

PROFESSIONAL TRAINING REPORT
at
Sathyabama Institute of Science and Technology
(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Technology Degree in
Information Technology

By

MOHANA PRIYA R
(Reg. No. 37120048)



DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119. TAMIL NADU

JULY 2020



SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Professional Training Report is the bonafide work of MOHANAPRIYA R (Reg. No. 37120048) who underwent the professional training in "PYTHON" under our supervision from July 2020 to September 2020

Internal Guide

Dr.R.JEBERSON RETNA RAJ

Head of the Department

Dr. R.SUBHASHINI

Submitted for Viva voce Examination held on_____

Internal Examiner

External Examiner

DECLARATION

I, MOHANA PRIYA R (Reg. No. 37120048) hereby declare that the Professional Training Report on “**ATM SIMULATOR**” done by me under the guidance of Dr.R.JEBERSON RETNA RAJ at Sathyabamass Institute Of Science And Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of technology degree in Information Technology.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successful .I am grateful to them.

I convey my thanks to **Dr.T.Sasikala M.E., Ph.D, Dean**, School of Computing , **Dr.SUBHASHINI M.E., Ph.D.** , Head of the Department of Information and Technology for providing me necessary support and details at the right time during the progressive reviews.

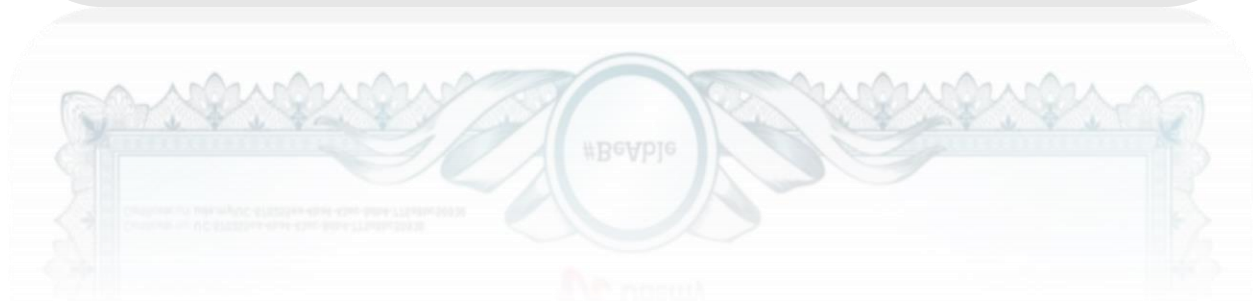
I would like to express my sincere and deep sense of gratitude to my Project Guide Dr. Jeberson Retna Raj M.E., Ph.D., for this valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my Thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

TRAINING CERTIFICATE



COURSE CERTIFICATE



This certificate is computer generated and can be verified by scanning the QR code given below. This will display the certificate from the NPTEL repository, <https://nptel.ac.in/noc/>

Roll No: NPTEL20CS36S1760904

To
MOHANA PRIYA.R
111(1) MARUTHI NAGAR, 3RD
CROSS STREET,
SEEVARAM, PERUNGUDI
CHENNAI
TAMIL NADU - 600096
PH. NO :7094146952



No. of credits recommended by NPTEL:1

An additional 1 credit may be awarded if the University deems it fit, based on the actual student effort involved.



NPTEL Online Certification

(Funded by the Ministry of HRD, Govt. of India)



This certificate is awarded to

MOHANA PRIYA.R

for passing the course

Python for Data Science

with Score* **90** %



Devendra Jaliha

Prof. Devendra Jaliha
Chairman
Centre for Continuing Education, IITM

Jan-Feb 2020
(4 week course)

Andrew Thangaraj

Prof. Andrew Thangaraj
NPTEL Coordinator
IIT Madras



Indian Institute of Technology Madras



***Continuous online assessment score**

To validate and check scores: <https://nptel.ac.in/noc>

ABSTRACT

The “ ATM SIMULATOR “ project is implemented using python. The aim of the project is to design a user interactive system that helps the user in performing various transaction it can perform some basic function such as enquiring some bank statements and interactive for the user .By using this concept user can able to reset their pin details. Its more helpful for the user.

