



## **SECP3744-01: Enterprise System Design and Modelling**

**SEMESTER 2 2022/2023**

---

### **Software Design Description (SDD)**

TSK Online Shopping

(Customer Relationship Management (CRM) Module)

#### **Submission**

Version 1.0

#### **Due Date**

10th June 2023

#### **Prepared by**

Group Octavia

# Revision Page

## a. Overview

This is the first version of System Documentation of TSK Online Shopping (Customer Relationship Management, CRM Module). This version contains two sections which are Introduction and Requirement Specification.

## b. Target Audience

- Sales Administrator

Sales representatives are the employees of TSK Synergy who use the CRM system to manage their sales and track customers.

- Customer

Customers are the individuals or businesses that interact with the TSK Online Shopping system. They use the CRM system to track their interactions with the company, manage their account information, and make purchases.

## c. Project Team Members

Member Name	Role	Task	Status
Qaisara binti Rohzan	Leader	Manage Review	Incomplete
Nur Irdina Aliah binti Abdul Wahab	Member		Incomplete
Nurfarrahin binti Che Aliah	Member		Incomplete

## d. Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Qaisara binti Rohzan	Completed Section 1,2 and 5	10/06/2023

# Table of Contents

<b>1. Introduction</b>	<b>4</b>
1.1 Purpose	4
1.2 Scope	4
1.3 Definitions, Acronyms and Abbreviation	5
1.4 References	5
1.5 Overview	6
<b>2. Specific Requirements</b>	<b>7</b>
2.1 External Interface Requirements	7
2.1.1 User Interfaces	7
2.1.2 Hardware Interfaces	7
2.1.3 Software Interfaces	7
2.1.4 Communication Interfaces	8
2.2 System Features	9
2.2.1 UC001: Use Case Manage User	10
Activity Diagram of UC001	11
System Sequence Diagram of UC001	11
2.2.2 UC002: Use Case Manage Customer	12
Activity Diagram of UC002	12
System Sequence Diagram of UC002	12
2.2.3 UC003: Use Case Manage Review	13
Activity Diagram of UC003	13
System Sequence Diagram of UC003	13
2.3 Performance and Other Requirements	14
2.4 Design Constraints	14
2.5 Software System Attributes	14
<b>3. System Architectural Design</b>	<b>14</b>
3.1 Architecture Style and Rationale	14
3.2 Component Model	14
<b>4. Detailed Description of Components</b>	<b>15</b>
4.1 Complete Package Diagram	15
4.2 Detailed Description	15
4.2.1 P001: <Name of Package 1> Subsystem	15
<b>5. Data Design</b>	<b>18</b>
5.1 Data Description	18
5.2 Data Dictionary	18

# 1. Introduction

## 1.1 Purpose

This System Documentation (SD) provides a comprehensive overview of the system development process for a room rental system. It includes three important documents: System Requirements Specification (SRS), System Design Document (SDD), and System Testing Document (STD). The SD is a reference tool for stakeholders involved in the development process. Developers, testers, project managers, and clients can all use the SD to ensure that the system is built correctly, identify any issues, meet deadlines and budget constraints, understand the system, and provide feedback. The SD is a detailed explanation of the system's requirements, design, and testing procedures. This documentation is structured and thorough, which helps to minimise the risk of mistakes, increase productivity, and improve the project's overall outcome.

## 1.2 Scope

This System Documentation (SD) pertains to the "Customer Relation Management (CRM) Module" software product. It is designed to provide a clear understanding of the system development process, including the System Requirements Specification (SRS), System Design Document (SDD), and System Testing Document (STD). The CRM Module is a comprehensive tool that enables users to manage customer relationships, track sales, and generate reports. In addition, it allows users to manage customer support, and product inventory. This software product is exclusively available to TSK Synergy employees and customers. Its main objective is to improve customer service and sales by providing a centralised platform for managing customer data and interactions.

The scope of the "CRM Module" includes the following:

- Identify the needs of users and the system, including functionality, performance, usability, reliability, portability, and security.
- Use an agile-driven approach to ensure scalability and maintainability.
- Follow best practices in software engineering, including coding standards, testing procedures, and quality assurance processes.
- Deliver the system with a user manual and installation guide.
- Manage customer data, including contact information, purchase history, and service requests.
- Track sales and generate reports.
- Manage customer support, and product inventory.

