

# **PROJECT REVIEW REPORT**

---

## **SECURED DATA ENCRYPTION**

---

---

**SUBMITTED BY**

**ARPIT BHATTAR [REG. NO. - 18BCE0124 ]  
HARSHIT GOEL [REG. NO. - 18BCE0165 ]**

**CSE3501 : INFORMATION SECURITY ANALYSIS**

**AND AUDIT**



**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING  
VIT – VELLORE CAMPUS  
FALL SEMESTER 2020 – 2021**

## I ABSTRACT

In the current scenario, Data Security is required to transfer confidential information over the network. In inclusive range of applications, Security is also challenging. For data security Cryptographic algorithms play a vital role against spiteful attacks. In the popular performances of Public Key Infrastructures, **RSA** algorithm is extensively used and hashing technique. The core theme of the project focuses on the development of Security in ATM Transaction System in **Python using Hash and RSA algorithms**. The report contains the strategy used in making ATM Transactions, Comparison with different kinds of algorithm, advantages of Hash algorithm.

## II INTRODUCTION

The growth in electronic transactions has resulted in a greater demand for fast and accurate user identification and authentication .Over the past three decades, consumers have been largely depending on and trusting the Automatic Teller Machine (ATM) to conveniently meet their banking needs. However, despite the numerous advantages of the ATM system, ATM fraud has recently become more widespread. So it becomes a necessity to secure the confidential data of the user like personal identification numbers( PIN ) and all the transactions . ATM card that is assigned a Personal ID Number (PIN). Hashing technique creates a unique encrypted result from a data set. The encrypted result is called a hash, a signature, or a digest, all of which mean the same thing. It is a one-way hash, which means that the hash resulting from the algorithm is not meant to be unencrypted. For this reason we use it in ATM security. You have an ATM card that is assigned a Personal ID Number (PIN). The card provider used a hash algorithm to create a hash from your PIN which is stored on the card on a magnetic stripe. When the card is used, it is inserted into a slot on a machine, and the machine asks for the PIN. The user is expected to enter the PIN, which the machine uses to create a hash, which is then compared to the hash stored on the card. The user is not allowed to use the card if the two hashes do not match.

### **III OVERVIEW**

In ATM Transactions we only focus on ATM features and mainly on security. The feature transactions is a topic of deep interest in the Finance domain. The system should be good enough to represent the ATM in feature form. Features of ATM are in two modules – admin and user. Admin module features include creation of new accounts, pin creation and updating them in the database. Admin module feature category includes taking the input profile information.

This method uses RSA and Hash function algorithm to implement this feature. This method is often termed as One way encryption method. Both these methods fall under admin module feature category. Then this data is decrypted to information set that represents user's bank information to server. Encryption is a multistage technique. Encryption uses all the relevant information from the ATM output information that is given by a user that is to be sent to the server. It forms a encrypted text of the account number, pin where the output of other is sent to the server using RSA. User module features include pin updating, withdrawal, balance enquiry etc. The method used to encrypt this user information is RSA and hash function algorithm. We have implemented this system user friendly and totally secure to protect user information.

In ATM transactions method, we can also implement SHA(Secure Hash Algorithm), MD5(Message Digest). ATM are the most important part to maintain of a user's financial life.

#### **Attacks on various points**

The digital banking faces many robbery cases which are divided into the following main categories: 1. Attack on the digital infrastructure for accessing information about different funds transfer. 2. Digital Infrastructure attack on ATM management 3. Clients Side attacks while performing e-banking

#### **Attacks related to ATM**

1. Fraud - Fraud is done by the usage of fake cards. Skimming is one of the method by which fraud is performed. The attackers install the skimmers in the ATM's card reading slots. These skimmers are not visible to the users of the ATM. The person coming for service of the machine checks these skimmers manually.
2. Physical -As the name suggests the attackers physically try to loot the cash out from the ATM. This is the most common attack. These are prevented deploying security personnel for the ATM.

3. Logic -These attacks mainly include use of malware. Skimmer is one of the examples of the logic attack. The attacker tries to get the knowledge about the algorithm from which the bank is performing the transactions.

#### **IV LITERATURE REVIEW**

Studies on the topic state that [1] In the modern years the need of security has amplified many folds. It explains that the need to secure data is not new and goes way back to the time of 1st world war and even further back to the time of Julius Caesar. This paper provides a history of the data encryption techniques used in the history and how they have developed from Caesar cipher to DES and from DES to Triple-DES, AES and recent algorithms. This study explained the pros and cons of earlier algorithms and why they aren't used today. The main focus of this paper is initially general data hiding methods and then cryptographic algorithms. A brief study of a paper [2] presented a new Hybrid security algorithm for RSA cryptography named as Hybrid RSA (HRSA). Here calculation of "public key" (P) plus "private key" (Q) depends on the value of M, i.e. the product of 4 prime numbers. So difficulty involved in factorizing the M increases. Another appealing feature about this algorithm is the fact that the computation of P and Q involves the calculation of some more midway factors which makes the calculation more complex. This approach eradicates the transfer of variable x and M, where x is the multiplication of 2 prime numbers a and b. Thus the proposed approach gives more safe path for encryption and decryption procedure. To substantiate this statement, the "key generation time", "encryption speed" and "decryption speed" of the proposed algorithm HRSA are compared with conventional RSA and ERSA techniques. Another research was referred because of the use of a wireless medium in the project whose problem statement was [3] ,Security over Vehicular ad hoc network by means of Wi- Fi IEEE 802.11p standard and recognizing accurate attacker vehicle is a main challenge over VANET. Reducing costs of computation, effective expenditure of limited resources and giving uncompromised security was always a challenge over VANET. In this study , RSA algorithm based encryption and decryption techniques and implementing of limit with double

RSU has been replicated using MATLAB software. Such background can be used while designing better MAC protocols, transmitting schemes, security features in VANETs. Providing confidentiality, message integrity, detecting and removing nasty and misbehaving nodes, from VANET is focus of this study.

The literature of behavioural studies on ATMs has mainly focused on adoption and diffusion of technology, impact of technology adoption from customers perspective, suppliers perspective, and bankers perspective [1].

Major Technology Adoption Models as per the study of Norris and Yin (2008), Technology Adoption Research is almost twenty-five years old and there are around eight important theories of adoption. All these eight theories are derived from the foundation of innovation diffusion and Technology acceptance model. The exception among eight models is the Social Cognitive Theory. Technology acceptance model is quite individual focused among eight adoption models. However, other models focus on how diffusion of innovation takes place within the firm. Oliveira and Maria (2011) have explored that there is a dearth of academic literature on reviews of adoption model at firm level used in Information Technology literature[1]. Author discussed Diffusion of Innovation Theory and Technology, Organization, and Environment (TOE)[7]. For more complex new technology adoption, it is important to integrate more than one theoretical model to achieve a better understanding of the adoption phenomenon. But in many cases, technology adoption research is a replication without substantive theoretical advances. However, there are sample opportunities to make theoretical advances using our current knowledge as the starting point[5]. These are the conclusions of the authors based on their review and comparison of major milestones of technology adoption research, Job Satisfaction research and Theory of Planned behaviour. Most of the theories have emphasized on the factor influencing adoption behaviour and the process of diffusion of technology. These models also reveal the individual difference and cultural difference in technology adoption.

