DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

PRESIDENCY UNIVERSITY

ItgalpurRajanakunte , Yelahanka, Bengaluru, Karnataka-560064

ATM MACHINE SYSTEM

A mini project report submitted by

JAGJEET SINGH (20181CSE0281)

JEEVANJOT SINGH (20181CSE0291)

KAARNIK JAMWAL (20181CSE0303)

as part of Lab based course Programming in Python, CSE 317

of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

under the supervision of

PROF.SHWETA SINGH, Assistant Professor

PRESIDENCY UNIVERSITY

(Established under the Presidency University Act, 2013 of the Karnataka Act 41 of 2013)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that the project report entitled, “ATM MACHINE SYSTEM” is a Bonafide record of Mini Project work done as part of CSE258 Problem Solving Using Python during the academic year2019-2020 by

JAGJEET SINGH(20181CSE0281)

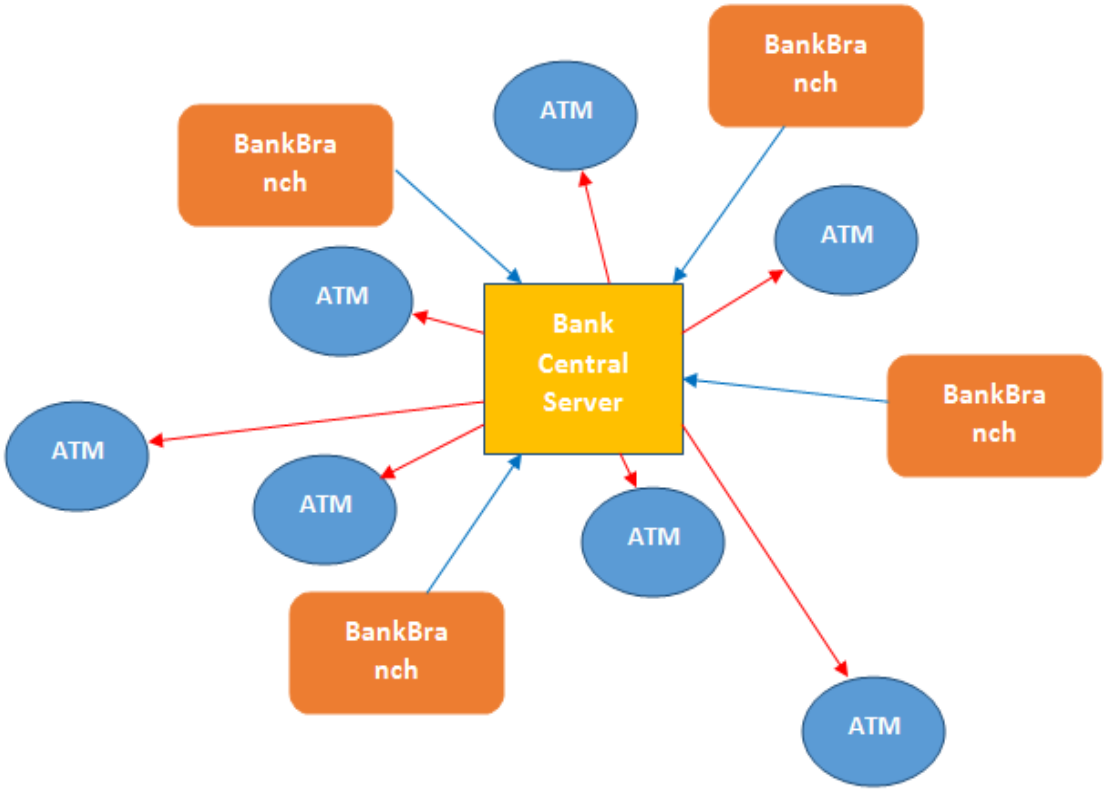
JEEVANJOT SINGH (20181CSE0291)

KAARNIK JAMWAL (20181CSE0303)

Table of Contents

INTRODUCTION

The aim of ATM MACHINE SYSTEM project is to build a Python based ATM (Automated Teller Machine) system. The introduction of ATM's by various banks have brought about freedom from the interminable queues in front of withdrawal counters at banks. This ATM Machine system requires the constant updating of records between the bank servers and a spread-out network of ATM's.



Security is the foundation of a good ATM system. This system will provide for secure authenticated connections between users and the bank servers. The whole process will be automated right from PIN (Personal Identification Number) validation to transaction completion ATM Machine System will enable two important features of an ATM, reduction of

human error in the banking system and the possibility of 24 hour personal banking. The card details and PIN database will be a secure module that will not open to routine maintenance, the only possibility of access to this database will be through queries raised from an ATM in the presence of a valid bank ATM card.

2. PROBLEM DEFINATION

THE SOURCE CODE DECLARED ABOVE FOR THE PROGRAM OF ATM MACHINE SYSTEM HAS BEEN TESTED AND IT HAS BEEN FOUND THAT THE ABOVE SOURCE CODE IS OKAY AND CORRECT. THE PROGRAM INVOLVES MANY TYPE OF CONVERSIONS. THESE CONVERSIONS HAS TO BE DONE CAREFULLY.

MAINLY THERE ARE TWO TYPES OF TESTING:

- SYSTEM TESTING
- INTEGRATION TESTING

SYSTEM TESTING INVOLVES WHOLE TESTING OF PROGRAM AT ONCE AND INTEGRATION TESTING INVOLVES THE BREAKING OF PROGRAM INTO MODULES & THEN TEST.

