CAB302 Project: Inventory Management Application

By:

Alexander Rozsa: n9992529

Liam Edwards: n\*\*\*\*\*\*\*

2018

# **Technical Description**

The aim of this project was to have a GUI-based application that takes defining back-end classes and displays outputs based on their values. The values needed to be displayed, of course, are the items contained within the “store” and their quantities as a means of managing the store’s inventory. The below diagram is a representation of the program’s classes and their dependencies ordered from the main class (Interface) down to the class which doesn’t have any dependencies (Item). Interface depends on Store to hold and provide the data of the store, and it also depends on Manifest to hold and provide the data of a manifest for store restocking. Manifest relies on Trucks since it’s a list of Trucks and that’s the object type it’s supposed to contain. Store and Trucks rely on Stock in order to store Item(s) and their quantities. Finally, Stock relies on Item to have many of them stored within it providing data to the program, whereas Item doesn’t have any dependencies; it just supplies data. IOHandler on the other hand doesn’t have any dependencies; it provides publicly available functions for importing and exporting data.

Interface

IOHandler

Item

Stock

Manifest

Trucks

**A**

= **A** has a direct

Store

**B** dependency on **B**

(contains a constructor

of that class).

Mention how every class works making mention to EVERY OOP concept.

# **GUI Test Report**