PRODUCT MANAGEMENT SYSTEM

GROUP APOLLO PROJECT
FINAL PROPOSAL

Members:
Daquillaña, Lue
Dionisio, Jarred
Jimenez, Nicole Andrei

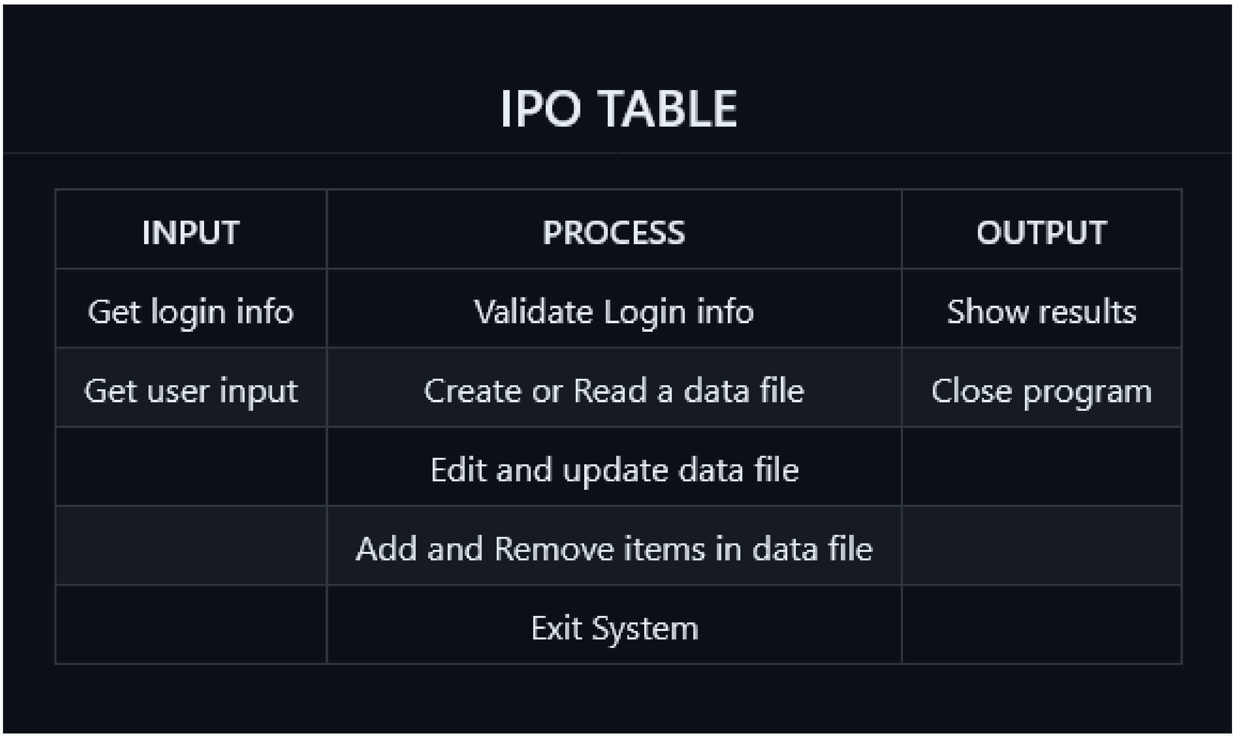
I. PROJECT DESCRIPTION:

A Product management system is an essential software that helps companies manage their products with ease. It is designed to keep track of product details, pricing, inventory, and sales. The system is protected by secure login information, which ensures that only authorised users can access the system.

Once logged in, the user can access various options that enable them to manage products efficiently. For example, the user can add new product details, update existing products, and view a comprehensive list of all products available. The system can also generate reports that provide an overview of the company's products, such as inventory reports and pricing reports.