3.Algorithm/ Step(Explain the different modules/functions used)

1. We have run the program in JetBrains PyCharm
2. Our project is based on ATM MACHINE SYSTEM.

3. We have used Graphical User Interface(GUI) using tkinter in our project.

4. The functions of ATM Machine System are:

- Enter the correct account number and password to login.
- Account Details : Name, Account Number and Type will be shown.
- Deposit Money:this functions helps us to deposite the money.
- Withdrawal Money:Through this function we can take our money from the machine.
- Balance Enquiry : we get the mini statement of our account.
- Quit: this function helps us to terminate the process.

4. IMPLEMENTATION (SOURCE CODE)

```
#shweta-> acc no = 10 password = trial
#jagjeet-> acc no = 14 password = singh
#kaarnik-> acc no = 16 password = jamwal
#jeevanjot-> acc no = 17 password =hello

from tkinter import *
from tkinter import messagebox
import sqlite3


ARIAL = ("arial",10,"bold")


class Bank:
def __init__(self,root):
self.conn = sqlite3.connect("t.db", timeout=100)
self.login = False
self.root = root
self.header = Label(self.root,text="PATIALA
BANK",bg="#50A8B0",fg="white",font=("arial",20,"bold"))
self.header.pack(fill=X)
```

```
self.frame =
Frame(self.root,bg="#728B8E",width=600,height=400)

#Login Page Form Components

self.userlabel =Label(self.frame,text="Account
Number",bg="#728B8E",fg="white",font=ARIAL)

self.uentry =
Entry(self.frame,bg="honeydew",highlightcolor="#50A8B0",

highlightthickness=2,

highlightbackground="white")

self.plabel = Label(self.frame,
text="Password",bg="#728B8E",fg="white",font=ARIAL)

self.pentry =
Entry(self.frame,bg="honeydew",show="*",highlightcolor="#
50A8B0",

highlightthickness=2,

highlightbackground="white")

self.button =
Button(self.frame,text="LOGIN",bg="#50A8B0",fg="white",fo
nt=ARIAL,command=self.verify)

self.q =
Button(self.frame,text="Quit",bg="#50A8B0",fg="white",font
=ARIAL,command = self.root.destroy)

self.userlabel.place(x=145,y=100,width=120,height=20)

self.uentry.place(x=153,y=130,width=200,height=20)
```

```
self.plabel.place(x=125,y=160,width=120,height=20)
self.pentry.place(x=153,y=190,width=200,height=20)
self.button.place(x=155,y=230,width=120,height=20)
self.q.place(x=480,y=360,width=120,height=20)



self.frame.pack()
def database_fetch(self):#Fetching Account data from database
self.acc_list = []
self.temp = self.conn.execute("select name,pass,acc_no,acc_type,bal from atm where acc_no = ? ",(self.ac,))
for i in self.temp:
self.acc_list.append("Name = {}".format(i[0]))
self.acc_list.append("Account no = {}".format(i[2]))
self.acc_list.append("Account type = {}".format(i[3]))
self.ac = i[2]
self.acc_list.append("Balance = {}".format(i[4]))


def verify(self):#verifying of authorised user
ac = False
```

```
self.temp = self.conn.execute("select
name,pass,acc_no,acc_type,bal from atm where acc_no = ?
", (int(self.uentry.get()),))
for i in self.temp:
self.ac = i[2]
if i[2] == self.uentry.get():
ac = True
elif i[1] == self.pentry.get():
ac = True
m = "{} Login SucessFull".format(i[0])
self.database_fetch()
messagebox._show("Login Info", m)
self.frame.destroy()
self.MainMenu()
else:
ac = True
m = " Login UnSucessFull ! Wrong Password"
messagebox._show("Login Info!", m)

if not ac:
m = " Wrong Acoount Number !"
messagebox._show("Login Info!", m)
```

```
def MainMenu(self):#Main App Appears after logined !
self.frame =
Frame(self.root,bg="#728B8E",width=800,height=400)
root.geometry("800x400")
self.detail = Button(self.frame,text="Account
Details",bg="#50A8B0",fg="white",font=ARIAL,command=self
.account_detail)
self.enquiry = Button(self.frame, text="Balance
Enquiry",bg="#50A8B0",fg="white",font=ARIAL,command=
self.Balance)
self.deposit = Button(self.frame, text="Deposit
Money",bg="#50A8B0",fg="white",font=ARIAL,command=sel
f.deposit_money)
self.withdrawl = Button(self.frame, text="Withdrawl
Money",bg="#50A8B0",fg="white",font=ARIAL,command=sel
f.withdrawl_money)
self.q = Button(self.frame, text="Quit", bg="#50A8B0",
fg="white", font=ARIAL, command=self.root.destroy)
self.detail.place(x=0,y=0,width=200,height=50)
self.enquiry.place(x=0, y=315, width=200, height=50)
self.deposit.place(x=600, y=0, width=200, height=50)
self.withdrawl.place(x=600, y=315, width=200, height=50)
```

```
self.q.place(x=340, y=340, width=120, height=20)
self.frame.pack()

def account_detail(self):
self.database_fetch()
text =
self.acc_list[0]+"\n"+self.acc_list[1]+"\n"+self.acc_list[2]
self.label = Label(self.frame,text=text,font=ARIAL)
self.label.place(x=200,y=100,width=300,height=100)

def Balance(self):
self.database_fetch()
self.label = Label(self.frame, text=self.acc_list[3],font=ARIAL)
self.label.place(x=200, y=100, width=300, height=100)

def deposit_money(self):
self.money_box =
Entry(self.frame,bg="honeydew",highlightcolor="#50A8B0",
highlightthickness=2,
highlightbackground="white")
self.submitButton =
Button(self.frame,text="Submit",bg="#50A8B0",fg="white",f
ont=ARIAL)
```

```
self.money_box.place(x=200,y=100,width=200,height=20)
self.submitButton.place(x=445,y=100,width=55,height=20)
self.submitButton.bind("<Button-1>",self.deposit_trans)

def deposit_trans(self,flag):
self.label = Label(self.frame, text="Transaction Completed !", font=ARIAL)
self.label.place(x=200, y=100, width=300, height=100)
self.conn.execute("update atm set bal = bal + ? where acc_no = ?",(self.money_box.get(),self.ac))
self.conn.commit()

def withdrawl_money(self):
self.money_box = Entry(self.frame,bg="honeydew",highlightcolor="#50A8B0",
highlightthickness=2,
highlightbackground="white")
self.submitButton = Button(self.frame,text="Submit",bg="#50A8B0",fg="white",font=ARIAL)

self.money_box.place(x=200,y=100,width=200,height=20)
```

