REPORT:
ATM MANAGEMENT SYSTEM

Computer Science and engineering.

Under The Guidance of

Dr. Deepika Ghai

Submitted By

Mugle Sruthi – 12109334 - (03)

Surya Mounika –12109394 -(05)

Manreet – 12109354 - (06)

In partial fulfilment for the award of the degree of

Department of computer science and engineering

Lovely Professional University, Punjab. (2022-2023)

ACKNOWLEDGEMENT

We express our deep sense of gratitude to our respected and learned guides Dr. Deepika ghai their valuable help and guidance, we are thankful to them for the encouragement they have given us in completing the project. we would like to thank the other faculty members also, at this occasion.

Lastly, we would like to express apperception towards our group members, our classmates and indebt to our parents for providing us the moral support and encouragement during our work.

Mugle Sruthi-12109334,

Surya Mounika-12109394,

Manreet-12109354.

INTRODUCTION:

This is the Mini Project of Python related ATM Management System. This is the group Project of Three Members for the purpose of CA 2 and to know and get more knowledge and overview on this mini project.

This is the final report which includes:

1)Module wise description.

2)Role and Responsibility of each group member.

3)Main code, source code and GUI interfaces

BASIC REQUIREMENTS:

1). Input:

The card reader – which captures the account information stored on the magnetic stripe on the back of an ATM/Debit or Credit Card. The host processor uses this information to route the transaction to the cardholder's bank.

2). Process Definition:

To Access cash from an ATM, you just need to follow a few simple steps:

Insert your debit or credit card into the ATM. Confirm your identity by entering your personal identification number (PIN)Choose the account you want to withdraw funds from.

3). Output:

The output devices in an ATM are **Speaker, LCD/CRT Screen, Receipt Printer and Cash Dispenser**. The Operating System used in the ATMs are mostly Microsoft Windows XP, 7, 8 or Linux variants. Time Out: It is a security feature enabled in ATMs to enter the PIN (Personal Identification Number) number within a stipulated time.

MODULES:

Transfer module: – This module lets users to transfer cash from one account to another. The inputs are the password of the sender and the receiver and the amount of money to be transferred.

Card transactions module: – This module has two sub-modules: Change password and dept paying.

SUPERVISOR MODULE:

- Bank Customer. This actor represents a person with a valid Bank Card.
- Cashier. From the ATM system point of view, the Cashier's only responsibility is to count the money in the security box to verify all deposits.
- Bank.
- Maintenance Person.

SYSTEM ATTRIBUTES:

It has three attributes:

Transaction ID, date, and type (This method is to update).

Class	Descriptive words and phrases
ATM	User is authenticated
Balance Inquiry	Account number
Withdrawal	Account number amount
Deposit	Account number amount

Robustness:

It is a fundamental property of the natural ecosystem that enables quick recovery after numerous disturbances occurring frequently. this vital ability of the ecosystem makes resilience a very desirable property of manufactured socio-technical systems, one of which is an ATM system.

It is necessary to develop conceptual definitions of resilience and robustness of an ATM System, which have a clear differentiation between these terms and enable their measurement.

Reusability:

We can update it next version. **Reusable software reduces design, coding, and testing cost by amortizing effort over several designs.**

Testability:

- Verify the 'ATM Card Insertion Slot' is as per the specification.
- Verify the ATM machine accepts card and PIN details.
- Verify the error message by inserting a card incorrectly.
- Verify the error message by inserting an invalid card (Expired Card)
- Verify the error message by entering an incorrect PIN.

SOFTWARE REQUIREMENTS:

Basic requirements:

The ATM system shall accept a unique ID from the customer. The ATM system shall accept the customer selection for a range of banking transactions. The ATM system shall accept the account type from the customer for deposits, balance query, and for payments on account.

Software requirements:

- **OS:** windows /Unix/Linux.
- **Database:** Transaction Manager's Database, My SQL (Free-Open Source).
- **Language:** Python.

Hardware Requirements:

Basic requirements:

Card readers, cash dispensers, PIN pads, receipt printers and monitors are some of the basic hardware components that go into all ATMs. Many ATMs also have components for wireless connectivity, check scanning, or even dispensing gift cards.

Hardware requirements:

- **Processor**: Transaction processor, **Zilog Z80 @ 3.5, and 7 MHz; 14 MHz (ZX Evolution).**
- **Hard disk:** Floppy disk, IDE devices (ATM Turbo 2).
- **Memory:** 128 to 512 KB (ATM Turbo 1); 128 to 1024 KB (ATM Turbo 2+); 4MB RAM (ZX Evolution).
- **Mouse:** Any Standard.

- **Keyboard:** Any standard.
- **Monitor:** Any colour monitor.
- **Local Area Network:** preferable.

LANGUAGES USED:

