ACCOUNTING SYSTEM

A Mini Project Report

A dissertation submitted to JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY in the partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE ENGINEERING

Submitted by

Mr. **M. Siddharth** (**06071A0543**)

Mr. **Sathagopam Srikanth** (**06071A0544**)

Mr. **K. Harish Kumar Reddy** (**06071A0547**)

Under the guidance of

Mr. **G. Ramesh Chandra**

Associate Professor

Department of Computer Science and Engineering

VNR Vignana Jyothi Institute of Engineering and Technology

(Affiliated to J.N.T.University, Hyderabad)

Bachupally(v), Hyderabad, A.P., India.

July, 2009.

VNR Vignana Jyothi Institute of Engineering and Technology

(Affiliated to J.N.T.University, Hyderabad)

Bachupally(v), Hyderabad, A.P., India.

CERTIFICATE

This is to certify that Mr. M.Siddharth (06071A0543), Mr. Sathagopam Srikanth (06071A0544) and Mr. K.Harish kumar reddy (06071A0547) have successfully completed his mini project work at VNR Vignana Jyothi institute of Engineering & Technology, Bachupally, Hyderabad, entitled "**Accounting System** "in the partial fulfillment of the requirement of award of the degree of Bachelor of Technology in Computer Science Engineering for the academic year 2009-2010.

The work is carried out under my supervision and has not been submitted to any other University /Institute for the award of the degree/diploma.

Mr. G. Ramesh Chandra	Dr. C. KiranMai
Project guide	Professor& Head
Assistant Professor	C.S.E. Department
C.S.E. Department	VNR VJIET.
VNR VJIET.	

DECLARATION

We hereby declare that the mini project entitled **“ACCOUNTING SYSTEM”** submitted in partial fulfillment of the requirements for award of the degree of Bachelor of Technology in Computer Science and Engineering at **VNR Vignana Jyothi Institute of Engineering and Technology,** affiliated to Jawaharlal Nehru Technological University, Hyderabad, is an authentic work and has not been submitted to any other University/Institute for award of any degree/diploma.

M.Siddharth	Sathagopam Srikanth	K.Harish Kumar Reddy
III/IV B.Tech	III/IV B.Tech	III/IV B.Tech
C.S.E.	C.S.E.	C.S.E.
06071A0543	06071A0544	06071A0547

ACKNOWLEDGEMENT

Firstly we would like to express our immense gratitude towards our institution ***VNR Vignana Jyothi Institute of Engineering & Technology,*** which created a great platform to attain profound technical skills in the field of Computer Science, thereby fulfilling our most cherished goal.

We are very much thankful to our Principal, ***Dr. C.D.Naidu*** garu and our H.O.D., ***Dr. C. KiranMai*** garu for extending their cooperation in doing this project.

We extend our thanks to our guide, **Mr. G.Ramesh Chandra**, Associate Professor, for his enthusiastic guidance throughout the course of our project.

Last but not the least, our appreciable obligation also goes to all staff members of Computer Science & Engineering Department who directly or indirectly helped us.

Sathagopam Srikanth

M. Siddharth

K. Harish Kumar Reddy

ORGANIZATION’S PROFILE.

This software has been developed for the institute ***V.N.R. Vignana Jyothi Institute of Engineering & Technology***. This institute is located at Bachupally village, Hyderabad. It is an engineering college having good infrastructural facilities. It runs seven courses in the field of engineering namely C.S.E., E.C.E., I.T., E.I.E., C.E., M.E. and E.E.E. It has a staff of about 400 members (both teaching & non-teaching) and intakes students of about 800 students per annum.

ABSTRACT

The project ‘Accounting System’ is developed for the use of Accounts department in Administrative Office of VNR Vignana Jyothi Institute of Engineering and Technology. It is software basically used for handling the financial accounts of the organization.

This software system helps the organization to reduce their work by automation in the process of making accounts and making the accounting process much faster and simpler.

The software is developed according to the Software Development Life Cycle (SDLC) process. It started with inception phase, and then elaboration, construction and finally transition phase.

The project is carried out in sub-phases i.e. Analysis phase, then Design, Coding, Testing phases and finally the Maintenance phase, for each phase, by dividing it into modules and later by integrating these modules. It majorly consists of three modules namely Manager or administrator, Accountant and Clerk.

The project is developed as per the requirements of the customer and tries to reduce the problems being faced by the organization in accounting.

The software is developed with HTML as front end, Oracle database and Java as back end. The detailed description of the project is given in this report.

INDEX

LIST OF FIGURES

LIST OF TABLES

Chapter 1. Introduction

1.1 Introduction

The software '***Accounting system'*** is developed for the use in the administrative office of the institute VNR Vignana Jyothi institute of Engineering & Technology.

This project is used to reduce the burden of financial accounting processing for the institute. Presently the organization is using Microsoft office software to manage the financial accounts which lead to laborious work.

The software helps them to perform accounting operations and manage their accounts easily and fastly.

The major users of the system are

- Manager or Administrative Officer.
- Accountant.
- Clerk.

This software helps to automate the process of accounting.

It uses HTML as front end and java, Oracle DB as backend.

1.2 Existing System

The institute has been using MS-Office (Excel) for their annual and monthly accounting purpose which gave them a laborious and tedious work due to re-entry of many values.

- No Automatic generation of reports .The user has to cross over all the sheets to get the account details and then prepare the consolidated report.

- Errors cannot be easily tracked & corrected. As it is a manual work, errors are easily prone and tracing that error value among the sheets is a very tedious task.
- Manual work is more. Each employee details are to be entered correctly and are to be copied among various sheets that is financial reports and details.
- There is no efficient system to handle the details of the employees.
- There is no efficient system to handle the leave details of the employee. The leave calculation for all the employees has to be done manually and has to be transmitted to the account officer for the calculation of the loss of pay.
- Paper work is also more.

1.3 Proposed System

The Proposed System has the following functionalities.

- It helps in **automating** the Accounting process by reducing the manual work to handle data.
- It helps the organization in storing the employee details of the organization, their leave details and their financial salary details.
- It makes the accounting process simpler and easier by reducing the paper work and automatically handling values in the database.
- It helps the accountant in Automatic generation of financial reports depending on date and also different kinds of reports can be generated.
- It helps to scaling of different features in the financial terms like City Compensatory Allowance Scale, Profession Tax Scale and other percentage scales.
- It provides restricted access (security) with help of passwords to access the data of the organization.

The project is made as per the requirements of the organization. They are collected from the Administrative Officer and the Accountant.

The project is also scalable. We can include other financial accounting features if required.

1.4 Comparison with other existing systems

The Proposed system has the following features when compared with the existing systems.

- It is as per the requirements of the customer.
- It is user friendly and requires no complex training as compared to other packages like tally, fox pro and pay genie.

  As per software engineer's point of view it reduces the **training costs**.
- It also handles employee details and leave details of the organization.
- It is easier to handle transactions and is also error traceable.
- The software is also scalable by adding other accounting features when ever required by the customer or user.

Chapter 2. Literature Survey

2.1 Java

Java is a programming language originally developed by James Gosling at Sun Microsystems and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code (class file) that can run on any Java virtual machine (JVM) regardless of computer architecture.

2.2 Java Server Pages

Java Server Pages or JSP for short is Sun's solution for developing dynamic web sites. JSP provide excellent server side scripting support for creating database driven web applications. It is a Java technology that allows software developers to create dynamically generated web pages, with HTML, XML, or other document types, in response to a Web client request. The technology allows Java code and certain pre-defined actions to be embedded into static content. The JSP syntax adds additional XML-like tags, called JSP actions, to be used to invoke built-in functionality. Additionally, the technology allows for the creation of JSP tag libraries that act as extensions to the standard HTML or XML tags. Tag libraries provide a platform independent way of extending the capabilities of a Web server.

2.2.1 JSP-Tags:

JSP tags can be divided into 4 different types. These are:

- **Directives**

In the directives we can import packages, define error handling pages or the session information of the JSP page.

- **Declarations**

This tag is used for defining the functions and variables to be used in the JSP.

- **Scriptlets**

In this tag we can insert any amount of valid java code and these codes are placed in JSP Service method by the JSP engine.

- **Expressions**

We can use this tag to output any data on the generated page. These data are automatically converted to string and printed on the output stream.

2.3 HTML

HTML, which stands for Hypertext Mark-up Language, is the predominant markup language for web pages. It is essentially a scripting language that marks up text to indicate to a web browser how the text is to be displayed, as well as to enable links, and other features.

2.3.1 HTML tags

A web page is created using a language called, Hypertext Markup Language, better known as HTML Code. You can write your own coding within a plain text editor, such as Note Pad, or use an HTML editor, which will write the code for you.

HTML codes, also referred to as HTML tags, are enclosed by the lesser than (<) and greater than (>) brackets and may be written in capital or lower case letters.

The opening bracket is followed by an element, which is a browser command, and ends with the closing bracket.

An element may also be followed by attributes, which are words describing the properties of the element, and further instruct the browser.

Attributes are only contained in the opening HTML tags to the right of the element and are separated by a space and followed by an equal (=) sign.The value follows the equal sign and is enclosed in quotes.

For example:

<font size=2>

Basic HTML Document Code Structure

Example:

<html>

<head>

<title>Your Page Title</title>

</head>

<body>

This area will contain everything that will be visible through a web browser, such as text and graphics. All of the information will be HTML coded.

</body>

</html>

<html> - Begins your HTML document.

<head> - Contains information about the page such as the TITLE, META tags for proper Search Engine indexing, STYLE tags, which determine the page layout, and JavaScript coding for special effects.

<title> - The TITLE of your page. This will be visible in the title bar of the viewers’ browser.

</title> - Closes the HTML <title> tag.

</head> - Closes the HTML <head> tag.

<body> - This is starting point of your document and placing your HTML codes.

</body> - Closes the HTML <body> tag.

</html> - Closes the <html> tag.

2.4 JavaScript

JavaScript is a scripting language used to enable programmatic access to objects within other applications. It is primarily used in the form of client-side JavaScript for the development of dynamic websites. JavaScript is a dialect of the ECMAScript standard and is characterized as a dynamic, loosely typed, prototype-based language with first-class functions. JavaScript was influenced by many languages and was designed to look like Java, but to be easier for non-programmers to work with.

2.4.1 Features

The following features are common to all conforming ECMA Script implementations, unless explicitly specified otherwise.

1. Imperative and structured

2. Dynamic
3. run-time evaluation
4. first-class functions
5. Inner functions
6. Prototype-based
7. functions as object constructors
8. functions as methods

2.5 Oracle Database

The Oracle Database (commonly referred to as Oracle RDBMS or simply Oracle) consists of a relational database management system (RDBMS) produced and marketed by Oracle Corporation. As of 2009, Oracle remains a major presence in database computing.

2.6 JDBC Connection:

JDBC is an API for the Java programming language that defines how a client may access a database. It provides methods for querying and updating data in a database. JDBC is oriented towards relational databases.

JDBC allows multiple implementations to exist and be used by the same application. The API provides a mechanism for dynamically loading the correct Java packages and registering them with the JDBC Driver Manager. The Driver Manager is used as a connection factory for creating JDBC connections.

JDBC connections support creating and executing statements. These may be update statements such as SQL's CREATE, INSERT, UPDATE and DELETE, or they may be query statements such as SELECT. Additionally, stored procedures may be invoked through a JDBC connection. JDBC represents statements using one of the following classes:

1 .Statement – the statement is sent to the database server each and every time.

2. Prepared Statement – the statement is cached and then the execution path is pre determined on the database server allowing it to be executed multiple times in an efficient manner.

3. Callable Statement – used for executing stored procedures on the database.

2.7 Servers

2.7.1 Apache Tomcat Server

Apache Tomcat is a servlet container developed by the Apache Software Foundation (ASF). Tomcat implements the Java Servlet and the Java Server Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run.

2.7.2 GlassFish Server

GlassFish is an open source application server project led by Sun Microsystems for the Java EE platform. The commercial version is called Sun GlassFish Enterprise Server. GlassFish is free software, dual-licensed under two free software licences: the Common Development and Distribution License (CDDL) and the GNU General Public License (GPL) with the class path exception.

GlassFish is based on source code donated by Sun and Oracle Corporation's TopLink persistence system. It uses a derivative of Apache Tomcat as the servlet container for serving Web content, with an added component called Grizzly which uses Java NIO for scalability and speed.

2.8 Integrated Development Environment:

2.8.1 Eclipse

Eclipse refers to both a platform for the development of applications for the network (using Java, JavaScript, PHP, Python, Ruby, Groovy, C, and C++), and an Integrated Development Environment (IDE) developed using the Eclipse Platform.

The Eclipse Platform allows applications to be developed from a set of modular software components called modules. A module is a Java archive file that contains Java classes written to interact with the Eclipse Open APIs and a manifest file that identifies it as a module. Applications built on modules can be extended by adding new modules. Since modules can be developed independently, applications based on the Eclipse platform can be extended by third party developers.

Chapter 3. System Analysis

3.1 System Requirements

These are the resources required for the effective functioning of the software.

3.1.1 Hardware requirements

They correspond to the hardware required for the efficient functioning of the software. The client and server side requirements are as follows.

3.1.1.1 Client side

Processor	Minimum: Intel Pentium-II or AMD-K6 or above 32-bit processor.
Hard-Disk	Minimum:2GB or above
Main-Memory	Minimum:32MB SD-RAM or above
Network Interface	Minimum:10BASE-T supporting Fast-Ethernet Card, as per the specifications of IEEE-802.3

3.1.1.2 Server side

Processor	Minimum: Intel Pentium -IV or AMD athlon or greater
Hard-Disk	Minimum: 40GB or above with provision for expansion or greater
Main-Memory	Minimum : 512MB DDR OR SD RAM
Network Interface	Minimum:10BASE-T supporting Fast-Ethernet Card, as per the specifications of IEEE-802.3

3.1.2 Software requirements

These correspond to the additional software support required for the functioning of the software.

These are classified into two types i.e. Client and Server side.

3.1.2.1 Client side

Operating System	Microsoft Windows 98 or later versions, Macintosh OS 8.0 or newer versions, GNU Linux 2.6 or above
Browser	HTML, JavaScript enabled browser.

3.1.2.2 Server side

Front End	Apache Tomcat 6.0 or above
Operating System	Microsoft Windows Xp or above/ GNU-Linux 2.6 or above/ Mac OS-X or above
Back end	Java ,Oracle 8 or greater

3.2 Modules

The system is divided into three major modules.

1. Manager Module (Administrator)
2. Accounting Module
3. Clerical Module

3.2.1 Manager Module (Administrator)

This module provides functionalities to the Manager or Administrator of the accounting system to maintain the Employee details and the details of users who access the system and their permanent salary details.

The functionalities of this module are

- Adding or removing Employees.
- Adding or removing Users.
- Change the access permissions of Users.
- Changing the personal or permanent salary details of Employees.
- Modifying the various scales involved in salary calculations.
- Handling accounting and leave details.
- Changing passwords.
- Updating leave limits.

3.2.2 Accounting Module

This module provides functionality to the Accountant to manage the monthly salary details of Employees, generate Pay slips, generate various reports etc. based on the Employee salaries.

The functionalities of this module are

- Insert salary details of Employees.
- Generate Pay slips.
- Generate Reports.(annual ,quarterly).
- Generate Bank, Tax Statements.
- Change password.
- View Employee’s personal and permanent salary details.

3.2.3 Clerical Module

This module provides functionality to the clerk to manage the leave details of individual Employees.

The functionalities of this module are:

- Insert Leave details.
- Modify Leave details.
- Remove Leave details.
- Preparation of leave reports.(for loss of pay calculation also)
- View Employee details.
- Checking limits & permissions for leaves.

3.3 Types of Users

The system consists of three types of users

1. Administrator
2. Accountant
3. Clerk

3.3.1 Administrator

The Administrator has the privileges to access all the functionalities of the system. He is responsible for the creation of other users of the system as well as creation of the new Employees accounts. Apart from this he is capable to adjust various scales (like Tax scales, CCA scales, leave limits etc.) which are used in salary calculation.

3.3.2 Accountant

The Accountant is the person responsible for managing the monthly salary details of the Employees. The Accountant can enter the salary details modify them and then generate pay slips for employees. He can also generate various reports and Statements regarding the employee Salaries.

3.3.3 Clerk

The Clerk is responsible for maintaining the Leave database of employees. He is restricted to leave database only he cannot access other parts of the system. He is responsible for entering number of leaves and the type of the leave (like casual leaves, special casual leave, half pay leave, loss of pay etc.) by the employee each month.

Chapter 4. System Design

4.1 System Design:

The software application consists of a client and server which can reside on the same system or on separate systems. The server end software would consist of central database (implemented in Oracle DB) as the backend, which would store all the accounting information. The front end and the client side software would be a GUI application that would allow the administrator or the other users to access the Database as per their access permissions.

4.2 UML Overview

Unified Modeling Language (UML) is a standardized general-purpose modeling language in the field of software engineering.

UML includes a set of graphical notation techniques to create abstract models of specific systems. The Unified Modeling Language (UML) is an open method used to specify, visualize, modify, construct and document the artifacts of an object-oriented software intensive system under development. UML offers a standard way to write a system's blueprints, including conceptual components such as:

* Actors,

* Business processes and

* System components and activities

As well as concrete things such as:

* Programming language statements,

* Database schemas, and

* Reusable software components.

UML combines best practices from data modeling concepts such as entity relationship diagrams, business modeling (work flow), object modeling and component modeling. It can be used with all processes, throughout the software development life cycle, and across different implementation technologies. UML has succeeded the concepts of the Booch method, the Object-modeling technique (OMT) and Object-oriented software engineering (OOSE) by fusing them into a single, common and widely usable modeling language. UML aims to be a standard modeling language which can model concurrent and distributed systems. UML is not an industry standard, but is taking shape under the auspices of the Object Management Group (OMG). OMG has initially called for information on object-oriented methodologies that might create a rigorous software modeling language. Many industry leaders have responded in earnest to help create the standard.

UML models may be automatically transformed to other representations (e.g. Java) by means of QVT-like transformation languages, supported by the OMG. UML is extensible, offering the following mechanisms for customization: profiles and stereotype. The semantics of extension by profiles have been improved with the UML 1.0 major revision.

4.2.1 Modeling

It is very important to distinguish between the UML model and the set of diagrams of a system. A diagram is a partial graphical representation of a system's model. The model also contains “semantic backplane documentation” such as written use cases that drive the model elements and diagrams.

UML diagrams represent two different views of a system model:

* Static (structural) view: Emphasizes the static structure of the system using objects, attributes, operations and relationships. The structural view includes class diagrams and composite structure diagrams.

