Python Group Project

Group F

Summer Semester 2021

Technology Campus Cham

QuickJabs: GUI for COVID-19 Vaccination Appointment using Tkinter

Submitted By:

Farouk Fedila	**MMC - 00803600**
Michael Bottenhofer	**MMC - 00714048**
Naseeb Nasarudeen	**MMC - 00811688**
Salumon Purushan	**MMC - 00098232**

Advisor:

Prof. Johannes Kigele

INDEX

What is QuickJabs?

Quick Jabs is a streamlined Vaccination booking application that links to the centralized vaccination centres directory to provide the public with an easy means of booking, modifying and viewing their vaccination appointments. It skips the waiting period and slot unavailability issues usually present in the centralized website.

This application intends to provide the user with a simplified Interface and easy access to available vaccinations in the selected state.

What can you do with QuickJabs?

A user can enter their basic details, choose a vaccination brand and select a state in which they wish to be vaccinated. The application then returns a list of hospitals that provide the vaccine of choice and displays a calendar to select dates for the vaccination appointment.

Prerequisites to run QuickJabs:

This program was coded using Python 3.8.5 and works without any errors when using the aforementioned version or above.

Before executing the program, it is necessary to install the TKCALENDER module and the SQLITE3 module. The instructions for the same are given below.

1. Open command terminal (windows key + R , then type ‘cmd’ and hit ENTER)
2. Type the following into the terminal
    a. pip install tkcalendar -> wait till it is done.
    b. pip install pysqlite3 -> wait till it is done.
3. The Calendar module and SQLite3 are now installed.
4. Before executing ‘main.py’, execute ‘createdatab.py’ to create the internal database.

Various Functions of QuickJabs:

1.New Registration for Vaccination

A simple registration form input basic user data such as FIRST NAME, LAST NAME and AGE in the form of entry fields.

SEX, PREFERRED VACCINE and PREFERRED STATE as drop-down selectable menus.

DATE for vaccination as a selectable calendar

TIME OF VACCINATION as a choice between Forenoon (8:00 – 12:00) and Afternoon (13:00 – 17:00).

The user will only be able to select a date from the current day until the next 45 days. All other dates are non-selectable.

The entered data is saved into an internal database by clicking the Submit button. The First name of the user is used as the search keyword to access information in MODIFY and VIEW functions.

Exception Handling:

The first and last name fields require plain text and the program will display a pop-up error message if any of the fields are left as blank.

2.Search Vaccination Centres

A list of Hospitals/Vaccination Centres is displayed in this tab. The list contains names of centres and the name of the Vaccine brand provided there. This database of Hospitals are also used as a basis for selecting the preferred state. It is available as a separate .csv file.

3.Modify Appointments

This page shows a field to enter the first name of an existing customer after a new registration.

Clicking the 'Search' button after entering the first name shows the details of the user which was entered in the 'New Registration' tab. This information can then be fully edited.

The user can modify the existing data by clicking the 'Submit' button, which prompts a pop-up confirmation of the intended action.

Exception Handling:

If the entered name is not present in the database, a pop-up dialog box will prompt the user showing 'No data found in Database'.

If the first name is not entered the SEARCH button is clicked, a pop-up dialog box will show 'No name entered'.

If all the fields are not filled in and the SUBMIT button is clicked, a pop-up dialog box will display 'Please fill in all required fields'.

4.View existing Appointments

This page shows a field to enter your first name. Upon clicking the 'Search First Name' button, data regarding the user's current appointment is displayed for their future reference.

Exception Handling:

If the entered name is not present in the database, a pop-up dialog box will prompt the user to check the entered data.

If the SEARCH FIRST NAME button is clicked without entering a name, a pop-up dialog box will display ‘No name entered’.

Additional Functions

5.Info and Contact:

Clicking this button displays a pop-up dialog box showing the version of the software, product licence, stock image credit notes and details of the developers.

6.Exit

Button to close the application which serves the same function as the CLOSE button on the upper right hand corner. Upon clicking, the user is prompted with a confirmation dialog box to close the application. Selecting ‘Yes’ closes the application while selecting ‘No’ returns to the previous page.

Function in Detail:

Registration Form

This is the second button user can find on GUI. Here he/she can register for the vaccine appointment. The input fields of the registration form are as follows,

First Name

Here the User can add his first name and it will be stored in the database as text .

