

HiLCoE School of Computer Science and Technology

CS341 Database Management Systems

Course Project

Section: DRBSE2302C

Title: Package Delivery Management system

Group Members:

1. Ephrem Musse-----AC7795
2. Ezana Tessfaye-----JS1613
3. Mikias Yimenu-----XA4798
4. Mohammed Yasin-----KF9005
5. Yeab Tsegaye-----OR6590

14/09/24

I. Introduction

Traditional delivery management systems often lack the real-time tracking and communication tools necessary to meet the expectations of today's consumers. Customers expect to know the exact status of their orders, while businesses require precise data on delivery routes, times, and performance. Without a comprehensive tool, managing this level of detail manually can lead to inefficiencies, lost packages, and dissatisfied customers. In addition to addressing customer needs, businesses have faced challenges optimizing their delivery operations. Factors such as inefficient routing, lack of delivery personnel coordination, and poor communication between teams have all contributed to higher costs and slower deliveries.

Due to the lack of a proper and well implemented mailing system in our country the alternative means of delivering packages in the city is using people with motorcycles. Although this method avoids the extra work of going to the postal office and filling out forms, there is no guarantee of the package safety nor will there be any form of accountability after misfortunes. The only contract between the customer and the delivery person is a verbal agreement. This delivery method also is limited to documents and small objects due to the carrying capacity of the vehicle.

As our country is getting more digitalized and catching up to the developed countries, looking for a solution that keeps up with the theme of change became the foundation for Fast Package Delivery Company (FPDC). FPDC connects customers with individual contractors in a well-organized and secure manner to provide fast and hassle-free delivery. It takes inspiration from the way digital apps like uber and ride revolutionized the transportation sector by providing accountability and flexibility to the customers and the vehicle owners.

I.I Scope of the Project

The project's primary focus is the development of a database system that will enable the FPDC to function with full capability. This database system will include a reliable package management system. The system will enable the independent contractors to pick and work on contractors.

This project will not encompass package tracking system, Transaction dispute handling, Counter-Offer and haggling system

I.II Objective of the project

I.II.I General Objective

The primary objective of this project is to design and implement a functional system for the FPDC, that will enhance its capabilities enough to solve the common flaws of the current delivery sector in our country.

I.II.II Specific Objectives

- Develop a system for managing clients and Contractor information with efficiency.
- Build a system that tracks the transaction related with each package with adequate information.

I.III Design methods and Development tools

After identifying the Entities and relationship, an E-R Diagram will be constructed after which it will be mapped to a relational model. The Physical model will be based on the relational model after being normalized.

The tools that will be used to design this system are Microsoft visio and LucidChart for the E-R Diagram, Microsoft Sql management studio 20 for the database and Microsoft Word for the Documentation.

II. Database Design

II.1 Conceptual Design

Entities:

- Customer: entity that represents individuals who order the package delivery.
- Individual Contractor: entity that represents individuals that deliver the package.
- Package: entity that describes the object that is to be delivered.
- Transaction: entity that describes the process that links the customer, individual contractor and package.

Attributes:

Customer:

- CustomerID (Customer Identification)
- CustomerFirstName (Customer first name),
CustomerMiddleName (Customer middle name),
CustomerLastName (Customer last name)
- CustomerDOB (Customer Date of Birth)
- CustomerPhoneNum (Customer phone number)
- CustomerEmail (Customer Email address)
- CustomerPasswordHash(Customer Password
post encryption)

Individual Contractor:

- IndividualContractorID (Individual Contractor Identification)
- IndividualContractorFirstName (Individual Contractor first name),
IndividualContractorMiddleName (Individual Contractor middle name),
IndividualContractorLastName (Individual Contractor last name)
- IndividualContractorDOB (Individual Contractor Date of Birth)
- IndividualContractorPhoneNum (Individual Contractor phone number)
- IndividualContractorEmail (Individual Contractor Email address)
- IndividualContractorPasswordHash (Individual Contractor Password post encryption)
- IndividualContractorVehicleType (Individual contractor vehicle type)
- IndividualContractorStatus (Individual Contractor working Status)

