SCHOOL OF COMPUTER SCIENCE & ENGINEERING

Academic year 2022-23

**Open Ended (OE) Design Document**

**On**

**Real World Modeling of Indian Import Export System**

**Object Oriented Programming (20ECSC204)  
Object Oriented Programming Lab (20ECSP203)**

Faculty In-charge:  Prof. K M M Rajashekharaiah

**Team Detail:**

Bibijan A Matte : 01fe21bcs287 (562)

Vaishnavi C G : 01fe21bcs271 (556)

**Problem Definition:**

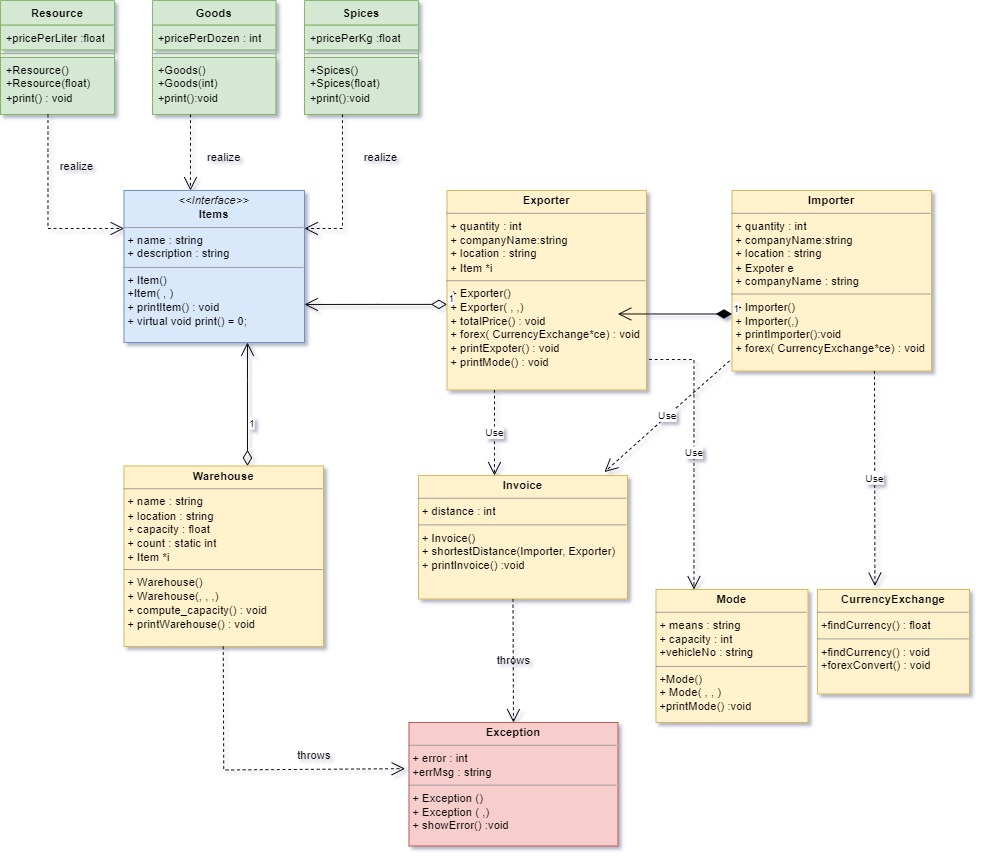
The Indian Import Export System keeps account of the transaction of goods and other supplies that are being imported or exported. Our project is a mini replica of Indian Import Export System. It replicates the process involved in exporting good to the foreign country, forex exchange and payment method with error handling mechanism.

**List Of Class**

* Resource
* Goods
* Spices
* Items
* Warehouse
* Exporter
* Importer
* Modes
* Exception
* CurrencyExchage
* Invoice