TABLE OF CONTENT

CHAPTER NO.	TITLE	PAGE NO
	ABSTRACT	8
	LIST OF FIGURES	11
	LIST OF ABBREVIATION	12
1	INTRODUCTION	13
2	AIM AND SCOPE OF THE PRESENT INVESTIGATION	14
	2.1 AIM	14
	2.2 PROBLEM DEFINITION	14
	2.3 INVESTIGATION AND EVIDENCE OF PROBLEM DEFINITION	14
	2.3 PROPOSED SYSTEM	14
	2.4. SCOPE OF THE SYSTEM	15
3	SYSTEM REQUIREMENT AND ANALYSIS	16
	3.1. HARDWARE REQUIREMENTS	16
	3.2. SOFTWARE REQUIREMENTS	16
	3.3 SYSTEM ANALYSIS	17
	3.3.1. STUDY OF EXISTING SYSTEM	17

	3.3.2DISADVANTAGES	17
	3.4 FEASABILITY STUDY	17
	3.5.ECONOMICAL FEASABILITY	17
4	METHODS AND ALGORITHM USED	18
	4.1 PACKAGE USED	18
	4.1.1 GETPASS	18
	4.1.1 OPERATING SYSTEM	18
	4.1.1 STRING	18
	4.2 ALGORITHM	19
	4.2.1 BRANCHING STATEMENT	19
5	SYSTEM DESIGN OR MODULE	21
	5.1 SYSTEM DESIGN	21
	5.2 PROGRAM FLOWCHART	22
6	SUMMARY AND CONCLUSION	23
	6.1.CONCLUSION	23
	6.2.APPENDIX	23
	6.2.1 RESULT ANALYSIS	23
	6.2.2 SOURCE CODE	28

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO
1.1	FLOW CHART	20
1.2	SYSTEM DESIGN	21
1.3	PROGRAM FLOWCHART	22

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
ATM	Automated Teller Machine
OS	Operating System
GUI	Graphical User Interface

CHAPTER 1

INTRODUCTION

Automated Teller Machine enables the client of the bank allow them to access their account without going to the bank. This achieved only by development of the application using some concepts.

When the product is implemented the user who uses the product will be able to see all the information and services provided by the ATM, when the user enters the necessary options and arguments. The product also provides services like request and done some operations deposit and statement etc. It is one of the most important process behind the ATM system.

The program is designed in such a way that the user has to enter the pin .once the pin is verified the user will be allowed to doing further steps like what the user wants to do after the verification process. Like statement checking, account details lodgement and withdrawal steps. When the option is entered particular argument for that option alone is displayed on the screen.

CHAPTER 2

AIM AND SCOPE OF THE PRESENT INVESTIGATION

2.1. AIM:

The purpose of this program is to create a simple Automatic Teller Machine(ATM).The ATM is used by the customers of various banks. Each customer has two accounts like checking account and savings account. This will allow the customer to deposit and withdraw money from the current account and it will update the amount details accordingly.

2. 2.PROBLEM DEFINITION:

The system mainly used by the bank customers. When a customer comes to the ATM centre to get update or verify about their own account details it takes more time so for these concerns this system is implemented to reduce customer time and in more effective manner. It involves various operations like changing pin and updating the balance details once the amount has been retrieved from the account.

2.3. INVESTIGATION AND EVIDENCE OF PROBLEM DEFINITION:

Now a day's everyone is very busy in their works. So they feel that the job issue must be easier so that the system is used to reduce their work pressure which is done in the ATM system. Instead of working with lots of paper into a record of file and it may missed somewhere in sometimes so this system helps to keep the record of the customer.It is very easy to access

.

2.4. PROPOSED SYSTEM:

The system customer transaction satisfies the requirements of the existing systems in full fledged manner. Through this system customer can make fast transaction and view the last transactions easily.

2.5. SCOPE OF THE SYSTEM:

The main objective of this system is to speed up the time of transaction. So no manual transaction is needed which is very important need in our daily life. The application terminates when the user enter the quit keyword. The application should interact with the user via a simple GUI. This system does not actually communicate with the ATM. Its just a simple user GUI.

It includes some important concepts like

- To render the accurate services to the customer.
- To reduce the fraud or mal functions of unwanted customers.
- Achieve the speedy processing of customer data.
- To reduce the error processing.

CHAPTER 3

SYSTEM REQUIREMENT AND ANALYSIS

3.1. HARDWARE REQUIREMENTS:

Processor: Intel Pentium 4 or later.

RAM: 1 GB

Hard Disk: 410 GB.

Pointing Device: keyboard or touch screen.

3.2. SOFTWARE REQUIREMENTS:

Operating System: Window 7,8,10.

Front End: Python.

Developing Platform: Jupyter Notebook.

3.3. SYSTEM ANALYSIS:

3.3.1. Study of existing system:

In the manual system, firstly the bank manager and its staff have to manage information regarding the accounts and transaction of all the customers manually. Doing this kind of job is really tedious.

3.3.2. Disadvantages:

- More time consumed.
- More hard work to maintain all the records.
- Bulk of paper is searched for the single record.

3.4. FEASABILITY STUDY:

The system is developed in python. It provides comprehensive function to make the system in user friendly way. The machine configuration also supports the software.

