DISTREX

CS261- Design Document



Project Supervisor

Mr. Samyan Qayyum Wahla

Project Members(Group 38)

Shakeel Ahmed 2021-CS-03

Ahsan Imtiaz 2021-CS-37

Usman Asghar 2021-CS-46

Department of Computer Science

University of Engineering and Technology, Lahore

Pakistan

Table of Contents

[Project Description: 3](#_Toc120312202)

[Project Features: 3](#_Toc120312203)

[Technology Stack: 4](#_Toc120312204)

[Project Actors: 4](#_Toc120312205)

[Use Cases: 4](#_Toc120312206)

[Use Case 1 4](#_Toc120312207)

[Use Case 2 4](#_Toc120312208)

[Use Case 3 4](#_Toc120312209)

[Use Case 4 4](#_Toc120312210)

[Use Case 5 5](#_Toc120312211)

[Use Case 6 5](#_Toc120312212)

[Use Case 7 5](#_Toc120312213)

[Use Case 8 5](#_Toc120312214)

[Use Case 9 5](#_Toc120312215)

[Use Case 10 5](#_Toc120312216)

[Use Case 11 6](#_Toc120312217)

[Use Case 12 6](#_Toc120312218)

[Use Case 13 6](#_Toc120312219)

[Use Case 14 6](#_Toc120312220)

[Use Case 15 6](#_Toc120312221)

[Use Case 16 6](#_Toc120312222)

[Use Case 17 7](#_Toc120312223)

[Use Case 18 7](#_Toc120312224)

[Use Interfaces: 7](#_Toc120312225)

[User Interface Details 19](#_Toc120312226)

[Classes: 20](#_Toc120312227)

[Object-Oriented Features: 20](#_Toc120312228)

[Composition: 20](#_Toc120312229)

[Inheritance: 20](#_Toc120312230)

[Multiple Inheritance: 20](#_Toc120312231)

[Multi-Level Inheritance: 20](#_Toc120312232)

[Polymorphism: 20](#_Toc120312233)

[Detailed Object-Oriented Design: 20](#_Toc120312234)

[Data Structure: 21](#_Toc120312235)

[Exceptions: 21](#_Toc120312236)

[Data Storage: 21](#_Toc120312237)

[Email Sending: 21](#_Toc120312238)

[Project Plan: 21](#_Toc120312239)

[Analytical Reports: 22](#_Toc120312240)

# Project Description:

The idea of our design is to make noncommercial easier. What's the" job of a wholesaler"? A company that buys many products from manufacturers and sells them to retailers (stores and shops) through their riders is known as an entire deal. Their primary issue is that they've to manage orders and keep track of each client manually. In addition, the company must cover product deals to determine which bones lower the company's value and induce further profit. Even though they're suitable to manage records using a hard form manually, they need help assigning each order to a rider and electing which lifts are suitable to pierce particular guests more fluently.

Some of their riders go to their regular guests to take orders. The riders take orders from guests by hand, record them in a journal, and shoot them to the business. This is also not their thing and is extremely delicate to manage. The company must manage the rider's track to determine the rider's energy charges, which the company will cover in addition to its payment. Also, detecting the client is difficult for the rider, which can delay the rider's transportation time. The retailer faces multitudinous difficulties when supplying the product because of the lengthy and time-consuming nature of the process. The retailer might only sometimes have the item the client wants.

We can take care of this large number of issues by creating an operation that can deal with every one of these errands. For illustration, we can give a climate to the association by an operation on which the association can deal with its stock, keep a record of benefits, deal, and mischance to work on its benefit. Organizations can keep up with track of the rider to work out its energy cost. This operation also works with the rider to explore its customer all the more effectively, which will save him energy, the rider can likewise take orders on the operation from the customer, and the customer can gain entrance to the association's stock throughout the rider's operation. Through our operation, the rider can fluently submit all of these orders to the business with just many clicks.