DEMO on YOUTUBE CHANNEL: https://youtube.com/@projectapollo_comprog

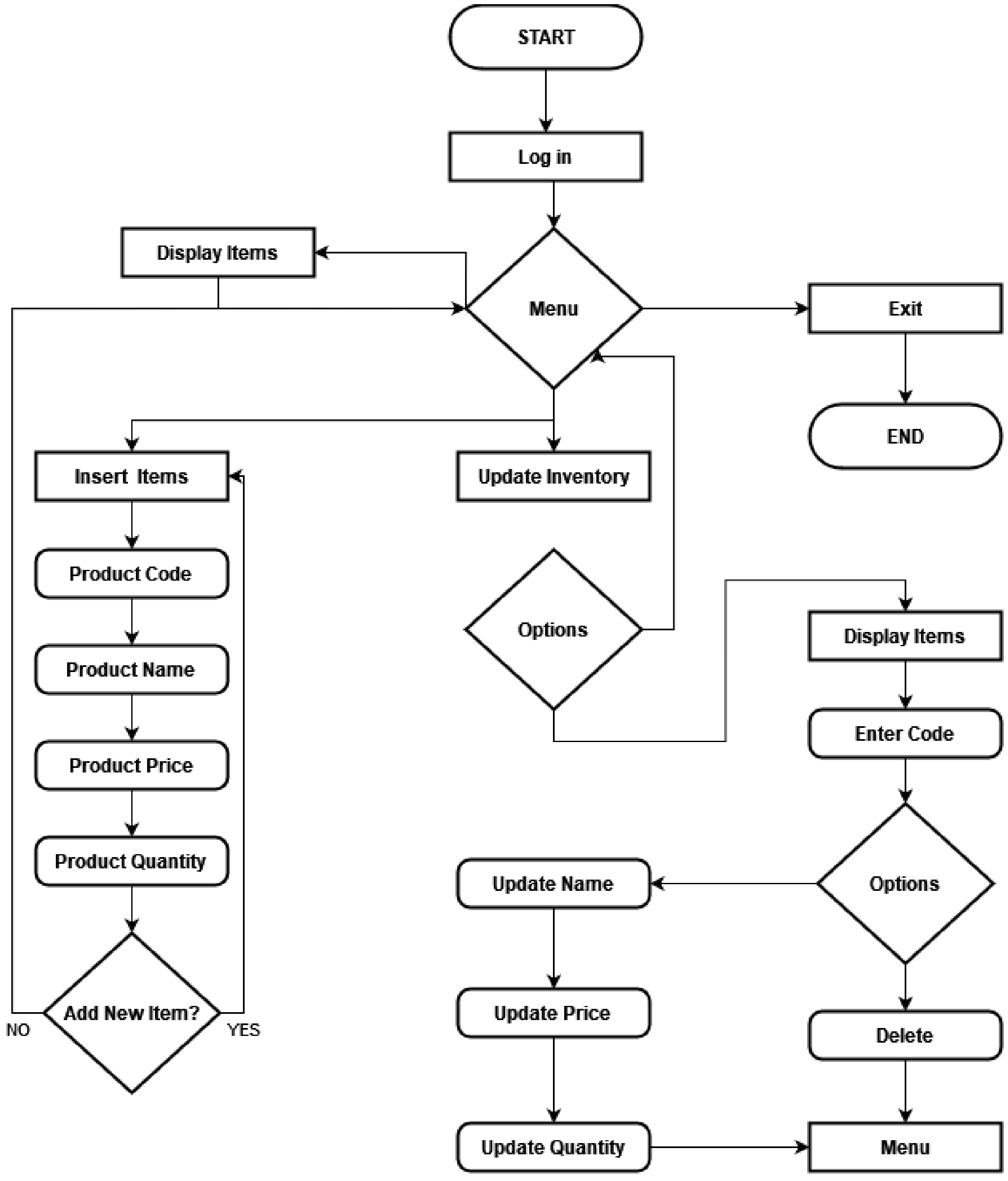
II. IPO TABLE:



IPO TABLE

INPUT	PROCESS	OUTPUT
Get login info	Validate Login info	Show results
Get user input	Create or Read a data file	Close program
	Edit and update data file	
	Add and Remove items in data file	
	Exit System	