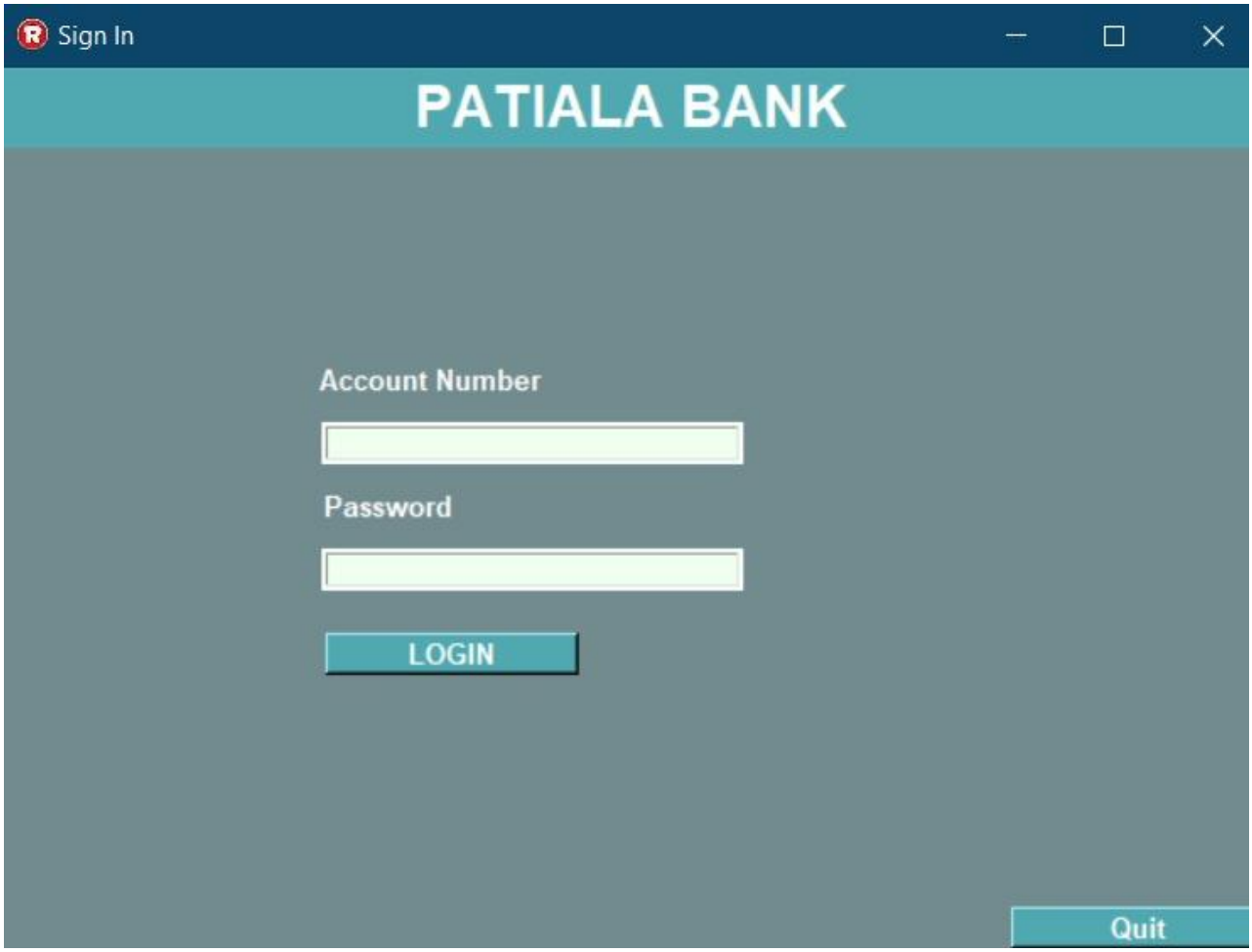
```
self.submitButton.place(x=445,y=100,width=55,height=20)
self.submitButton.bind("<Button-1>",self.withdrawl_trans)

def withdrawl_trans(self,flag):
self.label = Label(self.frame, text="Money Withdrawl !", font=ARIAL)
self.label.place(x=200, y=100, width=300, height=100)
self.conn.execute("update atm set bal = bal - ? where acc_no = ?",(self.money_box.get(),self.ac))
self.conn.commit()

root = Tk()
root.title("Sign In")
root.geometry("600x420")
icon = PhotoImage(file="icon.png")
root.tk.call("wm",'iconphoto',root._w,icon)
obj = Bank(root)
root.mainloop()
```

