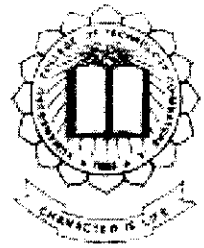



# TASK TRACKING SYSTEM

By

**Sathishbabu. R**

**Registration Number: 71206621046**

**Of**

**KUMARAGURU COLLEGE OF TECHNOLOGY**

**COIMBATORE**


**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 2009**

**KUMARAGURU COLLEGE OF TECHNOLOGY**

**COIMBATORE – 641006**

# BONAFIDE CERTIFICATE

Certified that this project report titled **"Task Tracking System"**is the bonafide work of **"Mr. R.Sathishbabu"** (Register Number: **71206621046**) 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** 06/07/09

6/7/09

**Internal Examiner**

6/7/09

**External Examiner**

AITHENT
innovating evolution

Date: 05-06-09

## CERTIFICATE

This is to certify that, Mr. Sathishbabu.R, III rd M.C.A Reg No: 71206621046 Student of Kumaraguru College of Technology, Coimbatore has successfully completed his project work, titled **Task Tracking System** as part of his course curriculum.

He has done the project using **Microsoft Visual Studio 2005** during the period of **December'08 to June'09** under the guidance and supervision of **Mr.Sandeep Kalmadi.M, Software Engineer, Aithent Technologies Pvt Ltd, Chennai.**

He has completed the assigned project well within the time frame. He is sincere, hardworking and his conduct during project is commendable.

We wish him all the best in his future endeavors.

For **Aithent Technologies Pvt Ltd**

**Project Manager**

**Project Guide**

RR Towers-II, Ground Floor, 94-95, Thiru-VI-KA Industrial Estate, Guindy, Chennai 600 032.
Ph: +91 (44) 4212 5772 Email: info@aithent.co.in www.aithent.com

# ABSTRACT

The project is easy to use, user friendly application for handling scheduling and work distribution problem in software development. The project is designed to handle different levels of users. The project is intended to handle work allocation and scheduling problems in software development process. Centralized server which holds database records and client machine running the application makes the entire system. Usually the allocation of work and distributing the information to the developers and other level managers is a tedious task in software management. This needs lots of manual work to be done for creating and distributing respective work schedule to each member in Software Development.

The application is developed by identifying user requirements at different levels. An employee at development level needs the information of his/her work, scheduling and updating the completed work. And at management level people need facility to assign process and keep track of work progress. And at Administration level configuring system and controlling user's accounts are the main task.

The systems main advantage over the existing system is user friendly System Interface and different services all levels of users. System provides way to Administrate users, Different type of work allocation to different users, way to view assigned work, monitor the work progress, update the work that are started/completed.

# ACKNOWLEDGEMENT

First and foremost I thank God for his good will and blessings showered on me throughout the project. The success of this project needs cooperation and encouragement from different quarters. Words are inadequate to express my profound and deep sense of gratitude to those who helped me in bringing out this project successfully.

I am very gladly taking this opportunity to express a special word of thanks to **Dr. M. Gururajan M.Sc., Ph.D,** Head of the Department, Kumaraguru College of Technology, Coimbatore for encouraging me to do this work.

I am very much indebted to **Mrs. V.Geetha** Assistant Professor Kumaraguru College of Technology, Coimbatore for her complete assistance, guidance and support given to me throughout my project.

I would express heartfelt thanks to our internal guide **Mr. S.Ganeshbabu** Senior Lecturer, Kumaraguru College of Technology as with out his best guidance it would not have been possible for me to successfully complete this project who also gave his innovative ideas at crucial times and tremendous encouragement.

It is my pleasure to express my profound gratitude to **Aithent Technologies Solutions**, Chennai for admitting into this project. I am thankful to **Mr. Sandeep Kalmadi** of Comprehensive Business Solutions, for his excellent guidance, timely suggestions and constant support in all my endeavors.

# Table of Contents

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| TTS | Task Tracking System |
| VB | Visual Basic |
| CMM | Capability Maturity Model |
| DB | Database |
| ER | Entity Relationship |
| PDF | Portable Document Format |

# 1. INTRODUCTION

## 1.1 Organisation Profile

Aithent Technologies, a CMM Level-5 global software services and consulting firm. Aithent Inc, a leading corporate with nation wide presence, is the single largest source for a wide range of software products and services. Aithent Inc is a matured software services provider with a perfect blend of technical skills, domain expertise and professionalism. Aithent is the reseller Alchemy's software across US and Canada focuses on modernizing customers' mainframe based legacy assets.

Aithent Inc, a global IT solutions provider and a leading provider of case management solutions announced today that a State-level government Department of Health and Mental Hygiene has selected Aithent's Fraud Prevention and Investigation (Aithent FPIT) system for use as their core Automated Fraud and Abuse tracking System to manage and control provider and recipient Medicaid fraud.

Aithent is currently a member of the HP Developer and Solutions Partner Program (DSPP). Aithent will work closely with HP to provide customers with high quality and cost-effective technology solutions to meet the demanding business requirements.

## 1.2 Problem Definition

Existing system uses Microsoft Excel to manage their scheduling and work allotment process. It seems too hard to handle large users and large scheduling information. There is a possibility to misuse of modify the information by the users. Since there is a use of Macro in the Excel Document, people at management level needed some basic knowledge about this. Updating information by multiple users and handling information in this type of system is a time consuming one and backing up of information is also time consuming task. This leads the existing system to change.

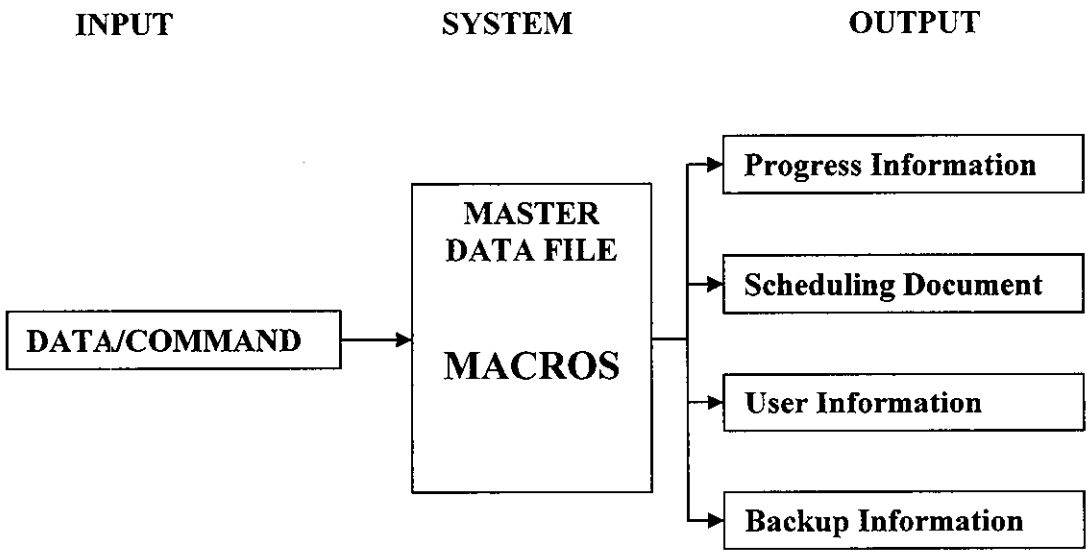
The proposed system uses Back-End Database to store information, application to access the system. System provides different functionality to different users. This system is user friendly that handles huge data in form of database tables. System stores the scheduling information and work allotted information in database. So that the users can view the work assigned to them. System provides service to assign work to employees and track the work progress. Employees can update the work which are started and completed. Thus this collated information gives the management, work progressed and status of the project. Also helps to track the issues in the work like work that are lagging the scheduling and crossing the red alert limit.

Information of people at work and in floor are identified easily so to assign work to them. This leads to use of human resource effectively. System provides way to track project by team and as whole project.

# 2. SYSTEM ANALYSIS

## 2.1 Existing System Architecture

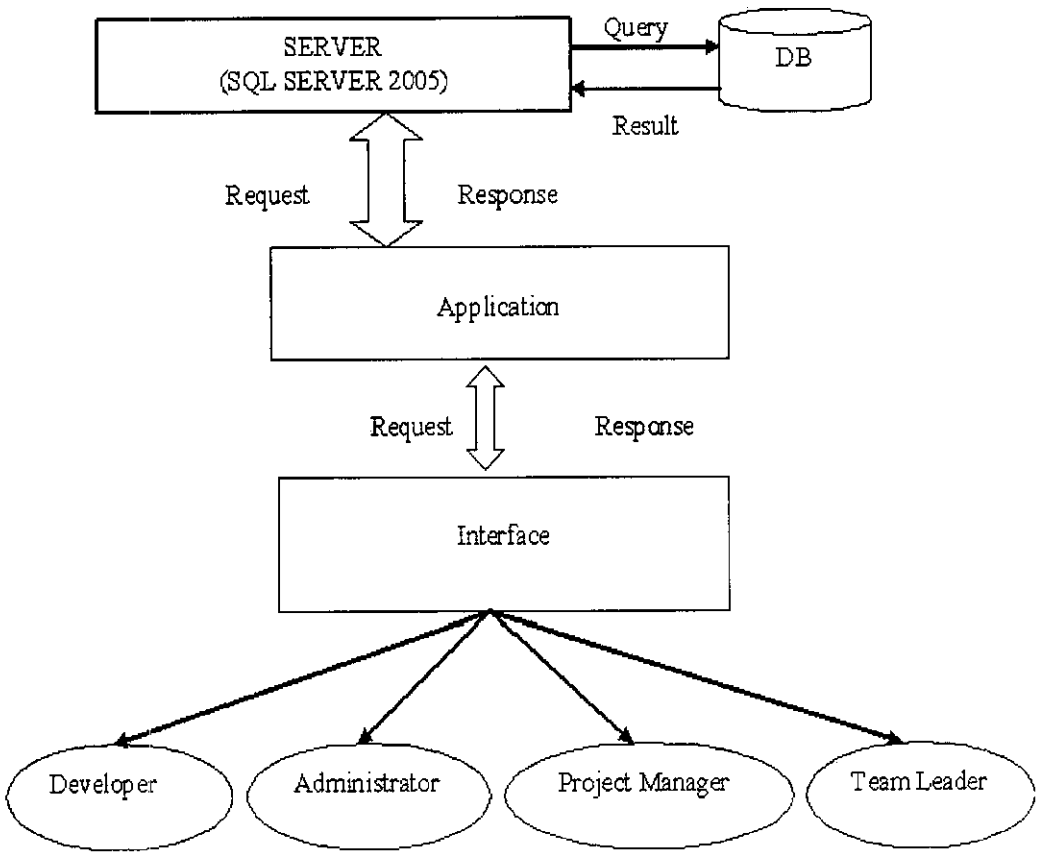
Existing system used simple excel documentation to process all type of information. Master Data File is the main document, which serves as processing system. This document runs under completely programmed Macro's. Inputs to the system are information/data to store and command to process the stored information. Outputs from the system are scheduling document which generated from stored information from master file, User information (Personal/Official), Work progress information which is collated information from related sources and Backup file which serves as backup since Master Data File can't hold huge data.



### 2.1.1 Limitations of Exiting System

- The system limits use of store information, not able to store huge information.
- Maintaining and programming Macro's which forms heart of the system is a tedious task and needs good technical skills.
- Distribution of work information to employees leads to share the information document or to have duplicate file for each employee.
- Regular Updation of information in Master Data File should be reflected in all other related documents, which is a time consuming factor.
- Employee's needs to update the work progress assigned at regular basis. This information also updated in Master Data File which needs link all file to Master file. This leads to information security problem.

## 2.2 Proposed System Architecture



The system combines Back-End Database to store information and Front-End User Interface to interact with End-User. Back-End of the system is SQL Server 2005. And the Front-End of the system is developed using VB.NET 2005.

The Front-End of the system sends request to the Database Server and in turn the server queries the database thus to store or retrieve information. The End-Users are Administrator, Project Manager, Team Leaders and Developers. Since the system is developed using Dot Net environment user gets lot of user friendly services. System provides various services to different kinds of users.

Centralized database server which runs the System database needs SQL Client to be configured in the network system to access the database. Once the client is configured the application can be just executed, and access the database and system is ready to use.

### 2.2.1 Advantages of Proposed System

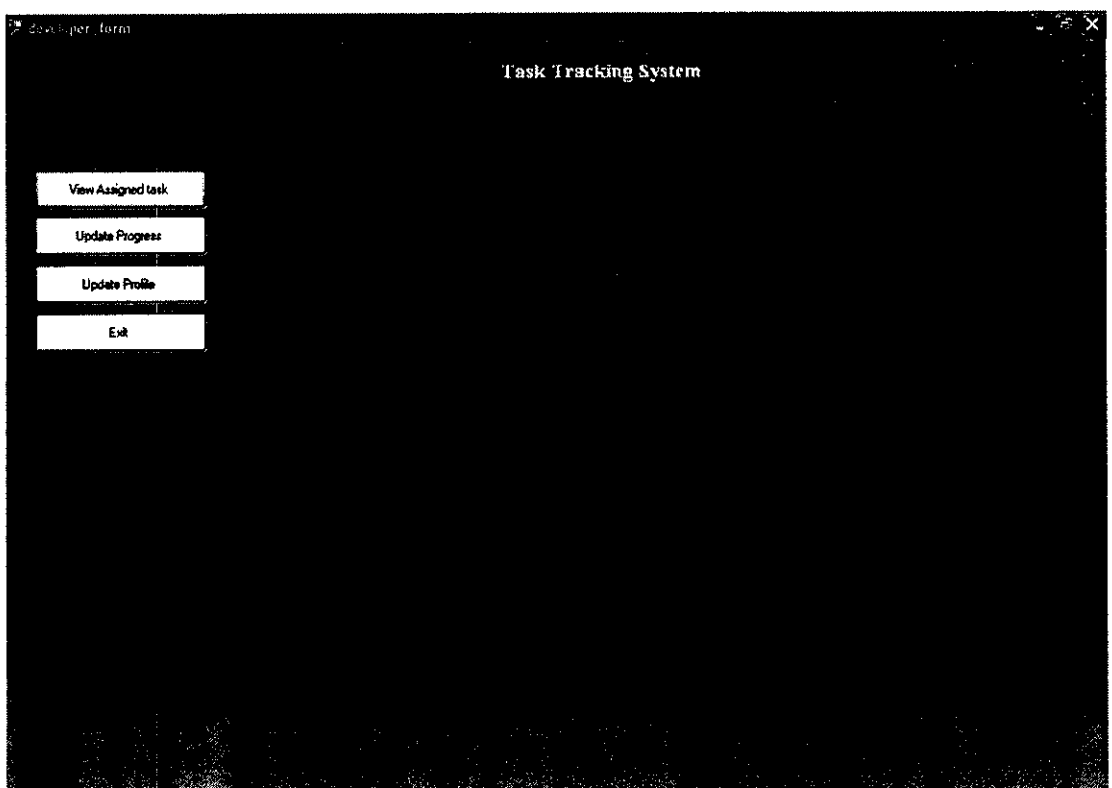
- Stores huge information and handles the information very simply since the information are stored in related tables.
- Back-End and Front-End are separated which improves the information security.
- Reduces the complexity in the process followed in proposed system.
- Information Updation at centralized database will reflect in all related tables.
- All kind of information's is retrieved by management which gives idea to make timely decisions.
- System is user friendly to use and gives all kind of informative data which are needed for all levels of user.
- Multiple users are supported by the system, which can run the application at different user systems.