III. METHODOLOGY:



IV. SOURCE CODE:

```
// HEADER FILES
#include <stdio.h>
#include <stdlib.h>

// VARIABLES
#define USERNAME "user"
#define PASSWORD "pass"

// NEW DATA TYPE
typedef struct items {
        char product_code[20];
        char product_name[20];
        int price;
        int quantity;
} ITEM;

ITEM item;

// Checks if the product code is available.
int isProdCodeAvail(char code[]) {
        FILE *file;
        file = fopen("Inventory.txt", "r");
        while (!feof(file)) {
        fread(&item, sizeof(item), 1, file);
        int i = 0;
        while (code[i] != '\0' && item.product_code[i] != '\0') {
        if (code[i] != item.product_code[i]) {
                break;
        }
        i++;
        } if (code[i] == '\0' && item.product_code[i] == '\0') {
        fclose(file);
        return 1;
        }
        }
        fclose(file);
        return 0;
```

```
}

// Checks if the product name is available.

int isProdNameAvail(char name[]) {
        FILE *file;
        file = fopen("Inventory.txt", "r");
        while (fread(&item, sizeof(item), 1, file)) {
        int i = 0;
        while (name[i] != '\0' && item.product_name[i] != '\0' && name[i] == item.product_name[i]) {
        i++;
        } if (name[i] == '\0' && item.product_name[i] == '\0') {
        fclose(file);
        return 1;
        }
        }
        fclose(file);
        return 0;
}

// Checks if the input is an integer.
int getinteger(int input) {
        int result = scanf("%d", &input);
        while (result != 1 || input <= 0) {
        if (result != 1) {
        scanf("%*[^\n]");
        }
        printf("\n You've entered an invalid input.");
        printf("Please try again: ");
        result = scanf("%d", &input);
        }
        return input;
}

// DISPLAY
// displays the list of items.
void display() {
        printf("Available Products:\n");
        FILE *file;
        int count = 0;
        file = fopen("Inventory.txt", "rb");

        if (file == NULL) {
```

```
    printf("\tNo Product is inserted.\n");
    void options();
    return;
    }

    while (fread(&item, sizeof(item), 1, file)) {
    count++;
    }

    if (count > 0) {
    rewind(file); // reset file pointer to beginning of file
    printf("+--------+-----------------+--------+----------+\n");
    printf("|  CODE  |       NAME        | PRICE  | QUANTITY |\n");
    printf("+--------+-----------------+--------+----------+\n");

    while (fread(&item, sizeof(item), 1, file)) {
    printf("|%8s|%17s|%8d|%10d|\n", item.product_code, item.product_name, item.price,
item.quantity);
    }

    printf("+--------+-----------------+--------+----------+\n");
    } else {
    printf("\nNo products listed. Please add products first.\n");
    }
    fclose(file);
}

// ADD ITEMS
// Creates a new item in the data file.
void addItem() {
    printf("Add Item: \n");

    FILE *file;
    char code[20];
    char x[4] = {0};
    int a, code_length = 0;
    file = fopen("Inventory.txt", "ab");

    printf("\nEnter \"end\" to exit.");

    printf("\nEnter Product code: ");
    scanf("%s", code);

    while (code[code_length] != '\0') {
```

```
    code_length++;
    } if (code[0] == 'e' && code[1] == 'n' && code[2] == 'd' && code[3] == '\0') {
    printf("\nReturning to options...\n");
    void options();
    return;
    }

    int available = isProdCodeAvail(code);
    if (available) {
    printf("\n* Product is already in the inventory.\n");
    fclose(file);
    system("pause");
    system("cls");
    return;
    } for (int i = 0; i < code_length; i++) {
    item.product_code[i] = code[i];
    } item.product_code[code_length] = '\0';

    printf("Enter Product Name: ");
    scanf("%s", item.product_name);

    printf("Enter Product price: ");
    item.price = getinteger(item.price);

    printf("Enter Quantity: ");
    item.quantity = getinteger(item.quantity);

    fwrite(&item, sizeof(item), 1, file);
    fclose(file);

    int choice;
    printf("\nEnter 1 to add another item or any other integer to return to options: ");
    scanf("%d", &choice);

    if (choice == 1) {
    addItem();
    } else {
    printf("\nReturning to options...\n");
    system("cls");
    void options();
    return;
    }
}
```

