



Presentation on Courier Management System

Project guide:

Sir Mr. Sushil Adhikari

Prepared by:

Sandesh Giri (NPI000041)

Suraj Pandey (NPI000051)

TABLE OF CONTENT

- Introduction
- Objective
- UML Diagram
 - ◆ Class Diagram
 - ◆ Use Case Diagram
- User Roles
- Use of Object Oriented Programming Language
- Advantages of Object Oriented System
- Limitations
- Conclusion
- References

Introduction

- Our project is developed under java programming language
- System can handle user account management, order management, view report and feedback management.
- Data are stored in txt files.

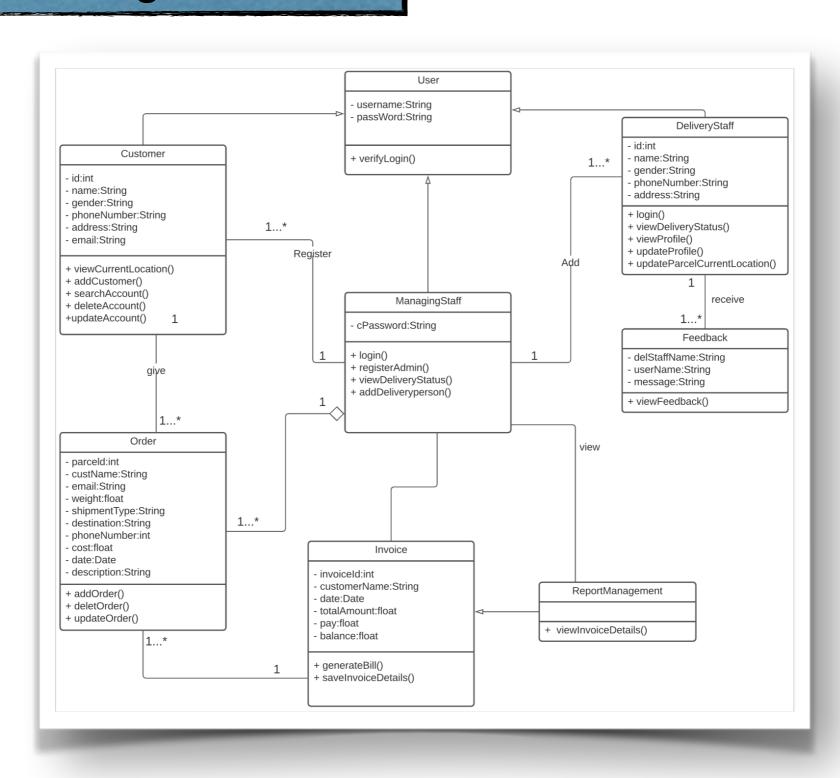
Objective

To manage details of customer, delivery staff, orders, invoice report.