The first developed algorithm for the ATM transactions used to extract features of the bank, for e.g. Cash withdrawal, deposit or balance enquiry. These features were selected in a region form to perform transaction[4]. There was only simple encryption technique used like for example play fair cipher technique. This method was good but didn't showed great results because of less amount of cryptography used.

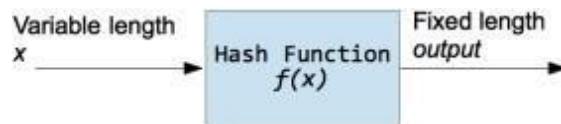
## V ALGORITHM USED IN OUR PROJECT

In ATM Transactions system, we need to secure sensitive data of user like PIN, etc.

To maintain this security, we implemented this system cryptography using hash function and RSA algorithm.

### HASH FUNCTION ALGORITHM

A hash function is a function that takes input of a variable length sequence of bytes and converts it to a fixed length sequence. It is a one-way function. This means if  $f$  is the hashing function, calculating  $f(x)$  is pretty fast and simple, but trying to obtain  $x$  again will take years. The value returned by a hash function is often called a hash, message digest, hash value, or checksum. Most of the time a hash function will produce unique output for a given input. However, depending on the algorithm, there is a possibility to find a collision due to the mathematical theory behind these functions.



Hash functions are used inside some cryptographic algorithms, in digital signatures, message authentication codes, manipulation detection, fingerprints, checksums (message integrity check), hash tables, password storage and much more.

Some of the Hash function algorithm examples are : • MD5 SHA (SHA1,SHA224, SHA256,SHA384,SHA512)

### RSA ALGORITHM

RSA algorithm is asymmetric cryptography algorithm. Asymmetric actually means that it works on two different keys i.e. Public Key and Private Key. As the name describes that the Public Key is given to everyone and Private key is kept private .

Key Generation: - 1. Generate two large prime numbers suppose  $p$  &  $q$ . 2.  $n = p * q$

3.  $m = \Phi(n) = (p-1) * (q-1)$  4. Choose a small number  $e$ , co-prime to  $m$  and  $\text{GCD}(m, e) = 1$  such that  $1 < e < \Phi(n)$  5. Find  $d$  such that  $(d * e) \bmod \Phi(n) = 1$

- Cipher text = (plain text)  $e \bmod n$
- Plain text = (Cipher text)  $d \bmod n$

The ATM sends the pin using RSA encryption to the server. The Server decrypts that and uses hash function for further processing.

## VI PROPOSED METHODOLOGY

To implement ATM transactions we have divided our project into two parts. One module comprises of the Administrator and the other one is the ATM.

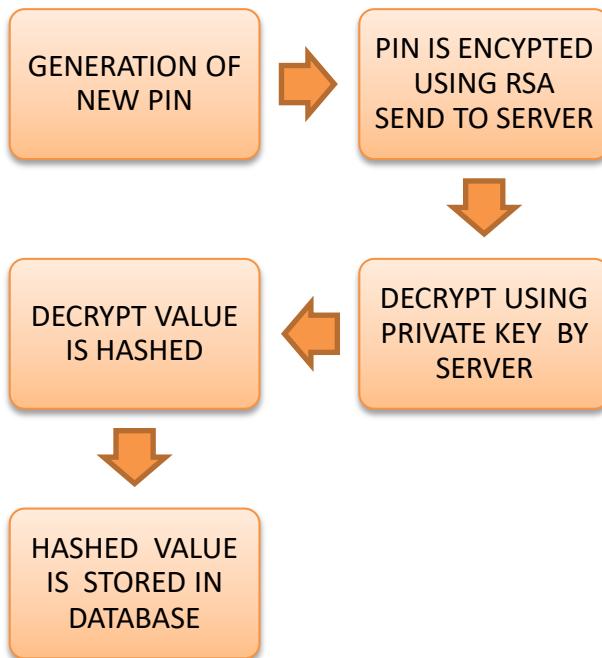
The Admin module is responsible for creation of the database, and handling the customers. Admin module just assigns the card no to the individuals that come to open a new bank account. The bank balance is updated and stored in the database. There is no pre-assigned pin for the users; the user sets his pin on his own when he goes to the ATM for the first time.

In the ATM when the user starts his transaction, the ATM sends the data to the server. The server then checks and compares them in the database. If the card number is wrong then it asks to enter the correct card else it asks for the pin.

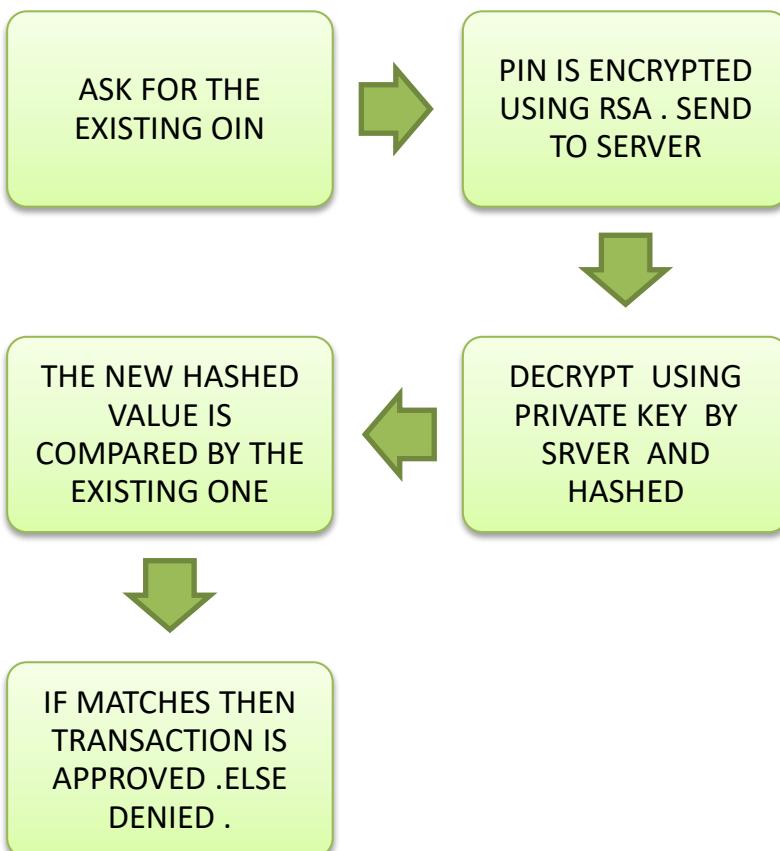
If coming for the first time then it asks for the new pin generation else for his transaction code. The pin is encrypted by the RSA algorithm and sent to the Server, the server firstly decrypts the pin and then hashes the pin.