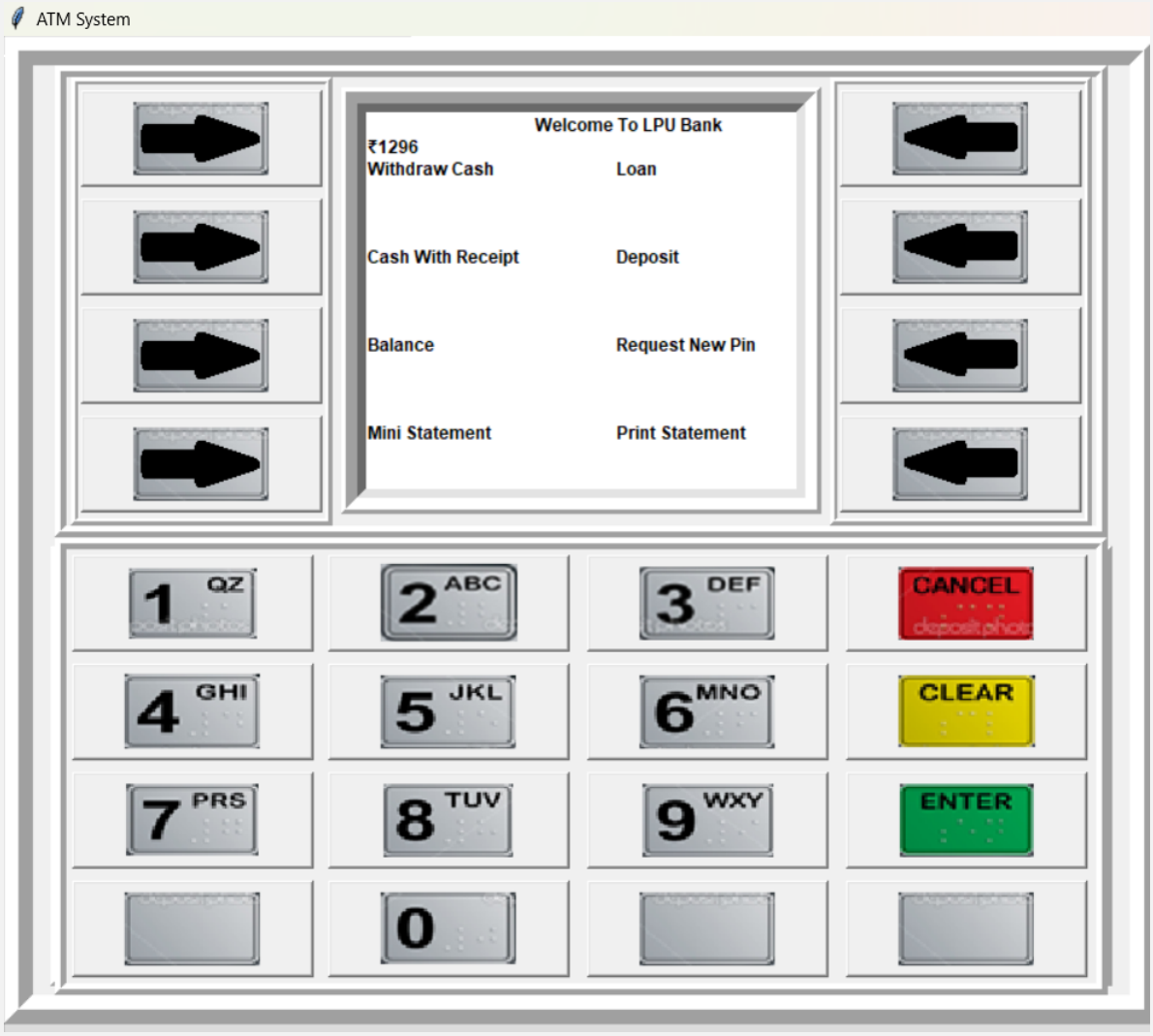
- **Front end:** python
- **Backend: c**/c++ and the code used in ATMC is pro*c

Purpose of the project: -

The system **calculates the total remaining balance of the respective account and displays to the user**. This simple console based ATM system provides the simple account balance management of a respective account. It contains fewer features but the essential ones are withdrawal, deposit, checking balances, mini statements. Thus this a digital way and the most coinvent way for the withdrawal of money and checking balance etc.....

ATM MANAGEMENT SYSTEM.

ATM Simulator project is written in Python. The project file contains a python script (atm.py). This is a simple console based system which is very easy to use. Talking about the system, it contains various functions which include Account Statement, Withdrawing, Depositing amount and changing the pin.



DESCRIPTION OF THE PROJECT: -

ATM Management System is an electronic telecommunication device which enables the customers of financial institutions like banks, to carry out financial transactions, such as cash withdrawal, Balance Enquiry without any requirement of a human cashier, cashier or bank clerk.

Here are the details about the project and you will find the overall description about the project such as source code that how it's functioning .

- **Main. Py** – Python main function is the beginning of any python program. When we run a program the interpreter runs the code sequentially and will not run the main function if imported as a module but the main function gets executed only when it is run as a Python program.

Brief Explanation about the project:

The purpose of this project is Making an ATM Management System. where a user can check the balance, Deposit Money, Withdraw Money and Change the ATM Pin.

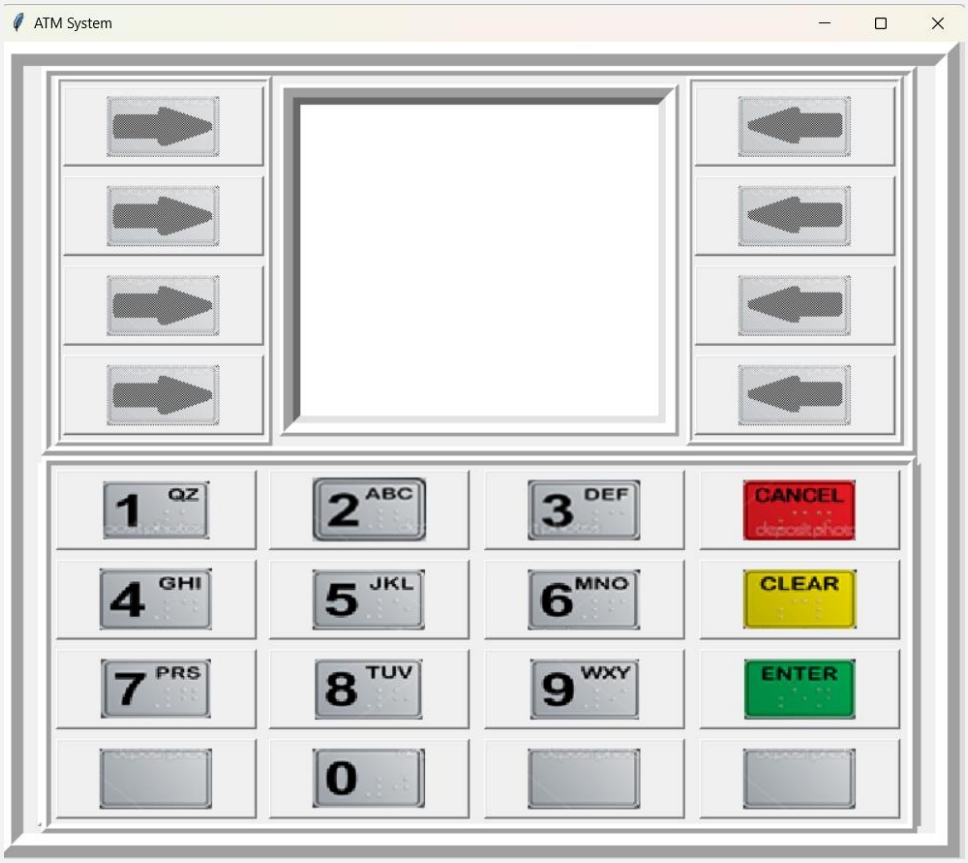
Frameworks of ATM Management System.

