Screenshots of the running program with multiple tests



Figure 1. Home Page

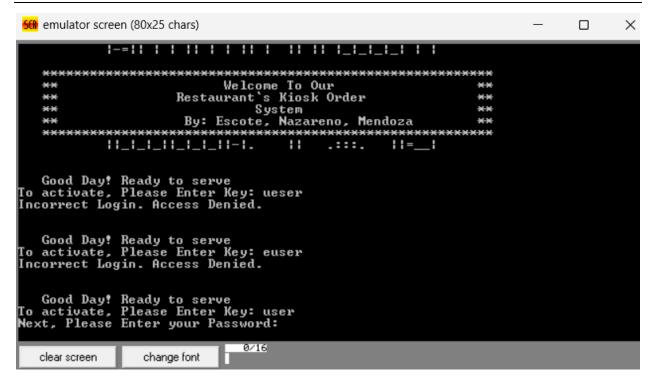


Figure 2. User Log In (With Validation)

```
60 emulator screen (80x25 chars)
                                                                                                                 X
     Welcome To Our
Restaurant's Kiosk Order
System
     ××
                                                                                             ××
     ××
                                                                                             ××
     ××
                                                                                             ××
                                 ××
                                                     ::
                  !!_!_!_!!_!_!_!.
                                                              .:::.
                                                                          | | =___|
Good Day! Ready to serve
To activate, Please Enter Key: user
Next, Please Enter your Password: abc
Incorrect Login. Access Denied.
Good Day! Ready to serve
To activate, Please Enter Key: user
Next, Please Enter your Password: 321
Incorrect Login. Access Denied.
Good Day! Ready to serve
To activate, Please Enter Key: user
Next, Please Enter your Password: _
                                             0/16
    clear screen
                         change font
```

Figure 3. Password Log In (Validation)



Figure 4. Password Log In (Verified)



Figure 5. Main Menu (After Log In Page)



Figure 6. Main Menu (Validation if Input != 1-5, Pick again)



Figure 7. Main Menu (If == Input 5, If Cancel Order == No, Return to Main Menu)

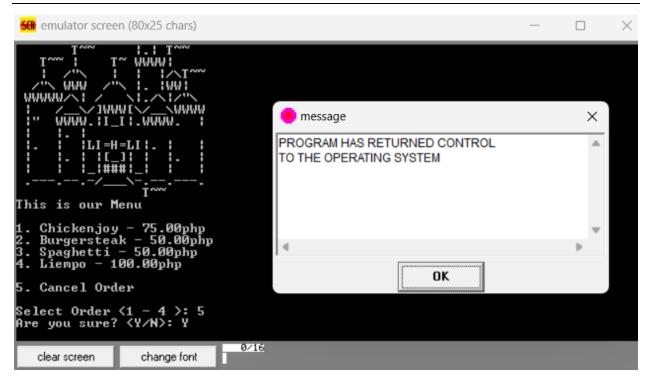


Figure 8. Main Menu (If Input == 5, If Cancel Order == Yes, Terminate Program)

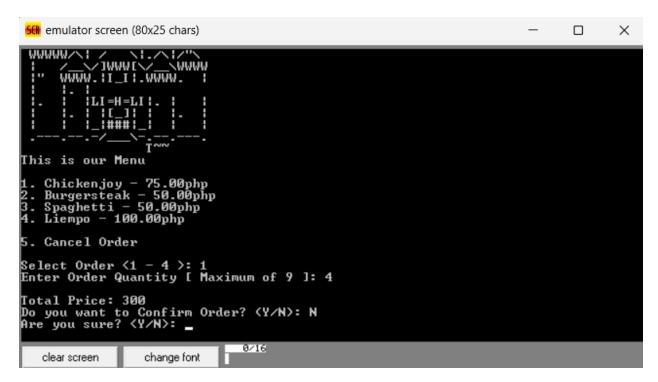


Figure 9. Select Order [Chickenjoy] (If Input == 1 (Chickenjoy).

Computation (Validation): Price x Quantity = Total $[75.00 \times 4 = 300] + \text{Confirmation Prompt}$

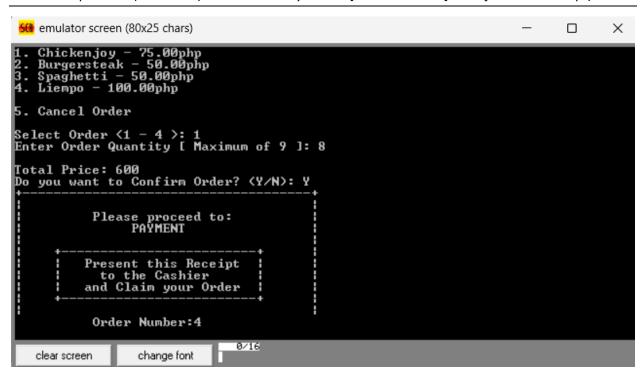


Figure 10. Select Order [Chickenjoy] (If Input == 1 (Chickenjoy).