Package:

- PackageId (Package Identification)
- PackageInitialLocation (Delivery starting point)
- PackageDestination (Delivery end point)
- PackageWeight (Package Weight)
- PackageVolume (Package Volume)
- PackagePrice (Delivery Price)
- PackageDescription (Package properties Description)
- PackageShipped (Package Shipped status)
- CustomerId (Customer Id)

Transaction:

- TransactionId (Transaction Identification)
- TransactionPrice (Transaction Final price)
- TransactionStatus (Transaction completion status)
- CustomerId (Customer Identification)
- IndividualContractorID (Individual Contractor Identification)
- PackageId (Package Identification)

Relationships

- Customer (1) to Package(M): A customer can create multiple packages.
- Independent Contractor(1) to Package (M): A contractor can handle multiple packages.
- Package (1) to Transaction(1): Each package corresponds to a single transaction.
- Customer (1) to Transaction (M): A customer can have multiple transactions.
- Independent Contractor (1) to Transaction (M): A contractor can be involved in multiple transactions.

II.II Relational Model

E-R diagram

Customer

- CustomerID
- CustomerFirstName,
- CustomerMiddleName
- CustomerLastName
- CustomerDOB
- CustomerPhoneNum
- CustomerEmail
- CustomerPasswordHash

Individual Contractor

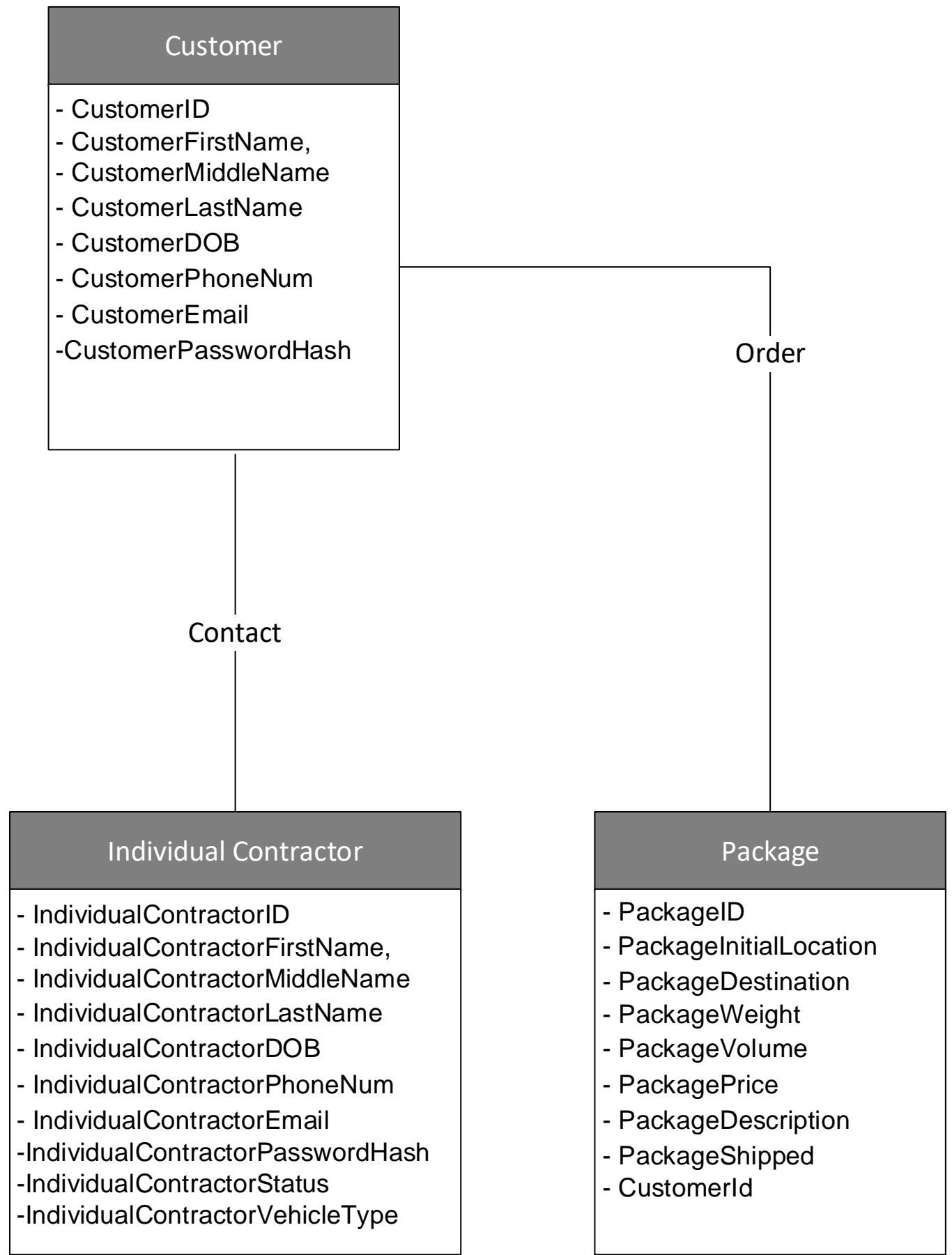
- IndividualContractorID
- IndividualContractorFirstName,
- IndividualContractorMiddleName
- IndividualContractorLastName
- IndividualContractorDOB
- IndividualContractorPhoneNum
- IndividualContractorEmail
- IndividualContractorPasswordHash
- IndividualContractorStatus
- IndividualContractorVehicleType