A generic framework for designing management applications for small- to medium-size heterogeneous ATM networks This section presents a methodological approach to the design of systems for the management of heterogeneous ATM networks, centred around the concept of abstract information models.

Various Frameworks: -

1).Main Frame: -

- Mainframe is used to access their bank Accounts to make cash withdrawals. Whenever the user needs to make Cash withdraws, they can enter their PIN number (Personal identification Number) and it will display the amount to be withdrawn in the form of 100’s 500’s and 1000’s. Then we can proceed with payment. Or else if the pin is incorrect then it will show to Re-Enter the pin.


ATM System
1 QZ
2 ABC
3 DEF
CANCEL
4 GHI
5 JKL
6 MNO
CLEAR
7 PRS
8 TUV
9 WXY
ENTER
0

❖Overall description about project: -

1). Main.py

To use the tkinter we need to import the tkinkter module. As stated above, we have imported each file so that we can make function calls from our main file.

```
from tkinter import *
from tkinter import ttk
import tkinter.messagebox
class atm:
   def __init__(self, root):
       self.root = root
       blank_space = "
 # by this we get our title in the center
self.root.title(110 * blank_space + "ATM System")
self.root.geometry("800x760+280+0")  # this the border size
 self.root.configure(background='gainsboro')
 #0).================Frame==================#
MainFrame = Frame(self.root, bd=20, width=760, height=700, relief=RIDGE)
 MainFrame.grid()
# in the output we can able to see the margins or border
TopFrame = Frame(MainFrame, bd=8, width=730, height=300, relief=RIDGE)
 # this is the topframe under mainframe
```

```
TopFrame.grid(row=2, column=0, padx=12)

TopFrame1 = Frame(MainFrame, bd=8, width=730, height=300,
relief=RIDGE)

TopFrame1.grid(row=2, column=0, padx=12)

TopFrame2 = Frame(MainFrame, bd=8, width=730, height=300,
relief=RIDGE)

TopFrame2.grid(row=1, column=0, padx=8)  # lets chnage the frame
spacing

 TopFrame2Left = Frame(TopFrame2, bd=5, width=190,

   height=300, relief=RIDGE)

  # this is the topframe2 left in topframe2

  TopFrame2Left.grid(row=0, column=0, padx=3)

  TopFrame2Mid = Frame(TopFrame2, bd=5, width=200 ,
height=300, relief=RIDGE)

     # this is the middle frame in topframe2

  TopFrame2Mid.grid(row=0, column=1, padx=3)

  TopFrame2Right = Frame(TopFrame2, bd=5, width=190,

                         height=300, relief=RIDGE)

    # this is the topframe2Right in topframe2

   TopFrame2Right.grid(row=0, column=2, padx=3)
```

```
 #==================functions==================#
 def enter_Pin():
          pinNo = self.txtReceipt.get("1.0","end-1c")
          if((pinNo == str("2213"))or (pinNo == str("4323"))or (pinNo
== str("5982"))):
        self.txtReceipt.delete("1.0",END)
      self.txtReceipt.insert(END,'\t\t         ATM' + "\n\n")
       self.txtReceipt.insert(END,'Withdraw Cash\t\t\t Loan' +
"\n\n\n\n")
      self.txtReceipt.insert(END,'Cash With Receipt \t\t\t Deposit' +
"\n\n\n\n")
      self.txtReceipt.insert(END,'Balance \t\t\t Request New Pin' +
"\n\n\n\n")
     self.txtReceipt.insert(END,'Mini Statement \t\t\t Print Statement' +
"\n\n\n\n")


#7).making button normal enable ==rightbutton==========#


self.btnrArrow1=Button(TopFrame2Right, width=160, height=60,
state =NORMAL,command=loan,
image=self.img_arrow_Right).grid(row=0,column =0,padx=2,pady
=4)
 self.btnrArrow2=Button(TopFrame2Right, width=160, height=60,
state =NORMAL,command=deposit,
image=self.img_arrow_Right).grid(row=1,column =0,padx=2,pady
=4)
```

```
self.btnrArrow3=Button(TopFrame2Right, width=160, height=60,
state =NORMAL,command=request_new_pin,

image=self.img_arrow_Right).grid(row=2, column =0,padx=2, pady
=4)

 self.btnrArrow4=Button(TopFrame2Right, width=160, height=60,
state =NORMAL,command=statement,

 image=self.img_arrow_Right).grid(row=3, column =0,padx=2, pady
=4)
```

#8).WE ARE ENABLING BUTTONS BY CLICKING ENTER BUTTON LEFT SIDE BUTTON ==========#
self.btnlArrow1=Button(TopFrame2Left, width=160, height=60, state=NORMAL,command=withdrawcash,

image=self.img_arrow_Left).grid(row=0,column =0,padx=2,pady =4)

self.btnlArrow2=Button(TopFrame2Left, width=160, height=60, stat=NORMAL,command=withdrawcash,

image=self.img_arrow_Left).grid(row=1, column =0, padx=2, pady =4)

self.btnlArrow3=Button(TopFrame2Left, width=160, height=60, state=NORMAL,command=balance,

image=self.img_arrow_Left).grid(row=2, column =0, padx=2, pady =4)

self.btnlArrow4=Button(TopFrame2Left, width=160, height=60, state=NORMAL,command=statement,