**Class Diagram(UML)**

Builder pattern



**Description of Each Class**

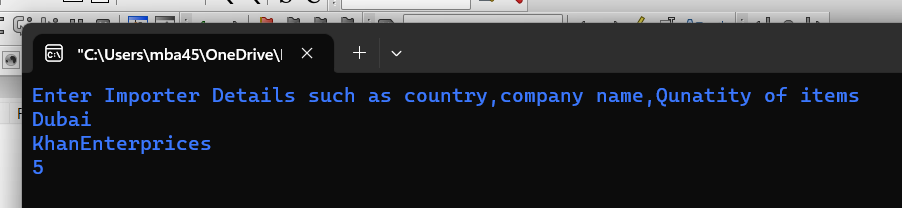
1. Resource: This class represents the resources that are exported. Resources such as Mines, Oil, Crude Oil, Minerals and other item of the same kind. This class is derived class. The function of this class is to print the details of the resources.
2. Goods : This class represents the goods such as timber, fertilizers, agriculture products and other item of the same kind. This class is derived class. The function of this class is to print the details of the goods.
3. Spices: This class represents the spices that are exported. Spices such as Chili, Cardamom, Nutmeg, Black Clove etc. This class is derived class. The function of this class is to print the details of the spices.
4. Items: This class is base class for class Resource, Goods, Spices. This class defines the quantity of the items that are being exported.
5. Warehouse: Warehouse is the class that holds the quantity of items. This class checks for the stocks and its quantity. If the stock of items becomes zero the class throws an exception.
6. Exporter: This class give information of the company that is exporting items. It also gives the count of the items it is exporting. The function of the class is to print the mode of transportation, exporter details. This class calculates the total price of the items.
7. Importer: This class has information of the importer that are the company name, location. This class prints the details of the importer. This class pay the amount to the exporter based on the currency exchange.
8. Modes: This class displays the mode of transport that is being used.
9. Exception: This is exceptional handling class. It prints the exceptions.
10. CurrencyExchange: This class is used to the money conversion, foreign exchange
11. Invoice: This class generates the bill and takes the importer, exporter details from Importer class. Prints the shortest distance between the exporter location to importer location.

**Design Patterns**

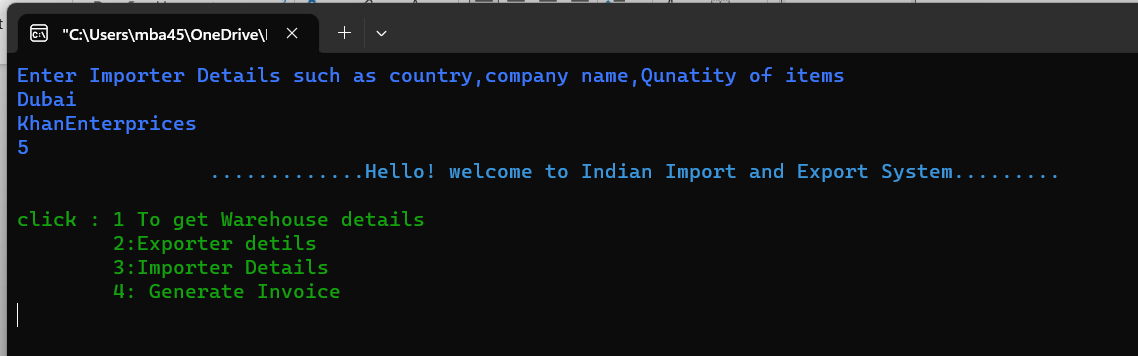
1. Builder Pattern

The builder that we have used is Builder pattern. We have applied this to base class Items. It provides a way to construct the object step by step, allowing the same construction process to create different representations of the object.

