

Assignment-Courier Management System Documentation

~ P325_Sarvesh Sharma

Project Overview:

- **Purpose:** A courier management system designed to efficiently manage user information, courier assignments, services, payments, and employee records.
- **Technologies Used:** Python, MSSQL

Directory Structure:S:\GIT\Hexaware_Assignments\Courier Management System

- **app:** Contains the main execution module **CourierManagementSystem.py**.
- **dao:** Data access object classes for interacting with the database, including user, courier, courier service, employee, and payment DAOs.
- **entity:** Classes representing the tables defined in SQL, including User, Courier, CourierService, Employee, and Payment.
- **myexceptions:** Custom exceptions for handling specific errors within the application.
- **util:** Contains utility classes, including **DBConnection.py** for managing database connections.

S:\GIT\Hexaware_Assignments\CourierManagementSystem

```
|— app
|  |— __init__.py
|  |— CourierManagementSystem.py
|— dao
|  |— __init__.py
|  |— UserDAO.py
|  |— UserDAOImpl.py
|  |— CourierDAO.py
```

- | |— *CourierDAOImpl.py*
- | |— *CourierServiceDAO.py*
- | |— *CourierServicesDAOImpl.py*
- | |— *EmployeeDAO.py*
- | |— *EmployeeDAOImpl.py*
- | |— *PaymentDAO.py*
- | |— *PaymentDAOImpl.py*
- |— *entity*
- | |— *__init__.py*
- | |— *User.py*
- | |— *Courier.py*
- | |— *CourierService.py*
- | |— *Employee.py*
- | |— *Payment.py*
- |— *myexceptions*
- | |— *__init__.py*
- | |— *TrackingNumberNotFoundException.py*
- | |— *InvalidEmployeeIdException.py*
- |— *util*
- | |— *__init__.py*
- | |— *DBConnection.py*
- |— *CourierManagement_DB.sql*
- |— *venv*
- |— *P325_Sarvesh Sharma_Assignment-Courier Management System .pdf*

Project Flow:

- **Database Setup:** Created SQL file to define the following main tables:
 - **User_table:** Stores user information such as ID, name, email, password, contact number, and address.
 - **Courier:** Contains details about each courier, including sender and receiver information, weight, status, tracking number, and delivery date.
 - **CourierServices:** Defines various courier services with service IDs, names, and costs.
 - **Employee:** Records employee details including ID, name, email, contact number, role, salary, and associated courier.
 - **Location:** Maintains location data for deliveries.
 - **Payment:** Tracks payment information related to couriers and locations.
- **Entity Classes**
 - Defined classes for User, Courier, CourierService, Employee, Location, and Payment.
 - Each class contains attributes with corresponding getter and setter methods.
- **DAO Implementation**
 - Developed interfaces for each entity (e.g., **UserDAO**, **CourierDAO**, **CourierServiceDAO**, **EmployeeDAO**, **PaymentDAO**) and their implementations (e.g., **UserDAOImpl**, **CourierDAOImpl**, **CourierServicesDAOImpl**, **EmployeeDAOImpl**, **PaymentDAOImpl**).
 - Methods implemented include creating, updating, deleting, and fetching records for users, couriers, services, employees, and payments.
- **Utility Class**
 - Created **DBConnection.py** to manage database connections for executing SQL queries.
- **Exception Handling**
 - Implemented custom exceptions such as **InvalidEmployeeIdException**, **TrackingNumberNotFoundException** to handle various error case
- **Main Application Logic**
 - Developed a menu-driven interface for user interactions.
 - Allowed users to create, update, and delete users, couriers, and employees, as well as manage courier services and payments.
 - Enabled functionalities for fetching all couriers, getting delivery history, and listing all courier services.
- **Sample Data**
 - Inserted sample data into the database for testing

Outputs:

- Tables defined in **CourierManagement_DB.sql**:

```
127
128 SELECT * FROM User_table;
129
```

