A

PROJECT REPORT

ON

CUSTOMER MANAGEMENT

Submitted by,

Ranjini P Nair

845342

CHN19AJ029

TABLE OF CONTENTS

ABSTRACT

Every organization, whether big or small, has human resource challenges to overcome. They have different customer management needs, therefore we have designed an exclusive employee management system. Customer management is defined as the process of managing the relationship between an organisation, its people and its customers over time. The goal of this project was a simple Customer Relationship Management tool, where users with proper authentication and authorization would be able to perform CRUD operations on a customer data set.

This application is developed for a customer manager where he is able to create new customers and also update, delete and retrieve the customer details. It also has a search module where the manager can search the customer names. Spring is one of the most used Java EE Frameworks. Hibernate is also one the most popular ORM frameworks used for mapping domain classes to relational databases. Here technology stack used was Spring - Hibernate combination ,more specifically Java Spring 5, Hibernate, MySQL and basic HTML with CSS. All the details are stored in a MySQL database. It is easy to update any customer details. All the customer records are integrated and so this makes it user-friendly and easy to use application.

INTRODUCTION

Customer Management system is an application that allows to create and store Customer records. This application is helpful to the department of the organization which maintains data of customers of that organization .It stores all the customer's information in a database. The main aim of Customer Management is to store the details of customers in a web-based system. The module of this system deals with a login page, here the manager in charge can logs into the system by entering the password provided for them. The goal of this project is to perform CRUD operations on a customer data set. The admin/manager have the following authorities:

1. Add a new Customer.
2. Search an existing customer.
3. View the customer list.
4. Update/Delete an existing customer.

In existing system, customer management works on manual process which is a time taking process and data organizing is not efficient. So in order to provide more efficient application using Java ,HTML and CSS, it can be implemented as a web application which will increase the efficiency of the system and data is stored in the database so accessing data is more secure.

This is build on Eclipse IDE for Enterprise Java Developers. Here the project is implemented based on MVC architecture (i.e Model View Controller). Model represents the data (i.e POJO class) .View represents user interface which consist of HTML, CSS, jsp pages. Controller makes the decision where to write the servlets. It consists of three layers Presentation layer, Business layer and Integration layer. User interacts with the database through these layers.

REQUIREMENT SPECIFICATION

This application is developed for a customer manager where he is able to create new customers, and update/delete/retrieve the customer details. It also has a search module where the manager can search the customer names. For building and executing this software, these requirements are used.

1.Java 1.8+

JDK is an acronym for Java Development Kit. JDK is a software development environment which is used to develop java applications and applets. It is a combination of JRE (Java Runtime Environment) + development tools. The JDK contains a private Java Virtual Machine (JVM) .JVM uses class loaders for loading classes. Other functionalities of JVM are verification of byte code and execution.

2. Tomcat Server 7.0 or above

Apache Tomcat (sometimes simply "Tomcat") is an open-source implementation of the Java Servlet, Java Server Pages, Java Expression Language and Web Socket technologies. Tomcat web container will create implicit objects for every JSP pages .It also provides a "pure Java" HTTP web server environment in which Java code can run.

3. MySQL database 5.0 or above

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. The data in MySQL database are stored in tables.

4. Eclipse IDE / STS 2018-2019

Eclipse is an integrated development environment used in computer programming. It contains a base workspace and an extensible plug-in system for customizing the environment. Spring Tool Suite is an IDE to develop Spring applications. It is an Eclipse-based development environment. It provides a ready-to-use environment to implement, run, deploy, and debug the application. It validates our application and provides quick fixes for the applications.

5. Maven 3.0 or above

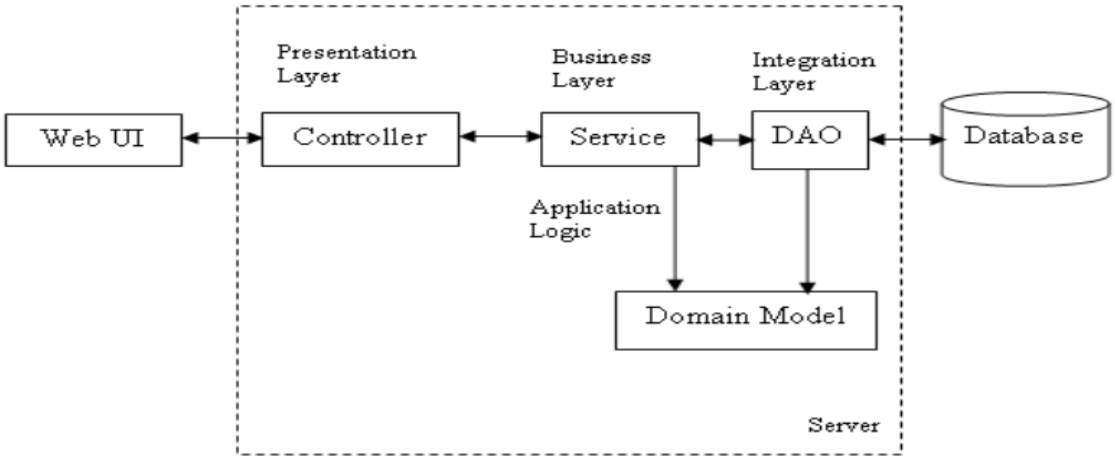
Maven is a software project management and comprehension tool. Based on the concept of a project object model (POM), Maven can manage a project's build, reporting and documentation from a central place.