## 2.3 User Interface Requirements

Modules involved in the system are Administrator, Project Manager, Team Leader and Developers. Each user has their common type of form to interact with the system. User interface commonly developed for Admin, Project Manager, Team Leader and Developers. Users have rights to access their common interface based upon their designation.

### User Interface for Developers

- Developer level information as group gives major progress
- Update profile like changing password and updating personal information
- View assigned activity and its scheduling information
- Update the progress information. By updating process started at that staff day and process completed at that staff day. So this forms basis for collating team progress and project progress.



**User Interface for Administrator**

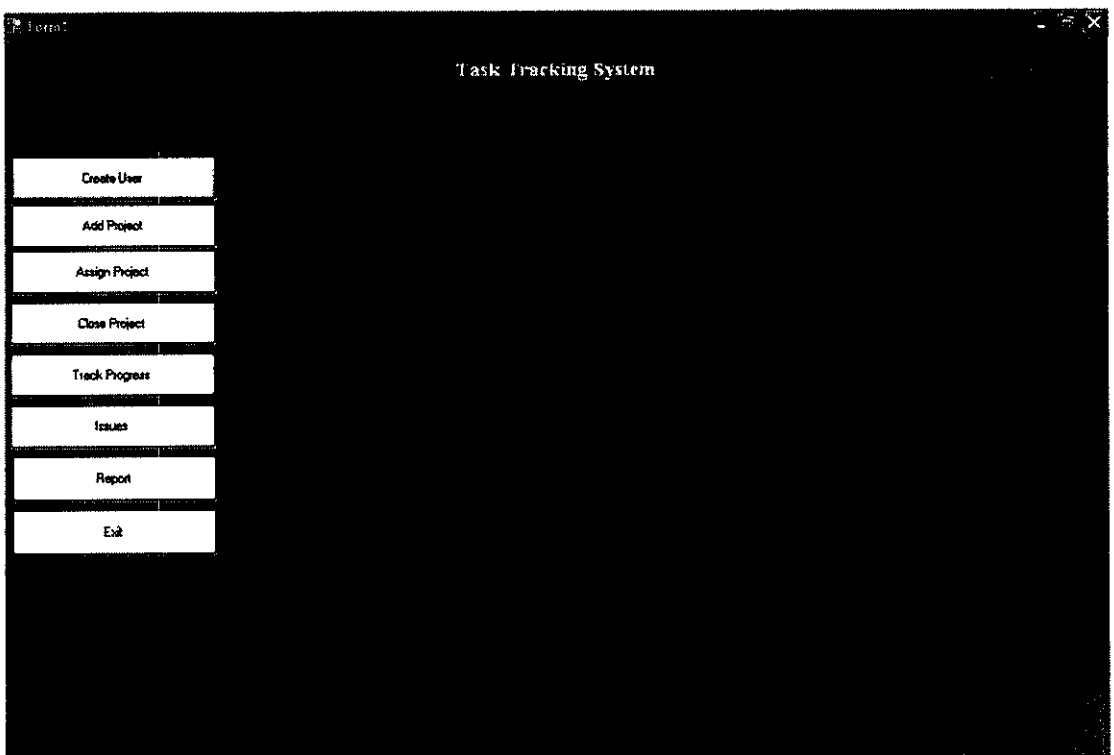
Administrator has functionality to manage users and project information.

Admin management over user are,

- Add a New User

Admin management over the project are,

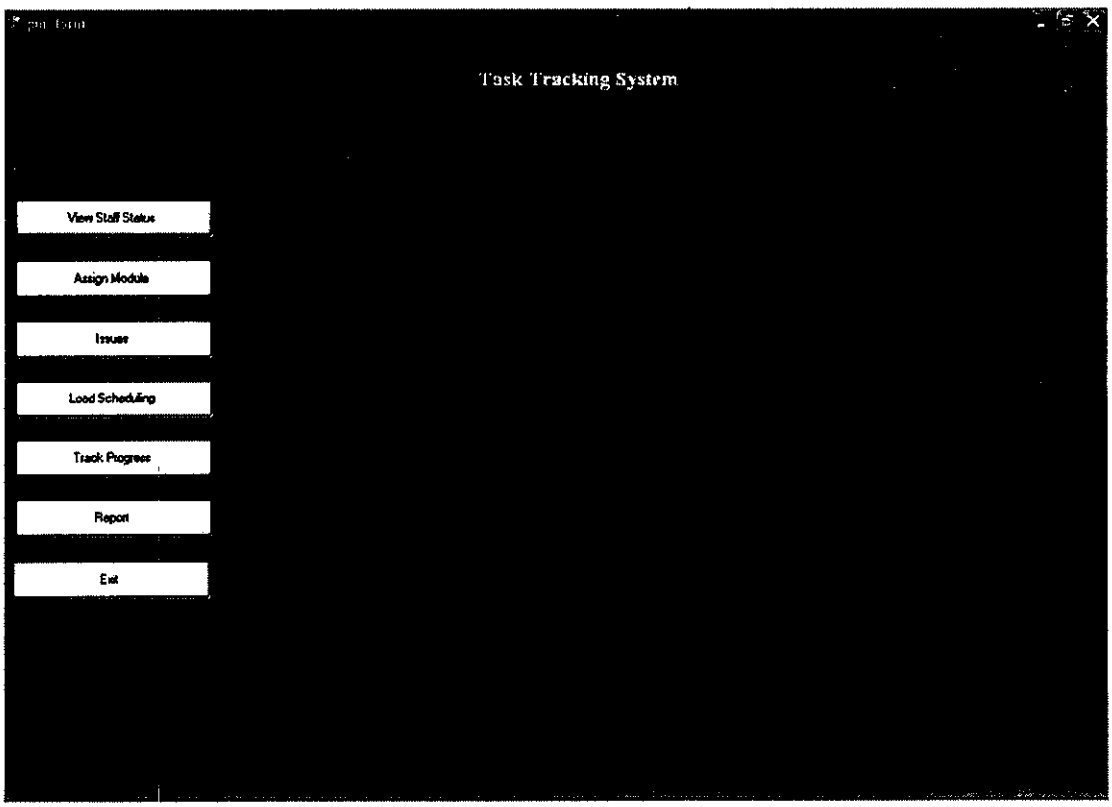
- Add a new Project and input data regarding, like Company Name, Location of Project Handling, and Project Name
- Assign a Project to Project Manager
- Closing a Project, so related data are grouped and stored in a file.
- Track the Project Progress by means Individual, Team Level and as Complete Project
- View any issues affecting the project progress like scheduling not met, process lagging certain time limit
- Reports are generated based upon various criteria like reports drawn for individual, team and as whole project



**User Interface for Project Manager**

Project Manager functions are,

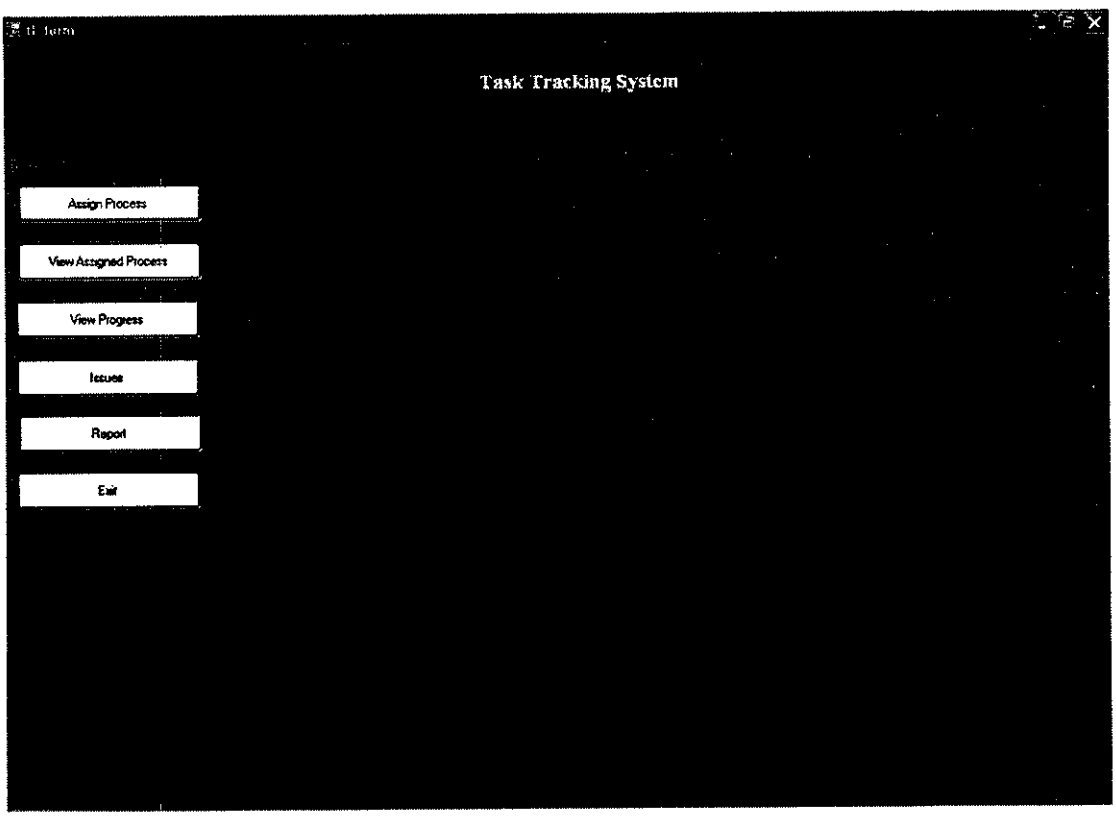
- View status of the Employee as he/she engaged in process or not assigned any process
- Can assign Modules to Team, by assigning it to Team Leader
- View any issues affecting the project progress like scheduling not met, process lagging certain time limit
- Once project is assigned to project manager, he/she can load the scheduling data so that project scheduling for that particular project is updated in database
- Track the Project Progress by means Individual, Team Level and as Complete Project
- Reports are generated based upon various criteria like reports drawn for individual, team and as whole project



**User Interface for Team Leader**

Team Leader functionality are,

- Assign Activities in the module to an individual
- Can view module assigned and its related activities, scheduling information
- Can view progress of module or activity so as to assure the progress met the scheduling
- Track the Project Progress by means Individual, Team Level and as Complete Project
- Reports related to module progress and individual activity can be generated



# 3. DEVELOPMENT ENVIRONMENT

## 3.1 Hardware Environment

| | | |
|---|---|---|
| Hard disk | : | 80 GB |
| Main memory | : | 512 MB Ram |
| Processor | : | Pentium IV 2.4 GHz |
| CD-Drive | : | 52x LG |
| Keyboard | : | Standard US Keyboard |
| Pointing Device | : | 2 or 3 button Mouse |

### 3.1.1 Related Working Environment

Standard network setup, for running centralized SQL Server. And it acts as central Database for all users. Thus standard LAN setup, server and client connect via network device switches or Hub).

## 3.2 Software Environment

| | | |
|---|---|---|
| Operating System | : | Windows XP |
| Software Package | : | Visual Studio 2005 with DotNet Frame work 2.0 |
| Database | : | SQL Server 2005 |

### 3.2.1 Related Working Environment

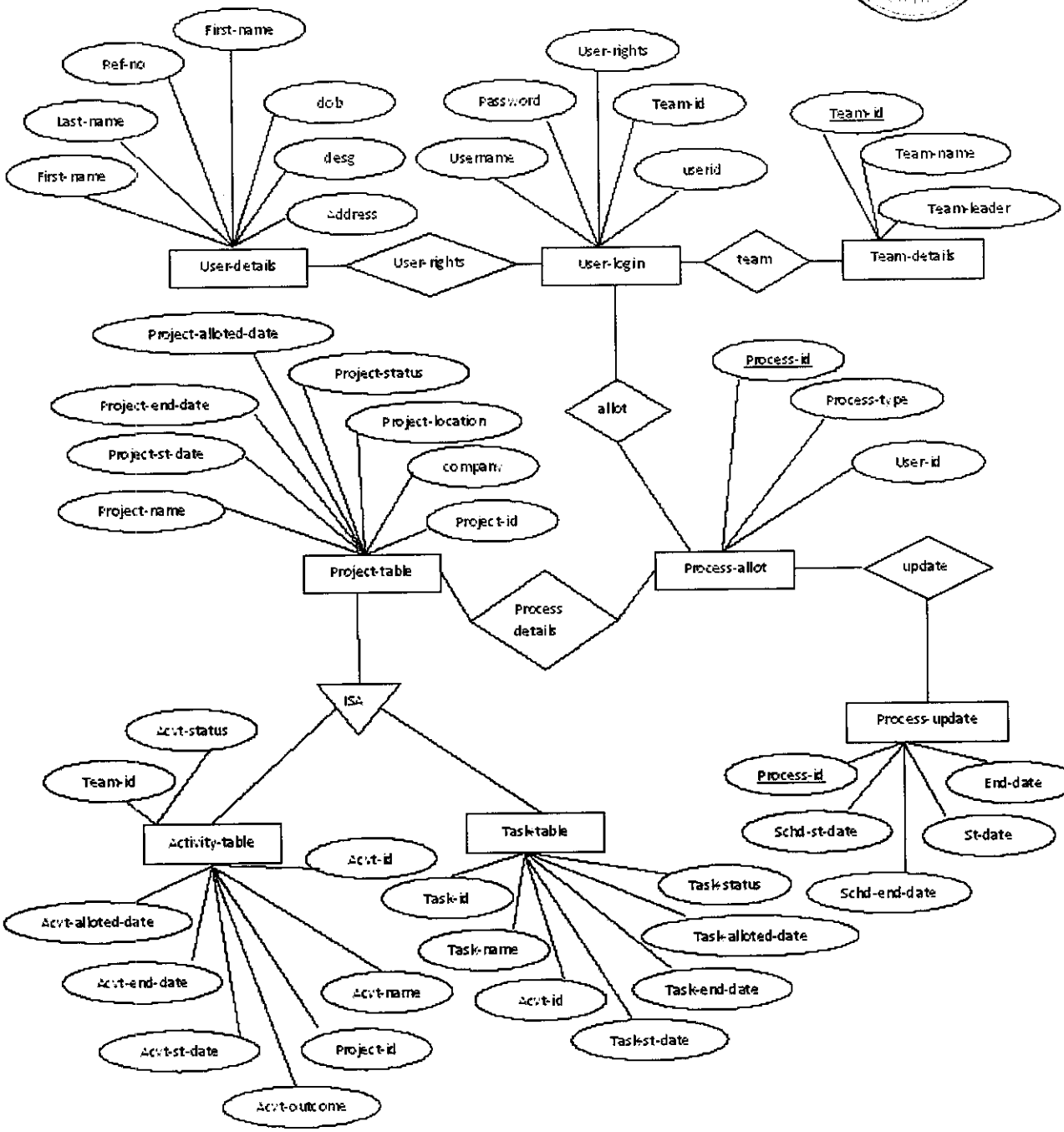
Developing the application needs VS 2005. To support running the application DotNet Framework 2.0 is needed to install in the Client. SQL Client need to be installed in Client System to access the SQL Server database and configuration are made related to the working environment.