### 1.3 Definitions, Acronyms and Abbreviation

Term	Definition
TSK Online Shopping System (CRM Module)	The software product is designed to assist admins of the sales department in viewing order sales and generating sales reports. This product also allows the salespersons to manage customer data and support.
CRM	Customer Relationship Management - A business system or strategy used to effectively manage customer interactions and relationships.
SRS	Software Requirements Specification - A comprehensive document that outlines the functional and nonfunctional requirements of a software system.
SD	System Design - Encompasses the process of defining the structure, components, modules, interfaces, and data of a system.
SDD	Software Design Document - An in-depth document that provides a detailed description of the software design. It includes comprehensive information about the software's architecture, system components, modules, data structures, algorithms, and interfaces.
STD	Software Test Document - Describes the approach to be followed during the testing phase, including the objectives, test environment, procedures, and expected outcomes.
ERD	Entity-Relationship Diagram - A graphical representation illustrating the relationships between entities (objects or concepts) within a database.
CRUD	Create, Read, Update, Delete - Represents the fundamental operations performed on data in a persistent storage system, like a database.

### 1.4 References

Specify a complete list of references using a standardised reference format.

1. Aghajani, E., Nagy, C., Linares-Vásquez, M., Moreno, L., Bavota, G., Lanza, M., & Shepherd, D. C. (2020, June). Software documentation: the practitioners' perspective. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering* (pp. 590-601).

2. Chomal, V. S., & Saini, J. R. (2014). Significance of software documentation in software development process. *International Journal of Engineering Innovations and Research*, 3(4), 410.
3. Forward, A., & Lethbridge, T. C. (2002). The relevance of software documentation, tools and technologies. *Proceedings of the 2002 ACM Symposium on Document Engineering*, 26–33. <https://doi.org/10.1145/585058.585065>

## 1.5 Overview

This System Documentation (SD) provides a comprehensive description of the customer relationship management (CRM) module of TSK Online Shopping System being developed.

The SD is organised as follow:

- **User Characteristics:**

This section focuses on identifying and describing the intended users of the CRM module in the TSK Online Shopping System. It provides information about their roles, needs, expectations, and skill levels within the system.

- **System Features:**

This section provides a detailed description of the functional and non-functional requirements specific to the CRM module in the TSK Online Shopping System. Each feature is explained in terms of its purpose, inputs, outputs, and any necessary user interactions. This includes use case descriptions, sequence diagrams, and activity diagrams.

- **Software System Attributes, Performance, and Other Requirements:**

This section focuses on the non-functional requirements that define how the CRM module in the TSK Online Shopping System behaves and performs. It covers aspects such as performance, scalability, reliability, security, and usability.

- **Design Constraints:**

This section outlines any constraints or limits that must be considered during the design and implementation of the CRM module in the TSK Online Shopping System. It encompasses factors like hardware and software dependencies, compatibility requirements, external interfaces, and regulatory constraints.

## 2. Specific Requirements

### 2.1 External Interface Requirements

#### 2.1.1 User Interfaces

The customer relationship management module of TSK Online Shopping System will be used by the sales department and customers only.

- a) The logical characteristics of each interface between the software product and its users:
  - To ensure interoperability with different screen sizes and devices, the system will use a responsive web design approach.
  - The user interface will include sensible page layouts and simple navigation menus that will allow users to effortlessly access different features and parts of the application.
  - To improve the user experience, the system will use modern and visually appealing screen formats, as well as appropriate colour schemes, typography, and graphical elements.
  - Validation and real-time feedback will be provided on user input forms to assist users in inputting accurate and valid information.
  - The interface will be multilingual, allowing users from various locations to engage with the system in their preferred language. The system will generate reports with relevant and useful content, presenting data in an ordered and easy-to-understand format.
  
- b) Optimising the interface with the person who must use the system:
  - The user interface will be built with a user-centric approach, taking user preferences, expectations, and usability requirements into account.
  - The system will display clear and succinct error messages to the user, indicating the cause of the mistake and recommending corrective measures.
  - The interface will stress simplicity and clarity above crowded layouts or complex navigation patterns that may be confusing to users. It will ensure that common tasks and actions may be completed with fewer steps and less cognitive stress.
  - Following recognised design patterns and industry best practises, the system will provide a consistent and familiar user interface. User documentation, help materials, and contextual tooltips will be supplied to help users understand and operate the system efficiently.

