**CHAPTER 1**

**INTRODUCTION**

A database management system (DBMS) refers to the technology for creating and managing databases. DBMS is a software tool to organize (create, retrieve, update and manage) data in a database.

A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the [database](https://searchsqlserver.techtarget.com/definition/database) and end users or [application programs](https://searchsoftwarequality.techtarget.com/definition/application), ensuring that data is consistently organized and remains easily accessible. The DBMS manages three important things: the data, the database [engine](https://whatis.techtarget.com/definition/engine) that allows data to be accessed, locked and modified -- and the database [schema](https://searchsqlserver.techtarget.com/definition/schema), which defines the database’s logical structure. These three foundational elements help provide [concurrency](https://searchoracle.techtarget.com/definition/concurrent-processing), security, [data integrity](https://searchdatacenter.techtarget.com/definition/integrity) and uniform administration procedures. Typical database administration tasks supported by the DBMS include [change management](https://searchcio.techtarget.com/definition/change-management), performance monitoring/tuning and [backup](https://searchdatabackup.techtarget.com/definition/backup) and [recovery](https://searchdisasterrecovery.techtarget.com/definition/data-recovery). Many database management systems are also responsible for automated [rollbacks](https://searchsqlserver.techtarget.com/definition/rollback), restarts and recovery as well as the [logging](https://whatis.techtarget.com/definition/log-log-file) and auditing of activity.

The main aim of a DBMS is to supply a way to store up and retrieve database information that is both convenient and efficient. By data, we mean known facts that can be recorded and that have embedded meaning. Normally people use software such as DBASE IV or V, Microsoft ACCESS, or EXCEL to store data in the form of database. A datum is a unit of data. Meaningful data combined to form information. Hence, information is interpreted data - data provided with semantics. MS. ACCESS is one of the most common examples of database management software.

Database systems are meant to handle a large collection of information. Management of data involves both defining structures for storage of information and providing mechanisms that can do the manipulation of those stored information. Moreover, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access. According to the principles of database systems, the data is stored in such a way that it acquires lot less space as the redundant data (duplicate data) has been removed before storage. Along with storing the data in an optimized and systematic manner, it is also important that we retrieve the data quickly when needed. Database systems ensure that the data is retrieved as quickly as possible.

The DBMS is perhaps most useful for providing a centralized view of data that can be accessed by multiple users, from multiple locations, in a controlled manner. A DBMS can limit what data the end user sees, as well as how that end user can view the data, providing many views of a single database schema. End users and software programs are free from having to understand where the data is physically located or on what type of storage media it resides because the DBMS handles all requests.

SQL stands for Structured Query Language. It is used for storing and managing data in relational database management system (RDMS). It is a standard language for Relational Database System. It enables a user to create, read, update and delete relational databases and tables. All the RDBMS like MySQL, Informix, Oracle, MS Access and SQL Server use SQL as their standard database language. SQL allows users to query the database in a number of ways, using English-like statements.

**1.1 Objectives**

* This project aims to simplify the task of maintaining records of the employee of Company.
* To develop an well-designed database to store employee information.
* Provides full functional reports to management of company.
* The objective of this project is to provide a comprehensive approach towards the management of employee information.
* **1.2 Proposed model**
* In this model data model is to store the data regarding the employee details that all are travel in a company. This data will be basically representing the information like employee name, ssn , place etc. Main purpose is to provide easy collection of employee details.
* In order to fetch details of a employee, first they should verify ssn. Then system will allow to view their details. For admin to log in to the system should enter username and password.

**Chapter 2**

**FRONT END DESIGN**

NetBeans is an open-source project dedicated to providing rock solid software development products that address the needs of developers, user and the business who rely on NetBeans as a basis for their products; particularly to enable them to develop these products quickly, efficiently and easily by leveraging the strengths of the Java platform and other relevant industry standards.

Application can install modules dynamically. Any application can include the Update Centre module to allow users of the application to download digitally signed upgrades and new features- directly into the running application. Reinstalling an upgrade or a new release does not force user to download the entire application again. The platform offers reusable services common to desktop application, allowing developers to focus on the logic specific to their application. The term “front-end” refers to the user interface, while “back-end” means the server, application and database that work behind the scenes to deliver information to the user. Here, in this project, to enter the Employee information, view them and to enter admins password to login, the front-end used is The Java Net Beans Software.