The rider will exclude all major issues in this manner, allowing the business to manage its deals more and increase gains. This operation also saves plutocrats on paper, which the company used to keep track of its records. Also, the client has easier access to the company's products.

The fact that some of the company's riders are unskillful and reckless causes the company to exclude them, which will help them ameliorate their delivery service. Our operation also makes it easier for the business to filter their riders by giving the client a standing system for the riders that can be saved for after use on the operation. One major issue that many businesses face is that it is difficult to keep track of every customer's information, including their phone number, email address, products purchased in the past, and address. Additionally, our application will resolve that company's major issue.

Our application will offer a medium for conversation between the rider and the employee. This will make it easier for both parties to communicate within the same platform, saving them time and making their work more efficient. The rider and the employee can have a conversation through our application. Both parties can communicate more easily within the same platform, saving them time and making their work more effective.

Many stores and supermarkets are using present-day computer systems to control many of their business practices, giving possessors further information to help them make opinions and automating some of the processes in the store to save time and plutocrat. The store is suitable to keep a force that they can fluently search, and that's streamlined each time an item is sold, so they always know if it's still available.

We will also allow customers to purchase online and pick it up at our store because doing so will reduce shipping costs, provide them with more options (in terms of colors, sizes, and texture quality), and speed up the shipping process. Additionally, it will boost customer trust in the company. In addition, anyone who urgently requires a product can use this facility.

# Project Features:

* Employees can manage inventory.
* Employees can manage deliveries.
* Employees can update their profiles.
* Riderser can update their profiles.
* Rider can update the status of the delivery.

# Technology Stack:

|  |  |
| --- | --- |
| Language | C# |
| IDEs | Microsoft Visual Studio Community |

# Project Actors:

1. **Employee:** The employee can manage inventory, assign orders to each rider, check the status of the order, take a list of orders from the rider, can make an analysis of the sales of each product, and its fuel expenses.
2. **Rider:** The purpose of the rider is to deliver the products and it is his responsibility to update the status of the delivery.

# Use Cases:

All the use cases should be written in the following format. The name of use should start with a verb e.g. Add Student. It should not be as Student/ Student Add. Add Student is a separate use case and Edit Student is a separate use case: (Reference Document for use cases: <https://www.craiglarman.com/wiki/downloads/applying_uml/larman-ch6-applying-evolutionary-use-cases.pdf> )

## Use Case 1

|  |  |
| --- | --- |
| Use Case ID | U01 |
| Name | Add product |
| Actor | Employee |
| Description | In this use case, an employee can add products to the company's inventory by entering their name, price, quantity, and brand. |
| Flow | Home>Manage Inventory>Add products |

## Use Case 2

|  |  |
| --- | --- |
| Use Case ID | U02 |
| Name | Update Product |
| Actor | Employee |
| Description | In this use case, an employee can update products to the company's inventory by entering their name, price, quantity, and brand. |
| Flow | Home>Manage Inventory>Update products |

## Use Case 3

|  |  |
| --- | --- |
| Use Case ID | U03 |
| Name | Update Name |
| Actor | Employee |
| Description | In this use case, the employee has the option to update their Name. By clicking the button "update name", they can change their first and last names. |
| Flow | Home>Update Profile>Update Employee Name |

## Use Case 4

|  |  |
| --- | --- |
| Use Case ID | U04 |
| Name | Update Contact Information |
| Actor | Employee |
| Description | In this use case, the employee has the option to update their contact information. By clicking the button that says "update contact information," they can change their email address and phone number. |
| Flow | Home>Update Profile>Update Contact Information |

## Use Case 5

|  |  |
| --- | --- |
| Use Case ID | U05 |
| Name | Update Username |
| Actor | Employee |
| Description | In this use case, the employee can modify the username. To do so, the employee must click the button labeled "Update Username." |
| Flow | Home >Update Profile>Update Username |

