

01

# PROJECT PROPOSAL

Inventory Management System

---

**Tammy Sujaritchai**

10544323

---

**CS 2 - Introduction to Computer Science  
Honors Project**

02

# Online Shopping Platform

---

Given prompt

[Back to Home Page](#)

03

# Objective

- Create an online inventory management program
  - Add/remove items
  - Add/remove employee info
  - View stock

Next: Persona



04

# Persona



## **Rachel (20)**

Online clothing business owner

---

Pain point:

- No management experience
- Tired of using different apps for different task
- Can't keep tracks pf employees



## **John (58)**

Retired Walmart Manager

---

Pain point:

- Not tech savvy
- Struggle to keep record of inventory



05

# Attempts

Total attempts: 4

```
1 #include <iostream>
2 using namespace std;
3
4 void produceGoods(int);
5 void electronicGoods();
6 void MenuOfAction();
7 void add();
8 void viewCart();
9 string cart[] = {};
10 int choice;
11
12
13 ▼ int main() {
14
15     cout << "Welcome to Metamart!\nWhat would you like to buy\n";
16     cout << "\t1. Produce\n";
17     cout << "\t2. Electronics\n";
18     cout << "\t3. Personal Care\n";
19     cout << "\t4. Exit\n" << "Your choice: ";
20     cin >> choice;
21
22 ▼ if(choice == 1){
23     cout << "Please enter the category of the product\n";
24     cout << "\t1. Fruits\n" << "\t2. Veggies\n" << "\t3. Dairy p\n";
25     cin >> choice;
26     produceGoods(choice);
27 }
28 ▼ else if(choice == 2){
29     electronicGoods();
30 }
31 }
32
33 }
34 ▼ void produceGoods(int x){
35     string fruits[] = {"Apple", "Orange", "Grape", "Durian",
36     int amountOfFruits[] = {5, 4, 6, 4, 2}; //Can I use zip()
37     string veggies[] = {"Pepper", "Tomato", "Onion", "Cabbage",
38     int amountOfVeggies[] = {3, 7, 4, 2};
39     string dairyProduct[] = {"Cheese", "Milk", "Yoghurt", "R"}
```

```
f main
21 ▼ class Product{
22     public:
23         string name;
24         int amount, price;
25         vector<string> cart;
26
27 ▼ void add(string x){
28     cout << "
29     cart.push_back(x);
30 }
31 Product produce;
32 produce.name = "carrot";
33 produce.amount = 6;
34 produce.price = 3;
35
36 };
37
38 ▼ int main(){
39     int choice, i = 0;
40     formatOutput();
41     cout << "Welcome to Metamart inventory managment system" << endl;
42
43 ▼ while (i != -1){
44     printMenu();
45     cin >> choice;
46 ▼ if(choice == 1){
47         //add item
48         add();
49     }
50 ▼ else if(choice == 2){
51         //remove item
52         remove();
53     }
54 ▼ else{
55         //manage staff
56         manageStaff();
57     }
58     cout << "\nEnter -1 to exit, else enter any number to continue\n";
59     cin >> i;
60 }
```

06

# Final program

Tools: C++ (vector, array, map, etc.)



[Click here to try](#)

```
1  #include <iostream>
2  #include <vector>
3  #include <map>
4  #include <iomanip>
5  using namespace std;
6
7  const int WIDTH = 12;
8  const char SPACE = ' ';
9
10 void printMenu();
11 void add();
12 void remove();
13 void printAllItem(); //Print all the item
14 void formatOutput(); //Format all output
15 void manageStaff(); //manage staff
16 void addStaff();
17 void removeStaff();
18 void printAllStaff();
19
20 vector<string> item;
21 vector<int> quantity;
22 vector<double> price;
23 map<string, double> staff;
24 //Array implementation
25 string menu[5] = {"Add item", "Delete item", "Print all item", "Print all staff", "Exit"};
26
```

## Add item

```
95 ▼ void add(){
96     string name;
97     int num;
98     double cost;
99
100     cout << "\nEnter the name of the item" <
101     cin.ignore(); //so it won't skip getline
102     getline(cin, name); //in case the name i
103     item.push_back(name);
104
105     cout << "\nEnter the quantity of the ite
106     cin >> num;
107     quantity.push_back(num);
108
109     cout << "\nEnter the price of the item"
110     cin >> cost;
111     price.push_back(cost);
112
113     char x;
114     cout << "\nContinue? [Y/N]" << endl;
115     cin >> x;
116 ▼ if (x == 'Y' || x == 'y'){
117     add();
118 }
119 ▼ else {
120     printAllItem();
121 }
122 }
```

## Remove item

```
123 ▼ void remove(){
124     int object;
125     //Print out all items for user to cho
126     cout << "\nAll item: \n";
127 ▼ for(int i = 0; i < item.size(); i++){
128     cout << i+1 << ". " << item.at(i) <<
129 }
130
131     cout << "Enter the number of item you
132     cin >> object;
133     int x = object - 1;
134
135     item.erase(item.begin() + x);
136     quantity.erase(quantity.begin() + x);
137     price.erase(price.begin() + x);
138
139     char choice;
140     cout << "\nContinue? [Y/N]" << endl;
141     cin >> choice;
142 ▼ if (choice == 'Y' || choice == 'y'){
143     remove();
144 }
145 ▼ else{
146     cout << "\n";
147     printAllItem();
148 }
149 }
```