Last Name

Last name of the user will also be saved as text in the second column named as last in the database

Sex

The gender of the user is also saved as text.

Age

The age of the user is saved as integer

Vaccine Prefered

Here the user can select his preferred vaccine from the selection list

State

User can select his preferred state for his/her vaccination

Select Date

It is a Tkinder calendar module where the user can his/her date for vaccination.The calendar will show from the current date also Tkcalendar should be imported to enable the calendar for selecting the date.

Select Time

It is a selection list with two options : Forenoon and Afternoon**.**

The database

The database is created "customer.db" using sqlite3 and inputs are saved in the table "customer" in the database and will be stored in each column in the table. The variables to store the inputs (above mentioned as respectively) are 'first' , 'last', 'sex', 'age', 'vaccine', 'state', 'dateT', and 'time'. In order to create the database sqlite3 should be imported *(import sqlite3).*

"conn = sqlite3.connect("customers.db")"

The database 'customers 'has been created..

Images of program operation :

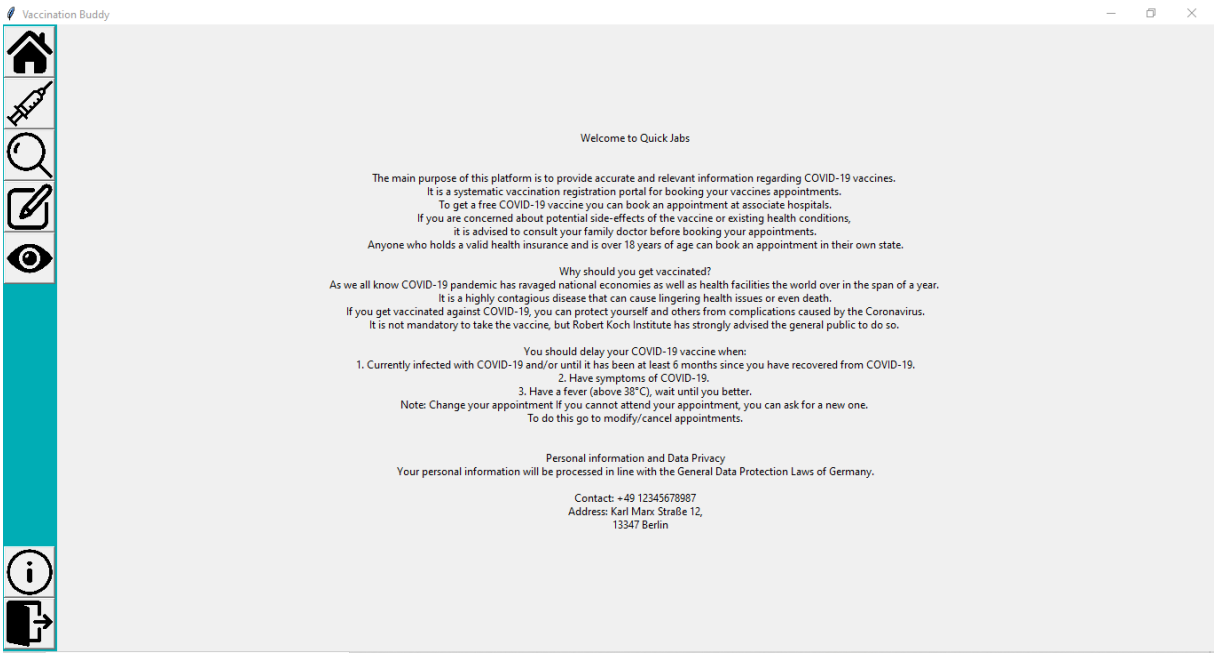

Figure 01: Home page.