## Use Case 6

|  |  |
| --- | --- |
| Use Case ID | U06 |
| Name | Update Password |
| Actor | Employee |
| Description | In this use case, the employee can change the password. By clicking the button that says "update password," they can add a new password and confirm it. |
| Flow | Home >Update Profile>Update Password |

## Use Case 7

|  |  |
| --- | --- |
| Use Case ID | U07 |
| Name | Add Delivery |
| Actor | Employee |
| Description | In this use case, the employee can add delivery to the company’s inventory by entering their product name, quantity, address, consignee, and rider. |
| Flow | Home >Manage Delivery>Add delivery |

## Use Case 8

|  |  |
| --- | --- |
| Use Case ID | U08 |
| Name | View Analysis |
| Actor | Employee |
| Description | In this use case, the employee can view the rider's analysis by clicking the download button and viewing the profit loss and fuel consumption reports. |
| Flow | Home >View Analysis |

## Use Case 9

|  |  |
| --- | --- |
| Use Case ID | U09 |
| Name | Update Name |
| Actor | Rider |
| Description | In this use case, the rider has the option to update their Name. By clicking the button "update name", they can change their first and last names. |
| Flow | Home >Update Profile>Update Rider Name |

## Use Case 10

|  |  |
| --- | --- |
| Use Case ID | U10 |
| Name | Update Contact Information |
| Actor | Rider |
| Description | In this use case, the rider has the option to update their contact information. By clicking the button that says "update contact information," they can change their email address and phone number. |
| Flow | Home >Update Profile>Update Contact Information |

## Use Case 11

|  |  |
| --- | --- |
| Use Case ID | U11 |
| Name | Update Username |
| Actor | Rider |
| Description | In this use case, the rider can modify the username. To do so, the rider must click the button labeled "Update Username." |
| Flow | Home >Update Profile>Update Username |

## Use Case 12

|  |  |
| --- | --- |
| Use Case ID | U12 |
| Name | Update Password |
| Actor | Rider |
| Description | In this use case, the rider can change the password. By clicking the button that says "update password," they can add a new password and confirm it. |
| Flow | Home >Update Profile>Update Password |

## Use Case 13

|  |  |
| --- | --- |
| Use Case ID | U13 |
| Name | Delivery Assignation |
| Actor | Employee |
| Description | In this use case, the employee will provide information about delivery to the rider. The rider will be able to click on "view map," where they will be able to enter the delivery address and order number provided by the employee. |
| Flow | Home>Assign Delivery>View Map |

## Use Case 14

|  |  |
| --- | --- |
| Use Case ID | U14 |
| Name | View Complete Delivery Information |
| Actor | Rider |
| Description | After all, deliveries have been completed in this use case, the rider will be able to view all delivery information. |
| Flow | Home >View Delivery Information |

## Use Case 15

|  |  |
| --- | --- |
| Use Case ID | U15 |
| Name | Sign Up |
| Actor | Employee and Rider |
| Description | In this use case, both the employee and the rider will sign up to use the application. Both will enter their first and last names, usernames, email addresses, phone numbers, passwords, and roles when signing up. |
| Flow | Sign In>Sign Up |

## Use Case 16

|  |  |
| --- | --- |
| Use Case ID | U16 |
| Name | Sign In |
| Actor | Rider and Employee |
| Description | In this use case, both the employee and the rider will sign in to open their dashboard respectively. |
| Flow | Sign In>Employee Dashboard/Rider Dashboard |

## Use Case 17

|  |  |
| --- | --- |
| Use Case ID | U17 |
| Name | Employee Dashboard |
| Actor | Employee |
| Description | In this use case, Employees will be able to manage inventory, manage deliveries, view analysis, update their profiles, and sign out of the application |
| Flow | Sign In>Employee Dashboard>Manage Inventory>Manage Deliveries>View Analysis>Update Profile>Sign Out |

