Project Report

on

DATA FLAIR – ADDRESS BOOK

Submitted by:

B Sireesha-(R170770)
C Rajeswari-(R170495)

Under the guidance of
M Muni Babu
M.Tech, (Ph.D), Assistant Professor

Department of Computer Science and Engineering

Rajiv Gandhi University of Knowledge and Technologies
(RGUKT), R. K. Valley, YSR Kadapa, Andra Pradesh

Rajiv Gandhi University of Knowledge Technologies
RK Valley, Kadapa (Dist), Andhra Pradesh, 516330

CERTIFICATE

This is to certify that the project work titled “DATA FLAIR – ADDRESS BOOK” is a bonafied project work submitted by Sireesha and Rajeswari in the department of COMPUTER SCIENCE AND ENGINEERING in partial fulfillment of requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering during the year 2021-2022 carried out the work under the supervision.

Internal Guide **Head Of The Department**

Mr. M. Muni Babu Mr. P. Harinadha

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose constant guidance and encouragement crown all the efforts success.

We are extremely grateful to our respected Director, Prof. K. SANDHYA RANI for fostering an excellent academic climate in our institution.

We also express our sincere gratitude to our respected Head of the Department Mr. HARINADH SIR for his encouragement, overall guidance in viewing this project a good asset and effort in bringing out this project.

We would like to convey thanks to our guide at college Mr. M.MUNIBABU for his guidance, encouragement, co-operation and kindness during the entire duration of the course and academics.

Our sincere thanks to all the members who helped me directly and indirectly in the completion of project work. We express my profound gratitude to all our friends and family members for their encouragement.

INDEX

Abstract

Nowadays, In many palces they were using physical things like books and software technologies like xml sheets and etc for storing details of the persons.

In this project we tried to develop a computerized Data Flair – Address book. Our main intention is to allow this application to be used in most retailing to store address, this system work evaluates the address book system of creating and keeping of people and then design and implementation a new system which will fasten the process and also make the process very accurate.

This system is very useful and we can easily the store the details of the persons its also provide some options like edit,view,add,delete.Using this system,an address book on information was gathered and organised to allow you to quicklyview the profile of a persons.

Introduction

As the name specifies "Address book" is a software developed for managing various details of persons.

Through this address book we can handle the details of persons properly which can not be possible to keep in mobile as there are less space in detail section.

This particular project deals with the problems on managing the details of person and avoid the problems of losing the details.

Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system which is more user friendly and more GUI oriented.

We can improve the efficiency of the system, thus overcome the drawbacks of the existing system.

- Less human error
- Strength and strain of manual labour can be reduced
- Easy to handle
- Easy data updating
- Easy record keeping

Purpose

The main purpose of the address book is to stores names,addresses and other contact information for a computer user.

Address books allow is access to the user friends,family,business associates and others by maintaining their contact details on their computer of address lists. The address book is a collection of address lists.

Scope

Address book perform various actions to make the data entry, indexing, archiving relevent,clear,concise and up to date.

Data, in this case,is a name,address and phone number. As your business grows,so will the number of business partners,customers, prospects,and leads in your address book.At one point,you won't be able to do it manually or with the help of simple contact apps.

It will allow you to apply filters to quickly identify specific groups in your contact list,find it in a matter of seconds, and use data as you see fit.

DESCRIPTION

Before developing software we keep following things in mind that we can develop powerful and quality software.

PROBLEM STATEMENT

- Problem statement was to design a module which is user friendly and provide some useful features.
- Which will restrict the user from accessing other user’s data.
- Which will help user in viewing his data and privileges.
- Which will help the administrator to handle all the changes.

FUNCTIONS TO BE PROVIDED:

The system will be user friendly and completely menu driven so that the users shall have on problem in using all options.

- The system will be efficient and fast in response.
- The system will be customized according to needs.
- It would help in reducing the complexity.
- Computerized Address Book system helps in managing things accurately
- It is also a time saving process.

Requirement Specification

Hardware Configuration:

Operating System: Windows 2000/NT/Xp/Vista
RAM: 256 MB or more
HARD DISK 40 GB or more
Processor P3 or High
Compiler Standard Python Compiler

Software Requirement:

Python programming language and it's GUI components.

Python :

Python is high level,general purpose programming language.It's design philosophy emphasizes code readability with use of significant identation. Python is dynamically-typed and garbage-collected.

It supports multiple programming paradigms,including structured,object oriented and functional programming.

Python is commonly used for developing websites and software,task automation, data analysis,and visualization.

Since it's relatively easy to learn,python has been adapted by many non programmers such as accontants and scientists,for a variety of every day task like organizing finances.

Python GUI Libraries:

- The GUI or graphical user interface is the one thing the user sees and interacts with when they open your application.

- A good looking GUI increases the reputation of your product.

- The Tkinter is the standard GUI library for python.

- Python when combined Tkinter provides a fast and easy way to create GUI application.

- Tkinter provides a powerful object oriented interface to the Tk GUI toolkit.

- Creating a GUI application using Tkinter is an easy task.

Analysis and Design

Analysis:

Today also when we go some places they were using the physical things for storing the details of the persons.As Technology is growing rapidly we are also moving to a technical world where everything we want to be online or in a advanced mode. So with the help of this project we are bringing the use of technology in the field of storing details of persons where people can store their details in an easy manner.This project make us easy to store the data and reduce the burden of the people.At a same time its help to edit the details,we can view the details,and it is also a time saving process.

Disadvantage of present system:

- **Not user friendly:** The present system not user friendly because data is not stored in structure and proper format.

- **Manual Control:** All report calculation is done manually so there is a chance of error.

- **Lots of paper work:** Visitors maintain in the register so lots of paper require storing details.

- **Time Consuming**

Design Introduction:

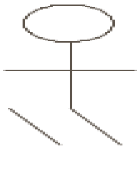
Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization. Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer’s requirements into finished software or a system.