3.5. ECONOMICAL FEASABILITY:

The cost of converting from manual system to new automation it more useful for the customer. It's easy to develop cost efficient process to implement this system.

CHAPTER 4

METHODS AND ALGORITHM USED

4.1. PACKAGES USED:

4.1.1. **Getpass:**

It promotes the user for a password without echoing. The get pass module provides a secure way to handle the password prompts where prompts interact with the users via the terminal. The get pass() function is used to prompt to user using string prompt and reads the inputs from the user as a password. In default the it shows the user pin in the star format like eg. password is "abc" but it shows password: "****".

4.1.2. **import os:**

The OS Module in python provides a way of using operating system functionality. The functions that the os module provides allows you to interface with underlying operating system that python is running on be that windows Mac or Linux.

4.1.3. **import string:**

Python strings are immutable which means they cannot be changed after they are created (java strings also use this immutable styles). Since can't be changed we construct new string as we go to represent computed values. The len(string) functions returns the length of a string. Its help us to use built-in methods like (length, upper(), lower()).

4.2. ALGORITHM:

Here simple algorithm is used like linear search algorithm is used in the pin verification .It will be used for searching the appropriate pin and user name is present or not. It requires more time complexity for enormous number of records. And it's easy to understand how the flow works. And simple branching statement used for condition checking.

Eg:

```
n=[2,6,7,3,7,8]
x=7
for i in range(len(n)):
    if(n[i]==x):
        print(i)
        break
```

4.2.1.Branching Statement:

Syntax :

```
if(expression/condition):
    statement 1
else if(expression/condition):
    statement 2
else if(expression/condition):
    statement 3
else:
    statement 4
```

FLOWCHART:

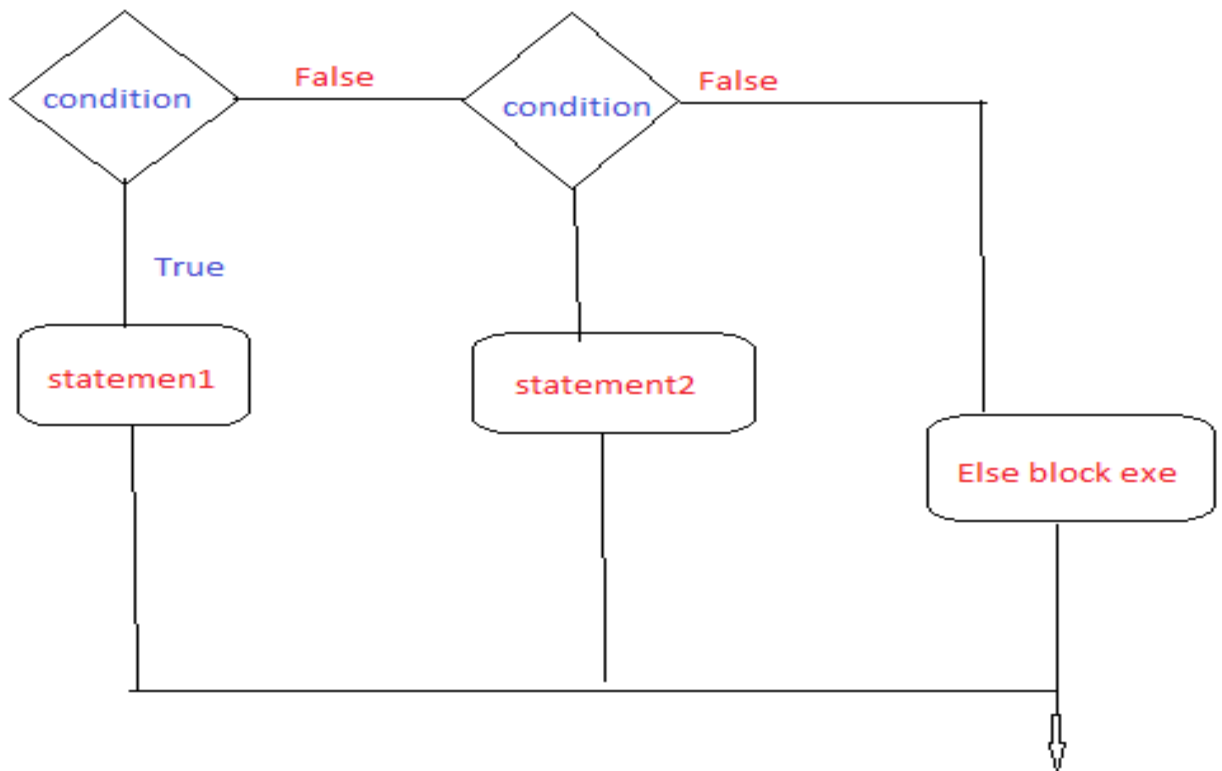


FIGURE 1.1

CHAPTER 5

SYSTEM DESIGN OR MODEL

5.1. SYSTEM DESIGN:

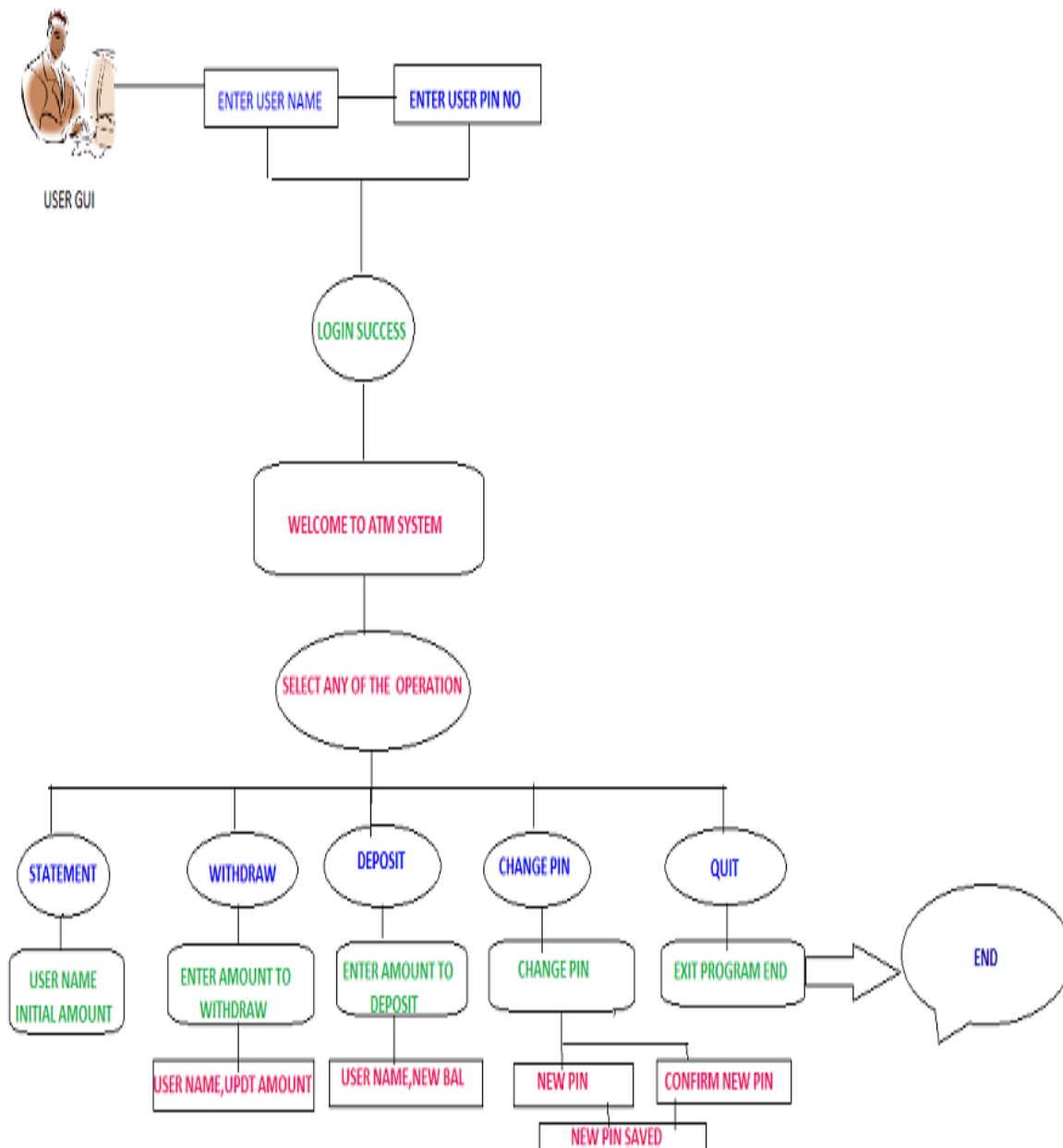


Figure 1.2 system design

5.2. PROGRAM FLOWCHART:

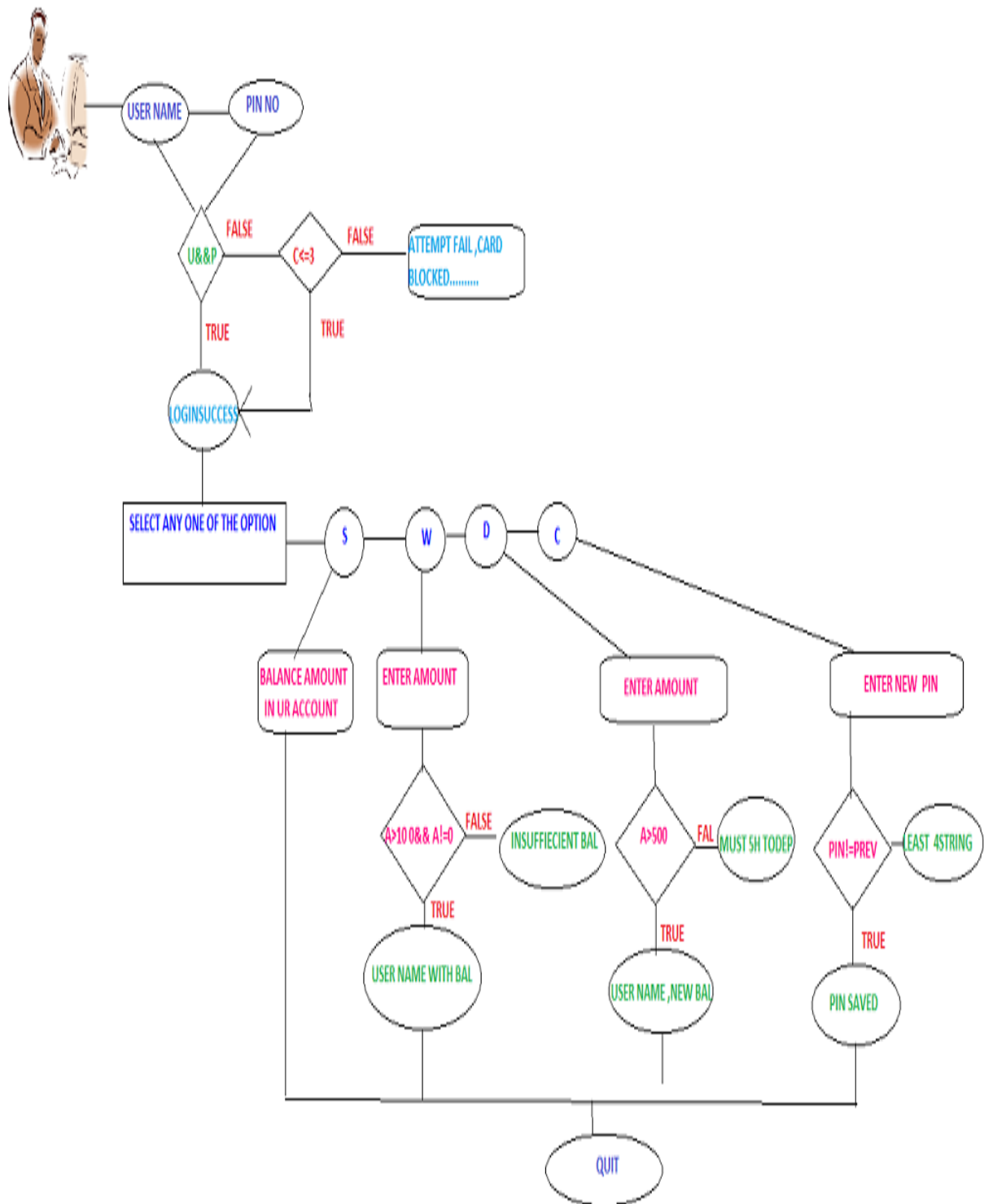


Figure 1.3 flow of program

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 CONCLUSION:

Information technology is developing day by day and its has positive influence on Bank management and further general management systems like net banking, hospital management systems. Private companies where developing more application lead to way of automation everywhere .It will reduce the time of customer and most help full for the senior citizens by using this they can able to access it via phone also without going to the near ATMs.