## Use Case 18

|  |  |
| --- | --- |
| Use Case ID | U18 |
| Name | Rider Dashboard |
| Actor | Rider |
| Description | In this use case, Rider will be able to assign deliveries, view the complete deliveries status, update the profile and sign out. |
| Flow | Sign In>Rider Dashboard>Assigned Deliveries>View Delivery>Update Profile>sign out |

# Use Interfaces:

|  |  |
| --- | --- |
| Interface ID | I01 |
| Name | Sign In |
| Linked Use Case | It is linked with use case Id numbers **U17, U18, and U15.** |
| UI Screen (Just in mind) |  |
| Validators | Password must be at least 8 digits, Username contains at least one number, and Username already exists. |

|  |  |
| --- | --- |
| Interface ID | I02 |
| Name | Sign Up |
| Linked Use Case | It is linked with use case Id number **U16.** |
| UI Screen (Just in mind) |  |
| Validators | Password must be at least 8-digit, Username contains at least one number |

|  |  |
| --- | --- |
| Interface ID | I03 |
| Name | Employee Dashboard |
| Linked Use Case | It is linked with Use Case Id number **U17.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I04 |
| Name | Rider Dashboard |
| Linked Use Case | It is linked to use case Id number **U18.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I05 |
| Name | Manage Inventory |
| Linked Use Case | It is linked with use case Id numbers **U17, U01, and U02.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I06 |
| Name | Add Product |
| Linked Use Case | It is linked with use case Id number **U17.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I06 |
| Name | Update Product |
| Linked Use Case | It is linked with use case Id number **U17.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I07 |
| Name | Manage Deliveries |
| Linked Use Case | It is linked with use case Id numbers **U17 and U07.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I08 |
| Name | Add delivery |
| Linked Use Case | It is linked with use case Id number **U17.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I09 |
| Name | View Analysis |
| Linked Use Case | It is linked with use case Id number **U17.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I10 |
| Name | Update Employee Profile |
| Linked Use Case | It is linked with use case Id numbers **U17, U03, U04, U05, and U06.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I11 |
| Name | Update employee name |
| Linked Use Case | It is linked with the use case Id number **U17 and the Update Profile form.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I12 |
| Name | Update employee username |
| Linked Use Case | It is linked with the use case Id number **U17 and the Update Profile form.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented, and the username already exists. |

|  |  |
| --- | --- |
| Interface ID | I13 |
| Name | Update employee contact information |
| Linked Use Case | It is linked with the use case Id number **U17 and the Update Profile form.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented, valid email addresses contain “@”, and contain 11 digits numbers. |

|  |  |
| --- | --- |
| Interface ID | I14 |
| Name | Update employee password |
| Linked Use Case | It is linked with the use case Id number **U17 and the Update Profile form.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented, and contain at least 8 digits. |

|  |  |
| --- | --- |
| Interface ID | I15 |
| Name | Update Rider Profile |
| Linked Use Case | It is linked with use case Id numbers **U18, U09, U10, U11, and U12.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I16 |
| Name | Update rider name |
| Linked Use Case | It is linked with the use case Id number **U18 and the Update Profile form.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented. |

|  |  |
| --- | --- |
| Interface ID | I17 |
| Name | Update rider username |
| Linked Use Case | It is linked with the use case Id number **U18 and the Update Profile form.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented, and the username already exists. |

|  |  |
| --- | --- |
| Interface ID | I18 |
| Name | Update rider contact information |
| Linked Use Case | It is linked with the use case Id number **U18 and the Update Profile form.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented, valid email addresses contain “@”, and contain 11 digits numbers. |

|  |  |
| --- | --- |
| Interface ID | I19 |
| Name | Update rider password |
| Linked Use Case | It is linked with the use case Id number **U18 and the Update Profile form.** |
| UI Screen (Just in mind) |  |
| Validators | Validation on this page is that control all unexceptional errors and by clicking on every button, all will be implemented, and contain at least 8 digits. |

