

JHI: PASA:

Kirtan Shrestha BSc. (Hons.)

Computing, Software College of IT and E-commerce, Coventry University

ST4008CEM Computing Activity LED Learning Project 1

Giriraj Rawat

February 23, 2023

Contents

JHI: PASA:	3
Introduction	3
Login Page	3
Figure 1	3
Figure 2	4
Sign Up page.....	5
Figure 3	5
Figure 4	6
Back End.....	7
Figure 5	7
Figure 6	8
GitHub Commits	8
Figure 7	8
Conclusion	9

JHI: PASA:

Introduction

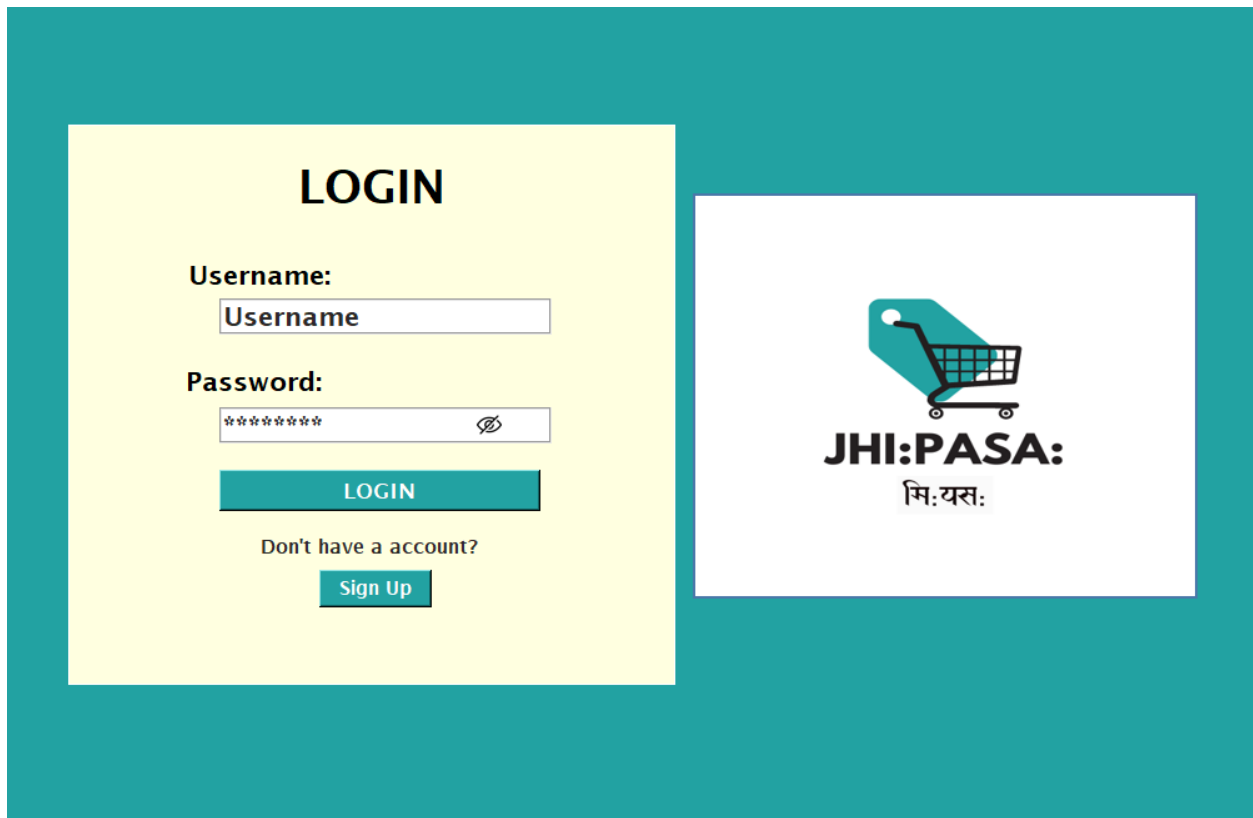
The main purpose of making this front end is to build a simple, user friendly, and interactive “Sign-in and Login” GUI for customer using Python Tkinter tools.

