# Cashier Application (POS System) for CRISTY’S LOVE BURGER HUB

Aryan Jung Karki

B.Sc. (Hons.) Computing, Softwarica College of IT and E-commerce, Coventry University

ST4008CEM: Computing Activity Led Learning Project 1

Giriraj Rawat

July 28, 2022

# Table of Contents

Cashier Application (POS System) for CRISTY’S LOVE BURGER HUB 4

Introduction 4

Login Function 4

Logout function 5

Takeaway/Dine-in function 8

Clear function 10

Complete order function 11

Conclusion 14

# Table of Figures

[Figure 1 **Error! Bookmark not defined.**](#_Toc109681937)

[Figure 2 **Error! Bookmark not defined.**](#_Toc109681938)

[Figure 3 **Error! Bookmark not defined.**](#_Toc109681939)

[Figure 4 **Error! Bookmark not defined.**](#_Toc109681940)

[Figure 5 8](#_Toc109681941)

[Figure 6 9](#_Toc109681942)

[Figure 7 10](#_Toc109681943)

[Figure 8 11](#_Toc109681944)

[Figure 9 **Error! Bookmark not defined.**](#_Toc109681945)

# Cashier Application (POS System) for CRISTY’S LOVE BURGER HUB

# Introduction

The focus of my backend part is to create functions for the POS system to form a relationship between database and frontend using python as a programming language.

# Login Function

The login function is assigned to the “LOGIN” button in the respective employee login windows. If the user credentials match the data in the database obtained through syntax .fetchall(), then it makes the employee status *‘active’* and opens their personal dashboard, else it prompts login ‘*failure’* message box. Figure 1 shows login functions in the project.

### Figure

Login function.