# User Interface Details

In this section, fill the table for the summary that which use case will have the required component. Inside each box, write the counts for each component. a If a component is not used, write zero.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Interface Id | Textbox | Dropdown | Password Box | Table | Date Field | Buttons | AutoComplete | Radio Button | Checkbox | Menu | Text Area | Progress bar |
| I01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| I02 |  |  |  |  |  |  |  |  |  |  |  |  |

# Classes:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Class Name** | **Software/ Domain** | **Is Abstract (Yes/No)** | **Is Singleton (Yes/No)** | **Is the class will has parametrized constructor(Yes/No)** |
| Employee BL |  | No |  | Yes |
| Employee DL |  | No |  | Yes |
| Rider BL |  | No |  | Yes |
| Rider DL |  | No |  | Yes |
| Product BL |  | No |  | Yes |
| Product DL |  | No |  | Yes |
| Delivery BL |  | No |  | Yes |
| Delivery DL |  | No |  | Yes |

# Object-Oriented Features:

## Composition:

## Inheritance:

## Multiple Inheritance:

## Multi-Level Inheritance:

## Polymorphism:

# Detailed Object-Oriented Design:

# Data Structure:

|  |  |  |
| --- | --- | --- |
| Use Case Id | Data Structures Used | Justification for the usage of data structure |
| U01 | Linked List | This will be space efficient. |
| U07 | Queue | This will help in handling the sequence of the deliveries |
| U13 | Queue | This will help in handling the sequence of the deliveries |
| U15 | Linked List | This will be efficient. |

# Exceptions:

|  |  |  |  |
| --- | --- | --- | --- |
| Type of Exception | Why this exception will occur | Use Case Id in which exception could occur | How you will handle the exception |
| Invalid Input | This exception will occur during entering data on products. | U01, U02, U03, U04, U05, U06, U07, U09, U10, U11, U12, U13, U15, U16 | We will handle exceptions through the Try-Catch block |
| Empty Field(s) | This will occur while signing up or signing in, entering data on products, entering data on deliveries, updating products or updating deliveries, and updating the contact information of employees or riders. | U01, U02, U03, U04, U05, U06, U07, U09, U10, U11, U12, U13, U15, U16 | We will handle exceptions through the Try-Catch block |

# Data Storage:

The file in “CSV” format named “products” will be used to store the products’ data.

The file in “CSV” format named “deliveries” will be used to store deliveries’ data.

The file in “CSV” format named “credentials” will be used to store riders’ and employees’ credentials.

# Email Sending:

1. When Account Creation
2. When contact info is updated.
3. When delivery will be assigned to a rider
4. When the product will deliver.
5. When password will change

Sample subject: Password updated successfully

Sample content: Hi, XYZ your password has been updated successfully.

# Project Plan:

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Id** | **Use Case Name** | **Member Name** | **Estimated Completion Date** |
| U01 | Add product | Shakeel Ahmed |  |
| U02 | Update product | Shakeel Ahmed |  |
| U03 | Update name | Shakeel Ahmed |  |
| U04 | Update contact information | Shakeel Ahmed |  |
| U05 | Update Username | Shakeel Ahmed |  |
| U06 | Update password | Ahsan Imtiaz |  |
| U07 | Add delivery | Ahsan Imtiaz |  |
| U08 | View analysis | Ahsan Imtiaz |  |
| U09 | Update name | Ahsan Imtiaz |  |
| U10 | Update contact information | Ahsan Imtiaz |  |
| U11 | Update username | Usman Asghar |  |
| U12 | Update password | Usman Asghar |  |
| U13 | Delivery assignation | Usman Asghar |  |
| U14 | View complete delivery information | Usman Asghar |  |
| U15 | Sign up | Usman Asghar |  |
| U16 | Sign in | Usman Asghar |  |

# Analytical Reports:

* Fuel consumption report (pdf)
* Profit-Loss report (pdf)
* Deliveries report (pdf)