100 %

Results Messages

UserID	Name	Email	Password	ContactNumber	Address
1	Alice Smith	alice@example.com	password123	1234567890	123 Elm St
2	Bob Johnson	bob@example.com	password123	0987654321	456 Oak St
3	Charlie Brown	charlie@example.com	password123	2345678901	789 Pine St
4	Diana Prince	diana@example.com	password123	3456789012	101 Maple St
5	Ethan Hunt	ethan@example.com	password123	4567890123	202 Birch St

```
130 SELECT * FROM Employee;
```

19 %

Results Messages

	EmployeeID	Name	Email	ContactNumber	Role	Salary	CourierID
1	1	John Doe	john@example.com	5678901234	Delivery Driver	3000.00	1
2	2	Jane Doe	jane@example.com	6789012345	Delivery Driver	3200.00	2
3	3	Mark Smith	mark@example.com	7890123456	Warehouse Staff	2800.00	NULL

```
132 SELECT * FROM Location;
133
```

119 %

Results Messages

	LocationID	LocationName	Address
1	1	Warehouse A	303 Warehouse St
2	2	Warehouse B	404 Storage St
3	3	Branch Office	505 Business Rd

133
134
135
136

SELECT * FROM Courier;

119 %

Results

Messages

	CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	1	1	123 Elm St	2	456 Oak St	2.50	Pending	TRACK12345	2024-10-20
2	2	2	456 Oak St	3	789 Pine St	1.00	Delivered	TRACK54321	2024-10-18
3	3	3	789 Pine St	4	101 Maple St	3.50	In Transit	TRACK67890	2024-10-19
4	4	4	101 Maple St	5	202 Birch St	2.00	Pending	TRACK98765	2024-10-21
5	5	5	202 Birch St	1	303 Warehouse St	4.00	Pending	TRACK45678	2024-10-22

133
136

SELECT * FROM CourierServices;

119 %

Results

Messages

	ServiceID	ServiceName	Cost
1	1	Standard Delivery	5.00
2	2	Express Delivery	10.00
3	3	Same Day Delivery	15.00

137
138
139

SELECT * FROM Payment;

119 %

Results

Messages

	PaymentID	CourierID	LocationID	Amount	PaymentDate
1	1	1	1	5.00	2024-10-15
2	2	2	2	10.00	2024-10-16
3	3	3	3	15.00	2024-10-17
4	4	4	1	5.00	2024-10-18
5	5	5	2	10.00	2024-10-19

➤ 1. Create User:

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get Delivery History
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 1
Enter User ID: 7
Enter User Name: qw
Enter Email: qw
Enter Password: wq
Enter Contact Number: 123214
Enter Address: sd
User created successfully!
```

127

128 `SELECT * FROM User table;`

119 %

Results Messages

	UserID	Name	Email	Password	ContactNumber	Address
1	1	Alice Smith	alice@example.com	password123	1234567890	123 Elm St
2	2	Bob Johnson	bob@example.com	password123	0987654321	456 Oak St
3	3	Charlie Brown	charlie@example.com	password123	2345678901	789 Pine St
4	4	Diana Prince	diana@example.com	password123	3456789012	101 Maple St
5	5	Ethan Hunt	ethan@example.com	password123	4567890123	202 Birch St
6	6	Demo_User	demo@gmail.com	demo@pass	145802361	Demo Street
7	7	qw	qw	wq	123214	sd

➤ 2. Delete User

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get Delivery History
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 2
Enter User ID to delete: 7
User deleted successfully!
```

128 SELECT * FROM User table;

119 %

	UserID	Name	Email	Password	ContactNumber	Address
1	1	Alice Smith	alice@example.com	password123	1234567890	123 Elm St
2	2	Bob Johnson	bob@example.com	password123	0987654321	456 Oak St
3	3	Charlie Brown	charlie@example.com	password123	2345678901	789 Pine St
4	4	Diana Prince	diana@example.com	password123	3456789012	101 Maple St
5	5	Ethan Hunt	ethan@example.com	password123	4567890123	202 Birch St
6	6	Demo_User	demo@gmail.com	demo@pass	145802361	Demo Street