** Code written with sincere efforts by Jagjeet,Kaarnik and Jeevanjot

5. APPLICATION PREVIEW


Sign In
PATIALA BANK
Account Number
Password
LOGIN
Quit

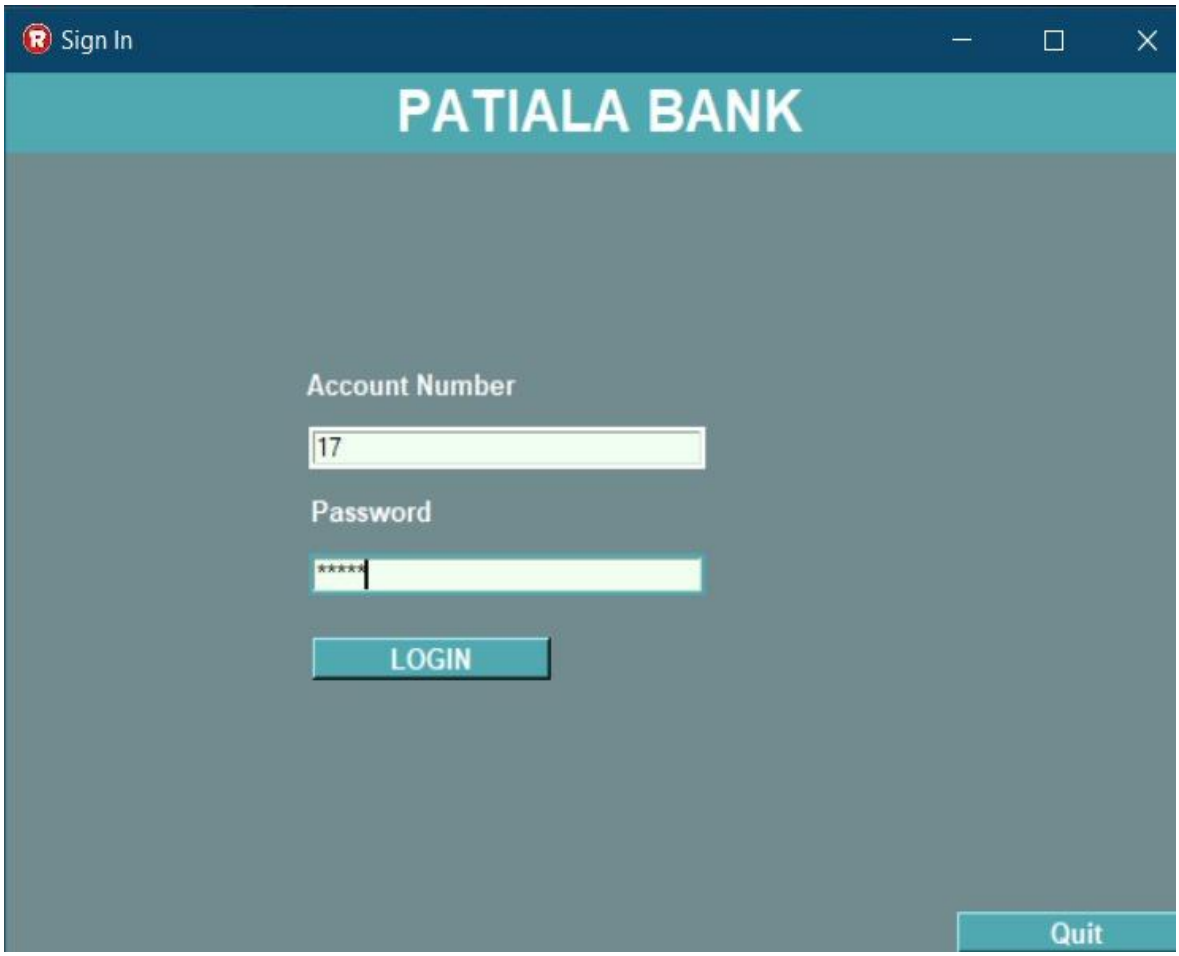
6. MYSQL DATABASE REGISTERED WITH CO

SQLite

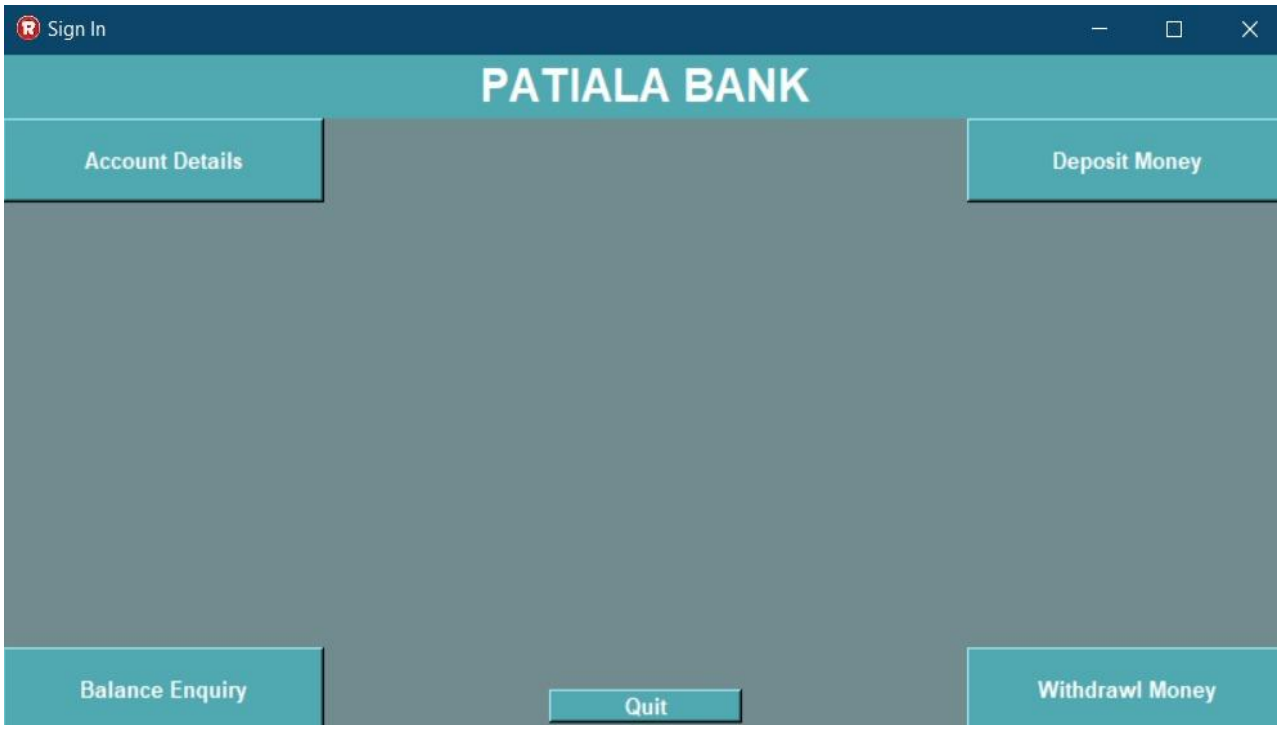