Hashing is used because it is one-way profile and no Man in the middle (MIM) attack can take place over it. If user comes for the first time, hashed pin is stored in the database. Else the hashed pin is then compared to the stored hashed pin. If the pin matches, it transacts the amount and the same is updated in the database. If pin doesn't match it asks for correct pin and aborts the transaction.

## USER COMING FOR THE FIRST TIME



## USER WANTS TO MAKE TRANSACTION IN THE EXISTING ACCOUNT



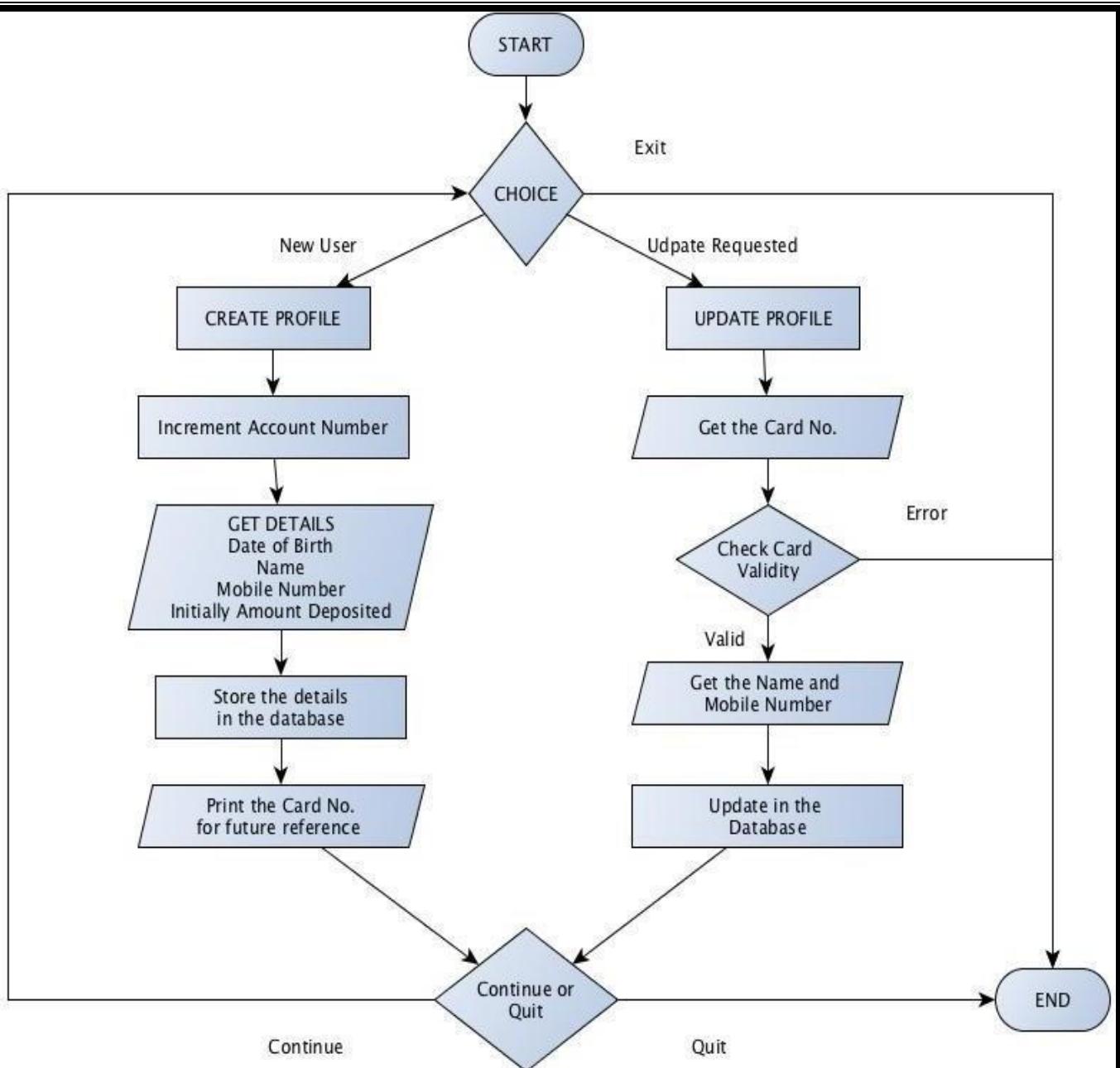
## **VII MODULE COMPONENTS:**

In this Project, we will be using two modules : • Admin • ATM (user)