### 2.1.2 Hardware Interfaces

- a) The software needs 1GB RAM, 10GB disk space, and a 1GHz processor.
- b) Supported devices: Chrome, Firefox, Safari, Edge, iPhone, Android, and iPad.
- c) Protocol: TCP/IP and HTTP.

### 2.1.3 Software Interfaces

- a) Web Server
  - **XAMPP 8.2.4:** Laravel's local development environment. Apache, MariaDB, PHP, and more technologies are included. XAMPP messages often include instructions for configuring and managing virtual hosts and databases.
  - **Apache Web Server 2.4.56:** Web server software that handles HTTP queries. HTTP requests and information such as URLs, headers, and body content are included in messages sent to Apache.
- b) Framework
  - **Laravel 9.52.7:** PHP framework for developing web applications. For specific operations or routes, Laravel uses HTTP requests. Text-based messages are used to format requests, which may include parameters, form data, or JSON/XML payloads.
- c) Web-development Scripting Language
  - **PHP 8.2.6:** In web applications, server-side scripting is used. PHP messages contain server-side instructions in PHP code, such as database queries or data manipulation, that adhere to syntax and norms.
- d) Database
  - **MariaDB 10.4.28:** Responsible for database operations. MariaDB handles SQL queries in text format. Keywords, table and column names, condition clauses, and data values can all be found in these searches.

### 2.1.4 Communication Interfaces

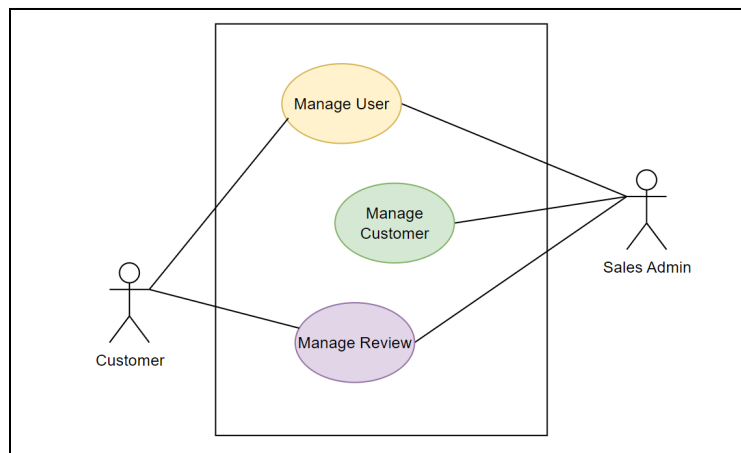
- a) **Hypertext Transfer Protocol (HTTP):** HTTP is a data transfer protocol that specifies how data is sent between a client and a server. It's used to ask for resources like web pages, photos, and files. To communicate with the browser, other web services, and other apps, Laravel employs HTTP.
- b) **Transmission Control Protocol/Internet Protocol (TCP/IP):** TCP/IP is a protocol suite that defines how devices communicate across a network. It is in charge of breaking down data into packets, routing them to the correct destination, and reassembling the original data. Laravel communicates with the database, other servers, and other devices using TCP/IP.



## 2.2 System Features

The CRM module in the TSK Online Shopping System is designed to enhance customer relationship management and improve overall business performance. It offers a range of system features to facilitate effective customer interactions and data management. This module allows customers or clients to directly interact with the customer support team, manage their profiles and even drop product reviews. As for the administrating side, this module offers the sales department personnels to track customers, sales and their purchasing preferences.

The system features in the customer module include manage user, manage customer, and manage reviews.



