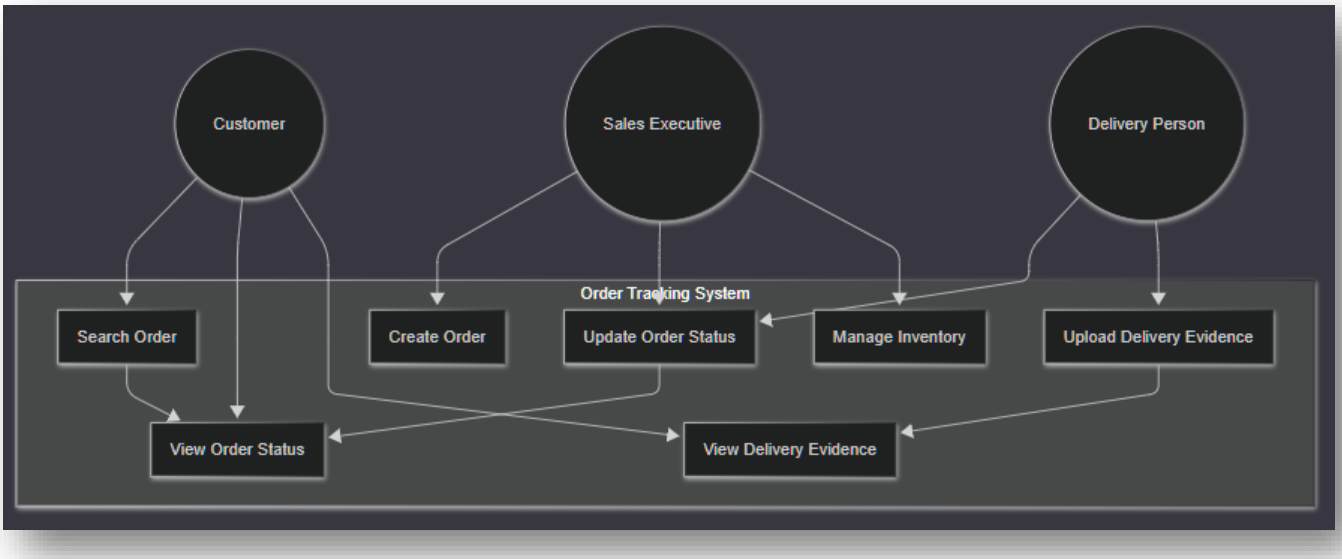
Nombre: Yaacob Jayim García Blancas	**Matrícula**: 2997543
Nombre del curso: Diseño de aplicaciones web	**Nombre del profesor**: Alan Ricardo Sánchez Maya
Módulo: 1	**Actividad**: Evidence 2
Fecha: 29/03/25	
Bibliografía:	

Class diagram

Use case diagram

The system is based on order management and order tracking until delivery. There are several types of users with different roles, such as administrators, sales, purchasing, warehouse and delivery. Each user has a specific role that defines what actions he/she can perform within the system.



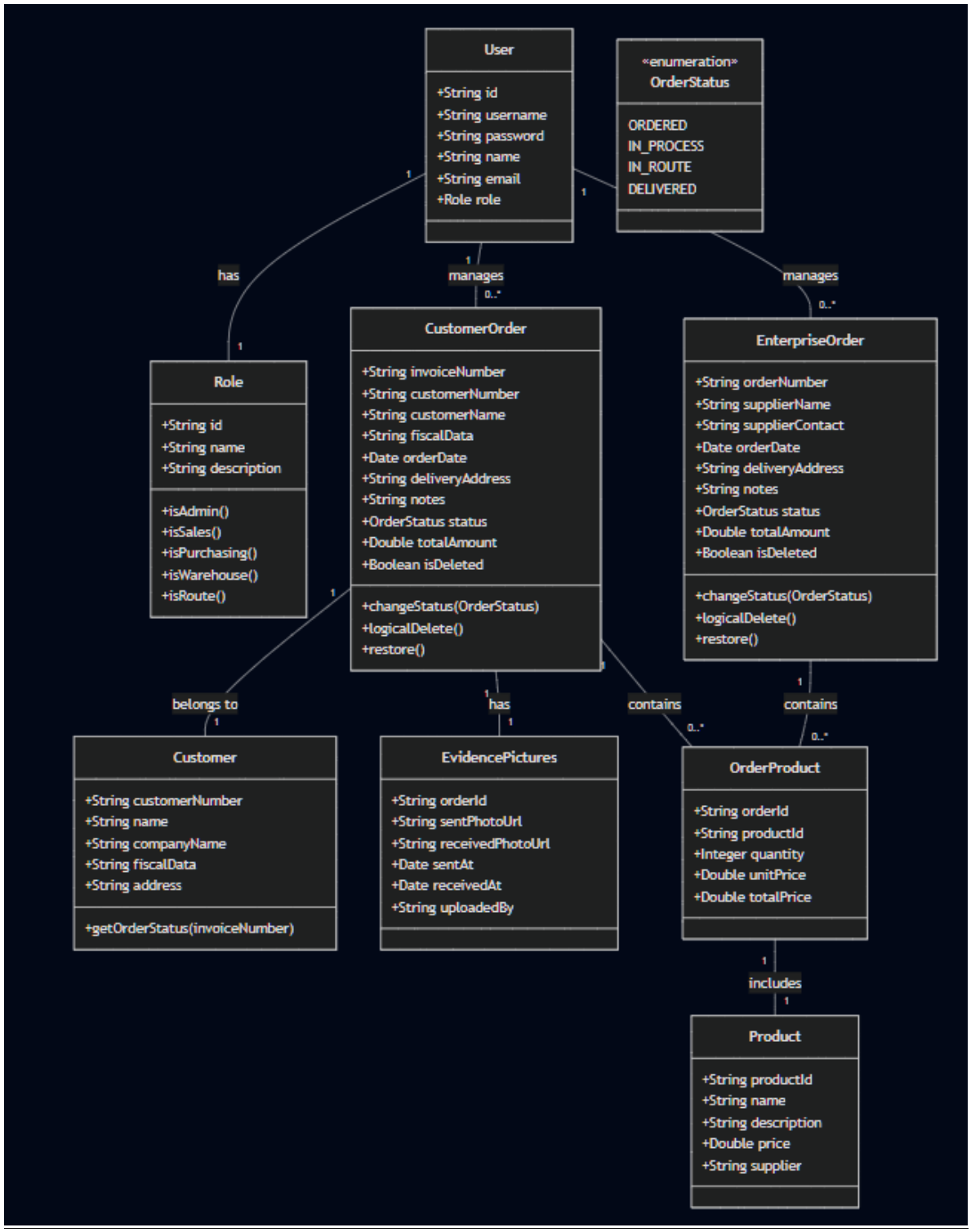
Customers have their own registered information, including fiscal data and addresses. Each order is associated with a customer and contains details such as invoice number, delivery address, notes and order status (ordered, in process, en route or delivered).