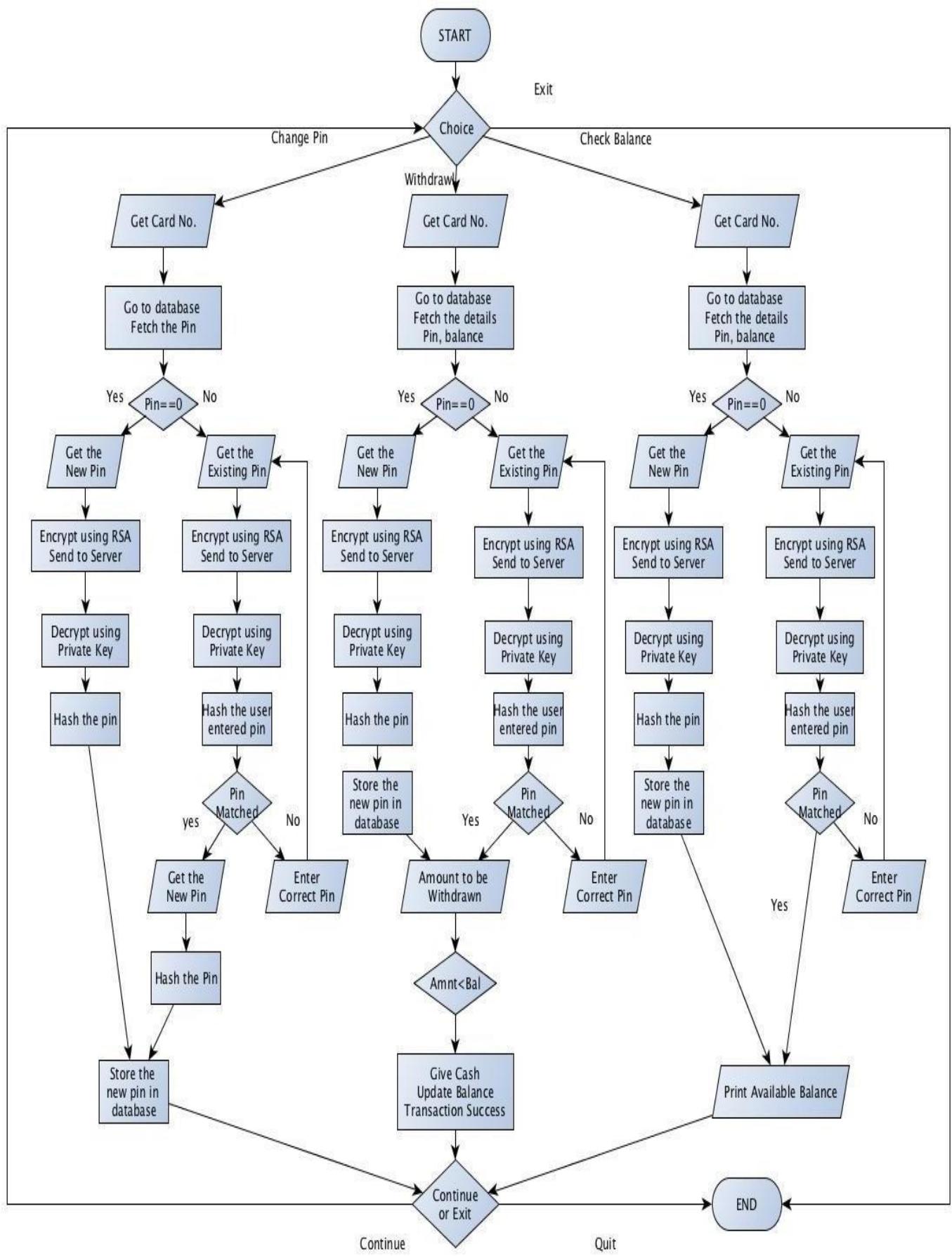
- Admin

If there comes a new user, he needs to create a new account/profile where the account number will be given serial wise and the user needs to enter his/her personal details. The given details by user are stored in encrypted form in database. After creating new account, the new user is provided with the card number for using ATM service.

After creating and giving his/her details to the server, if he/she needs to update his/her details, he/she should enter card number to login and to check validity. He/she enter the updated details and the same details is updated in the database.



- ATM (user)



## ATM Services:

- Update PIN
- Withdrawal and update balance
- Print Available balance

## CODE :-

### ATM.py

The screenshot shows the Spyder Python IDE interface. On the left, the code editor displays `ATM.py` with various functions like `check`, `gcd`, and `encrypt`. On the right, the IPython console shows the execution of the script and user interactions for profile management.

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/finalAdmin.py', wdir='C:/Users/harsh/Downloads')
        welcome

1. Create Profile
2. Update Profile
3. Exit
Enter the choice: 2

Enter the Card No.: 532009
Enter the Details

Enter the Name: ffsfsdf

Enter the phone number: 9987654321
Updated Successfully

Do you want to continue:
1. Yes 2. No
Enter the Choice: 2
Good Bye

In [2]:
```

At the bottom, the status bar shows the environment is `conda: base (Python 3.7.6)`, with other system details like date and time.

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\harsh\Downloads\finalATM.py

finalATM.py

```
37     def encrypt(msg):
38         c=(msg**e)%n
39         return c
40
41     def decrypt(c):
42         y=(c**d)%n
43         return y
44
45     def update(cardNo,pin):
46         conn=sqlite3.connect("AccountInfo.db")
47         cmd="select * from details where Card_No='"+str(cardNo)+"'"
48         cursor=conn.execute(cmd)
49         ifRecordExist=0
50         for row in cursor:
51             ifRecordExist+=1
52         if(ifRecordExist==1):
53             cmd="update details SET Pin='"+str(pin)+"' where Card_No='"+str(cardNo)+"'"
54             conn.execute(cmd)
55             conn.commit()
56             conn.close()
57
58     def updateB(cardNo,balance):
59         conn=sqlite3.connect("AccountInfo.db")
60         cmd="select * from details where Card_No='"+str(cardNo)+"'"
61         cursor=conn.execute(cmd)
62         ifRecordExist=0
63         for row in cursor:
64             ifRecordExist+=1
65         if(ifRecordExist==1):
66             cmd="update details SET Balance='"+str(balance)+"' where Card_No='"+str(cardNo)+"'"
67             conn.execute(cmd)
68             conn.commit()
69             conn.close()
70
71     def server(cardNo,pin_user):
72         pin_user=decrypt(pin_user)
73         profileCheck(cardNo)
74         if(profile!=None):
75             pin=profile[5]
76             Pin=int(pin)
77             balance=int(profile[4])
78             if(Pin==0):
79                 pin=str(input('\nEnter a new pin: '))
80             if(len(pin)!=4):
81                 print("\nPin must have 4 digits. Pls Re-enter: ")
82                 pin=str(input('Enter the Pin: '))
83                 print('Pin Created Successfully')
84                 pin=hash(str(pin))
85                 pin=hash(str(pin))
86                 update(cardNo,pin)
87             else:
88                 pin_user=str(pin_user)
89                 pin_user=hash(pin_user)
90                 pin_user=int(pin_user)
91             if(Pin==pin_user):
92                 if(balance>=Amt):
93                     balance=balance-Amt
94                     print ('\nTRANSACTION SUCCESS')
95                     print ('Available Balance is ',balance)
96                     updateB(cardNo,balance)
97                 else:
98                     print('\nYOU DONT HAVE SUFFICIENT BALANCE')
99             else:
100                 print('**Pin entered is INCORRECT**')
101         else:
102             print ('\n**ENTERED CARD NUMBER IS INVALID**')
103
104
105     def server2(cardNo,pin_user):
106         pin_user=decrypt(pin_user)
107         profileCheck(cardNo)
108         if(profile!=None):
109             pin=profile[5]
110             balance=profile[4]
111             if(pin == 0):
```

Console I/A

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
        Welcome
```

In [2]:

1. Create Profile
2. Update Profile
3. Exit

Enter the choice: 2

Enter the Card No.: 532009

Enter the Details

Enter the Name: ffsfsdf

Enter the phone number: 9987654321

Updated Successfully

Do you want to continue:

1. Yes 2. No

Enter the Choice: 2

Good Bye

In [2]:

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\harsh\Downloads\finalATM.py

finalATM.py

```
71     def server(cardNo,pin_user):
72         pin_user=decrypt(pin_user)
73         profileCheck(cardNo)
74         if(profile!=None):
75             pin=profile[5]
76             Pin=int(pin)
77             balance=int(profile[4])
78             if(Pin==0):
79                 pin=str(input('\nEnter a new pin: '))
80             if(len(pin)!=4):
81                 print("\nPin must have 4 digits. Pls Re-enter: ")
82                 pin=str(input('Enter the Pin: '))
83                 print('Pin Created Successfully')
84                 pin=hash(str(pin))
85                 pin=hash(str(pin))
86                 update(cardNo,pin)
87             else:
88                 pin_user=str(pin_user)
89                 pin_user=hash(pin_user)
90                 pin_user=int(pin_user)
91             if(Pin==pin_user):
92                 if(balance>=Amt):
93                     balance=balance-Amt
94                     print ('\nTRANSACTION SUCCESS')
95                     print ('Available Balance is ',balance)
96                     updateB(cardNo,balance)
97                 else:
98                     print('\nYOU DONT HAVE SUFFICIENT BALANCE')
99             else:
100                 print('**Pin entered is INCORRECT**')
101         else:
102             print ('\n**ENTERED CARD NUMBER IS INVALID**')
103
104
105     def server2(cardNo,pin_user):
106         pin_user=decrypt(pin_user)
107         profileCheck(cardNo)
108         if(profile!=None):
109             pin=profile[5]
110             balance=profile[4]
111             if(pin == 0):
```