# 4. SYSTEM DESIGN



## 4.1 Data Model

### 4.1.1 ER Diagram



### 4.1.2 Table Structure

**Table Name: Team**

This table is used to manage team details

| Field | DataType | Description | Key Values |
|---|---|---|---|
| team_id | int identity(1000,1) | Unique Identifier | Primary key |
| team_name | varchar(20) | Team Name | Not null |
| team_leader | varchar(75) | Team Leader Name | Not null |

**Table Name: User_details**

Table used to store personal details of the user

| Field | DataType | Description | Key Values |
|---|---|---|---|
| Id | int identity(1000,1) | Unique Identifier | Primary key |
| First_name | varchar(20) | Team Name | Not null |
| Last_name | varchar(75) | Team Leader Name | Not null |
| Dob | Date | Date of birth of the employee | Not null |
| Qualification | Varchar(20) | Educational qualification of the employee | Not null |
| Doj | Date | Date of joining of the employee | Not null |
| Desg | Varchar(20) | Designation of the employee | Not null |
| Mobile | Int | Mobile number | Not null |
| Residence | Int | Residence contact number | Not null |
| Address | Varchar(250) | Address of the Employee | Not null |
| Emailid | Varchar(100) | Email id of the employee | Not null |

**Table Name: User_login**

Table to store login details of the user, to change the user forms according to user rights.

| Field | DataType | Description | Key Values |
|---|---|---|---|
| User_id | Varchar(10) | User ID | Foreign Key |
| Password | varchar(30) | Password | Not null |
| User_rights | varchar(75) | User level of rights | Not null |
| Team_id | Int | User team identification | Not null |

**Table Name: project_table**

To store project details, company for which project is under taken.

| Field | DataType | Description | Key Values |
|---|---|---|---|
| project_id | int identity(1000,1) | Unique Identifier | Primary key |
| Project_name | varchar(100) | Team Name | Not null |
| Project_location | varchar(20) | Team Leader Name | Not null |
| Company | Varchar(30) | Comapany name for which project is under taken | Not null |
| Project_st_date | Date | Start date of project | Not null |
| Project_end_date | Date | End date of project | Not null |
| Project_alloted_date | Date | Date at which project is allotted to project manager | Not null |
| Project_status | int | Status of project | Not null |

**Table Name: module_table**

To store module details, scheduled date, status of the module and which project this module belongs to.

| Field | Data Type | Description | Key Values |
|---|---|---|---|
| Mod_id | int identity(10000 ,1) | Unique value identifies an module | Primary key |
| Mod_name | Varchar(30) | Module name | Not null |
| Mod_st_date | Date | Start date of the module | Not null |
| Mod_end_date | Date | End date of the module | Not null |
| Mod_completed_date | Date | Date at which the module really completed | Not null |
| Mod_status | Int | Status of the module | Not null |
| Project_id | Int | Project id, for which the module belongs to | Foreign Key |

**Table Name: activity_table**

To store details of activity and which module particular activity belongs to.

| Field | DataType | Description | Key Values |
|---|---|---|---|
| Acvt_id | int identity(10000 ,1) | Unique key identifies an activity | Primary key |
| Acvt _name | Varchar(30) | Name of the activity | Not null |
| Acvt _st_date | Date | Start date of the activity | Not null |
| Acvt _end_date | Date | End date of the activity | Not null |
| Acvt _completed_date | Date | Date at which activity really completed | Not null |
| Acvt _status | Int | Status of the activity | Not null |
| mod_id | int | Identifies which | Foreign Key |

**Table Name: process_table**

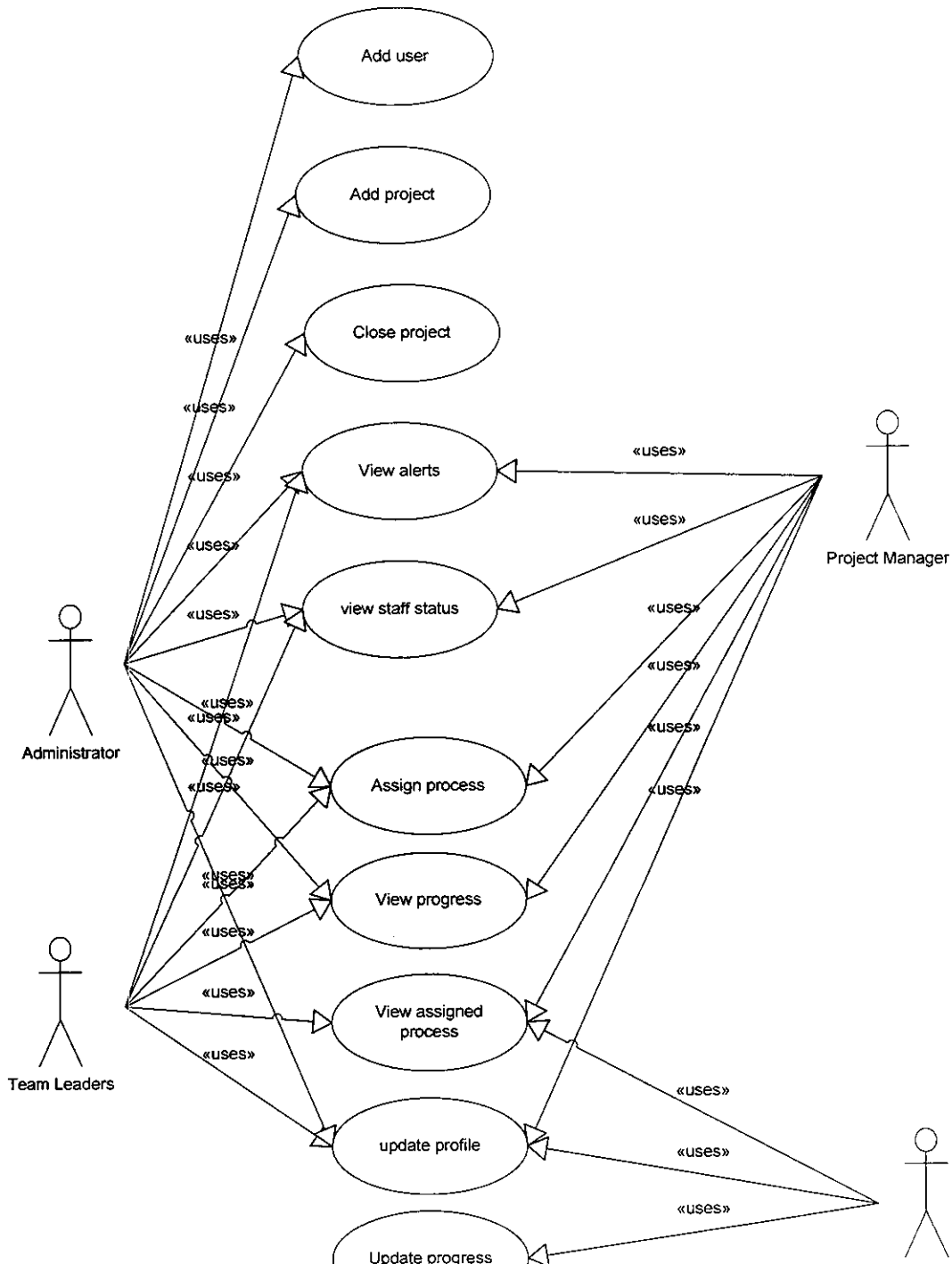
To store all process (project, module and activity) and details of the work assigned for particular employee. Helps to keep track of the work progr ss.

| Field | DataType | Description | Key Values |
|---|---|---|---|
| Process_id | Int | Unique key identifies all process(activity, module, project) | Primary Key |
| Process_type | Int | Type of process(activity, module, project) | Not null |
| User_id | Varchar(10) | User id, to identify the allotted process | Not null |
| Schd_st_date | Date | Scheduled start date of the process | Not null |
| schd st_date | Date | Scheduled end date of the process | Not null |
| St_date | Date | Realtime start date of the process | Not null |
| End_date | Date | Realtime end date of the process | Not null |
| Process_status | Date | Status of the process | Not null |

