## A REPORT ON

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN THE PARTIAL FULFILLMENT OF THE REQUIREMENT

FOR

**FUNDAMENTALS OF DATA STRUCTURES**

## SUBMITTED BY



**DEPARTMENT OF COMPUTER ENGINEERING**

## ARMY INSTITUTE OF TECHNOLOGY

DIGHI HILLS, ALANDI ROAD, PUNE 411015

## SAVITRIBAI PHULE PUNE UNIVERSITY 2019-2020



**CERTIFICATE**

This is to certify that the project report entitles

SUBMITTED BY

Sachin kumar (3244)

Arvind kumar (3211)

are bonafide students of this institute and the work have been carried out by them under the supervision of **Prof. Vaishali Ganganwar** and it has been approved for the partial fulfilment of the requirement of mini-project for the subject Fundamentals of Data Structures.

Place: Pune

Date: 4/02/2020

# ABSTRACT

We had basically created an address book. We have also a login feature in it.

The admin can control everything which the address book offers.

Following are the features offered by the program for the admin:

1. Delete by phone no
2. Delete by name
3. Search by address
4. Reverse the address book
5. Print details
6. Search phone
7. Search Name
8. Exit

But, only following features are offered to User

1.Search (Name)

2. Search (Phone no)

3. Search address ()

4. Print all

5. exit

**PROJECT IMPLEMENTATION**

**The implementation of our project is based on storage of data through file handling and the data structure used is linked list.**

**Whenever user access our address book then can perform a number of operations such as searching through phone number, searching through address book, Print information exit.**