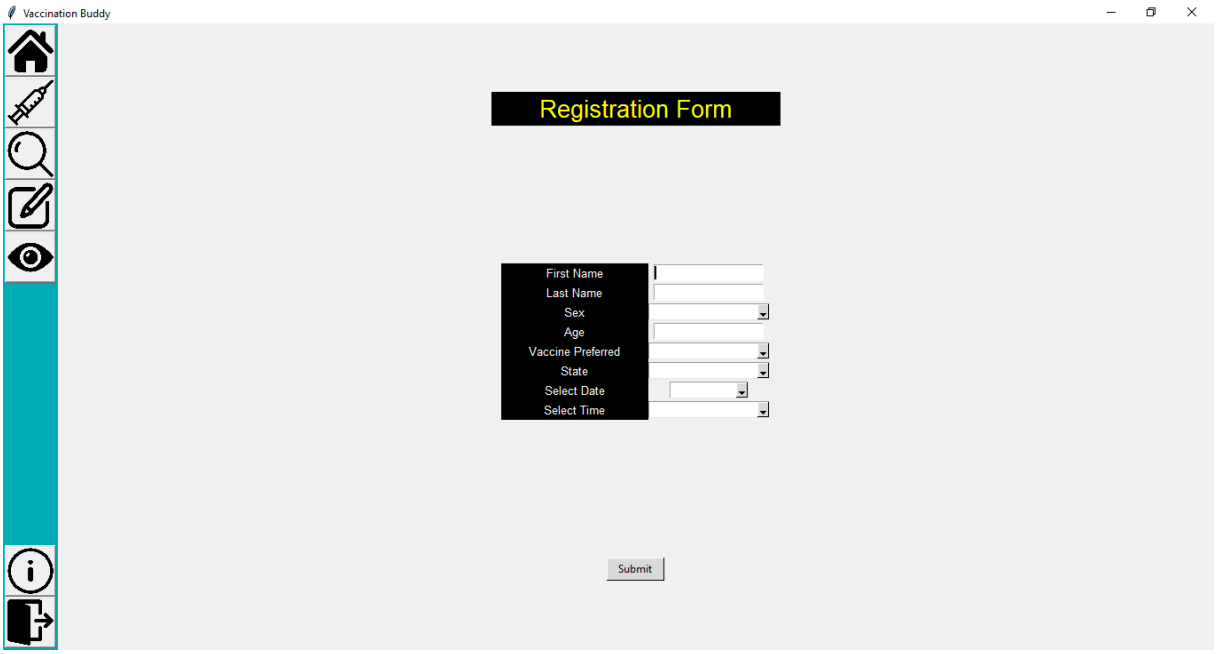

Figure 02: New Registration Form.