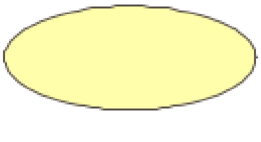
Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data

UML Diagrams:

Actor: A coherent set of roles that users of use cases play when interacting with the use cases. An observable result of value of an actor.



Use case: A description of sequence of actions, including variants, that a system performs yields an observable result of value of an actor. actor diagram is drawned in a eclipse shape



UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

USECASE DIAGRAMS:

Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what’s called an actor.

Use case diagram can be useful for getting an overall view of the system and clarifying that can do and more importantly what they can’t do.

Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

- The purpose is to show the interactions between the use case and actor.
- To represent the system requirements from user’s perspective.
- An actor could be the end-user of the system or an external system.

USECASE DIAGRAM:

A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors.

Use Case Diagrams:

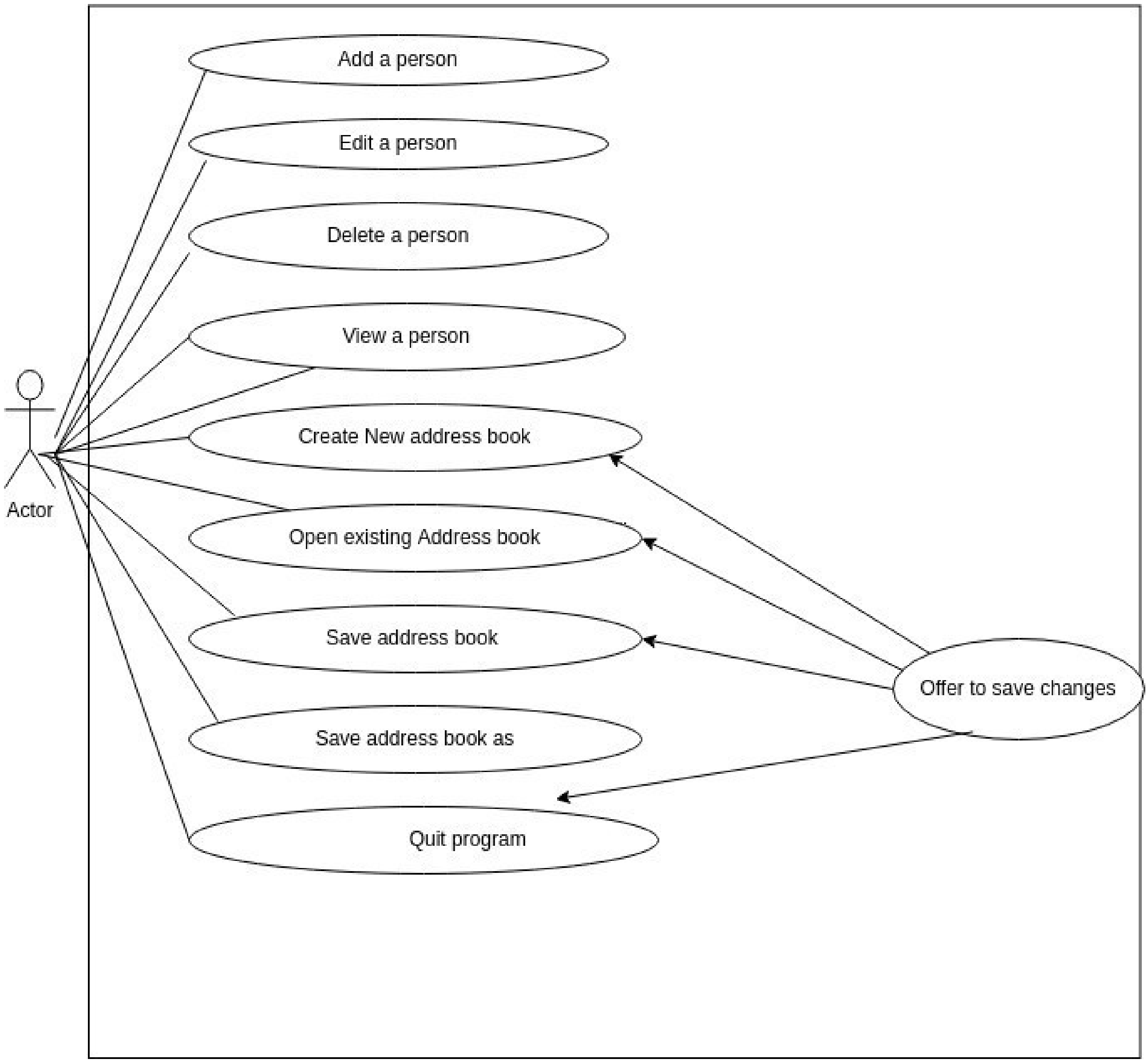


Fig. 1: Use case diagram of “Data Flair - Address Book”

Data Flow Diagram:

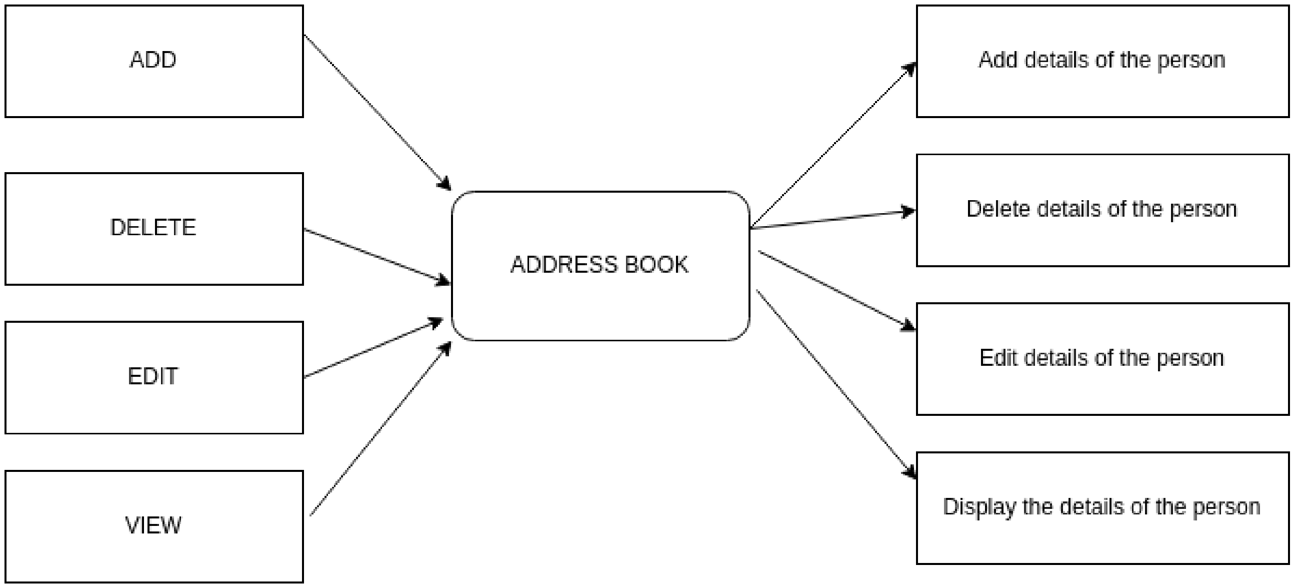

ADD
DELETE
EDIT
VIEW
ADDRESS BOOK
Add details of the person
Delete details of the person
Edit details of the person
Display the details of the person

Fig. 2: Data Flow diagram of “Data Flair - Address Book”

Pictorial View of Prototype:

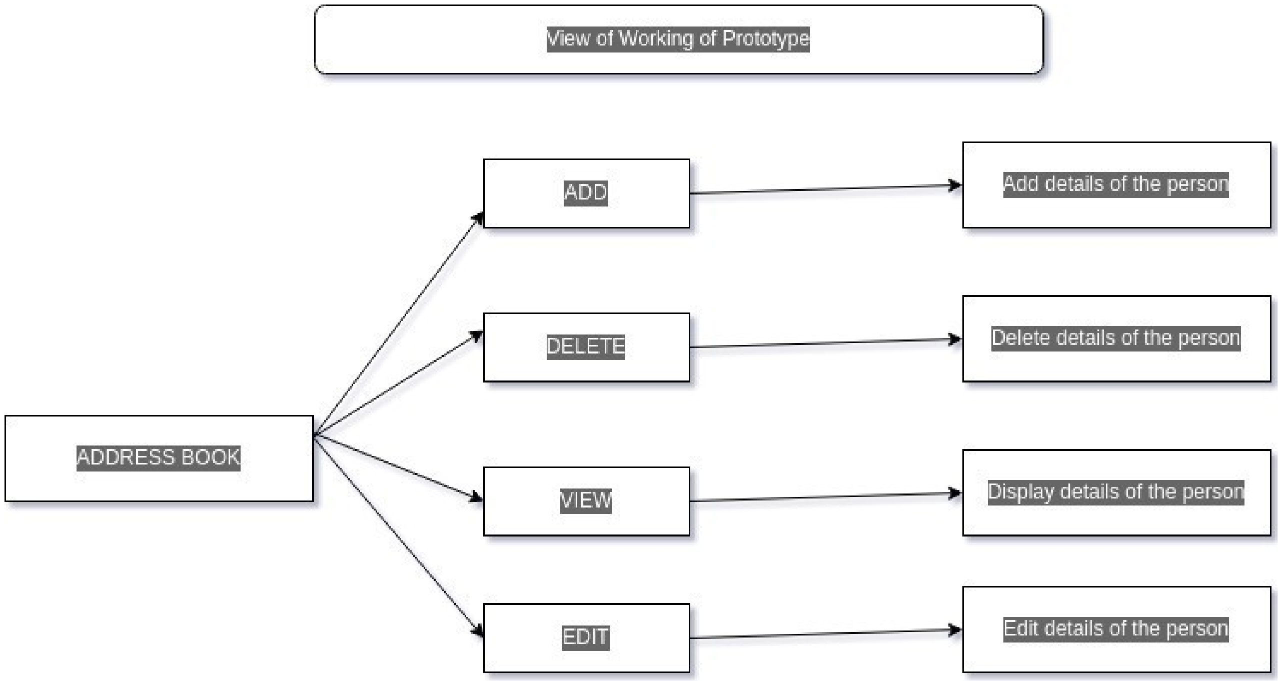

View of Working of Prototype
ADDRESS BOOK
ADD
DELETE
VIEW
EDIT
Add details of the person
Delete details of the person
Display details of the person
Edit details of the person

Fig. 3: Pictorial View of Prototype diagram of “Data Flair - Address Book”

ER Diagram:

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design for the database designer, the utility of the ER model is:

- It maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.

- It is simple and easy to understand with a minimum of training. Therefore, the model canbe used by the database designer to communicate the design to the end user.

- In addition, the model can be used as a design plan by the database developer to implement a data model in specific database management software.

ER Notation

There is no standard for representing data objects in ER diagrams. Each modeling methodology uses its own notation. The original notation used by Chen is widely used in academics texts and journals but rarely seen in either CASE tools or publications by non-academics. Today, there arem a number of notations used; among the more common are Bachman, crow's foot, and IDEFIX.

All notational styles represent entities as rectangular boxes and relationships as lines connecting boxes. Each style uses a special set of symbols to represent the cardinality of a connection. The notation used in this document is from Martin. The symbols used for the basic ER constructs are:

- **Entities** are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.

- **Relationships** are represented by a solid line connecting two entities. The name of the relationship is written above the line. Relationship names should be verbs

- **Attributes**, when included, are listed inside the entity rectangle. Attributes which are identifiers are underlined. Attribute names should be singular nouns.

- **Cardinality** of many is represented by a line ending in a crow's foot. If the crow's foot is omitted, the cardinality is one.