Computation (Validation): Price x Quantity = Total [75.00 x 8 = 600] + Confirmation Prompt == Y, Proceed to Receipt)

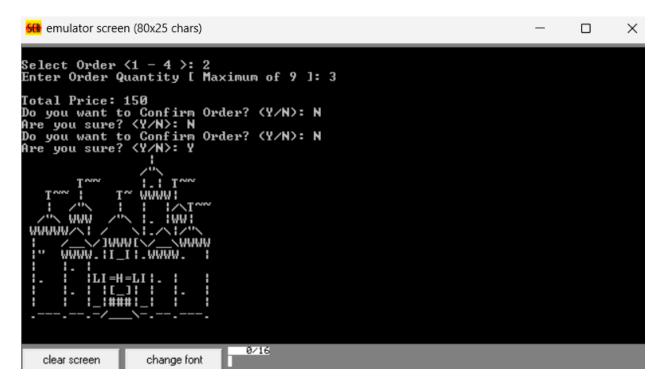


Figure 11. Select Order [Burgersteak] (If Input == 2 (Burgersteak).

Confirmation Prompt == N, Are you sure? == N, Ask again, Are you sure == Y, Create New Order

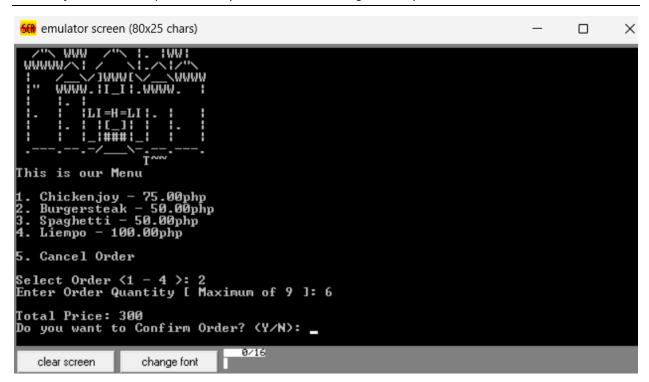


Figure 12. Select Order [Burgersteak] (If Input == 2 (Burgersteak).

Computation (Validation): Price x Quantity = Total $[50.00 \times 6 = 300] + Confirmation Prompt)$

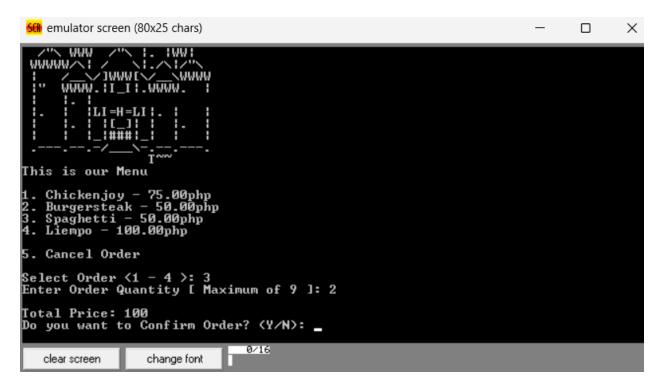


Figure 13. Select Order [Spaghetti] (If Input == 3 (Spaghetti).

Computation (Validation): Price x Quantity = Total [50.00 x 2 = 100] + Confirmation Prompt)

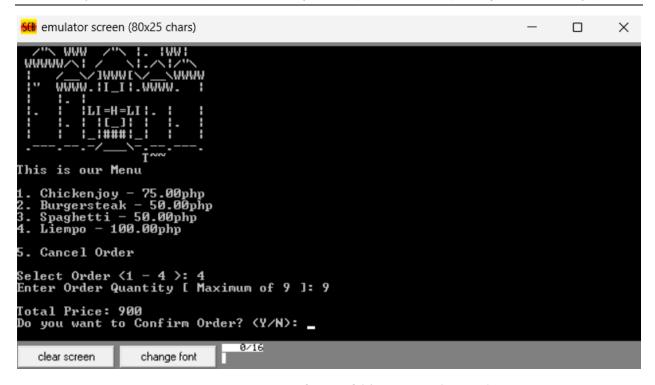


Figure 13. Select Order [Liempo] (If Input == 4 (Liempo).