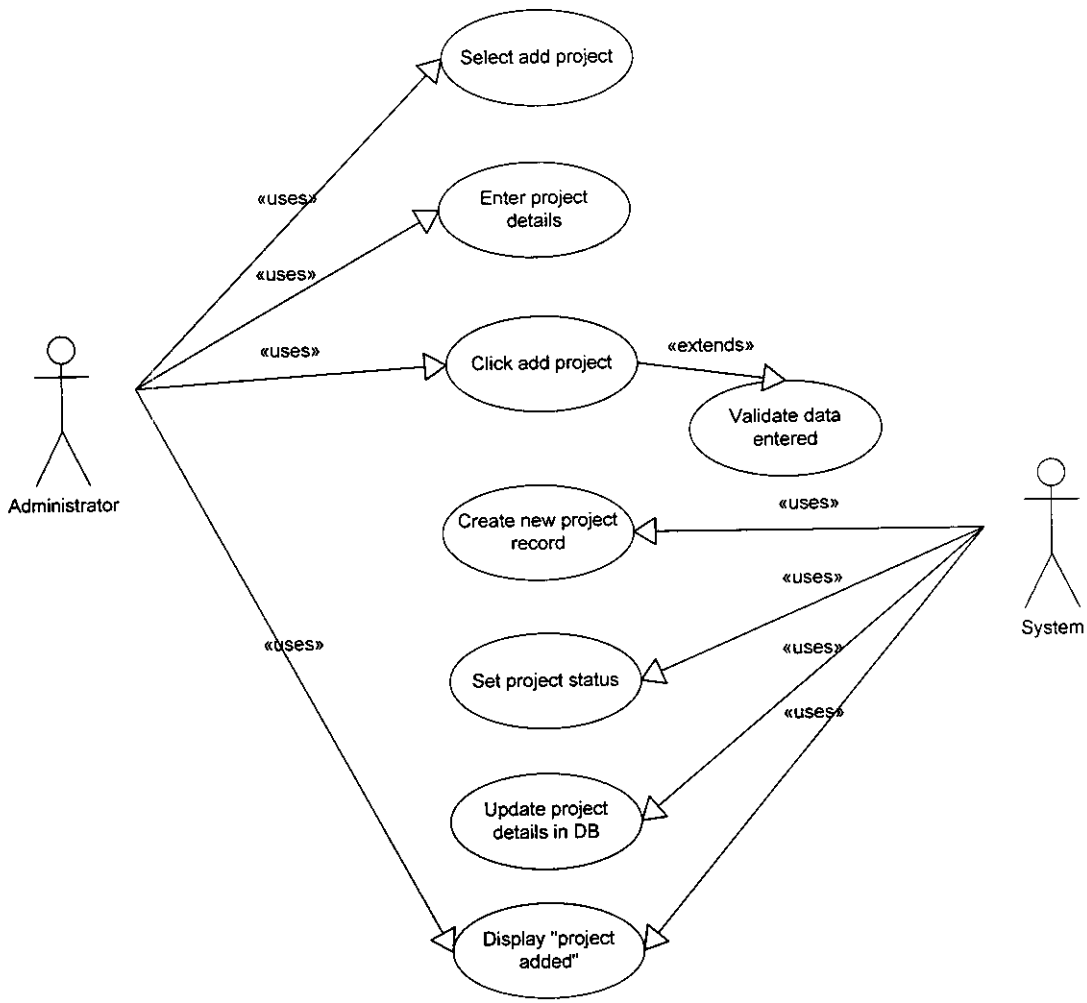
## 4.2 Process Model

### 4.2.1 Use Case Diagram

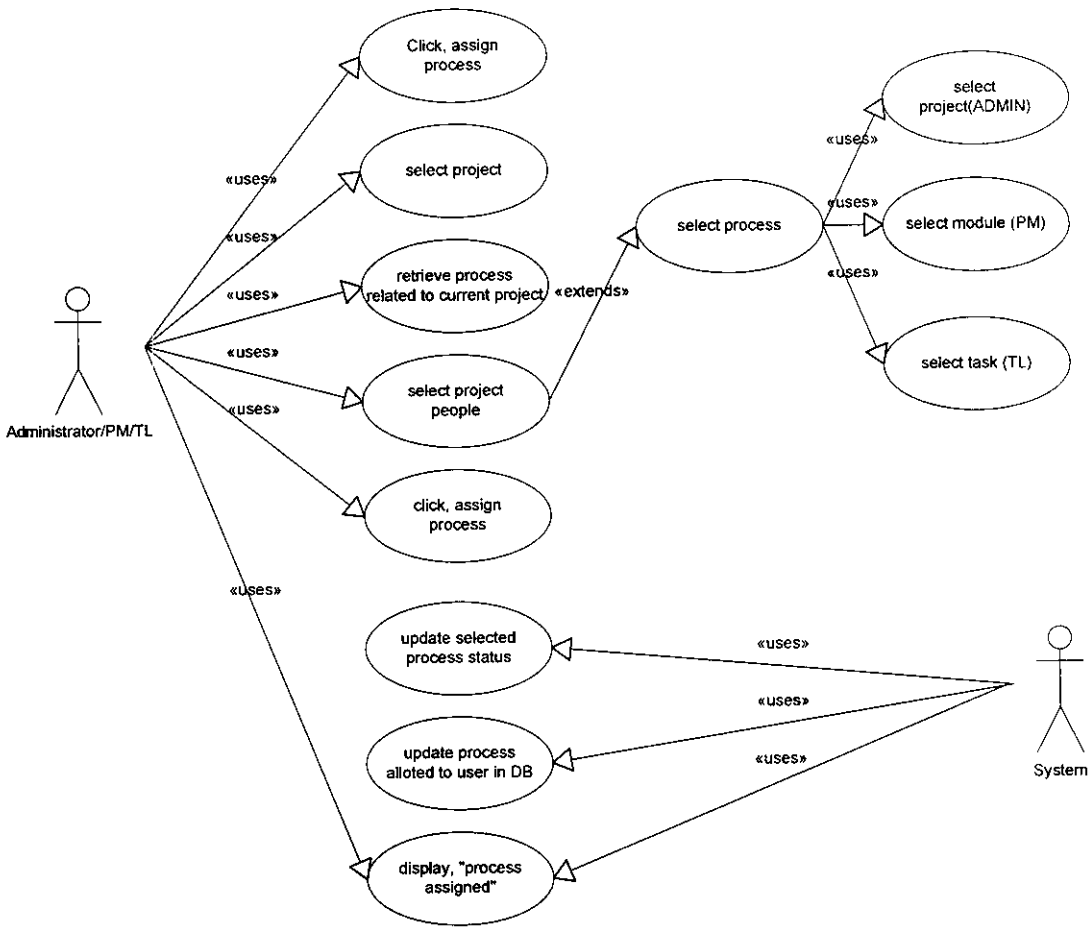
### Task Tracking System



## Add New Project



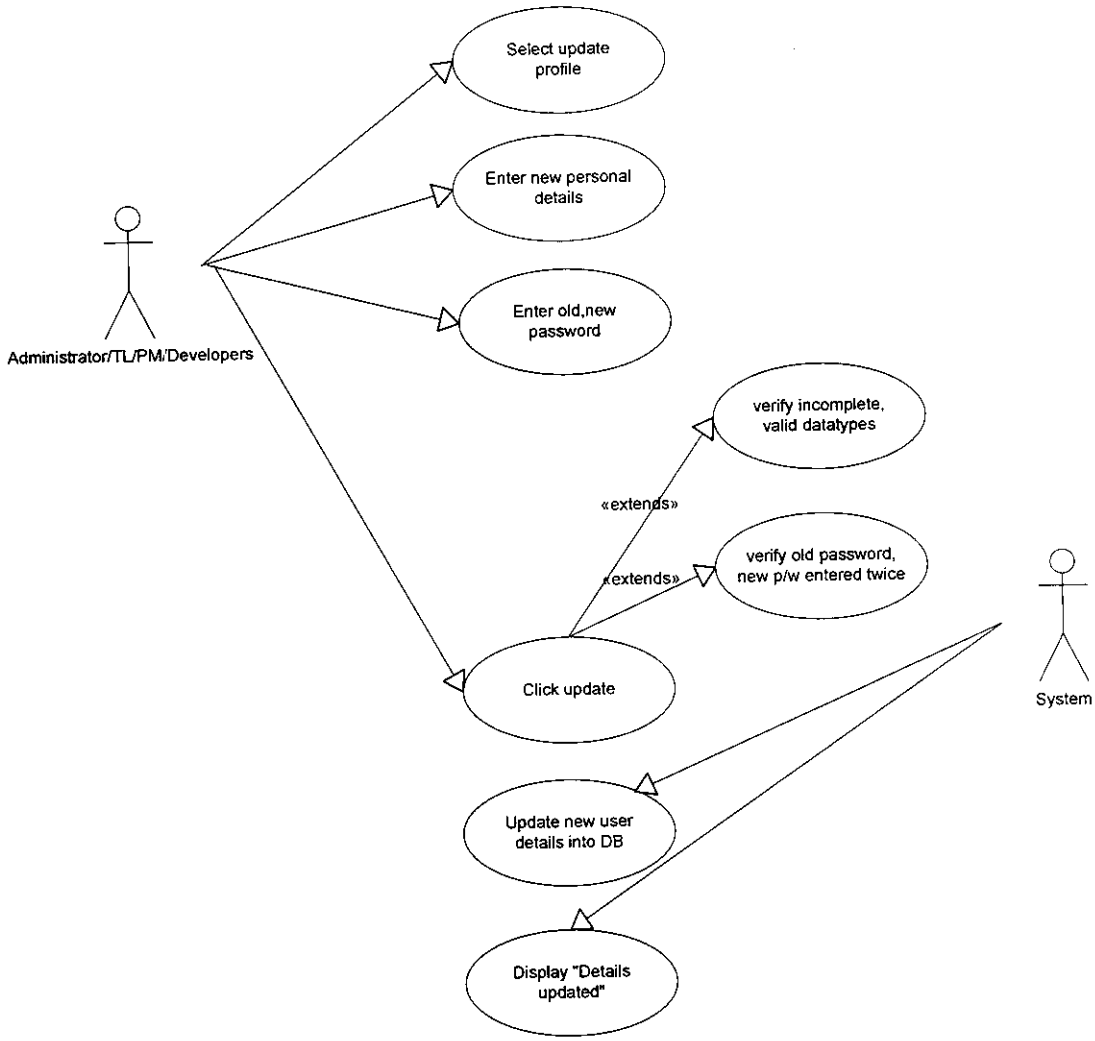
## Assign Process



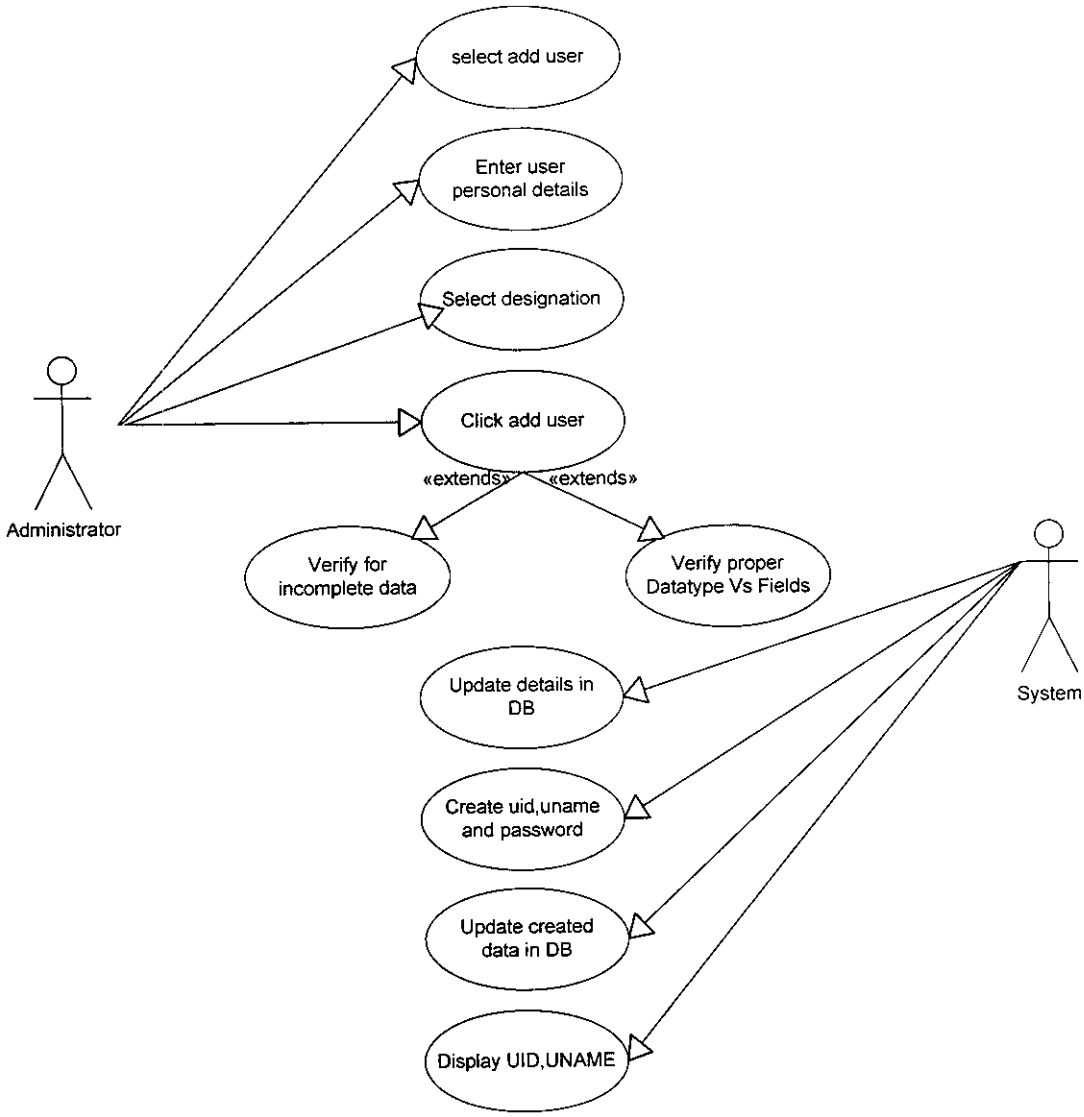
## View Staff Status

## Update Process Progress

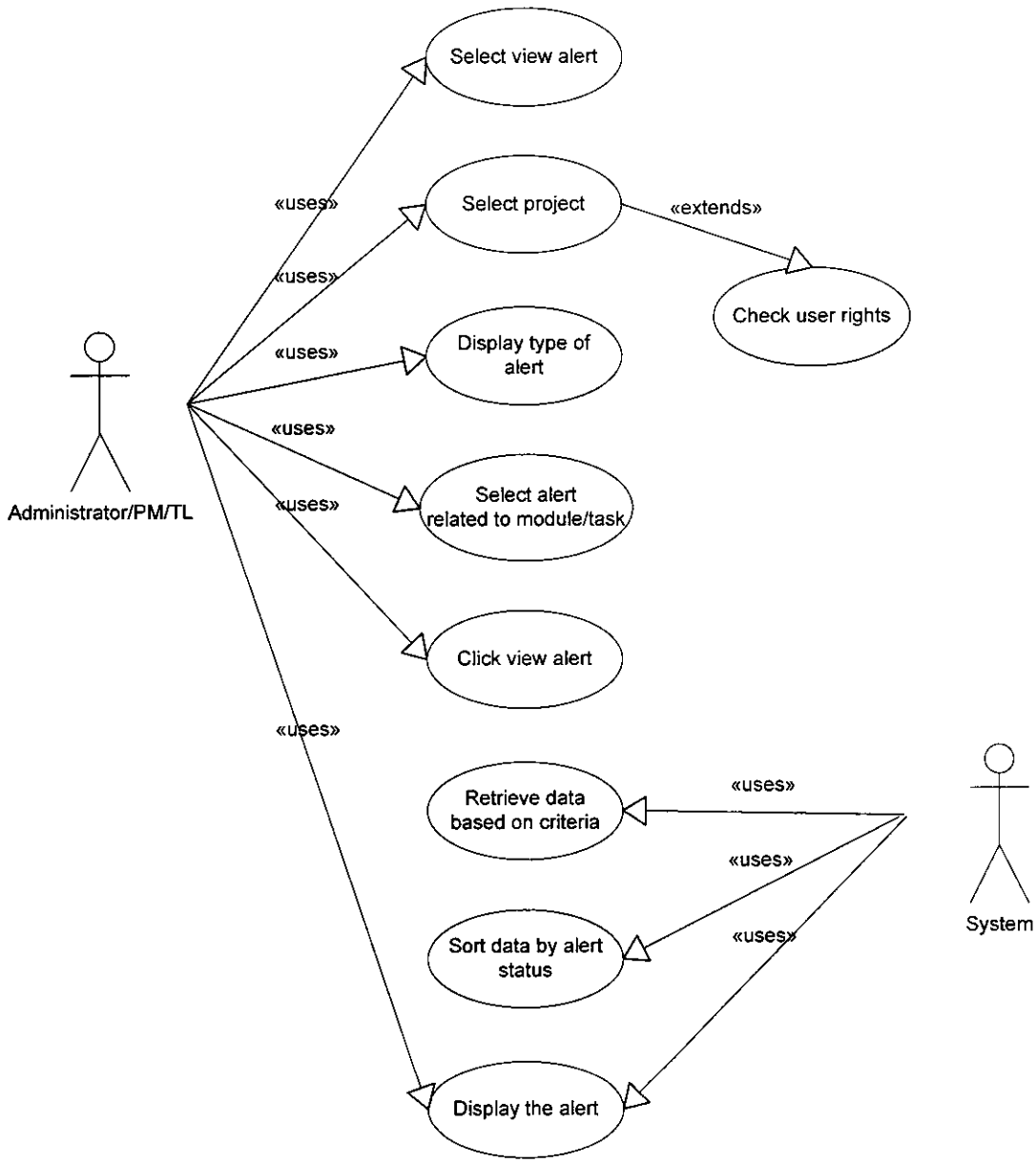
## Update User Profile



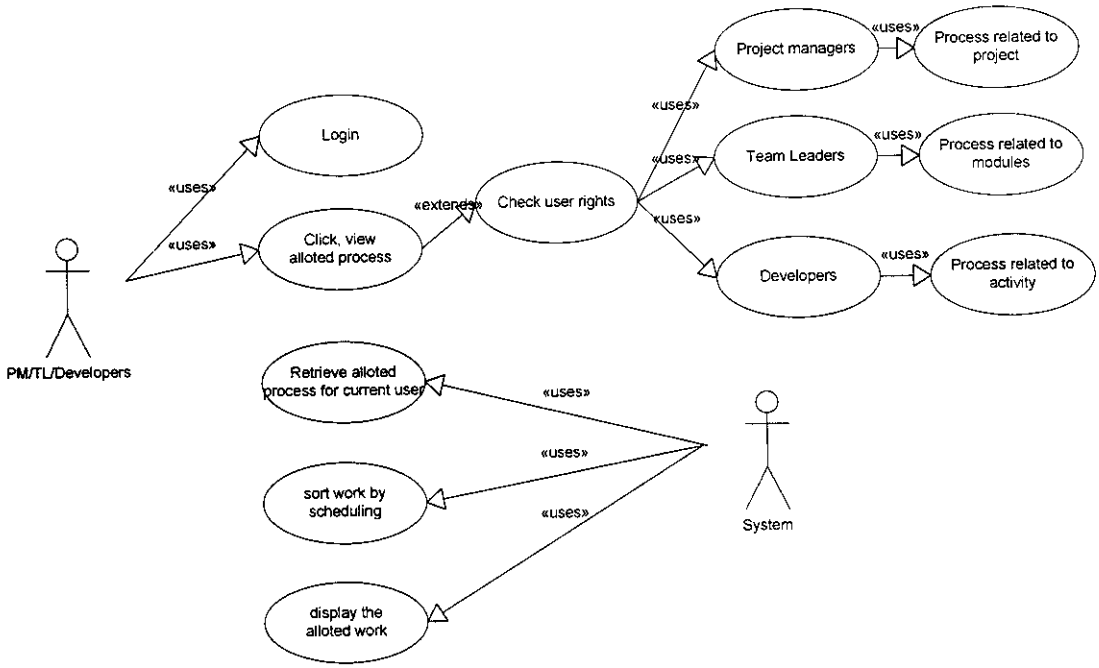
## Add New User


select add user
Enter user personal details
Select designation
Click add user
«extends»
«extends»
Administrator
Verify for incomplete data
Verify proper Datatype Vs Fields
System
Update details in DB
Create uid,uname and password
Update created data in DB
Display UID,UNAME