6. Junit 4

JUnit is a unit testing framework for the Java programming language. JUnit has been important in the development of test-driven development, and is one of a family of unit testing frameworks. Its main use is to write repeatable tests for our application code units.

ARCHITECTURAL DESIGN

OUTLINE OF THE PROJECT



The whole software design can be divided mainly into 3 parts - WebUI, a middleware layer and a persistence layer.

Web UI is nothing but what is seen by the user. It is composed of HTML,CSS, JavaScript components etc and is very appealing and simple. The UI is kept simple so that every guidelines or instructions to the user can be directly given through UI. This layer interacts with the preceding layer.

Next is a server unit that consist of another four blocks - the Presentation layer, Business layer, Integration layer and a domain model.

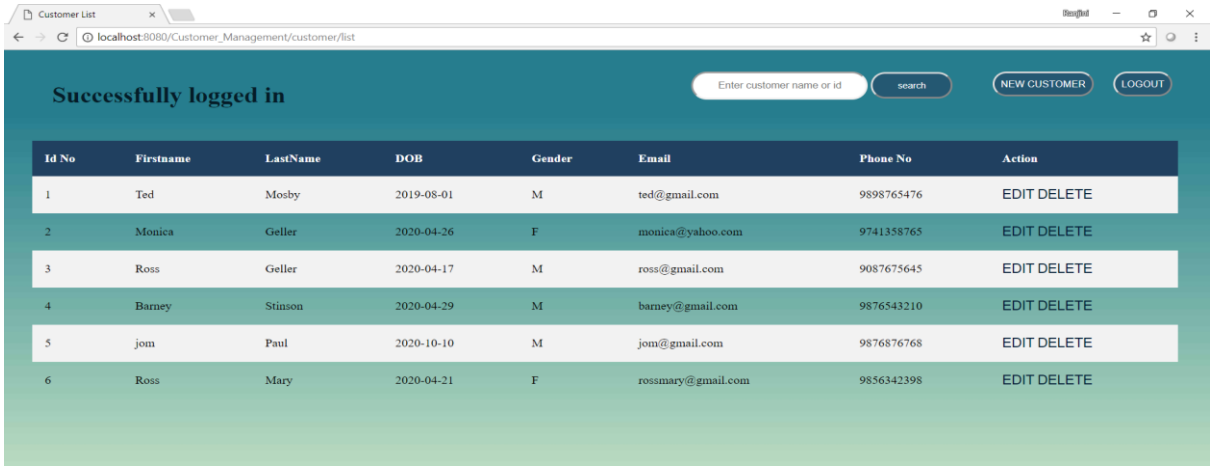
The presentation layer is also called a Controller. This is responsible for all the controlling and coordination between web UI and the server. According to the mapping given to each jsp pages the controller do the necessary actions. Next is the Business layer which is otherwise called as the service layer. The service layer helps in doing all the necessary calculations, application logics and functions related to business level actions. It acts as the connector between controller and Dao. Next is the integration layer or Dao that is responsible for getting connected to the database. All the methods that are developed through this web application like addition of employee details, deletion of employee details, updating, etc. are implemented into database by this layer.

Domain Model is the layer containing entity classes. Every POJO class relating to the application is included under this section. The last layer, that is persistence layer is nothing but the database where all the information are getting stored or manipulated.