Existence is represented by placing a circle or a perpendicular bar on the line. Mandatory existence is shown by the bar (looks like a 1) next to the entity for an instance is required. Optional existence is shown by placing a circle next to the entity that is optional.

ER Diagram

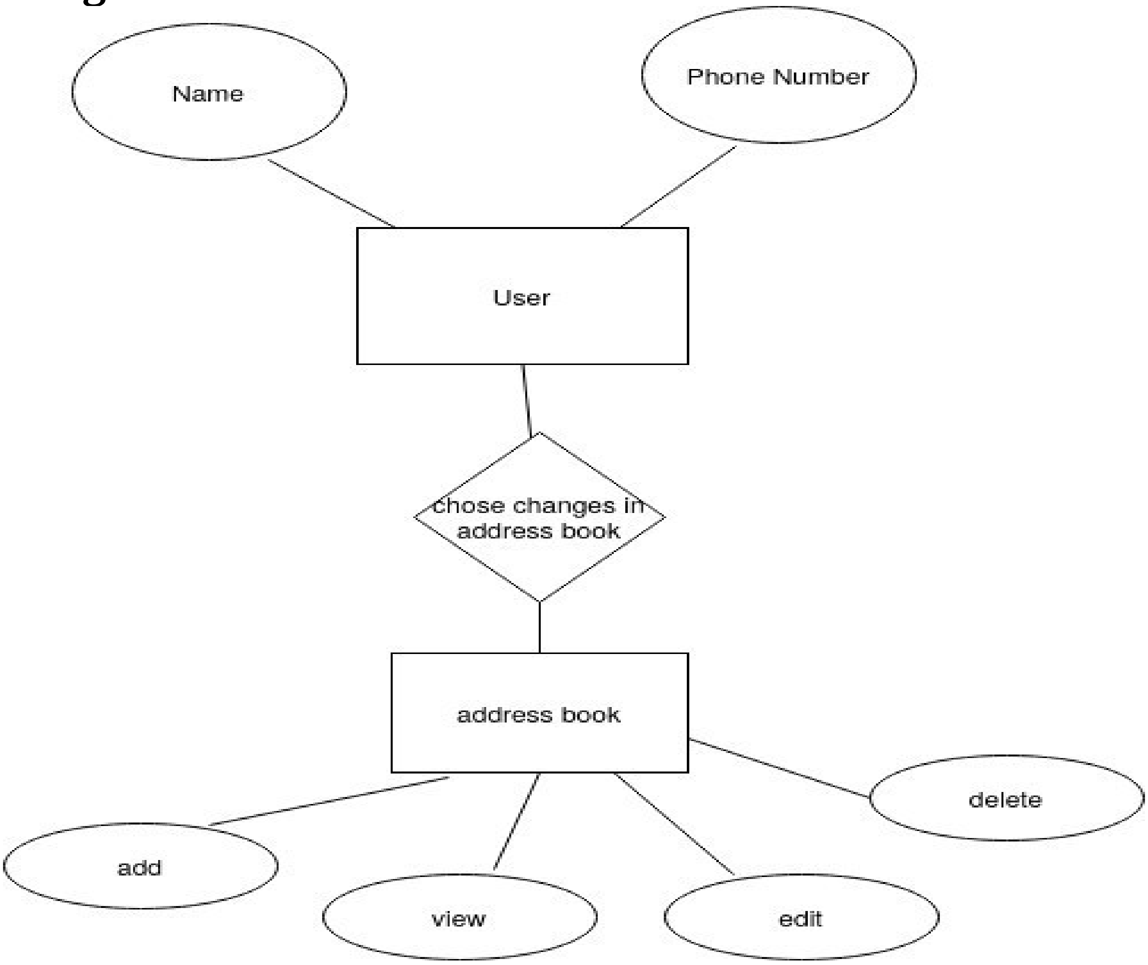


Fig. 4: ER diagram of “Data Flair - Address Book”

Implementation and System Testing

After all phase have been perfectly done, the system will be implemented to the server and the system can be used.

System Testing

The goal of the system testing process was to determine all faults in our project .The program was subjected to a set of test inputs and many explanations were made and based on these explanations it will be decided whether the program behaves as expected or not. Our Project went through two levels of testing.

1. Unit testing
2 .Integration testing

Unit Testing

Unit testing is commenced when a unit has been created and effectively reviewed .In order to test a single module we need to provide a complete environment i.e. besides the section we would require The procedures belonging to other units that the unit under test calls Non local data structures that module accesses .A procedure to call the functions of the unit under test with appropriate parameters.

1. Test for the admin module

Testing admin login form: This form is used for log in of administrator of the system. In this form we enter the username and password if both are correct administration page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask the
details.

Report Generation: admin can generate report from the main database.

Integration Testing

In the Integration testing we test various combination of the project module by providing the input.

The primary objective is to test the module interfaces in order to confirm that no errors are occurring when one module invokes the other module.

Source Code

Python and It's GUI libraries:

```
#import library
from tkinter import *
#initialize window
root = Tk()
root.geometry('1350x1350')
root.config(bg = 'SlateGray3')
root.title('DataFlair-AddressBook')
root.resizable(0,0)
contactlist = [
['Parv Maheswari', '0176738493'],
['David Sharma', '2684430000'],
['Mandish Kabra', '4338354432'],
['Prisha Modi','6834552341'],
['Rahul kaushik', '1234852689'],
['Johena Shaa' , '2119876543'],
]

Name = StringVar()
Number = StringVar()
#create frame
frame = Frame(root)
frame.pack(side = RIGHT)
scroll = Scrollbar(frame, orient=VERTICAL)
select = Listbox(frame, yscrollcommand=scroll.set, height=18)
scroll.config (command=select.yview)
scroll.pack(side=RIGHT, fill=Y)
```

```
select.pack(side=LEFT, fill=BOTH, expand=1)
########### function to get select value
def Selected():
return int(select.curselection()[0])
##fun to add new contact

def AddContact():
contactlist.append([Name.get(), Number.get()])

Select_set()
# fun to edit existing contact(first select the contact then click on view
button then edit
#the contact and then click on edit button)

def EDIT():
contactlist[Selected()] = [Name.get(), Number.get()]
Select_set()
#to delete selected contact
def DELETE():
del contactlist[Selected()]
Select_set()
# to view selected contact(first select then click on view button)

def VIEW():
NAME, PHONE = contactlist[Selected()]
Name.set(NAME)
Number.set(PHONE)
###exit game window
def EXIT():
root.destroy()
#empty name and number field

def RESET():
Name.set('')
Number.set('')
def Select_set() :
contactlist.sort()
select.delete(0,END)
for name,phone in contactlist :
```