**2.1 Pseudo Code**

|  |
| --- |
| **1.Employee login**  If the employee has to work in the system he has to login  **Step1**: If the employee is already having an account enter username and password and press login button  **Step2**: If employee is a not registered then employee must create a new account.  **Step3**: Then employee has to entering required detail then submit the detail. Employee can return back by pressing BACK button.  **Step4**: Enter username and password  **Step5**: Enter LOGIN  **Step6**: press VIEW to and enter ssn to display employee details. |

|  |
| --- |
| **2.Admin login**  If the admin has to work with the data he has to login to the system. The admin is provided with a username and password.  **Step1**: Enter the admin name  **Step2**: Enter the password  **Step3**: Press the LOGIN button to login to the system  **Step4**: Press ADD button to add employee details  **Step5**: Press VIEW and enter ssn to view details  **Step6**: Press EDIT to edit the employee details  **Step7**: Press BACK to cancel the process |

**2.2 Button**

First there is a page that has 2 buttons

1.Employee login

2.Admin login

If we press Employee login that button will take us to another page where we can view employee details.

If we press admin login button will take to next page where we can add, delete and view the employee details. Admin can edit the employee details.

**CHAPTER 3**

**BACK END DESIGN**

**3.1 SQL**

SQL stands for Structured Query Language. SQL used to communicate with a database. It is the standard language for relational database management systems. SQL statements are used to perform tasks such as update data on a database, or retrieve data from a database. Most of the database use SQL, most of them have their own additional proprietary extension that are usually only used on their system. However , the standard SQL commands such as “Select”, ”Insert”, ”Update”, ”Delete”, ”Create”, and “Drop” can be used to accomplish almost everything that one needs to do with a database. This tutorial will provide you with the instruction on the basis of each of these commands as well as allow you to put them to practice using SQL Interpreter.

**3.1.1 Features of SQL function**

SQL functions:

Can contain SQL Procedural Language statement and features which support the implementation of control-flow logic around traditional static and dynamic SQL statements. SQL are easy to implement, because they use a simple high-level, strongly typed language. SQL function are more reliable than equivalent external functions. Support input parameters. SQL scalar function returns a table result set. Support a simple, but powerful condition and error-handling model. Allow you to easily access the SQLSTATE and SQLCODE values as special variables.

Reside in the database and are automatically backed up and restored as part of backup and restore operation. Support nested function calls to other SQL functions or functions implemented in other languages. Many SQL statements can be included within SQL function, however these are exception.

**3.2 Database Tables**

The employee management system has 3 tables which are responsible for storing the required information based on the requirements

**Table 3.2.1: ADMIN\_LOGIN TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **NULL?** | **TYPE** | **DESCRIPTION** |
| User name | Not null | varchar | Id |
| Password |  | varchar | Password |

**Table 3.1:admin login table**

**Table 3.2.2: EMPLOYEE\_LOGIN TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **NULL?** | **TYPE** | **DESCRIPTION** |
| SSN | Not null | Integer | Id |
| Password |  | varchar | Password |

**Table 3.2:employee login table**

**Table 3.2.3: EMP\_DETAIL TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **NULL?** | **TYPE** | **DESCRIPTION** |
| Firstname |  | Varchar(10) | Firstname |
| Lastname |  | Varchar(10) | Lastname |
| Gender |  | Varchar(6) | Gender of employee |
| SSN | Not null | Int(10) | SSN of employee |
| DOB |  | Date | Date of birth |
| Department |  | Varchar(20) | Department name |
| Dept\_no |  | Varchar(10) | Department number |
| Phn\_number |  | Bigint(10) | Phone number |
| Address |  | Varchar(25) | Address of employee |

**Table 3.3:emp\_details table**

**Table 3.2.4: INSERT\_LOG\_TRIGGER**

|  |  |  |  |
| --- | --- | --- | --- |
| **NAME** | **NULL?** | **TYPE** | **DESCRPTION** |
| Id | Not null | Int | Id number |
| SSN |  | Int(10) | Ssn of employee |
| Dep\_no |  | Varchar(10) | Department number |
| Action |  | Varchar(10) | Action performed |

