```
#-----TMPORT
 1
                                                                                         \Box
 2
3
    from tkinter import *
4
    import tkinter.font as font
    from PIL import Image, ImageTk
 5
    import sqlite3
6
 7
    from tkinter import messagebox
8
    import serial
    import bluetooth
9
    import sys
10
11
    import os
12
    #for converting to windows software
13
    def resource_path(relative_path):
14
        """ Get absolute path to resource, works for dev and for PyInstaller """
15
16
        try:
            # PyInstaller creates a temp folder and stores path in _MEIPASS
17
            base_path = sys._MEIPASS2
18
19
        except Exception:
            base_path = os.path.abspath(".")
20
21
22
        return os.path.join(base_path, relative_path)
23
24
25
    #------REGISTRATION
                                                                                         \Box
26
27
    #saves the registered values into the database
    def register_user():
28
29
        name_info=name.get()
        age_info=age.get()
30
        conn_type_of_devices_info=conn_type_of_devices.get()
31
32
        conn_type_of_devices_info_1=conn_type_of_devices_1.get()
        gender_info=gender.get()
33
        username_info=username.get()
34
        password_info= password.get()
35
36
        #software.db contains registered info
37
        conn= sqlite3.connect(resource_path("software.db"))
38
        c=conn.cursor()
39
        c.execute("INSERT INTO person
40
        VALUES('"+name_info+"',"+age_info+",'"+conn_type_of_devices_info+"','"+conn_type_o J
        f_devices_info_1+"','"+gender_info+"','"+username_info+"','"+password_info+"')")
        messagebox.showinfo("Information", "Your record is saved!")
41
```

```
42
         conn.commit()
43
         conn.close()
44
         conn= sqlite3.connect(resource_path("user_info.db"))
45
         c=conn.cursor()
         c.execute("INSERT INTO Curr_session(name) VALUES(("+name_info+")")")
46
47
         conn.commit()
48
         conn.close()
49
         name_entry.delete(0,END)
50
         age_entry.delete(0,END)
51
52
         conn_devices_entry.deselect()
         conn_devices_entry1.deselect()
53
         gen_entry_m.deselect()
54
         gen_entry_f.select()
55
         username_entry.delete(0,END)
56
57
         password_entry.delete(0,END)
58
         Label(screen1, text="Registeration is Successful", fg="green",
59
                                                                                                 \Box
         font=("Calibri", 11)).pack()
60
61
62
     #these functions delete the screen
63
     def delete_s1():
64
         screen1.destroy()
65
66
     #for the registration page
67
     def register():
68
         global screen1
69
         screen1= Toplevel(screen)
70
         screen1.title("Register")
         screen1.geometry("400x650")
71
72
         screen1.configure(bg="#FFFEF2")
73
74
         global name
75
         global age
76
         global conn_type_of_devices
         global conn_type_of_devices_1
77
78
         global gender
79
         global username
         global password
80
81
82
         global name_entry
         global age_entry
83
         global conn_devices_entry
84
85
         global conn_devices_entry1
```

```
86
          global gen_entry_m
 87
          global gen_entry_f
 88
          global username_entry
 89
          global password_entry
 90
 91
          name=StringVar()
 92
          age=StringVar()
 93
          conn_type_of_devices=StringVar()
 94
          conn_type_of_devices_1=StringVar()
 95
          gender=StringVar()
 96
          username= StringVar()
 97
          password= StringVar()
 98
 99
          Label(screen1, text="Please Enter details below
                                                                                                 7
          ", width="400", height="3", bg="#caf0f8", font=("Times", 16)).pack()
100
101
          Label(screen1, text="", bg="#FFFEF2").pack()
102
103
          Label(screen1, text="Enter Name",font=("Times",13),bg="#FFFEF2").pack()
104
105
          name_entry= Entry(screen1,textvariable=name, width=30)
106
          name_entry.pack()
107
108
          Label(screen1, text="Enter Age",font=("Times",13),bq="#FFFEF2").pack()
109
110
          age_entry= Entry(screen1,textvariable=age, width=30)
111