Login Page

Login page is the first page which user sees after running the application. The user who has registered in the application, they can directly enter their username and password to login the Small Mart system. If the username and password which were matched the record stored in the database, then they will grant access. If the username and password were don’t match with the record stored in the database, it will show a message box with “Invalid Credentials”.

Figure 1

Login page



As we can see there is three features in login page they are “Login”, “Sign Up” and “Eye button”. Login button is used to login to the main dashboard. Sign Up button is used to sign up our account if you don’t have an account. Eye button is used to “hide and show” our password.

Figure 2

Login code

```
104 #labels and entry-----
105 login_lb=Label(logon,text="LOGIN",font=my_font2,bg="light yellow",fg="black")
106 login_lb.place(x=180,y=25)
107
108 user=Label(logon,text="Username:",font=my_font1,bg="light yellow",fg="black")
109 user.place(x=92,y=105)
110
111 user_entry=Entry(logon,bg="white",font=my_font1,fg="#2B2828",width=20,borderwidth=2,relief=GROOVE)
112 user_entry.insert(0,"Username")
113 user_entry.place(x=120,y=138)
114 user_entry.bind("<FocusIn>","del1")
115
116 #password-----
117 pw=Label(logon,text="Password:",font=my_font1,bg="light yellow",fg="black")
118 pw.place(x=90,y=190)
119
120 pw_entry=Entry(logon,bg="white",font=my_font1,fg="#2B2828",width=20,borderwidth=2,relief=GROOVE,show="*")
121 pw_entry.insert(0,"Password")
122 pw_entry.place(x=120,y=225)
123 pw_entry.bind("<FocusIn>","del2")
124
125 #verify function-----
126 def verify():
127     a=user_entry.get()
128     b=pw_entry.get()
129     if (a==" " or a=="Username:") or (b==" " or b=="Password"):
130         messagebox.showerror("Login","Error.")
131     else:
132         check()
133
134 a=Image.open('show.png')
135 a1=a.resize((20,20))
136 show_img=ImageTk.PhotoImage(a1)
137
138 b=Image.open('hide.png')
139 b1=b.resize((20,20))
140 hide_img=ImageTk.PhotoImage(b1)
141
142
143
144 #hide/show functions
145 def hide():
146     show_btn=Button(logon,image=show_img,command=show,bg="white",borderwidth=0,activebackground="#CBD8ED")
147     show_btn.place(y=228,x=325)
148     pw_entry.config(show="")
149
150 def show():
151     hide_btn=Button(logon,image=hide_img,command=hide,bg="white",borderwidth=0,activebackground="#CBD8ED")
152     hide_btn.place(y=228,x=325)
153     pw_entry.config(show="*")
```

Sign Up page.

Sign up page is the second page where user can create their account if they have not created an account in our application. User should fill all the details shown below to create an account to our application.

As we can see there is two button one is “Sign in and Login” after filling all details we can sign in it. If you miss to fill one entry box also it will show us the “Error”. After creating the account, you can click the login button to go the first page.

Figure 3

Sign up page.