* Dynamic (behavioral) view: Emphasizes the dynamic behavior of the system by showing time sequenced collaborations among objects and changes to the internal states of objects. This view includes sequence diagrams, activity diagrams and state chart diagrams.

UML models can be exchanged among UML tools by using the XMI interchange format.

4.2.2 Diagrams Overview

UML 2.0 has 9 types of diagrams divided into two categories. Five diagram types represent the structure application. Four diagrams represent the Dynamic behavior of the system

UML does not restrict UML element types to a certain diagram type. In general, every UML element may appear on almost all types of diagrams. This flexibility has been partially restricted in UML 2.0.

In keeping with the tradition of engineering drawings, a comment or note explaining usage, constraint, or intent is allowed in a UML diagram.

The following are the different types of UML diagrams.

4.2.2.1 Static Diagrams

- ✓ Class Diagram
- ✓ Object Diagram
- ✓ Component Diagram
- ✓ Deployment Diagram

4.2.2.2 Dynamic Diagrams

- ✓ Use case Diagram
- ✓ Sequence Diagram
- ✓ Collaboration Diagram
- ✓ Activity Diagram
- ✓ State Chart Diagram

4.3 Diagrams

4.3.1 Class Diagram

These diagrams describe the static view of the system by showing the system's classes, their attributes, and the relationships among the classes.

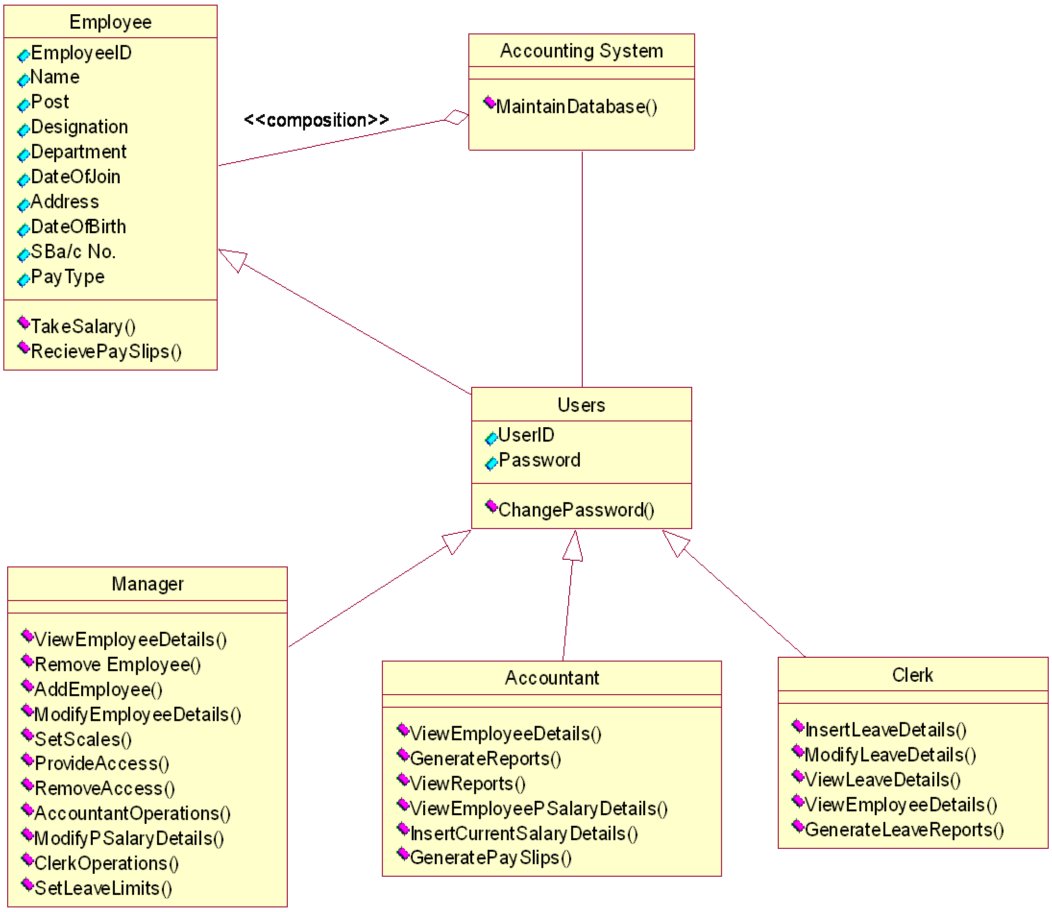


Fig 4.3.1 Class Diagram

4.3.2 Component Diagram

These diagrams depict how a software system is split up into components and shows the dependencies among these components.

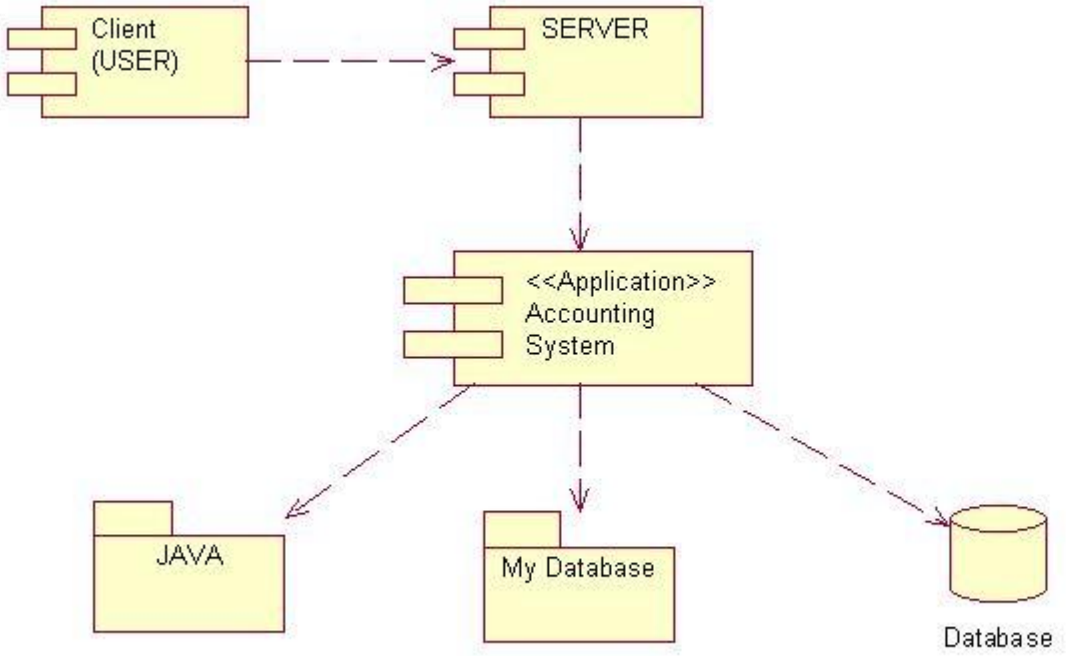


Fig 4.3.2 Component Diagram

4.3.3 Deployment Diagram

These diagrams serve to model the hardware used in system implementations, and the execution environments and artifacts deployed on the hardware

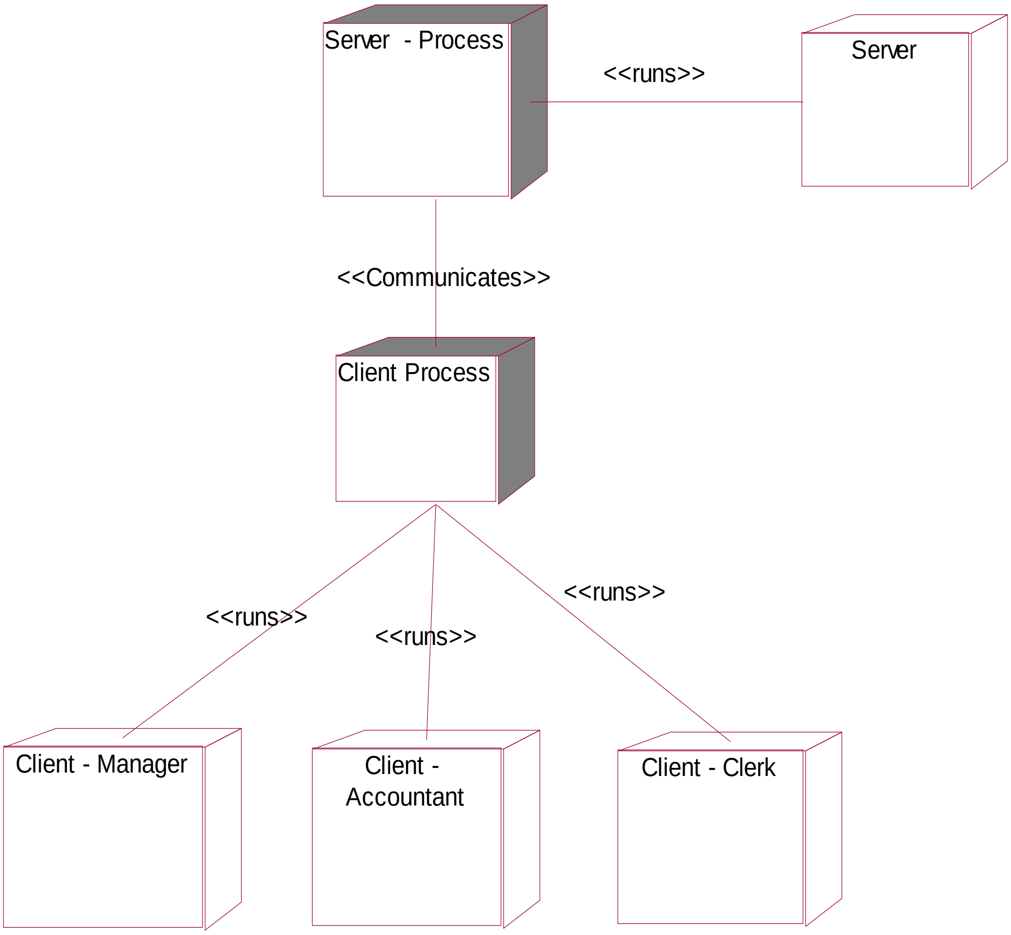


Fig 4.3.3 Deployment Diagram

4.3.4 Use Case Diagrams

It shows the functionality provided by a system in terms of actors, their goals represented as use cases, and any dependencies among those use cases.

4.3.4.1 Administrator (Manager)

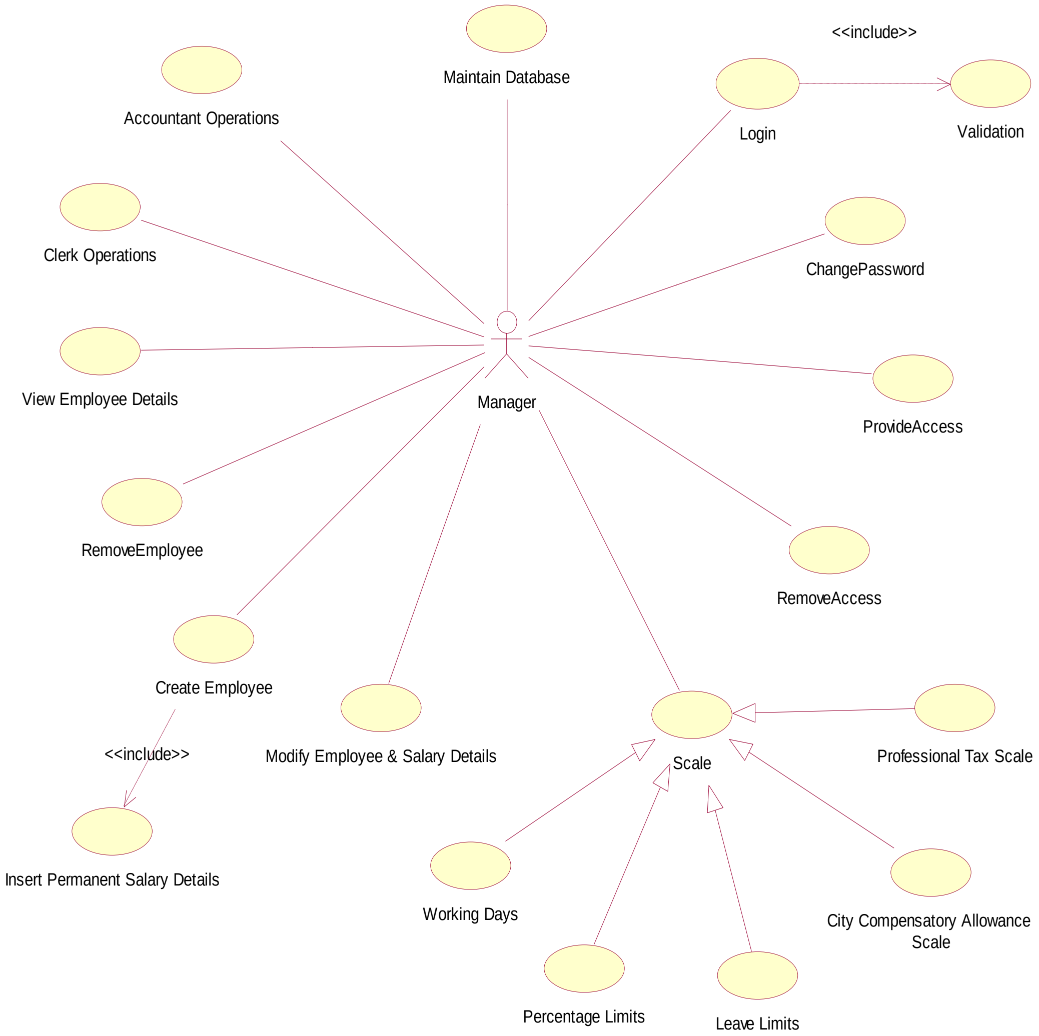


Fig 4.3.4.1 Administrator (Manager)