image=self.img_arrow_Left).grid(row=3, column =0, padx=2, pady =4)

```
 else:
 self.txtReceipt.delete("1.0",END)
self.txtReceipt.insert(END,'Invalid Pin Number' + "\n\n"
def clear():
       self.txtReceipt.delete("1.0",END)
        self.btnlArrow1=Button(TopFrame2Left, width=160, height=60, state=DISABLED,
         image=self.img_arrow_Left).grid(row=0, column =0, padx=2, pady =4)
        self.btnlArrow2=Button(TopFrame2Left, width=160, height=60, state=DISABLED,
        image=self.img_arrow_Left).grid(row=1, column =0, padx=2, pady =4)
      self.btnlArrow3=Button(TopFrame2Left, width=160, height=60, state=DISABLED,
      image=self.img_arrow_Left).grid(row=2, column =0, padx=2, pady =4)
      self.btnlArrow4=Button(TopFrame2Left, width=160, height=60, state=DISABLED,
      image=self.img_arrow_Left).grid(row=3, column =0, padx=2, pady =4)
     self.img_arrow_Right = PhotoImage(file ="rArrow.png")
 self.btnrArrow1=Button(TopFrame2Right, width=160, height=60, state =DISABLED,
      image=self.img_arrow_Right).grid(row=0, column =0,padx=2, pady =4)
```

```
      self.btnrArrow2=Button(TopFrame2Right, width=160, height=60,
state =DISABLED,

       image=self.img_arrow_Right).grid(row=1, column =0,padx=2,
pady =4)

     self.btnrArrow3=Button(TopFrame2Right, width=160, height=60,
state =DISABLED,

     image=self.img_arrow_Right).grid(row=2, column =0,padx=2,
pady =4)

    self.btnrArrow4=Button(TopFrame2Right, width=160, height=60,
state =DISABLED,

     image=self.img_arrow_Right).grid(row=3, column =0,padx=2,
pady =4)
```

```
def insert0():
        value0 = 0
        self.txtReceipt.insert(END,value0)
    def insert1():
        value1 = 1
        self.txtReceipt.insert(END,value1)

    def insert2():
        value2 = 2
        self.txtReceipt.insert(END,value2)
    def insert3():
        value3 = 3
        self.txtReceipt.insert(END,value3)
    def insert4():
        value4 = 4
        self.txtReceipt.insert(END,value4)
    def insert5():
        value5 = 5
        self.txtReceipt.insert(END,value5)

    def insert6():
        value6 = 6
        self.txtReceipt.insert(END,value6)
```

```
    def insert7():
        value7 = 7
        self.txtReceipt.insert(END,value7)
    def insert8():
        value8 = 8
        self.txtReceipt.insert(END,value8)
    def insert9():
        value9 = 9
        self.txtReceipt.insert(END,value9)
    def cancel():
        Cancel = tkinter.messagebox.askyesno("ATM","Confirm if
you want to cancel")
        Cancel > 0
        self.root.destroy()
        return
    def withdrawcash():
        enter_Pin()
        self.txtReceipt.delete("1.0",END)
        self.txtReceipt.focus_set()

    def loan():
        enter_Pin()
        self.txtReceipt.delete("1.0",END)
        self.txtReceipt.insert(END, 'Loan ₹')
        self.txtReceipt.focus_set()
```

```
    def deposit():
        enter_Pin()
        self.txtReceipt.delete("1.0",END)
        self.txtReceipt.focus_set()


      def request_new_pin():
        enter_Pin()
        self.txtReceipt.delete("1.0",END)
        self.txtReceipt.insert(END, '\t\t Welcome To LPU Bank \n')
        self.txtReceipt.insert(END, 'New Pin will be send your home
address \n')
        self.txtReceipt.insert(END,'Withdraw Cash\t\t\t Loan' +
"\n\n\n\n")


        self.txtReceipt.insert(END,'Cash With Receipt \t\t\t Deposit' +
"\n\n\n\n")


        self.txtReceipt.insert(END,'Balance \t\t\t Request New Pin' +
"\n\n\n\n")


        self.txtReceipt.insert(END,'Mini Statement \t\t\t Print
Statement' + "\n\n\n\n")
        self.txtReceipt.insert(END, '\t\t Thanks for using  LPU Bank
\n')
```

```
   def balance():
        self.txtReceipt.delete("1.0",END)
        self.txtReceipt.insert(END, '\t\t Welcome To LPU Bank \n')
        self.txtReceipt.insert(END, '₹1296' +"\n")
        self.txtReceipt.insert(END,'Withdraw Cash\t\t\t Loan' +
"\n\n\n\n")


        self.txtReceipt.insert(END,'Cash With Receipt \t\t\t Deposit' +
"\n\n\n\n")


        self.txtReceipt.insert(END,'Balance \t\t\t Request New Pin' +
"\n\n\n\n")


        self.txtReceipt.insert(END,'Mini Statement \t\t\t Print
Statement' + "\n\n\n\n")
        self.txtReceipt.insert(END, '\t\t Thanks for using  LPU Bank
\n')


      def statement():
        pinNo1 = str(self.txtReceipt.get("1.0", "end-1c"))
        pinNo2 = str(pinNo1)
        pinNo3 = float(pinNo2)
        pinNo4 = float(1296-(pinNo3))
        self.txtReceipt.delete("1.0",END)
        self.txtReceipt.insert(END, '\n\t' + str(pinNo4)+ "\t\t")
```

```
        self.txtReceipt.insert(END,'\t\t\t\n\n Account Balance ₹' +
str(pinNo4)+ "\n\n\n\n")


        self.txtReceipt.insert(END,'Rent\t\t\t\t ₹1200' + "\n\n")


        self.txtReceipt.insert(END,'Tesco\t\t\t\t ₹79.36' + "\n\n")


        self.txtReceipt.insert(END,'Sainsbury'+'s \t\t\t ₹53.87' + "\n\n")
        self.txtReceipt.insert(END,'Student Loan \t\t\t\t ₹69.72' +
"\n\n")
        self.txtReceipt.insert(END,'RupeesLand \t\t\t\t ₹19' + "\n\n")
```

```
#1).======================widget==================#
    self.txtReceipt = Text(TopFrame2Mid, height = 17, width=42,
bd=12, font=('arial',9,'bold'))
     self.txtReceipt.grid(row=0, column=0)
     self.img_arrow_Left = PhotoImage(file ="lArrow.png")
     self.btnlArrow1=Button(TopFrame2Left, width=160, height=60,
state=DISABLED,command=withdrawcash,
     image=self.img_arrow_Left).grid(row=0, column =0, padx=2,
pady =4)
    self.btnlArrow2=Button(TopFrame2Left, width=160, height=60,
state=DISABLED,command=withdrawcash,
    image=self.img_arrow_Left).grid(row=1, column =0, padx=2, pady
=4)
 self.btnlArrow3=Button(TopFrame2Left, width=160, height=60,
state=DISABLED,
  command=balance, image=self.img_arrow_Left).grid(row=2,
column =0, padx=2, pady =4)
 self.btnlArrow4=Button(TopFrame2Left, width=160, height=60,
state=DISABLED,command=statement,
  image=self.img_arrow_Left).grid(row=3, column =0, padx=2, pady
=4)
# here i have to add four times button so due to error it isn't working
and these are
 # added in the left side
```

```
#2).=============right button====================#
   self.img_arrow_Right = PhotoImage(file ="rArrow.png")
   self.btnrArrow1=Button(TopFrame2Right, width=160, height=60, state =DISABLED,command=loan,
 image=self.img_arrow_Right).grid(row=0, column =0,padx=2, pady =4)
self.btnrArrow2=Button(TopFrame2Right, width=160, height=60, state =DISABLED,command=deposit,
  image=self.img_arrow_Right).grid(row=1, column =0,padx=2, pady =4)
self.btnrArrow3=Button(TopFrame2Right, width=160, height=60, state =DISABLED,command=request_new_pin,
  image=self.img_arrow_Right).grid(row=2, column =0,padx=2, pady =4) self.btnrArrow4=Button(TopFrame2Right, width=160, height=60, state =DISABLED,command=statement,
  image=self.img_arrow_Right).grid(row=3, column =0,padx=2, pady =4)
 #3).========pin number button==================#
   #this are the pins buttons of the atm system
   self.img1 = PhotoImage(file ="one.png")
   self.btn1=Button(TopFrame1, width=160, height=60,command = insert1,image=self.img1).grid(row=2, column =0,padx=4, pady =4)
 self.img2 = PhotoImage(file ="two.png")
 self.btn2=Button(TopFrame1, width=160, height=60,command = insert2, image=self.img2).grid(row=2, column =1,padx=4, pady =4) self.img3 = PhotoImage(file ="three.png")
 self.btn3=Button(TopFrame1, width=160, height=60,command = insert3, image=self.img3).grid(row=2, column =2,padx=4, pady =4)
```

```
self.imgCE = PhotoImage(file ="cancel.png")

 self.btnCancel=Button(TopFrame1, width=160, height=60,command
= cancel,image=self.imgCE).grid(row=2, column =3,padx=4, pady
=4)

 #4).============******=======================#

 self.img4 = PhotoImage(file ="four.png")

 self.btn4=Button(TopFrame1, width=160, height=60,command =
insert4,image=self.img4).grid(row=3, column =0,padx=4, pady =4)

self.img5 = PhotoImage(file ="five.png")

 self.btn2=Button(TopFrame1, width=160, height=60,command =
insert5,  image=self.img5).grid(row=3, column =1,padx=4, pady =4)

 self.img6 = PhotoImage(file ="six.png")

 self.btn6=Button(TopFrame1, width=160, height=60,command =
insert6,image=self.img6).grid(row=3, column =2,padx=4, pady =4)

self.imgCl = PhotoImage(file ="clear.png")

self.btnClear=Button(TopFrame1, width=160,
height=60,command=clear,image=self.imgCl).grid(row=3, column
=3,padx=4, pady =4)
```

