Design and Implementation of Mess Tiffin Management System in the Python Environment

RiteshNimje, Aparna Gurjar

[1]Student, Department of Computer Application, Shri Ramdeobaba College of Engineering and Management, Nagpur, India

[2]Assistant Professor, Department of Computer Application, Shri Ramdeobaba College of Engineering and Management, Nagpur,India

Corresponding Author: RiteshNimje

ABSTRACT: *It is observed that, traditionally catering services or messes have relied on registers/book entry system for management of tiffin services of their clients. Many of these facilities are now switching over to the digital platform. This project aims to build a fully functional Mess Tiffin Management System taking into account the requirements posed by local constraints. This system can be used both by the mess customers as well as the owner. It implements various functionalities like customer management, food menu management and real time updates about tiffin orders through e-mails. The project has been created in Python language.We have used Sqlite3 for database operations and SMTP for sending the emails.*

KEYWORD: *Python, Mess, Tiffin Service, Customer, Owner, Tkinter, SQLite3,*

I. INTRODUCTION

As large workforce in India moves out of their homes for the purpose of employment, managing their daily meals becomes quite challenging for most of them. It is observed that most people prefer the tiffin service as many messes are available in their residential vicinity. Traditionally these messes have relied on registers/book entry system for management of mess tiffin services of their clients. It becomes difficult for the mess owners to manage each and every customers data. With the advent of smart phones and other communication facilities many messes are switching over to the digital platform. Currently there is a requirement for creating a digital platform, which is able to address local issues in this regard so that the mess tiffin services can be managed effectively and efficiently. We aim to implement a fully functional Mess Tiffin Management System which will be able to achieve the said purpose. In this project we have created an entire workflow for managing range of activities of the customers and the mess owner. The choice of programming language was Python as it has now established itself as one of the most popular languages for interactive client management.Python also supports a large number of libraries which can be simply imported into the project on requirement basis. In our project we have used the following python libraries. Their brief description follows.

- *Tkinter:* This is a module of python which contains GUI (Graphical User Interface) toolkit in a form of scripts. In our project this moduleis used to create the user interface for the mess owner and customers in which they can interact with each other.
- *SQLite3*: This is a standard python Library which helps to perform the database operations. It is self-contained, serverless, zero-configuration and transactional. It is light weight and very fast. In our project this module is used to create the owner and customer databases. The owner can store the information from user and this data can be used for responding to daily tiffin requirements, sending email messages, and managing the customer data. The customer in turn can also choose food menu and update tiffin order.
- *SMTP (Simple Mail Transfer Protocol*): This module handles the sending and routing of the email messages between servers. In our project SMTP is used for sending the emails to customer. At the end of every month the owner can send email to notify the customer for payment of monthly bills. For this functionality we have used the ‘smtplib’ library.

The main vision behind developing this project was that the people in our country who depend on mess services belong to different states and have different choices of food. They need tiffin at their particular destinations which need to be updated in real time if they are travelling to other parts of the city for few days. Each customer has his/her own timing at which he/she prefers to have his/her tiffin delivered. In case of paper-based system, it is quite possible that customers’ tiffin gets exchanged due to incorrect entry, and there is also possibility of loss of information due to damage/theft of pen-paper based data. To overcome all such problems this project has done survey of multiple messes, identified the problem areas as well as requirements and has developed a system model. The software used is Python 3.6 in Pycharm IDE as well as PAGE Software. The next section describes the system model with reference to figure 1.

II. THE MODEL OF MESS TIFFIN MANAGEMENT SYSTEM

This system identifies two types of users: Mess owner(Owner) and the Customer. The user interface has been developed using python Tkintermodule and handles the input of all owner-customer information. The main window has been created which allows role-based log-in to either customer or the owner. Both roles have different functionalities.

The owner section has included functionality like add customer, show customer data, enter today's special menu, show customerresponse etc.

The customer section has a different UI based on specific customer role. It has sections like sign-in window and dynamic user window.