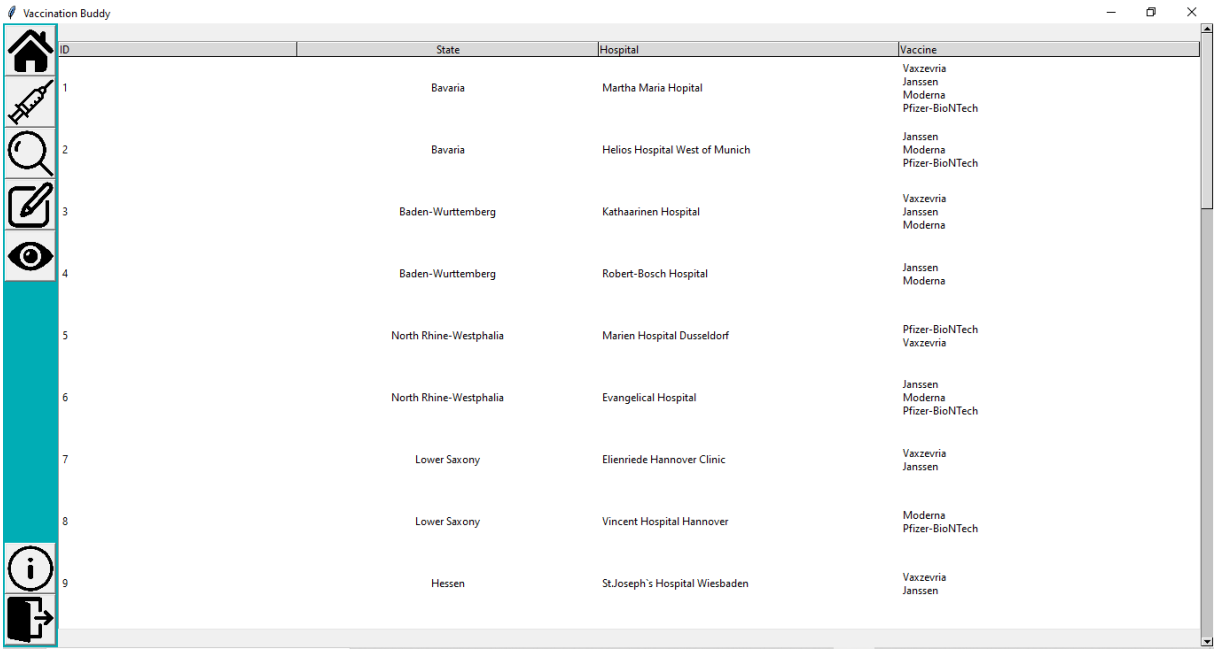

Figure 03: Search Registration Center.

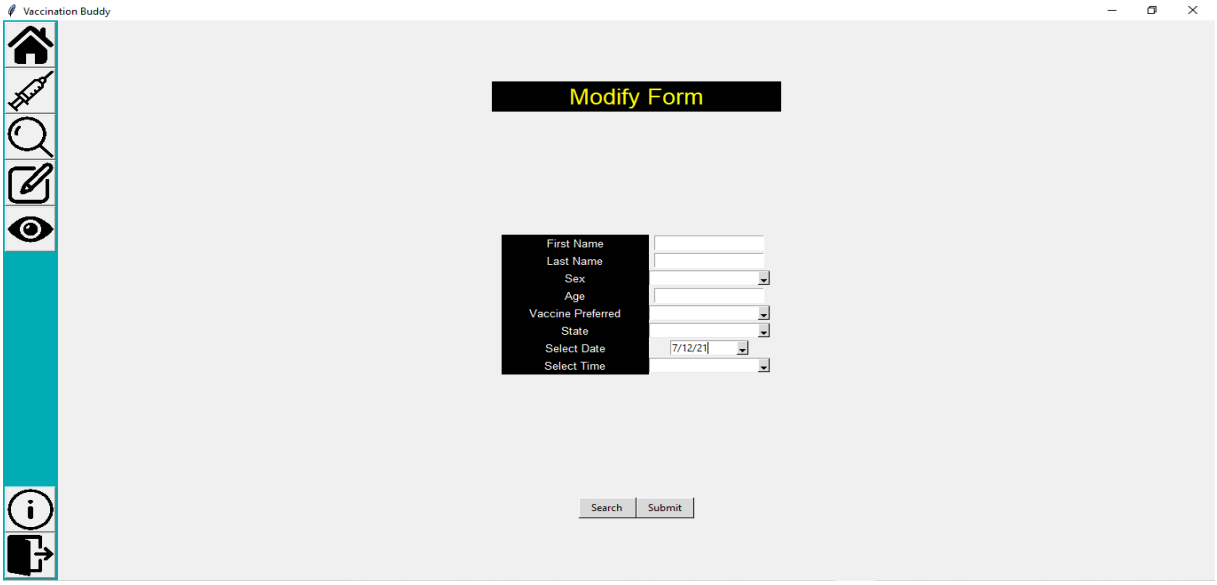

Figure 04: Modify Appointment Form.