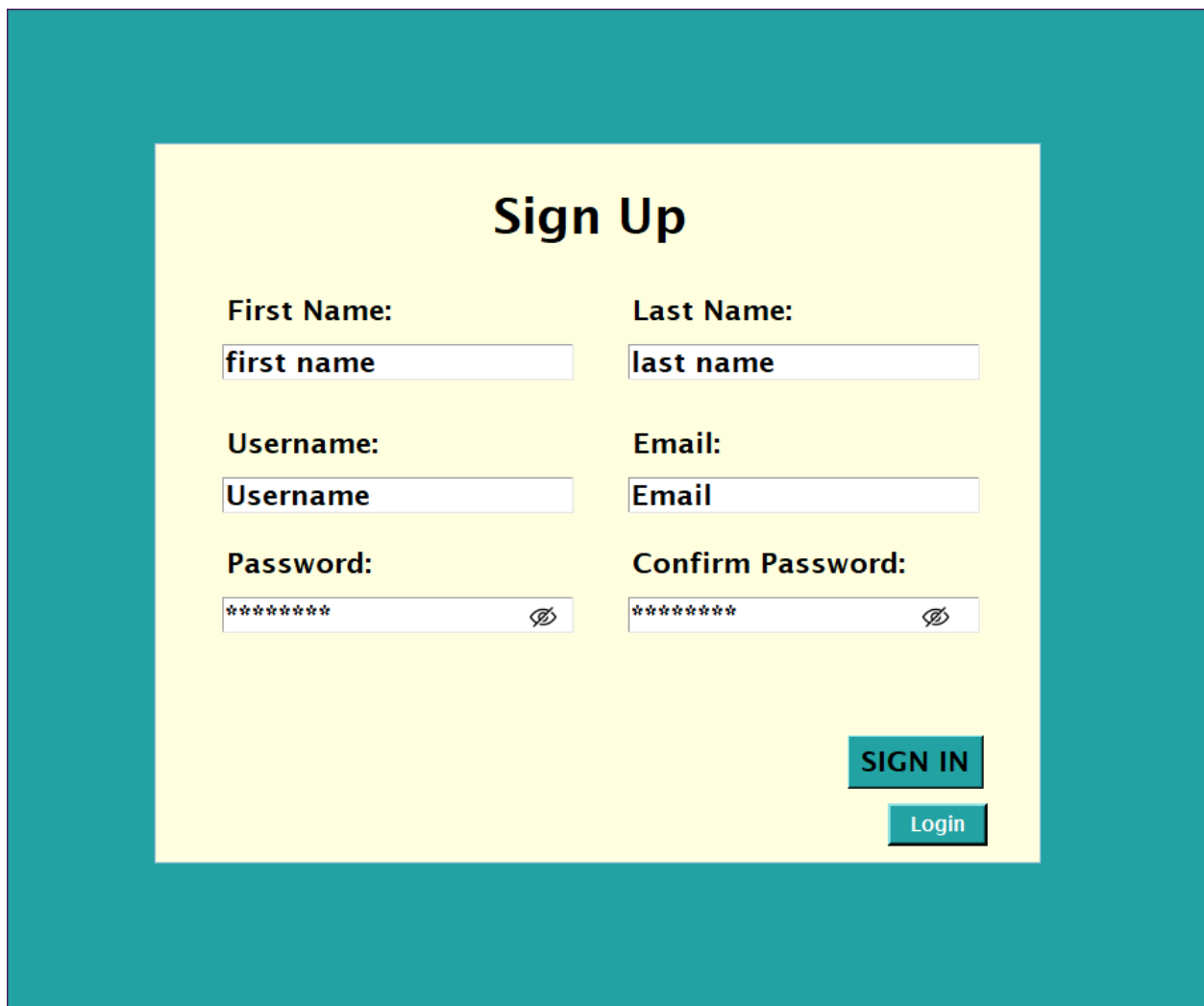
The image shows a 'Sign Up' form centered on a teal background. The form itself has a light yellow background and is titled 'Sign Up' in a large, bold, black font. Below the title, there are six input fields arranged in two columns. The left column contains 'First Name:' with a text box containing 'first name', 'Username:' with a text box containing 'Username', and 'Password:' with a text box containing '*****' and an eye icon. The right column contains 'Last Name:' with a text box containing 'last name', 'Email:' with a text box containing 'Email', and 'Confirm Password:' with a text box containing '*****' and an eye icon. At the bottom right of the form, there are two buttons: a teal button with white text 'SIGN IN' and a teal button with white text 'Login'.