          age_entry.pack()
112
113
          Label(screen1, text="Select types of
                                                                                                 \Box
          connection", font=("Times", 13), bg="#FFFEF2").pack()
114
115
          conn_devices_entry= Checkbutton(screen1,
                                                                                                 7
          text="Wifi", variable=conn_type_of_devices, bg="#FFFEF2")
          conn_devices_entry.deselect()
116
117
          conn_devices_entry.pack()
118
          conn_devices_entry1= Checkbutton(screen1,
119
                                                                                                 \Box
          text="Bluetooth", variable=conn_type_of_devices_1, bg="#FFFEF2")
120
          conn_devices_entry1.deselect()
121
          conn_devices_entry1.pack()
122
123
          Label(screen1, text="Select gender", font=("Times", 13), bg="#FFFEF2").pack()
124
125
          gen_entry_m= Radiobutton(screen1, text="Male", variable=gender,
                                                                                                 \Box
          value="Male", bg="#FFFEF2")
```

```
126
          gen_entry_m.deselect()
127
          gen_entry_m.pack()
128
129
          gen_entry_f= Radiobutton(screen1, text="Female",variable=gender,
                                                                                               \Box
          value="Female", bg="#FFFEF2")
          gen_entry_f.select()
130
131
          gen_entry_f.pack()
132
          Label(screen1, text="", bg="#FFFEF2").pack()
133
          Label(screen1, text="Enter Username", font=("Times", 13), bg="#FFFEF2").pack()
134
135
136
          username_entry= Entry(screen1,textvariable=username, width=30)
          username_entry.pack()
137
138
139
          Label(screen1, text="Enter Password", font=("Times", 13), bg="#FFFEF2").pack()
140
          password_entry= Entry(screen1, textvariable=password, width=30)
141
142
          password_entry.pack()
143
144
          Label(screen1, text="", bg="#FFFEF2").pack()
145
          Button(screen1, text="Register", bg="#d1ffea", height=2, width= 20, command=
146
                                                                                               \Box
          register_user).pack()
147
          Label(screen1, text="", bg="#FFFEF2").pack()
148
149
          Button(screen1, text="Direct to Login Page", bg="#d1ffea", height=2, width= 20,
                                                                                               \Box
          command= login).pack()
150
          Label(screen1, text="", bg="#FFFEF2").pack()
151
152
          Button(screen1, text="Close window", bg="#ffcccb", height=2, width= 20,
                                                                                               \Box
          command=delete_s1).pack()
153
154
155
                                  -----DASHBOARD------
156
      #these functions delete the screens
157
      def delete_s4():
158
          screen4.destroy()
159
      def delete_s5():
160
          screen5.destroy()
161
      def delete_s8():
162
          screen8.destroy()
163
164
      def delete_s9():
165
          screen9.destroy()
```

```
166
      1.1.1
167
168
      #not fully complete
169
      #purpose- saving the bluetooth info to new database
170
171
      def save_info():
172
          conn= sqlite3.connect(resource_path("user_info.db"))
173
          c=conn.cursor()
          # c.execute("INSERT INTO Curr_session(bluetooth_name, bluetooth_address)
174
                                                                                                  7
          VALUES('"+addr+"', '"+name+"')")
175
          c.execute("UPDATE Curr_session SET bluetooth_name='"+addr+"',
176
                                                                                                  \Box
          bluetooth_address='"+name+"' WHERE name='"+username1+"'")
          # messagebox.showinfo("Information", "Your record is saved!")
177
178
          conn.commit()
          conn.close()
179
180
          conn= sqlite3.connect(resource_path("user_info.db"))
          c=conn.cursor()
181
          c.execute("SELECT * FROM Curr_session")
182
          r=c.fetchall()
183
          for i in r:
184
185
              print(i)
