**PROJECT REPORT ON**

**CONTACT MANAGEMENT DATABASE SYSTEM**

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

##### **SUBMITTED BY**

|  |  |
| --- | --- |
| Name : RAJ MESTA  ADITI KHEDKAR   ANSHULA AWASTHI | Roll No : 120  116  101 |
|  |  |

****

## **DEPARTMENT OF COMPUTER ENGINEERING**

## **STES’S SMT. KASHIBAI NAVALECOLLEGE OF ENGINEERING**

**VADGAON BK, OFF SINHGAD ROAD, PUNE 411041**

**SAVITRIBAI PHULE PUNE UNIVERSITY**

## **2019 - 2020**

****

**CERTIFICATE**

This is to certify that the project report entitled

**CONTACT MANAGEMENT DATABASE SYSTEM**

Submitted by

**NAME EXAM NO:**

RAJ MESTA 120

ADITI KHEDKAR 116  
 AWASTHI101

is a bonafide work carried out by her/ him under the supervision of **Prof.** and it is approved for the partial fulfillment of the requirement of University of Pune as a part Database Management Lab work syllabus (Third year Computer Engineering).

|  |  |
| --- | --- |
| **( )** | **(Dr. P. N. Mahalle)** |
| Department of Computer Engineering | Head, Department of Computer Engineering |

**ACKNOWLEDGEMENT**

ABSTRACT

An average business organization tends to lose about fifty percent (50%) of its customers within a business cycle of five years. This mainly occurs due to poorly met customer needs, which keep changing from time to time. So it becomes increasingly important for an organization to effectively understand the requirements of its customers in an effort to keep them coming back for more.

The principle behind constructing a Contact Management System is to effectively retrieve and implement any information that an organization may have on a pre-existing customer. All the information related to a particular customer can be linked and archived only to be retrieved later when they are required most. What should be more noteworthy is the fact that all of this can be arranged over every single terminal and at an affordable cost.

Once a company has established a Contact Management System, it can cater to the needs of the customer better. The records of the customer can now be studied to understand his or her preferences and implement them for each individual lead. This gives way to contact personalization.

PROBLEM DEFINITION

The project is mainly based on the following objectives:

1. To create a project using python library tkinter and provide relational database connectivity using sqlite3.
2. To implement features like inserting,updating ,saving and deleting operation on records.
3. To be familiar with resource reusability by making user defined functions.
4. To make program easy while running it.
5. To provide user friendly GUI using the tkinter library as frontend,python as backend programming language and using sqlite3 for creating tables and relational connectivity with sqlite3 .
6. Centralized, simple and efficient management solution.
7. To give a brief idea about project architecture.
8. To achieve the normalization of queries to the highest form.
9. To draw precise ER and schema diagram emphasising on the relationships between the tables and their various constraints.
10. To make the system compatible and hardware-free to be run on any environment.

PROJECT ARCHITECTURE

The project is layered in three-tier architecture providing scalability, security, fault tolerance, reusability and maintainability. It helps developers to create flexible and reusable applications.

**The Three-tier Architecture is divided into 3 layers:**

Presentation Layer (Tkinter)

Business Layer:(Sqlite3)

Backend layer (Python and Query Processing using Sqlite3)

**Presentation Layer:**

It is also known as Client layer. Top most layer of an application. The main functionality of this layer is to communicate with Application layer. In the system, we use tkinter for providing graphical interface for the user to enter the contact details.

**Business layer:**

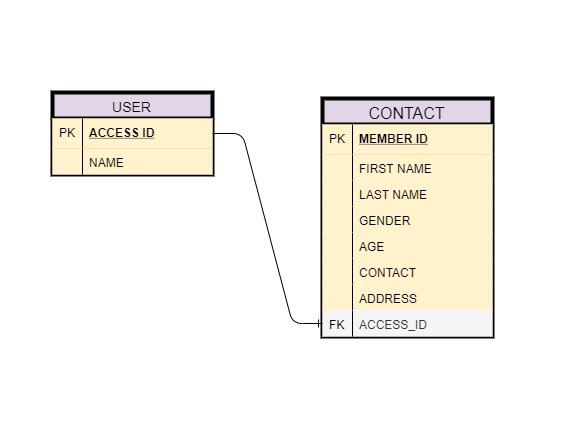
It is also called business logic layer or logical layer. As per the Contact login page example, once user enters the name and access id and clicks on clicks on the login button, business layer interacts with Database layer and sends the required information to the Presentation layer. It controls an application’s functionality by performing detailed processing. This layer acts as a mediator between the Presentation and the Database layer. Complete business logic will be written in this layer.

**Data layer:**

The data is stored in this layer. We use python as the backend language and business layer communicates with Database layer to retrieve the data. It contains methods that connects the database and performs required action e.g.: insert, update, delete etc.