4.3.4.2 Accountant

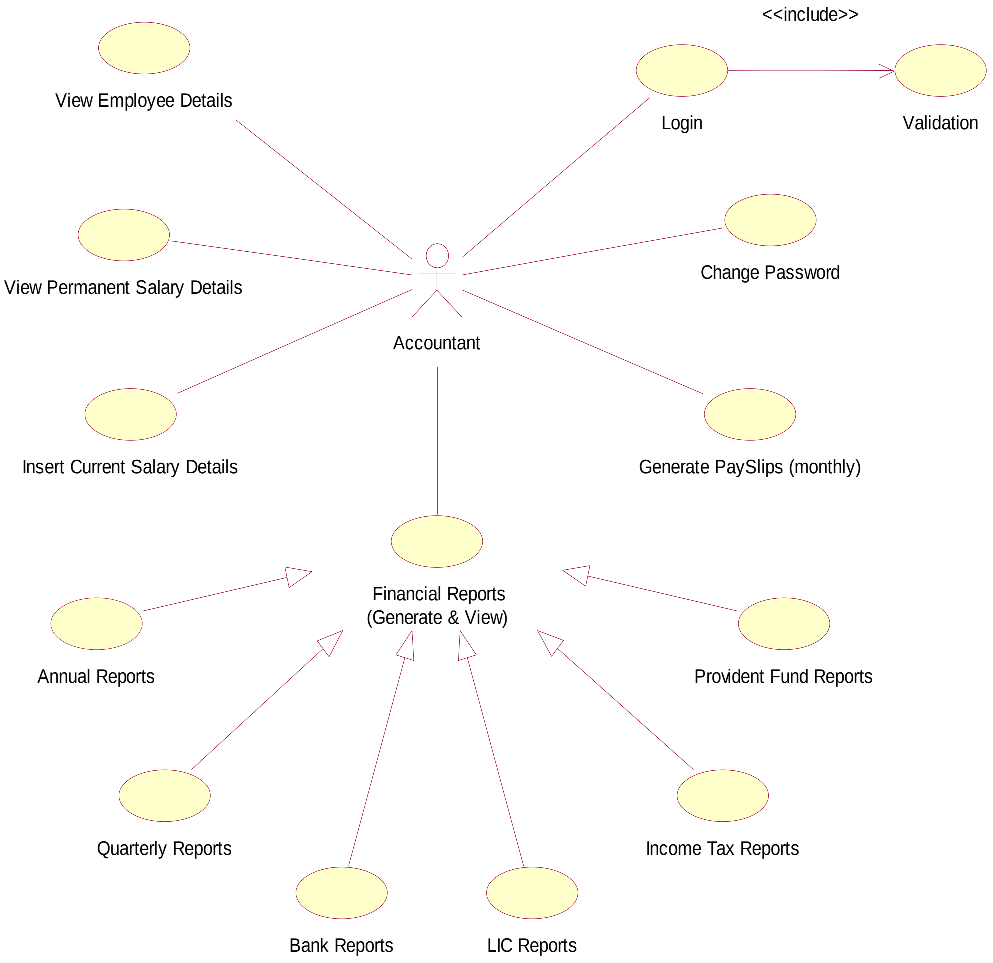


Fig 4.3.4.2 Accountant

4.3.4.3 Clerk

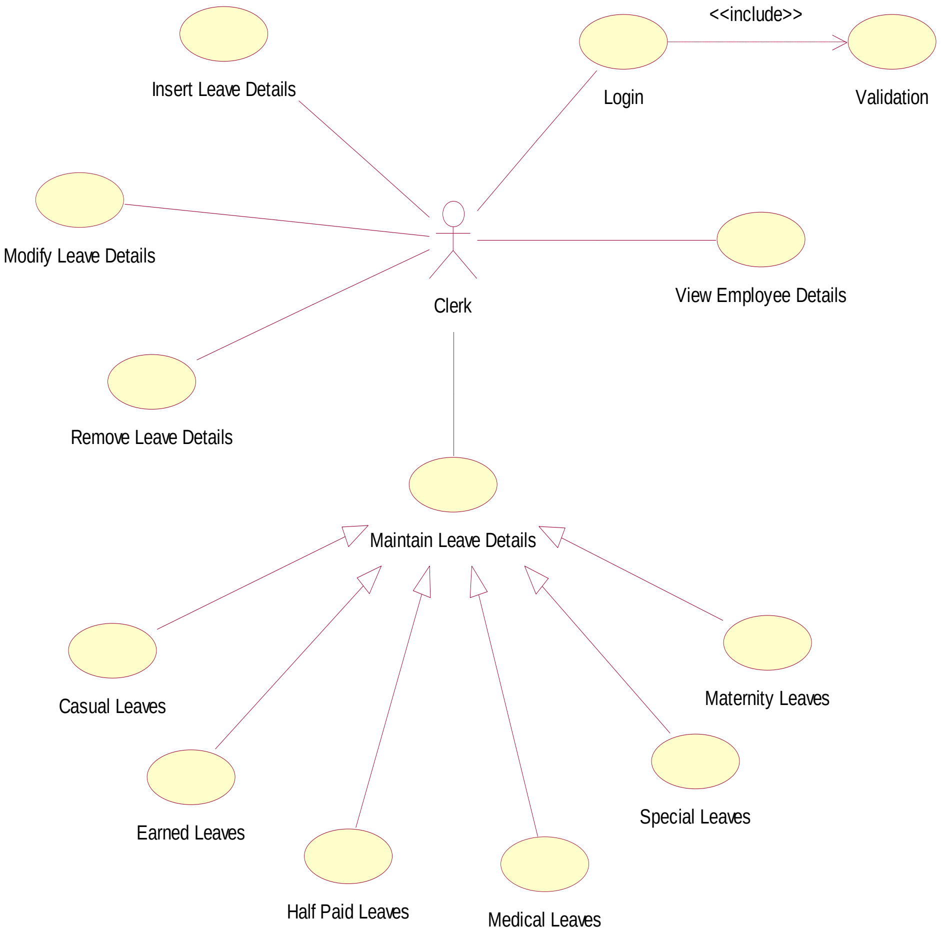


Fig 4.3.4.3 Clerk

4.3.5 Sequence Diagrams

4.3.5.1 Administrator

4.3.5.1.1 Create Employee

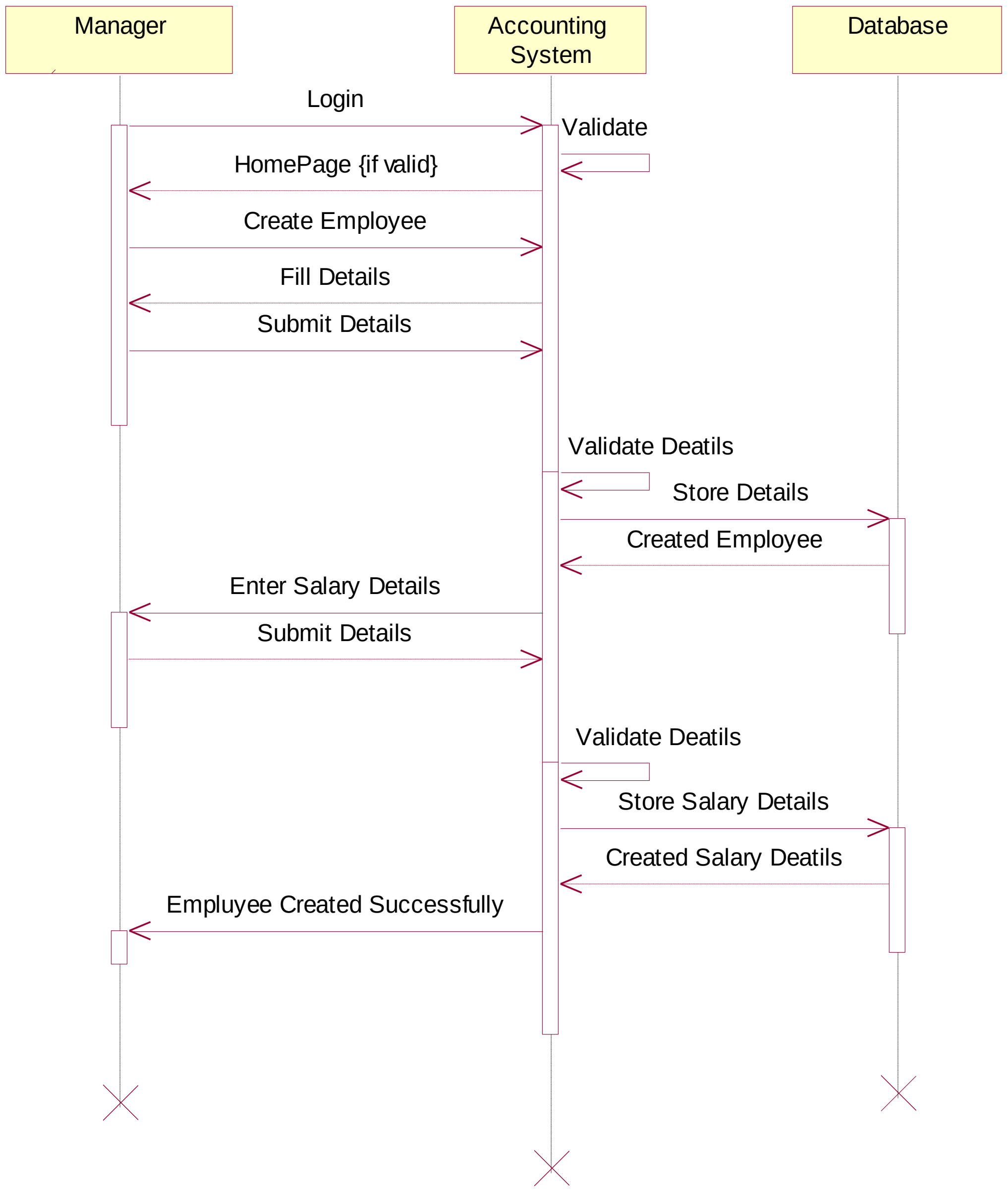


Fig 4.3.5.1.1 Create Employee

4.3.5.1.2 Modify details

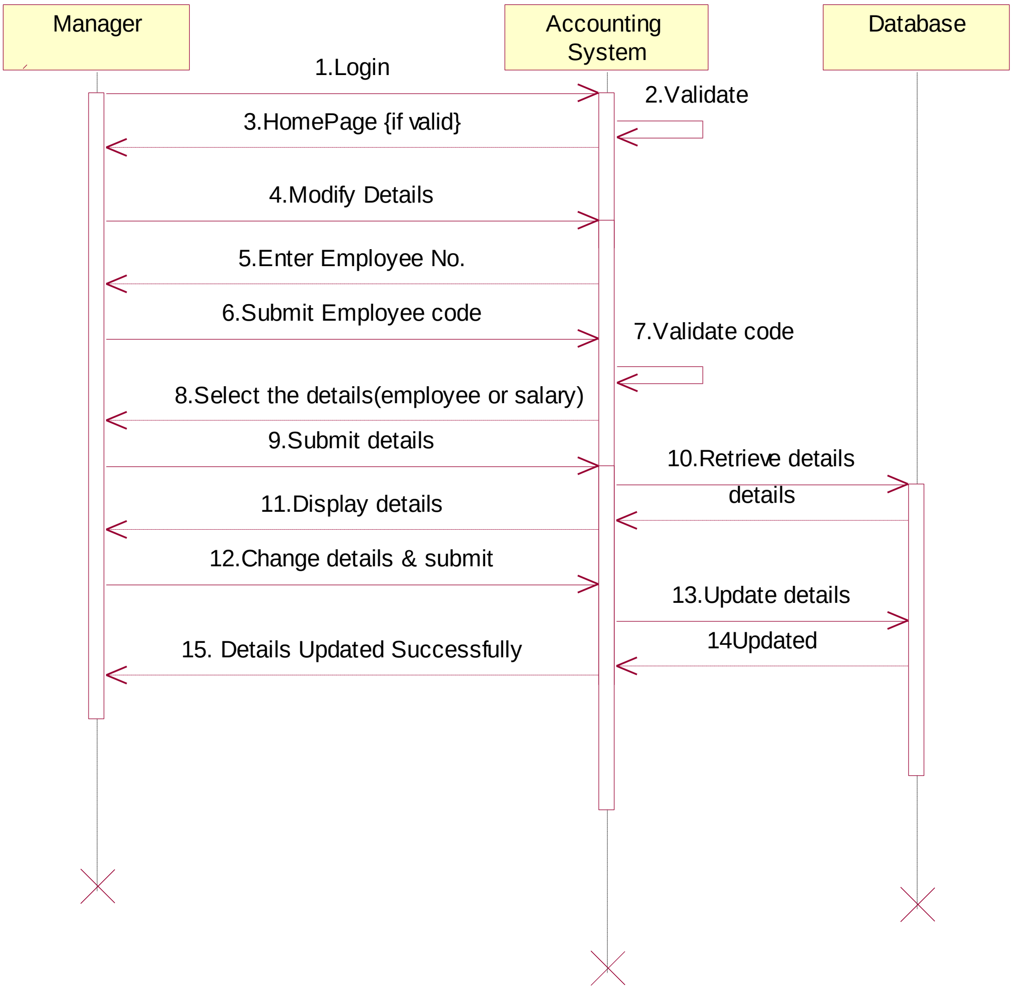


Fig 4.3.5.1.2 Modify details

4.3.5.1.3 Remove employee

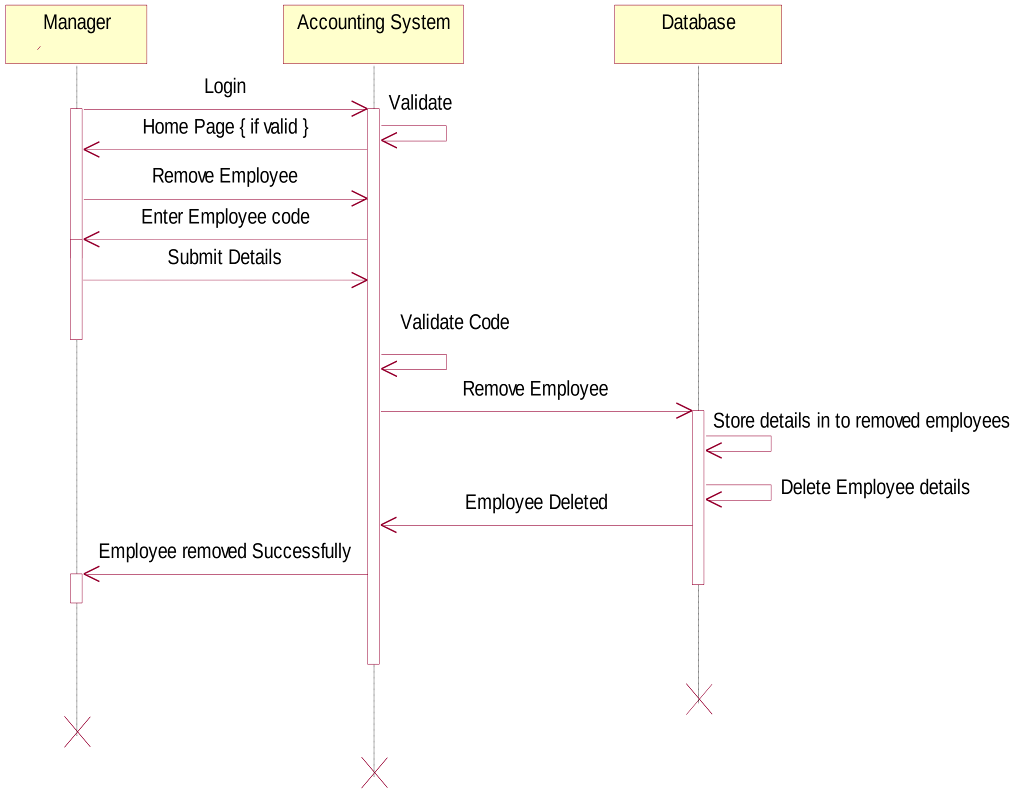


Fig 4.3.5.1.3 Remove employee

4.3.5.1.4 Scales Modification

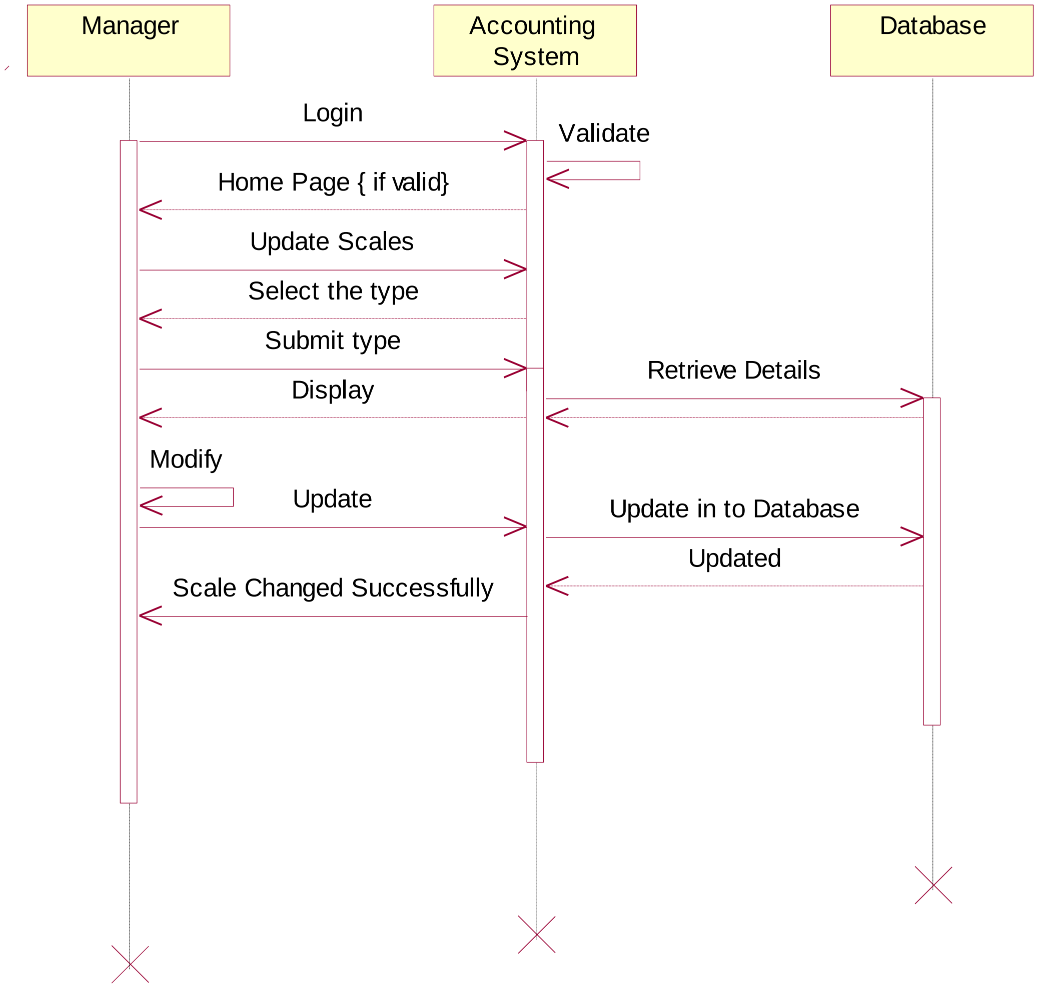


Fig 4.3.5.1.4 Scales Modification

4.3.5.1.5 Access Modifications

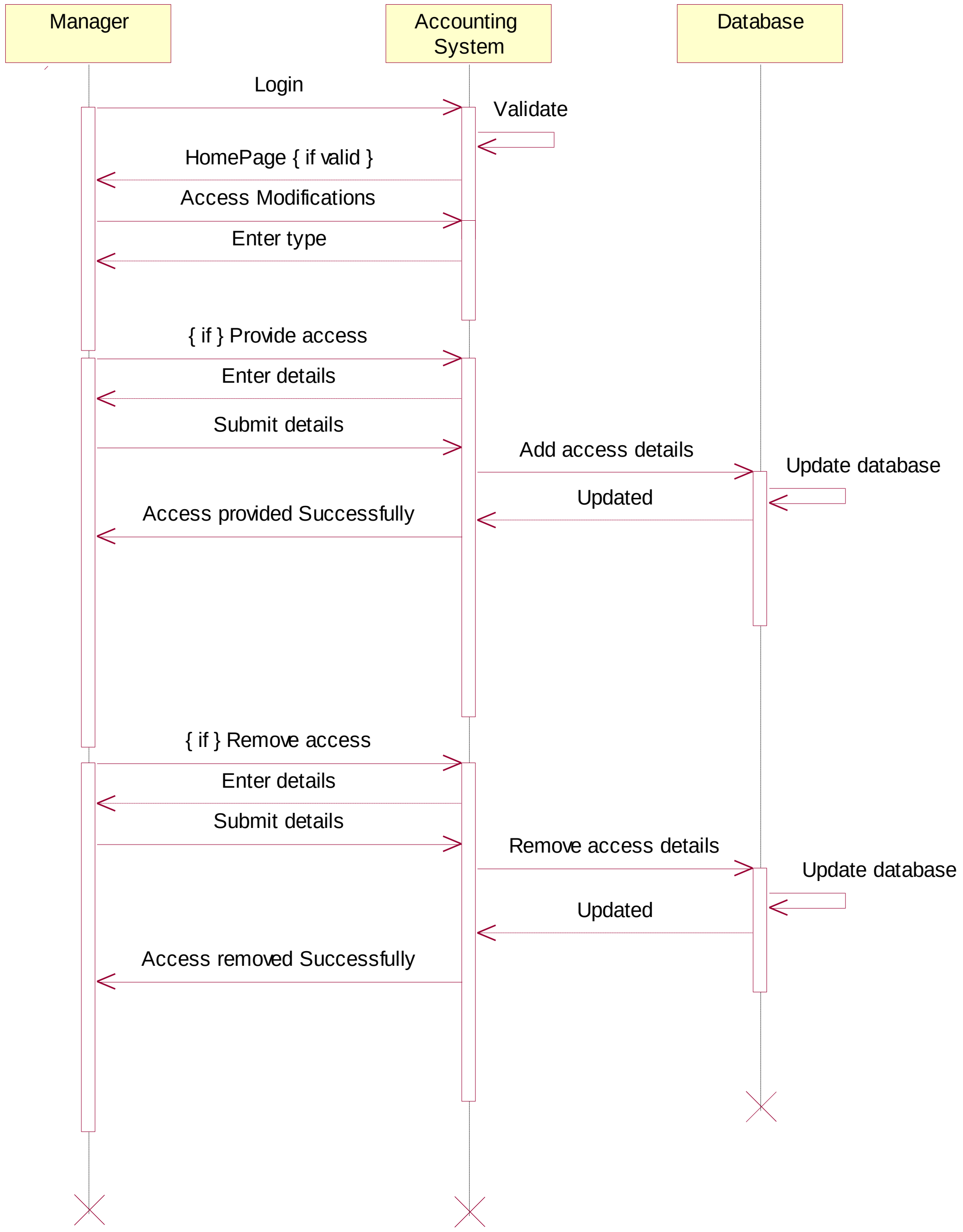


Fig 4.3.5.1.5 Access Modifications

4.3.5.2 Accountant

4.3.5.2.1 Salary Entries

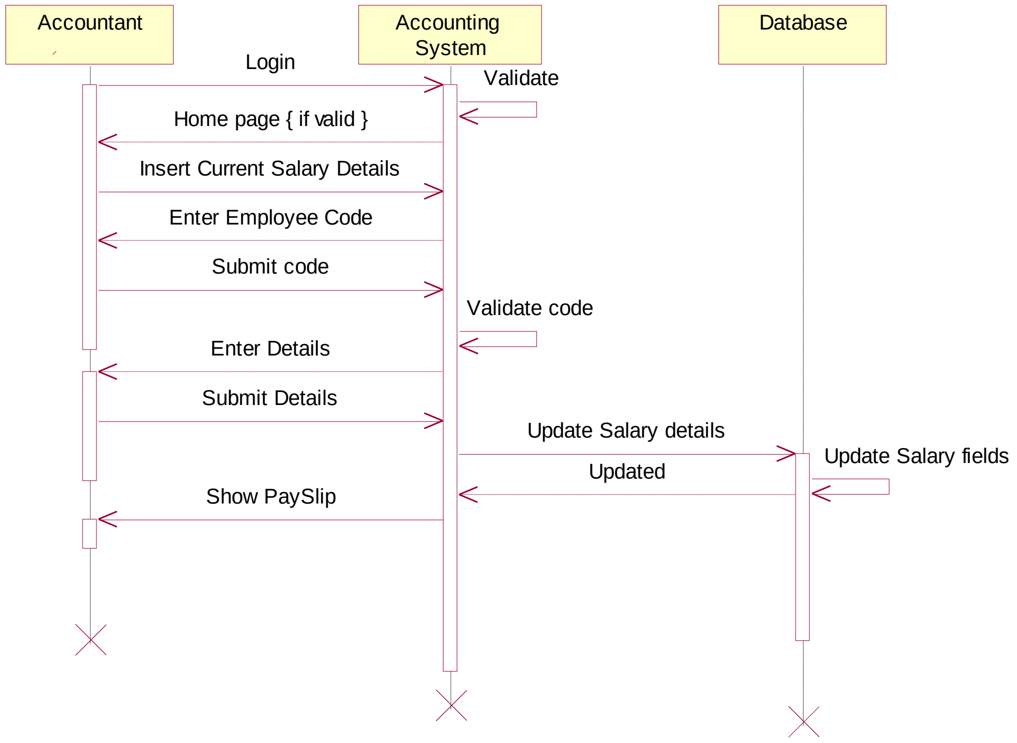


Fig 4.3.5.2.1 Salary Entries