➤ 3. Fetch User Details

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get Delivery History
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 3
Enter User ID to fetch details: 3
User Details: ID: 3, Name: Charlie Brown, Email: charlie@example.com, Address: 789 Pine St
```

➤ 4. Create Employee

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get Delivery History
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 4
Enter Employee ID: 4
Enter Employee Name: New_Employee
Enter Email: emp@gmail.com
Enter Contact Number: 7880235480
Enter Role: Intern
Enter Salary: 15000
Employee created successfully!
```


Results Messages							
	EmployeeID	Name	Email	ContactNumber	Role	Salary	CourierID
1	1	John Doe	john@example.com	5678901234	Delivery Driver	3000.00	1
2	2	Jane Doe	jane@example.com	6789012345	Delivery Driver	3200.00	2
3	3	Mark Smith	mark@example.com	7890123456	Warehouse Staff	2800.00	NULL
4	4	New_Employee	emp@gmail.com	7880235480	Intern	15000.00	NULL

➤ 5. Delete Employee

```

**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 7
Enter Employee ID to delete: 3
Employee deleted successfully.
Employee deleted successfully!

```

Results Messages							
	EmployeeID	Name	Email	ContactNumber	Role	Salary	CourierID
1	1	John Doe	john@example.com	5678901234	Delivery Driver	3000.00	1
2	2	Jane Doe	jane@example.com	6789012345	Delivery Driver	3200.00	2

➤ 6. Fetch All Employees

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get Delivery History
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 6
Employee ID: 1, Name: John Doe, Role: Delivery Driver
Employee ID: 2, Name: Jane Doe, Role: Delivery Driver
Employee ID: 3, Name: Mark Smith, Role: Warehouse Staff
Employee ID: 4, Name: New_Employee, Role: Intern
```

➤ 7. Fetch Employee by ID

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Delete Employee
6. Fetch All Employees
7. Fetch Employee by ID
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 7
Enter Employee ID to fetch: 2
Employee Details:
ID: 2
Name: Jane Doe
Email: jane@example.com
Contact Number: 6789012345
Role: Delivery Driver
Salary: 3200.00
```

```

127
128 SELECT * FROM User_table;
129 SELECT * FROM Employee;
130 SELECT * FROM Location;

```

9 %

Results Messages

EmployeeID	Name	Email	ContactNumber	Role	Salary	CourierID
1	John Doe	john@example.com	5678901234	Delivery Driver	3000.00	1
2	Jane Doe	jane@example.com	6789012345	Delivery Driver	3200.00	2
3	Mark Smith	mark@example.com	7890123456	Warehouse Staff	2800.00	NULL

➤ 8. Insert New Courier

```

**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get Delivery History
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 8
Enter Courier ID: 6
Enter Sender Name (UserID): 5
Enter Sender Address: SAddr
Enter Receiver Name (UserID): 6
Enter Receiver Address: RAddr
Enter Weight: 1.2
Enter Status: Pending
Enter Tracking Number: TRACK0000
Enter Delivery Date (YYYY-MM-DD): 2024-10-24
Courier order created successfully!

```

128
129
130

SELECT * FROM Courier;

119 %

Results

Messages

	CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	1	1	123 Elm St	2	456 Oak St	2.50	Pending	TRACK12345	2024-10-20
2	2	2	456 Oak St	3	789 Pine St	1.00	Delivered	TRACK54321	2024-10-18
3	3	3	789 Pine St	4	101 Maple St	3.50	In Transit	TRACK67890	2024-10-19
4	4	4	101 Maple St	5	202 Birch St	2.00	Pending	TRACK98765	2024-10-21
5	5	5	202 Birch St	1	303 Warehouse St	4.00	Pending	TRACK45678	2024-10-22
6	6	5	SAddr	6	RAddr	1.20	Pending	TRACK0000	2024-10-24

➤ 9. Update Courier Status

**** Courier Management System ****

1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get Delivery History
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit

Enter your choice: 9

Enter Courier ID to update status: 6

Enter New Status: Updated_Status

Courier status updated successfully!