To reduce the manual working

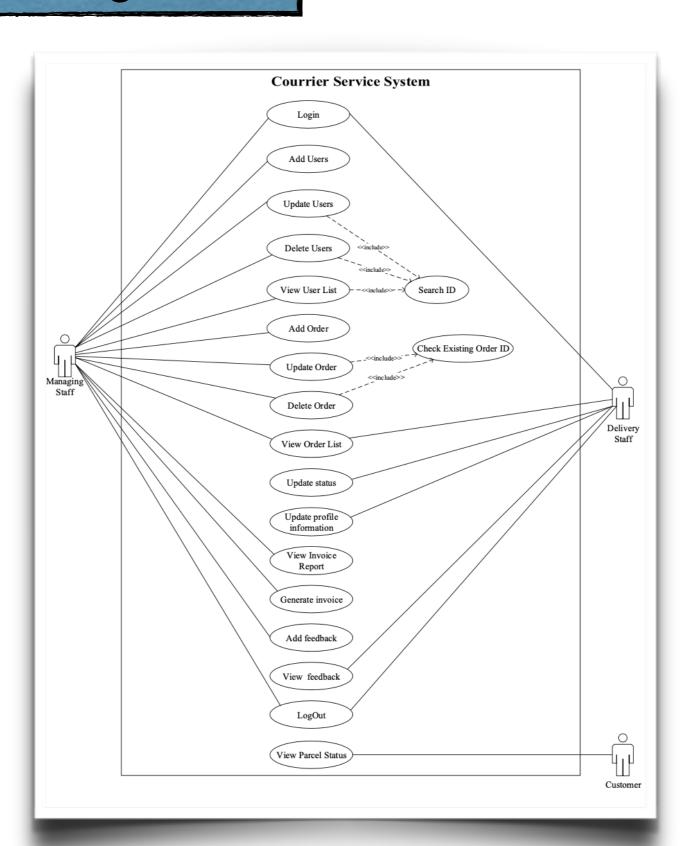
UML diagram

Class Diagram



UML diagram

Use Case Diagram



User Role

Managing Staff Roles



Manage User Account



Manage Order



Generate Invoice



View Invoice Report



Send Feedback



User Role

Delivery Staff Roles



View Delivery Status



Update Parcel current location



View Personal Profile



Update Profile

Creation of class and object

```
public class User {
    private String username;
    private String passWord;

User user = new User(usernamekb, passwordkb)
user.setUsername(usernamekb);
user.setPassWord(passwordkb);
user.login();
jTextField_un.setText(null);
jPasswordField_pass.setText(null);
```

Encapsulation

```
public class User {
   private String username;
   private String passWord;
    public User(String username, String passWord) {
       this.username = username;
       this.passWord = passWord;
    public String getUsername() {
        return username;
   public void setUsername(String username) {
        this.username = username;
    public String getPassWord() {
        return passWord;
    public void setPassWord(String passWord) {
       this.passWord = passWord;
```

Constructor

Inheritance

```
public class Invoice extends javax.swing.JFrame {
    /**
    */
    public Invoice() {
        initComponents();
    }

public Invoice(String name,String date, String idd,String total,String pay,String balance) {
        jLabel_customerName.setText(name);
        jLabel_date.setText(date);
        jLabel_parcelId.setText(idd);
        jLabel_total.setText(total);
        jLabel_pay.setText(pay);
        jLabel_balance.setText(balance);
}
```

Advantages of Object Oriented System

- Code Reusability through inheritance
- Flexible due to the use of polymorphism
- **Simple**
- Fast development of software
- Maintenance.

Limitations

- Cannot delete and update feedback.
- Data related to the customer are not deleted when customer account is deleted from the file.
- Can only view invoice report stored in file(cannot view weekly, monthly and annual report separately).
- Delivery staff can view all the orders.

Conclusion

- System is developed under OOP concept by the creation of class, object, inheritance, constructor and encapsulation.
- Software that is used to code this system was Netbeans.
- The targeted users are managing staff and delivery staff.

References

- ALTVATER, A. (2020). What Are OOP Concepts in Java? The Four Main OOP Concepts in Java, How They Work, Examples, and More. Retrieved 8 August 2021, from https://stackify.com/oops-concepts-in-java/
- Capka, D. (2021). Lesson 7 Inheritance and polymorphism in Java. Retrieved 10 August 2021, from https://www.ictdemy.com/java/oop/inheritance-and-polymorphism-in-java
- Encapsulation in Java GeeksforGeeks. (2021). Retrieved 9 August 2021, from https://www.geeksforgeeks.org/
 encapsulation-in-java/
- Gills, A., & Lewis, S. (2021). What is Object-Oriented Programming (OOP)?. Retrieved 8 August 2021, from https://searchapparchitecture.techtarget.com/definition/object-oriented-programming-OOP
- Inheritance in Java GeeksforGeeks. (2021). Retrieved 7 August 2021, from https://www.geeksforgeeks.org/inheritance-in-java/
- Java Constructors. (2021). Retrieved 11 August 2021, from https://www.tutorialspoint.com/java/java_constructors.htm
- Polymorphism in Java javatpoint. (2021). Retrieved 10 August 2021, from https://www.javatpoint.com/runtime-polymorphism-in-java

Work Division

	Sandesh Giri (NPI000041)	Suraj Pandey (NPI000051)
Delivery Account Management		
Customer Account Management		
Order Management		
Feedback Management		

!! THANK YOU!!