Console I/A

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
        Welcome
```

In [2]:

1. Create Profile
2. Update Profile
3. Exit

Enter the choice: 2

Enter the Card No.: 532009

Enter the Details

Enter the Name: ffsfsdf

Enter the phone number: 9987654321

Updated Successfully

Do you want to continue:

1. Yes 2. No

Enter the Choice: 2

Good Bye

In [2]:

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\harsh\Downloads\finalATM.py

finalATM.py

```
105     def server2(cardNo,pin_user):
106         pin_user=decrypt(pin_user)
107         profile_check(cardNo)
108         if(profile!=None):
109             pin=profile[5]
110             balance=profile[4]
111             if(pin == 0):
112                 pin=str(input('\nEnter a new pin: '))
113                 if(len(pin)!=4):
114                     print("\nPin must have 4 digits. Pls Re-enter: ")
115                     pin=str(input('Enter the Pin: '))
116                     print('Pin Created Successfully')
117                     pinhash(str(pin))
118                     update(cardNo,pin)
119             else:
120                 pin_user=str(pin_user)
121                 pin_user=hash(pin_user)
122                 if(pin==pin_user):
123                     print('Available Balance : ',balance)
124                 else:
125                     print("\n**Pin entered is INCORRECT**")
126             if(profile==None):
127                 print('**ENTERED CARD NUMBER IS INVALID**')
128
129     def server3(cardNo,pin_user):
130         pin_user=decrypt(pin_user)
131         profile_check(cardNo)
132         if(profile!=None):
133             pin=profile[5]
134             if(pin==0):
135                 pin=str(input('Enter a new pin: '))
136                 if(len(pin)!=4):
137                     print("\nPin must have 4 digits. Pls Re-enter: ")
138                     pin=str(input('Enter the Pin: '))
139                     print('Pin Created Successfully')
140                     pinhash(str(pin))
141                     update(cardNo,pin)
142             else:
143                 pin_user=hash(str(pin_user))
144                 if(pin==pin_user):
145                     newpin=str(input('Enter the new Pin: '))
146                     if(len(newpin)==4):
147                         print("\nPin must have 4 digits. Pls Re-enter: ")
148                         newpin=str(input('Enter the Pin: '))
149                         print('Pin Changed Successfully')
150                         pin=hash(newpin)
151                         update(cardNo,pin)
152             else:
153                 print("\n**Pin entered is incorrect**")
154         if(profile==None):
155             print('**ENTERED CARD NUMBER IS INVALID**')
156
157     print ('-----Welcome-----')
158     print('Please choose among the following options:')
159     a=int(input('\n1. Withdrawl \n2. Check Balance \n3. Create or Change Pin \n4. Exit\n'))
160     while(a!=4):
161         if(a==1):
162             cardNo=str(input('Enter the Card No.: '))
163             pin_user=decrypt(str(pin_user))
164             pin_user=int(input('Enter the pin: '))
165             pin_user=hash(pin_user)
166             Amt=int(input("\nEnter amount to be removed: "))
167             server(cardNo,pin_user)
168
169         if(a==2):
```

Console I/A

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
          Welcome

1. Create Profile
2. Update Profile
3. Exit
Enter the choice: 2

Enter the Card No.: 532009
Enter the Details

Enter the Name: ffsfsdf
Enter the phone number: 9987654321
Updated Successfully

Do you want to continue:
1. Yes 2. No
Enter the Choice: 2
Good Bye

In [2]:
```

conda: base (Python 3.7.6) Line 137, Col 67 ASCII LF RW Mem 84%

12:43 27-09-2020

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\harsh\Downloads\finalATM.py

finalATM.py

```
129     def server3(cardNo,pin_user):
130         pin_user=decrypt(pin_user)
131         profile_check(cardNo)
132         if(profile!=None):
133             pin=profile[5]
134             if(pin==0):
135                 pin=str(input('Enter a new pin: '))
136                 if(len(pin)!=4):
137                     print("\nPin must have 4 digits. Pls Re-enter: ")
138                     pin=str(input('Enter the Pin: '))
139                     print('Pin Created Successfully')
140                     pinhash(str(pin))
141                     update(cardNo,pin)
142             else:
143                 pin_user=hash(str(pin_user))
144                 if(pin==pin_user):
145                     newpin=str(input('Enter the new Pin: '))
146                     if(len(newpin)==4):
147                         print("\nPin must have 4 digits. Pls Re-enter: ")
148                         newpin=str(input('Enter the Pin: '))
149                         print('Pin changed Successfully')
150                         pin=hash(newpin)
151                         update(cardNo,pin)
152             else:
153                 print("\n**Pin entered is incorrect**")
154         if(profile==None):
155             print('**ENTERED CARD NUMBER IS INVALID**')
156
157     print ('-----Welcome-----')
158     print('Please choose among the following options:')
159     a=int(input('\n1. Withdrawl \n2. Check Balance \n3. Create or Change Pin \n4. Exit\n'))
160     while(a!=4):
161         if(a==1):
162             cardNo=str(input('Enter the Card No.: '))
163             pin_user=decrypt(str(pin_user))
164             pin_user=int(input('Enter the pin: '))
165             pin_user=hash(pin_user)
166             Amt=int(input("\nEnter amount to be removed: "))
167             server(cardNo,pin_user)
168
169         if(a==2):
```

Console I/A

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
          Welcome

1. Create Profile
2. Update Profile
3. Exit
Enter the choice: 2

Enter the Card No.: 532009
Enter the Details

Enter the Name: ffsfsdf
Enter the phone number: 9987654321
Updated Successfully

Do you want to continue:
1. Yes 2. No
Enter the Choice: 2
Good Bye

In [2]:
```

conda: base (Python 3.7.6) Line 137, Col 67 ASCII LF RW Mem 83%

12:44 27-09-2020

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\harsh\Downloads\finalATM.py

finalAdmin.py