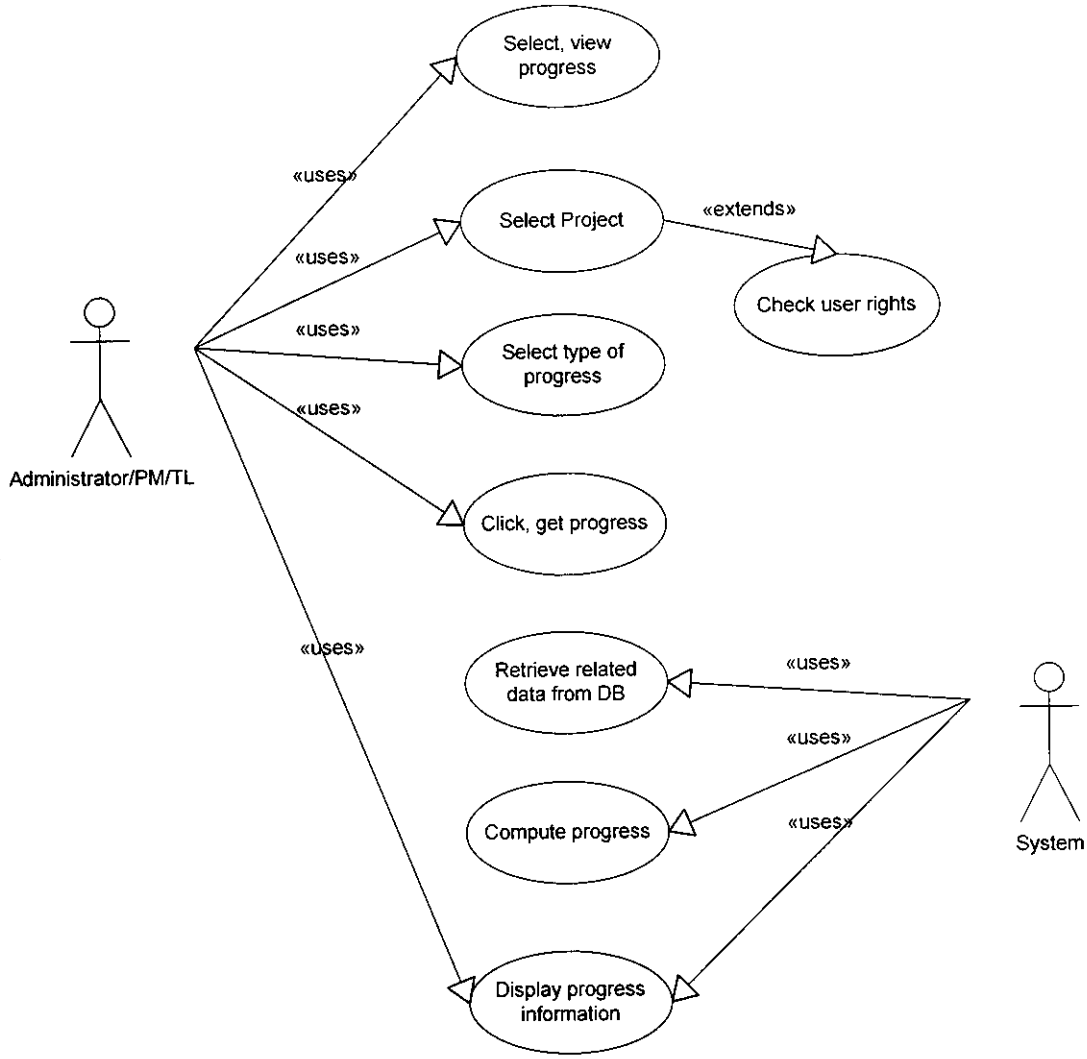
## View Alerts


Select view alert
Select project
«extends»
Check user rights
«uses»
«uses»
«uses»
«uses»
«uses»
«uses»
Display type of alert
Select alert related to module/task
Click view alert
Administrator/PM/TL
Retrieve data based on criteria
Sort data by alert status
Display the alert
«uses»
«uses»
«uses»
System

## View Allotted Process



## View Progress



# 5. ARCHITECTURAL DETAILS

## 5.1 Program Design language

The programming part of this system is done using VB.NET. And the database related queries are written using simple SQL and combined DotNet programming style. VB.NET support SQL related queries by implementing sqlclient Data Object of DotNet framework.

The system uses DotNet framework objects like Data to implement Sqlclient. This is to interact and manipulate sql queries. Crystal Reports to generate reports used for getting status of project. For manipulating and processing data on database the system. Dataset is used to get set of data that was an output of a query. DataAdapter is a connection that was made to interact with the database to manipulate on records. Datatable is used to create manual table that forms basis for Dataset. Or to format data that was extracted via Dataset. Sqlcommand is used to process the queries, to load or to store data into database. Sqlconnection to form connection to the SqlServer DB. Sql Adapter to process the query and retrieve the resultant data.

Datagrid are used to populate the data in form. Datagrid source will be Dataset. The Datagrid is very useful to view the data that was stored in Dataset. Crystal report viewer is used to view the reports that can be printed. The Crystal report document is one which holds the data that is to be displayed in Report viewer. We can format the fields in Report document to view report in user friendly format.

## 5.2 System Architecture

System can be implemented in standalone environment were the Database and the application runs in same system. The system also supports Database to run in server system and application access the database via Sqlclient interface. The system is not specifically designed for an environment. Since the system not uses any of network environment objects. And system uses and front end application and back database. Database can be accessed by configuring the client and server. The system

# 6. SYSTEM IMPLEMENTATION

Implementation is the state in the System where the theoretical design is turned into a working system. The system can be implemented only after through testing is done and if found to work according to the specification. The most crucial stage in achieving a new successful system relies in giving confidence for the users on the new system that will work efficiently and effectively.

It involves careful planning, investigation of the current system and to constraints on implementation, design of methods to achieve the changeover, an evaluation of changeover methods apart from planning. System Analysis and design efforts will be more complex system being used for writing program code.

**Program Code Preparation**

One of the important development activities is the code of programming. The system Use Case Diagram and other channels are converted to modular programs; they have to be complied, tested and debugged.

## 6.1 Configuring the System

The Architecture of the system includes Application running in client system and Database that runs either in Client system or in Server. If the client system runs or holds the centralized Database the SQL Server environment should be installed and Database for system should be created or installed using backups. If the system runs under server and Database is centralized and Runs in Server, then the SQL Server should be installed in the Server system and configured to support the server environment. And the SQL Native Client system should be installed in all the client system running the application, in order to access the Database in server.

Sqlclient should be configured to server in order to access the database. The Alias property is set name of the client in server so as to give access the database in server. And the port number should be given to interact with server so as to access the active server.

# 7. TESTING

## 7.1 Testing an Overview

**Introduction on Testing:**

Software testing is a critical element of software quality assurance and represents the ultimate reviews of specification, design and coding testing represents interesting anomaly for the software. During earlier definition and development phases, it was attempted to build software from an abstract concept to tangible implementation.

The testing phase involves the testing of developed system using various test data. Preparation of the test data plays vital role in the system testing. After preparing the test data the system under study was tested using those data. While testing the system, errors were found and corrected by using the following testing steps and corrections are also noted for future use. Thus, a series of testing is performed for the proposed system was ready for the implementation.

**Unit Testing:**

Unit testing focuses verification efforts even in the smallest design in each module. This is also known as "Unit Testing". Since the proposed project has four modules, the testing is done individually on each module and every form designing.

- Unit testing is done for each form. Once a form is to be tested, in solution property of the project, startup form is set to the currently designing form.
- Processing of data has steps like, connecting to DB, creating DataAdapter, Setting Command text, finally filling the resultant data into dataset.
- For every process appropriate message boxes are added to test at which step the process fails or stops. It will be useful to identify the bug in the process flow.

- Connection level errors may be, database not created, wrong database path or instance. Errors in command text may be syntax error or wrong type of datatype linked to table.
- And exception handling is used to handle the unknown error and exception is displayed in message box.
- After processing data, to known the process changed the database the related data are view using Datagrid. To confirm the process updated the DB.

**Integration Testing:**

Data can be lost across an interface, one module can have an adverse effect on the other sub-functions, when combined may not produce the desired functions. Integrated testing is the systematic testing to uncover the errors within the interface.

- After designing all four modules sample data are given and tested for this testing.
- For login form it is tested with various username and password. To known it processes the user login correctly. And to known any user getting wrong privilege access to the system.
- To know the data (scheduling, regular update) are updated correctly sample scheduling data are given and process is assigned to various users. And logging as different user, work assigned are viewed.
- And as user level as developer, data like process started and process completed are selected and updated. Then at high level user (TL) the process progress is viewed inorder to known the process updation done by the developer are updated in database.
- Data at modified at different levels of user, to known the DB getting updated and users are getting updated information and not the old information.

**White Box Testing:**

White box testing some times called glass box testing, is a test case design method that uses the control structure of the procedural design to derive test cases. Using white box testing methods, the software engineering can derive test cases.

Few test cases are listed in the chapter 7.2, where random inputs area given and tested for errors.

## 7.2 Test Case Reports

**Test Case: Assign Process**

| Test Case ID | Test Case Description | Procedure | Input | Actual Output | Expected Output | Result |
|---|---|---|---|---|---|---|
| AP001 | Data not Input | Leave the Project Name Field Empty | Nil | Error Message | Error Message | Pass |
| AP002 | Data not Input | Not Select any data for Project Location field | Nil | Error Message | Error Message | Pass |
| AP003 | Data not Input | Don't select or Input Company Name | Nil | Error Message | Error Message | Pass |
| AP004 | All Data are Given | Give or Select all mandatory data fields | Smart – Banking Application, Olympia, Intertek Solutions | Message "Project added" | Message "Project added" | Pass |
| AP005 | Exit without giving add project | After giving all correct data, exit from | Smart – Banking Application, Olympia, Intertek | Message "Do u want to quit without | Message "Do u want to quit without | Pass |

| | | | | | | |
|---|---|---|---|---|---|---|
| AP006 | Give Exiting Information | Give same Project Name already in Use | Smart – Banking Application | Message "Project Already Exists – Input Correct Name" | Message "Project Already Exists – Input Correct Name" | Pass |
| AP007 | Special Characters | Give Special Character in Project Name | VDI-Rec! | Message "unwanted special character" | Message "unwanted special character" | Pass |
| AP008 | Edit Combo box field | Edit the fixed combo box field of project location | RR Towers | Field not change by manual typing | Field not change by manual typing | Pass |

**Test Case: Update Progress**

| Test Case ID | Test Case Description | Procedure | Input | Actual Output | Expected Output | Result |
|---|---|---|---|---|---|---|
| UP001 | Empty Field | Don't select start process field | Nil | Message "Please select the Process" | Message "Please select the Process" | Pass |
| UP002 | Empty Field | Don't select complete process field | Nil | Message "Please select the Process" | Message "Please select the Process" | Pass |
| UP003 | Correct Input | Select an item from start process field | ACT10005 - Fetch User Pool | Message "Start process – Updated" | Message "Start process – Updated" | Pass |
| UP004 | Correct Input | Select an item from complete process field | ACT10005 - Fetch User Pool | Message "Complete process – Updated" | Message "Complete process – Updated" | Pass |
| UP005 | Exit without committing | Select an item in start process | ACT10005 - Fetch User Pool | Message "Do you want to | Message "Do you want to | Pass |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | current form | | saving data" | saving data" | |
| UP006 | Enter Redundant Data | Select already Started process and click Start Process | ACT10005 - Fetch User Pool | Message "Warning – Already existing Process" | Message "Warning – Already existing Process" | Pass |
| UP007 | Enter Redundant Data | Select already Completed process and click Start Process | ACT10005 - Fetch User Pool | Message "Warning – Already existing Process" | Message "Warning – Already existing Process" | Pass |
| UP008 | Edit Combo box field | Edit the fixed combo box field Start Process | ACT10009 - | Field not change by manual typing | Field not change by manual typing | Pass |

**Test Case: Assign Module**

| Test Case ID | Test Case Description | Procedure | Input | Actual Output | Expected Output | Result |
|---|---|---|---|---|---|---|
| AM001 | Empty Field | Don't Select Project Name field | Nil | Message "Please select the Project" | Message "Please select the Project" | Pass |
| AM002 | Empty Field | Don't Select Module Name field | Nil | Message "Please select the Module" | Message "Please select the Module" | Pass |
| AM003 | Empty Field | Don't Select Team Leader Name | Nil | Message "Please select Team Leader" | Message "Please select Team Leader" | Pass |
| AM004 | Correct Input | Give or Select all Mandatory Inputs | Virtual Desktop Interface – VDI, Arun Balaji, | Message "Module Assigned" | Message "Module Assigned" | Pass |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | Server - MOD10 01 | | | |
| AM005 | Redundant Data | Input Already Assigned module to other Team Leader | Virtual Desktop Interface – VDI, Allen, Connect to Server - MOD10 01 | Message "Module Already Assigned" | Message "Module Already Assigne d" | Pass |
| AM006 | Discard Data | Click Discard, as wrongly data are Selected | Virtual Desktop Interface – VDI, Allen, Connect to Server - MOD10 01 | Message "Do you Want to Discard the Input Data" | Message "Do you Want to Discard the Input Data" | Pass |
| AM007 | Select Yes in Warning message | Select "Yes in Warning Message of Test Case – AM006 | YES | Clear all Select Data and Focus in Select Project Field | Clear all Select Data and Focus in Select Project Field | Pass |
| AM008 | Select No in Warning message | Select "No in Warning Message of Test Case – AM006 | NO | Data Input Remains Unchanged | Data Input Remains Unchan ged | Pass |
| AM009 | Edit Combo box field | Edit the fixed combo box field Select Team Leader Name | Sandeep | Field not change by manual typing | Field not change by manual typing | Pass |
| AM010 | Exit without committing | Select all Correct data and Click Exit Button | Virtual Desktop Interface – VDI, winston, Load Setting - | Message "Do you want to exit without saving data" | Message "Do you want to exit without saving data" | Pass |

**Test case: View Progress**

| Test Case ID | Test Case Description | Procedure | Input | Actual Output | Expected Output | Result |
|---|---|---|---|---|---|---|
| VP001 | Correct Data | Give or Select all Mandatory Data | Virtual Desktop Interface – VDI, Activity, DEV_PHASE1 | Data are populated and viewed in Data Grid | Data are populated and viewed in Data Grid | Pass |
| VP002 | Edit Data in Data Grid | Try to Edit the Populated data in Data Grid. In Status Field | Pending | Data Remains unchanged | Data Remains unchanged | Pass |
| VP003 | Empty Field | Don't Select the Project Name | Nil | Message "Select the Project" | Message "Select the Project" | Pass |
| VP004 | Empty Field | Don't Select the Progress Type | Nil | Message "Select Progress Type" | Message "Select Progress Type" | Pass |
| VP005 | Empty Field | Don't Select the Select Team/Individual | Nil | Message "Select Team/Individual" | Message "Select Team/Individual" | Pass |
| VP006 | Edit Combo box field | Edit the fixed combo box field Select Progress Type | Task | Field not change by manual typing | Field not change by manual typing | Pass |
| VP007 | Empty Fields | Don't Select the Project Name, Progress Type but Select Project Name | Nil, Nil, DEV_PHASE3 | Message "Select the Project, Progress Type" | Message "Select the Project, Progress Type | Pass |

# 8. PERFORMANCE LIMITATIONS

## 8.1 Merits of the System

- Stores huge information and handles the information very simply since the information are stored in related tables.
- Back-End and Front-End are separated which improves the information security.
- Reduces the complexity in the process followed in proposed system.
- Information Updation at centralized database will reflect in all related tables.
- All kind of information's is retrieved by management which gives idea to make timely decisions.
- System is user friendly to use and gives all kind of informative data which are needed for all levels of user.
- Multiple users are supported by the system, which can run the application at different user systems.

## 8.2 Limitations of the System

- Needs the scheduling data to be imported from an external file. Since system doesn't provides any functionality to input all these data.
- System handles the database stored in a server, if the real time environment has more than one server at a location; system needs to be configured according to the user environment.
- If server location or information gets changed, the information should be updated in all the user work stations.
- Scope of the system is it only handles the operations (coding/designing) phase of the Software Project Management. And not the analysis, testing or maintenance phase of the system development.

