Requirements analysis documentation:

- **Main OOP concepts (classes, inheritance, and polymorphism):** We have one base class called *MenuItem*, three child classes called *Cashier*, *Inventory* and *Report* and one independent *book* class that all the other classes use. In the base class we have a virtual function called *DisplayMenu* and we used polymorphism to display the main menu and the different menus of the child classes.
- **Friends and operator overloading:** We have two friend functions on the book class that overload the insertion and extraction operators.
- **Exception Handling:** In the *book* class, for the member functions that set the quantity, wholesale value, and retail value, we throw an exception if the user enters a negative number and when the user inserts a value of the incorrect data type. We used exception handlers so that way the user can enter a valid data and the program would not have to terminate. We have another exception in *main*; if the program cannot allocate any more memory for the vector of books it will display an error message and terminate the program.

Pseudocode

```
1) Open an input file that holds the inventory of books
2) If the input file was found continue the program, else tell the user that it was not found and terminate the program
3) Read all the books from the input file and storage them in a vector
4) Create an object of the base class and the child classes
5) Create a pointer called menuitemptr of the base class type
6) Point the menuitemptr to the object of the base class and call the displaymenu virtual function
        6.1) Show the main menu
        6.2) Get the user selection for the module they want to use
        6.3) Validate the input
        6.4) Return the choice to main
7) If the user selects 1 (Cashier module)
a) Point the menuitemptr to the object of the cashier class and call the displaymenu function
b) Show the main menu for the cashier module, get the choice of the user and return it to main
8) If the user selects 1 (Add a book by title)
-Call the cashier member function to add a book by title
-Ask for the title of the book
-Search it on the vector
-If it was found
        -Ask the user how many books he/she wants
        -Take the quantity of the books the user wants out of the vector of books
-Else tell the user that it was not found
-Ask the user if he/she would like to buy another buy
        -if he/she say yes
        -show the cashier menu
    -else print the receipt
9) If the user selects 2 (Add a book by ISBN)
-Call the cashier member function to add a book by ISBN
-Ask for the ISBN of the book
-Search it on the vector
-If it was found
    -Ask the user how many books he/she wants
    -Take the quantity of the books the user wants out of the vector of books
-Else tell the user that it was not found
-Ask the user if he/she would like to buy another buy
-If he/she say yes
    -show the cashier menu
-else print the receipt
10) If the user selects 3 print the receipt
11) If the user selects 4 go back to the main menu
12) If the user selects 2(Inventory Module)
a) Point the menuitemptr to the object of the inventory class and call the displaymenu function
b) Show the main menu for the inventory module, get the choice of the user and return it to main
13) If the user selects 1 (Look up a book)
-Call the inventory member function to look up a book
-Ask the user how to look for the book
-If the user wants to search by ISBN
```

```
        -Call the member function of the base class that sorts the vector of books respect the ISBN
        -Call the member function of the base class that search the ISBN the user inserted
        -If the book was found display the information
        -Else tell the user that the book was not found
Else if the user wants to search by Title
        -Call the member function of the base class that sorts the vector of books respect the ISBN
        -Call the member function of the base class that search the ISBN the user inserted
        -If the book was found display the information of the book
        -Else tell the user that the book was not found
-Ask the user if he/she would like to look for another book
-If they want to look for another book call the look up a book function again else return to the inventory main menu
14) If the user selects 2 (Add a book)
-Ask the user for the information for the new book
-Add the new book at the end of the vector
-Ask the user if he/she would like to add another book
-If the user wants to add another book call this function again else return to the inventory main menu
15) If the user selects 3(Edit a book)
-Ask the user how he/she would like to look for the book he/she wants to edit by ISBN or Title
-If the user wants to look by ISBN
        -Ask for the ISBN
        -Call the member function of the base class that sorts the vector by ISBN
        -Call the member function of the base class that search the ISBN the user inserted
```

-If the book is found in the vector, display its information else tell the user

-Ask the user if the book that was found is the one that he/she would like to edit

-If the user wants to edit the book that was found, ask for the new information

-Ask the user if he/she would like to edit another book

-If the user wants to add a new book call this function again else send it to the inventory main menu

-Else if the user wants to look by title

-Ask for the title

-Call the member function of the base class that sorts the vector by title

-Call the member function of the base class that search the title the user inserted

-If the book is found in the vector, display its information else tell the user

-Ask the user if the book that was found is the one that he/she would like to edit

-If the user wants to edit the book that was found, ask for the new information

-Ask the user if he/she would like to edit another book

-If the user wants to add a new book call this function again else send it to the inventory main menu

16) If the user selects 4 Delete a book

-Ask the user how he/she would like to look for the book he/she wants to delete by ISBN or Title

-If the user wants to look for the book by ISBN

-Ask for the ISBN

-Call the member function of the base class that sorts the vector by ISBN

-Call the member function of the base class that search the ISBN the user inserted

-If the book is found in the vector, display its information else tell the user

-Ask the user if the book that was found is the one that he/she would like to delete

-If the user wants to delete the book that was found

-If the book is in the last position of the vector delete the last position

-Else, copy the information of the book that is in the last position of the vector in the position of the book that is going to be deleted and then erase the last position

-Ask the user if he/she would like to delete another book

-If the user wants to delete another book, call this function again else return to the inventory main menu

-If the user wants to look for the book by title

-Ask for the title

-Call the member function of the base class that sorts the vector by title

-Call the member function of the base class that search the title the user inserted

-If the book is found in the vector, display its information else tell the user

-Ask the user if the book that was found is the one that he/she would like to delete

-If the user wants to delete the book that was found

-If the book is in the last position of the vector delete the last position

-Else, copy the information of the book that is in the last position of the vector in the position of the book that is going to be deleted and then erase the last position

-Ask the user if he/she would like to delete another book

-If the user wants to delete another book, call this function again else return to the inventory main menu

17) If the user selects 5 Return to main menu

18) If the user selects 3 (Report module)

a) Point the *menuitemptr* to the object of the report class and call the *displaymenu* function

b) Show the main menu for the report module, get the choice of the user and return it to main

19) If the user selects 1 display the inventory listing report and return to the report main menu

20) If the user selects 2

-Calculate the wholesale value

-Display the inventory by wholesale value and the total wholesale value

-Return to the report main menu

21) If the user selects 3

-Calculate the retail value

-Display the inventory by retail value and the total retail value

-Return to the report main menu

22) If the user selects 4

-Sort the vector by quantity in decreasing order

-Display the Listing of the Inventory by quantity

-Return to the report main menu

23) If the user selects 5

-Sort the vector by cost in decreasing order

-Display the Listing of the Inventory by cost

-Return to the report main menu

24) If the user selects 6

-Sort the vector by date in decreasing order

-Display the Listing of the Inventory by date added

-Return to the report main menu

25) If the user selects 7

-Return to the main menu

26) If the user selects 4 Exit the program