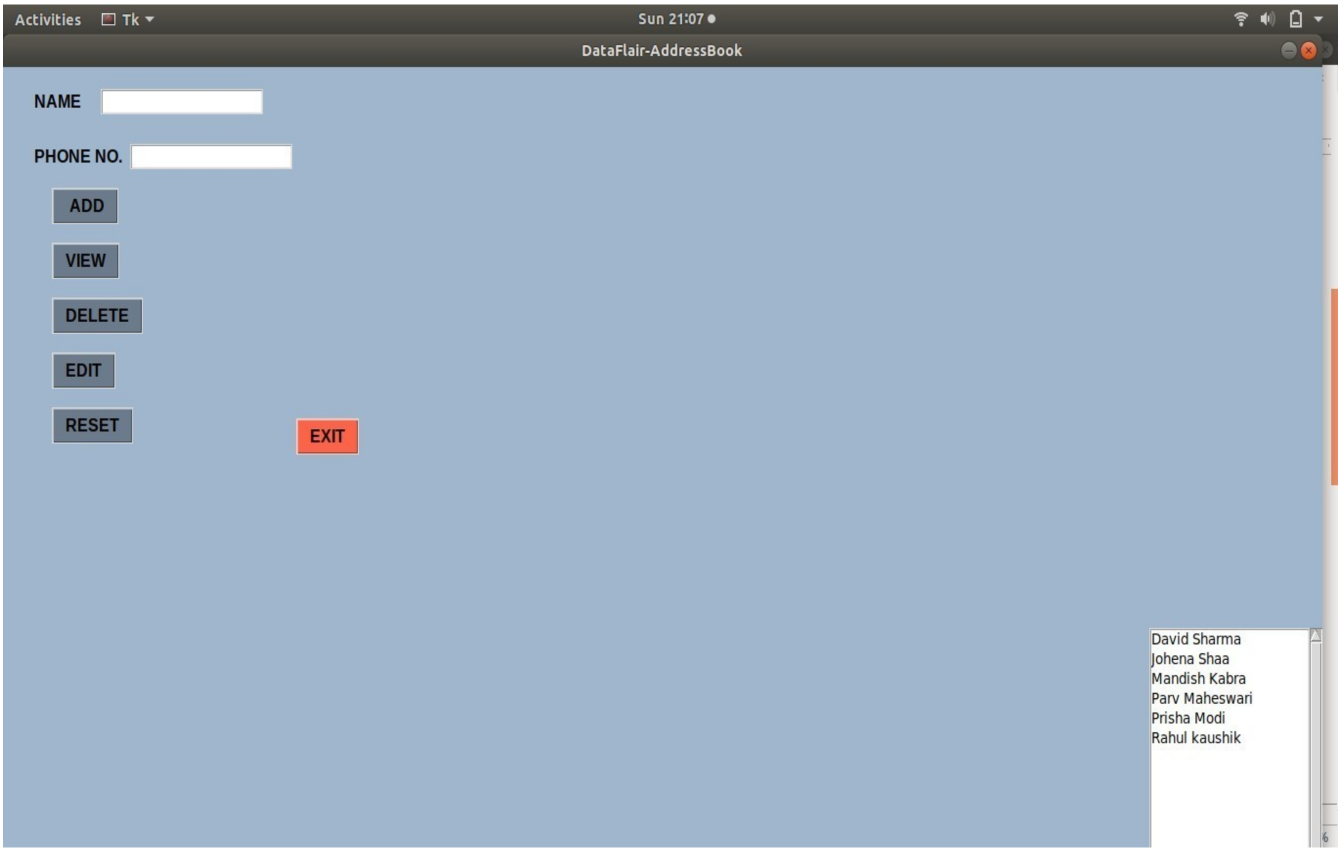
```
select.insert (END, name)


Select_set()
######define buttons #####labels and entry widget
Label(root, text = 'NAME', font='arial 12 bold', bg =
'SlateGray3').place(x= 30, y=20)
Entry(root, textvariable = Name).place(x= 100, y=20)
Label(root, text = 'PHONE NO.', font='arial 12 bold',bg =
'SlateGray3').place(x= 30,
y=70)
Entry(root, textvariable = Number).place(x= 130, y=70)
Button(root,text=" ADD", font='arial 12 bold',bg='SlateGray4', command
=AddContact).place(x= 50, y=110)
Button(root,text="EDIT", font='arial 12 bold',bg='SlateGray4',command =
EDIT).place(x= 50, y=260)
Button(root,text="DELETE",font='arial12 bold',bg='SlateGray4',command
= DELETE).place(x= 50, y=210) Button(root,text="VIEW", font='arial 12
bold',bg='SlateGray4', command = VIEW).place(x= 50, y=160)

Button(root,text="EXIT", font='arial 12 bold',bg='tomato', command =
EXIT).place(x= 300, y=320)
Button(root,text="RESET", font='arial 12 bold',bg='SlateGray4', command
=RESET).place(x= 50, y=310)
root.mainloop()
```