#5).==================******=====================#

self.img7 = PhotoImage(file ="seven.png")

self.btn7=Button(TopFrame1, width=160, height=60,command = insert7, image=self.img7).grid(row=4, column =0,padx=4, pady =4)

self.img8 = PhotoImage(file ="eight.png")

self.btn8=Button(TopFrame1, width=160, height=60,command = insert8, image=self.img8).grid(row=4, column =1,padx=6, pady =4)

self.img9 = PhotoImage(file ="nine.png")

self.btn9=Button(TopFrame1, width=160, height=60,command = insert9, image=self.img9).grid(row=4, column =2,padx=6, pady =4)

self.imgEnter = PhotoImage(file ="enter.png")

self.btnEnter=Button(TopFrame1, width=160, height=60,command= enter_Pin,image=self.imgEnter).grid(row=4, column =3,padx=6, pady =4)

#In this button we have to change the row and column , and we have to change the

#image which is neccsary .

#6).==========******==========#

```
    self.imgSp1 = PhotoImage(file ="empty.png")
    self.btnSp1=Button(TopFrame1, width=160, height=60,
    image=self.imgSp1).grid(row=5, column =0,padx=4, pady =4)
    self.img0 = PhotoImage(file ="zero.png")
     self.btn0=Button(TopFrame1, width=160,
height=60,command=insert0,
       image=self.img0).grid(row=5, column =1,padx=6, pady =4)
     self.imgSp2 = PhotoImage(file ="empty.png")
     self.btnSp2=Button(TopFrame1, width=160, height=60,
     image=self.imgSp2).grid(row=5, column =2,padx=6, pady =4)


     self.imgSp3 = PhotoImage(file ="empty.png")
     self.btnSp3=Button(TopFrame1, width=160, height=60,
      image=self.imgSp3).grid(row=5, column =3,padx=6, pady =4)


      #just trying to be consistent with naming  and check the
indentation and proper naming should#
       #be given   now go to function
```

```
if __name__=='__main__':
    root = Tk()
    application = atm(root)
    root.mainloop()
```

#the above part is the title as atm system for the project .

Now here is the GUI INTERFACE of ATM MANAGEMENT SYSTEM!