          # messagebox.showinfo("Information", "Your record is saved!")
186
187
          conn.commit()
          conn.close()
188
      1.1.1
189
190
191
      #Scanning nearby bluetooth devices
192
193
      def button_clicked1():
          # print("success")
194
195
          # Label(screen9,text="hi").pack()
196
197
          #print("Scanning")
198
          Label(screen9, text="Scanning", width="400", height="3",
                                                                                                  \Box
          font=("Times", 13, "bold")).pack()
199
200
          nearby_devices = bluetooth.discover_devices(lookup_names=True)
201
202
          #print("Found {} devices.".format(len(nearby_devices)))
203
          Label(screen9, text="Found {}
          devices.".format(len(nearby_devices)), width="400", height="3",
                                                                                                  \overline{a}
          font=("Times", 11, "bold")).pack()
204
205
          for addr, name in nearby_devices:
```

7

 $\overline{\downarrow}$ 

 $\Box$ 

7

```
#print(" {} - {}".format(addr, name))
206
              Label(screen9, text=" {} - {}".format(addr, name), width="400", height="3",
207
              font=("Times", 12, "bold")).pack()
208
209
210
      #Scan devices page
211
212
      def scan_devices():
213
          global screen9
214
          screen9=Toplevel(screen)
          screen9.title("Scanning for nearby Bluetooth devices")
215
          screen9.geometry("400x500")
216
217
          Label(screen9, text="Nearby Bluetooth devices", font=("Times", 16)).pack()
218
          Label(screen9, text="").pack()
          Button(screen9, text="Scan Devices", height=1, width= 30,
219
          bg="#0077b6", font=("Times", 13, "bold"), command=button_clicked1).pack()
220
          # Button(screen9, text="Save", height=1, width= 30,
          bg="#CD5C5C", font=("Times", 11, "bold"), command=save_info()).pack()
221
          #Label(screen9, text="hi").pack()
          #Label(screen9, text="").pack()
222
          Button(screen9, text="Exit", height=1, width= 30,
223
          bg="#CD5C5C", font=("Times", 13, "bold"), command=delete_s9).pack(side=BOTTOM)
224
225
226
227
      #Page for Dashboard
228
229
      def session():
230
          # delete_s()
231
          delete_s2()
232
          global screen8
233
          screen8= Toplevel(screen)
234
          screen8.title("Dashboard")
235
236
          screen8.geometry("400x400")
237
          Label(screen8, text="Welcome to the Dashboard", font=("Times", 16)).pack()
238
239
          Button(screen8, text="Sign_out", bg="#ffcccb", command=delete_s8).place(x=340, y=10)
240
241
          conn= sqlite3.connect(resource_path("software.db"))
242
          c1=conn.cursor()
          c1.execute("SELECT name, gender FROM person WHERE username='"+username1+"'")
243
244
          r1=c1.fetchall()
245
246
          for i in r1:
```

```
/home/kshitij/Downloads/Tkinter_Loginpage-main/slr.py
Page 7 of 11 Wed 09 Aug 2023 02:10:12 PM IST
```

```
247
              print(i[0])
              print(i[1])
248
249
              if(i[1]=="Female"):
250
                  Label(screen8, text="Hello! Ms. {}".format(i[0]), font=("Times", 14)).pack()
                  # print("Ms. {}".format(i[0]))
251
252
              else:
                  Label(screen8, text="Hello! Mr. {}".format(i[0]),font=("Times",14)).pack()
253
254
                  # print("Mr. {}".format(i[0]))
255
256
          conn.commit()
257
          conn.close()
258
259
          Label(screen8, text="").pack()
260
          Button(screen8, text="Your Saved Devices", width=20, height=2, bg="#d1ffea").pack()
261
          Label(screen8, text="").pack()
262
263
          Button(screen8, text="Scan Nearby
                                                                                                 7
          Devices", width=20, height=2, bg="#d1ffea", command=scan_devices).pack()
264
265
266
267
      #if login is successful
268
269
      def login_success():
270
          session()
271
272
      #if passwoord is not recognised
273
      def password_not_recognized():
274
          global screen4
275
          screen4 = Toplevel(screen)
          screen4.title("Failed")
276
          screen4.geometry("150x100")
277
          screen4.configure(bg="#ffcccb")
278
279
          Label(screen4, text="", bg="#ffcccb").pack()
          Label(screen4, text="Incorrect password",bg="#ffcccb",font=("Times",13)).pack()
280
281
          Button(screen4, text="Try again", bg="#ff6863", command=delete_s4).pack()
282
283
      #to allow to go to registration page
284
      def direct_register():
285
          register()
          delete_s5()
286
287
      #if user is not found
288
      def user_not_found():
289
```

7

```
290
          global screen5
291
          screen5 = Toplevel(screen)
          screen5.title("Failed")
292
          screen5.geometry("150x150")
293
          screen5.configure(bg="#ffdcd1")
294
          Label(screen5, text="User not found!",bq="#ffdcd1",font=("Times",13)).pack()
295
          Button(screen5, text="Try again", bg="#ff6863", command=delete_s5).pack()
296
297
          Label(screen5, text="", bg="#ffdcd1").pack()
298
          Button(screen5, text="Register new
          user", bg="#d1ffea", command=direct_register).pack()
299
300
301
      #to check whether login credentials are correct or not
302
      def login_verify():
303
          global username1
          global password1
304
305
          username1=username_verify.get()
306
          password1=password_verify.get()
307
          usernamee_entry1.delete(0,END)
          password_entry1.delete(0,END)
308
309
310
          conn= sqlite3.connect(resource_path("software.db"))
311
312
          c=conn.cursor()