The presentation of the project

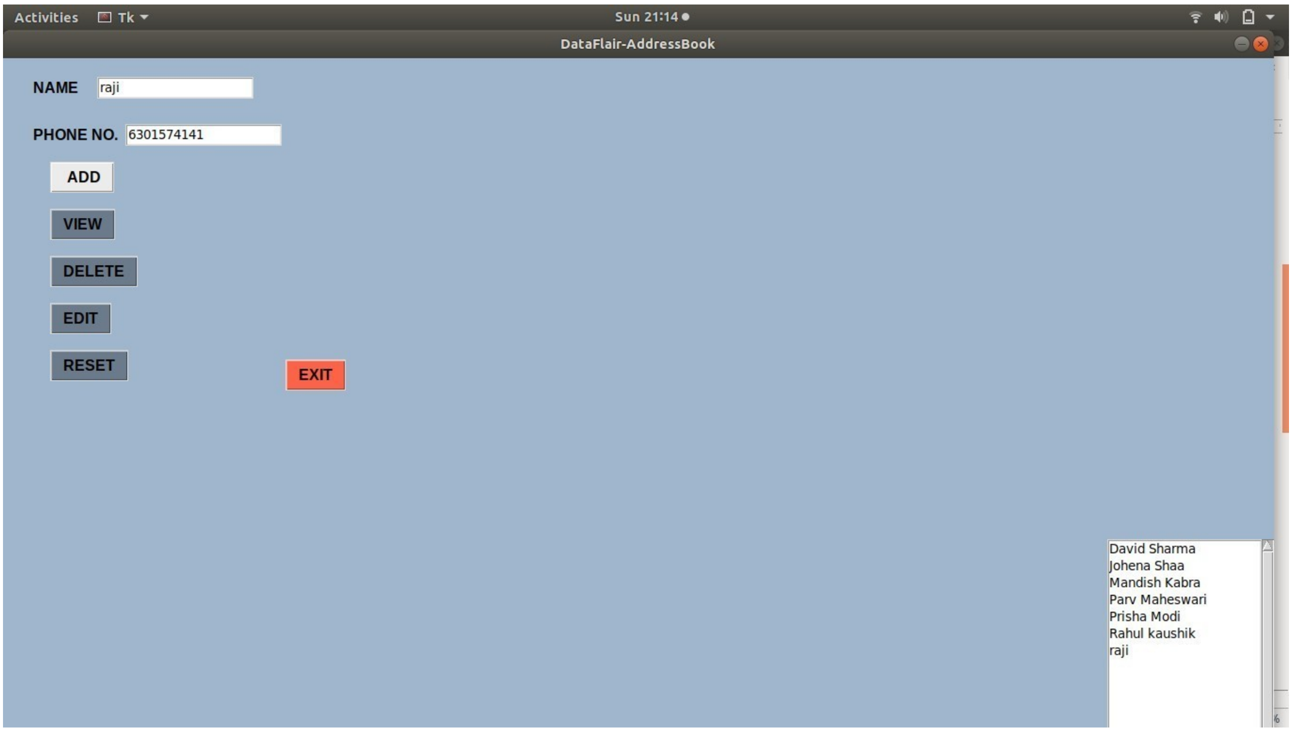
Home page:



David Sharma
Johena Shaa
Mandish Kabra
Parv Maheswari
Prisha Modi
Rahul kaushik

Fig.5:Home page of "Data Flair – Address Book"

Add new: This tool use to add a new name to the address book where the user can add the names fast and easy to the Address Book. First, you must click on the Add New



NAME raji

PHONE NO. 6301574141

David Sharma
Johena Shaa
Mandish Kabra
Parv Maheswari
Prisha Modi
Rahul kaushik
raji

Fig.6: Output of the add new module

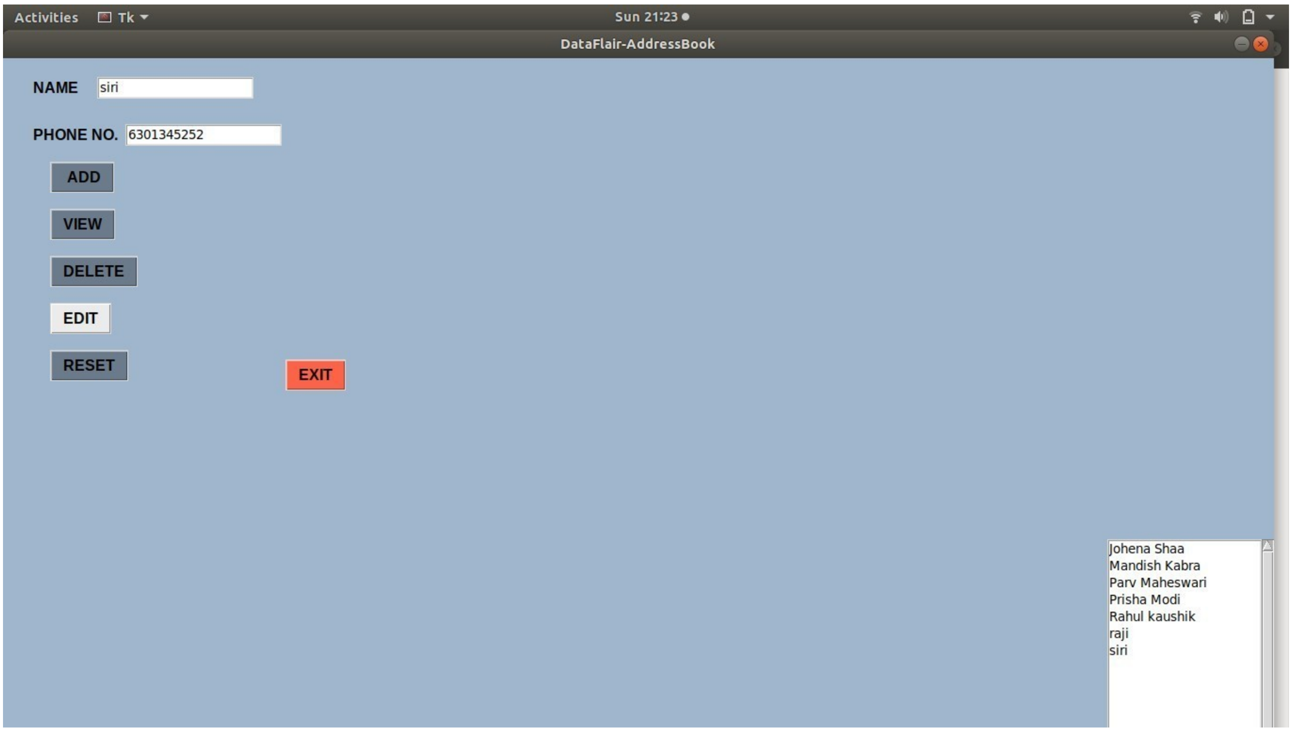
Edit:

This tool is used to edit the existing information of the user within the Address book.

