

Customer Relationship Management (CRM) System

Practical - 2

Software Engineering Lab

Class UML Diagram

- By Aman Joharapurkar (A2-32)

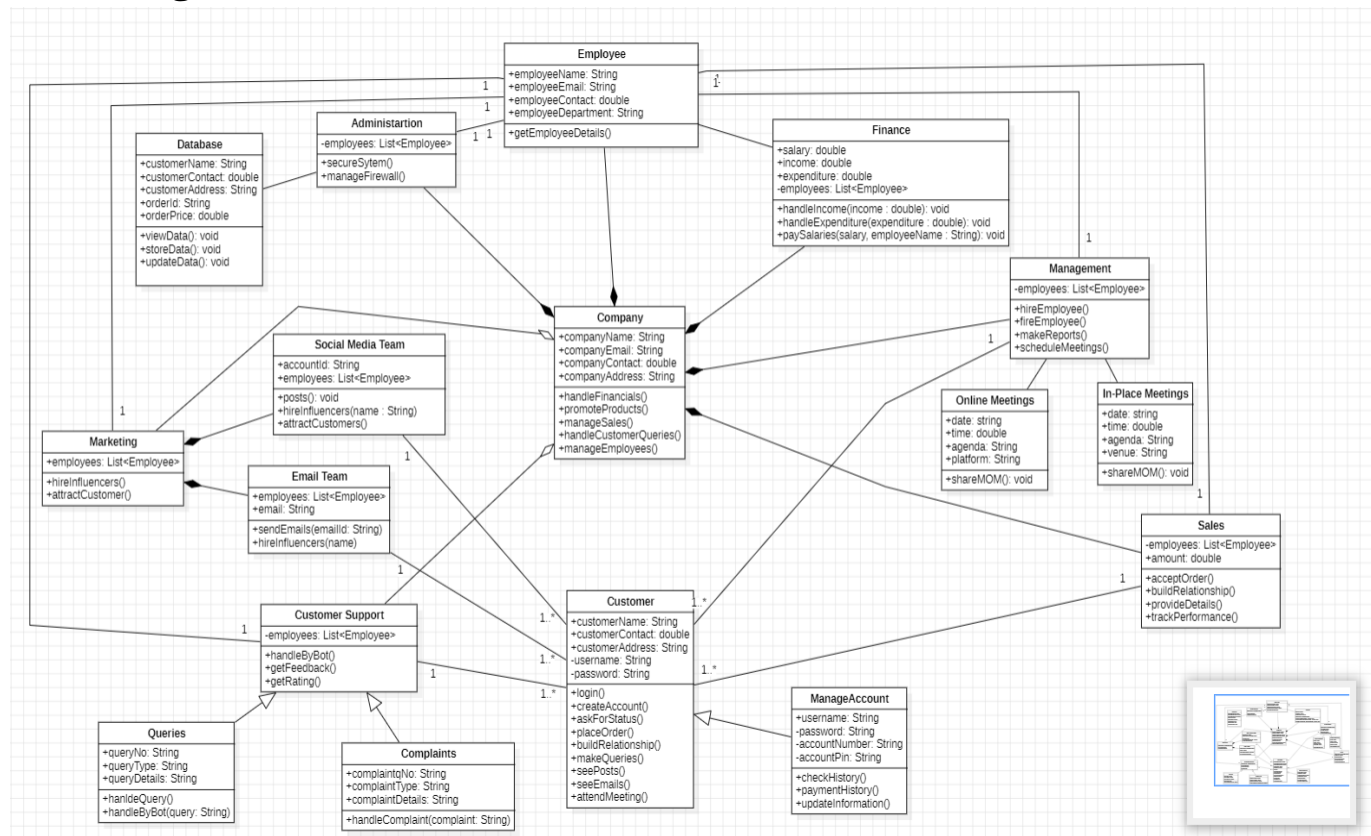
Class Diagrams:

Class diagrams are a type of UML (Unified Modeling Language) diagram used in software engineering to visually represent the structure and relationships of classes within a system i.e. used to construct and visualize object-oriented systems.

Notations:

- **Class Name:**
- **Attributes**
- **Methods**
- **Visibility Notation**

Class Diagram:



- **Company Class:**

All the departments of the company come under this.

Attributes of this class are:

- ➔ Company Name: String
- ➔ Company Email: String
- ➔ Company Contact: String
- ➔ Company Address: String

Methods of this class are:

- ➔ handleFinancials()
- ➔ promoteProducts()
- ➔ manageSales()
- ➔ handleCustomerQueries()
- ➔ manageEmployees()

It has a **Composition** Relationship with departments like Finance, Administration, and Management. This is because if any of the above departments gets closed, the company will not be able to work.

It has an **Aggregation** Relationship with Customer Support; Marketing
This is because if any of the above departments gets closed the company still will be able to operate.

- **Marketing**

Attributes of this class are:

- ➔ List of Employee: List<Employee>

Methods of this class are:

- ➔ hireInfluencers()
- ➔ attractCustomers()

Marketing consists of 2 teams i.e. Social Media Team and the Email Team. Therefore, we have 2 Child classes of the same name.