```
1 SELECT * FROM atm;
```

name	acc_no	acc_type	bal	pass
shweta	10	savings	37460	trial
jagjeet	14	savings	35000	singh
kaarnik	16	savings	90000	jamwal
jeevanjot	17	savings	10000	hello

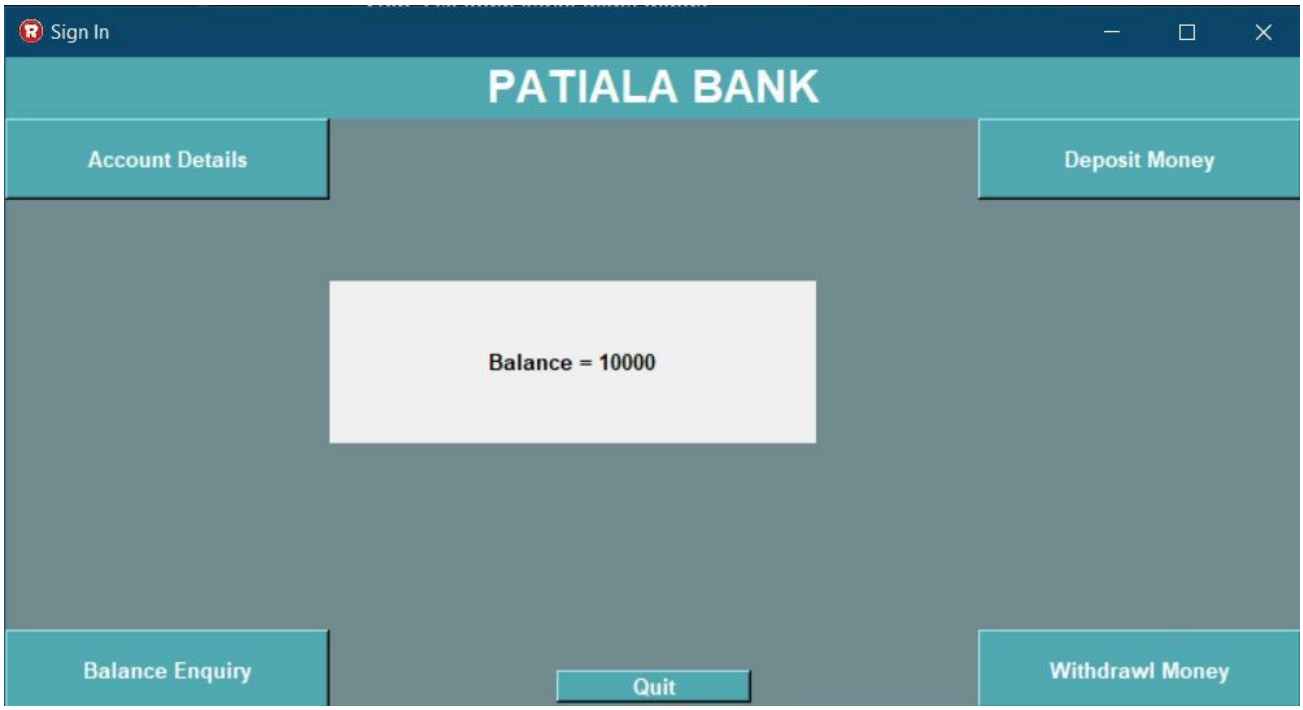
7. LOGGING IN BY ENTERING ACCOUNT NO AND PASSWORD