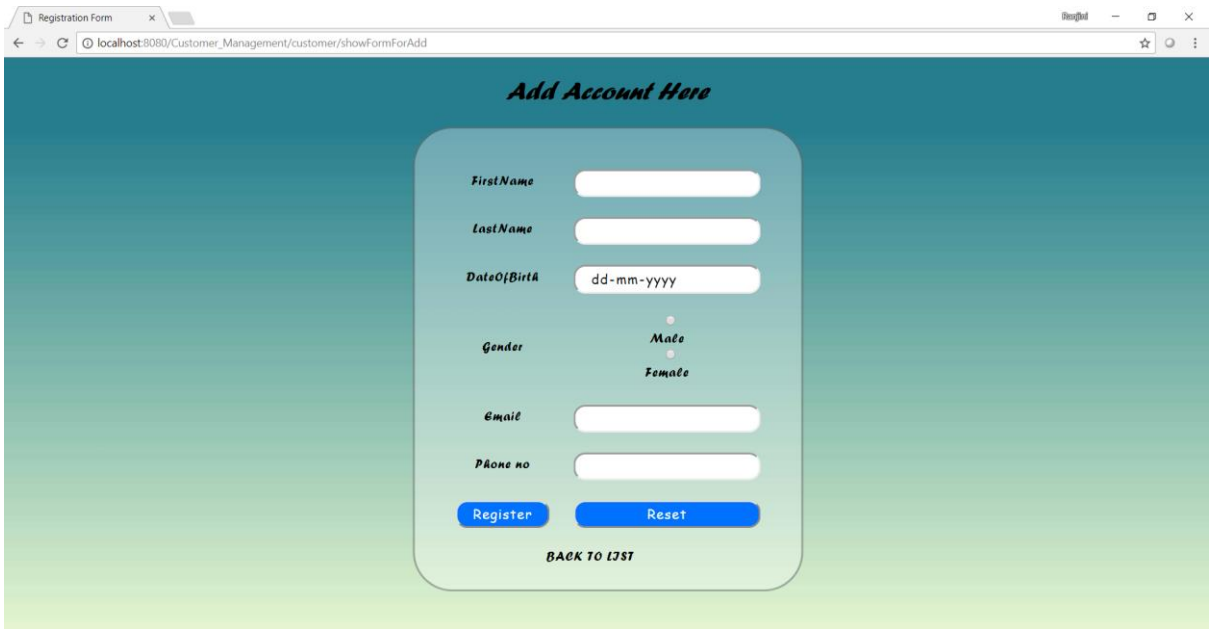
Given below are the web pages:

1. Login page

2.Customer List page



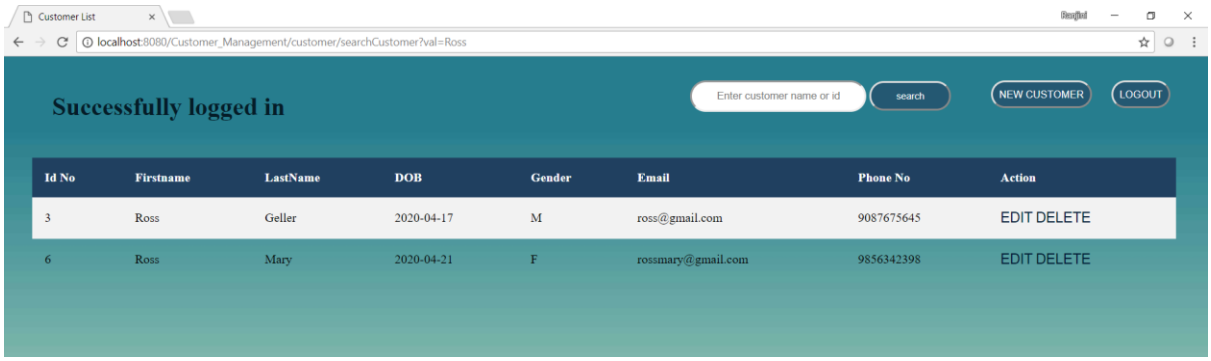
3.Add new Customer page



4.Validation page

5.Search Customer page

This is the output on searching the name “Ross”.



6.Error page

CONCLUSION AND FUTURE SCOPE

In this world of growing technologies everything has been computerized. With large number of work opportunities the Human workforce has increased. Thus there is a need of a system which can handle the data of such a large number of Customers. In existing system, customer management works on manual process which is a time taking process and data organizing is not efficient. So in order to provide more efficient application using Java ,HTML and CSS, it can be implemented as a web application which will increase the efficiency of the system and data is stored in the database so accessing data is more secure.

This project simplifies the task of maintain records because its users friendly nature. This Web Application is supported to eliminate and in some cases reduce the hardships faced by this existing system.

This application is reduced as much as possible to avoid the errors while entering the data. It also provides error messages while entering the invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user friendly.

In future, I would implement the web application as a mobile application supportable by both android and ios mobile phones. I would also try to implement some extra features in the Customer management web application like allowances, medical details etc. I would also like to modify our current UI to a more attractive UI by using angular in the future.

REFERENCES

[1] https://docs.spring.io/spring/docs/current/spring-framework-reference/

[2] https://docs.spring.io/spring/docs/current/spring-framework- reference/core.html

[3] https://docs.spring.io/spring/docs/current/spring-framework-reference/web.html

[4] https://hibernate.org/orm/documentation/5.0/

[5] https://maven.apache.org/guides/getting-started/index.html