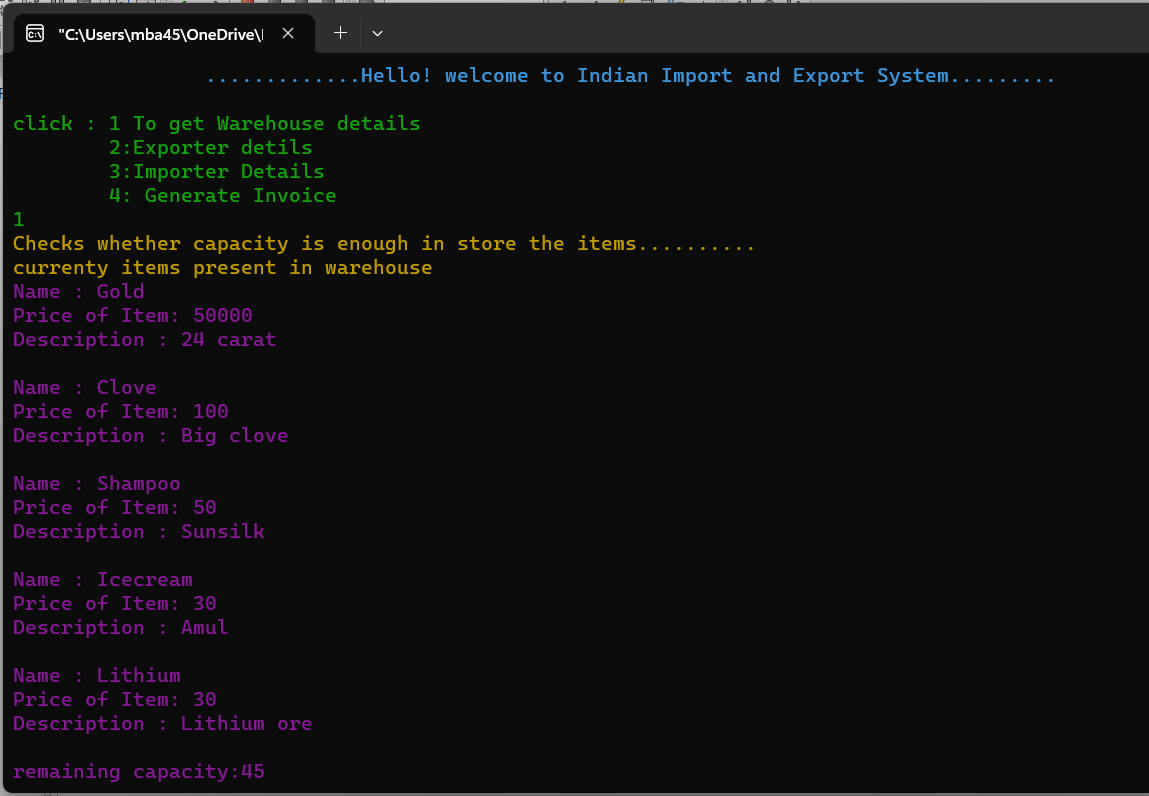
**Implementation**



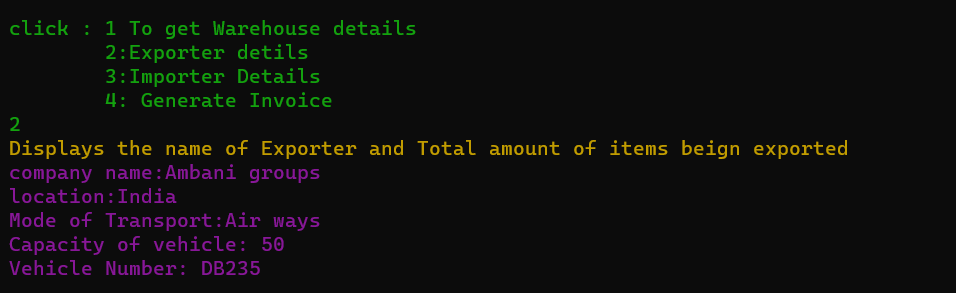
Importer’s location is given by the user



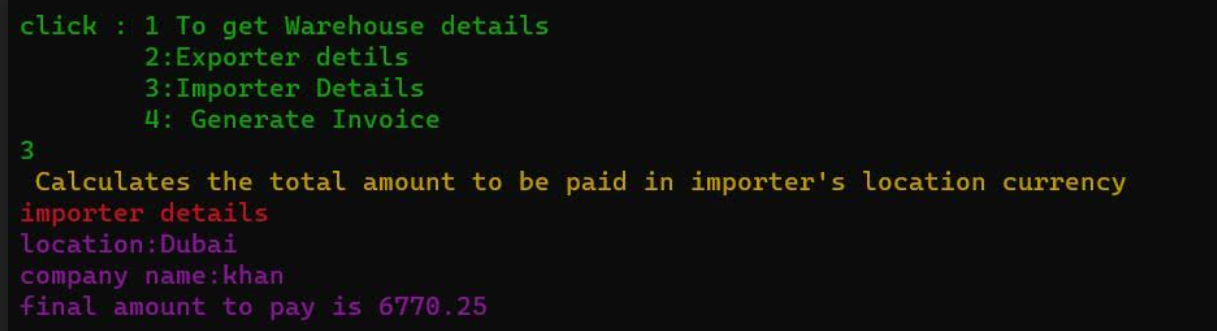
After entering the details, we will get options to choose from



