**ORDER PROCESSING SYSTEM**

**Introduction:**

An extensive interface is provided by an order processing system to link businesses with potential clients. Scalable order processing systems that provide a pleasant user experience, maintain the integrity of orders, and safeguard customer and company data are required by the already booming e-commerce industry. Customers can use our system to make an order through a company's employees and receive a detailed quote for their order. If the Order's specifications and price meet the demands of the Customer, they can approve it, and then an Invoice will be created. The Overview section discusses further features.

**Overview:**

This Readme describes the project created by team "RuntTime Terror" for the training program's project competition "Code Fury" organized by HSBC. The two key elements in this project—the employee and the customer—serve as the order processing system.

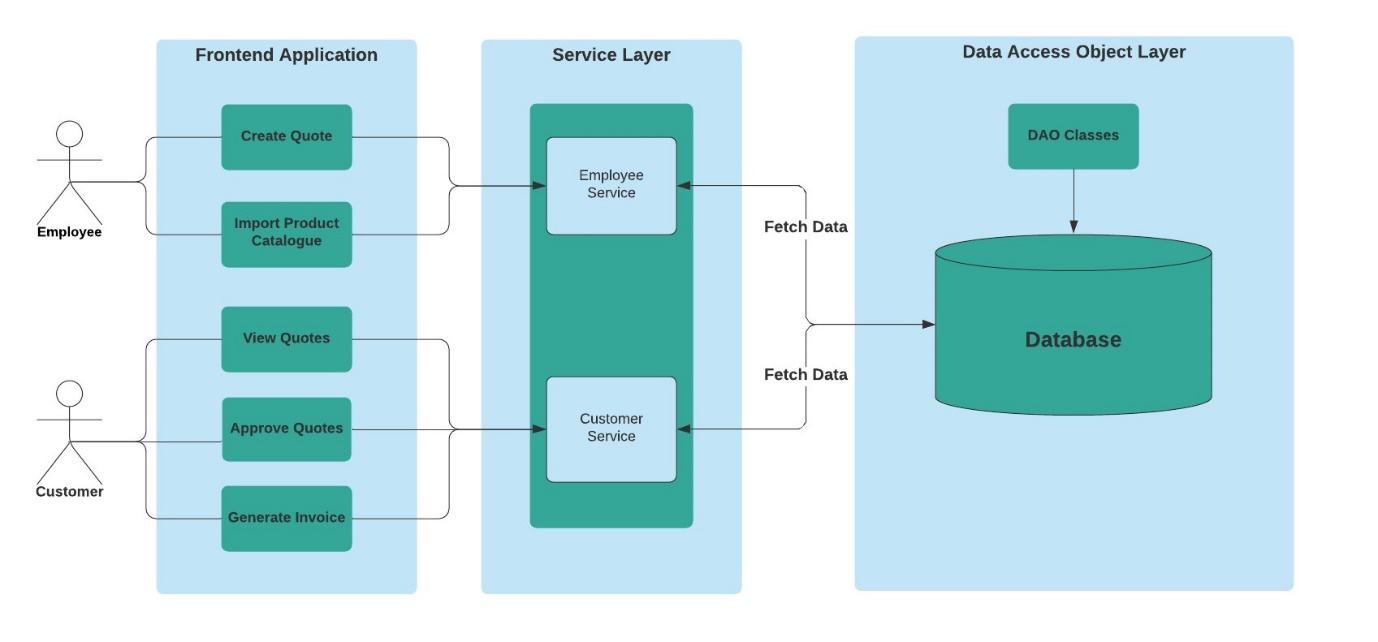
**1.The Employee's two primary functions are as follows:**

* using a customer's ID or name to identify a specific client before sending them a fresh quote with a list of products.
* adding several new products to the database by importing a JSON file of them.

**2.The following duties belong to the customer:**

* selecting and approving a certain quote sent by the employee.
* obtaining a display of invoices for orders that the customer has already approved. The invoices, however, will only be displayed if the order has been approved more than a day ago.