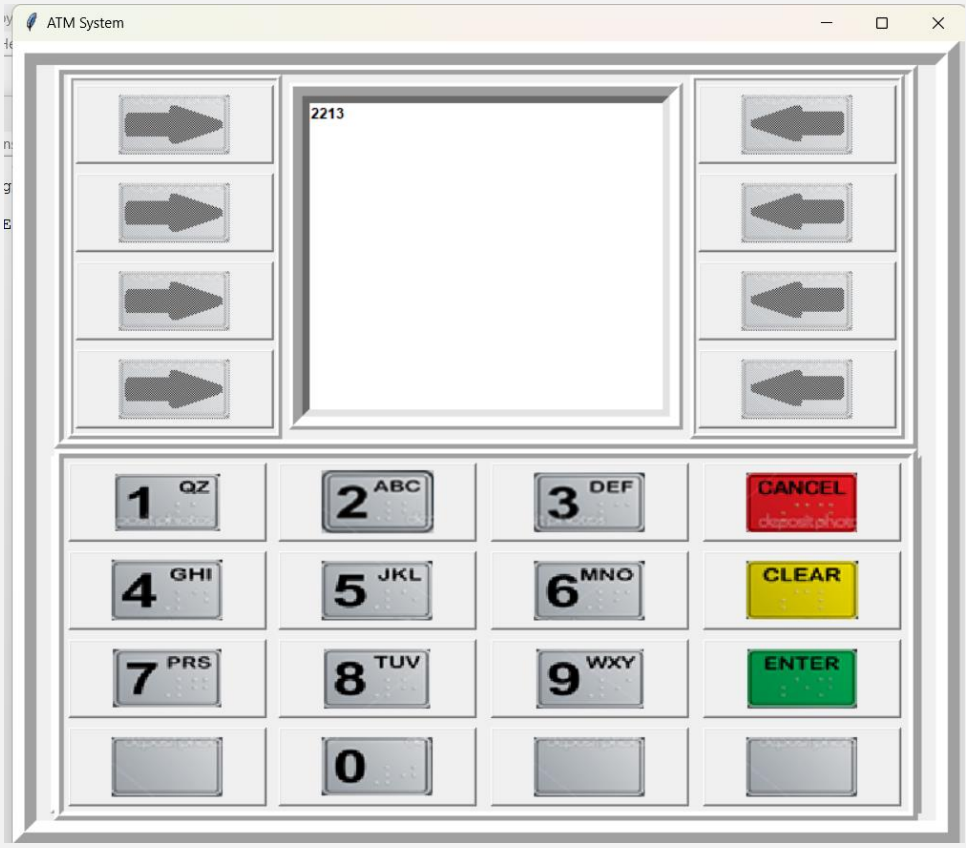
ENTER PIN BUTTON:-

```
def enter_Pin():
        pinNo = self.txtReceipt.get("1.0","end-1c")
        if((pinNo == str("2213"))or (pinNo == str("4323"))or (pinNo == str("5982"))):
            self.txtReceipt.delete("1.0",END)
            self.txtReceipt.insert(END,'\t\t        ATM' + "\n\n")
            self.txtReceipt.insert(END,'Withdraw Cash\t\t\t Loan' + "\n\n\n\n")
            self.txtReceipt.insert(END,'Cash With Receipt \t\t\t Deposit' + "\n\n\n\n")
            self.txtReceipt.insert(END,'Balance \t\t\t Request New Pin' + "\n\n\n\n")
            self.txtReceipt.insert(END,'Mini Statement \t\t\t Print Statement' + "\n\n\n\n")
```

Here I have defined a function enter pin to withdraw the cash , check balance, mini statement....

The Pin Number is “2213”,”4323”,”5982”.

Self.txtReceipt represents :- 'self' represent the class instance itself but it is an artifact you can call it as you like it can be 'mama' if you want but it is name "self" to help the dev remember that this is the instance itself.

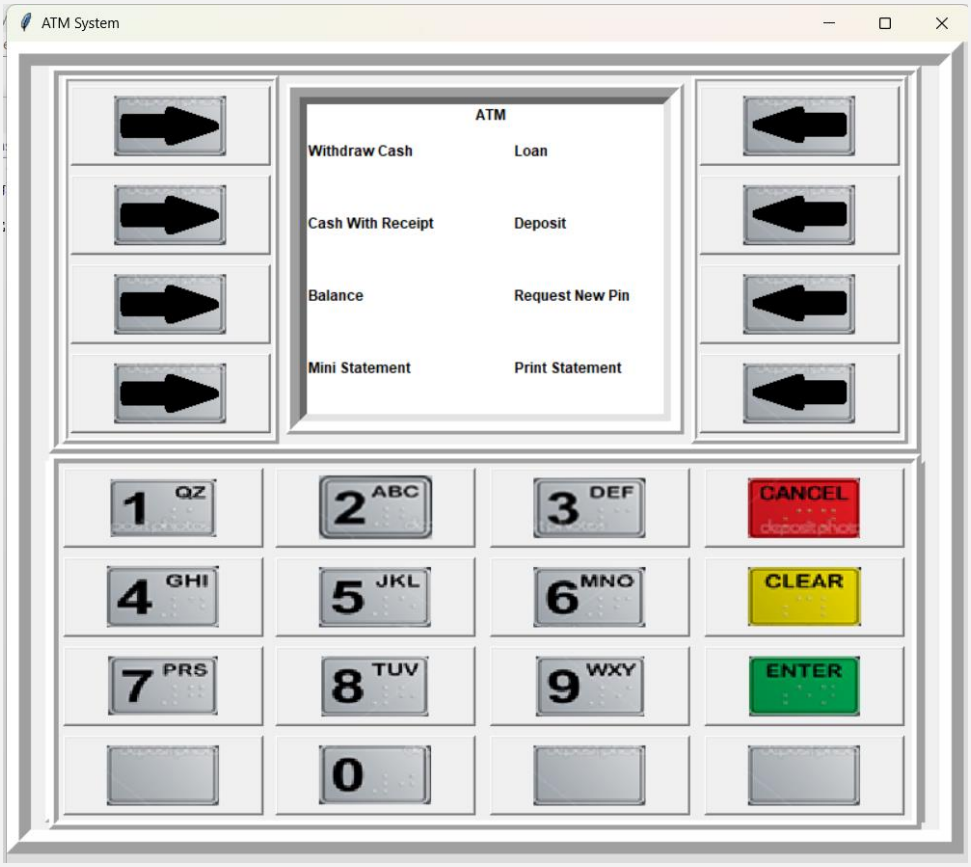


BUTTON:

Now after entering the pin, we must click on enter button then the arrow button's get enabled and the interface gets started and then we can see the multiple options.

self.btnlArrow1=Button(TopFrame2Left, width=160, height=60, state=NORMAL,command=withdrawcash,

This is the source code of button for left side and we are giving command to it to enable .