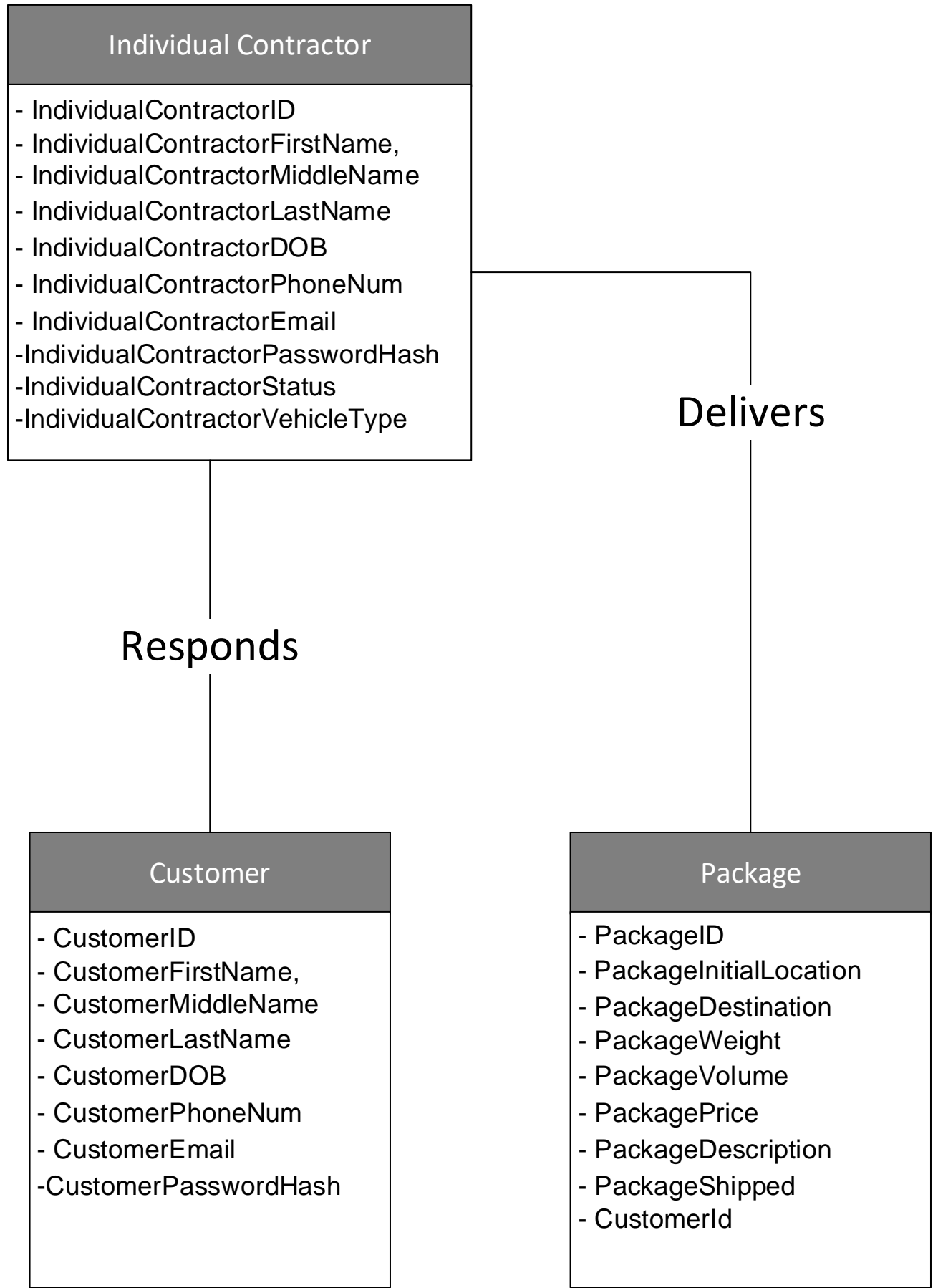
Package

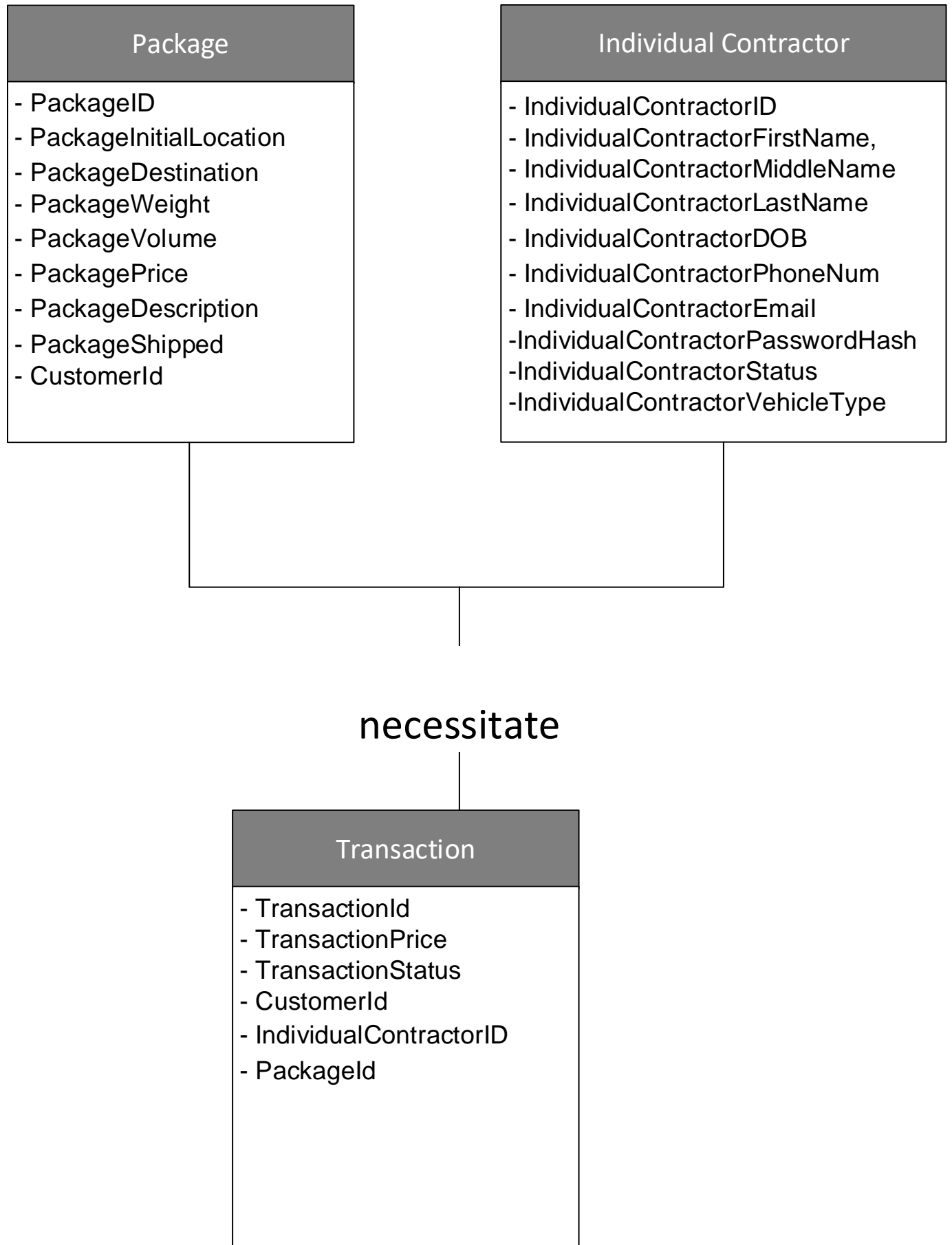
- PackageID
- PackageInitialLocation
- PackageDestination
- PackageWeight
- PackageVolume
- PackagePrice
- PackageDescription
- PackageShipped
- CustomerId