**High Level View:**

[](https://github.com/dhruvinamdar/TheSemicolonsProject/blob/main/docs/High%20Level%20View.jpeg)

**Technology Stack:**

Frontend: HTML, CSS, Bootstrap, JavaScript

Backend: Java, MySQL

IDE: Eclipse IDE for Enterprise Java and Web Developers - 2021-06

Data-interchange format: JSON

**Scope:**

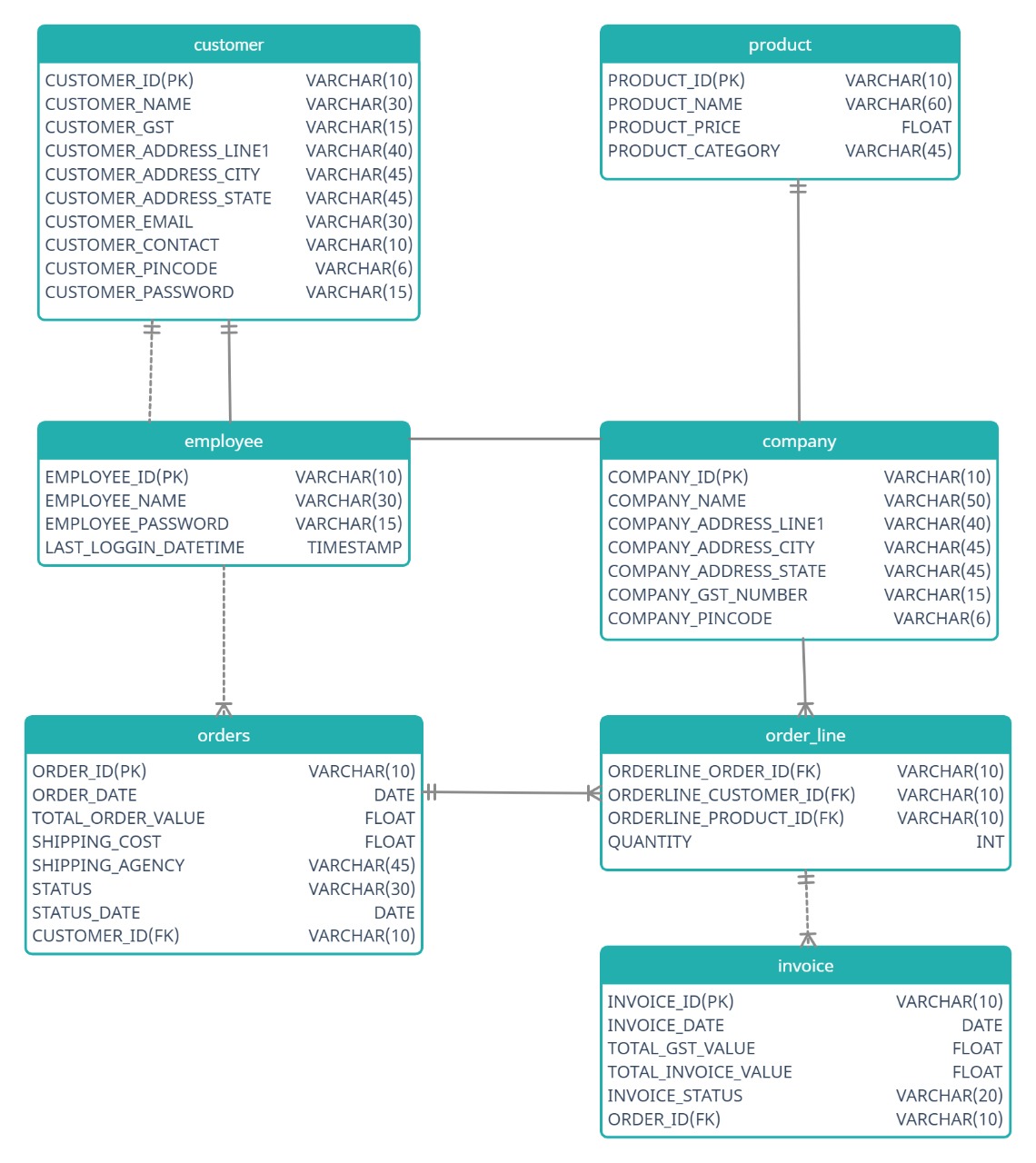
**1.Employee**

* By selecting the items that need to be used, the employee is able to create a new quote. He has the option to create a new quote after selecting a specific existing customer.
* By giving a list of products in a JSON file, the employee can easily import products into the database. Only a particular JSON object structure is accepted. The quantity of goods that have been successfully imported is communicated to the employee.
* Once the employee creates the order, overheads like the shipping cost and the GST are dynamically calculated.
* As of right now, no new employees may be established through the application.