Computation (Validation): Price x Quantity = Total [100.00 x 9 = 900] + Confirmation Prompt)

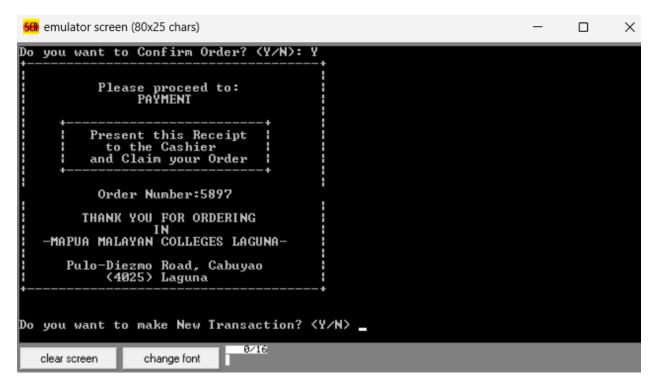


Figure 14. Confirmed Order (Receipt Layout + 1st Proof of Random 'Order Number' Every Transaction)

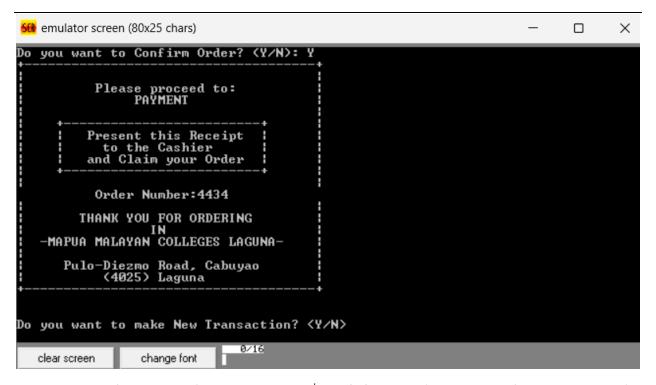


Figure 15. Confirmed Order (Receipt Layout + 2nd Proof of Random 'Order Number' Every Transaction)



Figure 16. Confirmed Order (Receipt Layout + Confirmation Validation, Are you sure? (That you don't want to make New Transaction) == \mathbf{N} , Ask User again for Input)

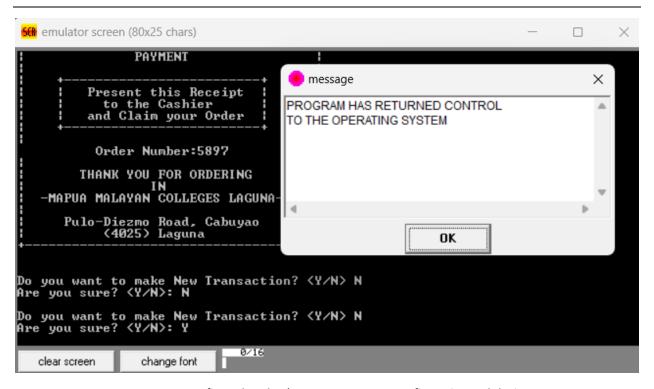


Figure 17. Confirmed Order (Receipt Layout + Confirmation Validation,

Are you sure (That you don't want to make New Transaction)? == Y, Terminate Program)

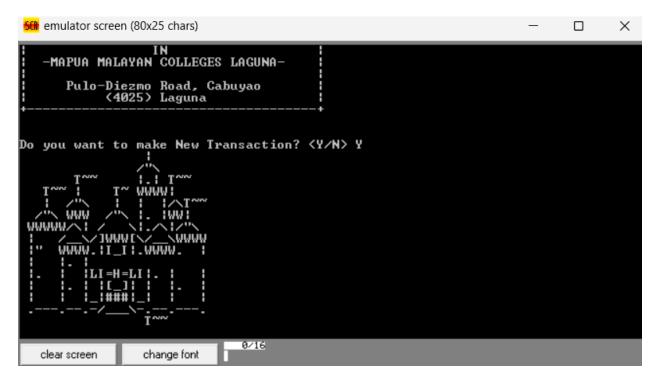


Figure 18. Confirmed Order (Receipt Layout + Confirmation Validation,

Are you sure (That you want to make New Transaction)? == Y, Create New Order, Proceed to Main Menu)

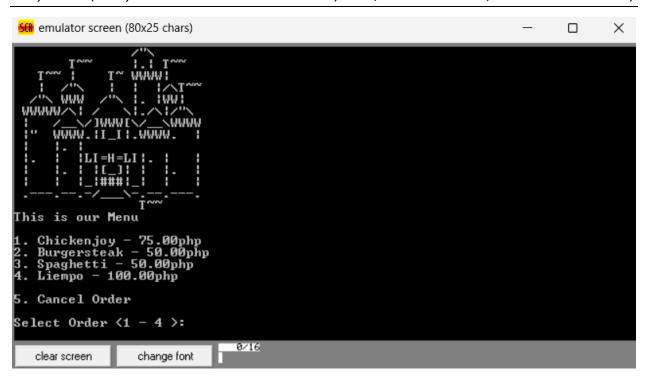


Figure 19. Returned to Main Menu, After Input == Y,

Prompt:

Are you sure (That you want to make New Transaction)? == Y, Arrived at Main Menu)