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

Rishav Kumar Shah

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](#_Toc109873091)

[Introduction 4](#_Toc109873092)

[Product details 4](#_Toc109873093)

[GUI Menu 7](#_Toc109873094)

[Version Control 11](#_Toc109873095)

[Conclusion 12](#_Toc109873096)

# Table of Figures

[Figure 1 5](#_Toc109926794)

[Figure 2 6](#_Toc109926795)

[Figure 3 7](#_Toc109926796)

[Figure 4 8](#_Toc109926797)

[Figure 5 8](#_Toc109926798)

[Figure 6 9](#_Toc109926799)

[Figure 7 10](#_Toc109926800)

[Figure 8 11](#_Toc109926801)

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

# Introduction

Interface for order taking, billing, and product details (*Create-Record-Update-Delete*) are the aim of GUI menu in this *Point of Sale (*POS) system so that order are easy to be taken. Tkinter framework of Python is used in GUI.

# Product details

This page is used for listing items and its price on the menu page. And we can also add, delete, or update an item in on database as per requirement. It has a ‘BACK’ button to return to previous page if there is no need of any action in this page. Its UI and code snippet are shown in figure 1 and 2.

### Figure 1

UI of product details

Website

Description automatically generated

### Figure 2

Code snippet of Product details

*Graphical user interface, text, application

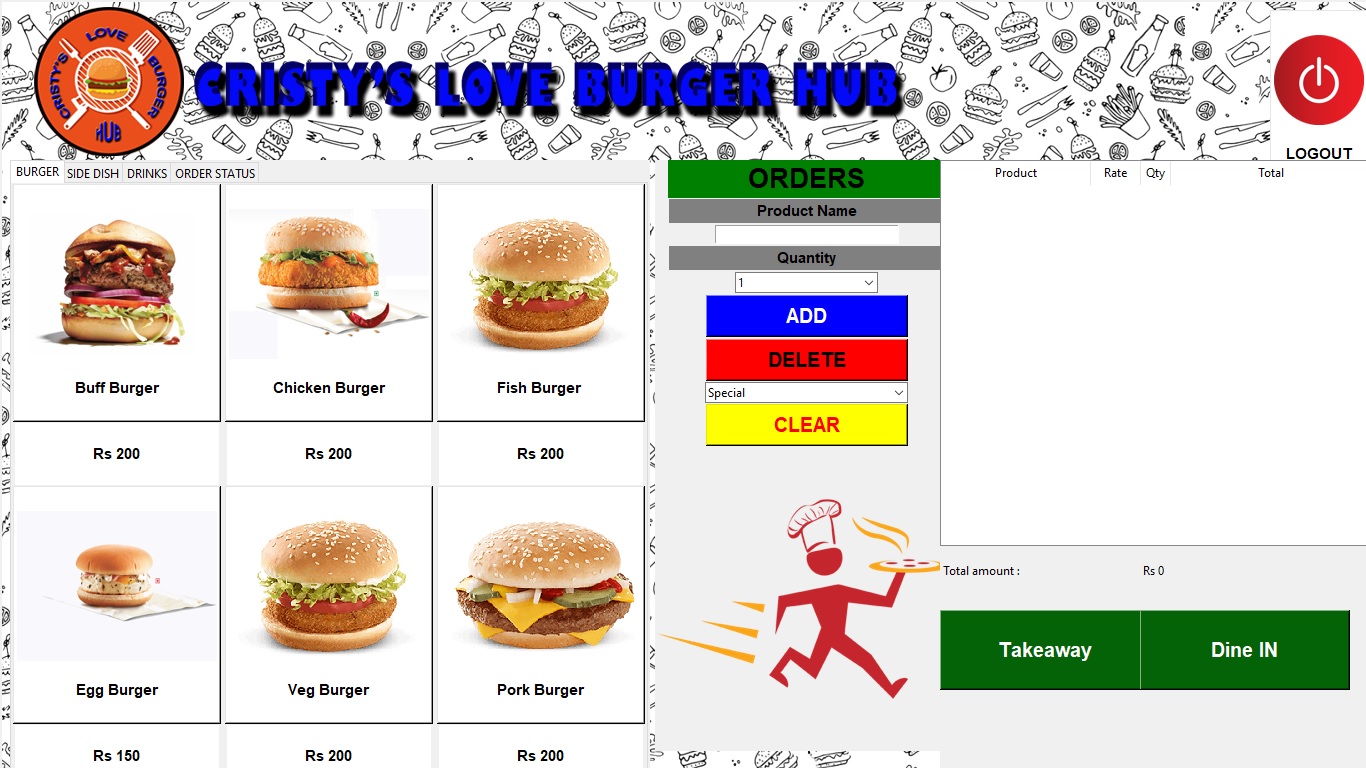
Description automatically generated*

# GUI Menu

This page opens after successful staff login. Its main purpose is to display self-descriptive restaurant products, order taking, and billing of the order. This menu page uses ‘tab’ tools of Tkinter to categorize the foods into burger, side-dish, and drinks. Order status also has its own separate tab. The ordered food items can be noted with just a click of button. *Takeaway* and *Dine-in* buttons are also present for respective delivery in *‘Billing frame’*. Special products added in ‘Product details’ can also be ordered by customers. Same food items can be ordered at different quantities and there is no need to enter a single food multiple time. ‘Order status’ tab has ‘complete’ button to record when an order is delivered. The related UI and snippet of related codes are in figures 3-7.

### Figure 3

UI of Burger

**

### Figure 4

UI of Side-dish

*Graphical user interface

Description automatically generated with medium confidence*

### Figure 5

UI of Drinks

*Graphical user interface

Description automatically generated*

### Figure 6

UI of Order status

*Graphical user interface

Description automatically generated with medium confidence*

### Figure 7

Snippet of GUI

**

# Version Control

**Github:**<https://github.com/softwarica-github/st4008cem-cw-32a-cristy-s-love-burger-hub-3.git> as shown in figure 8.

### Figure 8

Github Commits

Graphical user interface, text

Description automatically generated

# Conclusion

Creation of an interactive interface with order taking, billing, and the CRUD of product details using Tkinter tools were the aim of my frontend part. The main plan was to create a GUI page that has all the items on the menu, including side dishes, drinks, order status and dining option. I achieved all that using ‘tabs’ to categorize all the items (food, side dish, drinks) in an ordered format. There is a billing section for the customer to see the total amount with delivery options. Overall, GUI menu will help users to take an order with a click, inform the total cost, and the order status.

I have learned how to work in a group and collaborate with my teammates especially my other GUI partner. To make both of our GUI’s consistent and follow similar style, we had regular meetings to share designing ideas. A last hour problem was noticed in ‘order menu’, it was ignorance of ‘special menus’ added in ‘Product details’ of my partner’s work contribution. We solved it as the solution was easy. But this taught me an important lesson that errors in software engineering just don’t happen due to mistake in backend functions. Problems also arise due to overlooking some components in GUI, thereby GUI also has a major role in a successful software development. I will never forget this lesson in my future projects, double check my work and compare it to the SRS document to provide a complete working software.