**2.Customer**

* A list of quotes the client has obtained from the worker is available. He can then decide whether to review and approve a specific order.
* Additionally, the customer has access to a list of bills for earlier orders that were approved more than a day prior.
* He can also print his invoice as a PDF.
* The application currently does not allow for the creation of new customers.

**Database Schema:**

[](https://github.com/dhruvinamdar/TheSemicolonsProject/blob/main/docs/DBSchema.jpeg)

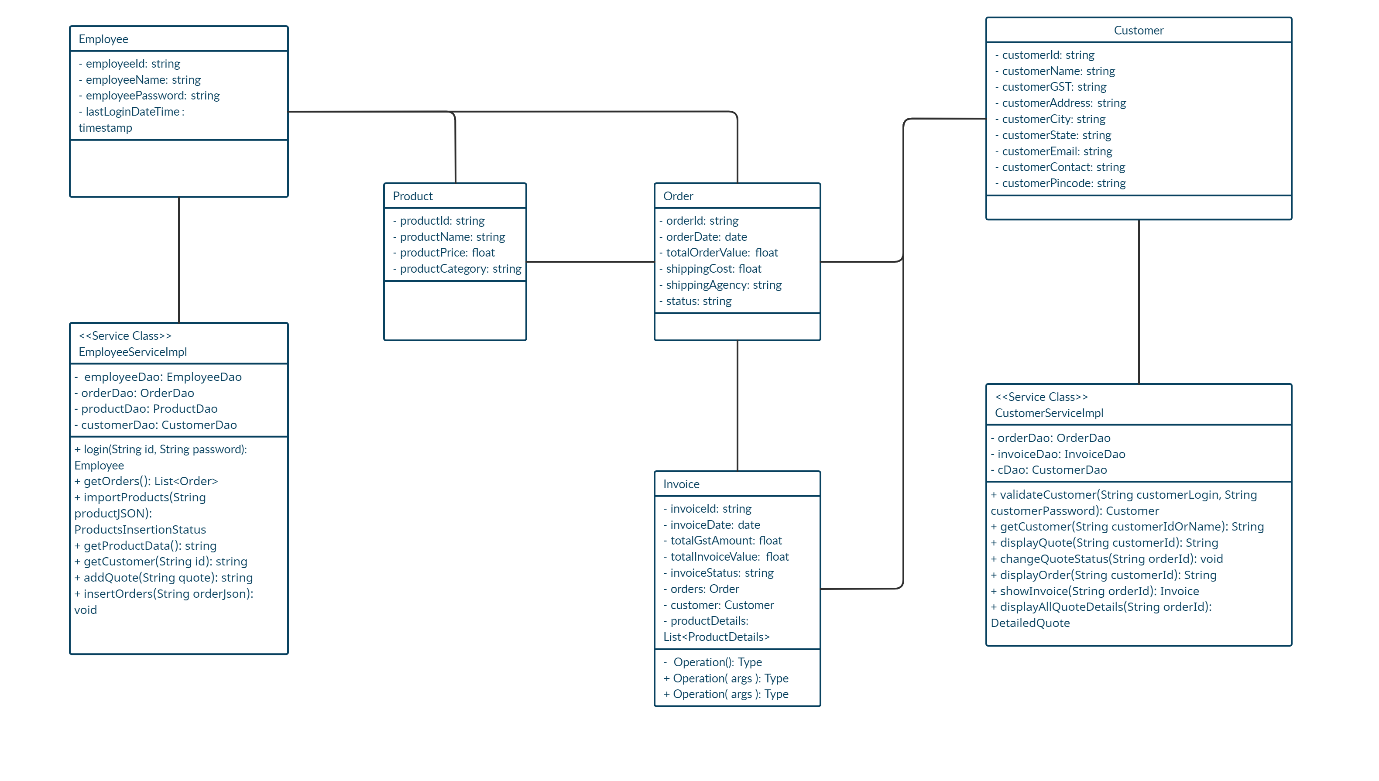
**Requirements:**

jdk 16

mysql 8.0

apache-tomcat 9.0.52

**Class Diagram:**

[](https://github.com/dhruvinamdar/TheSemicolonsProject/blob/main/docs/Class%20Diagram.jpg)

**Future Scope:**

* Give Employee rights to create a new Customer and enter details.
* Give Customer an option to select their Order on their own.
* Add third-party Payment Gateway APIs.
* Add images for the product catalogue.