Few functional highlights of the system are as follows

1. The mess owner creates new customer.
2. The owner can also delete the customer.
3. Mess owner can update the menu every day.
4. The owner has a record about payment information of each user.
5. Mess owner can print the customer data in a form of List view of customers.
6. According to the information from database the owner handles the management of mess service.
7. The owner can also send the message to customer about mess bill payment.
8. The customer can choose the menu items.
9. The customer can record the response about menu. Based on this food can be prepared. As a result, there is no food wastage problem.
10. The customer can also skip the tiffin and inform the mess about it.
11. Each customer has a separate record in the database.
12. Each customer provides the credentials to login to the window.
13. Customer window has a lot of information about customer
14. Customer window contains the notification related to number of days are left for payment.
15. In customer window, the customer knows about the remaining days left for payment and the due date of payment

The functionality provided by various buttons is described briefly.

Owner section buttons:

- ***Add Customer:*** In this, the owner gets the information from customer like name, address, mobile number, email id, etc and this information is saved into the database (sqlite3).
- ***Show Customer data:*** In this we create a list view for fetching the data from database. This contains data of all added customers. Due to this information the owner can perform operations like update the information of customer, delete the customer data, send email by selecting the customer.
- ***Enter Today's Menu:*** Each registered Customercan select from multiple optionsin the given menu, which is then added by the owner as the special menu required by the customer.
- ***Show Response:*** The customer can select the dish from options given by owner. Each customer response can be recorded into the database and seen by the owner. Due to these responses the owner can prepare the lunch and dinner.

Customer section buttons:

- ***Sign in Window:*** In this UI the customer can log-in to the system by providing the user Id and password.
- ***User Window:*** When customer signs in then according to the Id, we get data from database and create a dynamic window which contains all the information about the user and also shows how many days are left for payment. When owner adds the options of special menu,then in User window buttons are created. Users can click on them and select food options. There is a constraint that each customer can give response only one time. If customer has already given a response then the button is disabled.