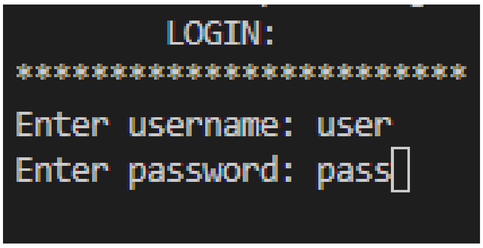
```
// UPDATE ITEMS
// updates and/or deletes an item on the data file.
void update() {
	printf("\tUpdate Product\n");
	printf("************************\n");

	FILE *file1, *file2;
	char code[20], product[20];
	int available, choice, productCount = 0;

	file1 = fopen("Inventory.txt", "rb");
	if (file1 != NULL) {
	while (fread(&item, sizeof(item), 1, file1)) {
	productCount++;
	}
	fclose(file1);
	}

	if (productCount == 0) {
	printf("\nThere are no products listed. Please add an item first.\n");
	system("pause");
	system("cls");
	return;
	}

	do {
	system("cls");
	int option;

	printf("Choose an option: \n");
	printf("1. Display Inventory\n");
	printf("2. Go back to main interface.\n");
	scanf("%d", &option);


	if (option == 1) {
	display();
	printf("\nEnter the Product code to update or delete from the Inventory: ");
	scanf("%s", code);
	available = isProdCodeAvail(code);
	} else if (option == 2) {
	system("cls");
	return;
	} else {
    printf("\nInvalid option. Please try again.\n");
    return;
    }

    if (available == 0) {
    printf("\nNo product found for update or delete.\n");
    printf("Press 1 to try again, or any other key to go back to the main menu.\n");
    int retryChoice;
    scanf("%d", &retryChoice);
    if (retryChoice != 1) {
        return;
    }
    } else {
    do {
        printf("\nWhat do you want to do with the product?\n");
        printf("1. Update\n");
        printf("2. Delete\n");
        printf("Enter your choice (1-2): ");
        scanf("%d", &choice);
    } while (choice != 1 && choice !=2);

    file1 = fopen("Inventory.txt", "rb");
    file2 = fopen("tempfile.txt", "wb");

    while (fread(&item, sizeof(item), 1, file1)) {
        int i = 0;
        while (code[i] != '\0' && item.product_code[i] != '\0' && code[i] ==
item.product_code[i]) {
        i++;
        }

        if (code[i] == '\0' && item.product_code[i] == '\0') {
        if (choice == 1) {
        printf("\nUpdating Data of the chosen product. %s\n", code);

        printf("Enter Product Name: ");
        scanf("%s", item.product_name);

        printf("Enter Product price: ");
        scanf("%d", &item.price);

        printf("Enter Quantity: ");
        scanf("%d", &item.quantity);
        printf("\n");

        fwrite(&item, sizeof(item), 1, file2);
        }
        else if (choice == 2) {
        printf("\nDeleting Product. %s\n", code);
        }
        } else {
        fwrite(&item, sizeof(item), 1, file2);
        }
    }

    fclose(file1);
    fclose(file2);

    if (choice == 2) {
        remove("Inventory.txt");
        rename("tempfile.txt", "Inventory.txt");
    } else {
        file1 = fopen("Inventory.txt", "wb");
        file2 = fopen("tempfile.txt", "rb");

        while (fread(&item, sizeof(item), 1, file2)) {
        fwrite(&item, sizeof(item), 1, file1);
        }

        fclose(file1);
        fclose(file2);
    }

    printf("\nPress 1 to update or delete another product, or any other key to go back to the main menu.\n");
    int retryChoice;
    scanf("%d", &retryChoice);
    if (retryChoice != 1) {
        return;
    }
    }
    } while (1);
}

// EXIT SOFTWARE
void exitapp() {
    char choice;
```

```
        printf("\n Do you want to close the applications?(Y/n)");
        scanf("%s", &choice);
        if (choice == 'Y') {
        exit(0);
        } else (choice == 'n'); {
        system("cls");
        void options();
        }
}

// . . .
void options();

// LOGIN INTERFACE
int main() {
        printf("\tLOGIN:\n");
        printf("************************\n");
        char username[15], password[10];

        while (1) {
        printf("Enter username: ");
        scanf("%s", username);
        printf("Enter password: ");
        scanf("%s", password);

        int username_valid = 1;
        int password_valid = 1;

        for (int i = 0; username[i] != '\0' || USERNAME[i] != '\0'; i++) {
        if (username[i] != USERNAME[i]) {
                username_valid = 0;
                break;
        }
        } for (int i = 0; password[i] != '\0' || PASSWORD[i] != '\0'; i++) {
        if (password[i] != PASSWORD[i]) {
                password_valid = 0;
                break;
        }
        } if (username_valid && password_valid) {
        printf("\nLogin successful!!\n");
        system("cls");
        options();
        } else {
        printf("\nSorry, you entered the wrong information.\n");
```