```
150     pin=hash(newpin)
151     update(cardNo,pin)
152 else:
153     print ('\\n**Pin entered is incorrect**')
154 if(profile==None):
155     print ('\\n**ENTERED CARD NUMBER IS INVALID**')
156
157
158 print ('\\n-----Welcome-----\\n')
159 print('Please choose among the following options:')
160 a=int(input('\\n1. Withdrawl \\n2. Check Balance \\n3. Create or Change Pin \\n4. Exit\\nE'))
161 while(a!=4):
162     if(a==1):
163         cardNo=str(input('Enter the Card No.: '))
164         pin_user=int(input('Enter the pin: '))
165         pin_user=int(encrypt(pin_user))
166         Amt=int(input('\\nEnter amount to be removed: '))
167         server(cardNo,pin_user)
168
169     if(a==2):
170         cardNo=str(input('Enter the Card No.: '))
171         pin_user=int(input('Enter the pin: '))
172         pin_user=int(encrypt(pin_user))
173         server2(cardNo,pin_user)
174
175     if(a==3):
176         cardNo=str(input('Enter the Card No.: '))
177         pin_user=int(input('Enter the pin: '))
178         pin_user=int(encrypt(pin_user))
179         server3(cardNo,pin_user)
180
181 z=int(input('\\nDo you want to continue:\\n1. Yes\\n2.No \\nEnter your Choice: '))
182 if(z==1):
183     a=int(input('\\n1. Withdrawl \\n2. Check Balance \\n3. Create or Change Pin \\n4.'))
184 if(z==2):
185     break
186 if(a==4 or z==2):
187     print ("Good Bye")
188
189
```

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')

Welcome

1. Create Profile  
2. Update Profile  
3. Exit  
Enter the choice: 2

Enter the Card No.: 532009  
Enter the Details  
Enter the Name: ffsdfsf  
Enter the phone number: 9987654321  
Updated Successfully

Do you want to continue:  
1. Yes 2. No  
Enter the Choice: 2  
Good Bye

In [2]:

conda:base CodeVIT 7.6) Line 137, Col 67 ASCII LF RW Mem 84%

12:44 27-09-2020

## ADMIN.py

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\harsh\Downloads\finalAdmin.py

finalTM.py

```
1 import sqlite3
2 import uuid
3
4 class birth:
5     date = int
6     month = str
7     year = int
8
9 def update(Id):
10    conn=sqlite3.connect("AccountInfo.db")
11    cmd="select * from details where Card_No='"+str(Card)+"'
12    cursor=conn.execute(cmd)
13    ifRecordExist=0
14    for row in cursor:
15        ifRecordExist=1
16    if(iRecordExist==1):
17        print('Enter the Details')
18        name=input("Enter the Name: ")
19        cmd="update Details set Name = '"+str(name)+"' where Card_No= '"+str(Card)+"'"
20        conn.execute(cmd)
21        phone=int(input("Enter the phone number: "))
22        cmd="update Details set Phone_No = '"+str(phone)+"' where Card_No='"+str(Card)+"'
23        conn.execute(cmd)
24        conn.commit()
25    else:
26        print('Enter the Correct Details')
27        conn.close()
28
29 def insert(accNo):
30    conn=sqlite3.connect("AccountInfo.db")
31    cmd="select * from details where Account_No='"+str(accNo)+"'
32    cursor=conn.execute(cmd)
33    cmd="insert into details (Account_No,Name,Card_No,Phone_No,Balance,Pin) values('"+str(
34    conn.execute(cmd)
35    conn.commit()
36    conn.close()
37
38 cursor=0
39
40 def count(cursor):
41    conn=sqlite3.connect("AccountInfo.db")
```

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')

Welcome

1. Create Profile  
2. Update Profile  
3. Exit  
Enter the choice: 2

Enter the Card No.: 532009  
Enter the Details  
Enter the Name: ffsdfsf  
Enter the phone number: 9987654321  
Updated Successfully

Do you want to continue:  
1. Yes 2. No  
Enter the Choice: 2  
Good Bye

In [2]:

conda: base (Python 3.7.6) Line 20, Col 20 ASCII LF RW Mem 84%

12:44 27-09-2020

Spyder (Python 3.7)

```
File Edit Search Source Run Debug Consoles Projects Tools View Help
```

C:\Users\harsh\Downloads\finalAdmin.py

finalATM.py finalAdmin.py

```
40     def countcursor():
41         conn=sqlite3.connect("AccountInfo.db")
42         cmd="select count(*) from details"
43         cursor=conn.execute(cmd)
44         results=cursor.fetchone()[0]
45         conn.commit()
46         conn.close()
47         return results
48
49     def check(ck):
50         conn=sqlite3.connect("AccountInfo.db")
51         cmd="select * from details where Card_No='"+str(ck)+"'
52         cursor=conn.execute(cmd)
53         profile=None
54         for row in cursor:
55             profile=row
56         conn.close()
57         return profile
58
59 cardNo=232000
60 print ('____Welcome_____\n')
61 a=int(input('\n1. Create Profile \n2. Update Profile \n3. Exit\nEnter the choice: '))
62 while(a!=3):
63     if(a==1):
64         i=count(cursor)
65         accNo=i+1
66         v=accNo
67         accNo=birth()
68         print ('\nEnter the following Details\n')
69         accNo_date=int(input('Enter the Day of birth: '))
70         accNo_month=str(input('Enter the Month of birth: '))
71         accNo_year=int(input('Enter the Year of birth: '))
72         name=input("Enter the Name: ")
73         phone=int(input("Enter the phone number: "))
74         bal=int(input('Enter the amount deposited: '))
75         y=accNo.date
76         x=accNo.month
77         j=accNo.year
78         pin=0
79         cardNo+=v
80         print ('Card No:',cardNo)
81         insert(v)
82         print ('Created')
83     if(a==2):
84         Card=int(input('Enter the Card No.: '))
85         profile=update(Card)
86         print("Updated Successfully")
87         z=int(input('\nDo you want to continue:\n1. Yes\n2. No\nEnter the Choice: '))
88     if(z==1):
89         a=int(input('\n1. Create Profile \n2. Update Profile \n3. Exit\n\nEnter the choice: '))
90     if(z==2):
91         break
92 if(a==3 or z==2):
93     print ("Good Bye")
```

Console I/A

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
          Welcome_____

1. Create Profile
2. Update Profile
3. Exit
Enter the choice: 2

Enter the Card No.: 532009
Enter the Details

Enter the Name: ffsfsdf
Enter the phone number: 9987654321
Updated Successfully

Do you want to continue:
1. Yes 2. No
Enter the Choice: 2
Good Bye
```

In [2]:

Spyder (Python 3.7)

```
File Edit Search Source Run Debug Consoles Projects Tools View Help
```