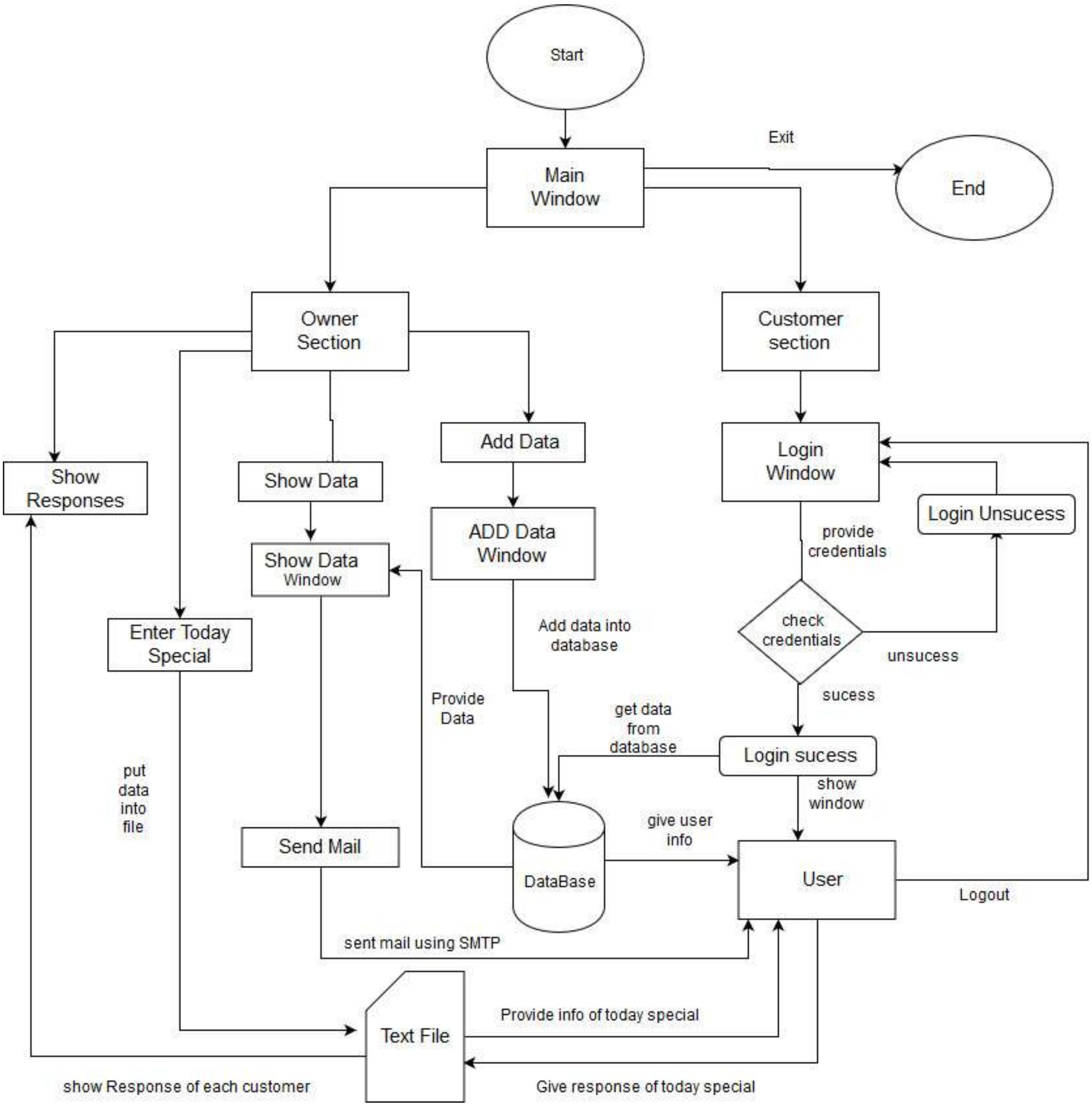


Figure 1

III. IMPLEMENTATION AND CODE DESIGN

This project runs from main page in which we choose the section. For creating this we create a class called ‘main’ and create the method ‘mainmethod’. In this method, we include the code of tkinter and similarly, we create the UI of customer and owner window.

In main window, we create the two buttons

1. Owner Section
2. Customer Section

In Tkinter this button can initialize onclick command and call the UI code of owner and customer.When we click on owner section onclick method is called and owner UI opens. This UI has button like add customer, response, show data show response.

In owner section, we create the button called as ‘add data’. On clicking it a different UI opens which contains the entryboxes where the owner can put information of customer. There are two buttons, add and cancel buttons. we import sqlite3 and addthe datain the table ‘user_data’. The cancel button closes the window of add data.

In menu section, owner can enter the information about today menu. In this, the popup window opens and gets the data of special dish and saves information in a file. In customer section the customer cansubmit the response.

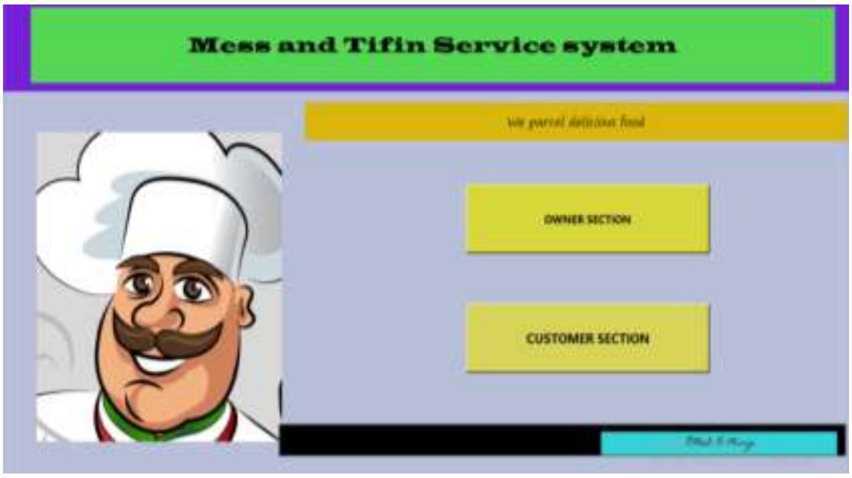
In show data section, we create a list view in which we can fetch all the data of customer from user_data table in to the “mess.db” database for owner’s purpose. Here weinclude the buttons delete and send mail. In delete button, we import sqlite and delete the data of selected customer. When we refresh the page, the list box gets updated.In mail button we import in sqlite in which it can fetch the email Id of customer and using SMTP module, we can send the message to customer via owner's email.

In customer section, we create login Id of each customer due to which the customer can login and access our data. If any wrong information is provided, then error message will popup. In this window, we have created the buttons formenus; whenever the owner adds the data of special dish the buttons can be updated. Thesebuttons help the customer to select their favorite dish from given dishes. If customer clicks the button then all menu buttons are disabled, and with their names and response the data is stored in 'response.txt' file and send to owner.Each customer gives only one response.The skip tiffin button allows the customer to skip the tiffin and thisresponseis also stored in 'response.txt' file. Using date comparison, the window has a message of days remaining this value is show in a form of label in the customer window.