119 %

DBOD TABLE Location:

	CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	1	1	123 Elm St	2	456 Oak St	2.50	Pending	TRACK12345	2024-10-20
2	2	2	456 Oak St	3	789 Pine St	1.00	Delivered	TRACK54321	2024-10-18
3	3	3	789 Pine St	4	101 Maple St	3.50	In Transit	TRACK67890	2024-10-19
4	4	4	101 Maple St	5	202 Birch St	2.00	Pending	TRACK98765	2024-10-21
5	5	5	202 Birch St	1	303 Warehouse St	4.00	Pending	TRACK45678	2024-10-22
6	6	5	SAddr	6	RAddr	1.20	Updated_Status	TRACK0000	2024-10-24

➤ 10. Get All Couriers

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 10
All Couriers:
ID: 1, Sender: 1, Receiver: 2, Status: Pending
ID: 2, Sender: 2, Receiver: 3, Status: Delivered
ID: 3, Sender: 3, Receiver: 4, Status: In Transit
ID: 4, Sender: 4, Receiver: 5, Status: Pending
ID: 5, Sender: 5, Receiver: 1, Status: Pending
ID: 6, Sender: 5, Receiver: 6, Status: Updated_Status
```

➤ 11. Create Courier Service

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 11
Enter Service ID: 4
Enter Service Name: InstantDelivery
Enter Cost: 50
Courier service created successfully!
```

119 %

Results			
	ServiceID	ServiceName	Cost
1	1	Standard Delivery	5.00
2	2	Express Delivery	10.00
3	3	Same Day Delivery	15.00
4	4	InstantDelivery	50.00

➤ 12. Update Courier Service

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 12
Enter Service ID to update: 4
Enter New Service Name (Leave blank for no change):
Enter New Cost (Leave blank for no change): 100
Courier service updated successfully!
```

119 %

Results		Messages	
	ServiceID	ServiceName	Cost
1	1	Standard Delivery	5.00
2	2	Express Delivery	10.00
3	3	Same Day Delivery	15.00
4	4	InstantDelivery	100.00

➤ 13. Delete Courier Service

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 13
Enter Service ID to delete: 4
Courier service deleted successfully!
```

119 %

Results			
	ServiceID	ServiceName	Cost
1	1	Standard Delivery	5.00
2	2	Express Delivery	10.00
3	3	Same Day Delivery	15.00

➤ 14. Fetch Courier Service Details

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 14
Enter Service ID to fetch details: 2
Courier Service Details: ID: 2, Name: Express Delivery, Cost: 10.00
```

➤ 15. List All Courier Services

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 15
Service ID: 1, Name: Standard Delivery, Cost: 5.00
Service ID: 2, Name: Express Delivery, Cost: 10.00
Service ID: 3, Name: Same Day Delivery, Cost: 15.00
Service ID: 4, Name: InstantDelivery, Cost: 100.00
```

➤ 16. Read Payment

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 16
Enter Payment ID to read: 3
Payment Details: ID: 3, Amount: 15.00, Date: 2024-10-17
```

119 %					
Results Messages					
	PaymentID	CourierID	LocationID	Amount	PaymentDate
1	1	1	1	5.00	2024-10-15
2	2	2	2	10.00	2024-10-16
3	3	3	3	15.00	2024-10-17
4	4	4	1	5.00	2024-10-18
5	5	5	2	10.00	2024-10-19

➤ 17. 0 Exit

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 0
Exiting Courier Management System...
```

➤ 18. 99 Invalid Argument

```
**** Courier Management System ****
1. Create User
2. Delete User
3. Fetch User Details
4. Create Employee
5. Update Employee
6. Fetch All Employees
7. Delete Employee
8. Insert New Courier
9. Update Courier Status
10. Get All Couriers
11. Create Courier Service
12. Update Courier Service
13. Delete Courier Service
14. Fetch Courier Service Details
15. List All Courier Services
16. Read Payment
0. Exit
Enter your choice: 99
Invalid choice! Please try again.
```