**3.3 Schema diagram for Employee management system**

**Emp\_account**

|  |  |
| --- | --- |
| SSN | Password |

**Admin\_account**

|  |  |
| --- | --- |
| User\_name | Password |

**Emp\_details**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Fname | Lname | Gender | SSN | DOB | Department | Dep\_no | Phn\_no | Address |

**3.4 E-R diagram for Employee Management System**

**EMP\_DETAIL**

**manages**

**view**

**LOGIN**

**ADMIN**

**as**

**EMPLOYEE**

3.4:

**Fig 3.4**:E-R diagram

**3.5 Triggers**

In DBMS , a trigger is a SQL procedure that initiates an action when an event (INSERT,DELETE OR UPDATE) occurs. Since triggers are event-driven specialized procedures, they are stored in and managed by the DBMS. In employee management system trigger is used on the event insert.

CREATE TRIGGER ‘log\_trigger’ AFTER INSERT ON ‘emp\_details’

FOR EACH ROW

BEGIN

Insert into insert\_log\_trigger values(null,NEW,ssn,NEW.dep\_no,’inserted’);

END

GO

**3.6 Stored Procedure**

In database management systems an opration that is stored with the database server. Typically, stored procedures are written in SQL.

DELIMITER $$

CREATE DEFINER=’root’@’localhost’ PROCEDURE ‘getdata’(IN

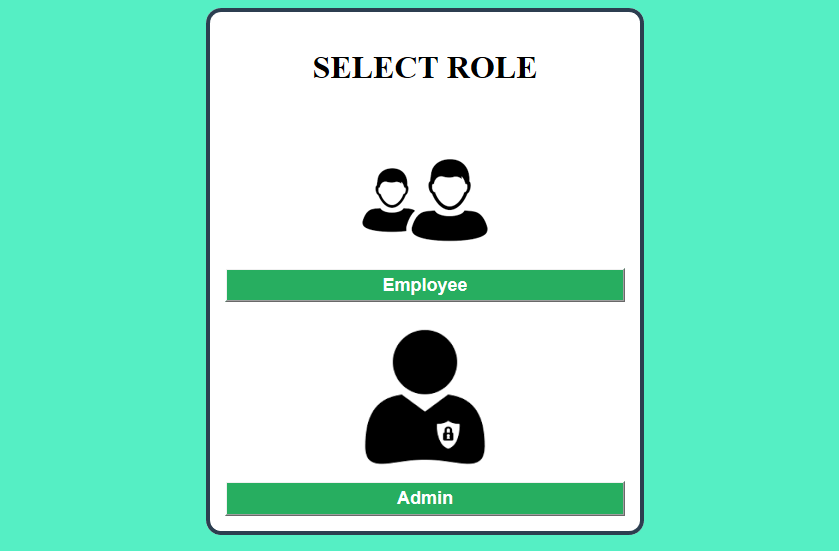
‘sn’ INT(10))

SELECT \* FROM emp\_details where ssn=sn$$

DELIMITER ;