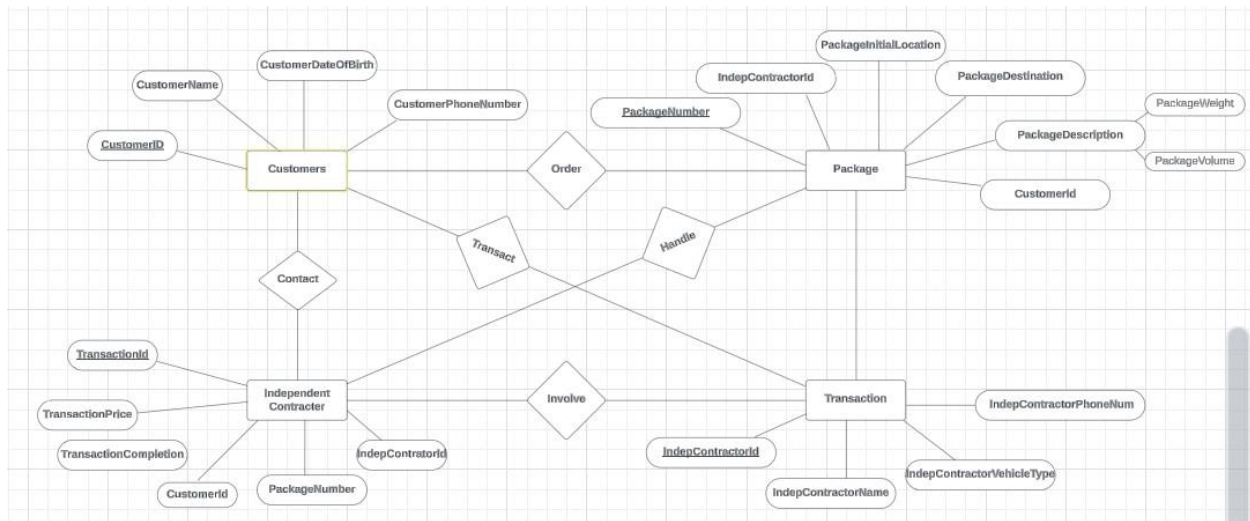
Transaction

- TransactionId
- TransactionPrice
- TransactionStatus
- CustomerId
- IndividualContractorID
- PackageId