C:\Users\harsh\Downloads\finalAdmin.py

finalATM.py finalAdmin.py

```
57     return profile
58
59 cardNo=232000
60 print ('____Welcome_____\n')
61 a=int(input('\n1. Create Profile \n2. Update Profile \n3. Exit\nEnter the choice: '))
62 while(a!=3):
63     if(a==1):
64         i=count(cursor)
65         accNo=i+1
66         v=accNo
67         accNo=birth()
68         print ('\nEnter the following Details\n')
69         accNo_date=int(input('Enter the Day of birth: '))
70         accNo_month=str(input('Enter the Month of birth: '))
71         accNo_year=int(input('Enter the Year of birth: '))
72         name=input("Enter the Name: ")
73         phone=int(input("Enter the phone number: "))
74         bal=int(input('Enter the amount deposited: '))
75         y=accNo.date
76         x=accNo.month
77         j=accNo.year
78         pin=0
79         cardNo+=v
80         print ('Card No:',cardNo)
81         insert(v)
82         print ('Created')
83     if(a==2):
84         Card=int(input('Enter the Card No.: '))
85         profile=update(Card)
86         print("Updated Successfully")
87         z=int(input('\nDo you want to continue:\n1. Yes\n2. No\nEnter the Choice: '))
88     if(z==1):
89         a=int(input('\n1. Create Profile \n2. Update Profile \n3. Exit\n\nEnter the choice: '))
90     if(z==2):
91         break
92 if(a==3 or z==2):
93     print ("Good Bye")
```

Console I/A

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
          Welcome_____

1. Create Profile
2. Update Profile
3. Exit
Enter the choice: 2

Enter the Card No.: 532009
Enter the Details

Enter the Name: ffsfsdf
Enter the phone number: 9987654321
Updated Successfully

Do you want to continue:
1. Yes 2. No
Enter the Choice: 2
Good Bye
```

In [2]:

# OUTPUT

## ADMIN.py

The screenshot shows the Spyder Python 3.7 environment. The left pane displays the code for `finalAdmin.py`, which includes functions for inserting, updating, and checking records in a SQLite database named `AccountInfo.db`. The right pane shows the IPython 7.12.0 console output. A user runs the script and chooses option 2 ('Update Profile'). They are prompted to enter details for a card, including name, phone number, and balance. After entering 'ffsdfs' as the name and '9987654321' as the phone number, they are asked if they want to continue. The user types '1. Yes' and '2. No', then '1'. The console also shows the amount deposited as 4534535 and the card number as 532089.

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
Welcome

1. Create Profile
2. Update Profile
3. Exit
Enter the choice: 1

Enter the following Details

Enter the Day of birth: 87
Enter the Month of birth: gdfgdg
Enter the Year of birth: 8776
Enter the Name: ffsdfs
Enter the phone number: 9987654321
Enter the amount deposited: 4534535
Card No: 532089
Created

Do you want to continue:
1. Yes 2. No
Enter the Choice: 1
```

The screenshot shows the Spyder Python 3.7 environment again. This time, the code in `finalTM.py` is executed. The user chooses option 2 ('Update Profile') and is prompted to update a record in the `details` table. They enter the card number '532089' and the new phone number '9987654321'. The console output shows the updated record with the new phone number.

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
Welcome

1. Create Profile
2. Update Profile
3. Exit
Enter the choice: 1

Enter the following Details

Enter the Day of birth: 87
Enter the Month of birth: gdfgdg
Enter the Year of birth: 8776
Enter the Name: ffsdfs
Enter the phone number: 9987654321
Enter the amount deposited: 4534535
Card No: 532089
Created

Do you want to continue:
1. Yes 2. No
Enter the Choice: 1
```

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\harsh\Downloads\finalAdmin.py

finalATM.py finalAdmin.py

Console I/A

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
          Welcome
```

1. Create Profile  
2. Update Profile  
3. Exit  
Enter the choice: 2

Enter the Card No.: 532009  
Enter the Details

Enter the Name: ffsfdf  
Enter the phone number: 9987654321  
Updated Successfully

Do you want to continue:  
1. Yes 2. No  
Enter the Choice: 2  
Good Bye

In [2]: |

conda: base (Python 3.7.6) Line 52, Col 26 ASCII LF RW Mem 87% 12:39 27-09-2020

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\harsh\Downloads\finalAdmin.py

finalATM.py finalAdmin.py

Console I/A

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/harsh/Downloads/FinalAdmin.py', wdir='C:/Users/harsh/Downloads')
          Welcome
```

1. Create Profile  
2. Update Profile  
3. Exit  
Enter the choice: 2

Enter the Card No.: 532009  
Enter the Details

Enter the Name: ffsfdf  
Enter the phone number: 9987654321  
Updated Successfully

Do you want to continue:  
1. Yes 2. No  
Enter the Choice: 2  
Good Bye

In [2]: |

conda: base (Python 3.7.6) Line 52, Col 26 ASCII LF RW Mem 87% 12:38 27-09-2020

# ATM.py

```
Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:/Users/harsh/Downloads/finalATM.py
finalATM.py finalAdmin.py
1 import sqlite3
2 import math
3
4 def check(ck):
5     conn=sqlite3.connect("AccountInfo.db")
6     cmd="select * from details where Card_No='"+str(ck)
7     cursor=conn.execute(cmd)
8     profile=None
9     for row in cursor:
10         profile=row
11     conn.close()
12     return profile
13
14 def gcd(a, b):
15     while(b > 0):
16         c = a % b
17         a = b
18         b = c
19     return a
20
21 p=97
22 q=107
23 n=p*q
24 phi=(p-1)*(q-1)
25 for e in range(2,phi):
26     if gcd(e,phi)== 1:
27         break
28 f=2*e+1
29 g=n-1
30
31 for i in range(1,10):
32     x = 1 + 1^phi
33     if x % e == 0:
34         d = int(x/e)
35         break
36
37 def encrypt(msg):
38     c=(msg**e)%n
39     return c
40
41 def decrypt(c):
```

1. Withdrawal  
2. Check Balance  
3. Create or Change Pin  
4. Exit

Enter your choice: 1

Enter the Card No.: 532009

Enter the pin: 1234

Enter amt to be removed: 12

TRANSACTION SUCCESS

Available Balance is 4534523

Do you want to continue:  
1. Yes 2.No

Enter your Choice: 1

1. Withdrawal  
2. Check Balance  
3. Create or Change Pin  
4. Exit

Enter your choice: 2

Enter the Card No.: 532009

Enter the pin: 1234

Available Balance : 4534523

Do you want to continue:  
1. Yes 2.No

Enter your Choice: 2

Good Bye

In [3]:

conda: base (Python 3.7.6) Line 188, Col 1 ASCII LF RW Mem 85%

12:36 27-09-2020