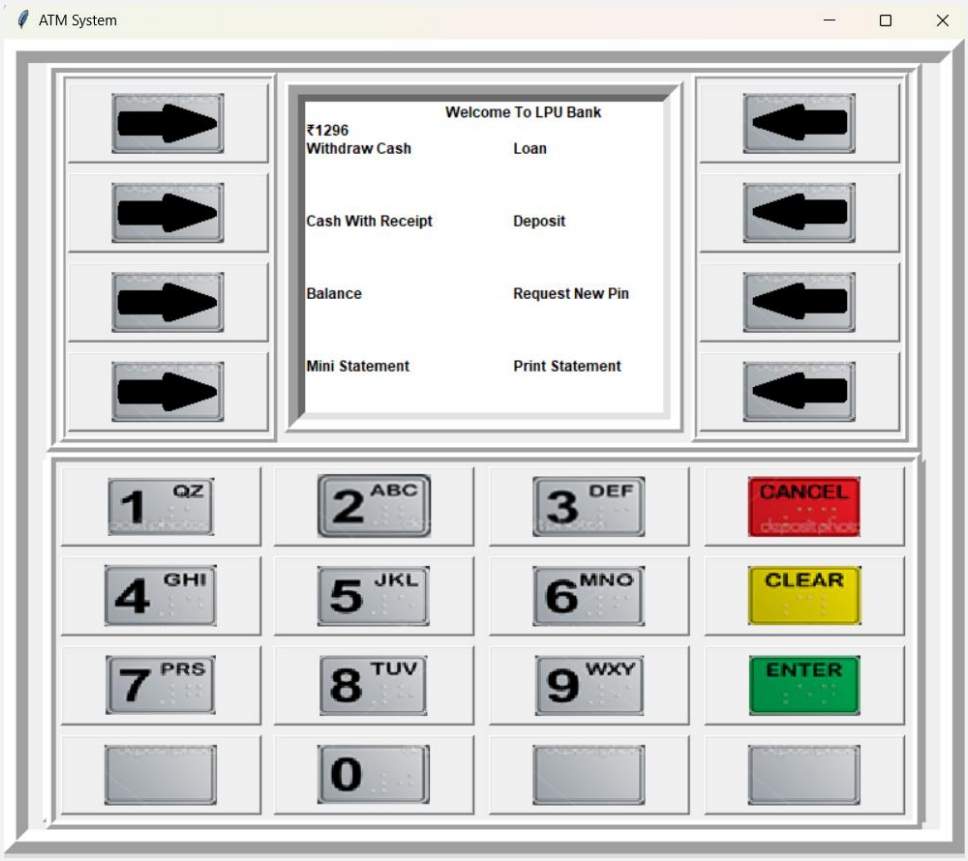


This is the interface of ATM

We can insert our card and withdraw the cash, check the mini statement, balance, and generate new pin.

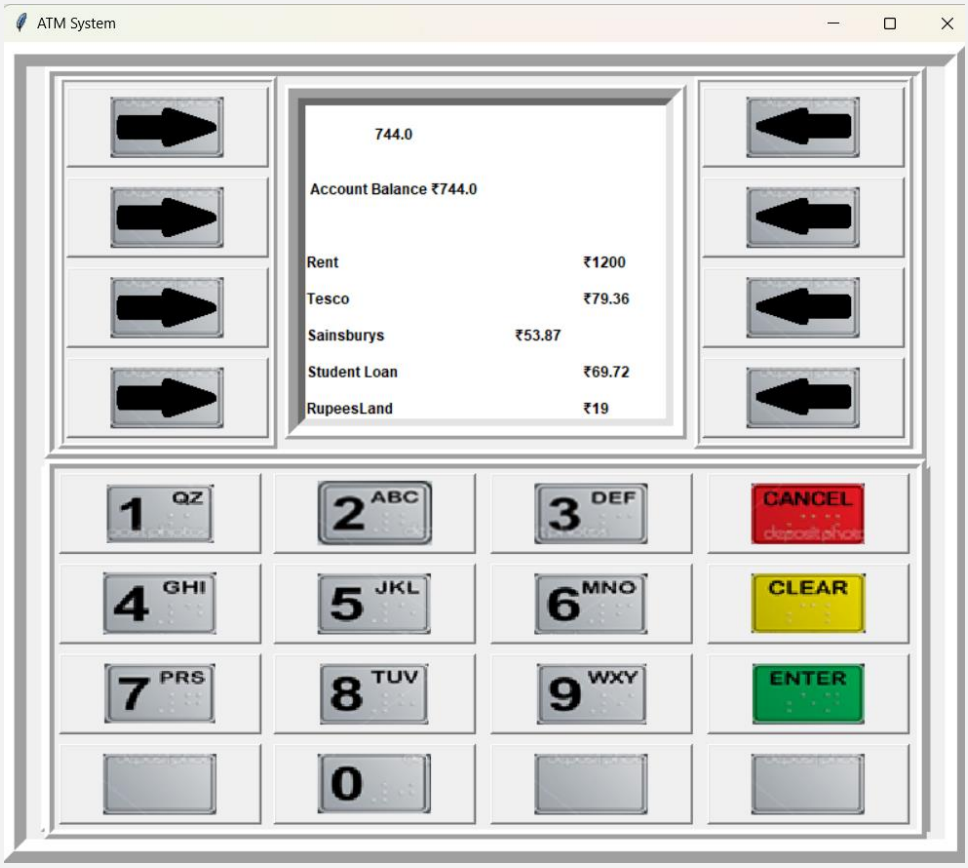
Check balance: -

To check balance, we must click on the balance button:



See here the total amount is Rupees 1296/-

```
def balance():
        self.txtReceipt.delete("1.0",END)
        self.txtReceipt.insert(END, '\t\t Welcome To LPU Bank \n')
        self.txtReceipt.insert(END, '₹1296' +"\n")
        self.txtReceipt.insert(END,'Withdraw Cash\t\t\t Loan' + "\n\n\n\n")
        self.txtReceipt.insert(END,'Cash With Receipt \t\t\t Deposit' + "\n\n\n\n")
        self.txtReceipt.insert(END,'Balance \t\t\t Request New Pin' + "\n\n\n\n")
        self.txtReceipt.insert(END,'Mini Statement \t\t\t Print Statement' + "\n\n\n\n")
        self.txtReceipt.insert(END, '\t\t Thanks for using  LPU Bank \n')
```



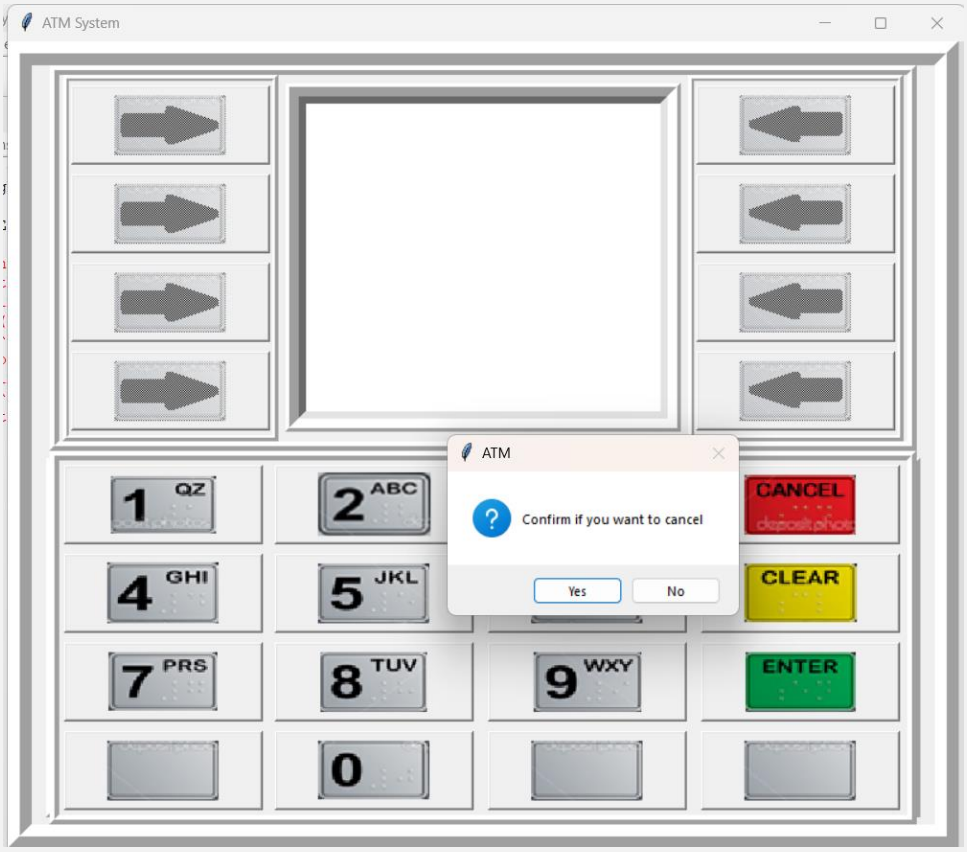
After withdrawing from ATM check the current savings or balance.

Or the statement.

```
def statement():
        pinNo1 = str(self.txtReceipt.get("1.0", "end-1c"))
        pinNo2 = str(pinNo1)
        pinNo3 = float(pinNo2)
        pinNo4 = float(1296-(pinNo3))
        self.txtReceipt.delete("1.0",END)
        self.txtReceipt.insert(END, '\n\t' + str(pinNo4)+ "\t\t")
        self.txtReceipt.insert(END,'\t\t\t\n\n Account Balance ₹' + str(pinNo4)+ "\n\n\n\n")
        self.txtReceipt.insert(END,'Rent\t\t\t\t ₹1200' + "\n\n")
        self.txtReceipt.insert(END,'Tesco\t\t\t\t ₹79.36' + "\n\n")
        self.txtReceipt.insert(END,'Sainsbury'+'s \t\t\t ₹53.87' + "\n\n")
        self.txtReceipt.insert(END,'Student Loan \t\t\t\t ₹69.72' + "\n\n")
        self.txtReceipt.insert(END,'RupeesLand \t\t\t\t ₹19' + "\n\n")
```

CANCEL OR EXIT BUTTON:

After withdrawing Money from the money, the very last step is to follow is to exit the interface so we have the Cancel button in the interface by which we can exit from the interface.




```
def cancel():
        Cancel = tkinter.messagebox.askyesno("ATM","Confirm if you want to cancel")
        Cancel > 0
        self.root.destroy()
        return
```

Now , here We Conclude our project, We have taken the source and help from the Internet and YouTube and the buttons We have placed in Our project is from the YouTube source zip file.

Link of the buttons:-https://drive.google.com/file/d/1IQRV...

Source of Project:- https://youtu.be/KkUhokBwVFU

CONCLUSION:

We have successfully designed an **ATM MANAGEMENT SYSTEM** using python and tkinter with a decent UI. To make things easy, this tutorial divided the various tasks into different python files. In the real world, we follow such practices to make things easy to build. Hence, whenever making a project, divide your goals into modules and integrate them at a later stage.