          c.execute("SELECT username, password FROM person")
313
314
          r=c.fetchall()
315
          user_list=[]
316
317
          pass_list=[]
318
          for i in r:
              user_list.append(i[0])
319
320
              pass_list.append(i[1])
          print(user_list)
321
          print(pass_list)
322
323
          for i in range(len(user_list)):
              if(username1 in user_list):
324
325
                   if(password1 in pass_list):
                       idx=user_list.index(username1)
326
327
                       if(pass_list[idx]==password1):
328
                           # print("YES")
329
                           login_success()
330
                           break
                       else:
331
                           # print("Incorrect Password")
332
                           messagebox.askretrycancel("Try Again", "Password Incorrect")
333
```

```
334
                           break
335
                   else:
                       # print("incorrect password")
336
337
                       messagebox.askretrycancel("Try Again", "Password Incorrect")
338
              else:
339
                   # print("User Not Found")
340
341
                   user_not_found()
342
                   break
343
344
          conn.commit()
345
          conn.close()
346
347
348
      def delete_s2():
349
          screen2.destroy()
350
351
      #The login page
352
      def login():
          print("Login session started")
353
354
          global screen2
          screen2=Toplevel(screen)
355
          # screen2=Tk()
356
          screen2.title("Login page")
357
          screen2.geometry("400x350")
358
          screen2.configure(bg="#E5F6DF")
359
360
          Label(screen2, text="Please enter details below to
                                                                                                  \Box
          login", width="400", height="3", bg="#b6e9f2", font=("Times", 16)).pack()
          Label(screen2, text="", bg="#E5F6DF").pack()
361
362
          global username_verify
363
364
          global password_verify
365
          username_verify= StringVar()
366
          password_verify=StringVar()
          Label(screen2, text="Enter your registered
367
                                                                                                  \Box
          Username", font=("Times", 13), bg="#E5F6DF").pack()
          global usernamee_entry1
368
          global password_entry1
369
370
          usernamee_entry1= Entry(screen2, textvariable=username_verify, width=30)
371
          usernamee_entry1.pack()
372
          Label(screen2, text="Enter your Password", font=("Times", 13), bg="#E5F6DF").pack()
373
          password_entry1=Entry(screen2,textvariable=password_verify, show="*",width=30)
374
          password_entry1.pack()
          Label(screen2, text="", bg="#E5F6DF").pack()
375
376
          Button(screen2, text="Login", width=20, height=2,
                                                                                                  \Box
```

```
command=login_verify, bg="#d1ffea").pack()
          Label(screen2, text="", bg="#E5F6DF").pack()
377
378
          Button(screen2, text="Close window", bg="#ffcccb", height=2, width= 20,
                                                                                                 7
          command=delete_s2).pack()
379
380
381
                                                                                                 7
382
383
      def main_screen():
384
385
          global screen
386
          screen=Tk()
          screen.geometry("400x400")
387
388
          screen.minsize(400,400)
389
          screen.maxsize(400,400)
390
391
          #QUERIES to create databases(implemented for only once)
392
          # conn= sqlite3.connect("software.db")
393
          # c=conn.cursor()
          # c.execute("CREATE TABLE person(name TEXT, age INT, conn_type TEXT,
394
                                                                                                 7
          conn_type_other TEXT, gender TEXT, username TEXT, password TEXT)")
395
          # conn.commit()
396
          # conn.close()
          # conn= sqlite3.connect("user_info.db")
397
398
          # c=conn.cursor()
399
          # c.execute("CREATE TABLE Curr_session(name TEXT, bluetooth_name TEXT,
                                                                                                 7
          bluetooth_address TEXT)")
400
          # conn.commit()
401
          # conn.close()
402
403
          screen.title("Sign up/ Login page")
404
          screen.configure(bg="#caf0f8")
405
          Label(text="Sign up/ Log in", width="400", height="3", fg="#caf0f8", bg="#04035e",
          font=("Times", 16, "bold")).pack()
          photo = PhotoImage(file=resource_path("image.png"), width="400", height="150")
406
          new=Label(image=photo)
407
408
          new.pack()
          Label(text="", bg="#caf0f8").pack()
409
          Button(text="Login", height=1, width= 30, bg="#0077b6", font=("Times", 13, "bold"),
410
                                                                                                 \Box
          command=login).pack()
          Label(text="", bg="#caf0f8").pack()
411
          Button(text="Register", height=1, width=
412
                                                                                                 \Box
          30, bg="#0077b6", font=("Times", 13, "bold"), command=register).pack()
413
          Label(text="", bg="#caf0f8").pack()
```

## /home/kshitij/Downloads/Tkinter\_Loginpage-main/slr.py Page 11 of 11 Wed 09 Aug 2023 02:10:12 PM IST

```
Button(text="Quit", height=1, width= 30,bg="#0077b6",font=("Times",13,"bold"), command=lambda: screen.quit()).pack()
screen.mainloop()

416
417 main_screen()
418
```