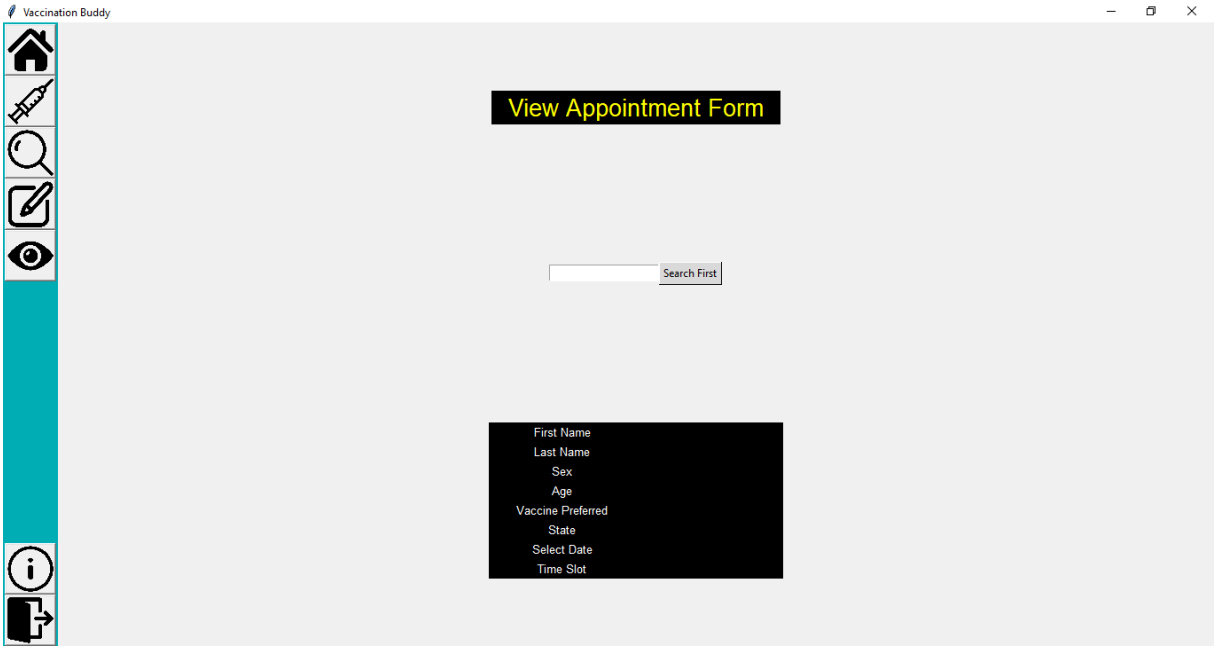

Figure 05: View Appointment Form.

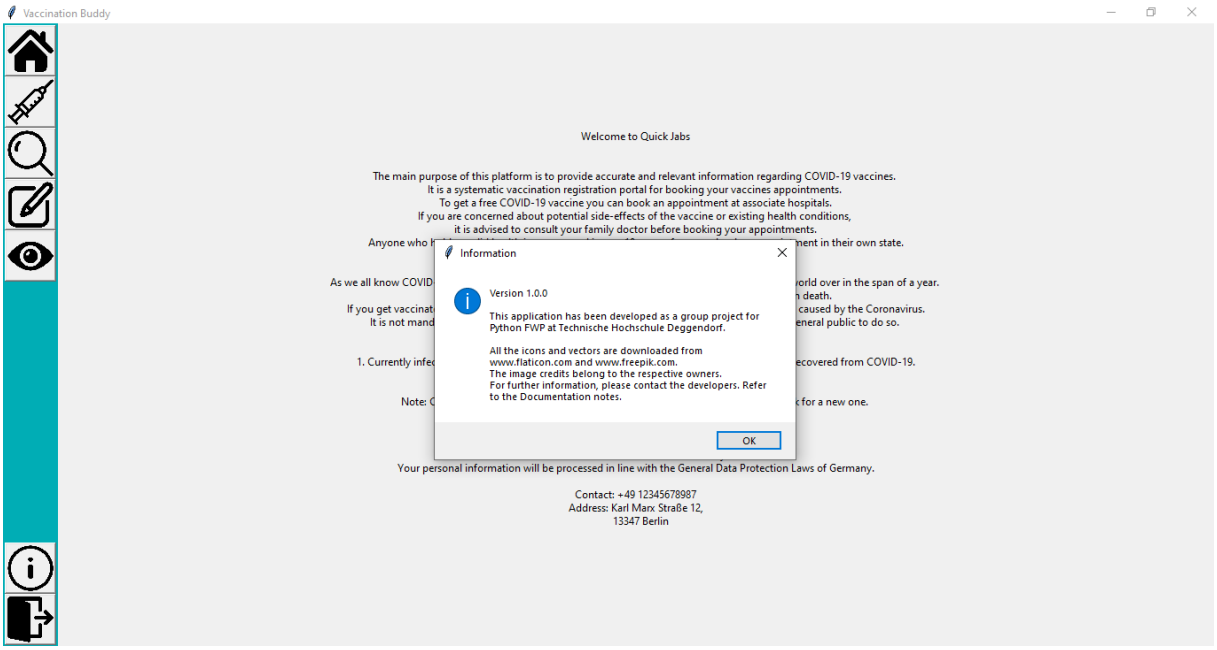

Figure 06: Information and Contact.

