"Food Delivery Application": `
 Use Case Diagram - Food Delivery Application

```
@startuml
title Use Case Diagram for Food Delivery Application

' Define actors
actor "Customer" as Customer
actor "Restaurant" as Restaurant
actor "Delivery Agent" as Agent
actor "Admin" as Admin

' System boundary
rectangle "Food Delivery Application" as FDA {
    usecase "Register / Login" as UC1
    usecase "Browse Restaurants & Menus" as UC2
    usecase "Place Order" as UC3
    usecase "Track Order" as UC4
    usecase "Make Payment" as UC5
    usecase "Provide Rating / Feedback" as UC6
    usecase "Manage Menu Items" as UC7
    usecase "Accept / Reject Order" as UC8
    usecase "Deliver Order" as UC9
    usecase "Manage Users & Reports" as UC10
}

' Relationships
Customer --> UC1
Customer --> UC2
Customer --> UC3
Customer --> UC4
Customer --> UC5
Customer --> UC6

Restaurant --> UC7
Restaurant --> UC8
UC3 --> UC5 : <<include>>
UC3 --> UC4 : <<extend>>

Agent --> UC9
UC8 --> UC9 : <<include>>

Admin --> UC10

@enduml
```

Class Diagram – Food Delivery Application

```
@startuml
title Class Diagram for Food Delivery Application

' Define main classes
class User {
  +userID: int
  +name: string
  +email: string
  +password: string
  +login()
  +logout()
}

class Customer {
  +browseRestaurants()
  +placeOrder()
  +makePayment()
  +trackOrder()
  +giveFeedback()
}

class Restaurant {
  +restaurantID: int
  +restaurantName: string
  +menuItems: List<MenuItem>
  +acceptOrder()
  +updateMenu()
}

class MenuItem {
  +itemID: int
  +name: string
  +price: float
  +availability: boolean
}

class Order {
  +orderID: int
  +orderDate: date
  +status: string
  +calculateTotal()
}

class Payment {
  +paymentID: int
  +amount: float
```

```
    +paymentType: string
    +processPayment()
}

class DeliveryAgent {
    +agentID: int
    +name: string
    +phone: string
    +updateDeliveryStatus()
}

class Admin {
    +generateReports()
    +manageUsers()
    +manageRestaurants()
}

' Relationships
User <|-- Customer
User <|-- Restaurant
User <|-- DeliveryAgent
User <|-- Admin

Customer "1" -- "many" Order : places >
Restaurant "1" -- "many" Order : receives >
Order "1" -- "1" Payment : has >
Order "1" -- "many" MenuItem : contains >
DeliveryAgent "1" -- "many" Order : delivers >

@enduml

`,
```