First, you must choose the name.

After that press the edit button.

After that show information stored within the addressbook to be modified.



NAME siri

PHONE NO. 6301345252

Johena Shaa
Mandish Kabra
Parv Maheswari
Prisha Modi
Rahul kaushik
raji
siri

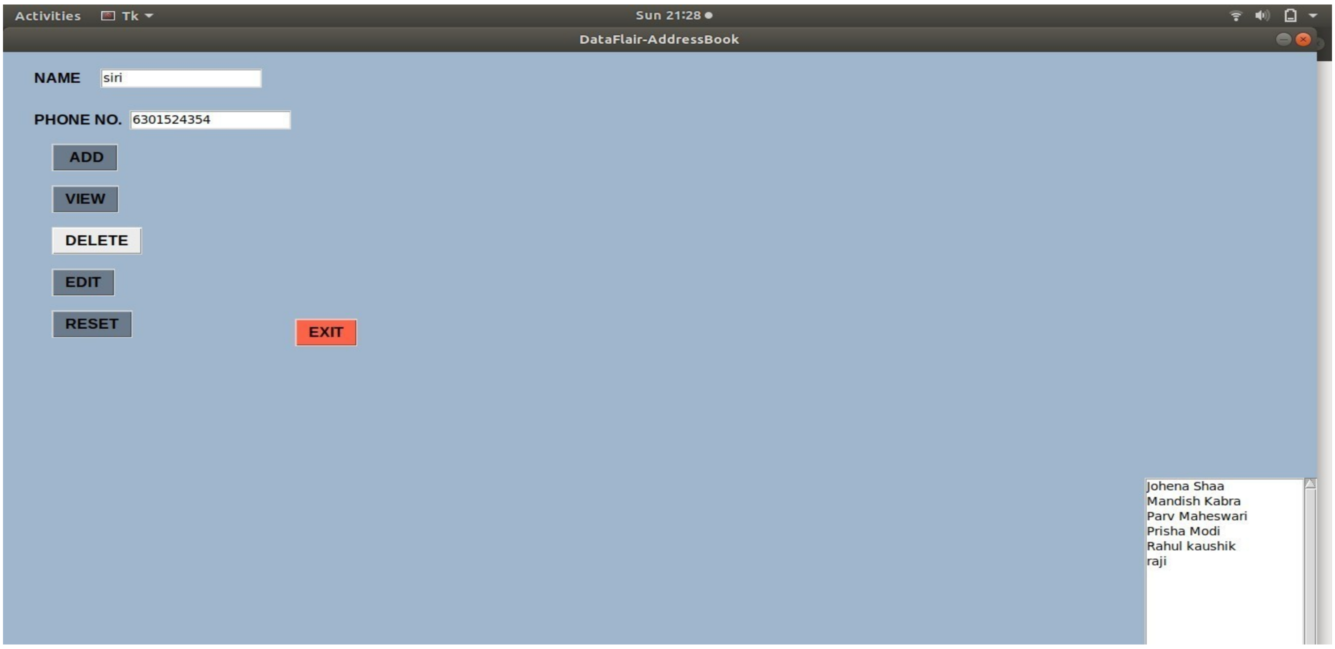
Fig.7: Output of the edit module

Delete

This tool used to delete the existing details of the person within the address book.

First, you must choose the name

After that press the Delete button.



NAME siri

PHONE NO. 6301524354

```
Johena Shaa
Mandish Kabra
Parv Maheswari
Prisha Modi
Rahul kaushik
raji
```

