PROJECT REPORT
on
<u>Friends Info Book</u>

Submitted by

Name of Student: Aksh Saxena
University Roll No: 201500066

Name of Student: Anamika Varshney
University Roll No: 201500085

Name of Student: Anikate Agrawal
University Roll No: 201500088

Department of Computer Engineering & Applications

Institute of Engineering & Technology
GLA University
Mathura- 281406, INDIA

Department of Computer Engineering and Applications

GLA University, Mathura

17 km. Stone NH#2, Mathura-Delhi Road, P.O. –

Declaration

I hereby declare that the work which is being presented in the Project Report **"Title-Friends Info Book",** in partial fulfillment of the requirements for Project is an authentic record of my own work carried under the supervision of Mr. Vinay Agrawal**, Assistant Professor, GLA University, Mathura**.

Sign

Name of Candidate: Anikate Agrawal

University Roll No.: 201500088

Sign

Name of Candidate: Aksh Saxena

University Roll No.: 201500066

Sign

Name of Candidate: Anamika Varshney

University Roll No.: 201500085

Certificate

This is to certify that the above statements made by the candidate are correct to the best of my knowledge and belief.

_____

Project Supervisor

Mr. Vinay Agrawal

Designation of Supervisor: Assistant Professor

Date: 11/12/2021

Table of Contents

Introduction

Overview

This is a program to manage the information about friends and our known ones. This program uses concepts of dictionary, lists and file handling of python. This mini project has several modules for performing different tasks. This is a password protected system. For the very first-time user can create a password and for subsequent login he needs to enter correct password. If the entered password is correct user can perform several different functions related to information of his known ones. This ensures his privacy. The program is very user friendly and guides the user at every step. It is a menu driven program therefore user can easily interact by entering the desired choice option.Using this program, we can easily add and store the information of our friends. We can also check, update and modify the data at any time very easily. If there is no further need of any friend's information, we can even delete his record.The program also has feature to allow the user to erase entire record and start new record. The program also allows the user to decide whether he wants to continue or to exit. It does its specified tasks well and help users to keep track of all information regarding their friends and family.

Objectives

The objective of this project is to provide users various facilities:

1. To setup a password for login.
2. To add a friend and save his various details such as:
   a. Name
   b. Email
   c. Address
   d. Phone number
   e. Birth Date
3. To search a friend and display his details.
4. To display list of friends.
5. To update details of friend.
6. To delete a friend.
7. To clear all records.
8. All data must be stored on a file so that stored data is not lost.

The program must ease the management of information. Allow easy and fast access to the required data. Prevent any unknown person from checking our data.

Project Design

Problem Statement

We need to design a program that is password protected and helps the users to store the information of friends. The user will enter a password. Program must check if it is valid or not. If password is not valid, it must allow user to retry. If Password is correct, the login must be successful and user must be allowed to perform further tasks. The program should have options to

i)   add a friend and enter his various details such as name, email, number, address and birthdate.

ii)   show all the friends, user has stored previously.

iii) display all the details of friend searched.

iv)   allow modifications and updation to stored details of any friend.

v)   delete record of any friend.

vi)   delete all the friends.

vii) update password and set a new one.

Also, there must be relevant options and messages to guide the user to perform all the above-mentioned operations.

Modular Design

In this program, we have used function-oriented design. We have partitioned the problem into manageable number of modules. Each module performs one of the requirements of the problem statement and is independent of other modules. Several modules required for problem solving include:

1. addFriend()
2. deleteFriend()
3. updateFriendDetails()
4. showFriends()
5. login()
6. friendDetails()
7. updatePassword()
8. fetch_friends()
9. modify_file()
10. clearFriendsBook()