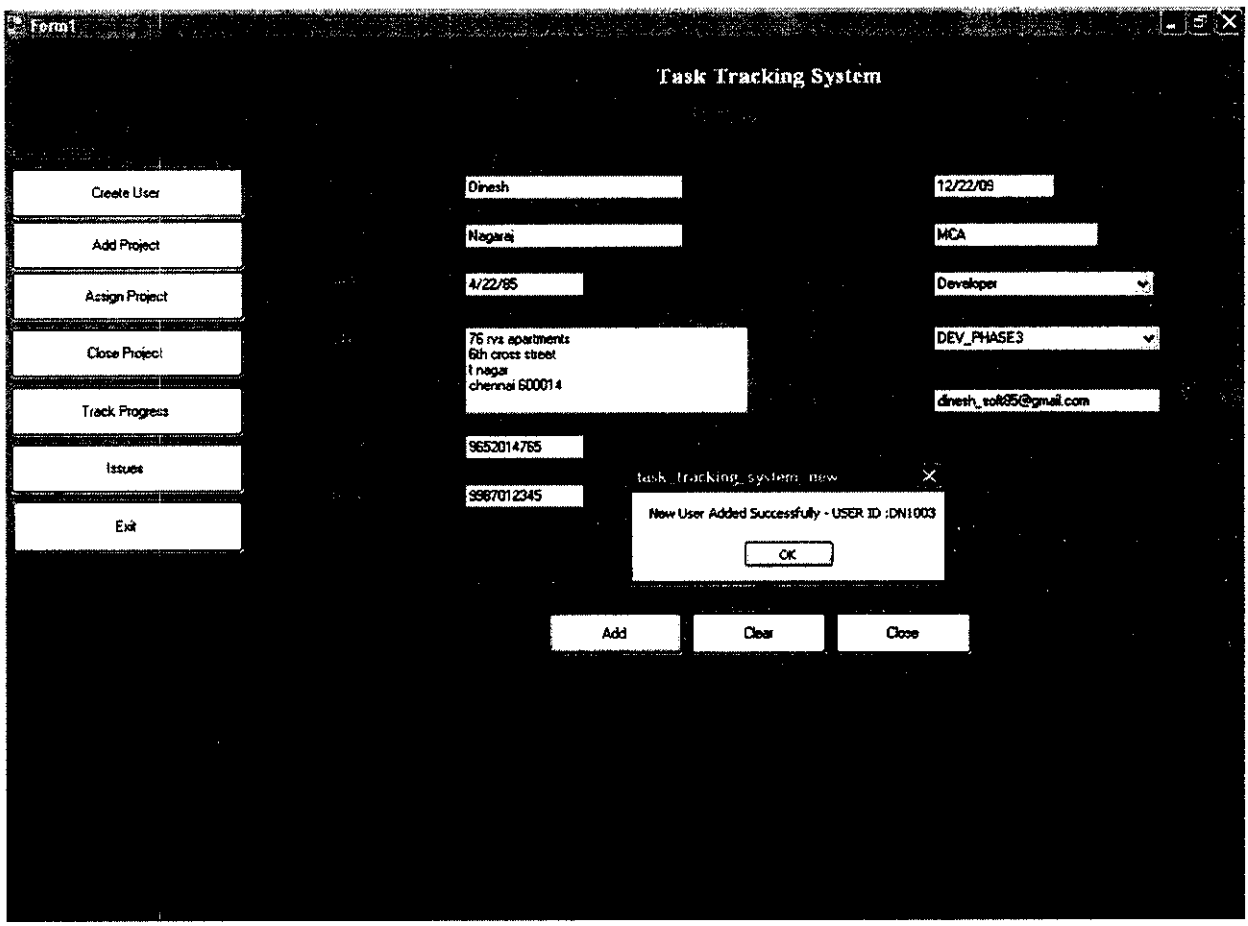
### 8.3 Future Enhancements

- The system can be developed to extend the functionality to cover all the phases in software development cycle.
- And functionality can be provided for input scheduling information into system. Needs a user type Data Entry operator, interface need to be provided to handle this function.
- Graphical information like charts, exporting report in other formats like word, excel and pdf can be implemented in the enhanced system. Since these will provide useful information to End-User.

