```cpp
// !Menu.h
#pragma once
#include "iostream"
#include "vector"
#include "MenuItem.h"
class Menu {
private:
std::string title;
std::vector<MenuItem*> menuItems;
public:
explicit Menu();
Menu(std::string name);
~Menu();
void append(MenuItem* menuItem);
void print();
void setName(std::string name);
std::string getName();
void awaitUserInput();
};

// !Menu.cpp
#include "Menu.h"
Menu::Menu() {}
Menu::Menu(std::string name) {
this->title = name;
menuItems = {};
}
Menu::~Menu() {
for (auto item : this->menuItems) {
delete item;
}
menuItems.clear();
}
void Menu::append(MenuItem* menuItem) {
menuItems.push_back(menuItem);
}
void Menu::print() {
std::cout << this->title << std::endl;
for (auto menuItem : menuItems) {
std::cout << menuItem->getId() << ") "
<< menuItem->getTitle() << std::endl;
}
}
void Menu::setName(std::string name) {
this->title = name;
}
std::string Menu::getName() {return this->title;
}

void Menu::awaitUserInput() {
this->print();
int input = 0;
std::cout << " >> Select: "; std::cin >> input;

for (auto menuItem : this->menuItems) {
if (menuItem->getId() == input) {
menuItem->executeCallback();
awaitUserInput();
}
}
std::cout << "!Incorrect selection!" << std::endl;
awaitUserInput();
}
```