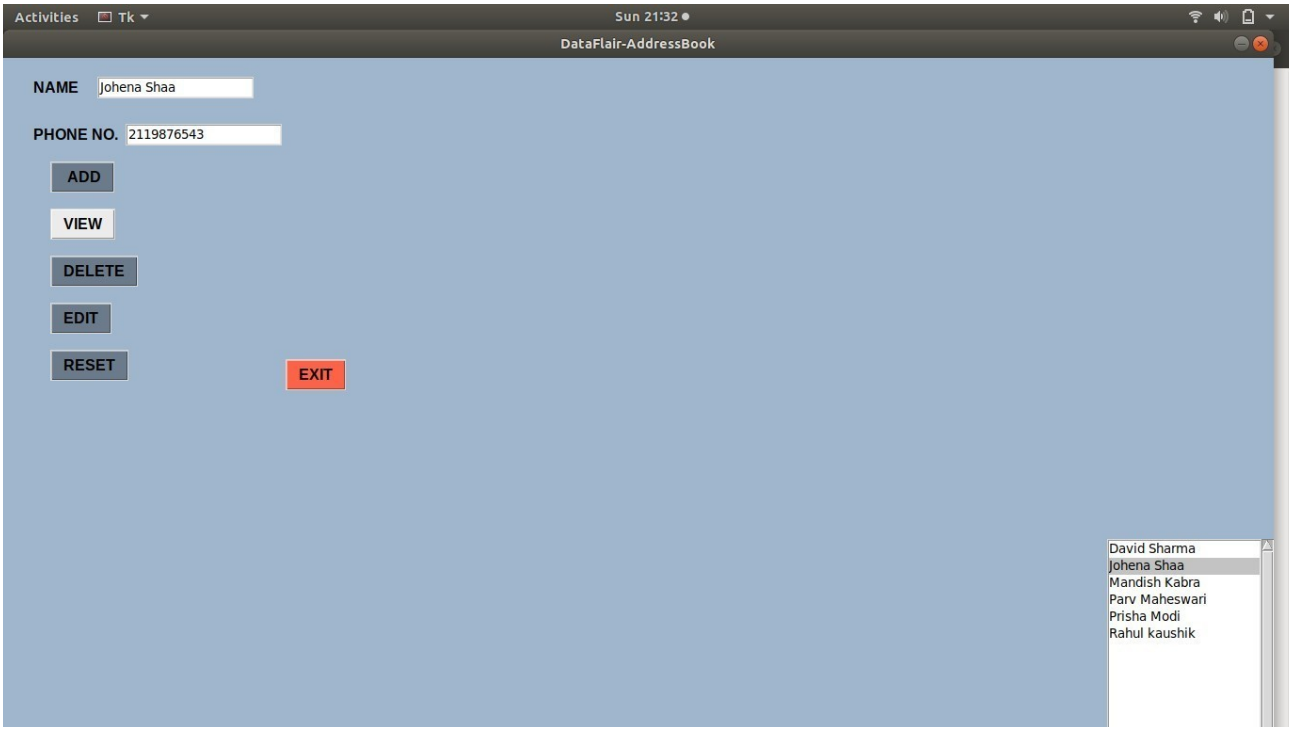
Fig.8: Output of the delete module

View:

In order to view the details of the existing person

first we need select the name

and then we have click on the view button.



NAME: Johena Shaa

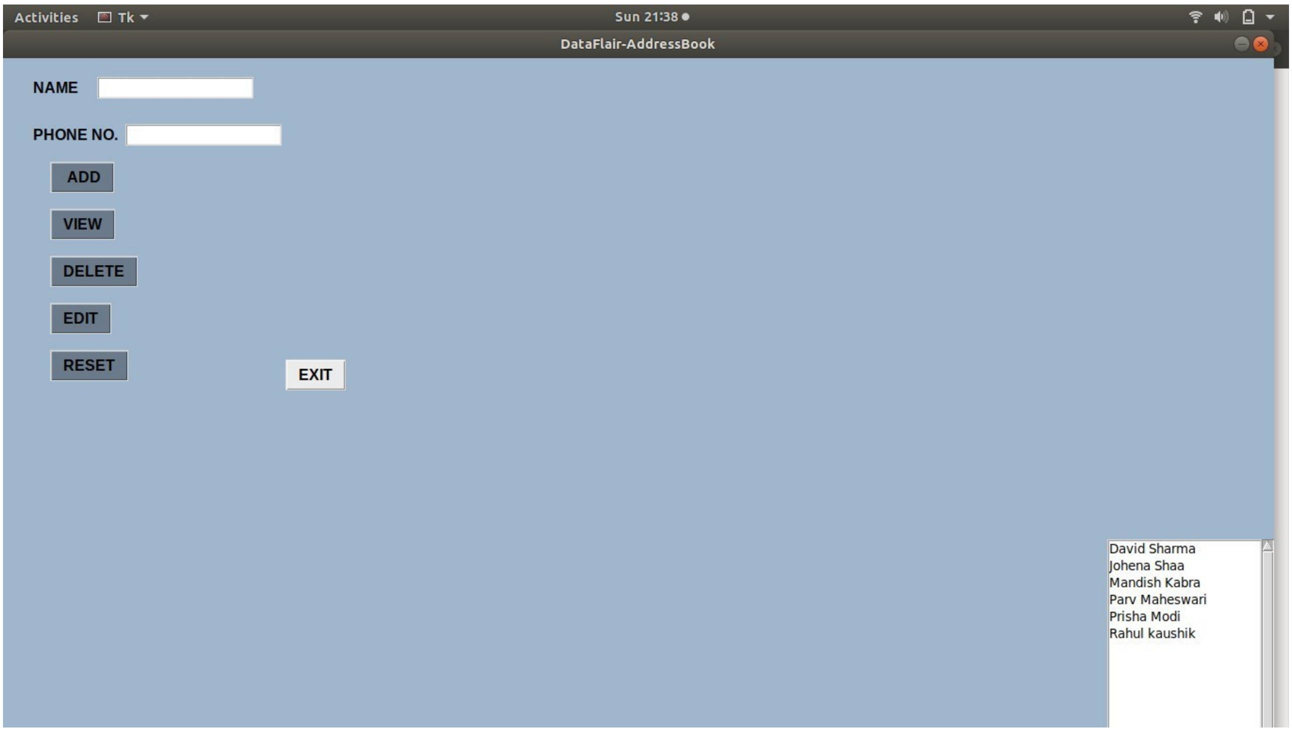
PHONE NO.: 2119876543

David Sharma
Johena Shaa
Mandish Kabra
Parv Maheswari
Prisha Modi
Rahul kaushik

Fig.9:Output of the view module

Exit:

In order to exit from the address book we need click on the Exit button.



David Sharma
Johena Shaa
Mandish Kabra
Parv Maheswari
Prisha Modi
Rahul kaushik

Fig.10:Output of the exit module

CONCLUSION:

This is the application which can be used in real time projects. Through this application we can store many user details and we can able to access the details of the particular user.

Address Book is a contact management software, where you can note the addresses, phone numbers, mobile, country, city, website address and email addresses of your contacts and partners. You can easy print list of contacts and details of selected contact, has a local database and user interface for finding and editing information about people.

--->> Updating of information becomes so easie

--->>Strength and strain of manual labour can be reduced

--->>Easy data updating

--->>Easy record keeping

References

For Python :

https://www.geeksforgeeks.org/python

https://wiki.python.org/moin/GuiProgramming

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

For Python GUI:

https://www.tutorialspoint.com/pytho/python_gui_programming.html

https://www.geeksforgeeks.org/python-gui-tkinter/