II.III Normalization

The database design follows normalization principles to reduce redundancy:

- 1NF (First Normal Form): Each table has a primary key, and all attributes contain atomic values.
- 2NF (Second Normal Form): All non-key attributes are fully functionally dependent on the primary key.
- 3NF (Third Normal Form): There are no transitive dependencies; non-key attributes depend only on primary keys.

Physical Design

Customer Table

Attribute	Data Type	Constraints
CustomerId	varchar	PRIMARY KEY
CustomerName	varchar	NOT NULL
CustomerPhoneNum	char	NOT NULL
CustomerDateOfBirth	Date	NOT NULL

Package Table

Attribute	Data Type	Constraints
PackageNumber	varchar	PRIMARY KEY
PackageInitialLocation	varchar	NOT NULL
PackageDestination	varchar	NOT NULL
PackageWeight	int	NOT NULL
PackageVolume	int	NOT NULL
CustomerId	varchar	FOREIGN KEY (Customer.CustomerId)
IndepContractorId	varchar	FOREIGN KEY (IndependentContractor.IndepContractorId)

Independent Contractor Table

Attribute	Data Type	Constraints
IndepContractorId	varchar	PRIMARY KEY
IndepContractorName	varchar	NOT NULL
IndepContractorVehicleType.	varchar	NOT NULL
IndepContractorPhoneNum	char	NOT NULL

Transaction Table

Attribute	Data Type	Constraints
-----------	-----------	-------------

----- ----- -----			
TransactionId	varchar	PRIMARY KEY	
TransactionPrice	money	NOT NULL	
TransactionCompletion	boolean	NOT NULL	
CustomerId	varchar	FOREIGN KEY	
(Customer.CustomerId)			
PackageNumber	varchar	FOREIGN KEY	
(Package.PackageNumber)			
IndepContractorId	varchar	FOREIGN KEY	
(IndependentContractor.IndepContractorId)			

Implementation

```
create database DeliveryApp
```

```
use DeliveryApp
```

```
create table Customer(
    CustomerId int identity(1,1) primary key,
    CustomerFirstName varchar(20) not null,
    CustomerMiddleName varchar(20),
    CustomerLastName varchar(20) not null,
    CustomerDOB Date not null,
    CustomerPhoneNum char(13) not null unique,
    CustomerEmail nvarchar(255) unique not null,
    CustomerPasswordHash nvarchar(32) not null,
    constraint check_customer_age check (year(getdate()) - year(CustomerDOB) >= 18
AND year(getdate()) - year(CustomerDOB) < 100),
    constraint check_customer_phone check (CustomerPhoneNum like '+251[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'),
    constraint check_customer_email check (CustomerEmail like '%@%.%_')
)
```

```
create table IndividualContractor(
    IndividualContractorId int identity(1,1) primary key,
    IndividualContractorFirstName varchar(20) not null,
    IndividualContractorMiddleName varchar(20),
    IndividualContractorLastName varchar(20) not null,
    IndividualContractorDOB Date not null,
    IndividualContractorPhoneNum char(13) not null unique,
    IndepContractorVehicleType int not null,
```

```

        IndividualContractorStatus varchar(8) not null check(IndividualContractorStatus
in('onjob','idle','inactive')),
        IndividualContractorEmail nvarchar(255) unique not null,
        IndividualContractorPasswordHash nvarchar(32) not null,
        constraint check_IndividualContractor_age check (year(getdate()) -
year(IndividualContractorDOB) >= 18 AND year(getdate()) - year(IndividualContractorDOB)
< 100),
        constraint check_IndividualContractor_phone check (IndividualContractorPhoneNum
like '+251[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'),
        constraint check_IndividualContractor_email check (IndividualContractorEmail like
'%_@%.%_')
    )