4.3.5.2.2 Generate Pay Slips

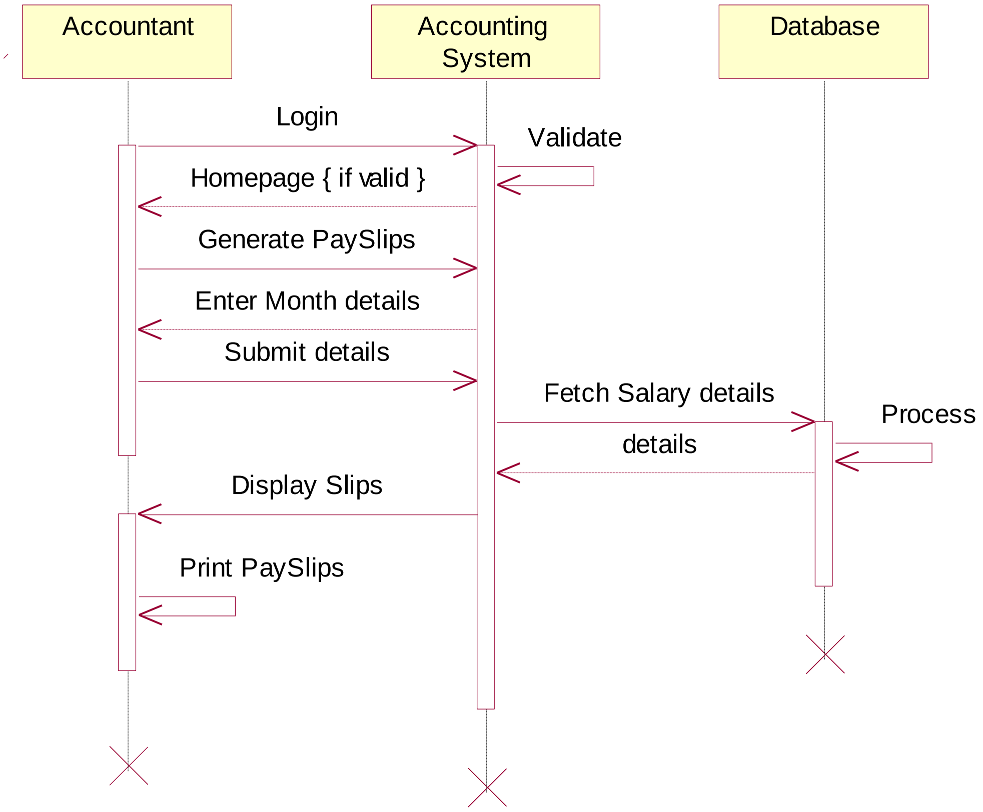


Fig 4.3.5.2.2 Generate Pay Slips

4.3.5.2.3 Report Generation

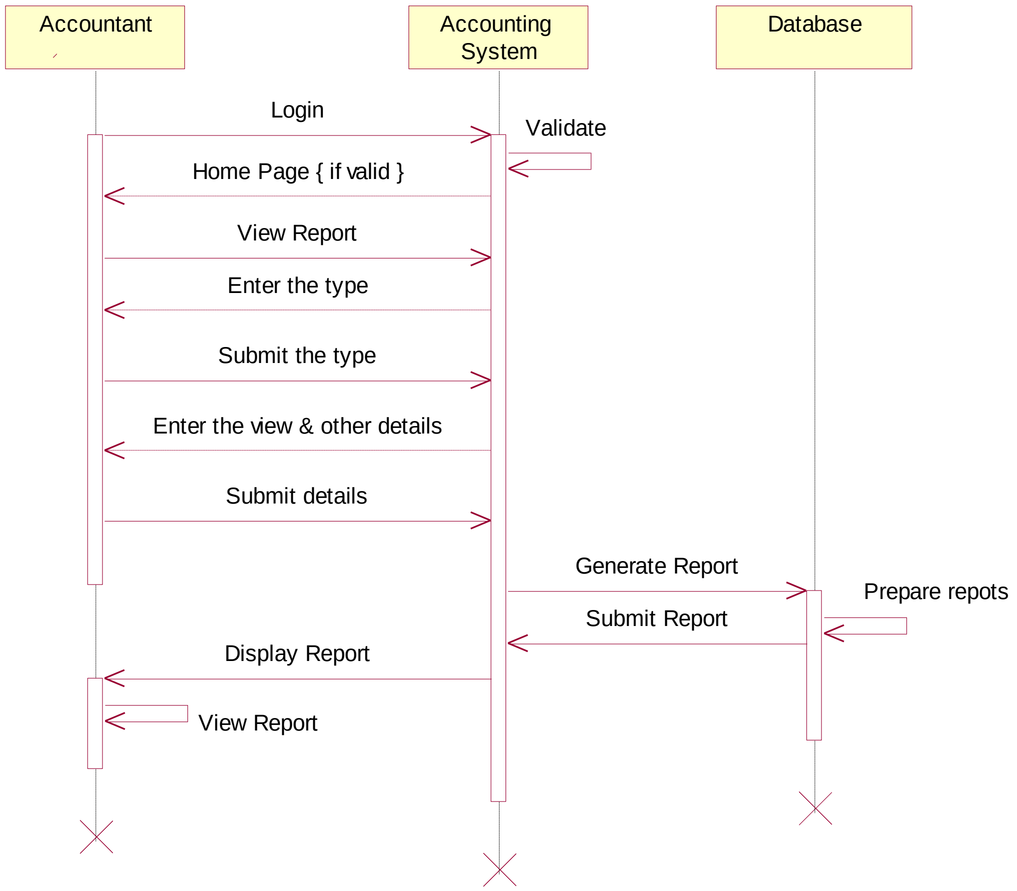


Fig 4.3.5.2.3 Report Generation

4.3.5.3 Clerk

4.3.5.3.1 Modify Leave Details

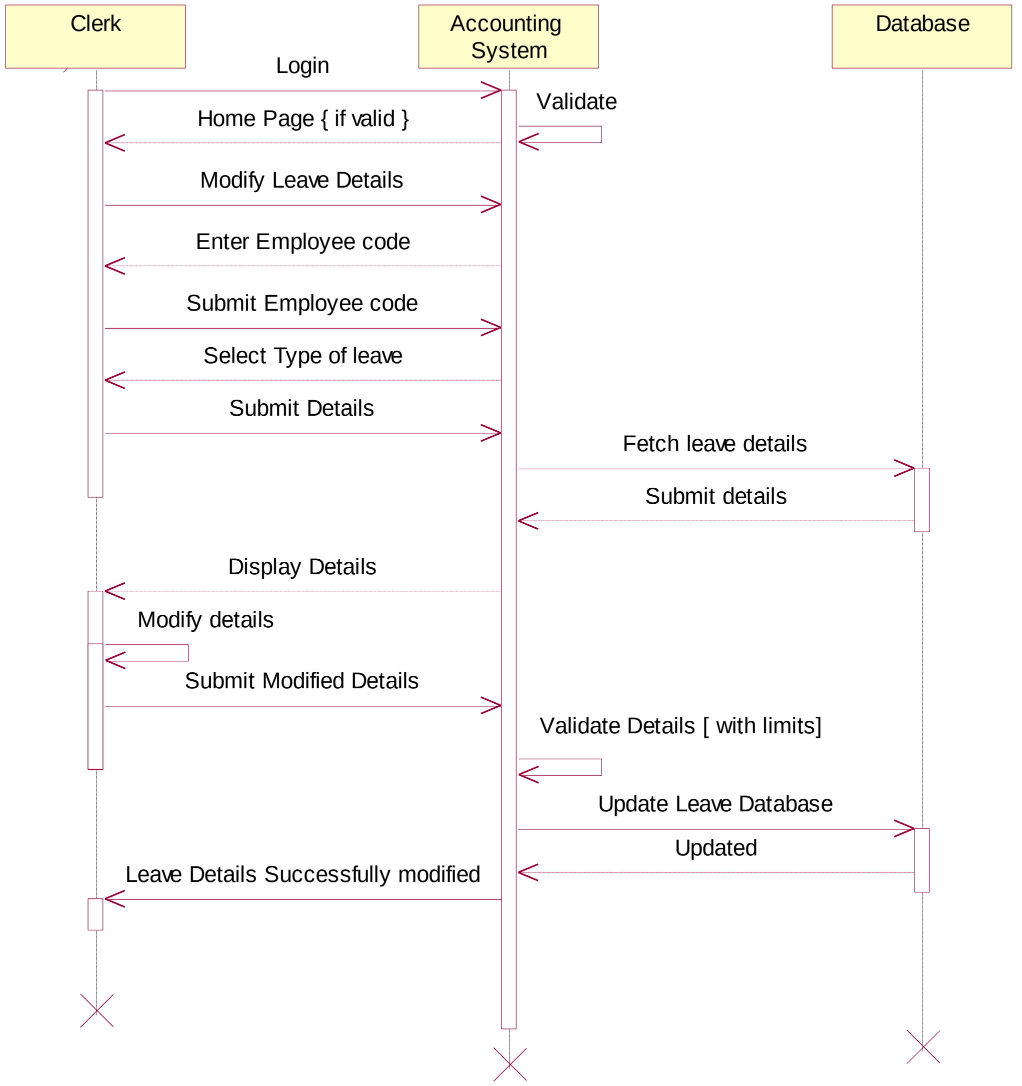


Fig 4.3.5.3.1 Modify Leave Details

4.3.5.4 Change Password (Common for all the users)

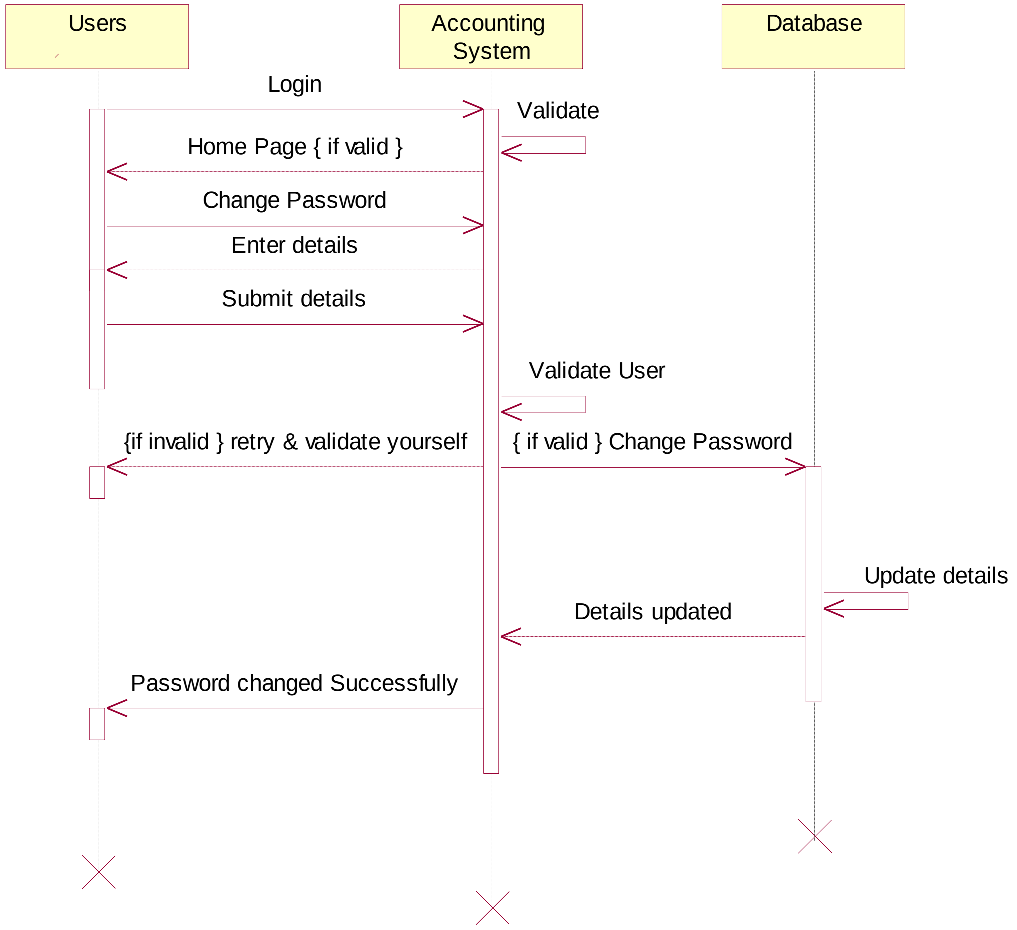


Fig 4.3.5.4 Change Password

4.3.6 Activity Diagram

It represents the business and operational step-by-step workflows of components in a system.

4.3.6.1Login

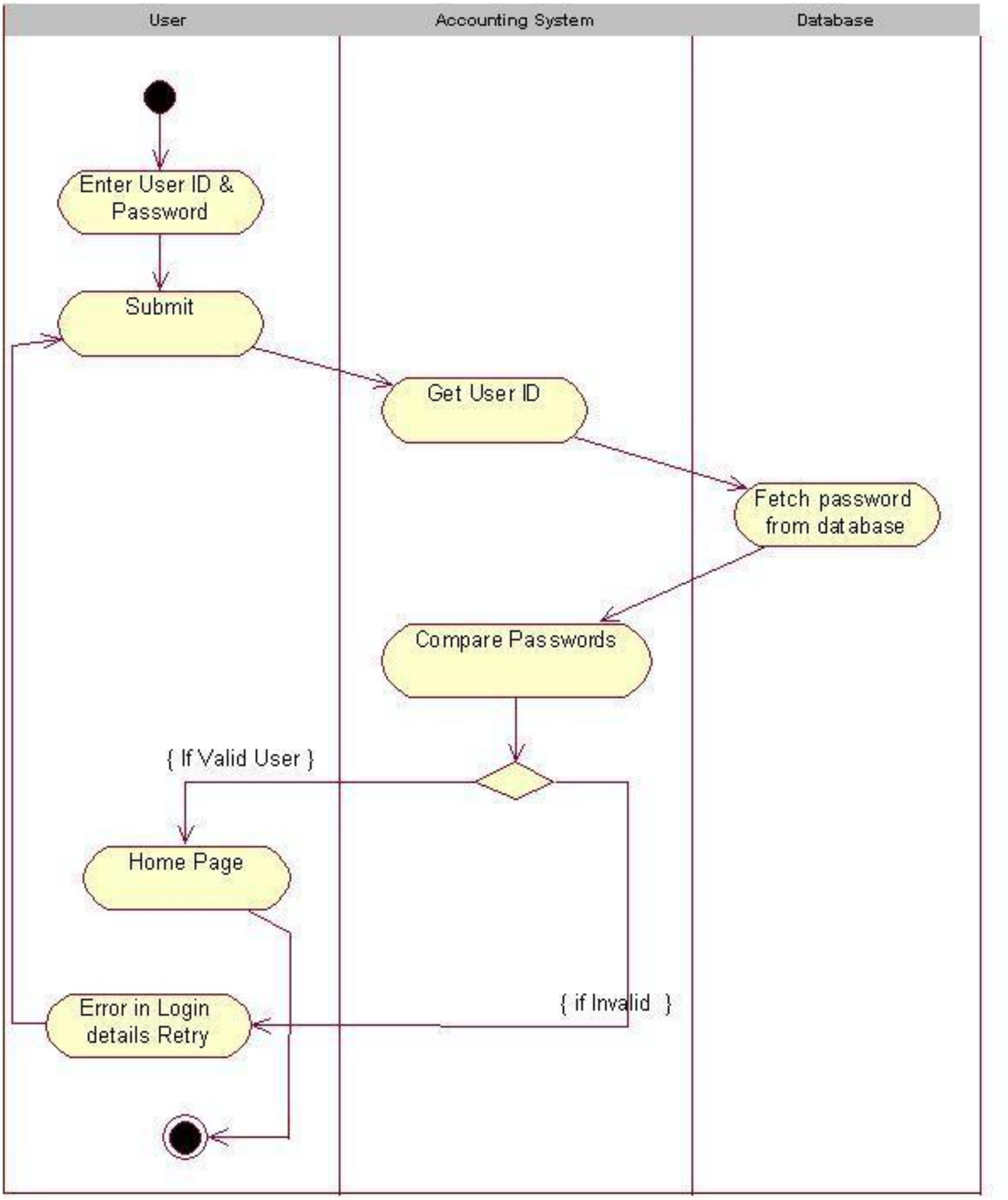


Fig 4.3.6.1 Login

4.4 Database Design

4.4.1 Entity-Relationship Diagram

ER diagram is used to represent schema of the database. It gives the relationship between different tables and attributes.