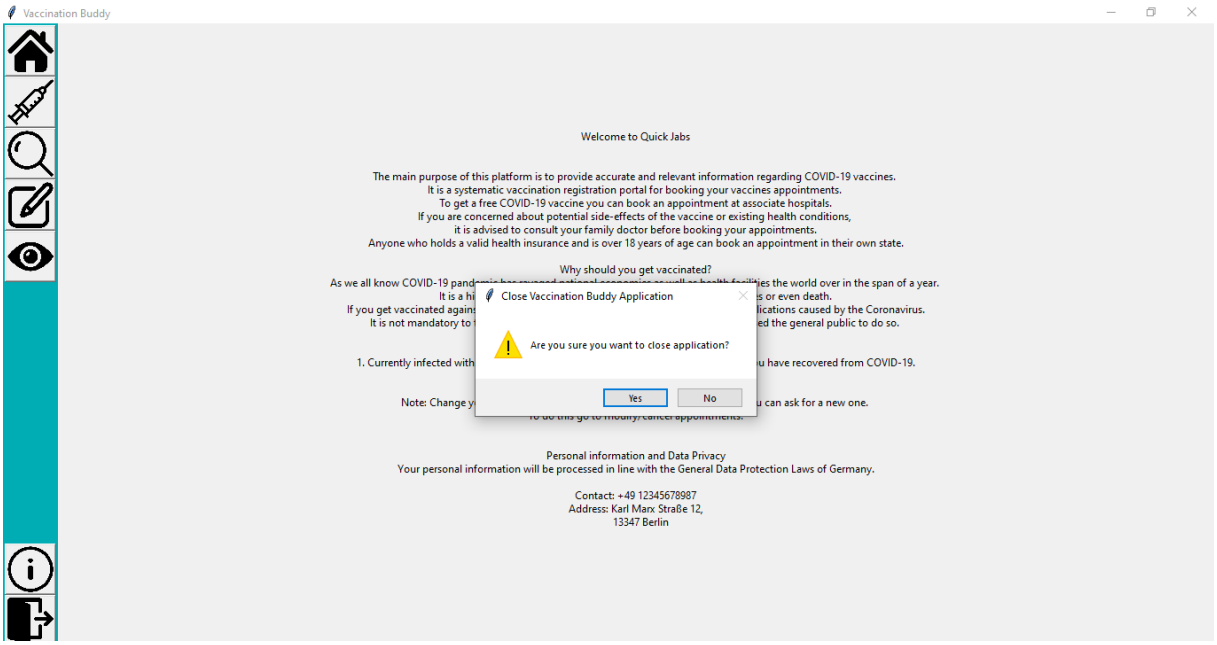

Figure 07: Exit button