```

```

create table Package(
    PackageId int identity(1,1) primary key,
    PackageInitialLocation varchar(60) not null,
    PackageFinalDestination varchar(60) not null,
    PackageWeight float not null,
    PackageVolume float not null,
    PackagePrice money not null,
    PackageDescription varchar(100) not null,
    PackageShipped int not null check(PackageShipped between 0 and 1),
    CustomerId int not null,
    foreign key (CustomerId) references Customer(CustomerId),
    constraint check_package_weight check (PackageWeight>=0 and
PackageWeight<2000),
    constraint check_package_volume check (PackageVolume>=0 and
PackageVolume<16),
    constraint check_package_price check (PackagePrice>=0),
)

```

```

create table Transactions(
    TransactionId int identity(1,1) primary key,
    TransactionPrice money not null,
    TransactionStatus int not null check(TransactionStatus in(-1,0,1)),
    CustomerId int not null,
    IndividualContractorId int not null,
    PackageId int not null,
    foreign key (CustomerId) references Customer(CustomerId),
    foreign key (IndividualContractorId) references
IndividualContractor(IndividualContractorId),
    foreign key (PackageId) references Package(PackageId),
)

```

```

CREATE VIEW Vpendingpackage
as
select CustomerId,PackageId,PackageInitialLocation,PackageFinalDestination
from Package
where PackageShipped=0

```

```

create view Vsentpackages
as

```


Results		Messages								
	IndividualContractor	IndividualContractor	IndividualContractor	IndividualContractorDOB	IndividualContractor	IndividualContractorVehicleType	IndividualContractorStatus	IndividualContractor	IndividualContractorPasswordHash	
1	1	Jack	M.	Smith	1999-01-12	+251911009811	1	onjob	lomb1@gmail.com	
2	2	Mekonen		Yebeley	1982-11-12	+251914009844	2	idle	Mekon@gmail.c...	
3	3	Alex		Seleshi	1989-04-03	+251921032123	3	inactive	Al@yahoo.com	
4	4	Hanf		Rashid	2000-11-12	+251987659811	4	onjob	Hanf@gmail.com	
5	5	Ronda	Sen	Lee	2001-07-07	+251998457218	1	idle	rs1@gmail.com	

Results Messages									
	PackageId	PackageInitialLocation	PackageFinalDestination	PackageWeight	PackageVolume	PackagePrice	PackageDescription	PackageShipped	CustomerId
1	1	bole	megenagna	0	0	100.00	Documents	0	1
2	2	Kazanchis	Ayat	12	3	400.00	Printer	0	4
3	3	Figa	Arat Kilo	1	1	500.00	GLass decoration *Handle with care	0	3
4	4	bole	megenagna	0	0	600.00	Documents *Urgent	0	1
5	5	Haya Hulet	Mexico	5	11	2000.00	Water pipes	0	2
6	6	Piasa	Merkato	3	2	850.00	Box Of cloth	0	5

Results Messages						
	TransactionId	TransactionPrice	TransactionStatus	CustomerId	IndividualContractorId	Packageld
1	1	400.00	1	4	4	2
2	2	100.00	-1	1	5	1
3	3	2000.00	0	2	3	4

Security Considerations

- Data Integrity: Enforce foreign key constraints to maintain referential integrity between tables.
- Access Control: Implement user roles to restrict access to sensitive data.
- Data Encryption: Encrypt sensitive information, such as customer passwords.

Future Enhancements

- Real-Time Tracking: Enable real-time package tracking for customers.
- Mobile Application: Develop a user-friendly mobile app for enhanced customer experience.
- Rating System: Implement a rating and feedback system for independent contractors.

Conclusion

The Package Delivery Management System provides a robust framework for managing package deliveries, ensuring secure transactions and efficient connections between customers and independent contractors. The database design adheres to relational model principles, promoting data integrity and organization.