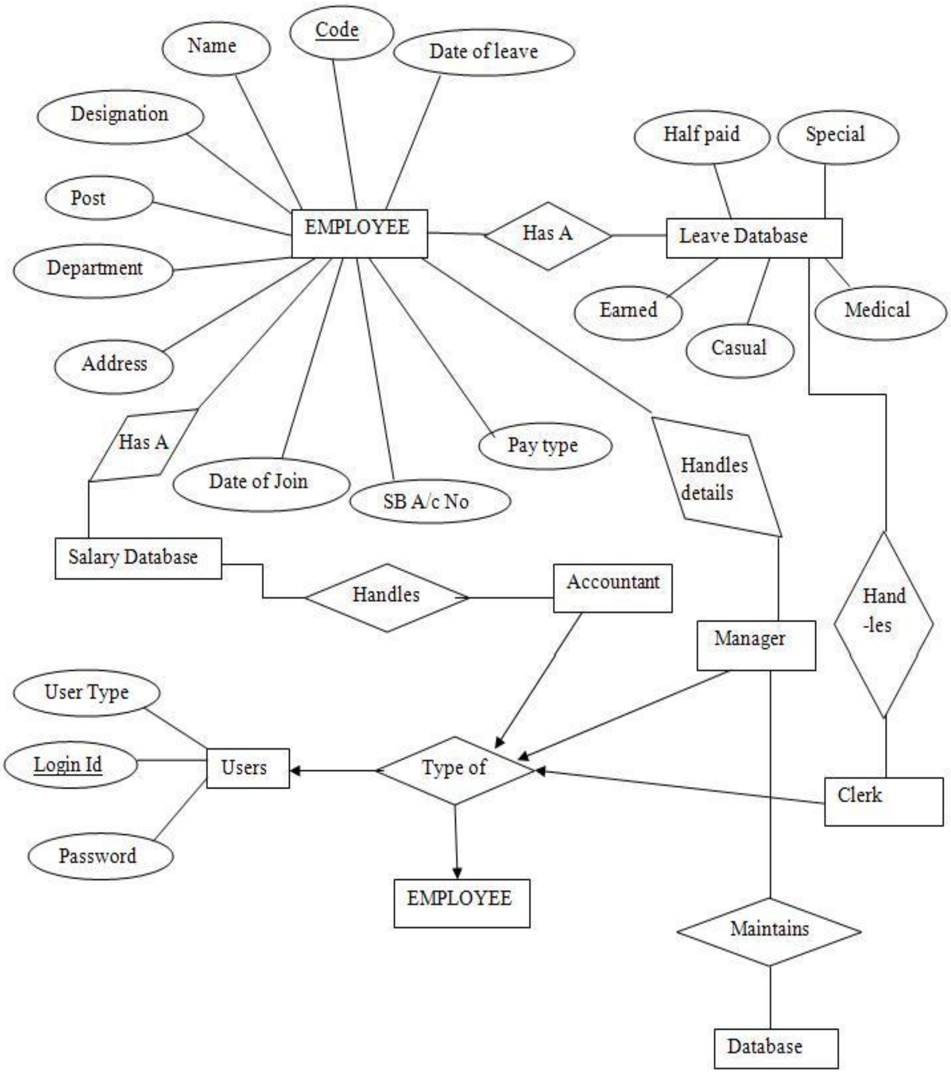


Fig4.4.1 Entity-Relationship Diagram

Chapter 5. Implementation

5.1 Implementation

Accounting system software is implemented in two parts

- Front End
- Back End

5.1.1 Front End

The front end is implemented using HTML & JSP technologies.HTML

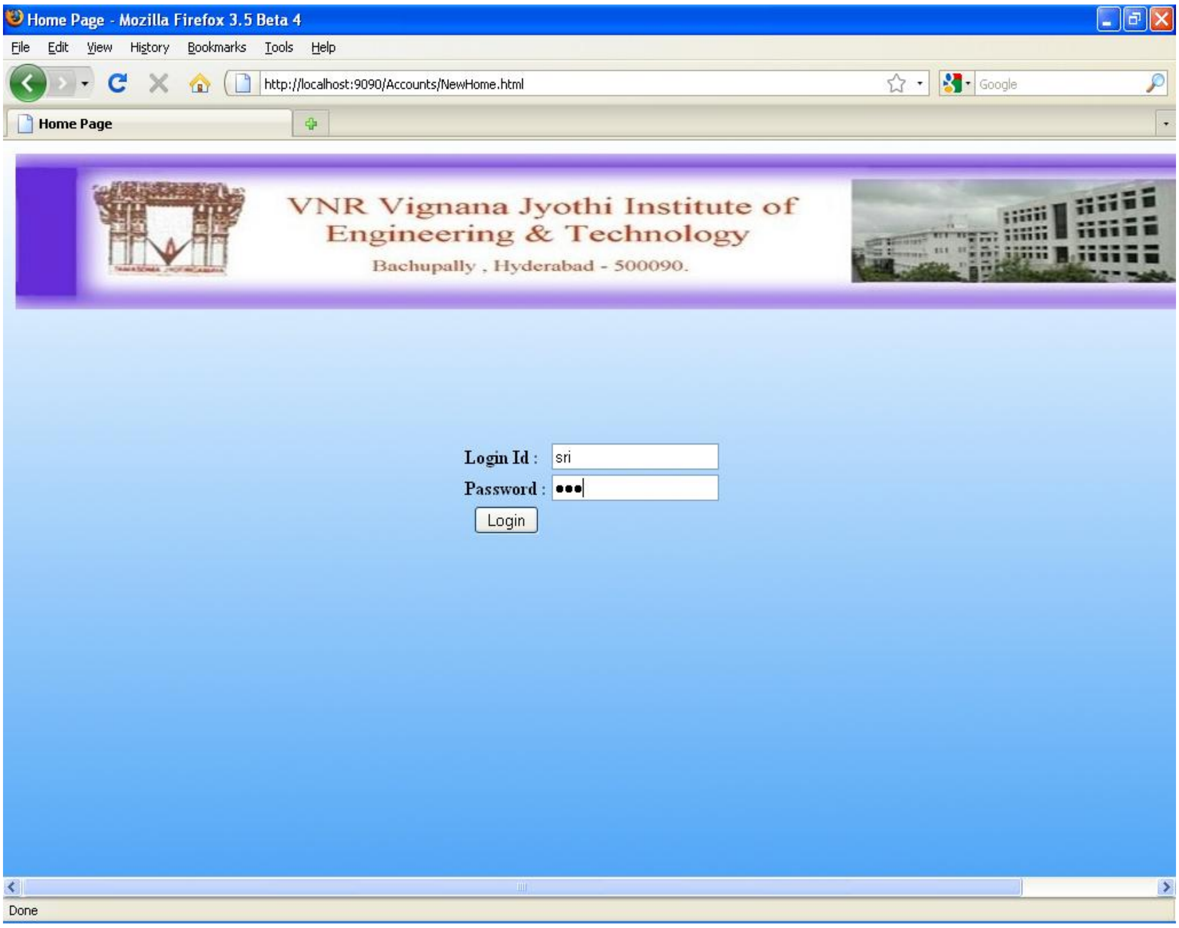
is used to create static web pages where as JSP's is used to create dynamic web pages.

5.1.2 Back End

The back end is implemented using Oracle package. Oracle is a product which is used to handle large databases. It provides features to Create, Alter and Remove table schemas and also to Insert, Delete, Modify the contents of each table.

5.2 Screenshots

5.2.1 Login


Home Page - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/NewHome.html
Google
Home Page
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Login
Done

Login Id : sri

Password : •••

Fig 5.2.1 Login

5.2.2 Manager Home

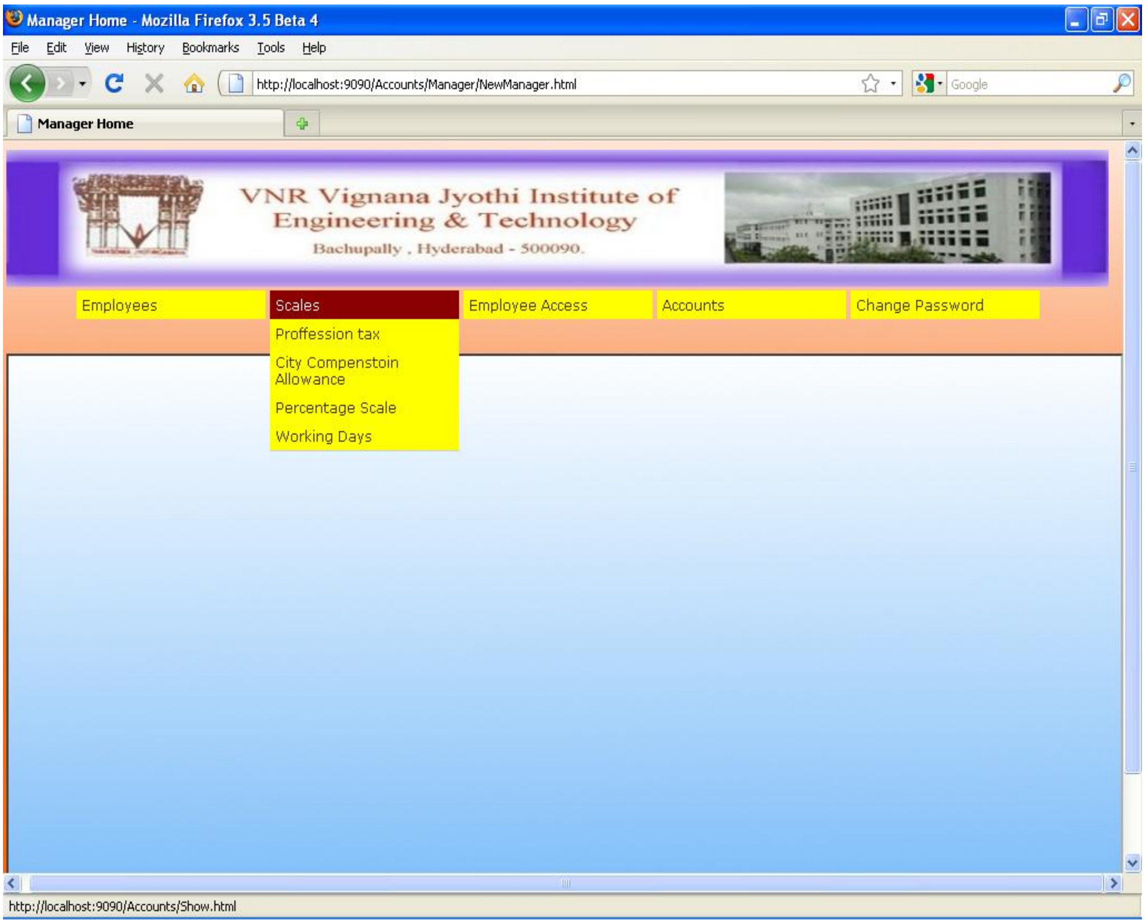
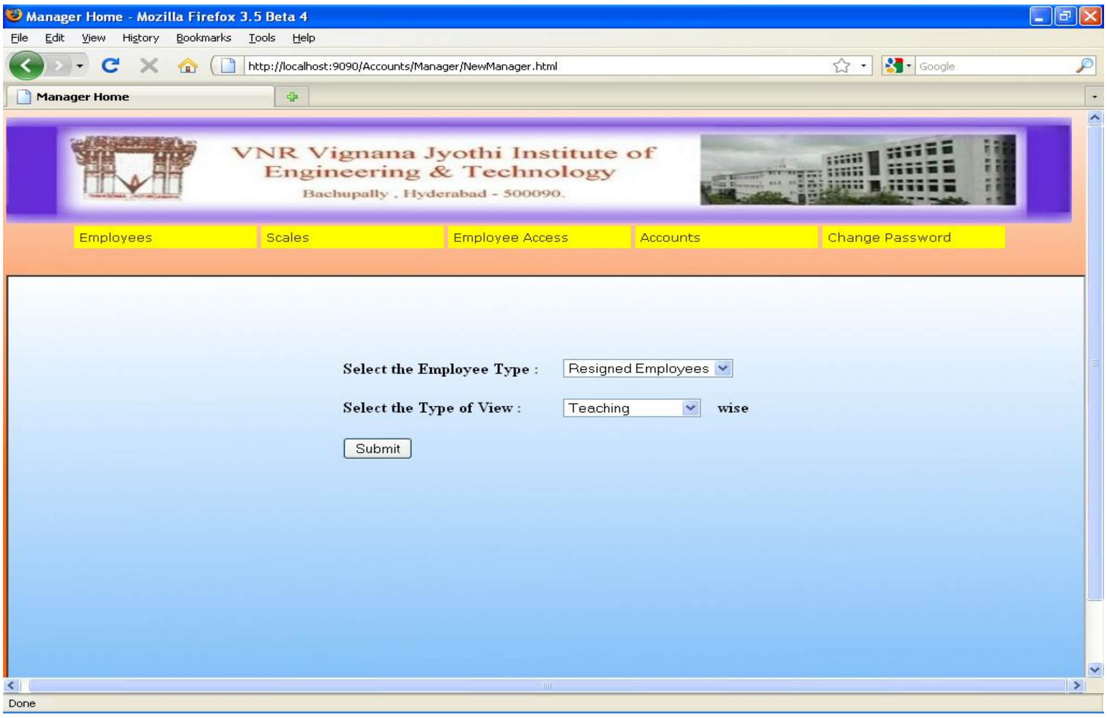

Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Proffession tax
City Compenstoin Allowance
Percentage Scale
Working Days
http://localhost:9090/Accounts/Show.html

Fig 5.2.2 Manager Home

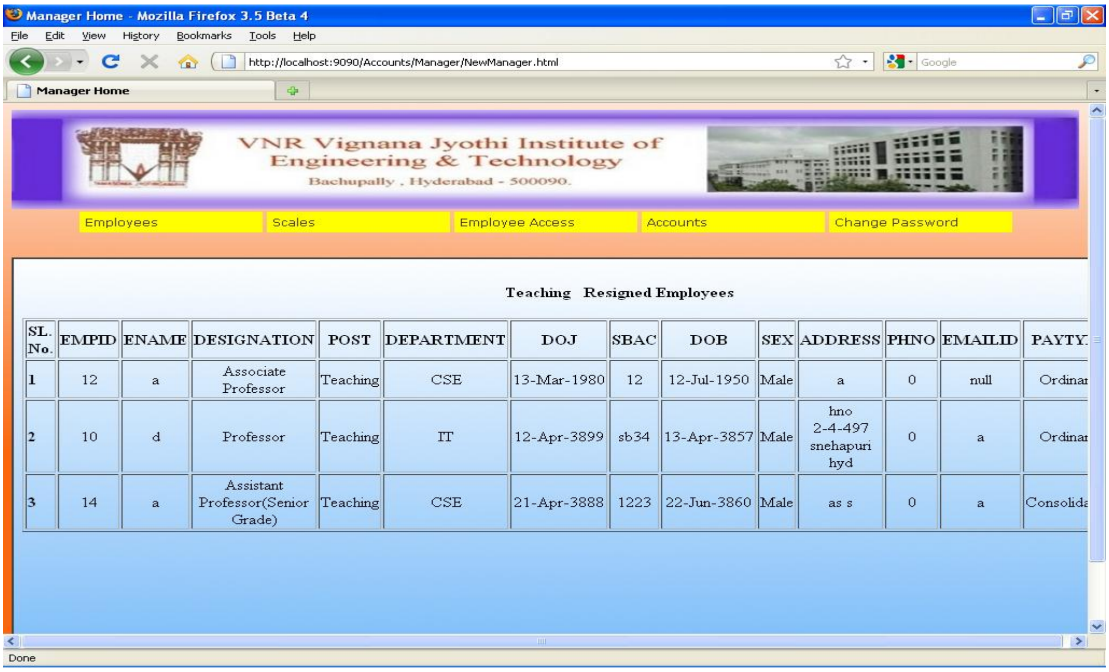
5.2.3 View Employee Details


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Submit
Done

Select the Employee Type : Resigned Employees

Select the Type of View : Teaching wise

Fig 5.2.3(a) View Details Prompt

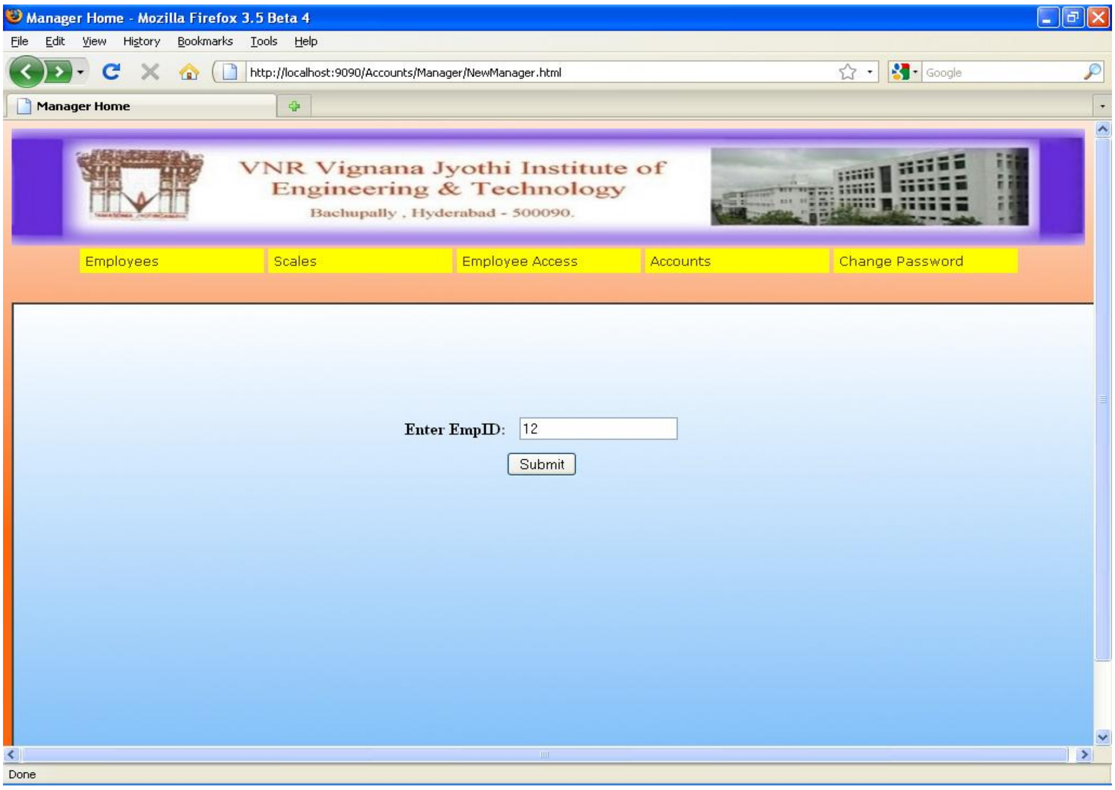

Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Done

Teaching Resigned Employees

SL. No.	EMPID	ENAME	DESIGNATION	POST	DEPARTMENT	DOJ	SBAC	DOB	SEX	ADDRESS	PHNO	EMAILID	PAYTY
1	12	a	Associate Professor	Teaching	CSE	13-Mar-1980	12	12-Jul-1950	Male	a	0	null	Ordinar
2	10	d	Professor	Teaching	IT	12-Apr-3899	sb34	13-Apr-3857	Male	hno 2-4-497 snehapuri hyd	0	a	Ordinar
3	14	a	Assistant Professor(Senior Grade)	Teaching	CSE	21-Apr-3888	1223	22-Jun-3860	Male	as s	0	a	Consolida