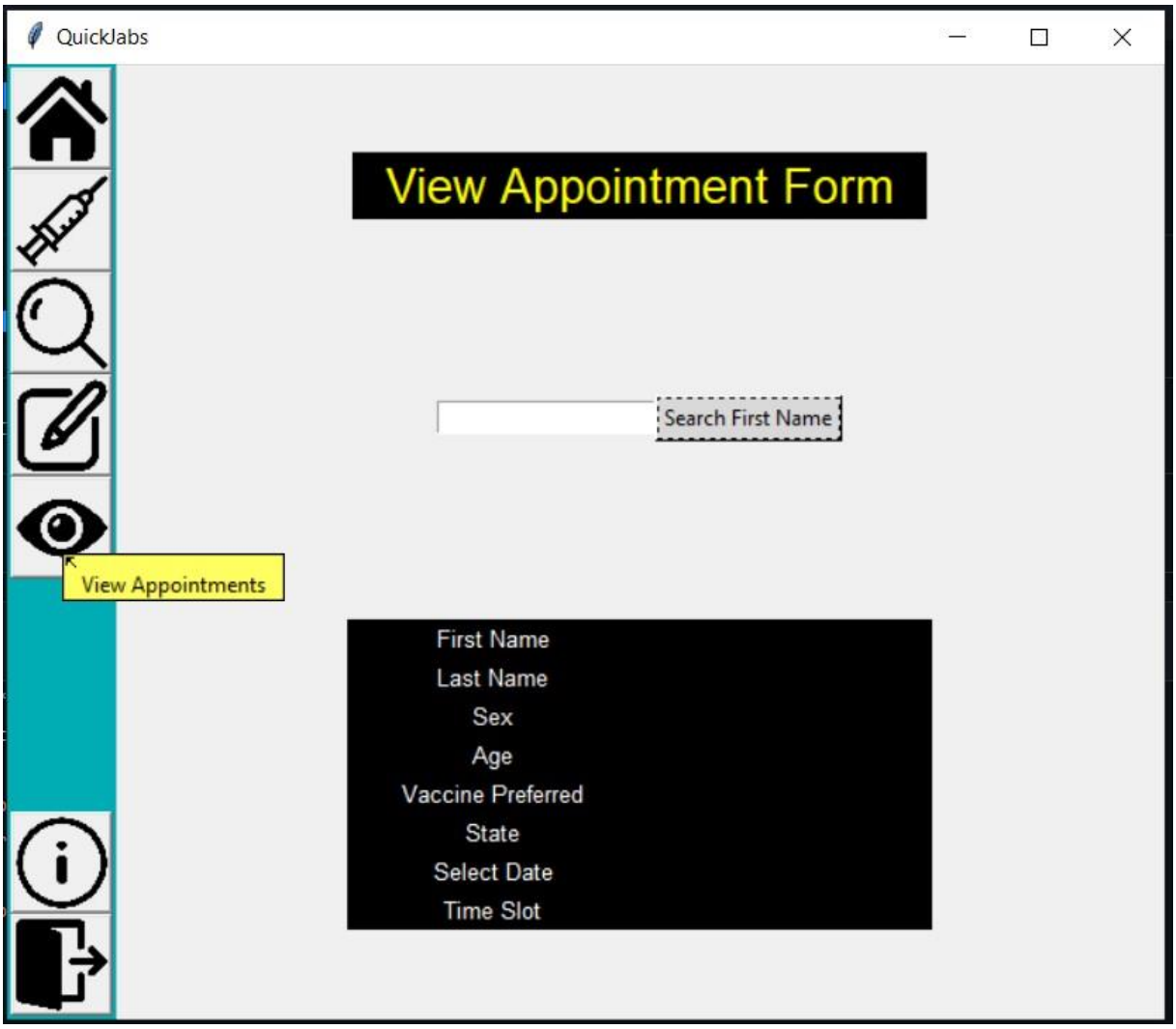

Figure 08: Cursor hover display

Exception Handling:

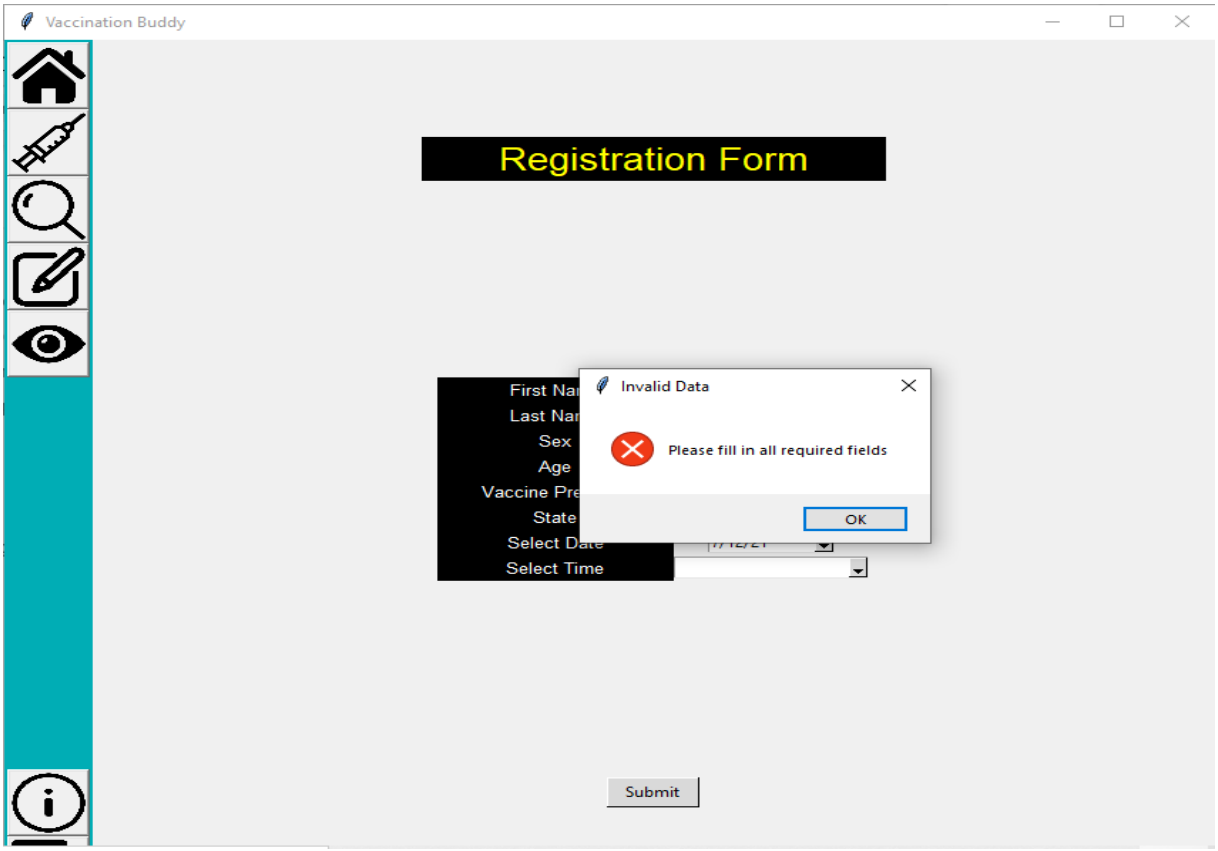

Figure 09: Error message no data entered or field not filled

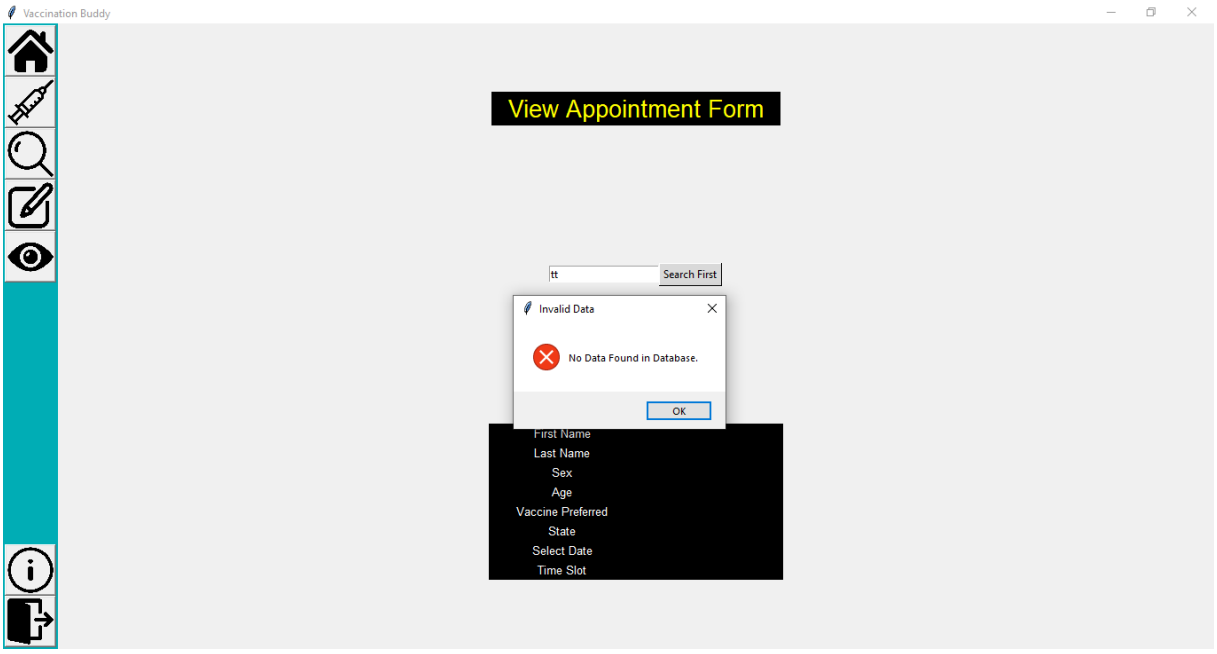

Figure 10: Message box name is not present in the database