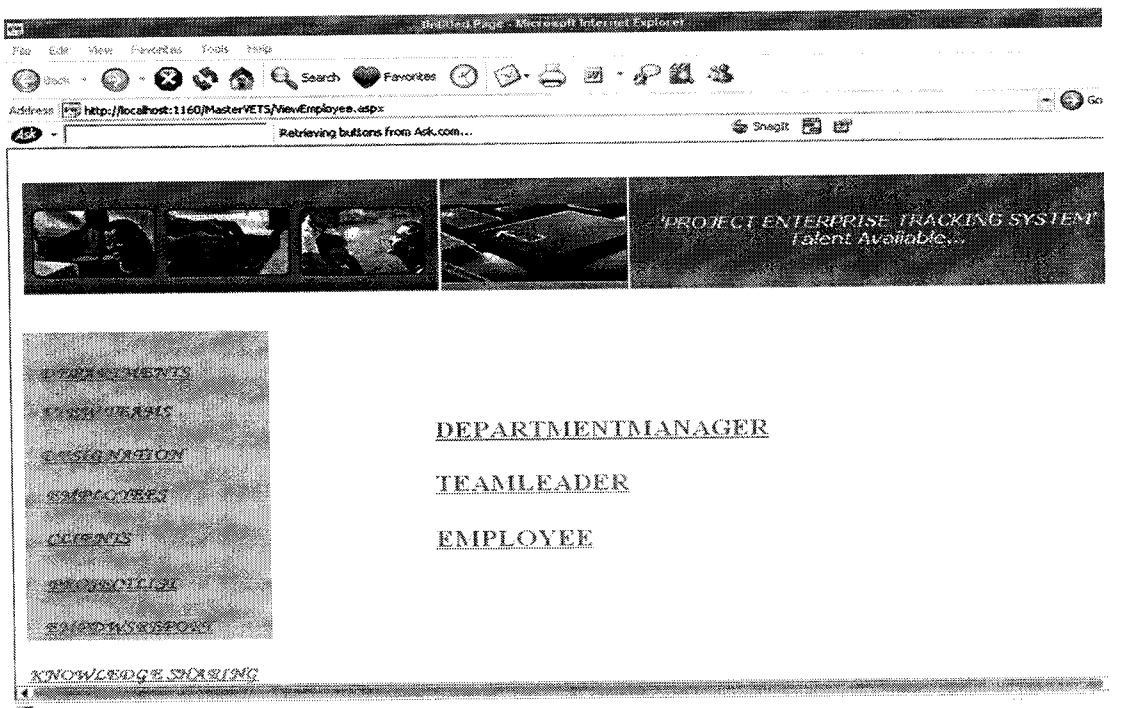

Figure A.11 Employees

SUGGESTION:

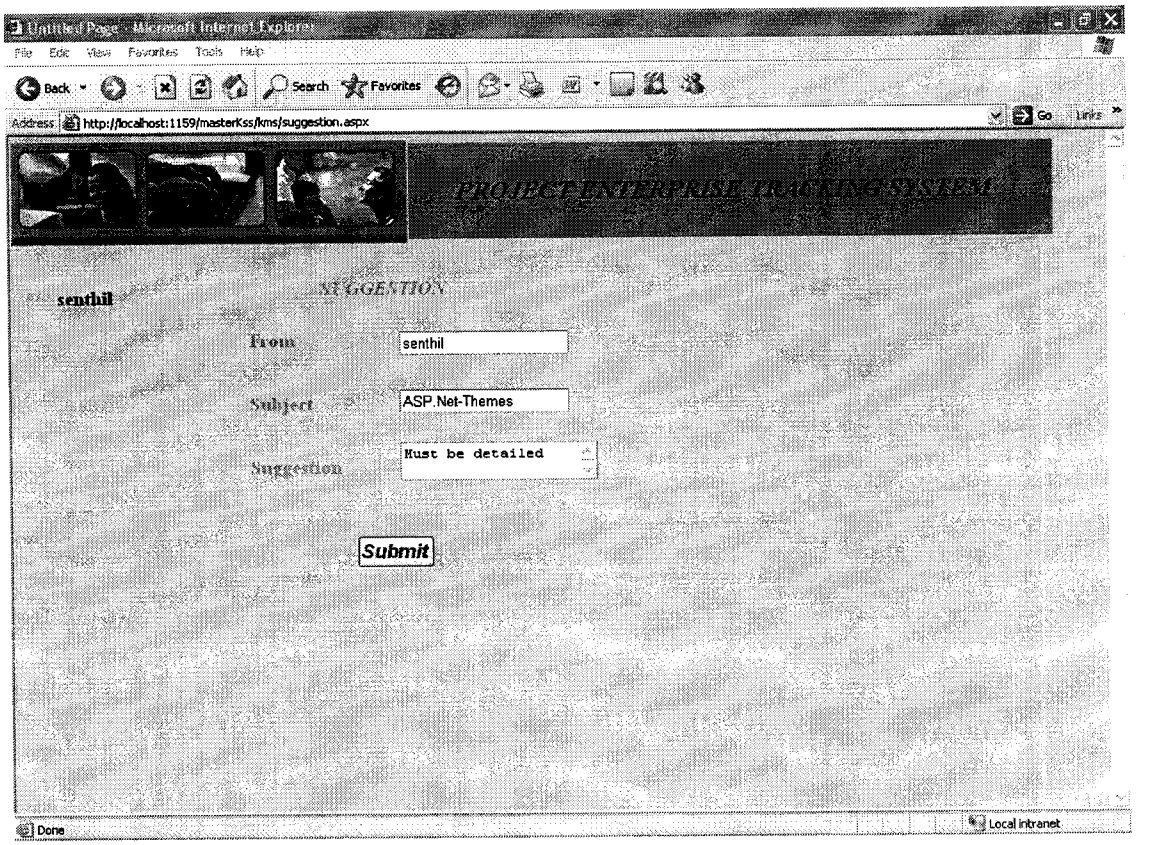

Figure A.12 Suggestion

PROJECTLIST

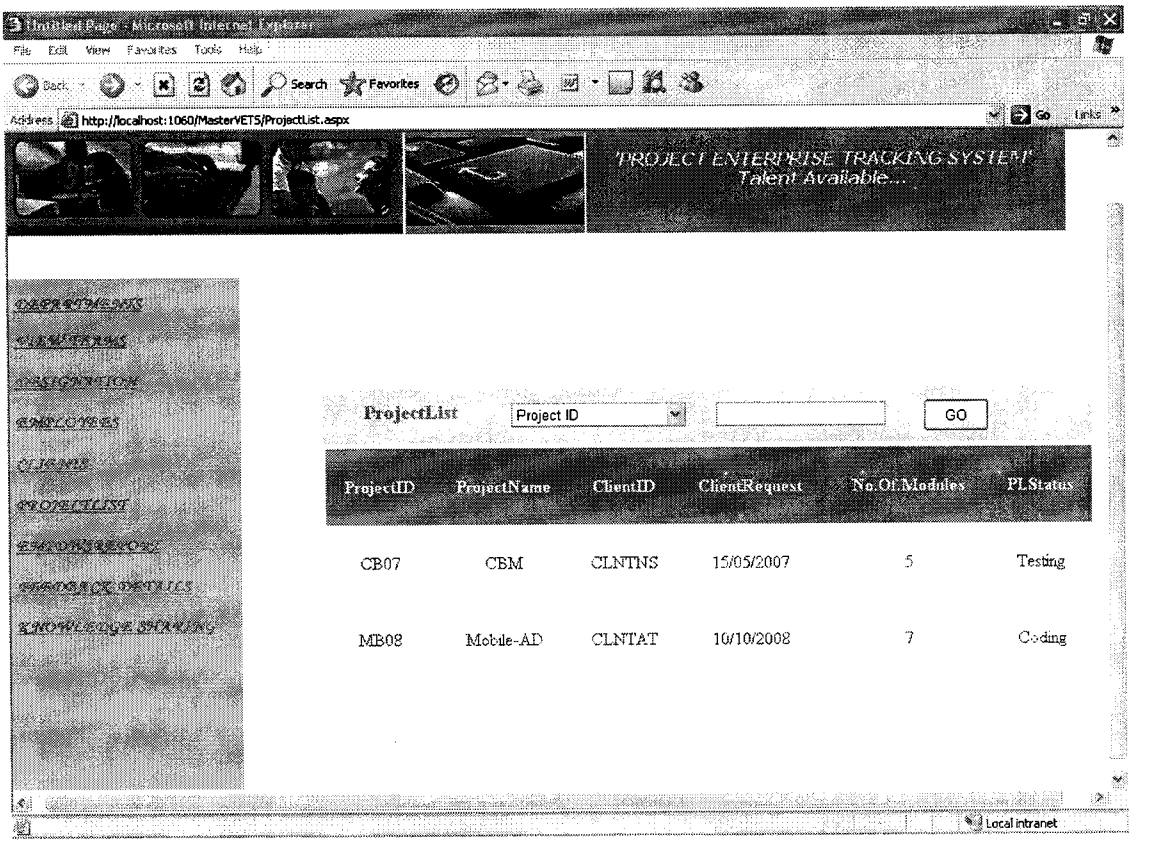

Figure A.13 Project List

REFERENCES

BOOKS

1. Ben Chary, Mark Scardina and Stefan Kiritzov, **"Oracle 9i XML Handbook"**, McGraw-Hill, 2001.
2. Brian Francis and Alex Homer, **"Professional ASP .NET"**, Shroff Publishers, 2001.
3. Graeme Malcolm, **"Programming Microsoft SQL Server 2000 with XML"**, WP Publishers, 2002.
4. Roger S Pressman, **"Software Engineering – A Practitioner's Approach"**, McGraw-Hill, 2005.

Web sites

5. http://mysqlcity.com
6. http://www.oracle.com/technology