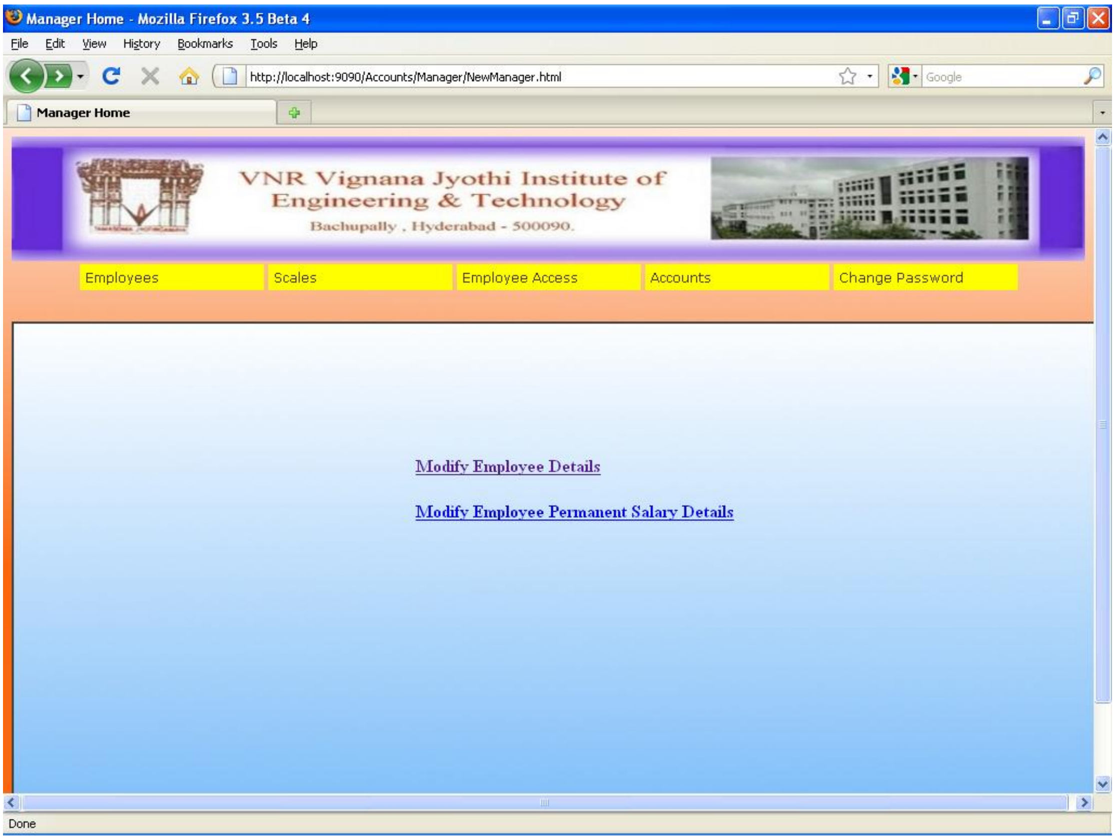
Fig 5.2.3(b) Display Employee details

5.2.4 Modify Employee Details


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Submit
Done

Enter EmpID: 12

Fig 5.3.4(a) Modify Details Prompt

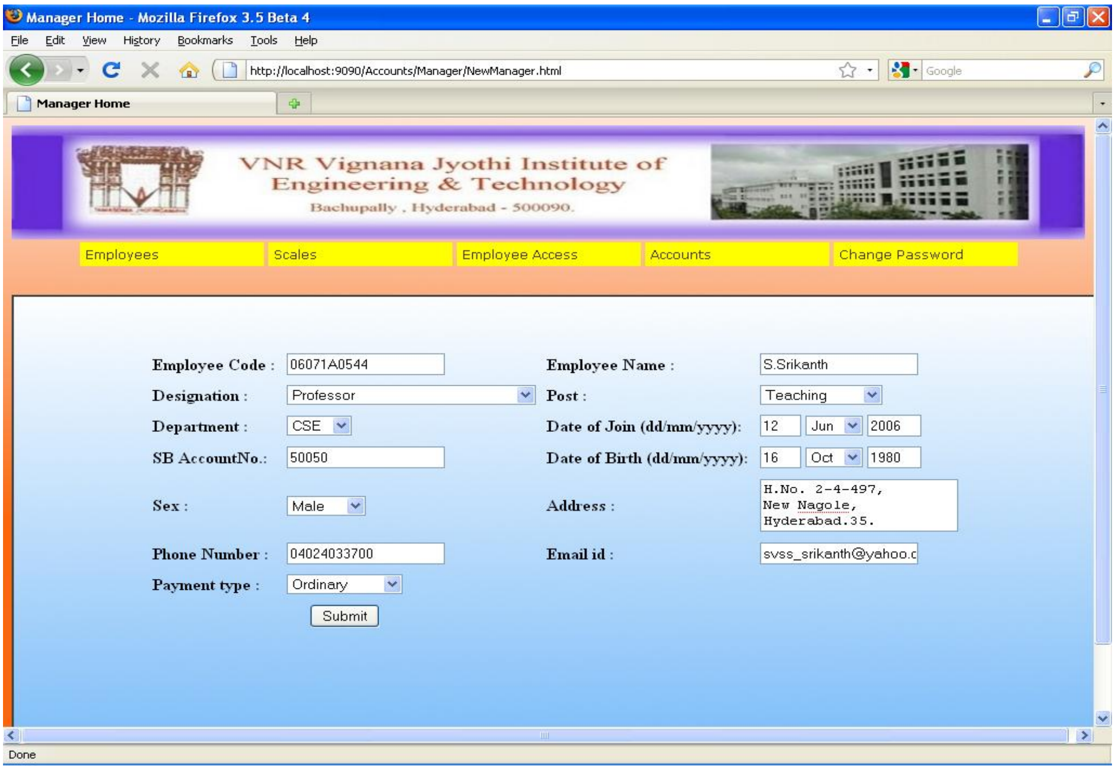

Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Done

Modify Employee Details

Modify Employee Permanent Salary Details

Fig 5.3.4(b) Modify Details

5.2.5 Add Employee


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Submit
Done

Employee Code : 06071A0544

Employee Name : S.Srikanth

Designation : Professor

Post : Teaching

Department : CSE

Date of Join (dd/mm/yyyy): 12 Jun 2006

SB AccountNo.: 50050

Date of Birth (dd/mm/yyyy): 16 Oct 1980

Sex : Male

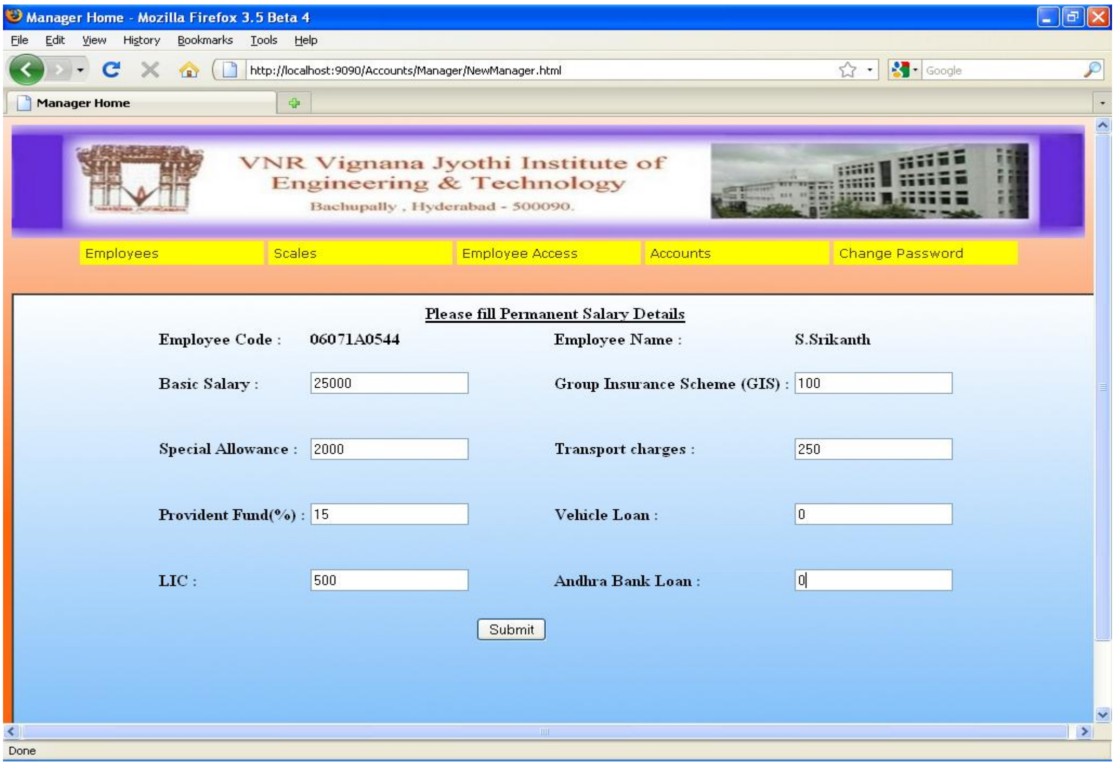
Address : H.No. 2-4-497,
New Nagole,
Hyderabad.35.

Phone Number : 04024033700

Email id : svss_srikanth@yahoo.c

Payment type : Ordinary

Fig 5.2.5(a) Personal Details


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Submit
Done

Please fill Permanent Salary Details

Employee Code : 06071A0544

Employee Name : S.Srikanth

Basic Salary : 25000

Group Insurance Scheme (GIS) : 100

Special Allowance : 2000

Transport charges : 250

Provident Fund(%) : 15

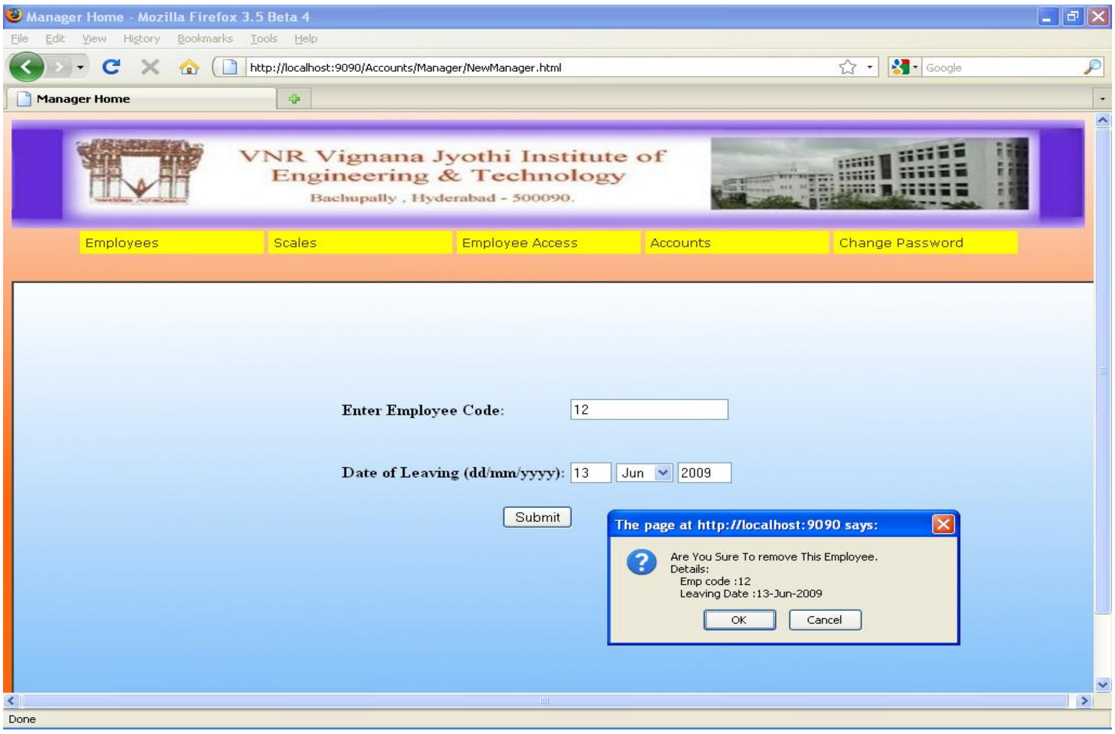
Vehicle Loan : 0

LIC : 500

Andhra Bank Loan : 0

Fig 5.2.5(b) Salary Details

5.2.6 Remove Employee


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Done

Enter Employee Code: 12

Date of Leaving (dd/mm/yyyy): 13 Jun 2009

Submit

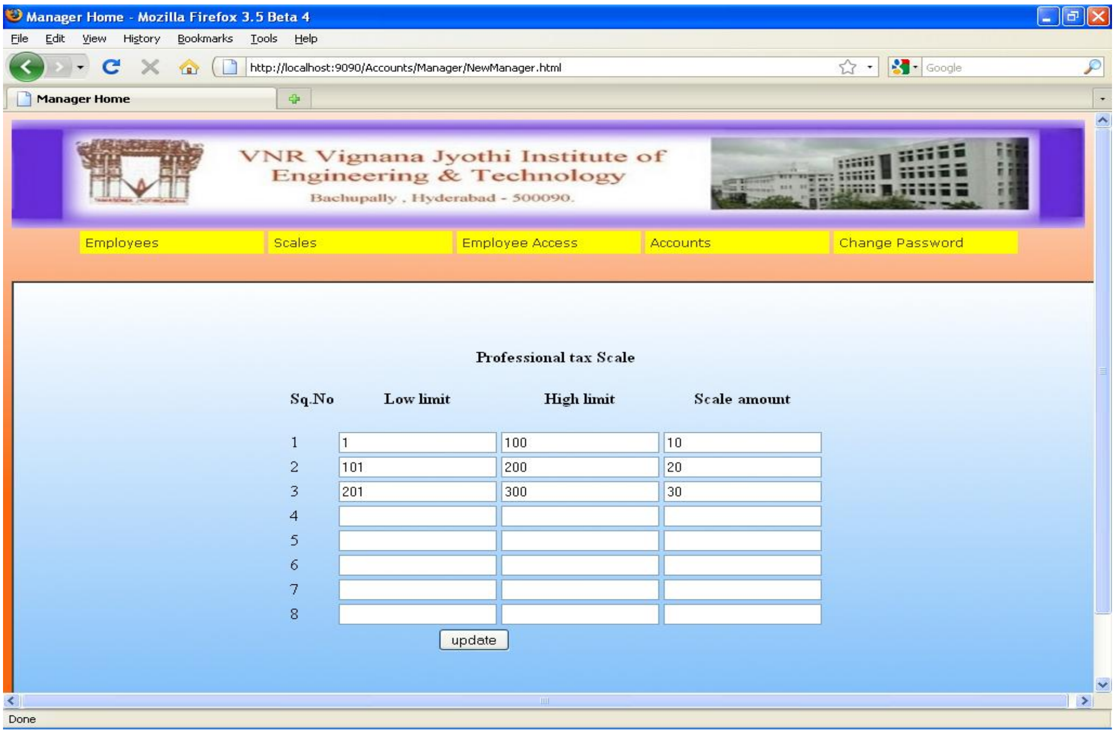
The page at http://localhost:9090 says:

Are You Sure To remove This Employee.
Details:
Emp code :12
Leaving Date :13-Jun-2009