Figure 4

Code of Signup

```
f_name=Label(frame_r,text="First Name: ",font=my_font1,bg="light yellow",fg="black")
f_name.place(x=50,y=110)
f_entry=Entry(frame_r,font=my_font1,bg="white",fg="black")
f_entry.insert(0,"first name")
f_entry.place(x=50,y=150)
f_entry.bind("<FocusIn>",del1)

l_name=Label(frame_r,text="Last Name: ",font=my_font1,bg="light yellow",fg="black")
l_name.place(x=355,y=110)
l_entry=Entry(frame_r,font=my_font1,bg="white",fg="black")
l_entry.insert(0,"last name")
l_entry.place(x=355,y=150)
l_entry.bind("<FocusIn>",del2)

u_name=Label(frame_r,text="Username: ",font=my_font1,bg="light yellow",fg="black")
u_name.place(x=50,y=210)
u_entry=Entry(frame_r,font=my_font1,bg="white",fg="black")
u_entry.insert(0,"Username")
u_entry.place(x=50,y=250)
u_entry.bind("<FocusIn>",del3)

email=Label(frame_r,text="Email: ",font=my_font1,bg="light yellow",fg="black")
email.place(x=355,y=210)
email_try=Entry(frame_r,font=my_font1,bg="white",fg="black")
email_try.insert(0,"Email")
email_try.place(x=355,y=250)
email_try.bind("<FocusIn>",del4)

pw=Label(frame_r,text="Password: ",font=my_font1,bg="light yellow",fg="black")
pw.place(x=50,y=300)

pw_entry=Entry(frame_r,font=my_font1,bg="white",fg="black",show="*")
pw_entry.insert(0,"password")
pw_entry.place(x=50,y=340)
pw_entry.bind("<FocusIn>",del5)

con_pw=Label(frame_r,text="Confirm Password: ",font=my_font1,bg="light yellow",fg="black")
con_pw.place(x=355,y=300)

cpw_try=Entry(frame_r,font=my_font1,bg="white",fg="black",show="*")
cpw_try.insert(0,"re-enter")
cpw_try.place(x=355,y=340)
cpw_try.bind("<FocusIn>",del6)
```

Back End

In the back end the data which were written in the sign-up page were stored in database. It stored the value permanently and only the admin can delete or edit it. In backend import sqlite3 to connect and create a database where all the records were stored.

Figure 5

Backend code

```
def form_page():
    try:
        log=sqlite3.connect('register.db')
        log1=log.cursor()
        log1.execute("""CREATE TABLE signup(
            f_name text,
            l_name text,
            u_name PRIMARY KEY,
            email varchar,
            pw varchar
        )
        """)
        log.commit()
        log.close()
    except:
        pass

def login_page():
    reg_p.destroy()
    import loginreal
```

Figure 6
Data store in database

SQL ▾				
< 1 / 1 > 1 - 1 of 1				
f_name	l_name	u_name	email	pw
Kirtan	Shrestha	kirtan1232	shresthakirtan4@gmail.com	9862242899

GitHub Commits

Figure 7
Commits

main ▾				
Commits on Feb 20, 2023				
Add files via upload	kirtan1232 committed 1 hour ago	Verified	d962510	<>
Commits on Jan 24, 2023				
Add files via upload	kirtan1232 committed last month	Verified	ef23f3b	<>
Add files via upload	kirtan1232 committed last month	Verified	84f9944	<>
Commits on Dec 29, 2022				
Add files via upload	kirtan1232 committed on Dec 29, 2022	Verified	0691732	<>
Delete README.md	kirtan1232 committed on Dec 29, 2022	Verified	c294fb1	<>
Add files via upload	kirtan1232 committed on Dec 29, 2022	Verified	c3ef2a2	<>
Commits on Dec 22, 2022				
Update README.md	SuvamKhadgi committed on Dec 22, 2022	Verified	e336856	<>
Update README.md	kirtan1232 committed on Dec 22, 2022	Verified	1cc4ef2	<>
Initial commit	shresthanikhil16 committed on Dec 22, 2022	Verified	c12c8a8	<>

Conclusion

The goal of our application is to make sure the customer can see which products are available in our shop. They won't need to visit our shop to check out the products which they want to buy. They can check our application by staying at home and add the product into their cart and purchase it. It can save the time of people and also they won't need to search our mart location. It also saved the bill of the customer which incase if they have any problem with our product they can contact us.

I feel that in terms of coding, I was not able to keep up with my fellow team members, but I was able to calmly gather all the required information necessary for this application and was able to systematically arrange them to help others.