All these modules are integrated together for the desired operations to be performed by the user.
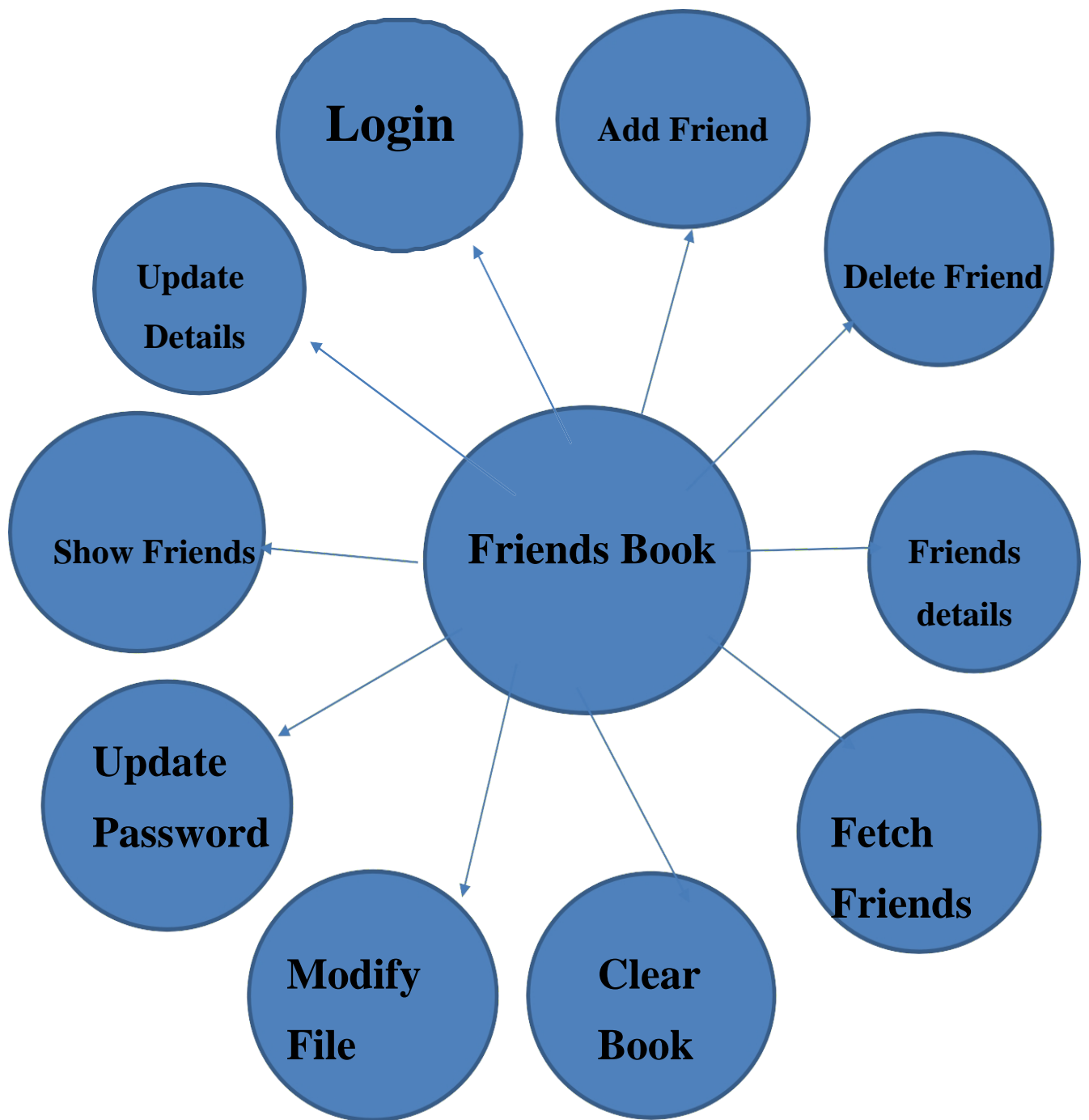
Project Implementation

Python Source Code

```python
def login():

    if 'password' not in d:

        password=input("Create new password\n")

        d['password']=password

        print("password created")

        return False

    else:

        password=input("Enter password\n")

        if d['password']==password:

            print("login successful")

            return True

        else:

            print("incorrect password")

            return False


def updatePassword():

    d['password']=input("Enter new password\n")


def addFriend():

    name=input("Enter name\n")

    if name in d:

        print("Name already exists choose another")

    else:

        l=[]

        print("Press enter for blank entry")
```

```python
        l.append(input("Enter number\n"))

        l.append(input("Enter email\n"))

        l.append(input("Enter address\n"))

        l.append(input("Enter birthday\n"))

        d[name]=l

def showFriends():

    for i in d:

        if i != 'password':

            print(i)

def friendDetails():

    name=input("Enter name to search\n")

    if name in d:

        print("Name :",name)

        print("Contact Number :",d[name][0])

        print("Email :",d[name][1])

        print("Address :",d[name][2])

        print("BirthDay :",d[name][3])

    else:print("Friend Not Found")

def fetch_friends():

    f=open("FriendsBook.txt","a")

    f.close()

    f=open("FriendsBook.txt","r")

    l=f.readlines()

    for i in l:

        a=i.split()

        n=a[0]

        a=a[1:]

        a=' '.join(a)
```

```python
        if n=='password':
            d[n]=a
        else:
            d[n]=eval(a)
    f.close()
def updateFriendDetails():
    name=input("Enter name to update\n")
    if name not in d:
        print("Friend name not found")
    else:
        c=int(input("1.Update number\n2.Update Email\n3.Update address\n4.Update BirthDay\n"))
        if c=='1':
            d[name][0]=input("Enter new number\n")
            print("number Updated")
        elif c=='2':
            d[name][1]=input("Enter new email\n")
            print("email Updated")
        elif c=='3':
            d[name][2]=input("Enter new address\n")
            print("address Updated")
        elif c=='4':
            d[name][3]=input("Enter new BirthDay\n")
            print("BirthDay Updated")
        else:
            print("Wrong choice")


def modify_file():
```

```python
    f=open("FriendsBook.txt","w")
    for i in d:
        s=i+" "+str(d[i])+"\n"
        f.write(s)
    f.close()


def deleteFriend():
    name=input("Enter friend name to delete\n")
    if name in d:
        d.pop(name)
        print("Friend Deleted")
    else:
        print("Friend not found")


def clearFriendsBook():
    password=d['password']
    d.clear()
    d['password']=password
d={}
fetch_friends()
while(True):
    log=login()
    if log==False:
        l=int(input("Enter 1 to login , 2 to exit\n"))
        if l==2:
            break
    else:
        break
```

```python
while(log):

    print("Menu\n")

    print("1.Add Friend\n2.Delete Friend\n3.Show Friend Details\n4.Show all
friends\n5.Update friend details\n6.Clear All friends\n7.Update password\n8.Exit")

    ch=eval(input("Enter your choice\n"))

    if (ch=='1'):

        addFriend()

    elif ch=='2':

        deleteFriend()

    elif ch=='3':

        friendDetails()

    elif ch=='4':

        showFriends()

    elif ch=='5':

        updateFriendDetails()

    elif ch=='6':

        clearFriendsBook()

    elif ch=='7':

        updatePassword()

    elif ch=='8':

        break

    else:

        print("Wrong choice try again")

    c=input("If you want to continue enter y else n\n")

    if c=='n':break


modify_file()
```

