

# Programming Project / C++ - Inventory and Order Processor

Student 1: Craciunoiu Petru

Student 2: Balan Alexandra

## I. Task Description

Student 1 is responsible for managing the inventory:

- Adding new products or updating existing stock (based on ID).
- Viewing the inventory contents.

Student 2 is responsible for processing orders:

- Reducing the stock based on ordered quantities.
- Rejecting orders if stock is insufficient or the product doesn't exist.

## II. Data Structures Used by the Team

The following classes will be used:

- Product:

string id, string name, int quantity

- Inventory:

Aggregates a list of Product objects and provides logic for modification and access.

## III. File Structure

The following files will be used:

data.txt

This is the shared file where all product data is stored. Each line represents one product:

<id>,<name>,<quantity>

Example:

001,Apple,10

002,Banana,20

## IV. Interacting with Executables

There are two applications, each with different purposes. All input is given through command line arguments, no cin is used.

### Application 1 - main\_inventory

Command:

./main\_inventory <file> <id> <name> <quantity>

Example:

./main\_inventory data.txt 001 Apple 10

Functionality:

- Adds a new product to the inventory, or updates its quantity if it already exists.

### Application 2 - main\_order

Command:

./main\_order <file> <id> <quantity>

Example:

./main\_order data.txt 001 3

Functionality:

- Attempts to process an order by reducing stock.

- Rejects order if stock is insufficient or ID does not exist.

## **V. Code Structure**

The project is divided into the following source files:

- Product.hpp/cpp: Contains definition and implementation of the Product class
- Inventory.hpp/cpp: Contains Inventory class that manages the stock
- main\_inventory.cpp: Entry point for Student 1's responsibilities
- main\_order.cpp: Entry point for Student 2's responsibilities
- data.txt: Shared product database