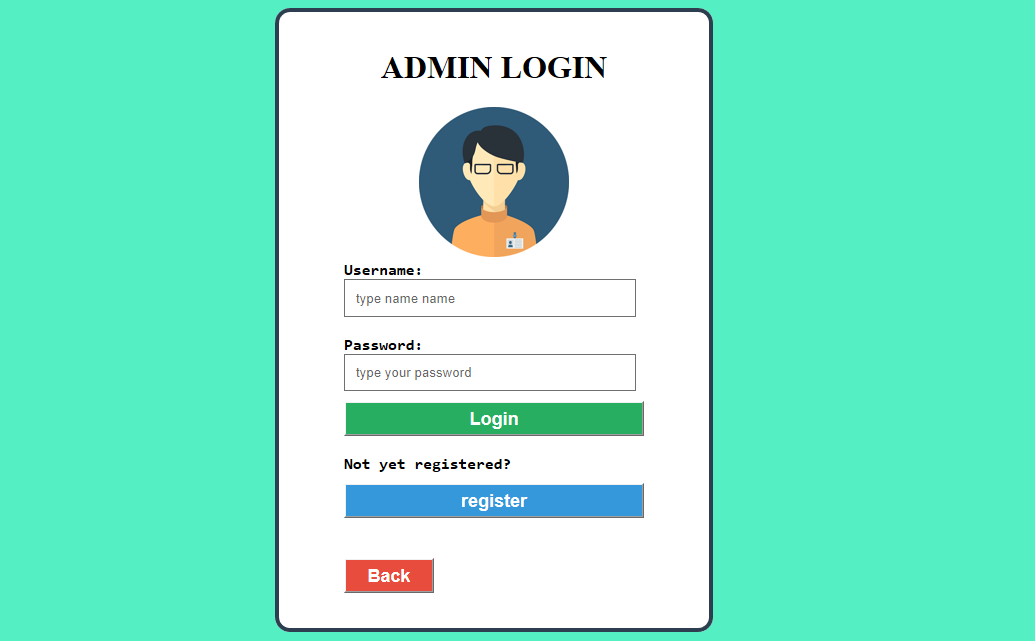
**CHAPTER 4**

**SNAPSHOTS**

****

**Fig 4.1: HOME PAGE**

The above figure shows home page, where admin login and employee login is present.Admin login is used for company maintenance and employment login is used for employee purpose. Employee should employee login that button will take to next page. If employee is not registered employee then employee can sign up then login. If employee is registered student then can directly login.