OK Cancel

Fig 5.2.6 Remove Employee

5.2.7 Professional Tax


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Done

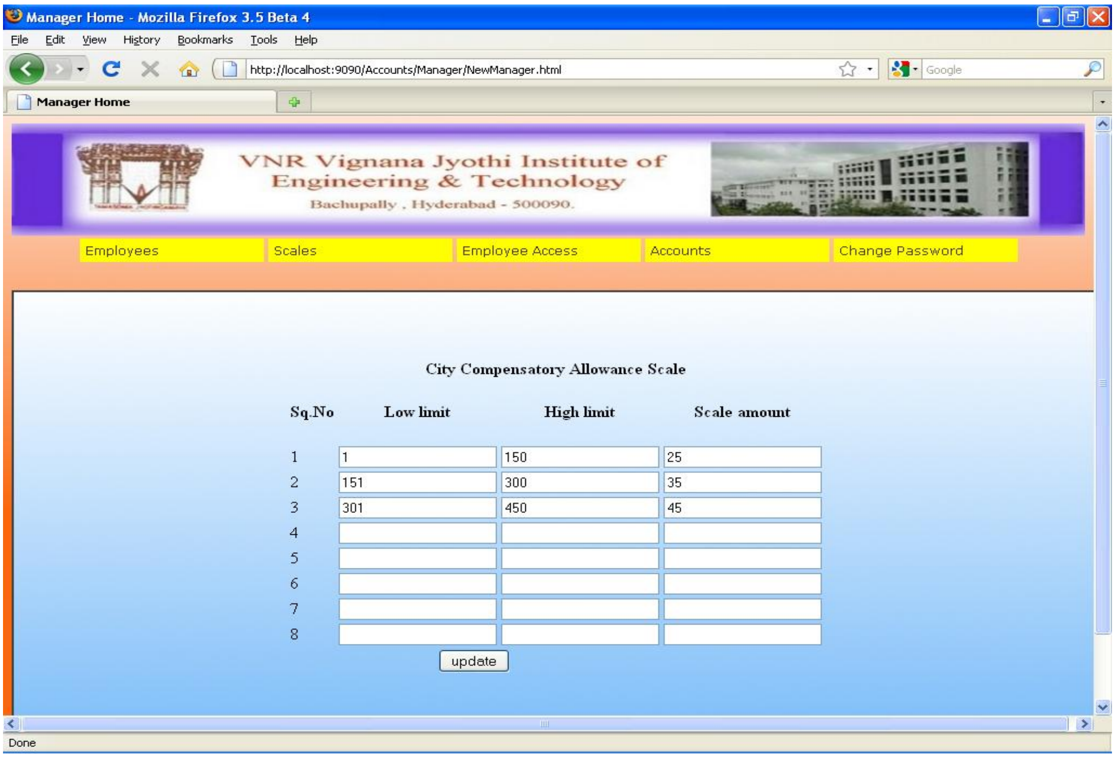
Professional tax Scale

Sq.No	Low limit	High limit	Scale amount
1	1	100	10
2	101	200	20
3	201	300	30
4			
5			
6			
7			
8			

update

Fig 5.2.7 Professional Tax

5.2.8 City Compensatory Allowance

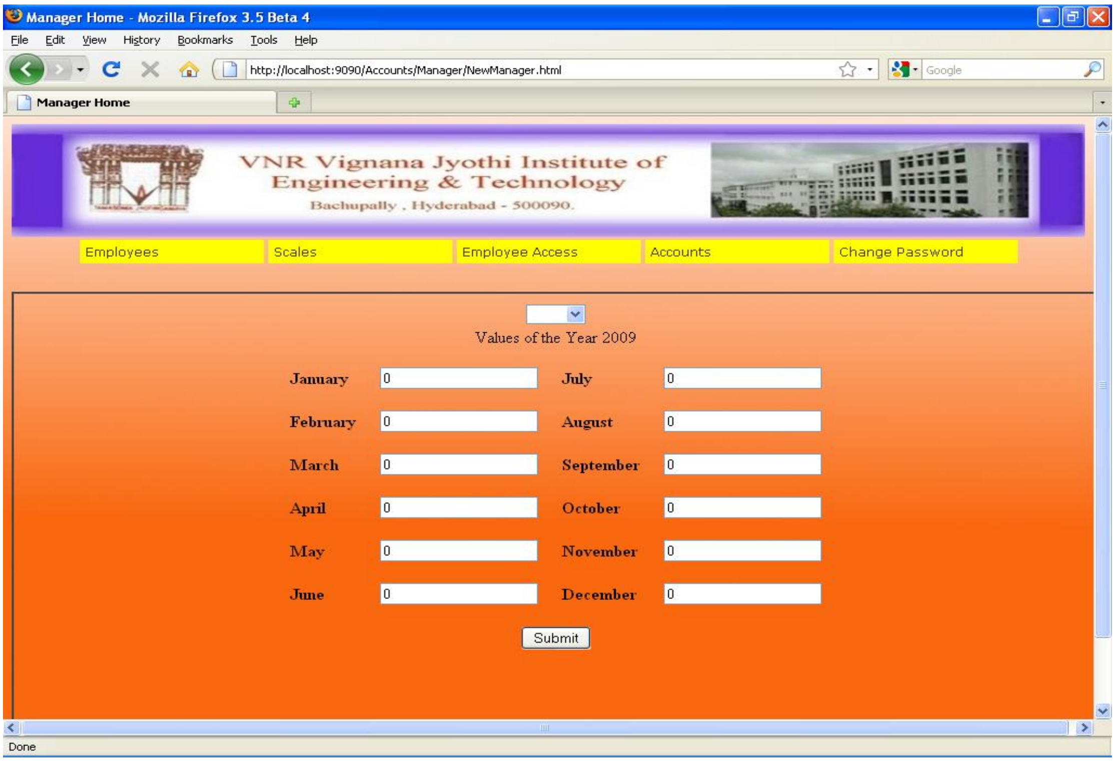

Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
update
Done

City Compensatory Allowance Scale

Sq.No	Low limit	High limit	Scale amount
1	1	150	25
2	151	300	35
3	301	450	45
4			
5			
6			
7			
8			

Fig5.2.8 City Compensatory Allowance

5.2.9 Working Days

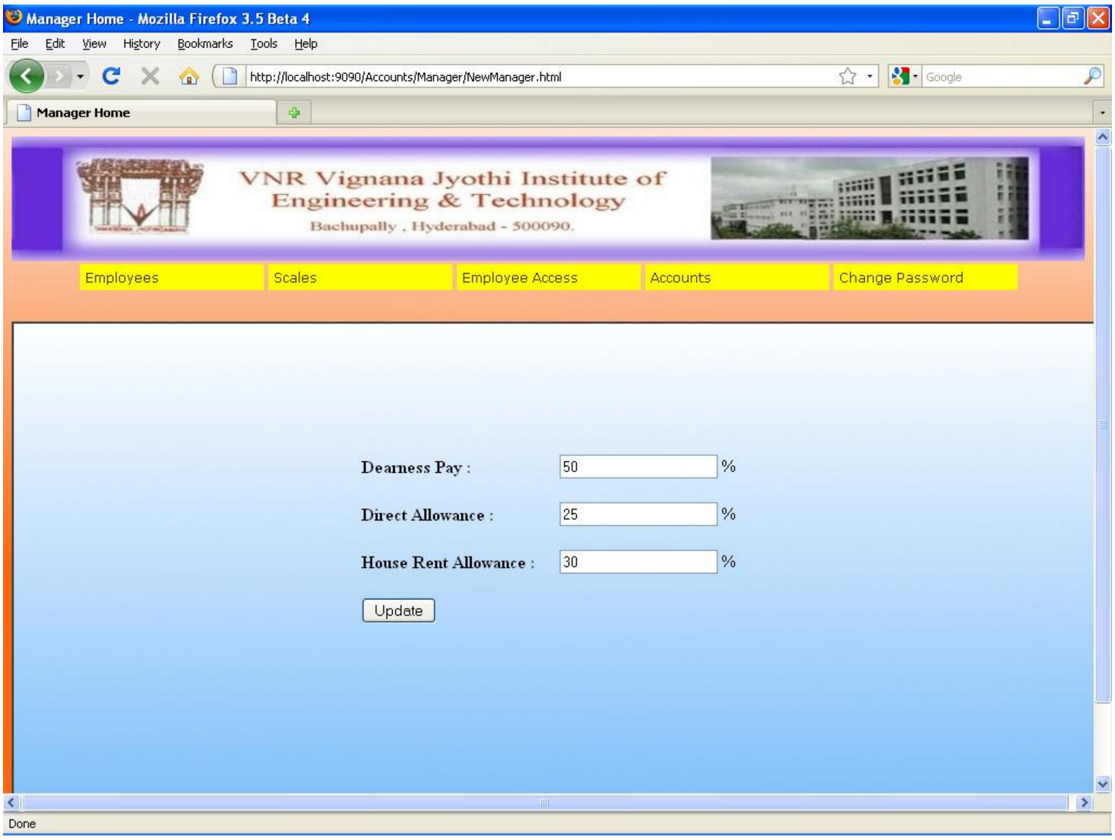

Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Submit
Done

Values of the Year 2009

January	0	July	0
February	0	August	0
March	0	September	0
April	0	October	0
May	0	November	0
June	0	December	0

Fig 5.2.9 Working Days

5.2.10 Percentage Scales


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Update
Done

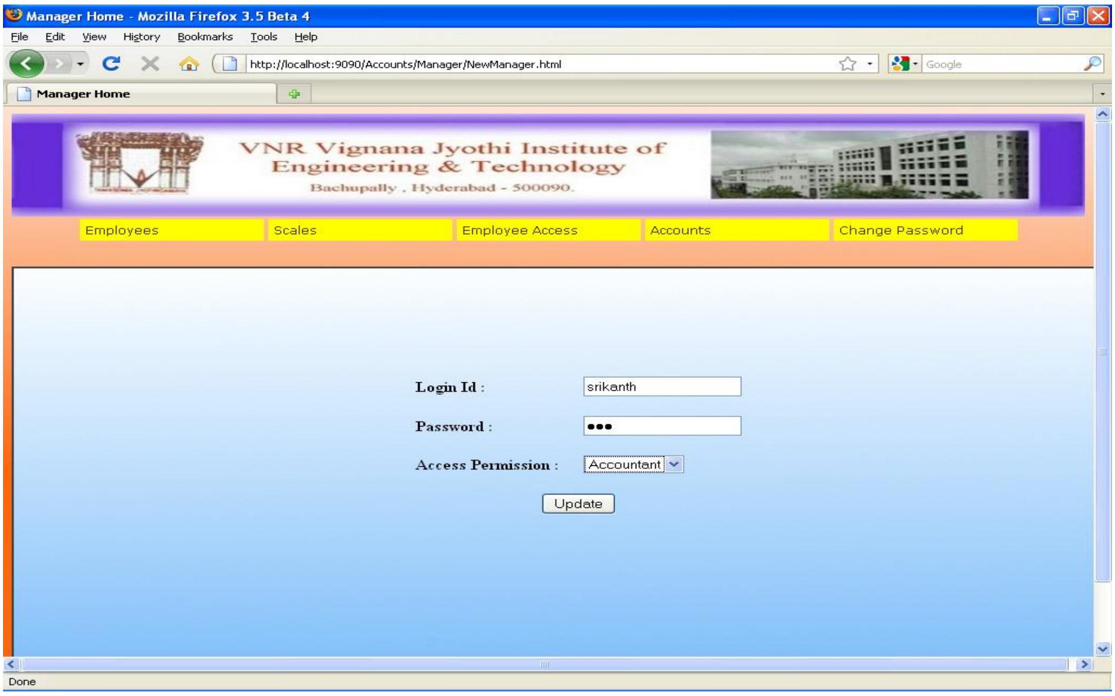
Dearness Pay : 50 %

Direct Allowance : 25 %

House Rent Allowance : 30 %

Fig 5.2.10 Percentage Scales

5.2.11 Provide Access


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Update
Done

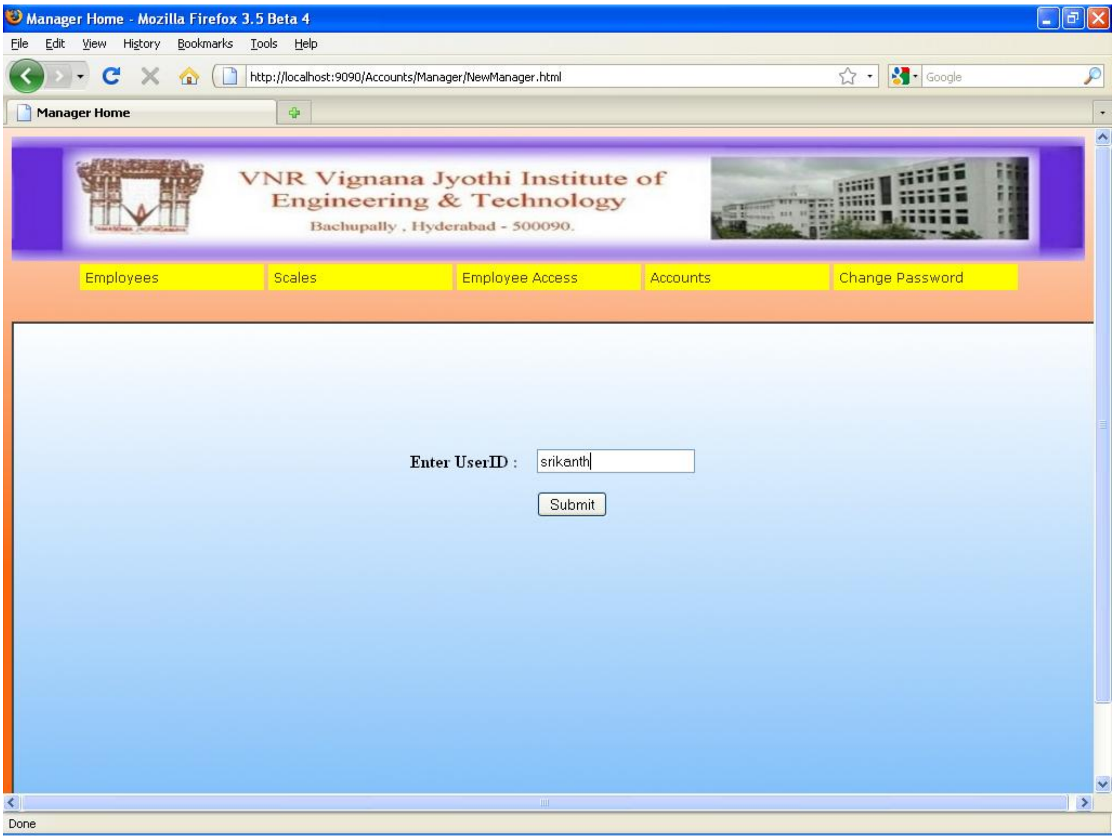
Login Id : srikanth

Password : ●●●

Access Permission : Accountant

Fig 5.2.11 Provide Access

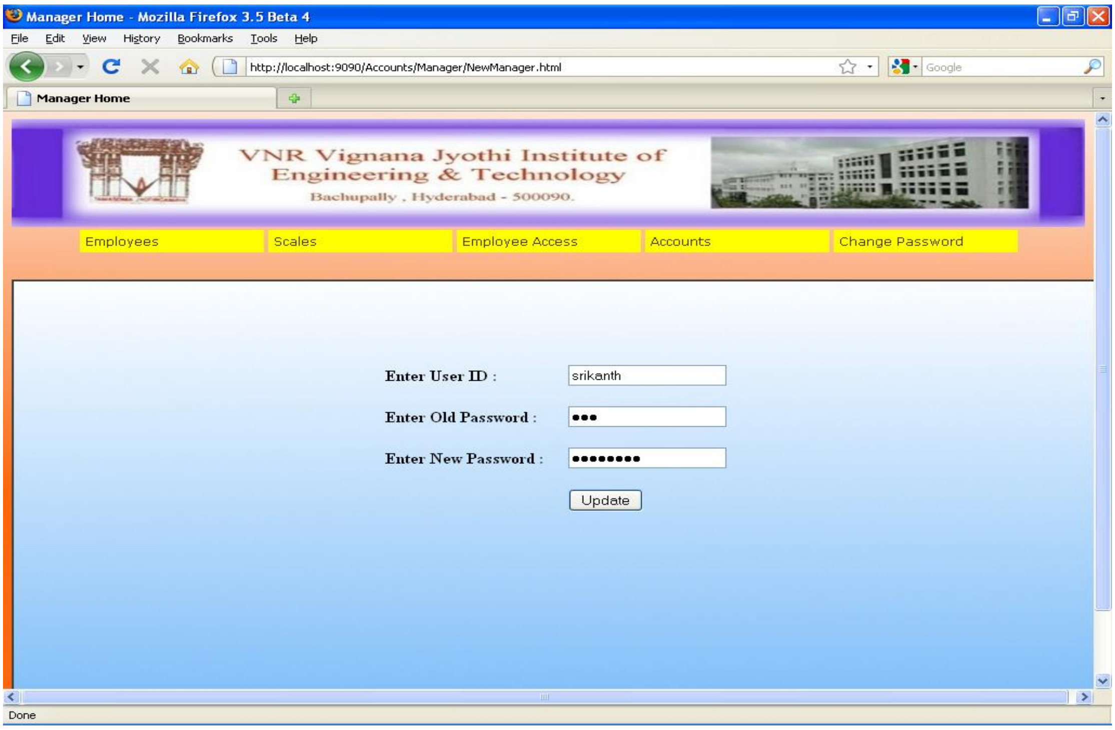
5.2.12AccessRemoval


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Submit
Done

Enter UserID : srikanth

Fig 5.2.12 Access Removal

5.2.13 Change Password


Manager Home - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Manager/NewManager.html
Google
Manager Home
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
Employees
Scales
Employee Access
Accounts
Change Password
Update
Done