```
Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:/Users/harsh/Downloads/finalATM.py
finalATM.py finalAdmin.py
1 import sqlite3
2 import math
3
4 def check(ck):
5     conn=sqlite3.connect("AccountInfo.db")
6     cmd="select * from details where Card_No='"+str(ck)
7     cursor=conn.execute(cmd)
8     profile=None
9     for row in cursor:
10         profile=row
11     conn.close()
12     return profile
13
14 def gcd(a, b):
15     while(b > 0):
16         c = a % b
17         a = b
18         b = c
19     return a
20
21 p=97
22 q=107
23 n=p*q
24 phi=(p-1)*(q-1)
25 for e in range(2,phi):
26     if gcd(e,phi)== 1:
27         break
28 f=2*e+1
29 g=n-1
30
31 for i in range(1,10):
32     x = 1 + 1^phi
33     if x % e == 0:
34         d = int(x/e)
35         break
36
37 def encrypt(msg):
38     c=(msg**e)%n
39     return c
40
41 def decrypt(c):
```

1. Yes 2.No

Enter your Choice: 1

1. Withdrawal  
2. Check Balance  
3. Create or Change Pin  
4. Exit

Enter your choice: 1

Enter the Card No.: 532009

Enter the pin: 1234

Enter amt to be removed: 12

TRANSACTION SUCCESS

Available Balance is 4534523

Do you want to continue:  
1. Yes 2.No

Enter your Choice: 1

1. Withdrawal  
2. Check Balance  
3. Create or Change Pin  
4. Exit

Enter your choice: 2

Enter the Card No.: 532009

Enter the pin: 1234

Available Balance : 4534523

Do you want to continue:  
1. Yes 2.No

Enter your Choice: 1

conda: base (Python 3.7.6) Line 188, Col 1 ASCII LF RW Mem 84%

12:35 27-09-2020

Spyder (Python 3.7)

```
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\harsh\Downloads\finalATM.py
finalATM.py finalAdmin.py
1 import sqlite3
2 import math
3
4 def check(ck):
5     conn=sqlite3.connect("AccountInfo.db")
6     cmd="select * from details where Card_No='"+str(ck)
7     cursor=conn.execute(cmd)
8     profile=None
9     for row in cursor:
10         profile=row
11     conn.close()
12     return profile
13
14 def gcd(a, b):
15     while(b > 0):
16         c = a % b
17         a = b
18         b = c
19     return a
20
21 p=97
22 q=107
23 n=p*q
24 phi=(p-1)*(q-1)
25 for e in range(2,phi):
26     if gcd(e,phi)== 1:
27         break
28     f=e+1
29     g=e-1
30
31 for i in range(1,10):
32     x = 1 + i*phi
33     if x % e == 0:
34         d = int(x/e)
35         break
36
37 def encrypt(msg):
38     c=(msg**e)%n
39     return c
40
41 def decrypt(c):
```

Console I/A

1. Withdrawal
2. Check Balance
3. Create or Change Pin
4. Exit

Enter your choice: 3

Enter the Card No.: 532009

Enter the pin: 0

Enter the new Pin: 1234

\*\*Pin changed Successfully\*\*

Do you want to continue:

1. Yes 2.No

Enter your Choice: 1

1. Withdrawal

2. Check Balance

3. Create or Change Pin

4. Exit

Enter your choice: 1

Enter the Card No.: 532009

Enter the pin: 1234

Enter amt to be removed: 12

TRANSACTION SUCCESS

Available Balance is 4534523

Do you want to continue:

1. Yes 2.No

Enter your Choice:

conds: base (Python 3.7.6) Line 188, Col 1 ASCII LF RW Mem 84%

12:35 27-09-2020

Spyder (Python 3.7)

```
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\harsh\Downloads\finalATM.py
finalATM.py finalAdmin.py
1 import sqlite3
2 import math
3
4 def check(ck):
5     conn=sqlite3.connect("AccountInfo.db")
6     cmd="select * from details where Card_No='"+str(ck)
7     cursor=conn.execute(cmd)
8     profile=None
9     for row in cursor:
10         profile=row
11     conn.close()
12     return profile
13
14 def gcd(a, b):
15     while(b > 0):
16         c = a % b
17         a = b
18         b = c
19     return a
20
21 p=97
22 q=107
23 n=p*q
24 phi=(p-1)*(q-1)
25 for e in range(2,phi):
26     if gcd(e,phi)== 1:
27         break
28     f=e+1
29     g=e-1
30
31 for i in range(1,10):
32     x = 1 + i*phi
33     if x % e == 0:
34         d = int(x/e)
35         break
36
37 def encrypt(msg):
38     c=(msg**e)%n
39     return c
40
41 def decrypt(c):
```

Console I/A

2. Check Balance
3. Create or Change Pin
4. Exit

Enter your choice: 3

Enter the Card No.: 532009

Enter the pin: 0

Enter a new pin: 123456

Pin must have 4 digits. Pls Re-enter:

Enter the Pin: 0

Pin Created Successfully

Do you want to continue:

1. Yes 2.No

Enter your Choice: 1

1. Withdrawal

2. Check Balance

3. Create or Change Pin

4. Exit

Enter your choice: 3

Enter the Card No.: 532009

Enter the pin: 0

Enter the new Pin: 1234

\*\*Pin changed Successfully\*\*

Do you want to continue:

1. Yes 2.No

Enter your Choice:

conds: base (Python 3.7.6) Line 188, Col 1 ASCII LF RW Mem 84%

12:35 27-09-2020

	Account	Name	Card No.	Phone No.	Balance	Pin
1	1	Archit Goyal	632001	8708170952	11000	4814036714628610464
2	2	Sahil	632002	9478915417	20000	65947338413136959
3	3	Ayush	632003	9865432100	17000	481403771460023416
4	4	Saranya	632004	6379455237	20000	45377514337852329301
5	5	Neha	632009	9195273701	77140	4036789986166528717
6	6	ritika	632006	6379455237	23000	-579529862229579058
7	7	Damodar sharma	632007	6767565657	2340000	-
8	8	"Harshit Goel"	632008	9654683467	90000	2456734126405025421
9	9	ffsd	532009	9987654321	4534523	-2614848780589184395

**Database updated after the user has set a PIN which is stored in the form of a hash value.**

## CONCLUSION:

We have made Secured Data Encryption System using Python language and it is storing details in encrypted form of user ensuring security to user. The RSA technique used helps to prevent the third party from knowing the keys of the user. The Hash function prevents the Man in Middle attack. It can implemented in real world for personal uses.

## REFERENCES

- [1] Jyotiranjan Hota, "Automated Teller Machines in India", Proceedings of GLOGIFT 13, Dec-2013
- [2] Oliveira, T. and Martins, M, F. (2011) Literature Review of Information Technology Adoption Models at Firm Level, The Electronic Journal Information Systems Evaluation, 14(1): 110-121.
- [3] Hota, J.R. (2013) Growth of ATM Industry in India, CSI Communications, 36(11): 23-25.

- [4] Agarwal, R. and Prasad, J. (1998) A Conceptual and Operational Definition of Personal Innovativeness in the Domain of Information Technology, *Information Systems Research*, 9(2): 204-215.
- [5] Identifiers for Digital Identity Management.
- [6] Laplante, P. A. (1977). Real-time systems design and analysis (2nd ed.). Washington, DC: IEEE Press.
- [7] Devinaga, R. (2010). ATM risk management and controls. *European journal of economic, finance and administrative sciences*. ISSN 1450-2275 issue 21.
- [8] Heather Crawford (2011). Applying Usable Security Principles to Authentication.

\*\*\*\*\*