

**CST1510 COURSEWORK**

**2021 Oct Intake**

**PARKE ALEXANDER SHEKHAR**

**M00832048**

CLIENT SIDE

CONNECTING TO SERVER

LOGIN PAGE

STOCK COMPARISON

RECEIPT

SERVER SIDE

CONNECTING TO DATABASE

MANAGING DATABASE

CREATING SOCKET CONNECTION

CREATING AND EDITING CSV FILE

STOCK DATABASE

PRODUCT NAME

QUANTITY

PRICE

RECEIPT

PRODUCT

PRICE TOTAL PER PRODUCT

TOTAL

SHOPKEEPER DATABASE

NAME

PASSWORD

SERVER CONNECTION

IP ADDRESS DECLARATION

CONNECTTING TO PORT

LOGIN PAGE

SHOPKEEPER LOGIN DETAILS

CUSTOMER NAME

DESCRIPTION

The project is based upon designing a client-server shop system. The project includes a shop for vehicle engines. The shopkeeper logs on the client side and inputs the order of the customer/client on the GUI implemented. Based on an available stock the program will produce a certain output. If the amount ordered is more than the available stock the program issues a warning.

On the server side the program manages the database and updates it accordingly. A file is created where the order placed is stored.

STEPS:

* Logging in the system on the client side by the shopkeeper
* Adding customer name
* Choosing products from the available(specified quantity input by customer)
* Saving the order
* Ordering the products
* Creating a receipt for the products
* Displaying the order with the total amount to pay

IMPLEMENTATION:

From the design each part was roughly made modular prior to the creation of the program.

* Designing the file writing function
* Designing the database
* Designing the login page
* Designing the receipt page

PROGRAMMING PROCESS:

* Creating server connection on the server side

Using the built-in function ‘socket’ within python, we can create a server connection so that we may transfer data from one python code to another. The socket library was imported in python and then used.



* Creating the connection on the client side

As the server side, the socket library was imported for the connection to be established between the client and the server

* Testing the connection to ensure proper transfer of data

Within the code test data was sent from the client to the server through a direct input on the CLI within the IDE used to make the program

* Creating GUI on the client side

To create the GUI, the tkinter library was imported to enable the creation of a GUI window for interaction between the user and the program.

* Creating login page using TopLevel widget of tkinter

From the tkinter library, instead of creating multiple windows, the TopLevel widget found with the tkinter library has been used for faster and more seamless switch between the login page and the main window.

* Creating main window to display products
* Creating the csv writer on the server side
* Testing the file writer

After completing the file writer, test data was used to make sure the code worked as it should

* Creating the database manager
* Testing the database manager

After completing the manager, test data was used to check whether the database’s data was being updated properly

* Creating the bill on the client side

A new window was created to display the contents of the receipt and the user was givn the option between confirming the order or cancelling.

SNIPPETS OF PROGRAM GUI:

Graphical user interface, application

Description automatically generated

Login page to input user data

Graphical user interface, application, Teams

Description automatically generated

Warning message when requirements have not been met

Timeline

Description automatically generated

Main window with products and their images

A picture containing chart

Description automatically generated

New window for the receipt with options of confirmation or cancellation.

Each product with its quantity and its total price per product

LIMITATIONS:

No modification for the order list

No dashboard for the shopkeeper

Not enough details on the receipt

ACKNOWLEDGEMENT:

I would like to thank Ms Chitisha Gunnoo and Mrs Waseemah Moedeen for their help in the development of this program.