6.2. APPENDIX:

6.2.1. Result Analysis:

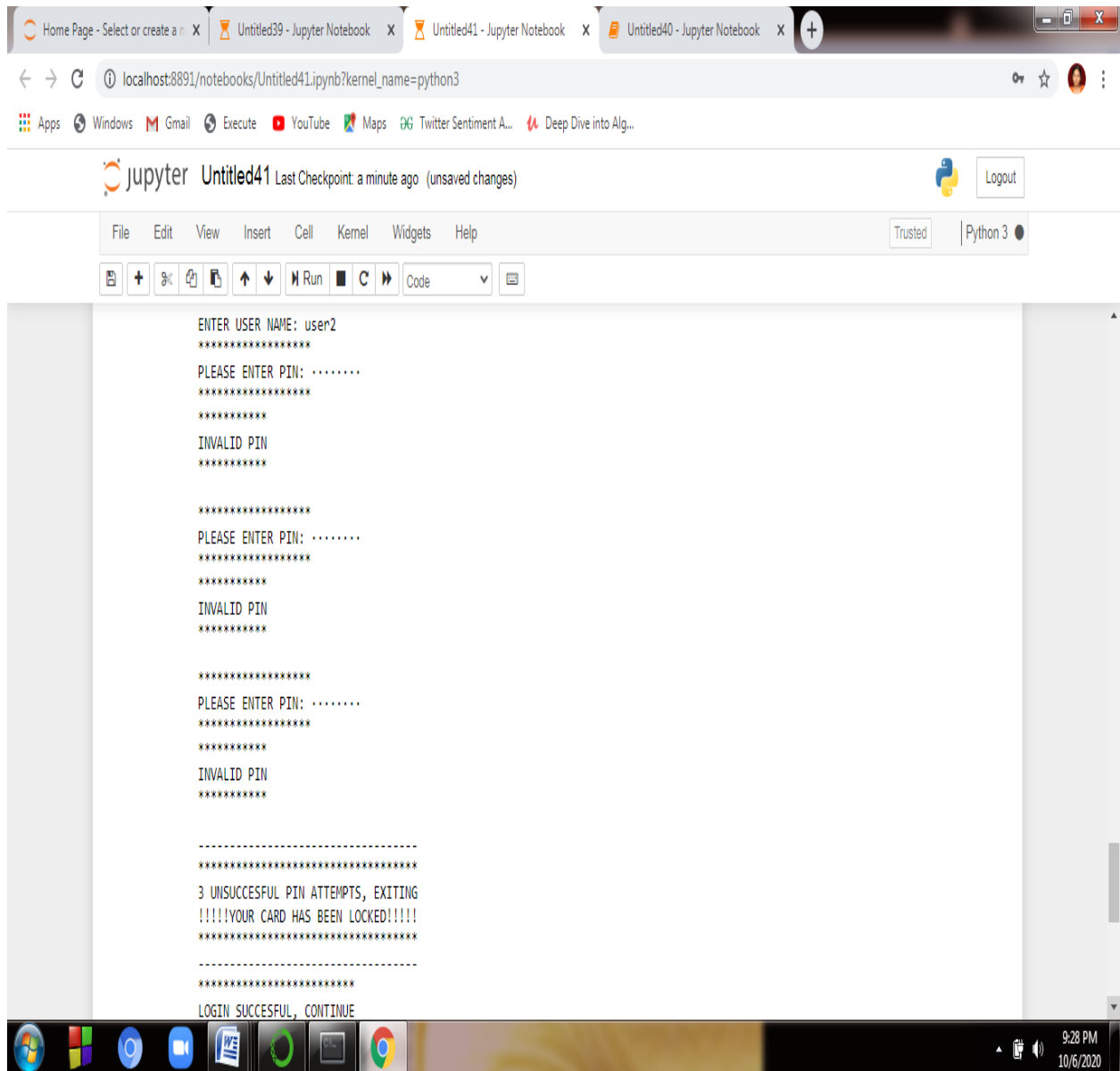


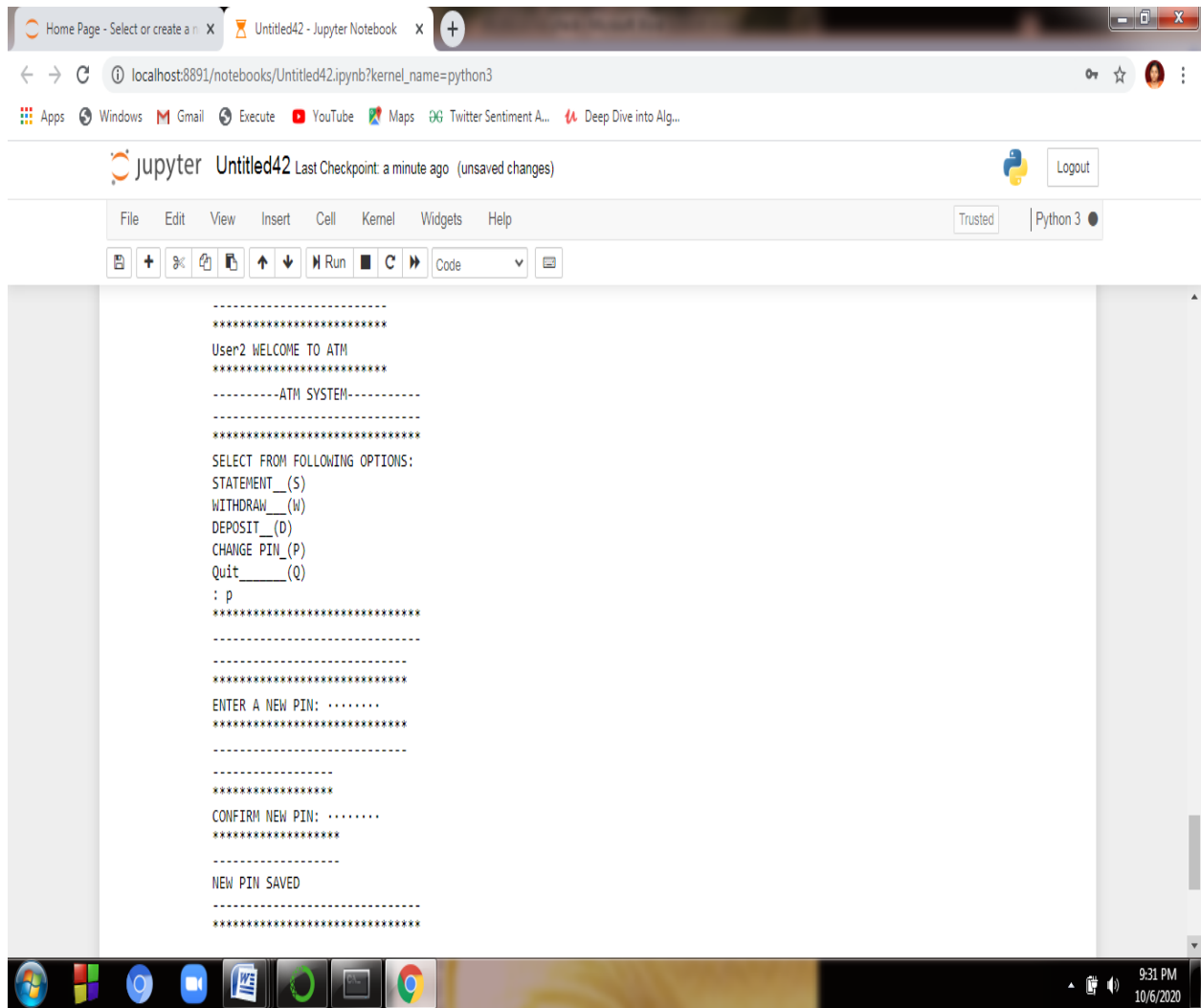
The screenshot shows a Jupyter Notebook titled 'Untitled39' running on a local host. The notebook contains a Python script that simulates a banking application menu. The menu options are: DEPOSIT (D), CHANGE PIN (P), and Quit (Q). The user has selected 'D' for deposit. The script prompts the user to enter the amount they want to deposit, and the user has entered '50000'. The script also displays the user's current balance of 25000. The Jupyter Notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for running, saving, and other actions. The status bar at the bottom shows the time as 9:24 PM on 10/6/2020.

```
DEPOSIT__(D)
CHANGE PIN__(P)
Quit__(Q)
: s
*****
*****
*****
User2 YOU HAVE : 25000 BALANCE AMOUNT ON YOUR ACCOUNT.
*****
*****
SELECT FROM FOLLOWING OPTIONS:
STATEMENT__(S)
WITHDRAW__(W)
DEPOSIT__(D)
CHANGE PIN__(P)
Quit__(Q)
: d
*****
*****
ENTER AMOUNT YOU WANT TO DEPOSIT: 50000
*****
*****
```

The screenshot shows the same Jupyter Notebook titled 'Untitled39' running on a local host. The notebook contains a Python script that simulates a banking application menu. The menu options are: DEPOSIT (D), CHANGE PIN (P), and Quit (Q). The user has selected 'D' for deposit. The script prompts the user to enter the amount they want to deposit, and the user has entered '50000'. The script also displays the user's current balance of 25000. The Jupyter Notebook interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for running, saving, and other actions. The status bar at the bottom shows the time as 9:25 PM on 10/6/2020.

```
ENTER AMOUNT YOU WOULD LIKE TO WITHDRAW: 20000
*****
*****
YOUR NEW BALANCE IS: 55000 BALANCE
*****
*****
SELECT FROM FOLLOWING OPTIONS:
STATEMENT__(S)
WITHDRAW__(W)
DEPOSIT__(D)
CHANGE PIN__(P)
Quit__(Q)
: q
*****
*****
*****
SELECT FROM FOLLOWING OPTIONS:
STATEMENT__(S)
WITHDRAW__(W)
DEPOSIT__(D)
CHANGE PIN__(P)
Quit__(Q)
: 
```