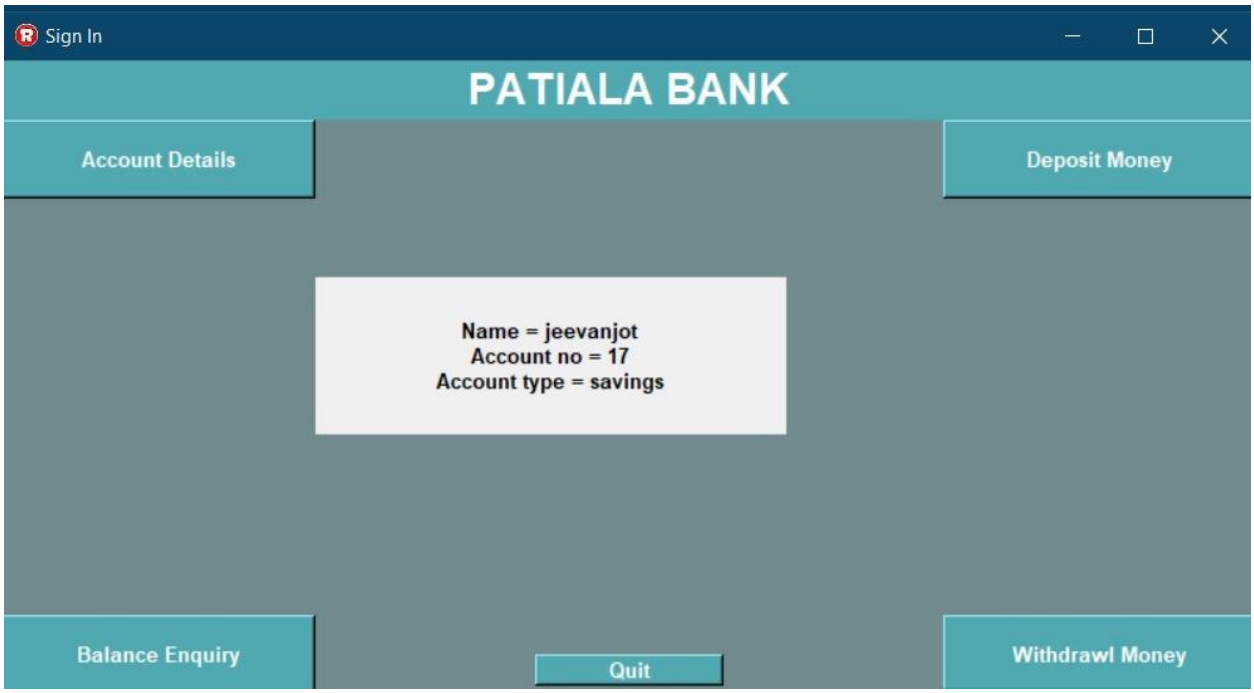
8.LOGGED IN SUCCESSFULLY



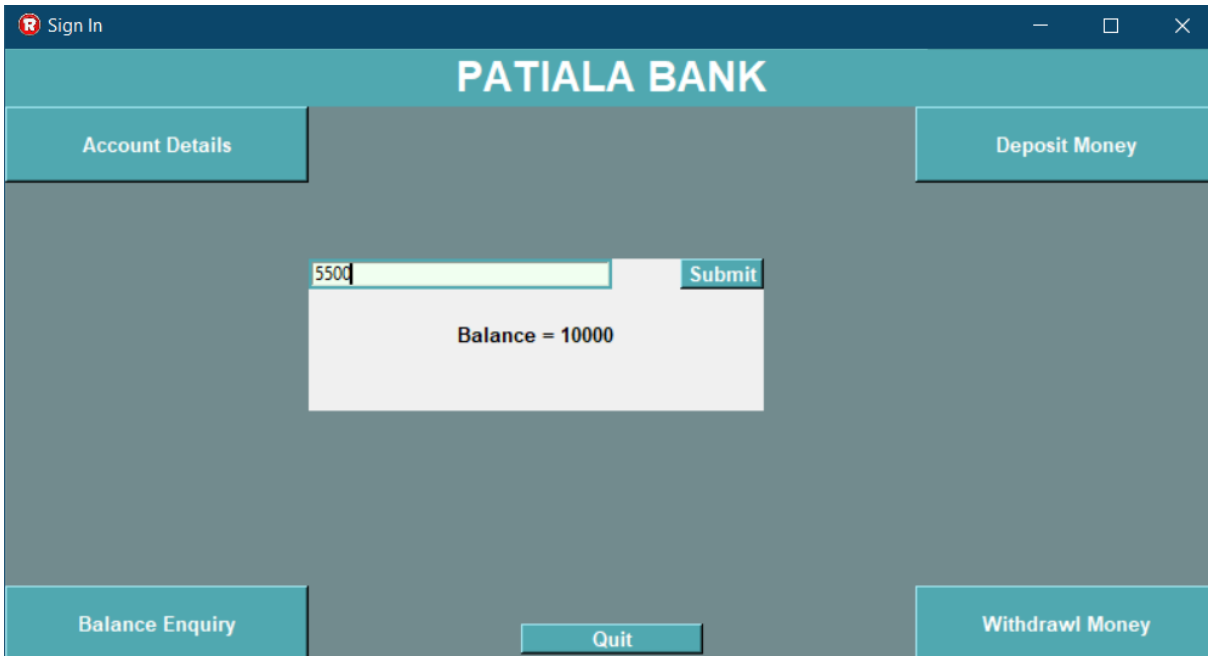
9.BALANCE ENQUIRY



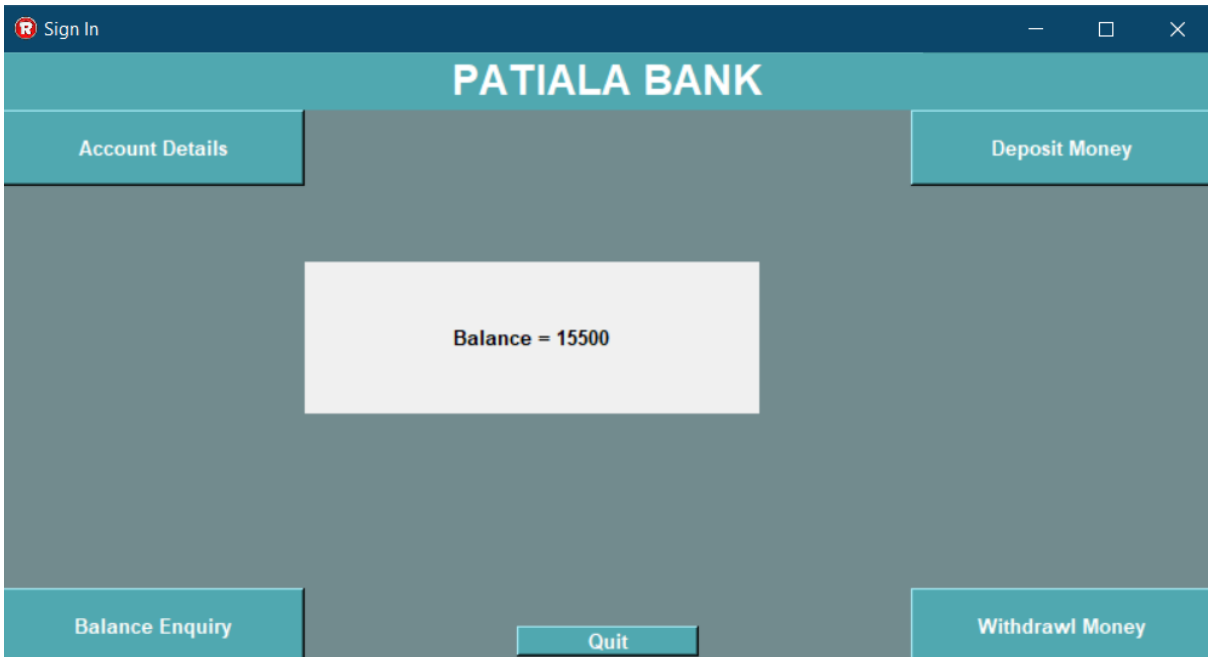
10.ACCOUNT DETAILS



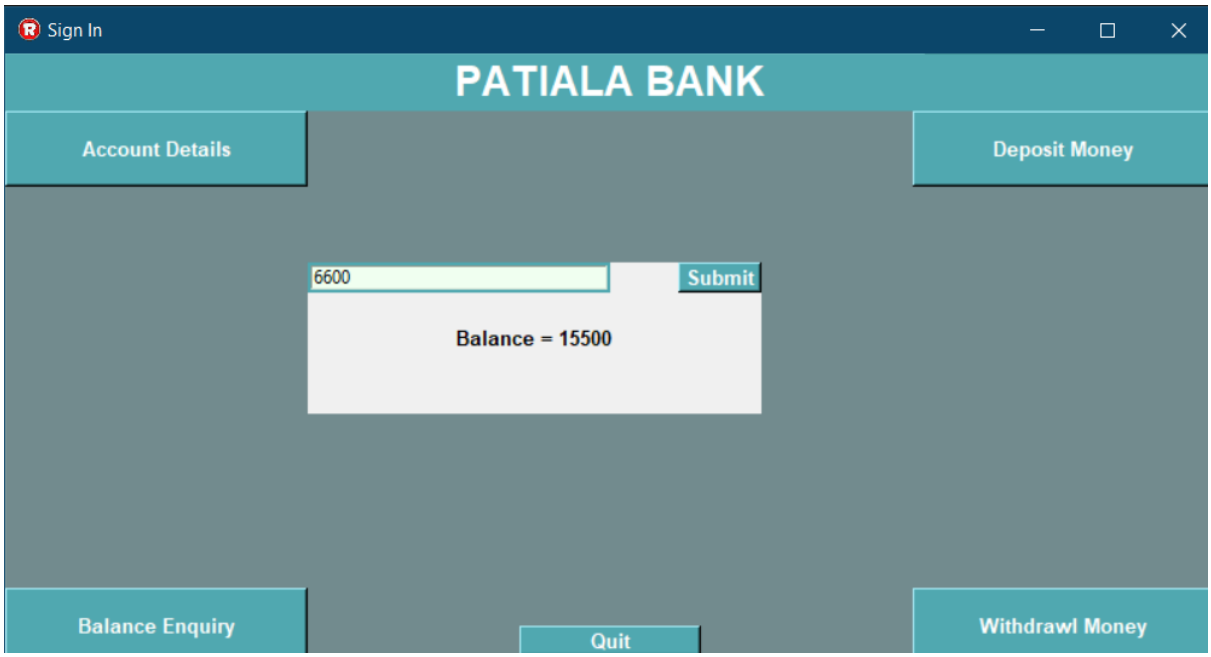
11. Depositing Money



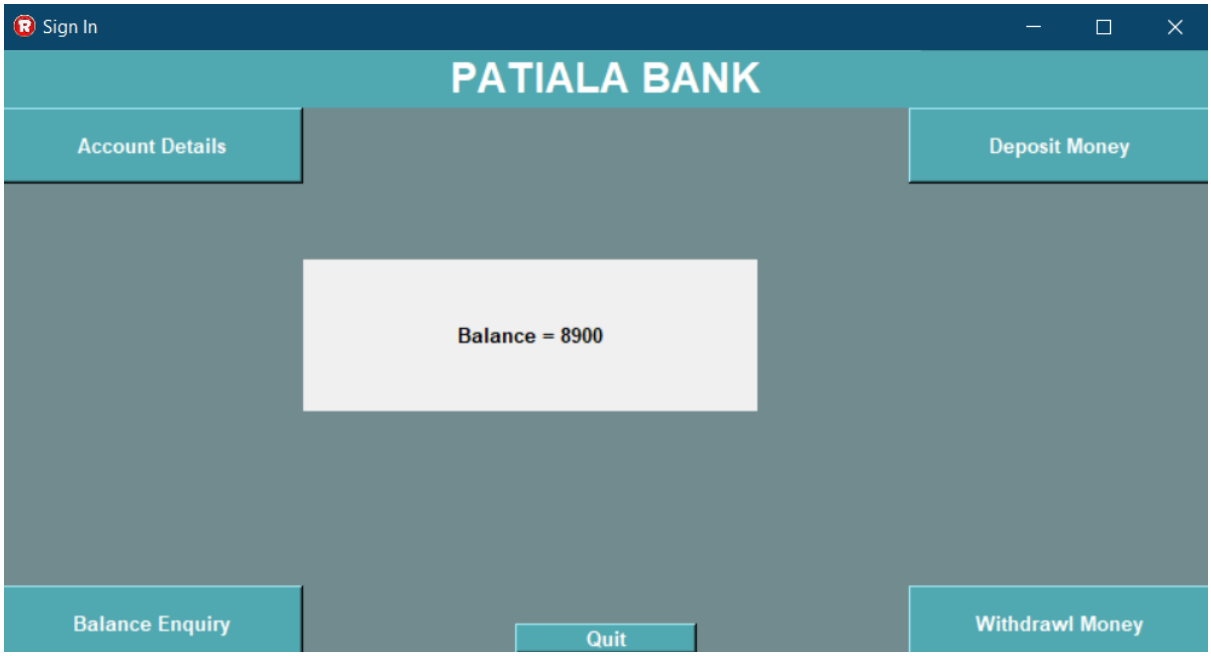
12.Balance after Depositing money



13. Withdrawing Money



14.Balance after Withdrawing Money



CONCLUSION

We have successfully completed our mini project “ATM MACHINE SYSTEM” IN PYTHON. We have used Graphical User Interface(GUI) using tkinter in our project.

Future scope

The future of ATM will surely see a big curve in the coming years,but in which direction is still a big question.Before diving towards the conclusion let us see some important points published in a recent report by “cash product office” .

1) Cash is still the most used form of payment in retail, accounting for 40% of all transactions.

2) Cash is the most used payment for purchases under $50.

3) Low-income consumers are the biggest users of cash.

4) 40% of people 18-24 mainly use cash as their preferred form of payment.

5) Households earning under $25,000/year prefer to use cash to pay bills over any other form of payment.