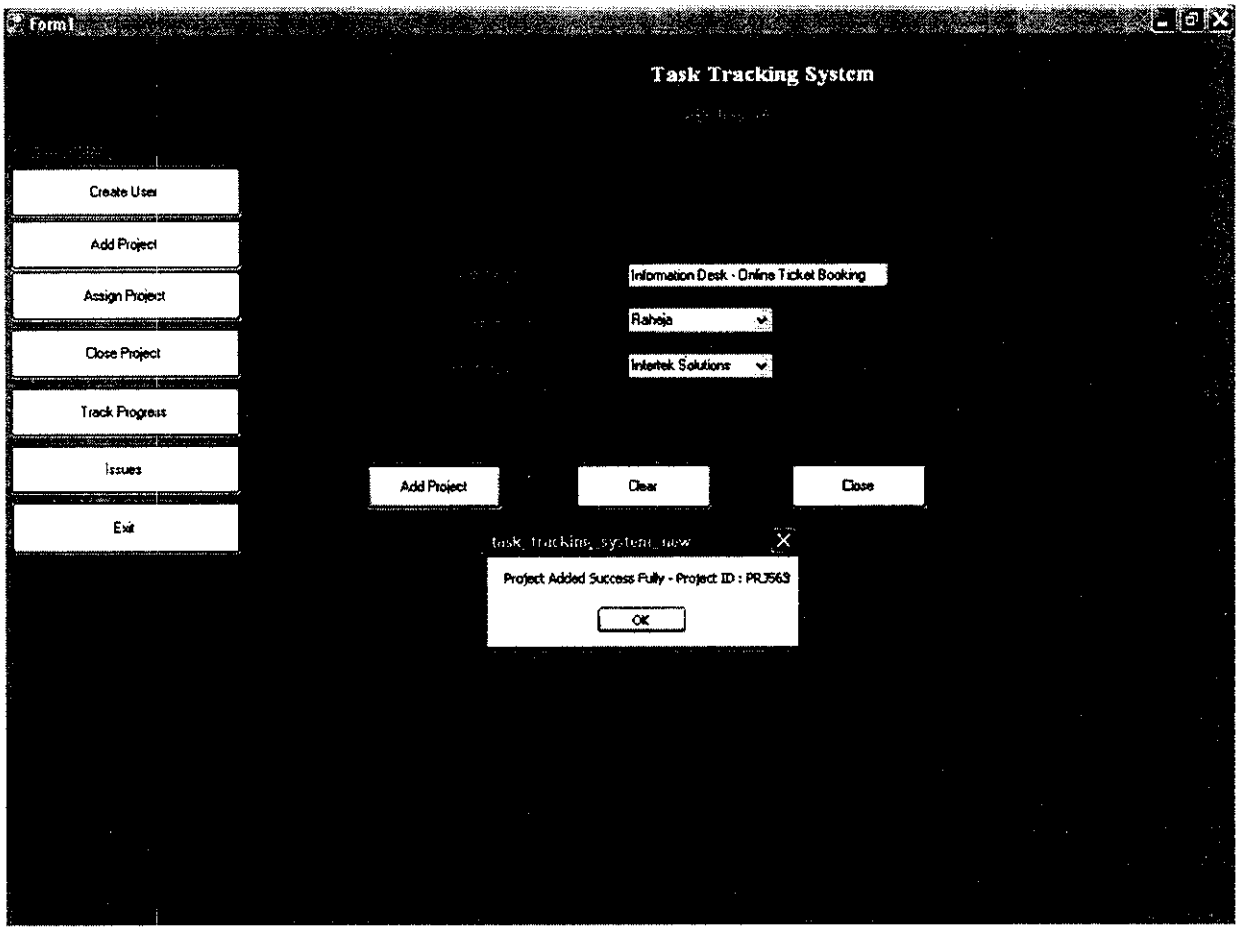
# 9. APPENDICES

## 9.1 Sample Screen Shots

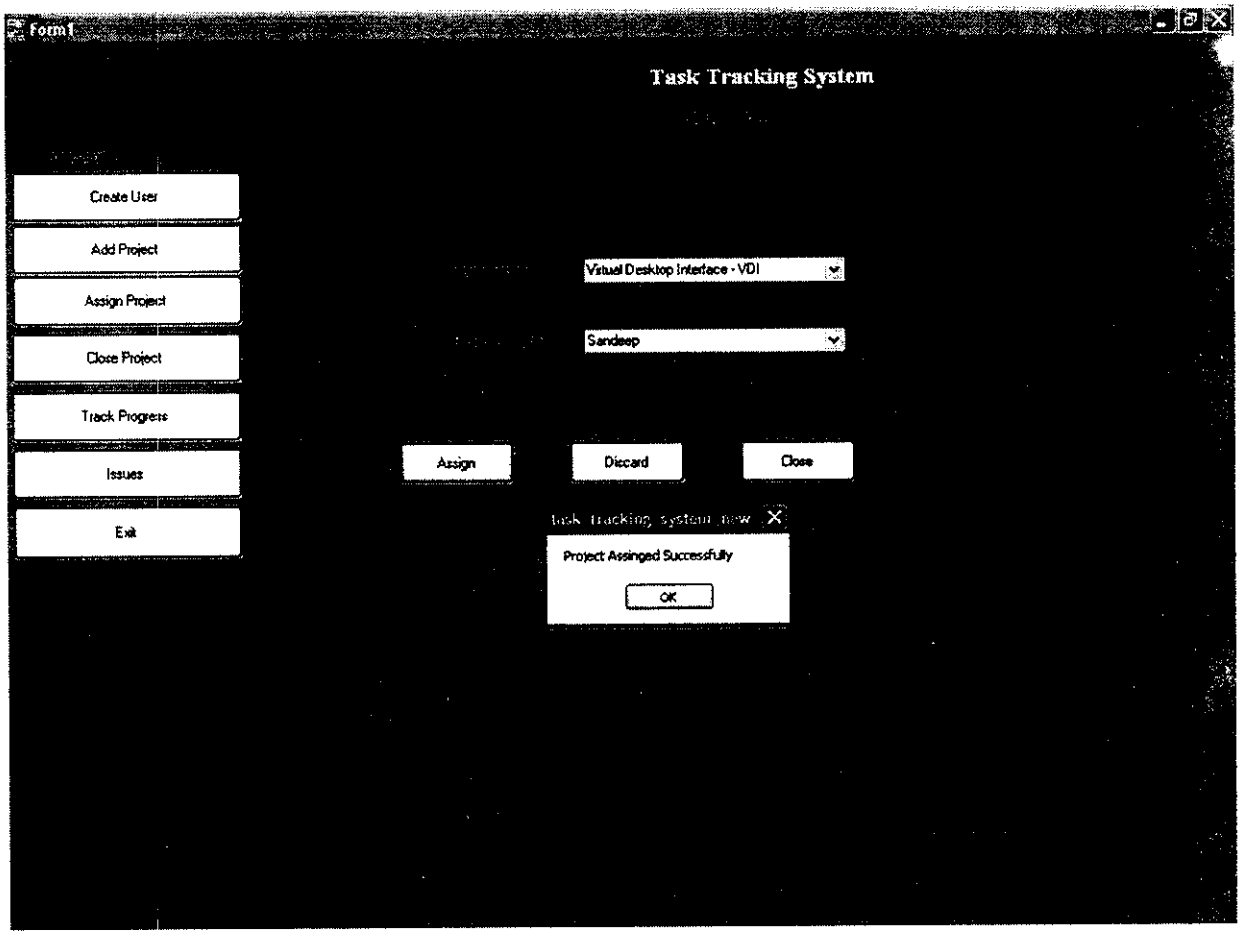
### Add New User



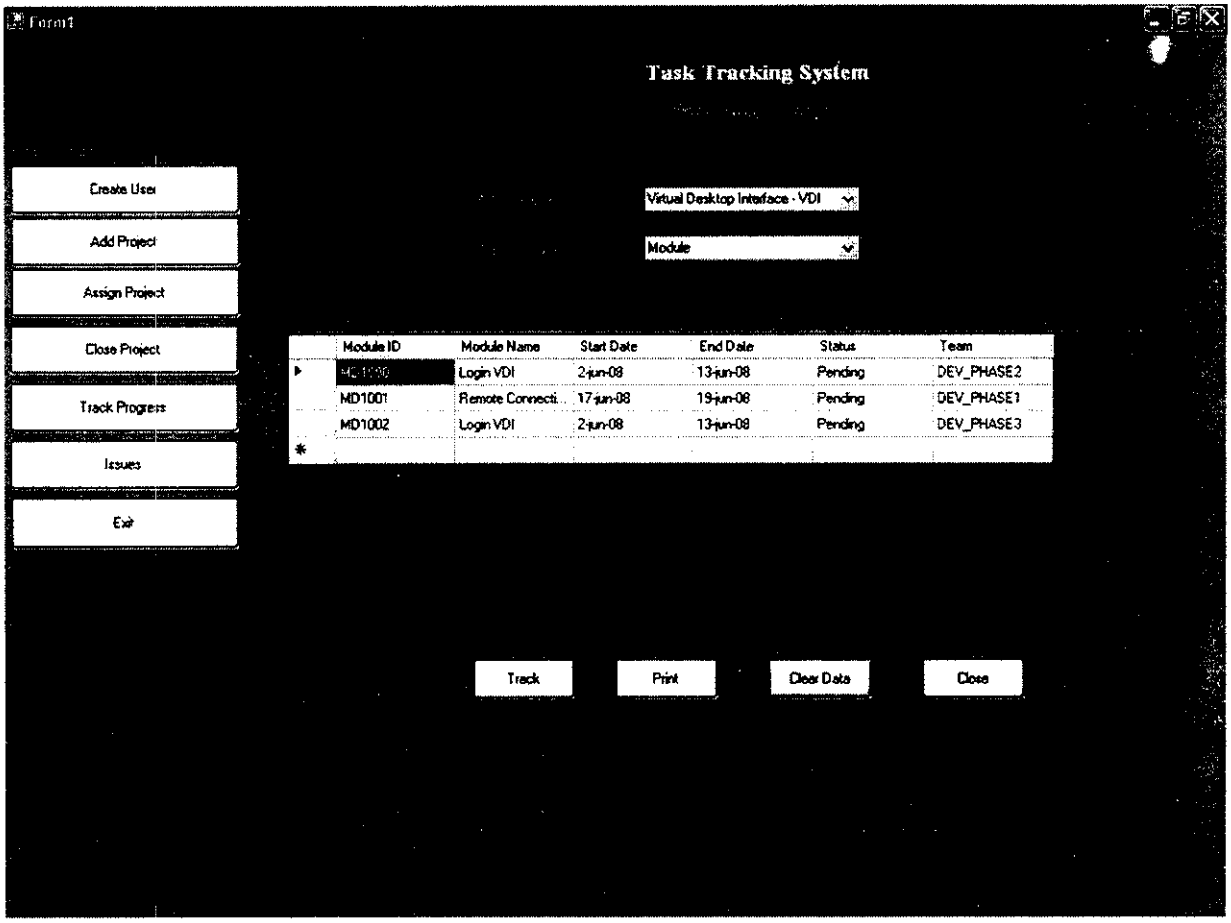
## Add New Project



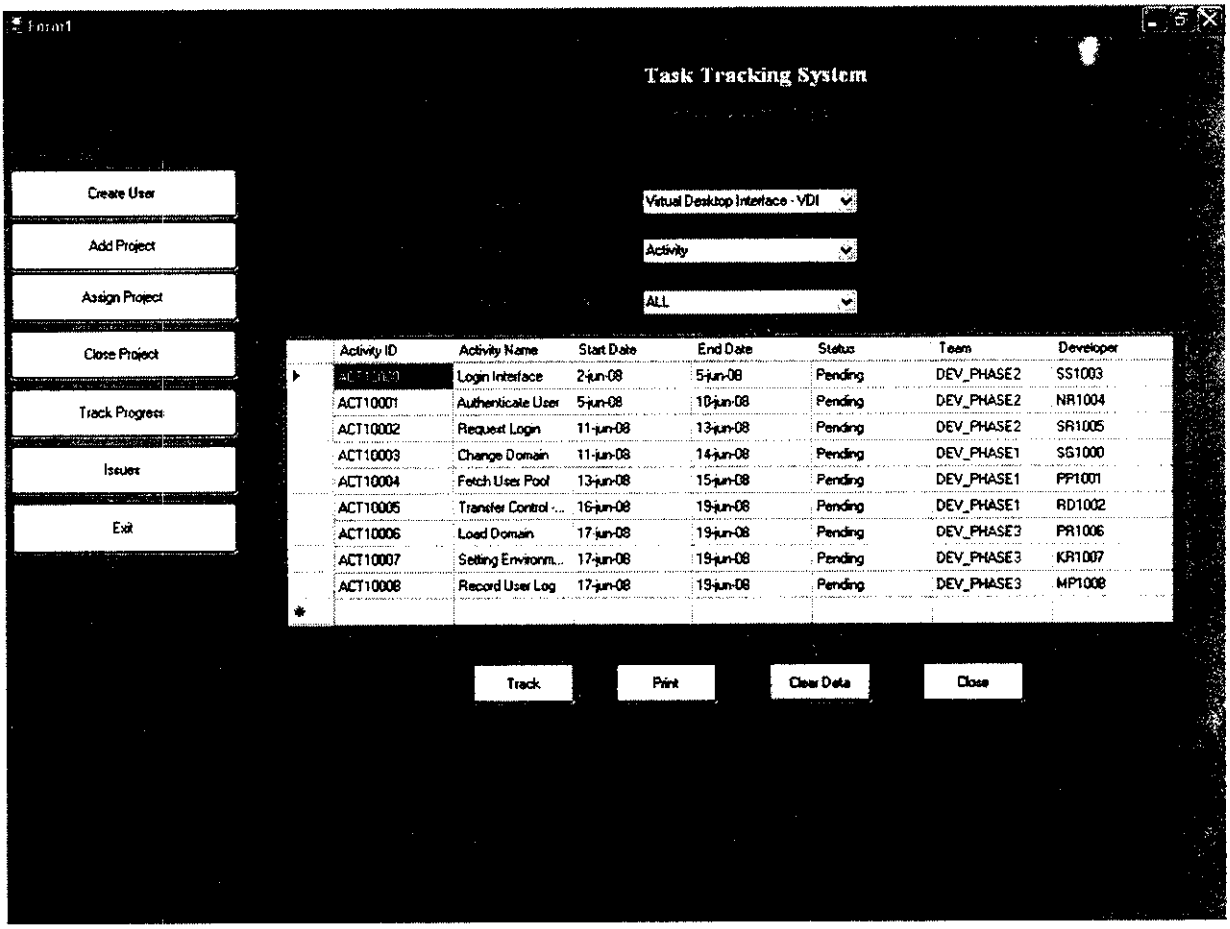
## Assign Project



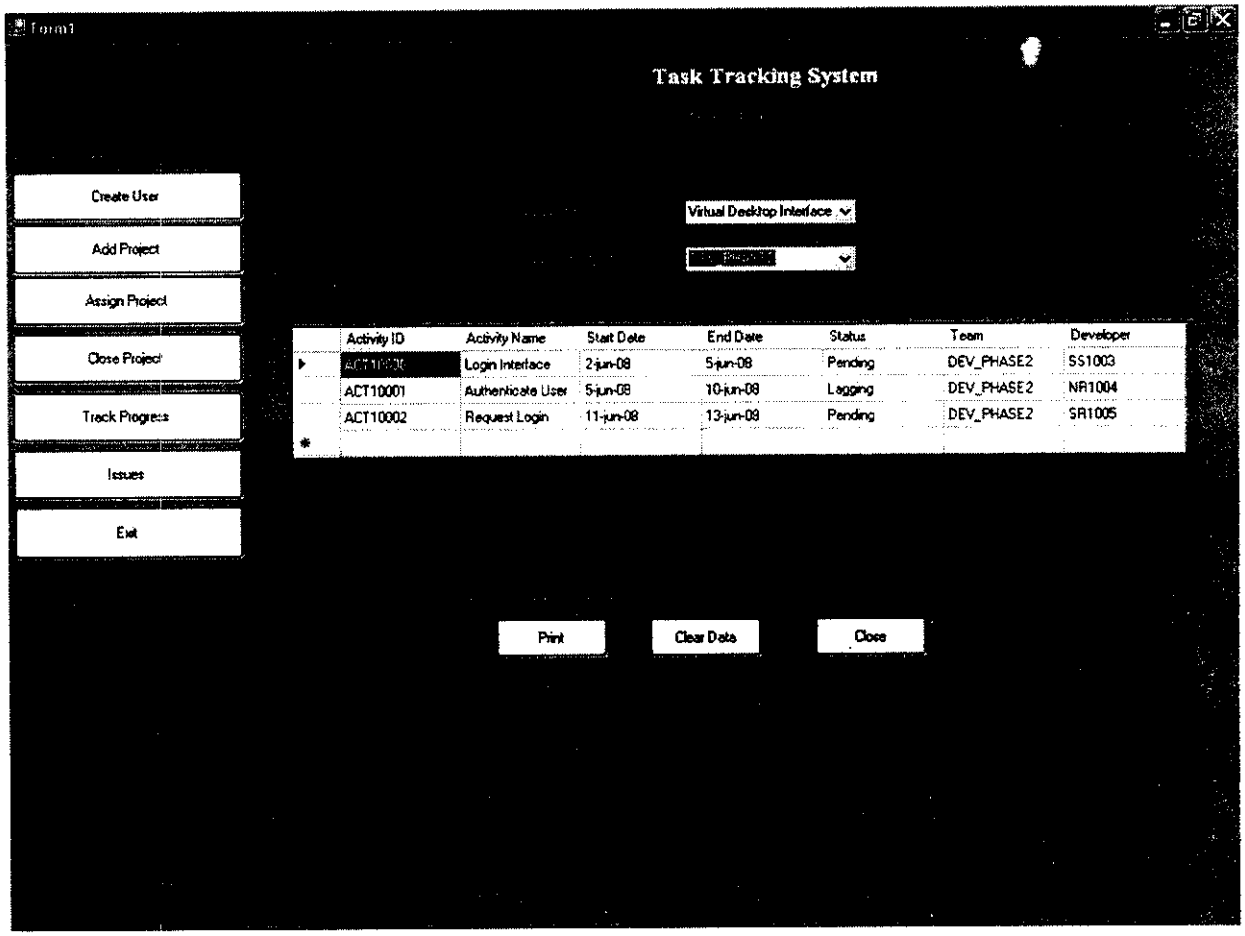
## Track Progress - Module



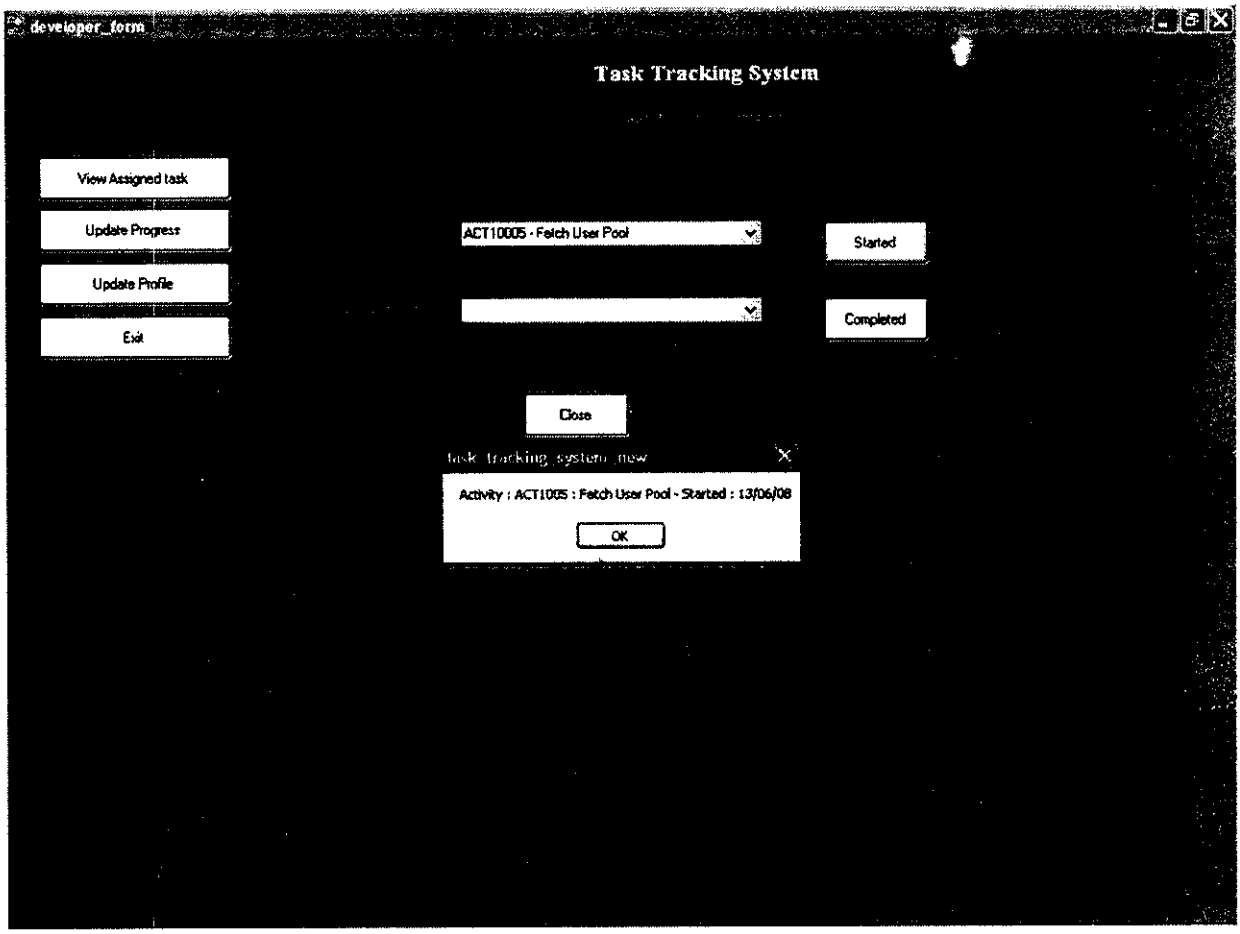
## Track Progress – Activity



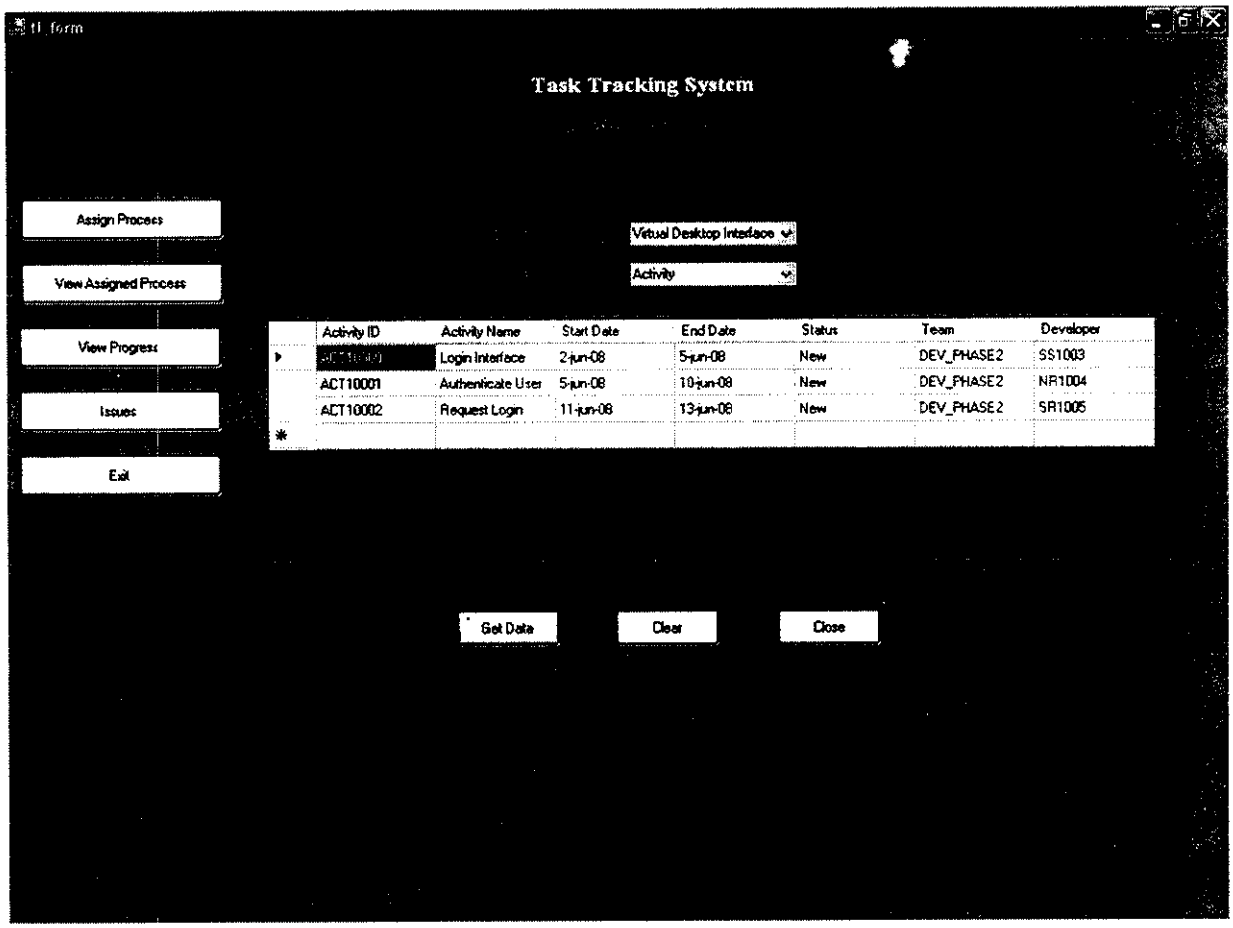
## Track Issue


Form1
Task Tracking System
Create User
Add Project
Assign Project
Close Project
Track Progress
Issues
Exit
Virtual Desktop Interface
Activity ID
Activity Name
Start Date
End Date
Status
Team
Developer
Login Interface
2-jun-08
5-jun-08
Pending
DEV_PHASE2
SS1003
ACT10001
Authenticate User
5-jun-08
10-jun-08
Lagging
DEV_PHASE2
NR1004
ACT10002
Request Login
11-jun-08
13-jun-08
Pending
DEV_PHASE2
SR1005
Print
Clear Data
Close