**Figure 2.1: Use Case Diagram for Customer Relationship Management (CRM) Module**

**Figure 2.2: Activity Diagram for Customer Relationship Management (CRM) Module**

**Figure 2.3: Domain Model for Customer Relationship Management (CRM) Module**

## 2.2.1 UC001: Use Case Manage User

**Table 2.2.1.1: Use Case Description for Manage User**

Use case: Manage User
<b>ID:</b> UC001
<b>Actors:</b> Customer
<b>Preconditions:</b> <ol style="list-style-type: none"><li>1. Customers must Sign in into the system with appropriate privileges.</li></ol>
<b>Flow of events:</b> <ol style="list-style-type: none"><li>1. Create an account<ol style="list-style-type: none"><li>a. Create an account</li><li>b. Customers create an account.</li><li>c. Customers must input information such as their email address and password.</li></ol></li><li>2. Log in<ol style="list-style-type: none"><li>a. The system displays a login page.</li><li>b. Customers submit their login information, such as their email address and password.</li><li>c. Customers have access to the website.</li></ol></li><li>3. Update user profile<ol style="list-style-type: none"><li>a. The customer arrives at the profile page.</li><li>b. The system displays consumer profile information such as name, phone number, and address.</li><li>c. The profile information was changed by the customer.</li><li>d. The modifications are validated by the system, and the user account is updated.</li></ol></li><li>4. Reset password<ol style="list-style-type: none"><li>a. Customer goes to the password reset page.</li><li>b. The new password form is displayed by the system.</li><li>c. Customers type in their new password.</li><li>d. The changes are validated by the system, and the user password is updated.</li></ol></li></ol>
<b>Postconditions:</b> <ol style="list-style-type: none"><li>1. Customers are either granted or denied access to the system.</li><li>2. Customer profile information is updated.</li></ol>
<b>Alternative flow:</b> <ol style="list-style-type: none"><li>1. If the customer chooses to cancel any action, the system returns to the previous page without making any changes.</li><li>2. If there are no customer accounts available in the system, the system displays a message indicating the absence of user accounts and provides options to sign up for a new account.</li></ol>
<b>Postconditions:</b> <ol style="list-style-type: none"><li>1. Customers are granted to browse the website and purchase orders.</li></ol>
<b>Exception flow:</b> <ol style="list-style-type: none"><li>1. If the customer enters invalid information during signup, sign in or update, the system displays an error message and prompts the actor to correct the invalid fields.</li></ol>

**Activity Diagram of UC001**

**System Sequence Diagram of UC001**

### 2.2.2 UC002: Use Case Manage Customer

*Table 2.2.2.1: Use Case Description for Manage Customer*

Use case: Manage Customer
<b>ID:</b> UC002
<b>Actors:</b> Sales Admin
<b>Preconditions:</b> <ol style="list-style-type: none"><li>1. Sales admin must Sign in into the system with appropriate privileges</li><li>2. The customer must exist in the system.</li></ol>
<b>Flow of events:</b> <ol style="list-style-type: none"><li>1. The sales admin selects the customer to manage.</li><li>2. The sales admin views the customer's details.</li><li>3. The sales admin can update the customer's details.</li><li>4. The sales admin can add or remove the customer from a mailing list.</li><li>5. The sales admin can view the customer's purchase history.</li></ol>
<b>Postconditions:</b> <ol style="list-style-type: none"><li>1. The customer's details are updated.</li><li>2. The customer is added or removed from a mailing list.</li><li>3. The customer's purchase history is updated</li></ol>
<b>Alternative flow:</b>
<b>Postconditions:.</b>
<b>Exception flow:</b> <p>If the sales admin does not have permission to manage customers, an error message is displayed.</p>

#### Activity Diagram of UC002

#### System Sequence Diagram of UC002

### 2.2.3 UC003: Use Case Manage Review

*Table 2.2.2.1: Use Case Description for Manage Review*

Use case: Manage Review
<b>ID:</b> UC003
<b>Actors:</b> Customers and Sales Admin
<b>Preconditions:</b> <ul style="list-style-type: none"><li>1. Customer<ul style="list-style-type: none"><li>a. The customer is logged in to the system</li><li>b. The customer has an arrived order and can create product reviews and ratings</li><li>c. The customer is browsing products on the system</li></ul></li><li>2. Sales Admin<ul style="list-style-type: none"><li>a. The admin is logged in to the system.</li><li>b. The product review exists in the system.</li></ul></li></ul>
<b>Flow of events:</b> <ul style="list-style-type: none"><li>1. Browsing products<ul style="list-style-type: none"><li>a. The customer selects a product to view when browsing the system</li><li>b. The customer can read past reviews on the product page</li></ul></li><li>2. An order arrived<ul style="list-style-type: none"><li>a. The customer can select their purchased product then give ratings and reviews on the item.</li></ul></li></ul>
<b>Postconditions:</b> <p>The customer can read and submit product ratings and reviews.</p>
<b>Alternative flow:</b> <p>If the customer does not have an arrived order, the customer can only view ratings and reviews. If there are no reviews on the product in the system, the user would not be able to see any reviews and ratings.</p>
<b>Exception flow:</b> <p>If the customer's review is not submitted and saved, it will not be displayed on the product page.</p>