The system can also capture evidence of delivery in the form of photos, which include details such as the date of upload and who recorded it. In addition, a user can manage multiple

orders and change their status according to their progress. If an order is deleted, it can be restored if necessary.

THE DOCUMENT HAVING THE TWO DIAGRAMS CAN BE FOUND IN THE PUBLIC FOLDER IN THE GITHUB PROJECT, FilesForEvidences

https://github.com/yaacob117/Construcion-app-WebAppsClass/tree/main/public


User
+String id
+String username
+String password
+String name
+String email
+Role role
«enumeration»
OrderStatus
ORDERED
IN_PROCESS
IN_ROUTE
DELIVERED
has
manages
manages
Role
+String id
+String name
+String description
+isAdmin()
+isSales()
+isPurchasing()
+isWarehouse()
+isRoute()
CustomerOrder
+String invoiceNumber
+String customerNumber
+String customerName
+String fiscalData
+Date orderDate
+String deliveryAddress
+String notes
+OrderStatus status
+Double totalAmount
+Boolean isDeleted
+changeStatus(OrderStatus)
+logicalDelete()
+restore()
EnterpriseOrder
+String orderNumber
+String supplierName
+String supplierContact
+Date orderDate
+String deliveryAddress
+String notes
+OrderStatus status
+Double totalAmount
+Boolean isDeleted
+changeStatus(OrderStatus)
+logicalDelete()
+restore()
belongs to
has
contains
contains
Customer
+String customerNumber
+String name
+String companyName
+String fiscalData
+String address
+getOrderStatus(invoiceNumber)
EvidencePictures
+String orderId
+String sentPhotoUrl
+String receivedPhotoUrl
+Date sentAt
+Date receivedAt
+String uploadedBy
OrderProduct
+String orderId
+String productId
+Integer quantity
+Double unitPrice
+Double totalPrice
includes
Product
+String productId
+String name
+String description
+Double price
+String supplier

EVIDENCE 2:

Added all of the crud functionalities to every single one of the 8 tables and 1 enum, that because after further investigation an enum was the most suitable option for adding static values such as statuses.

Order Products

Id	Order	Product	Quantity	Unit Price	Total Price	Actions
1	Customer Order: INV-001	Product A	5	500	2500	Show Edit Delete
2	Customer Order: INV-001	Product B	3	750	2250	Show Edit Delete
3	Customer Order: INV-002	Product A	2	500	1000	Show Edit Delete

Create New Order Product

User List

Create User

ID	Username	Name	Email	Role	Actions
1	admin	Administrator	admin@example.com	Administrator	View Edit Delete
2	sales_user	Sales User	sales@example.com	Sales	View Edit Delete
3	purchasing_user	Purchasing User	purchasing@example.com	Purchasing	View Edit Delete
4	warehouse_user	Warehouse User	warehouse@example.com	Warehouse	View Edit Delete
5	routes_user	Routes User	routes@example.com	Routes	View Edit Delete

---

Every single one of the functionalities have been tested