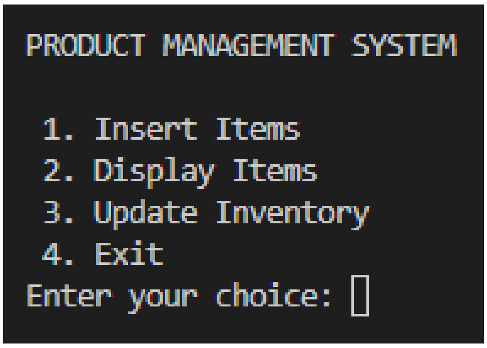
```
	printf("Please try again.\n\n");
	}
	}
	return 0;
}

// MENU INTERFACE
void options() {
	printf("\nPRODUCT MANAGEMENT SYSTEM\n");

	int num, choice;
	while (1) {
	printf("\n 1. Insert Items");
	printf("\n 2. Display Items");
	printf("\n 3. Update Inventory");
	printf("\n 4. Exit\n");
	printf("Enter your choice: ");
	scanf("%d", &choice);
	system("cls");

	if (choice == 1) {
	addItem();
	} else if (choice == 2) {
	display();
	} else if (choice == 3) {
	update();
	} else if (choice == 4) {
	exitapp();
	} else {
	printf("Invalid choice.\n");
	}
	}
}
```

V. POSTER:

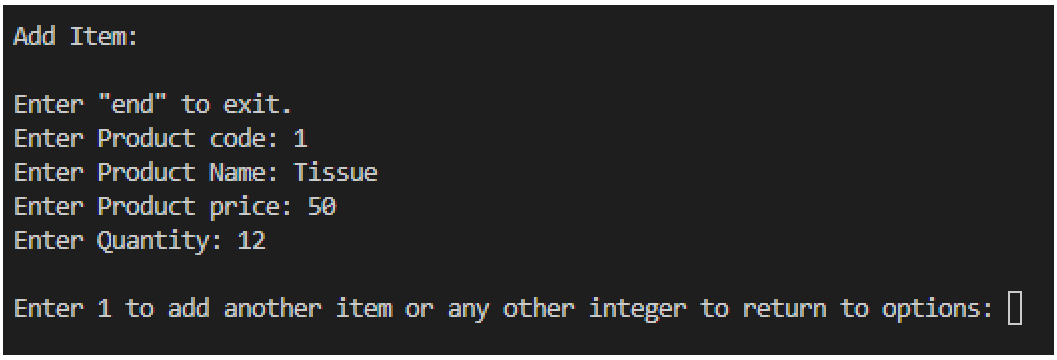
VI. OUTPUT:



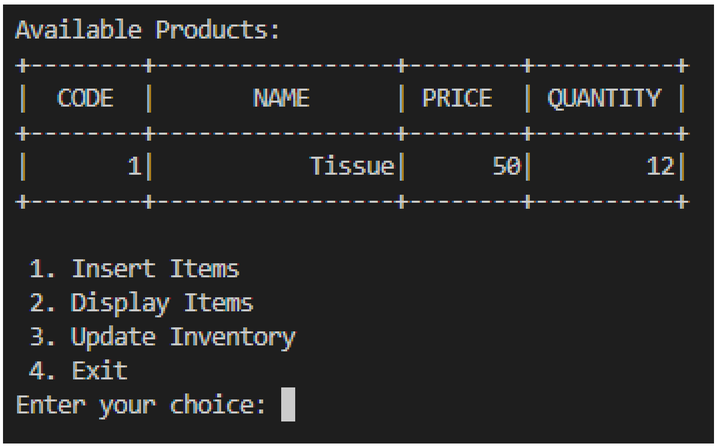
```
         LOGIN:
*************************
Enter username: user
Enter password: pass
```



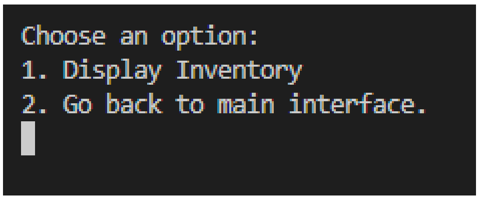
```
PRODUCT MANAGEMENT SYSTEM

 1. Insert Items
 2. Display Items
 3. Update Inventory
 4. Exit
Enter your choice:
```