Role of Different functions in source code:

1. login() – To take password from user and check whether it is valid or not.
2. updatePassword() – Allows the user to change password.
3. addFriend() – Allows the user to add friend and store his name, address, email, phone number and birthdate.
4. showFriends() – Displays all the friends of user.
5. friendDetails() – Shows all the details stored of specified friend.
6. fetch_friends() – This function reads all the data from the file.
7. updateFriendDetails() –Used to update any detail of the friend.
8. modify_file() – writes the modified data onto the file.
9. deleteFriend() – This is used to delete any particular friend.
10. clearFriendsBook() – This function deletes entire record.

Role of different variables in source code:

1. name – to store name of friend.
2. password – to store password.
3. d – is the python dictionary to store all record.
4. l – is python list to store details of friend.
5. log – is used to store login result.
6. ch – is used to store choice of user.
7. c – to store whether user wants to continue or exit.
8. f – used for file pointer.
9. a – used to store read line from file.

User Interface

The program has a menu driven user interface. A list of options is displayed, user has to enter his choice and the specified action is performed by the program. User decides whether he wants to continue or exit.

Output

```
Enter password
Ani
incorrect password
Enter 1 to login , 2 to exit
1
Enter password
a
login successful
Menu

1.Add Friend
2.Delete Friend
3.Show Friend Details
4.Show all friends
5.Update friend details
6.Clear All friends
7.Update password
8.Exit
Enter your choice
1
Enter name
Anikate
Press enter for blank entry
Enter number
1234456789
Enter email
xyz
Enter address
abc
Enter birthday
11/11/21
If you want to continue enter y else n
y
Menu

1.Add Friend
2.Delete Friend
3.Show Friend Details
4.Show all friends
5.Update friend details
6.Clear All friends
7.Update password
8.Exit
Enter your choice
4
Anikate
If you want to continue enter y else n
```

```
5.Update friend details
6.Clear All friends
7.Update password
8.Exit
Enter your choice
1
Enter name
Anikate Agrawal
Press enter for blank entry
Enter number
9923344999
Enter email
anikate@gmail.com
Enter address
mathura
Enter birthday
31/12/21
If you want to continue enter y else n
y
Menu

1.Add Friend
2.Delete Friend
3.Show Friend Details
4.Show all friends
5.Update friend details
6.Clear All friends
7.Update password
8.Exit
Enter your choice
3
Enter name to search
Anikate Agrawal
Name : Anikate Agrawal
Contact Number : 9923344999
Email : anikate@gmail.com
Address : mathura
BirthDay : 31/12/21
If you want to continue enter y else n
```

Conclusion

This is to conclude that the project was a success. We got to learn a lot doing this project. It was implementation of various concepts of python we learnt throughout the year such as list, dictionary and file handling. It is a menu driven program to help us to store several details of our friends and easily find them any time. It is a login enabled program to ensure user safety and privacy.

We also learnt group coordination and working together on a project. Special thanks to our supervisor who helped us at each and every step and guided us throughout the project.

A word of thanks to our families and friends who helped and guided us for the completion of this project.