Here we have a **Composition** Relationship with the Marketing Class, as if a Marketing Team is not formed then sub-teams **cannot** be formed.

1. Social Media:

Attributes of this class are:

- Account Id: String
- List of Employee: List<Employee>

Methods of this class are:

- hireInfluencers()
- attractCustomers()
- posts ()

2. Email:

Attributes of this class are:

- Email Id: String
- List of Employee: List<Employee>

Methods of this class are:

- hireInfluencers()
- sendEmails()

Association with the Employee class as Employee work in this department. One Employee can work in only one department so it is a **One-to-One Relationship**.

- **Finance:**

Attributes of this class are:

- ➔ Salary: double
- ➔ Expenditure: double
- ➔ Income: double
- ➔ List of Employee: List<Employee>

Methods of this class are:

- ➔ handleIncome()
- ➔ promoteExpenditures()
- ➔ paySalaries()

This class has a **Composition** Relationship with the Company Class & **Association** with the Employee class as Employees work in this department. One Employee can work in only one department so it is a **One-to-One Relationship**.

- **Sales:**

Attributes of this class are:

- ➔ List of Employee: List<Employee>
- ➔ Amount:double

Methods of this class are:

- ➔ acceptOrder()
- ➔ buildRelationship()
- ➔ provideDetails()
- ➔ trackPerformance()

This class has an **Association** with the Customer Class. The customer starts building a relationship with the Company which starts with the Sales department. (Many to One Relationship)

This class has a **Composition** Relationship with the Company Class & **Association** with the Employee class as the Employee works in this department. One Employee can work in only one department so it is a **One-to-One Relationship**.

- **Customer Support:**

Attributes of this class is:

- ➔ List of Employee: List<Employee>

Methods of this class are:

- ➔ getFeedback()
- ➔ getRating()
- ➔ handleByBot()

This class has an **Aggregation** Relationship with the Company Class & **Association** with the Employee class as Employees work in this department. One Employee can work in only one department so it is a **One-to-One Relationship**.

This class has 2 Child classes **Queries & Complaints**

1. **Queries:**

Attributes of this class are:

- Query No: String
- Query Type: String
- Query Details: String

Methods of this class are:

- handleQuery ()
- handleByBot()

2. **Complaints:**

Attributes of this class are:

- Complaint No: String
- Complaint Type: String
- Complaint Details: String

Methods of this class are:

- handleComplaint()

• **Management:**

Attributes of this class is:

➔ List of Employee: List<Employee>

Methods of this class are:

- ➔ hireEmployee()
- ➔ fireEmployee()
- ➔ makeReports()
- ➔ scheduleMeetings()

This class has a **Composition** Relationship with the Company Class & **Association** with the Employee class as Employees work in this department. One Employee can work in only one department so it is a **One-to-One Relationship**.

This class has an **Association** with 2 classes **Online Meetings & In-Place Meetings**

1. Online Meetings:

Attributes of this class are:

- date: String
- time: String
- agenda: String
- platform: String

Methods of this class are:

- shareMOM()

2. In-Place Meetings:

Attributes of this class are:

- date: String
- time: String
- agenda: String
- venue: String

Methods of this class are:

- shareMOM()

- **Administration:**

Attributes of this class is:

➔ List of Employee: List<Employee>

Methods of this class are:

➔ secureSystem()

➔ manageFirewall()

This class has a **Composition** Relationship with the Company Class & **Association** with the Employee class as Employees work in this department. One Employee can work in only one department so it is a **One-to-One Relationship**.

This class has an **Association** with Database class.

1. Database:

Attributes of this class are:

- Customer Name: String
- Customer Contact: String
- Customer Address: String
- Order Id: String
- Order Price: double

Methods of this class are:

- viewData()
- storeData()
- updateData()
- deleteData()

- **Employee:**

Attributes of this class are:

- ➔ Employee Name: String
- ➔ Employee Email: String
- ➔ Employee Contact: String
- ➔ Employee Department: String

Methods of this class are:

- ➔ getEmployeeDetails()

This class has an **Association** with all the Department Classes (One-to-one Relationship).

This class has a **Composition** Relationship with the Company Class.

- **Customer:**

Attributes of this class is:

- ➔ Customer Name: String
- ➔ Customer Email: String
- ➔ Customer Contact: String
- ➔ Customer Address: String
- ➔ Customer username: String
- ➔ Customer Password: String

Methods of this class are:

- ➔ login ()
- ➔ createAccount()
- ➔ askForStatus()
- ➔ placeOrder()
- ➔ buildRelationship()
- ➔ makeQueries()
- ➔ seePosts()
- ➔ seeEmails()
- ➔ attendMeetings()

This class has an **Association** with the Sales, Customer Support and Management Department. (Many to One Relationship)

This class has a Child Manage Account class.

- 1. Manage Account:**

Attributes of this class are:

- Username: String
- Password: String
- Account Number: String
- Account Pin: String

Methods of this class are:

- checkHistory()
- paymentHistory()
- updateInformation()