## View all items

```
151 ▼ void printAllItem(){
152
153     //check if vector is empty
154 ▼ if (item.empty()){
155     cout << "No item available" << endl;
156 }
157 ▼ else{
158     cout << "item" << setw(17) << setfill(SPACI
159     << setw(13) << setfill(SPACE) << "price" <
160 ▼ for(int i = 0; i < item.size(); i++){
161     cout << item.at(i) << setw(WIDTH) << set
162     << setw(WIDTH) << setfill(SPACE) << "$" .
163 }
164 }
165 }
166 }
```

## Add item

1. Enters 1 in the menu option
2. Enter the name, quantity, and price of the item.
3. Enter 'Y' to continue, 'N' to end the action
4. If yes, the process is repeated.  
Else, all items are printed out.

## Remove item

1. Enters 2 in the menu option
2. Enter the number of items they want to remove from list (ex: 1 for 1. Banana)
3. The item is removed
4. Enter 'Y' to continue, 'N' to end and print out all items

## View all items

1. Enters 3 in the menu option
2. The item's name, quantity, and price are printed out

**Problems keeping records of items:**

**solved**





## Add staff

```
197 ▼ void addStaff(){
198     string name, input;
199     double salary;
200
201     //Get staff's name and salary
202     cout << "\nEnter staff name: "
203     cin.ignore();
204     getline(cin, name);
205     cout << "Enter his/her annual
206     cin >> salary;
207     staff[name] = salary;
208
209     char choice;
210     cout << "\nContinue? [Y/N]" <<
211     cin >> choice;
212 ▼ if (choice == 'Y' || choice ==
213     addStaff();
214 }
215 ▼ else{
216     printAllStaff();
217 }
218
219 }
```

## Remove staff

```
//Remove staff along with
▼ void removeStaff(){
    string name;

    //Print our list of staff
    printAllStaff();

    cout << "Enter the name of staff to be removed: "
    cin >> name;
    staff.erase(name);

    //Print our list of staff
    printAllStaff();
}
```

## View all staff

```
//Print out all the staff's names
▼ void printAllStaff(){
    //check if vector is empty
▼ if (staff.empty()){
    cout << "No staff registered\n";
}
▼ else{
    cout << "\nStaff name\t\tSalary\n";
▼ for (auto i = staff.begin(); i != staff.end(); i++){
    cout << (*i).first << "\t" << (*i).second << "\n";
}
▼ /*while(staff.begin() != staff.end()){
    cout << staff.begin()->first << "\t" << staff.begin()->second << "\n";
    staff.erase(staff.begin());
}*/
    cout << "\nTotal staff: " << staff.size() << "\n";
}
```

## Add staff

1. Enters 3, then 1
2. Enter the name and annual salary.
3. Enter 'Y' to continue, 'N' to end the action
4. If yes, the process is repeated. Else, all staffs are printed out.

## Remove staff

1. Enters 3, then 2
2. Enter the number of staff they want to remove from list (ex: 1 for 1. Bill)
3. The staff is removed
4. Enter 'Y' to continue, 'N' to end and print out all staffs

## View all staff

1. Enters 3, then 3
2. All staffs names and salaries are printed out

**Problems with managing employee:**

**solved**



# Maintenance & Updates

Maintenance	Future updates
<ul style="list-style-type: none"><li>• No major maintenance plan required</li><li>• Verify the version of C++ and the compatibility of the program in each device</li></ul>	<ul style="list-style-type: none"><li>• Implement more features<ul style="list-style-type: none"><li>◦ log of changes</li><li>◦ real-time inventory management</li><li>◦ role-based access</li><li>◦ data storage in external files</li></ul></li></ul>



08

# Resources

## & Credits

- <https://asana.com/resources/project-scope>
- <https://www.tutorialspoint.com/how-to-print-out-the-contents-of-a-vector-in-cplusplus>
- <https://stackoverflow.com/questions/15079057/arrays-vs-vectors-introductory-similarities-and-differences>
- <https://iq.opengenus.org/ways-to-remove-elements-from-vector-cpp/>
- <https://www.educative.io/answers/how-to-iterate-through-a-vector-in-cpp>
- <https://stackoverflow.com/questions/15151480/simple-dictionary-in-c>
- <https://cplusplus.com/reference/map/map/empty/>
- <https://www.quora.com/What-are-some-alternatives-for-if-statements-in-C++>
- <https://stackoverflow.com/questions/14765155/how-can-i-easily-format-my-data-table-in-c>
- <https://stackoverflow.com/questions/21567291/why-does-stdgetline-skip-input-after-a-formatted-extraction>

Special thanks to Professor Balajadia

[Link to project specification](#)

# Thank you!

---

Do you have any questions?