Text

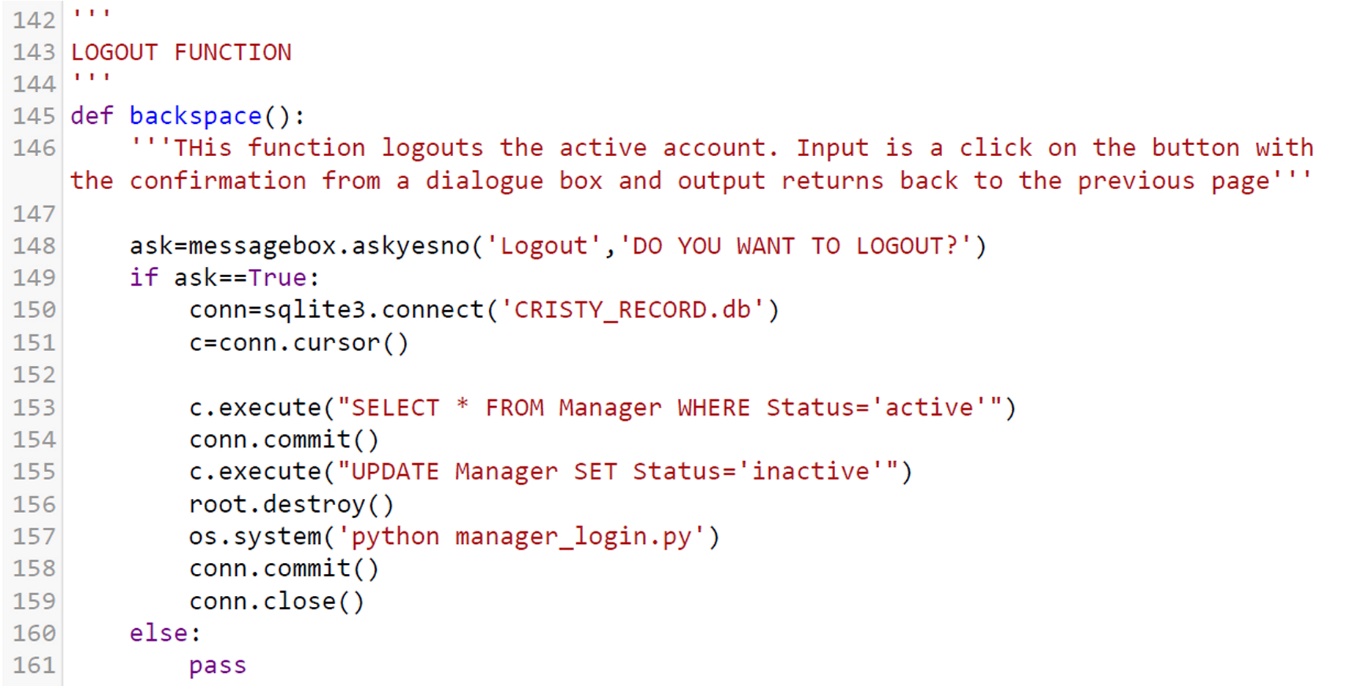
Description automatically generated

# Logout function

The logout function is assigned to the ‘LOGOUT’ button in the GUI. This function prompts the user if they want to logout or not in the form of a yes/no question, the user’s decision is stored in the variable “ask”. If the value of “ask” is “True”, then it sets the ‘*active’* employee status as ‘*inactive*’ as shown in figure 2.

### Figure

Logout function

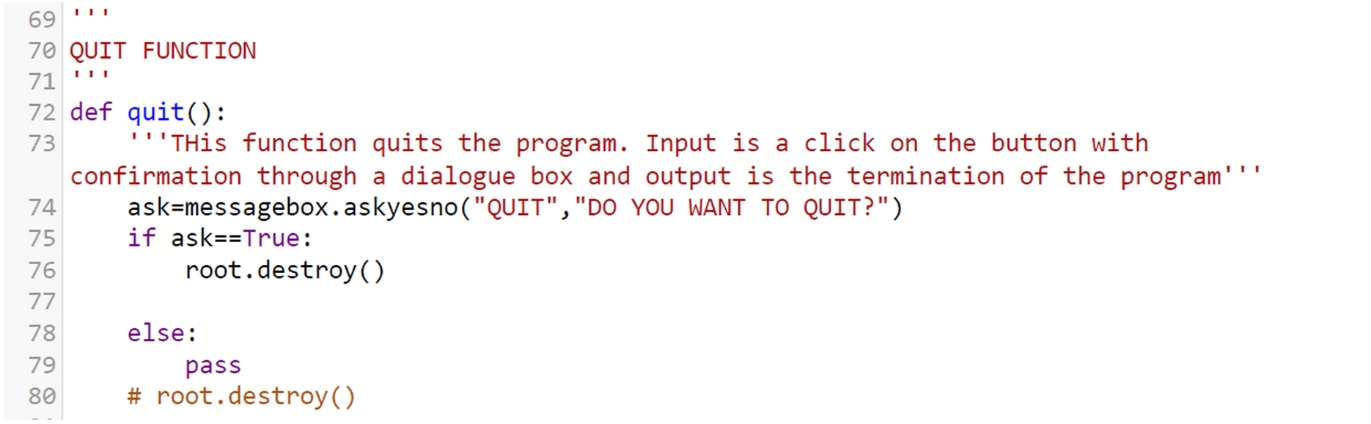
****

# Quit/Back function

The quit/backspace function destroys the current active window and opens another window designated within the os.system() function. Additionally, the ‘QUIT’ button questions the users if they want to quit the window or not. A positive answer closes the program. Figure 3 and 4 represents quit and back functions respectively.

### Figure

Quit function



### Figure

Back function

Graphical user interface, text, application, email

Description automatically generated

# Takeaway/Dine-in function

Takeaway/Dine-in function helps the staff to prepare food either on tray or package for dine-in or takeaway respectively. The codes are shown in figure 5 and 6.

### Figure 5

*Takeaway function*

### Figure 6

*Dine-in function*

# Clear function

The clear function clears the table so that new order can be placed. It deletes whatever data was inserted into the order table and pops a message prompt saying that the order was deleted successfully.

### Figure 7

*Clear function*

# Complete order function

It updates the status of the order as complete in the database after it is delivered to the customer. Code is shown in figure 8.