**So we will se step wise step what the flow of code is all about**

**On boarding screen**

**The onboarding screen will contain following things**

**Bool admin Login?**

**If No return**

**Else Input Username and password**

**Verify From Secret file**

**If found Authentic and Admin Login**

**Else User Login**

**TO store the user information in hard disk we have taken help of file handling**

**Pseudo code of file handling**

**If file doesnot exist return -1**

**If file is empty return 0**

**Else read the file and store in linked list**

**Return the length of Linked List**

**Now we will look at different ooeration we can perform**

**Searching using Name**

**Modifying Using name**

**Delete using name**

**Search Using phone number**

**Modifying Using phone number**

**Delete using Phone Number**

**Reversing Address Book**

**Appending New Contact**

**Print all contact**

**Searching Using Name**

**Pseudo code:**

**Input Name to be searched**

**If list is empty Print Empty print Empty Record and Return**

**Else while List End Is not Reached**

**If name to be searched is equal to current Name**

**Print record**

**Continue**

**Return Count**

**Modifying Using Name**

**Pseudo-code:**

**Input name, Phone number address to be changed**

**If list is empty print empty record and return**

**Else while list End is not reached**

**And return**

**Else while list end is not reached**

**If name is to be searched is equal to current name**

**Update new data**

**Continue**

**Return continue**

**Delete using Name**

**Pseudo code:**

**Input contact name to be deleted**

**If list is empty print record and return**

**Else while list end is not reached**

**If name is to be searched is equal to current name**

**Delete record**

**Continue**

**Return count**

**Searching using phone number**

**Pseudo code:**

**Input phone number to be searched**

**If list is empty print empty record and return**

**Else while list end is not reached**

**If phone to be searched is equal to to current phone number**

**Print record**

**Continue**

**Return count**

**Modifying using phone number**

**Pseudo-code:**

**Input name, phone number address to be changed**

**If list is empty print empty record and return**

**Else while list end is not reached**

**If number to be searched is equal to current number**

**Update new data**

**Continue**

**Return count**

**Delete using phone number**

**Pseudo-code**

**Input phone number to be deleted**

**If list is empty print empty record and return**

**Else while list end is not reached**

**If number to be searched is equal to current number**

**Delete record**

**Continue**

**Return count**

**Reverse Contacts**

**Pseudo-code:**

**Create temp variable**

**Rev=null be reversed list**

**While temp!= Null**

**Te= rev**

**Rev=temp**

**Temp->Next=temp**

**Start=rev**

**Reverse contacts**

**Pseudo-code:**

**Create Temp variable**

**Rev=null be reversed list**

**While temp!=Null**

**Te=rev**

**Rev=Temp**

**Temp->next=temp**

**Start=rev**

**Display all**

**Pseudo-code:**

**If start is null print empty return**

**While start is not null**

**Print record**

**Continue**

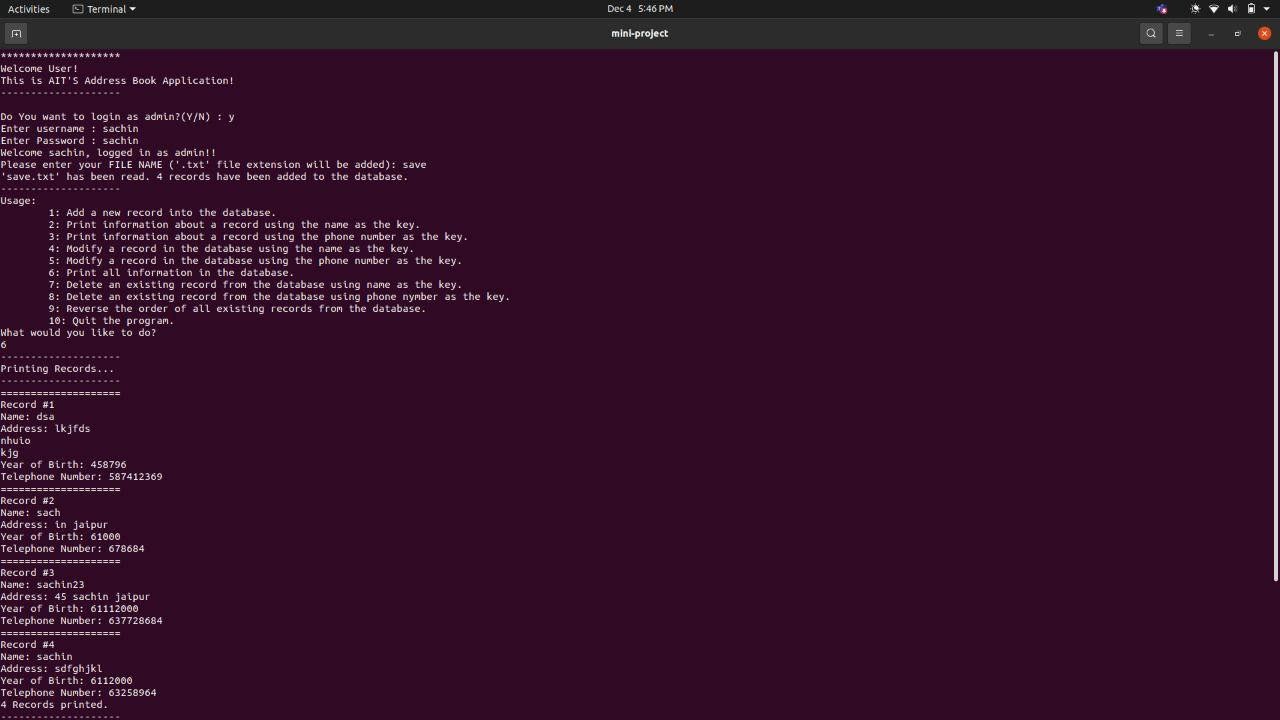
**And we have seen all implementation**

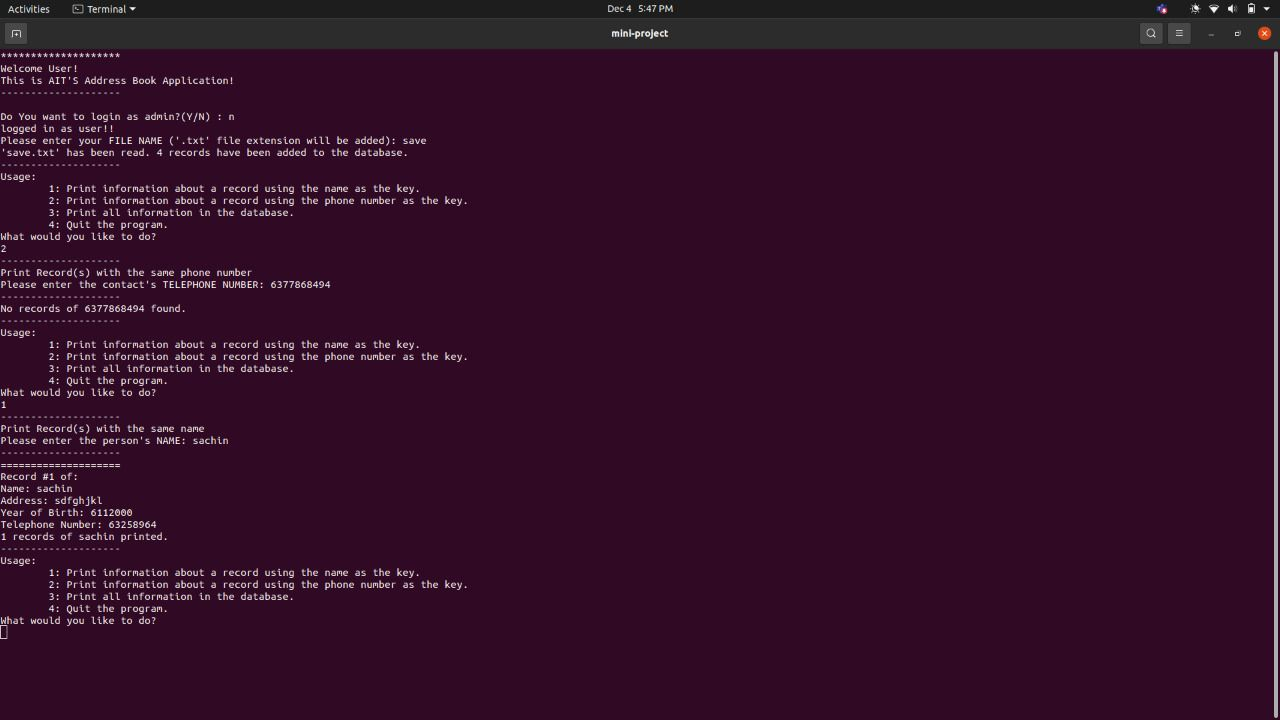
**TOOLS AND TECHNOLOGIES USED**

C++ and OOP

Codeblocks

**RESULT**





**CODE**

**We had submitted code file.**