#### Activity Diagram of UC003

#### System Sequence Diagram of UC003

## 2.3 Performance and Other Requirements

The TSK Online Shopping website characteristics serve as the foundation for the programme, and they include the following:

- **Functionality:** The system must be able to satisfy the demands of its users. This involves viewing and adding product reviews, managing customer lists and tracking customer sales and preferences.
- **Reliability:** The system must perform its operations accurately and consistently in order to prevent crashing or encountering faults on a regular basis.
- **Usability:** The system should have an easy-to-use interface and be simple to understand.
- **Maintainability:** Requires a well-documented system with a modular architecture.

Non-functional needs that do not fall into the software system characteristics or performance categories are referred to as other requirements. These prerequisites are as follows:

- **Security:** The system must guard against unauthorised access, alteration, or destruction of its data. This includes the safeguarding of user data, such as personal information.
- **Response time:** The website should load quickly and respond quickly to user activity, even during peak traffic hours.
- **Compliance:** All applicable rules and regulations regulating data privacy and security must be followed by the system.
- **Safety:** The capacity of the system to operate safely without endangering people or the environment.
- **Legal and regulatory:** Compliance entails adhering to laws, rules, and standards.
- **Compatibility:** The website is compatible with a variety of platforms, including web browsers, mobile devices, and desktop computers.

## 2.4 Design Constraints

1. **Scalability:** The system should be built to accommodate an increasing number of users, items, and transactions without sacrificing performance or user experience. This may be accomplished by efficient database architecture, caching technologies, and horizontal scalability strategies.
2. **Protection:** The system must prioritise the protection of user data, such as personal information, payment information, and order history. It should have strong authentication and authorisation systems, encryption for sensitive data, and safeguards against typical security flaws like SQL injection and cross-site scripting.
3. **Availability:** The system should be highly accessible, allowing users to use the platform at any time without substantial downtime. This may be accomplished through the use of redundant server configurations, load balancing, and fault-tolerant architecture.
4. **Performance:** The system's reaction time and latency should be optimised. Better speed may be achieved by the use of efficient algorithms, caching methods, and good database indexing. Continuous monitoring and performance testing are required to detect and fix bottlenecks.

## 2.5 Software System Attributes

1. **User-Friendly Interface:** To deliver a seamless purchasing experience, the system should have an intuitive and user-friendly interface. It should be simple to navigate, search for products, examine product information, add things to the shopping basket, and check out. A well-designed user interface/user experience is critical for attracting and retaining clients.
2. **Mobile Compatibility:** Because mobile devices are increasingly being used for online purchasing, the system should be created using a flexible design approach. It should be able to adjust to multiple screen sizes and orientations, ensuring a consistent user experience across devices.
3. **Payment Integration:** The system should allow safe and simple payment methods such as credit/debit cards, digital wallets, and online banking.

### 3. System Architectural Design

*This section of the SD serves as part of SDD that should describe the architectural style and the rationale or justification of your selection. The component and subsystem diagram should also be included.*

#### 3.1 Architecture Style and Rationale

*State your chosen architectural style and the rationale of choosing that particular style.*

State here

#### 3.2 Component Model

*Develop a component model and explain the relationships between the components to achieve the complete functionality of the system. This is a high level overview of how responsibilities of the system were partitioned and then assigned to subsystems. Identify each high level subsystem and the roles or responsibilities assigned to it. Describe how these subsystems collaborate with each other in order to achieve the desired functionality. Do not go into too much detail about the individual subsystem. The main purpose is to gain a general understanding of how and why the system was decomposed, and how the individual parts work together. Provide a diagram showing the major subsystems and data repositories and their interconnections. Describe the diagram clearly.*

**Figure 3.1: Component Diagram of <Name of the System>**



## 4. Detailed Description of Components

*This section of SD serves as part of SDD that describes each module or subsystem in the project.*

### 4.1 Complete Package Diagram

*Include the overall package diagram of your system here. [Example package diagram for a Sale System adapted from Satzinger (2011)] Indicate the navigation visibility based on the dependency among classes in the design class diagram. If the diagram is too cluttered, simplify the classes by showing the class name only without showing the attributes and methods. The details can be shown in the following class diagram sub-section for respective subsystem/package.*