Enter User ID : srikanth

Enter Old Password : ●●●

Enter New Password : ●●●●●●●●●

Fig 5.2.13 Change Password

5.2.14 Accountant Home

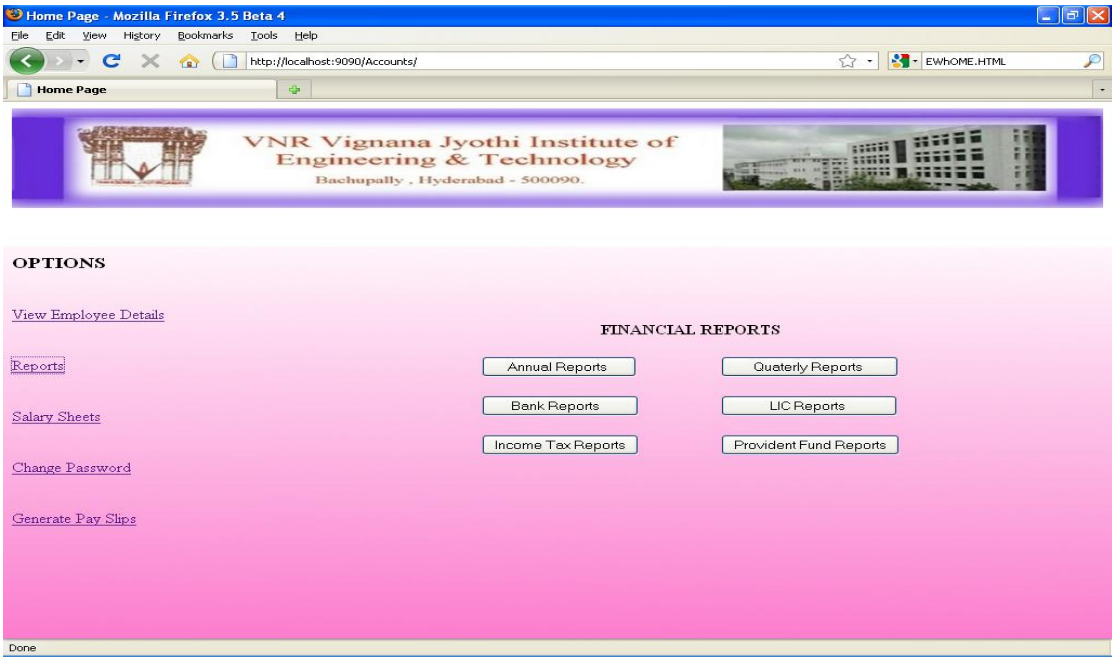


Fig 5.2.14 Accountant Home

5.2.15 Insert Current Salary Details

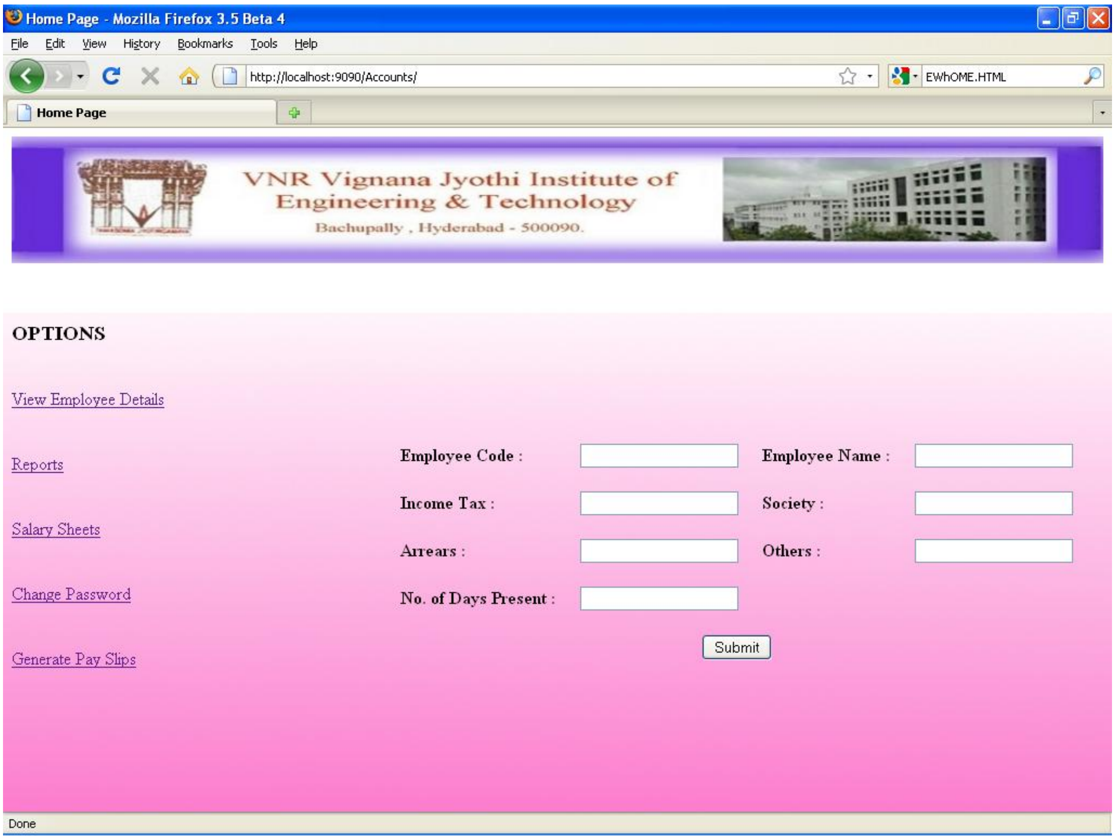
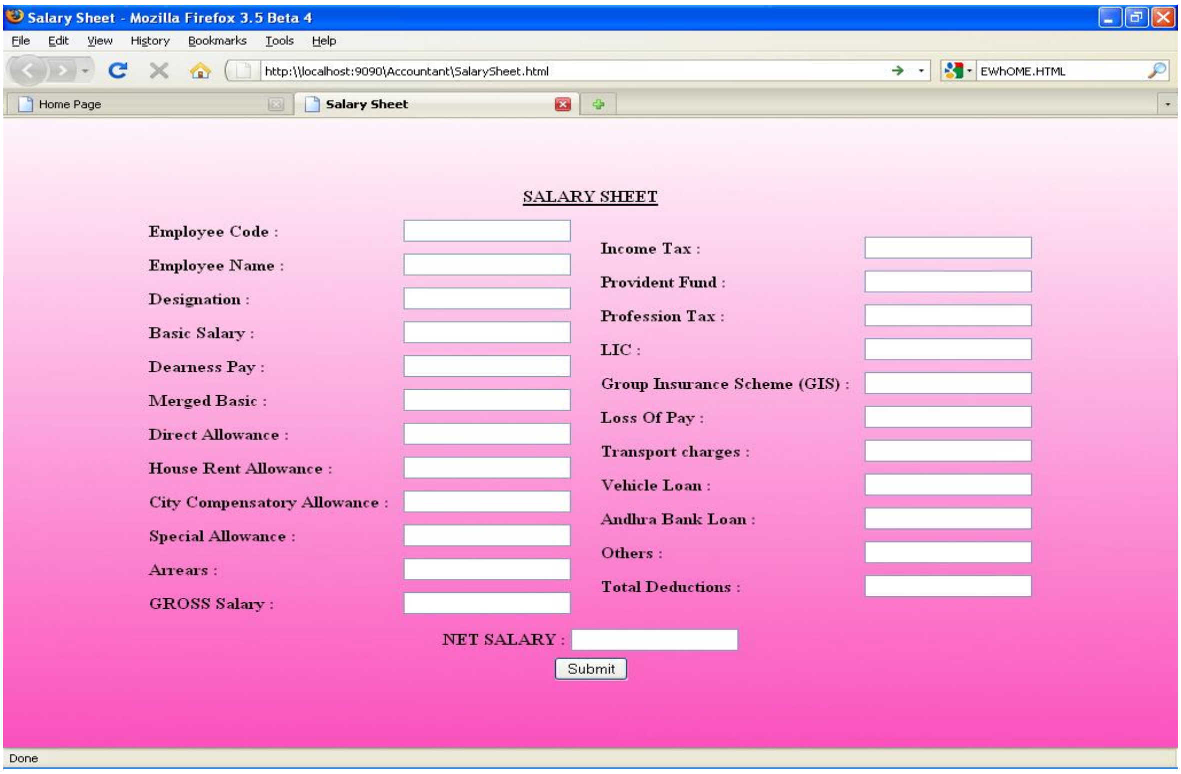

Home Page - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/
EWhOME.HTML
Home Page
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
OPTIONS
View Employee Details
Reports
Salary Sheets
Change Password
Generate Pay Slips
Employee Code :
Employee Name :
Income Tax :
Society :
Arrears :
Others :
No. of Days Present :
Submit
Done

Fig 5.2.15 Insert Current Salary Details

5.2.16 Pay Slip


Salary Sheet - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http:\\localhost:9090\Accountant\SalarySheet.html
EWhOME.HTML
Home Page
Salary Sheet
Submit
Done

SALARY SHEET

Employee Code :

Employee Name :

Designation :

Basic Salary :

Dearness Pay :

Merged Basic :

Direct Allowance :

House Rent Allowance :

City Compensatory Allowance :

Special Allowance :

Arrears :

GROSS Salary :

Income Tax :

Provident Fund :

Profession Tax :

LIC :

Group Insurance Scheme (GIS) :

Loss Of Pay :

Transport charges :

Vehicle Loan :

Andhra Bank Loan :

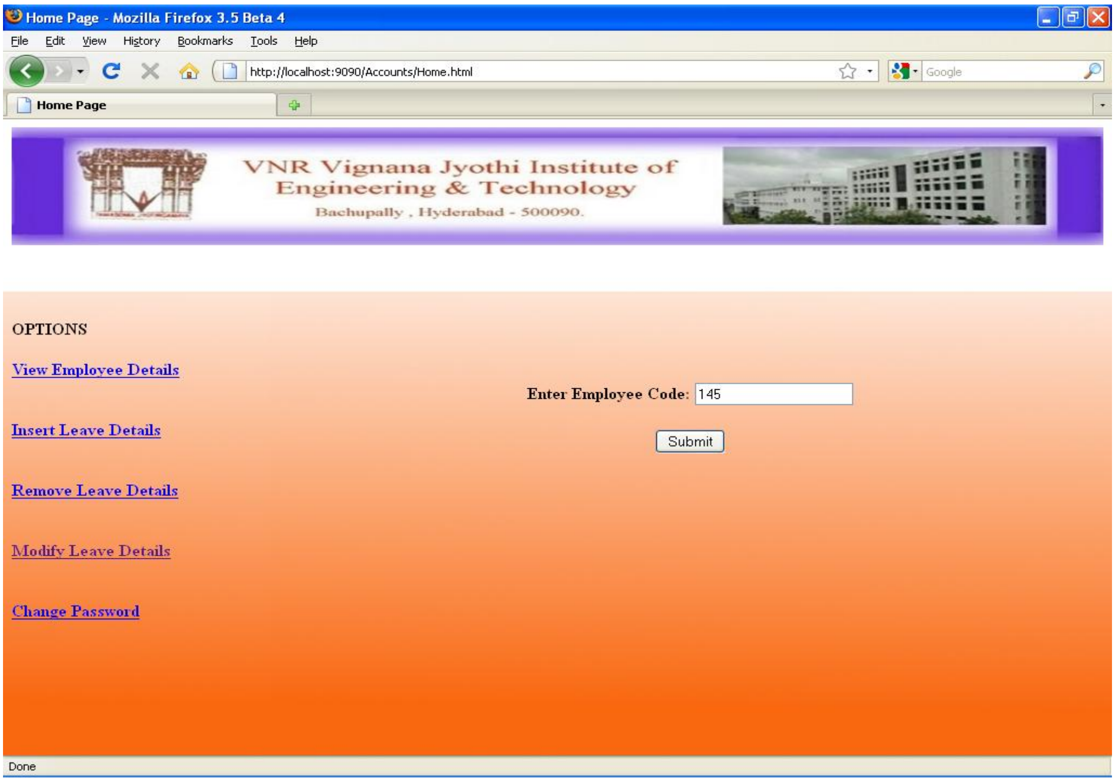
Others :

Total Deductions :

NET SALARY :

Fig 5.2.16 Pay Slip

5.2.17 Clerk Home



Home Page - Mozilla Firefox 3.5 Beta 4
File Edit View History Bookmarks Tools Help
http://localhost:9090/Accounts/Home.html
Google
Home Page
VNR Vignana Jyothi Institute of Engineering & Technology
Bachupally , Hyderabad - 500090.
OPTIONS
View Employee Details
Insert Leave Details
Remove Leave Details
Modify Leave Details
Change Password
Submit
Done

Enter Employee Code: 145

Fig 5.2.17 Clerk Home

5.2.18 Leave Type Select

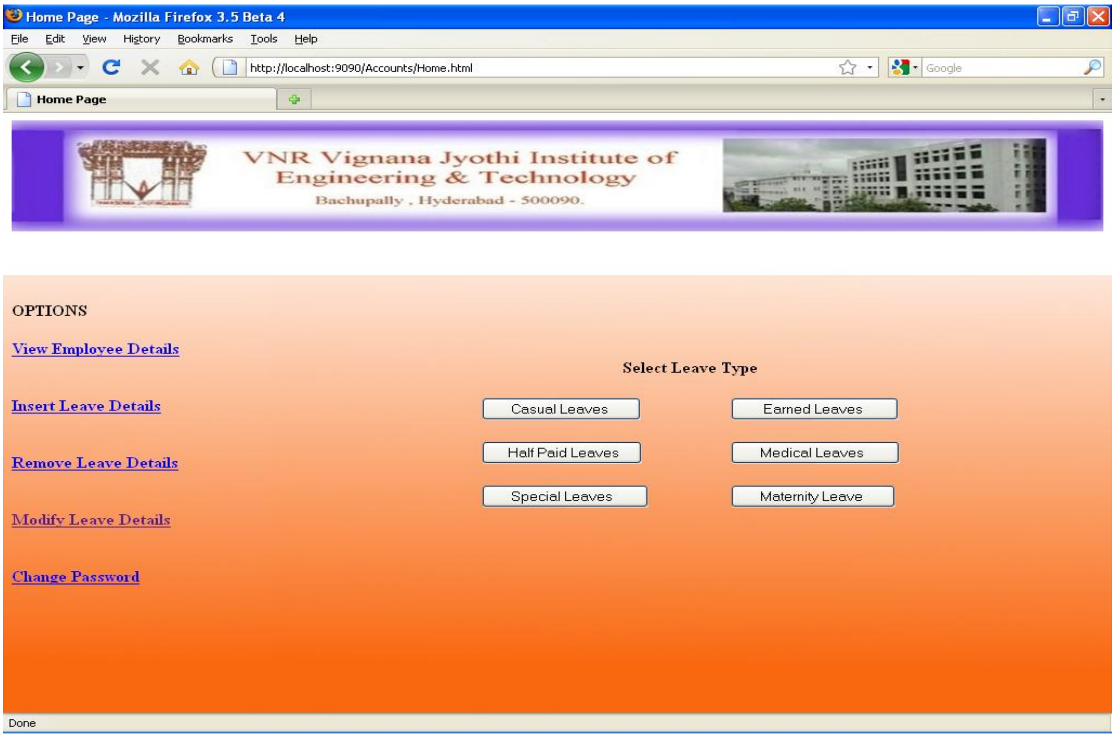


Fig 5.2.18 Leave Type Select

Chapter 6. Testing & Validation

6.1 Testing Fundamentals

Software Testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test, with respect to the context in which it is intended to operate. Software Testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks at implementation of the software. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs. Software Testing can also be stated as the process of validating and verifying that a software program/application/product (1) meets the business and technical requirements that guided its design and development; (2) works as expected; and (3) can be implemented with the same characteristics.

Software Testing, depending on the testing method employed, can be implemented at any time in the development process, however most of the test effort occurs after the requirements have been defined and the coding process has been completed.

6.2 White-Box Testing

White box testing is when the tester has access to the internal data structures and algorithms including the code that implement these.

6.2.1 Types

The following types of white box testing exist:

* API testing (application programming interface) - Testing of the application using Public and Private APIs

* Code coverage - creating tests to satisfy some criteria of code coverage (e.g., the test designer can create tests to cause all statements in the program to be executed at least once)

* Fault injection methods

* Mutation testing methods

* Static testing - White box testing includes all static testing

Code completeness evaluation

White box testing methods can also be used to evaluate the completeness of a test suite that was created with black box testing methods. This allows the software team to examine parts of a system that are rarely tested and ensures that the most important function points have been tested.

Two common forms of code coverage are:

* Function coverage, which reports on functions executed

* Statement coverage, which reports on the number of lines executed to complete the test

They both return code coverage metric, measured as a percentage.

6.3 Black-Box Testing

Black box testing treats the software as a "black box"—without any knowledge of internal implementation. Black box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix, exploratory testing and specification-based testing.

Specification-based testing: Specification-based testing aims to test the functionality of software according to the applicable requirements. Thus, the tester inputs data into, and only sees the output from, the test object. This level of testing usually requires thorough test cases to be provided to the tester, who then can simply verify that for a given input, the output value (or behavior), either "is" or "is not" the same as the expected value specified in the test case.

Specification-based testing is necessary, but it is insufficient to guard against certain risks.

6.4. Testing Phases

6.4.1 Unit Testing

Unit testing is a software verification and validation method in which a programmer tests that individual units of source code are fit for use. A unit is the smallest testable part of an application. In procedural programming a unit may be an individual program, function, procedure, etc., while in object-oriented programming, the smallest unit is a class, which may belong to a base/super class, abstract class or derived/child class.

6.4.2 Integration testing

Integration testing (sometimes called Integration and Testing, abbreviated "I&T") is the activity of software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before system testing.

Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

6.4.3 System Testing

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic.

6.5 Test Cases:

Test cases are used to simulate the real-time scenario under which a software is supposed to run. A test case helps to find, classify and profile the bugs in the software system. In order to see if the system is working as per the requirements without any hassles, we propose the testing the following modules.

6.5.1 User Login

Purpose: To test if only authorized users can login.

INPUTS	INPUT SPECIFICATIONS	EXPECTED RESULT	OBSERVED RESULT
UserID Password	Valid Valid	User Logs in	User Logs in
UserID Password	Valid Invalid	Access Denied	Access Denied
UserID Password	Invalid Valid	Access Denied	Access Denied
UserID Password	Invalid Invalid	Access Denied	Access Denied
UserID Password	No Input No Input	Access Denied	Access Denied

6.5.2 Changing User Password

Purpose: This test is used to determine if password is changed only when all

INPUTS	INPUT SPECIFICATIONS	EXPECTED RESULT	OBSERVED RESULT
UserID Old Password New Password	Valid Valid Valid	Changed	changed
UserID Old Password New Password	Invalid valid valid	Not changed	Not changed
UserID Old Password New Password	Valid invalid valid	Not changed	Not changed
UserID Old Password New Password	Valid valid invalid	Not changed	Not changed
UserID Old Password New Password	Valid valid empty	Not changed	Not changed

Chapter 7. Conclusion

7.1 Future Enhancements

This project is developed as per the requirements of the organization. The enhancements possible for the software are as follows.

- This project is scalable as it covers only the required features for the institute. We can include other accounting features when ever required by the organization.
- We can improve security features by including encryption mechanisms for the data being stored.
- We can develop different types of reports as required by the customer.
- The personal data fields of employees can be modified i.e. the database can be scalable.
- The other advanced views can be implemented.

7.2 Conclusion

This project 'Accounting System' is a web based application developed for the institute VNR Vignana Jyothi Institute of Engineering & Technology. This project offers an easier and faster process for handling the accounting process of the organization. It offers a smooth flow in the functionality and reduces the laborious work for the employees in the administrative office.

Chapter 8. Bibliography

8.1 References

Sl. no.	Author (s)	Book title	Publisher
1.	Dietel & Dietel	Java: How To Program 6th Edition	Pearson Educations
2.	Simon Brown, Robert Burdick	Professional JSP 2nd Edition	WROX
3.	Ramakrishnan and Gehrke	Database Management Systems	Tata McGraw-Hill Companies, The
4.	Grady Booch, James Rumbaugh	The Unified Modeling Language User Guide (Second Edition)	Addison-Wesley
5.	Roger S. Pressman	Software Engineering : A practitioner's Approach	Tata McGraw-Hill Companies
6.	Chris Bates	Web Programming & Building Internet Applications	Wiley India

8.2 Online References

- http://java.sun.com/products/jsp/docs.html
- http://www.uml.org/
- http://www.omg.org/gettingstarted/what_is_uml.html
- http://education.oracle.com/
- http://www.orafaq.com/
- http://w3schools.com/