

201914024 | CSE-106 | October 22, 2019

Inventory Software design

by shad reza

# **Name of the project:** Inventory Software design

# **Programming Language used:** C

# **Developed by:**

Name: Shad Reza

Id: 201914024

Batch: CSE - 19

Course: Computer Science & Engineering

Level: 01

Term: 02

Course Code: CSE – 106

Course Title: Structured Programming Language Sessional

Military Institute of Science & Technology

# **Supervisor:**

Lecturer Onik Sir

CSE department

MIST

# **Overview**

A simple software for a management system or inventory shop, where one can do the job of both the management side and also the customer side.

This project can satisfy both the users as their needs. As it can take inputs modify or edit them. Delete a whole product, show the zero products as well as the profit and existing bars.

To make both the management and consumer experience easy this project was build to help in that regards.

Again, it can sell products take feedback and also make the bill viable easily.

# **Difference with others**

I would not say that it has a massive difference but the implementation steps are quite personal but the extra ordinary thing I feel to be the addition of music.

Have collaborated with the management of files and the main focus is to maintain a database that stores the data even though the program ends or unfortunately crashes.

If a data is deleted it may be deleted from the display of available products but will remain always in a file that keeps the data stored until the file is not deleted.

The administration runs only if the entered username password and user id is validated.

# **Components / Features**

The first thing to point is this software works for both the management side and also the consumer side

It has the options that control the management side

* entry of the products
* modification and entering new products
* deleting product
* showing the available products
* editing the vat and discount of the shop
* show zero quantity
* notify in every step if any product is less than 10 quantities
* show the profit (%)
* show the bar chart of the present amounts of the products

It has the options that control the consumer side

* have option to store feedback
* have option to purchase

# **How the components interact with each other**

In my project there are multiple times that there needs to be collaboration between various files. Mainly the connection between the files are a vital role.

NB. If the files are deleted then the program can crash though the chances are few.

By calling a function by one to another the connections are made and bridge of the gap is made between the input and output.

These communications are really vital for the internal process over all.

# **Code of important components and their explanation**

Here I am specifying the function names and their purpose

* int main (): It mainly controls the flow and direction of the entire program
* void bar(): It checks the quantities for the function that prints the bar
* void bar\_print(char \* a,int n): It prints the bar of the quantity
* void change\_vat\_disc(): Changes the vat and discount
* consumerinfo consumerinfogather(): Gets the information of the consumer
* void consumeroptions (): Controls the flow of the consumer options
* void del\_prd(): Deletes a certain product
* void digit (int num): Prints a number in large scale
* void feedback (): Gets feedback from the consumer
* void firstdisplay(): Shows the first appearance
* int how\_many\_prd\_ever\_enterd(): lets to know how many products have ever been entered

by the management side

* int how\_many\_prd\_to\_sell(): lets to know how many products are there to be sold
* void is\_less\_than\_10(): lets to know if a certain product’s quantity who are less than 10 ammounts
* int isidok(int x): lets to know if a certain product is okay for further inspection or not
* int isidok(int x): lets to know if a certain product that has been entered is okay for further inspection or not
* int isidokok(int x): lets to know if a certain product that can be displayed for sale is okay for further inspection or not
* int isidoktosell(int x): lets to know if the product can be sold or not
* int log\_in(): log in option that lets the management side to gain control over the software
* int managementaccess(): hands the floor to the management side if they have gained the access respectfully by entering the valid and correct username user id and password
* int managementchoice (): show the choices that the management side will have
* void managementopt (): shows the option valid for the management
* void managementtitleprint(): decoration that prints the title with underline style
* int prdentry(): allows the management side to enter products to the shop
* void purchase(): purchase method for the customer side
* void show\_prd\_to\_sell(): shows the products that are available for the sale
* const char \* show\_time (): shows time and date and day
* void show\_zero\_qty(): shows if any product is at zero quantity
* void time\_date: function that generates the time and date
* void underline(): decoration under the title
* int wanttologout (): asks the management user if he/she wants to log out of the program
* PlaySound(TEXT("D:\\programming\\c\\cse106\_project\\INVENTORY\_soft\_by\_shadreza\\my.wav"),NULL,SND\_ASYNC); … … … this is the code that generates the background music

These are mainly the functions that have been used in the program to simplify the work.   
Also, the use of structure and pointers have made the work really easy.

Must say the FILE has made the data storing an ease of work cause the programs reads and writes or appends on the files as they help the project to complete the tasks easily.

# **Difficulties faced while implementation of the features of the code**

* At first while implementing the file I faced a lot of complexities but it seemed that if I had just steeled the mind calmly and used the make up the algorithm on a piece of paper and the flow then it helped me a lot
* Sometimes the code crashed for some unknown reasons but then I saw that some similar problems repeated i.e. using not the correct file pointer while fprintf and fscanf

* There was a point that when I had to implement the file option then I had to totally start from the beginning again though the work was done because then the whole structured changed. So, remaking the project once again was a big barrier
* Printing the digits in the large scale was a tough task with printing the graph bar because printing the blocked characters was a new thing. First I used wprintf but that nullified the use of the normal printf so then had to change the strategy
* Using the music, I faced a problem how to make both the song and the project run parallelly but without using SND\_ASYNC , other options just played the sound but the later on lines of the project were not executing and was stuck there
* Sometimes I was stuck on a single function for days but could not find a correct answer. But giving proper time and respect in the project resulted in this successful and complete one
* As such a big project the time management was a vital issue. So, had to spend quality hours for the project and which was sometimes a bit hard but at last ALHAMDULILLAH by the grace of THE ALMIGHTY the project is completed.

# **Thanks to the resources**

* At first I would like to thank my project supervisor, Lec. Onik Sir for his continuous moderations and directional process which lead to this final project
* Then thanks to Google & Youtube because many questions got answered from them with many articles and video tutorials
* Geeks for Geeks, Stack overflow these websites really helped in the process with their rich resources and articles
* Shout out to Eiro Nareth because the titanic background music was his content on his youtube chanel
* Again, Thanks to ALMIGHTY because without HIS blessings this project could have never been such a successful one

**…………………………………………………………… The End ………………………………………………………………………**