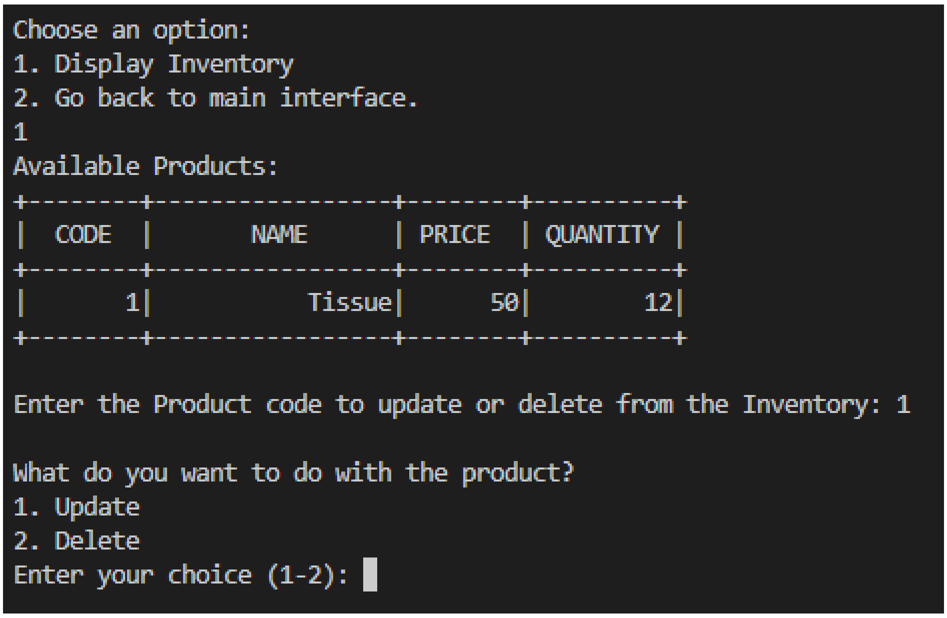
```
Add Item:

Enter "end" to exit.
Enter Product code: 1
Enter Product Name: Tissue
Enter Product price: 50
Enter Quantity: 12

Enter 1 to add another item or any other integer to return to options:
```



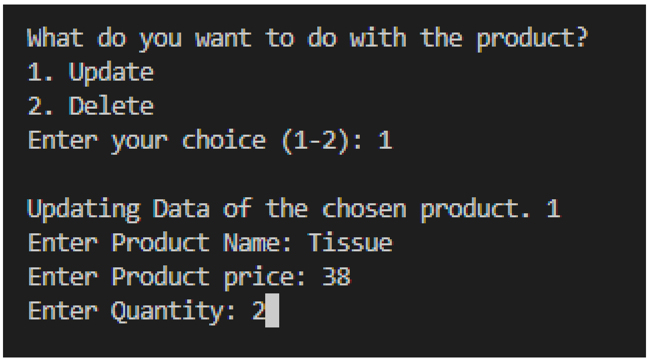
```
Available Products:
+--------+----------------+--------+----------+
|  CODE  |      NAME      | PRICE  | QUANTITY |
+--------+----------------+--------+----------+
|       1|          Tissue|      50|        12|
+--------+----------------+--------+----------+


 1. Insert Items
 2. Display Items
 3. Update Inventory
 4. Exit
Enter your choice: 
```



```
Choose an option:
1. Display Inventory
2. Go back to main interface.
```



```
Choose an option:
1. Display Inventory
2. Go back to main interface.
1
Available Products:
+--------+----------------+--------+----------+
|  CODE  |      NAME      | PRICE  | QUANTITY |
+--------+----------------+--------+----------+
|       1|          Tissue|      50|        12|
+--------+----------------+--------+----------+

Enter the Product code to update or delete from the Inventory: 1

What do you want to do with the product?
1. Update
2. Delete
Enter your choice (1-2): 
```



```
What do you want to do with the product?
1. Update
2. Delete
Enter your choice (1-2): 1

Updating Data of the chosen product. 1
Enter Product Name: Tissue
Enter Product price: 38
Enter Quantity: 2
```

VII. CONCLUSION:

In conclusion, as a mechanical engineering student, developing a product management system using C programming language is an excellent way to acquire programming skills and gain practical experience in developing software applications. This project offers a great opportunity to design and implement a complete system for managing products, with features such as adding new products, viewing product lists, and updating existing products. By working on this project, students can enhance their understanding of software development principles, and develop valuable skills such as problem-solving, coding, testing, debugging, and documentation. Overall, developing a product management system using C programming language is an exciting and challenging project that can help mechanical engineering students to acquire practical programming skills and develop a better understanding of software development.