****

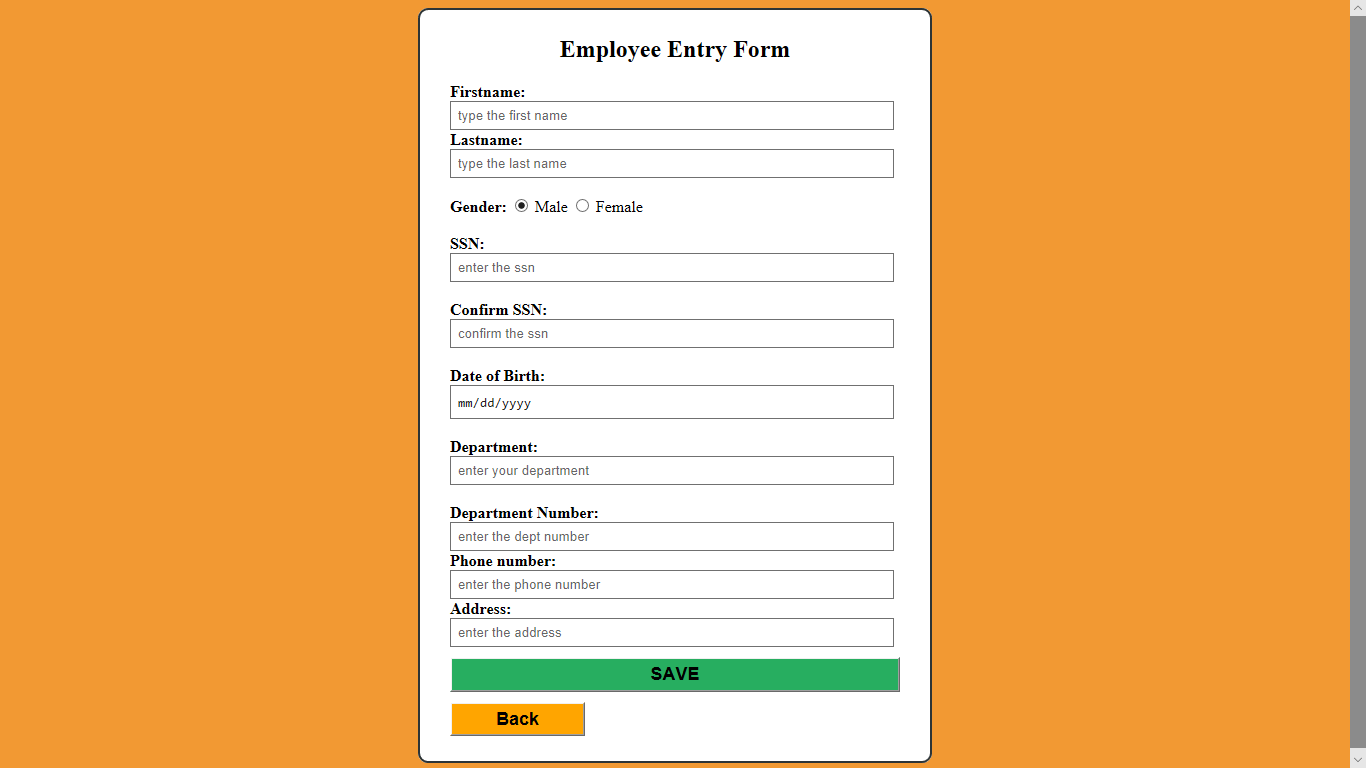
**Fig 4.2: ADMIN LOGIN PAGE**

The above figure shows admin login. Here admin insert username and password to login.



**Fig 4.3: ADMIN WORKSPACE PAGE**

The above figure shows admin workspace page. Here we have four buttons i.e add employee is used to add the employee details. View employee is used to retrieve the employee details. Update employee is used to edit the details. Delete employee is used to delete the details.



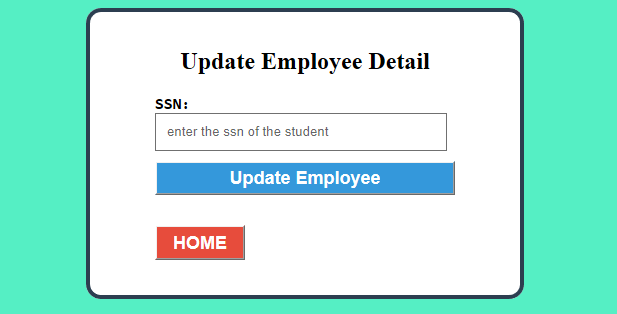
**Fig 4.4: EMPLOYEE ENTRY PAGE**

The above figure shows employee details entry form. Admin will fill the details of employee in the form inorder to add the employee to the database.



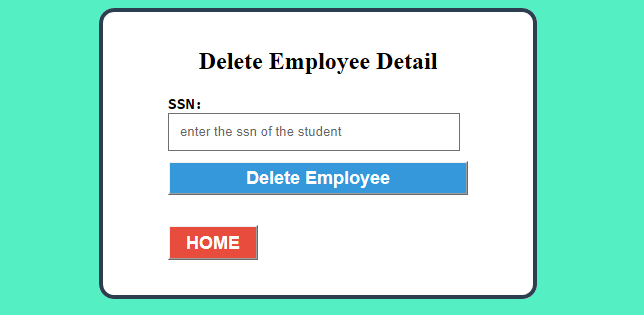
**Fig 4.5: EMPLOYEE VIEW PAGE**

The above figure shows employee view page. Here admin can retrieve employee details by entering ssn of employee.



**Fig 4.6: EMPLOYEE UPDATE PAGE**

This above figure shows employee update page. Here if admin want to edit any of employee details he has to enter the ssn of employee.

****

**Fig 4.7: EMPLOYEE DELETE PAGE**

This above figure shows employee delete page. If admin want to delete any of employee details he has to enter ssn of that employee.

****

**Fig 4.8: EMPLOYEE LOGIN**

This above figure shows employee login page. After employee being sign up to system he can login to system by entering ssn and password.

****

**Fig 4.9: EMPLOYEE SIGN UP PAGE**

The above figure shows sign up page for employee. For a new employee before login to system he or she should sign up to the system



**Fig 4.10: EMPLOYEE VIEW PAGE**

The above figure show employee view page where the employee will be entering ssn to view the details of employee.

**Chapter 5**

**CONCLUSION**

In this project we developed the employee management system. It is developed in NetBeans and the database has been built in SQL only, keeping in mind the specification of the system. For elaborating the system we have used simple schema and E-R diagrams. This system will be useful in maintaining employee details. This system is user friendly and accurate. The system has reached a steady state where all bugs have been eliminated.

**REFERENCES**

1) Fundamental Database System By Elmasri Navathe.

2)<https://www.google.co.in/search?q=netbeans&rlz=1C1CHZL_enIN770IN771&oq=netbeans&aqs=chrome.0.0l6.8600j0j7&sourceid=chrome&ie=UTF-8>

3) <https://netbeans.org>

4) https://xampp-windows.en.softonic.com/

5)https://netbeans.org/kb/docs/java/quickstart.html