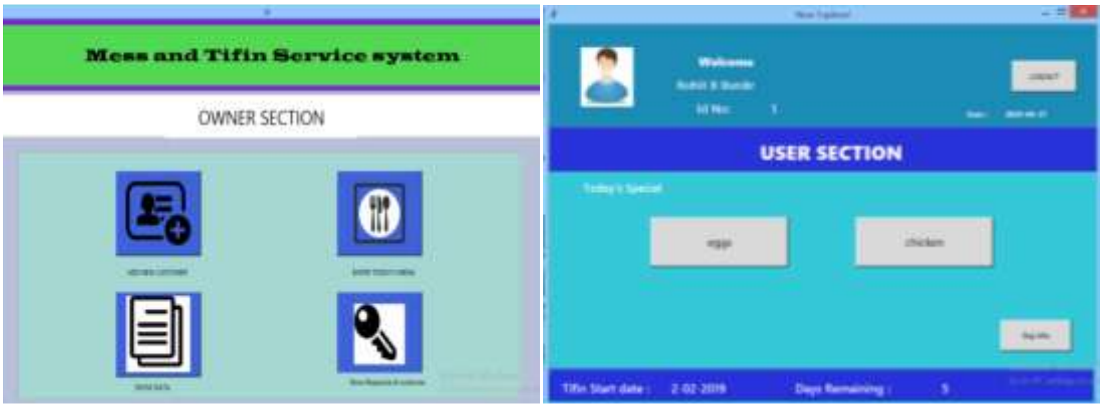
We have also used the 'error handling' mechanism to handle the different kinds of error. We have put validation on all entry boxes in Tkinter UI. If the user forgets to fill the information or enters irrelevant information then system givesan error message.

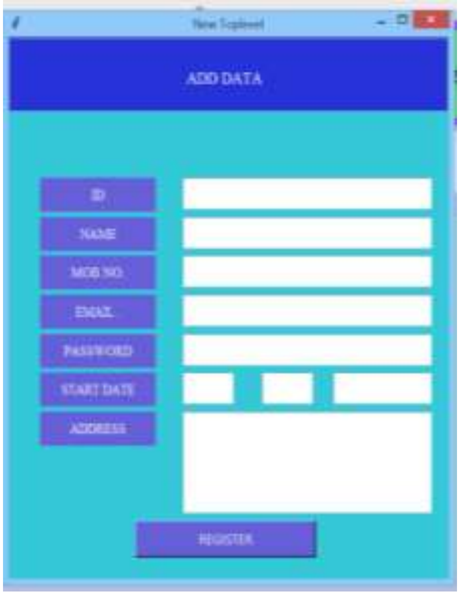
IV. PROJECT SCREENSHOTS

Main screen:

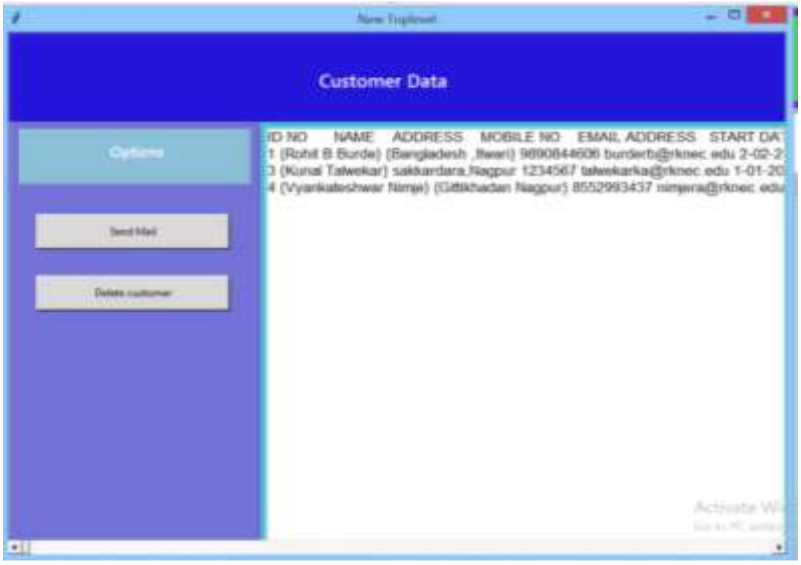


Owner and customer screen:



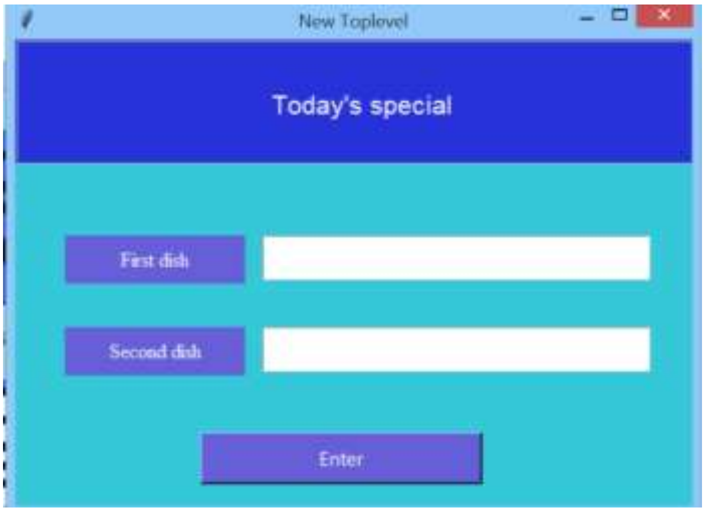


Add data



```
ID NO     NAME     ADDRESS     MOBILE NO     EMAIL ADDRESS     START DA
1 (Rohit B Burde) (Bangladesh ,Itwari) 9890844606 burderb@rknec.edu 2-02-2
3 (Kunal Talwekar) sakkardara,Nagpur 1234567 talwekarka@rknec.edu 1-01-20
4 (Vyankateshwar Nimje) (Gittikhadan Nagpur) 8552993437 nimjera@rknec.edu
```

Show data



[illegible]

Today'sspecial screenSend Mail to Customer

V. CONCLUSION AND FUTURE SCOPE

In this project we have implemented a role-based mess tiffin management system which efficiently manages the entire functionality of mess owner and customer. The owner can manage customer records and preferences easily whereas the customer can register and enjoy tiffin service based on personal preferences. It also handles the mess bill payment according to tiffin consumed by the customer. Food wastage is reduced as the customer orders only that food which he/she prefers to order from range of available menu items. The mess owner can prepare less food in case the customer decides to skip tiffin for a particular day. This will help reduce food wastage and save the owner's money. The tiffin delivery can be done at given address correctly.

Currently this has been developed as a desktop application. We aim to deploy this project in an Android Application as a Google Playstore app. In this era of smart phones if we deploy it on a portable device then many users can download and use it. In future, we are also planning to add functionality of sending SMS and WhatsApp messages along with real time calling.

BIBLIOGRAPHY

[1]. Allen Downey. (2012).“Think Python: How to think like a computer scientist (2nd edition).” Needham, Massachusetts: Green Tea Press.
[2]. David M. Beazley. (2009). “Python Essential Reference (4th edition).” United States of America: Pearson Education, Inc.
[3]. Abraham Silberschatz, Henry F. Korth, S. Sudarshan. (2011). “Database System Concepts (6th edition).” New York, America: The McGraw-Hill Companies, Inc.
[4]. Anirudh Rao. “GUI programming using Tkinter in Python.”*Edureka*.https://www.edureka.co/blog/tkinter-tutorial/
[5]. Don Rozenberg. “PAGE GUI generator.”*sourceforge*. http://page.sourceforge.net/
[6]. Article: “ListBox in Python.”java2. https://www.java2s.com/Code/Python/GUI-Tk/GetselectedvaluefromListbox.htm/
[7]. “The Python Tutorial.” Python Foundation. https://docs.python.org/3/tutorial/
[8]. “sqlite3 –DB-API 2.0 interface for SQlite Databases.” Python Foundationhttps://docs.python.org/3/library/ sqlite3.html

RiteshNimje" Design and Implementation of Mess Tiffin Management System in the Python Environment" International Journal of Computational Engineering Research (IJCER), vol. 09, no.7, 2019, pp 07-12