6) Cash is the preferred back-up form of payment.

Referring to above points one can surely say that cash will continue to be in use in huge amounts which in turn will increase the demand of cards as one cannot afford to visit bank branch every time for cash. So basically ATM’s are not a thing of short run. So the question now arises will the use pattern of these machines change or not. Well, the use pattern can surely change. We could surely witness more secure, safer usage in the ATM’s. They can be even card less in near future as in his report, Bill verson said ATMs will adopt biometric identifiers like fingerprint and facial recognition to enable

cardless usage. They'll also include "services as diverse as applying for loans, buying lottery tickets and dispensing foreign currency,"

So in short we would like to say Cash currency is needed the world over. And, especially for locations where a physical bank branch isn’t available, having an ATM to use for your banking needs is essential. . As a result, ATMs are catching on like wildfire.

Also some developed countries like USA have already started using the crypto currency ATMs and as we are expecting a increase in crypto currency in near future there is no way that ATMs number would go down.

It would not be wrong if we believe that ATMs will become the banks of the future as more branches close.

So at last, as far as the future scope of the ATMs is concerned, we would say that ATMs are here to stay and would not go anywhere in the near future as the need of these machines is very crucial in working of the banks as banks cannot physically transfer the cash to its every customer.

Also, it would not be wrong to say that as long as cash is in use ATMs will be in role.

REFERENCES

***https://stackoverflow.com/questions/4969543/colour-chart-for-tkinter-and-tix/6932500*.**

https://sqliteonline.com/

https://www.tutorialspoint.com/python/tk_colors.htm

https://www.google.com/search?q=python+file+to+exe&oq=pyhon&aqs=chrome.1.69i57j35i39i305l2j0i10i433i457j0i10i433.6891j0j4&client=ms-android-xiaomi-rev1&sourceid=chrome-mobile&ie=UTF-8#

https://www.w3schools.com/python/python_mysql_getstarted.asp

https://www.google.com/amp/s/www.geeksforgeeks.org/convert-python-script-to-exe-file/amp/

https://realpython.com/python-gui-tkinter/


Thank You!