**Figure 4.1: Package Diagram for <Name of the System>**

### 4.2 Detailed Description

*For each subsystem/package there must be ONE class diagram and several sequence diagrams based on how many use cases you have in the subsystem/package. Use branching in sequence diagram to combine alternate flow in the same sequence diagram. If the diagram is cluttered, consider a new sequence diagram for respective scenario/alternate flow. The given example includes view, domain and data access layer in respective packages. Organise subsystem/package according to the chosen architectural style. Note that if you choose model-view-controller (MVC) or other architectural styles, then the packages should follow the selected styles. However, for the scope of this course, you may follow the example.*

#### 4.2.1 P001: <Name of Package 1> Subsystem

**Figure 4.2: Package Diagram for <Name of Package 1> Subsystem**

##### 4.2.1.1 Class Diagram

*Include class diagram to represent all classes in the respective subsystem/package. Include the controller classes.*

**Figure 4.3: Class Diagram for <Name of Package 1> Subsystem**

*List all methods in a table for respective entity (add the table accordingly), then write its algorithm. Example of algorithm as shown below.*

*Step1: Start*

*Step2: Read/input A and B*

Step3: If A greater than B then C=A

Step4: If B greater than A then C=B

Step5: Print C

Step6: End

<b>Entity Name</b>	e.g. Order
<b>Method Name</b>	e.g. createOrder
<b>Input</b>	
<b>Output</b>	
<b>Algorithm</b>	1. Start 2. ... 3. End

#### 4.2.1.2 Sequence Diagram

*Include sequence diagram for each respective use case in your package. In these examples only sequence diagram Create New Phone Order Scenario and Cancel an Order Scenario are shown. Include the final sequence diagram that comprises view layer, controller, and its problem domain (entity) and data access layer. Provide a unique code for each scenario of sequence diagram to be used in Section 7: Requirements Matrix. If you have only one scenario, then you only need one sequence diagram.*

a) SD001: Sequence diagram for Create New Phone Order

**Figure 4.4: Sequence Diagram for <Create New Phone Order Scenario>**

b) SD002: Sequence diagram for Create Cancel an Order Scenario

**Figure 4.5: Sequence Diagram for <Cancel an Order scenario>**

#### 4.2.2 P002: <Name of Package 2> Subsystem

##### 4.2.2.1 Class Diagram

##### 4.2.2.2 Sequence Diagram

## 5. Data Design

### 5.1 Data Description

The major data or systems entities are stored into a relational database named as db\_eis, processed and organised into 5 entities as listed in Table 5.1.

**Table 5.1: Description of Entities in the Database**

No.	Entity Name	Description
1.	<b>User</b> tb_user	The table stores information about the users of the TSK Online Shopping System (Customer/Sales Admin/Logistic Admin/Purchase Admin/ Store Department Manager)
2.	<b>Review</b> tb_review	The table stores information about the ratings and reviews given by TSK Online Shopping System's customers.
3.	<b>Customer</b> tb_customer	The table stores information about the customer's personal information and purchasing preferences

### 5.2 Data Dictionary

#### 5.2.1 Entity: User

Attribute Name	Description	Data Type & Length	Null	Multi-Valued
u_id	Uniquely identifies a user	Integer	No	No
u_email	The user's email address	255 variable character	No	No
u_name	Name of user	255 variable character	No	No
u_contact	Contact number of users	15 variable character	No	No
u_pwd	Password for the user's account	255 variable character	No	No
u_type	The user type	10 variable character	No	No

#### 5.2.2 Entity: Review

Attribute Name	Description	Data Type & Length	Null	Multi-Valued
rv_id	Uniquely identifies a review	Integer	No	No
rv_orderID	Uniquely identifies an order	Integer	No	No
rv_productID	Uniquely identifies a product	Integer	No	No
rv_rating	Customer's rating of the product	Integer	No	No
rv_message	Customer's review of the product	255 variable character	No	No

### 5.3.3 Entity: Customer

Attribute Name	Description	Data Type & Length	Null	Multi-Valued
c_id	Uniquely identifies a customer	Integer	No	No
c_name	Name of the customer	100 variable character	No	No
c_address	Address of customer	100 variable character	No	No
c_state	Customer address's state	50 variable character	No	No
c_postalCode	Customer address's postal code	20 variable character	No	No
c_contact	Contact number of the customer	20 variable character	No	No