6.1.2. Source Code:

```
import getpass
import string
import os

users = ['user', 'user2', 'user3']
pins = ['1234', '2222', '3333']
amounts = [15000, 25000, 35000]
count = 0

while True:
    user = input('\nENTER USER NAME: ')
    user = user.lower()
    if user in users:
        if user == users[0]:
            n = 0
        elif user == users[1]:
            n = 1
        else:
            n = 2
        break

    else:
        print('*****')
        print('INVALID USERNAME')
        print('*****')

while count < 3:

    print('*****')
    pin = str(getpass.getpass('PLEASE ENTER PIN: '))
    print('*****')

    if pin.isdigit():
        if user == 'user1':
            if pin == pins[0]:
                break
            else:
                count += 1
                print('*****')
                print('INVALID PIN')
                print('*****')
```

```

        print()

    if user == 'user2':
        if pin == pins[1]:
            break
        else:
            count += 1
            print('*****')
            print('INVALID PIN')
            print('*****')
            print()

    if user == 'user3':
        if pin == pins[2]:
            break
        else:
            count += 1
            print('*****')
            print('INVALID PIN')
            print('*****')
            print()

    else:
        print('-----')
        print('*****')
        print('PIN CONSISTS OF 4 DIGITS')
        print('*****')
        print('-----')
        count += 1

if count == 3:
    print('-----')
    print('*****')
    print('3 UNSUCCESSFUL PIN ATTEMPTS, EXITING')
    print('!!!!!!YOUR CARD HAS BEEN LOCKED!!!!!!')
    print('*****')
    print('-----')
    exit()

print('*****')
print('LOGIN SUCCESSFUL, CONTINUE')
print('*****')
print()
print('-----')
print('*****')
print(str.capitalize(users[n]), 'WELCOME TO ATM')
print('*****')

```

```

print('-----ATM SYSTEM-----')
# Main menu
while True:
    #os.system('clear')
    print('-----')
    print('*****')
    response = input('SELECT FROM FOLLOWING OPTIONS:
\nSTATEMENT__(S) \nWITHDRAW__(W) \nDEPOSIT__(D) \nCHANGE
PIN_(P) \nQuit_____(Q) \n: ').lower()
    print('*****')
    print('-----')
    valid_responses = ['s', 'w', 'd', 'p', 'q']
    response = response.lower()
    if response == 's':
        print('-----')
        print('*****')
        print(str.capitalize(users[n]), 'YOU HAVE :',
amounts[n], 'BALANCE AMOUNT ON YOUR ACCOUNT.')
        print('*****')
        print('-----')

        elif response == 'w':
            print('-----')
            print('*****')
            cash_out = int(input('ENTER AMOUNT YOU WOULD LIKE TO
WITHDRAW: '))
            print('*****')
            print('-----')
            if cash_out%10 != 0:
                print('-----')
                print('-----')

            print('*****')
            print('AMOUNT YOU WANT TO WITHDRAW MUST TO MATCH
100 RUPEES NOTES')

            print('*****')
            print('-----')
            print('-----')
            elif cash_out > amounts[n]:

                print('*****')
                print('YOU HAVE INSUFFICIENT BALANCE')
                print('*****')
            else:

```

```

        amounts[n] = amounts[n] - cash_out
        print('-----')
        print('*****')
        print('YOUR NEW BALANCE IS: ', amounts[n],
'BALANCE')

        print('*****')
        print('-----')

    elif response == 'd':
        print()
        print('-----')
        print('*****')
        cash_in = int(input('ENTER AMOUNT YOU WANT TO DEPOSIT:
'))

        print('*****')
        print('-----')
        print()
        if cash_in%10 != 0:
            print('-----
-----')

            print('*****
')

            print('AMOUNT YOU WANT TO DEPOSIT MUST TO MATCH
500 RUPEES NOTES')

            print('*****
')

            print('-----
-----')
            else:
                amounts[n] = amounts[n] + cash_in
                print('-----')
                print('*****')
                print('AFTER DEPOSIT YOUR NEW BALANCE IS: ',
amounts[n], 'AMOUNT')
                print('*****')
                print('-----')

    elif response == 'p':
        print('-----')
        print('*****')
        new_pin = str(getpass.getpass('ENTER A NEW PIN: '))
        print('*****')
        print('-----')
        if new_pin.isdigit() and new_pin != pins[n] and
len(new_pin):
            print('-----')

```

```

        print('*****')
        new_ppin = str(getpass.getpass('CONFIRM NEW PIN:
'))

        print('*****')
        print('-----')
        if new_ppin != new_pin:
            print('-----')
            print('*****')
            print('PIN MISMATCH')
            print('*****')
            print('-----')
        else:
            pins[n] = new_pin
            print('NEW PIN SAVED')

    else:
        print('-----')
        print('*****')
        print('    NEW PIN MUST CONSIST OF 4 DIGITS \nAND
MUST BE DIFFERENT TO PREVIOUS PIN')
        print('*****')
        print('-----')
    elif response == 'q':
        exit()
    else:
        print('-----')
        print('*****')
        print('RESPONSE NOT VALID')
        print('*****')
        print('-----')

```

6.4.REFERENCES:

<http://W3schools.com>

www.tutorialpoints.com