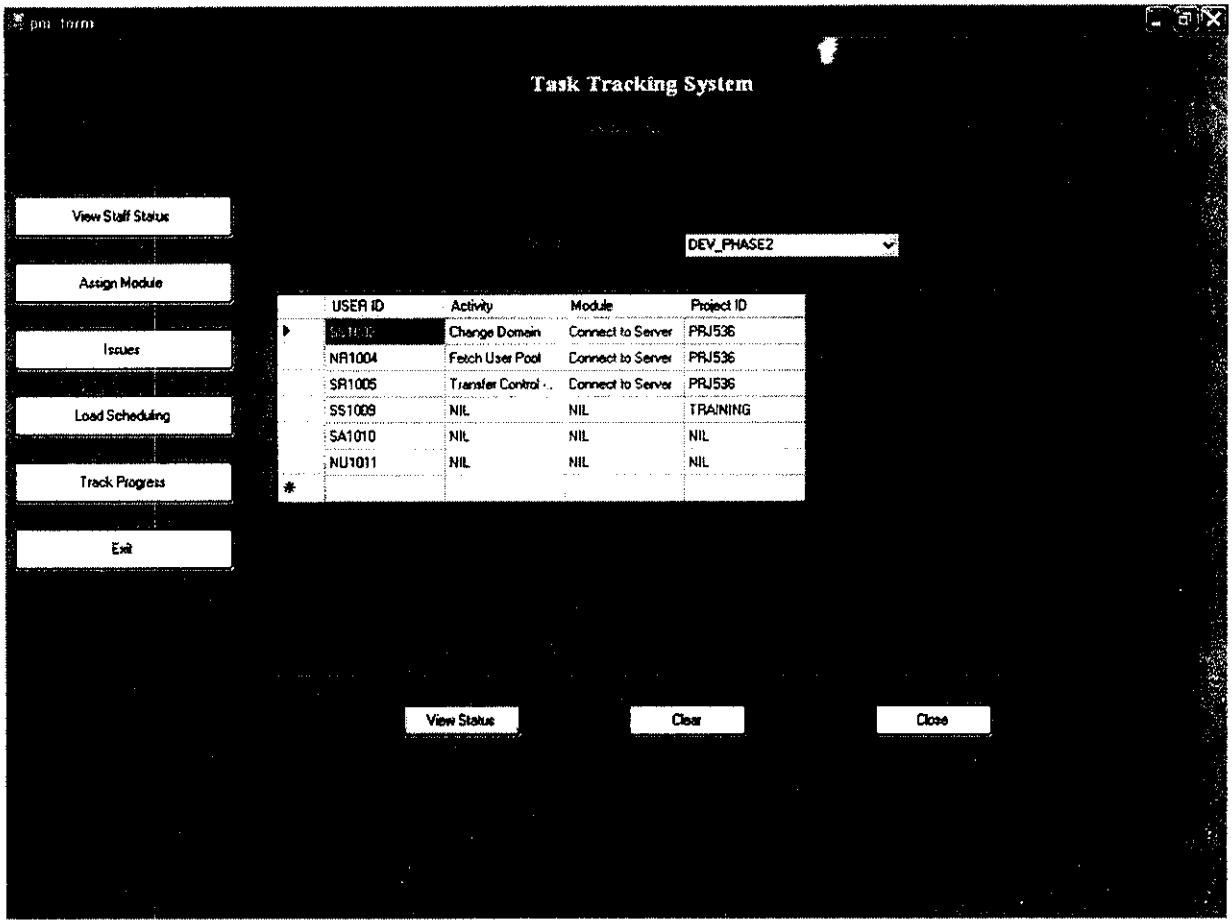
## Update Process Progress – Start a Project



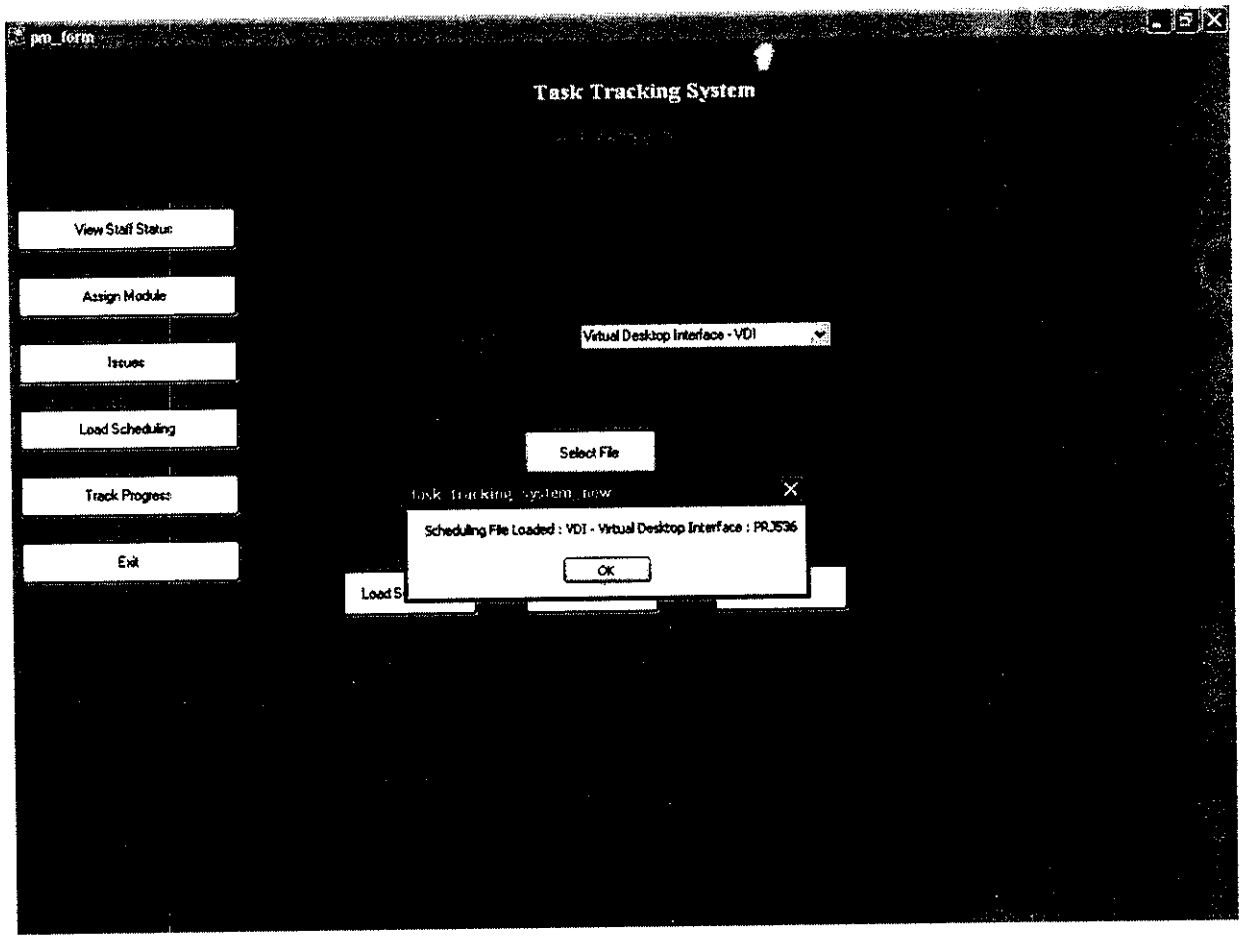
## View Assigned Process



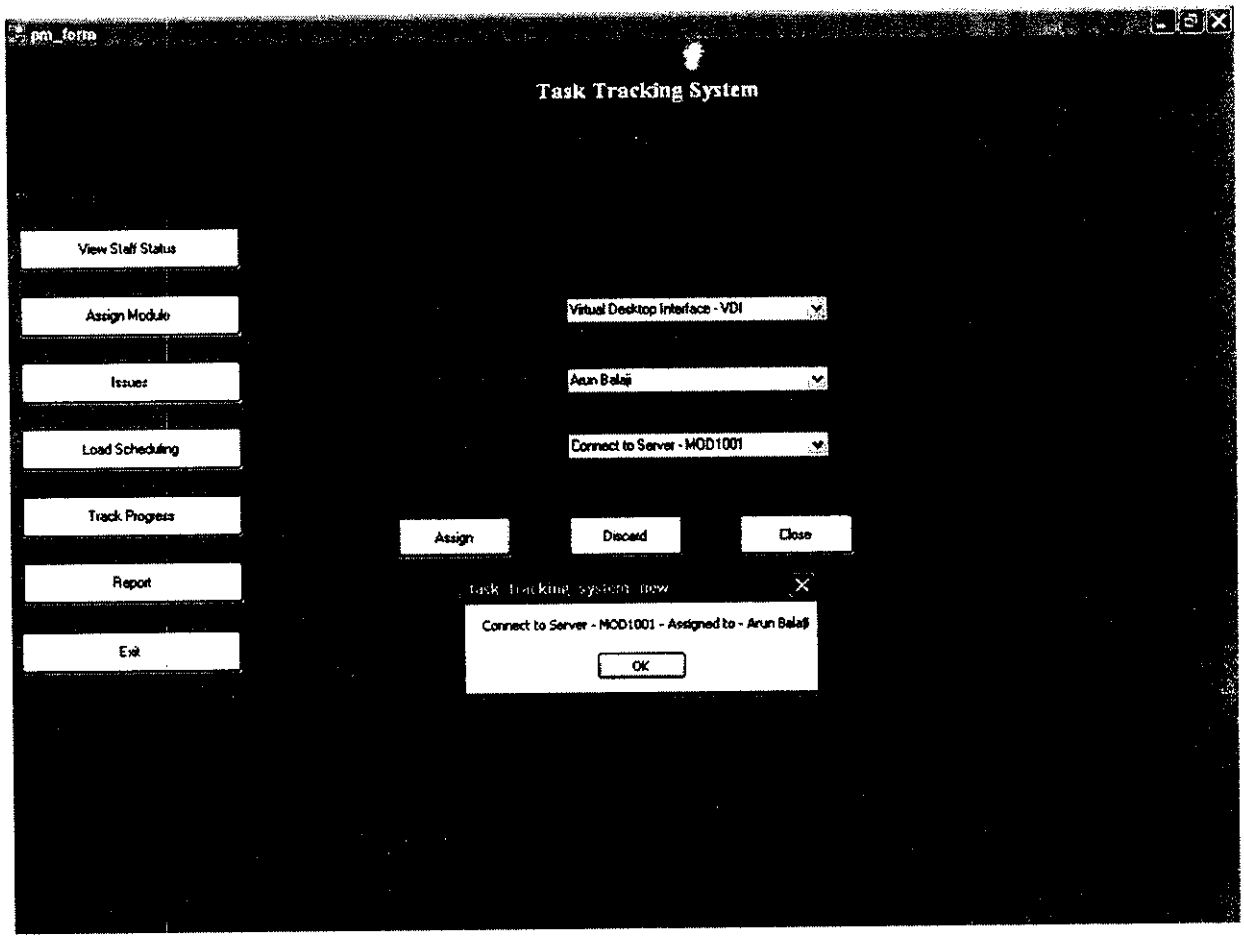
## View Staff Status



## Load Scheduling – Message

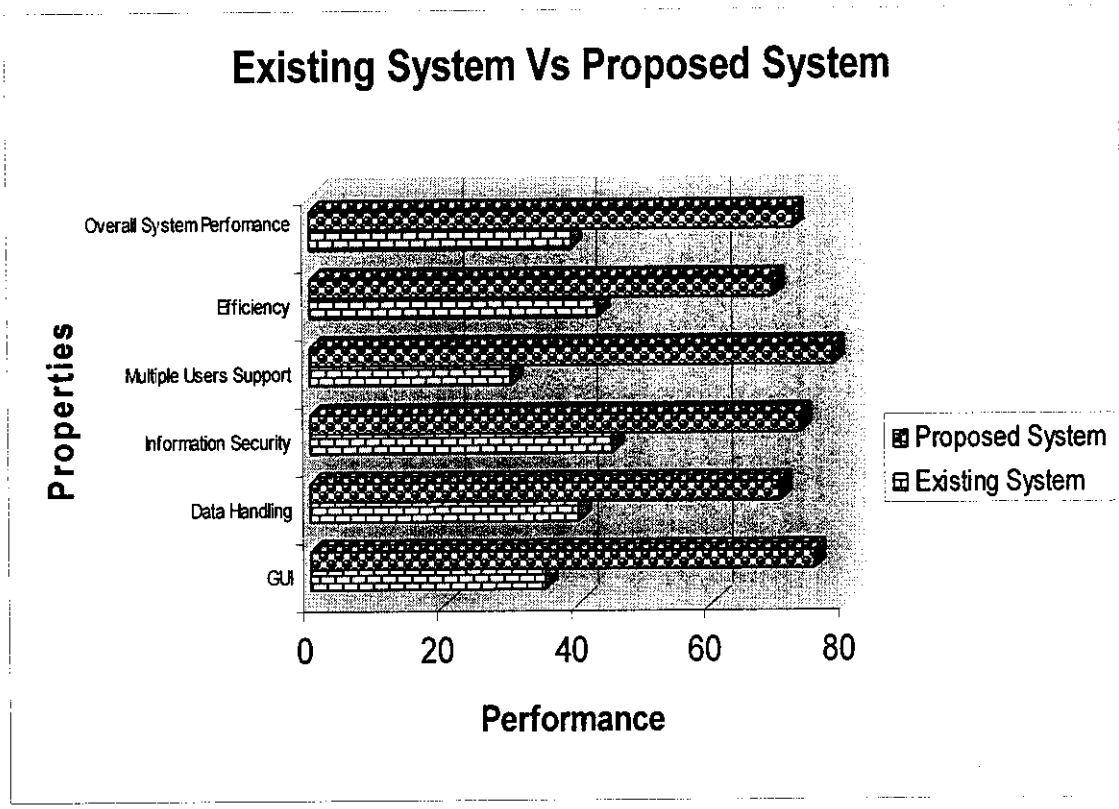


## Assign Module



## 9.2 Graphs

The following graph shows the Comparisons between the Proposed System and the Existing System.



# 10. REFERENCES

Hemlata "**SQL SERVER 2005 REFERENCES**", First Edition, Cyber-Tech Publications.

Herbert Schmidt (2000),"**VIEW ON SQL QUERY ANALYSER**", Four Edition, Tata McGraw-Hill.

Theoron Willis, Bryan Newsome, "Beginning Visual Basic 2005", Wiley Publishing, Inc.

Mathew McDonald, "Visual Basic 2005 .Net Insight for Classic Developers", No Starch Press Publication.

**WESITES**

http://www.wilsonmar.com/1projs.htm

http://vb.net-informations.com/crystal-report/dynamic_crystal_report_from_sql_query_string.htm

http://en.wikipedia.org/wiki/Microsoft_Visual_Studio

http://www.xtremevbtalk.com/forumdisplay.php?f=16

http://functionx.com