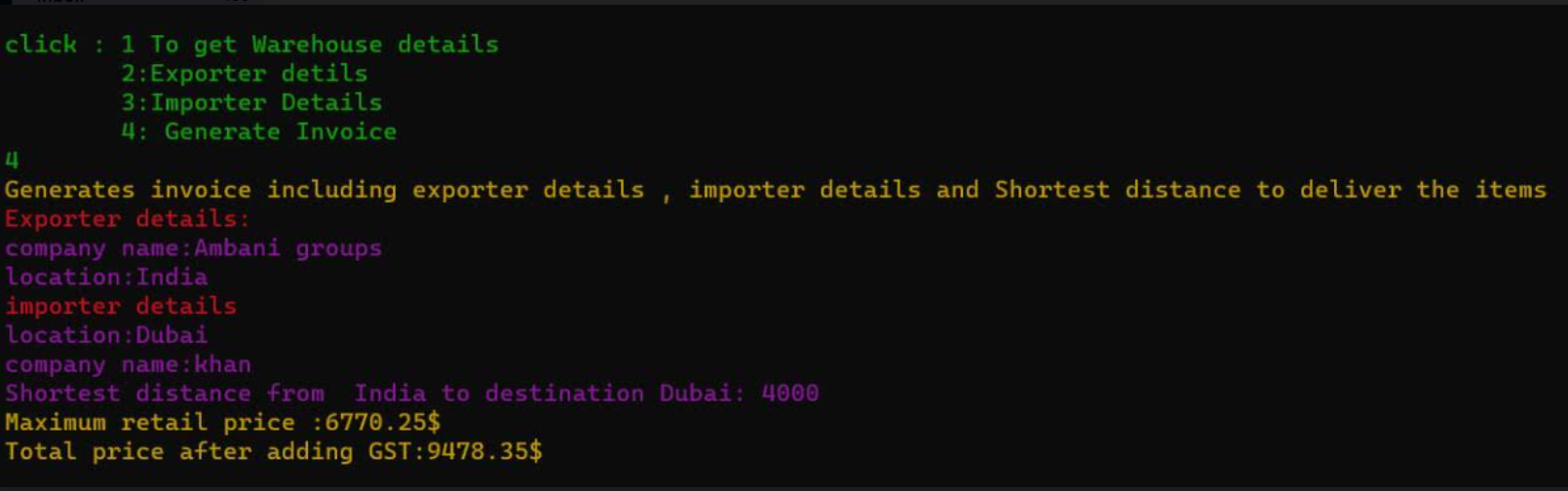
On clicking 1 we get the warehouse details. The original capacity of the warehouse is 50, because the Importer has asked for 5 unit, therefor 50-5=45 is the current capacity of warehouse.



On clicking 2 we get the Exporter details. Also we get to know the means of transport.

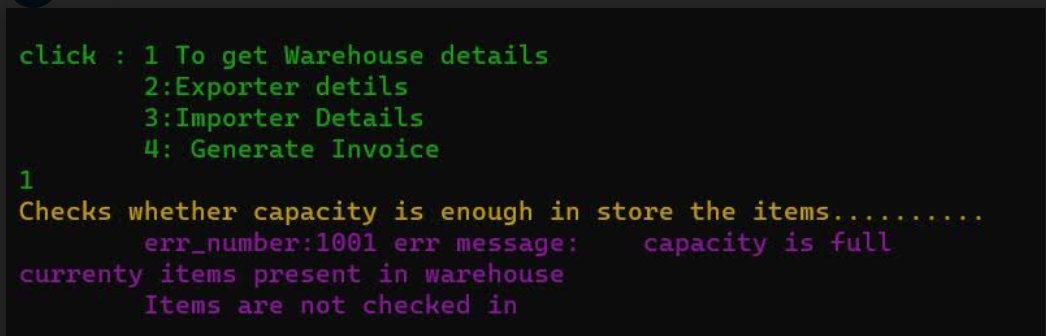


On clicking 3 we get the Importer’s details, including the amount in USD that Importer has to pay to Exporter. This amount payment uses class CurrencyExchange. In class CurrencyExchange the forex conversion is done based on the location of Importer.



On clicking 4 the invoice for the whole process is generated. The invoice has the details of how Importer and Exporter. Also there is a function in class Invoice that calculates the shortest distance between locations of Importer and Exporter. This was implemented using Dijikstra’s Algorithm.

Exception case:



This Exception is thrown by class Warehouse.