### Figure 8

*Function for complete order*

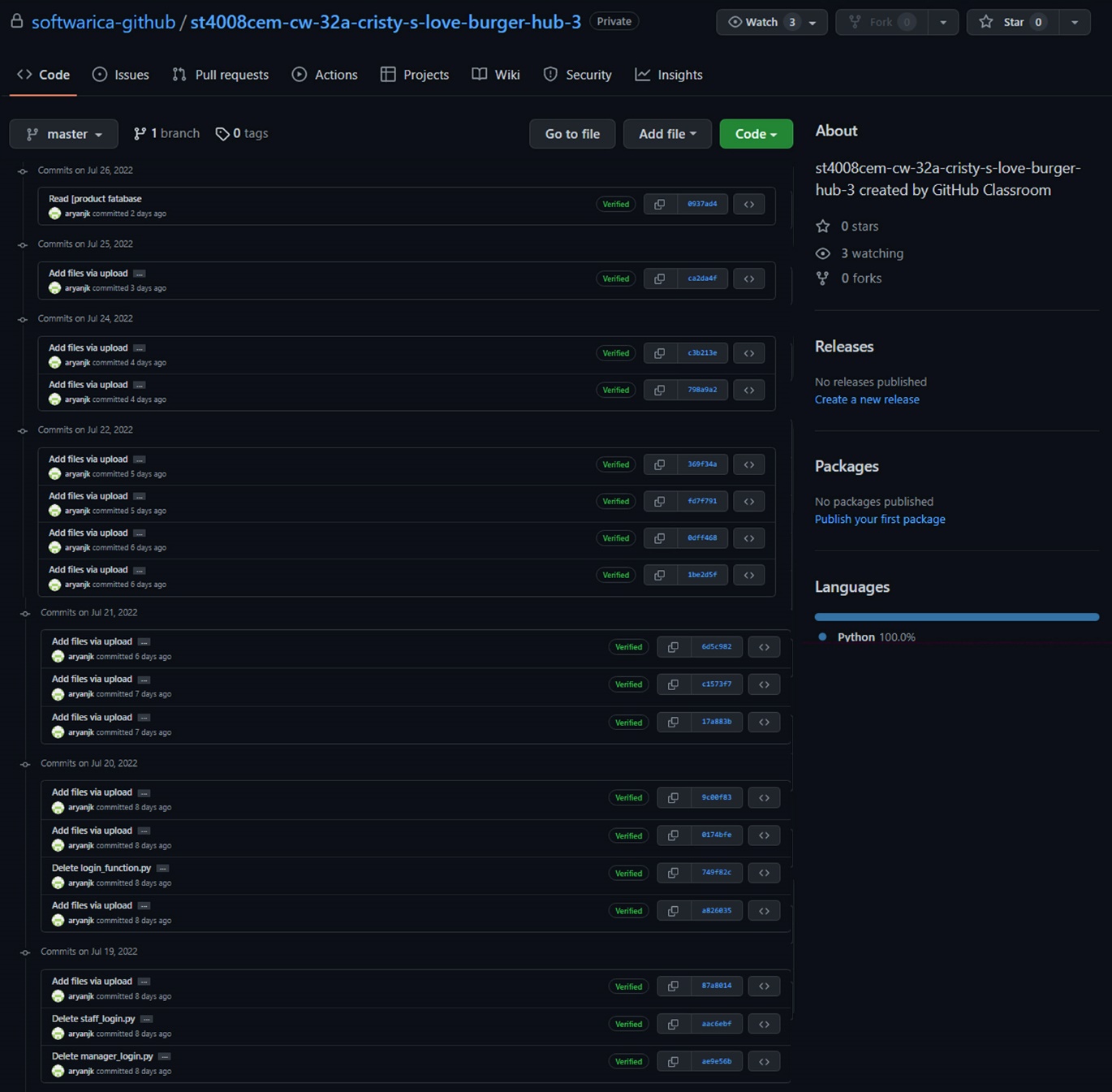
# Version Control

All the functions coded were committed in the github repository provided by Softwarica College of IT and E-commerce as shown in figure 9.

**Github Link:** <https://github.com/softwarica-github/st4008cem-cw-32a-cristy-s-love-burger-hub-3.git>

### Figure 9

Github commits

**

# Conclusion

The aim of my backend part is to create functions for the POS system to form a relationship between database and frontend. For this, I used python programing language, mainly user-defined functions, if-else conditions, and loops integrating logics taught in mathematics.

Due to many indicators name in GUI, confusion arose while creating relationships between elements of GUI and database. To solve this problem our team had several interactions and group discussions to make use of comments and arrange the GUI codes and database codes separately and systematically as much as possible. I also inserted docstrings in every function that I created so that it is easily understandable to my teammates.

Undertaking this project revealed my strong problem-solving skills and logical thinking. Personalitywise, I am an introvert. I am always shy to work in groups, however due to several group discussions to solve our coding problems among my friendly teammates, I started to mingle with them slowly but steadily. This has boosted my confidence to work in a team and express my thoughts to the members of the group.