

Flower Shop Manager

“Flower Shop Manager” is an application that helps run a small flower shop. The functions of the application will include creating and managing inventory, a point of sale to perform a sale or a return and creating and retrieving transactions from accounts history.

High-level Requirements:

Menu

1. Create a main menu with 4 options to select – Items, Register, Accounts, and Quit
2. Create a sub-menu for Items – View Item, Update Item, Add Item, and Quit
3. Create a sub-menu for Register – Perform a sale, Perform a return, and Quit
4. Create a sub-menu for Accounts – Display Sales & Returns total, and Quit

Functionality – Items

1. User should be able to create new items using the menu option
2. User will be providing the details for the new item – Name, Color, Cost, Price, Sale pack quantity, and Inventory
3. User should be able to view all items and update any user defined item attributes but not an item id, which is auto-generated.

Functionality – Register

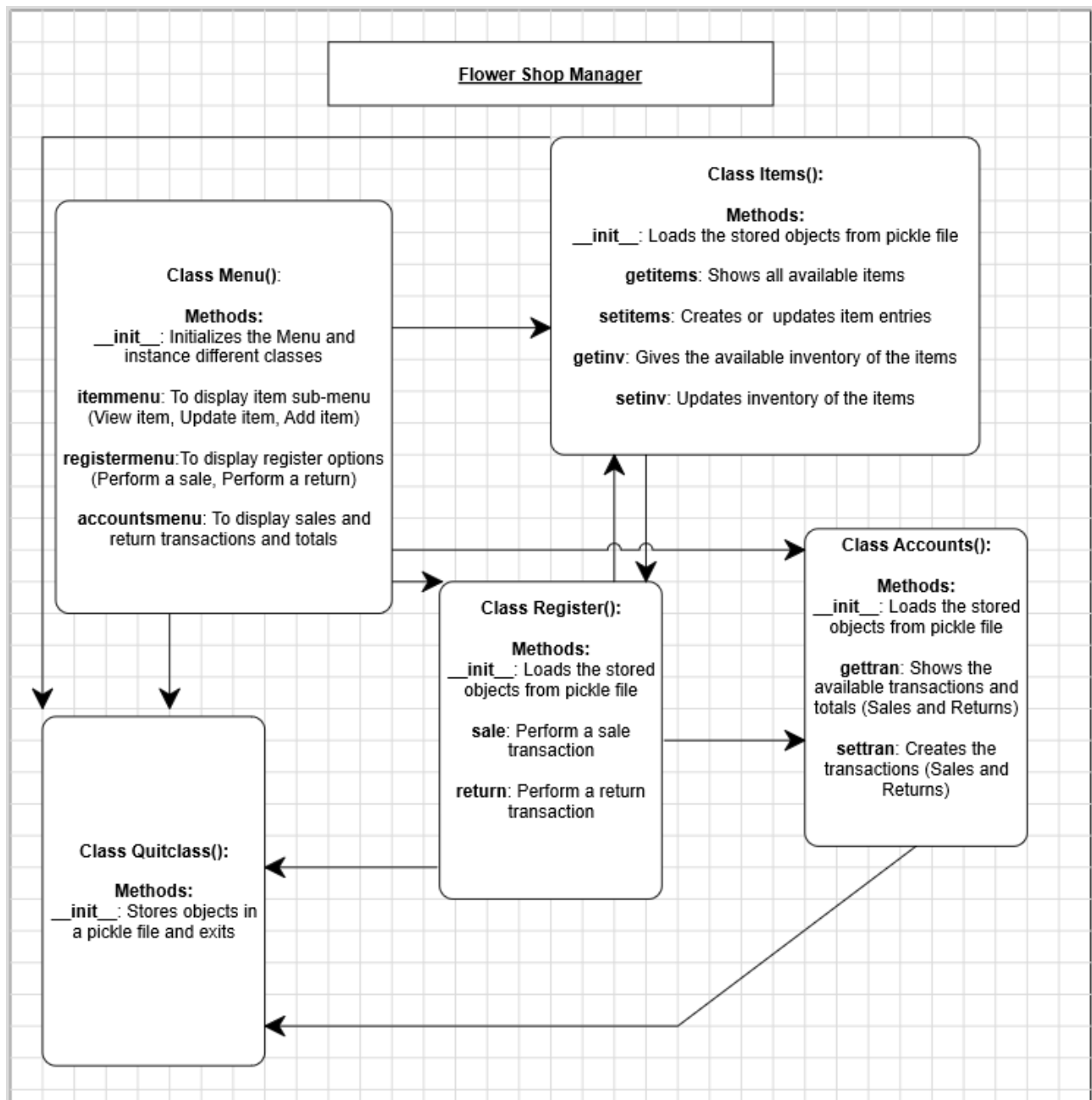
1. User should be able to perform a sale transaction by entering the Item ID and Quantity.
 - a. Note: All items will be labelled with the item ID and a barcode, so the user is expected view and enter the item id during the sale
2. User should be able to perform a return transaction by entering the Item ID and Quantity.
 - a. Note: All items will be labelled with the item ID and a barcode, so the user is expected view and enter the item id during the return. If a barcode is missing, user can view item id from View item screen.
3. The return transaction need not be linked to the original sale transaction. Returns are based on original receipt and goodwill and can be performed even without an original sale transaction. Return inventory is not updated as item may not be saleable.

Functionality – Accounts

1. User should be able to view all transactions (sales and returns) from accounts history

Functionality – Quit

1. If the user quits from any menu or a sub-menu, the completed transactions will be stored and can be accessed when the application is restarted



Menu – Class

This class will display the main menu and the sub-menus depending on the user option selection

__init__ - Method

This method will initialize the main menu display and captures a user option input. Other methods are initiated based on the user input

itemmenu – Method

This method will display the item sub-menu. This method will capture a user option input and will initiate the Items class and corresponding methods

registermenu – Method

This method will display the register sub-menu. This method will capture a user option input and will initiate the Register class and corresponding methods

accountsmenu – Method

This method will display the accounts sub-menu. This method will capture a user option input and will initiate the Accounts class and corresponding methods

Items – Class

This class will be used to create, update and display items and its attributes

__init__ - Method

This method will load stored objects from a pickle file

getitems - Method

This method will get all the items and its attributes that are stored

setitems - Method

This method will create new or update existing items with item attributes. Inputs required – Name, Color, Cost, Price, Sale pack quantity, and Inventory

getinventory – Method

This method will be used by the Register class to fetch inventory details during the sale

setinventory – Method

This method will be used by the Register class to update inventory after a sale

Register – Class

This class will be used to perform a sale or a return and update inventory and accounts.

__init__ - Method

This method will load stored objects from a pickle file

Sale - Method

This method will create a sale transaction, update Accounts class, update inventory in Items class. Inputs are Item ID and quantity.

Return - Method

This method will create a return transaction, update Accounts class and **does not** update inventory in Items class.

Account - Class

This class will be used to create sales and returns transactions and view the history.

__init__ - Method

This method will load stored objects from a pickle file

gettran - Method

This method will get all available transactions from history

settran - Method

This method will create a sale or a return transaction into accounts history. Inputs are item id, quantity, transaction type and total

Quitclass - Class

This class will save required objects in a pickle file and exits the application