ENTITY RELATIONSHIP DIAGRAM

SCHEMA DIAGRAM



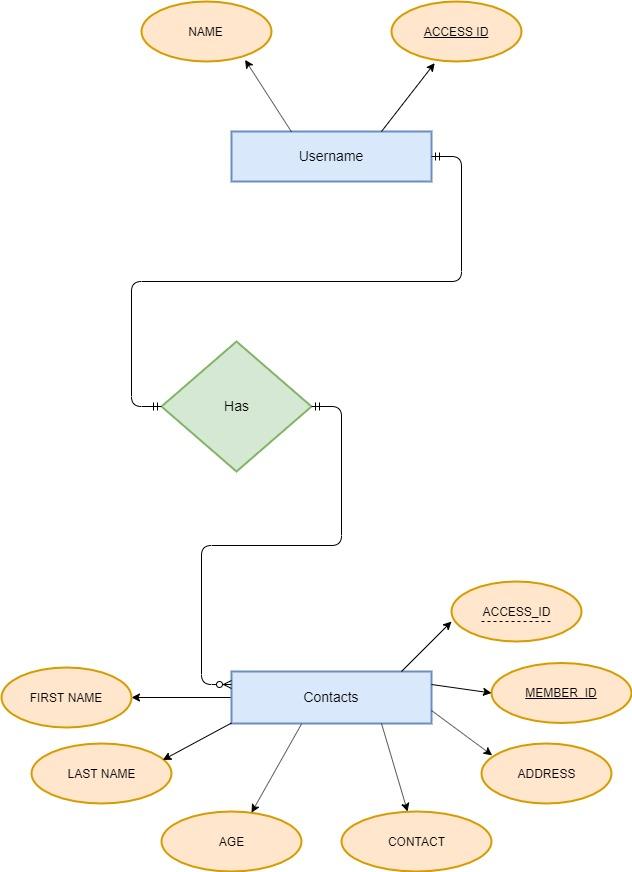
Here are the geometric shapes and their meaning in an E-R Diagram.

**Rectangle**: Represents Entity sets.

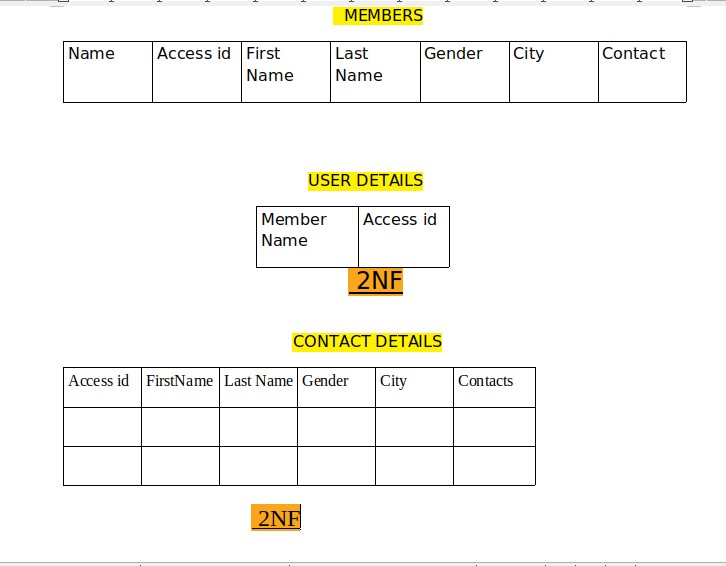
**Ellipses**: Attributes

**Diamonds**: Relationship Set

**Lines**: They link attributes to Entity Sets and Entity sets to Relationship Set



Database Normalization



The table “members” has attributes member id,access id,first name, Last name,Gender,City,Contact.It does not follow any normalization form. Hence, we divide it into two tables “user name” having attributes “member name” and “access id” and “contact details” having attributes “access id”,” first name”, “Last name”,”City”,”Contact”.

Here username has the highest normal form as 2NF.

Contact Details has the highest normal form as 2NF.

Software and Hardware requirement

**Hardware specification**

Processor : i5 Core Processor

Clock speed : 2.5GHz

Monitor : 1024 \* 768 Resolution Color

Keyboard :Any Keyboard

RAM : 1 GB

Input Output Console for interaction

**Software specification**

Sqlite3

Python libraries -tkinter

IDE- Spyder,Pycharm,Terminal

Apache Tomcat Server

Operating system : Windows10,Linux.

Project description

The purpose of this project is to outline Contact Management data and requirements, The purpose of this project is to develop a data management system to document, store and distribute information related to contact management system. The project performs operations like insertion,displaying,updating and deleting contacts maintaining the integrity and normalization of the database.

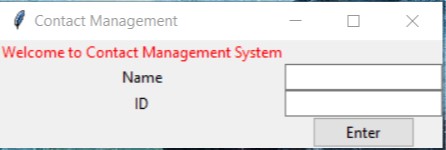
The project initially asks the user to enter his/her name and the given access id.If access id matches the access id of the contact system,user is directed to his/her contact details. During access, user cannot view other people’s contact details without id and hence maintains the security of the database. Furthermore, User is asked to insert the record with fields like ,Member id,First Name, Last name,Gender,City,Contact,Access id.

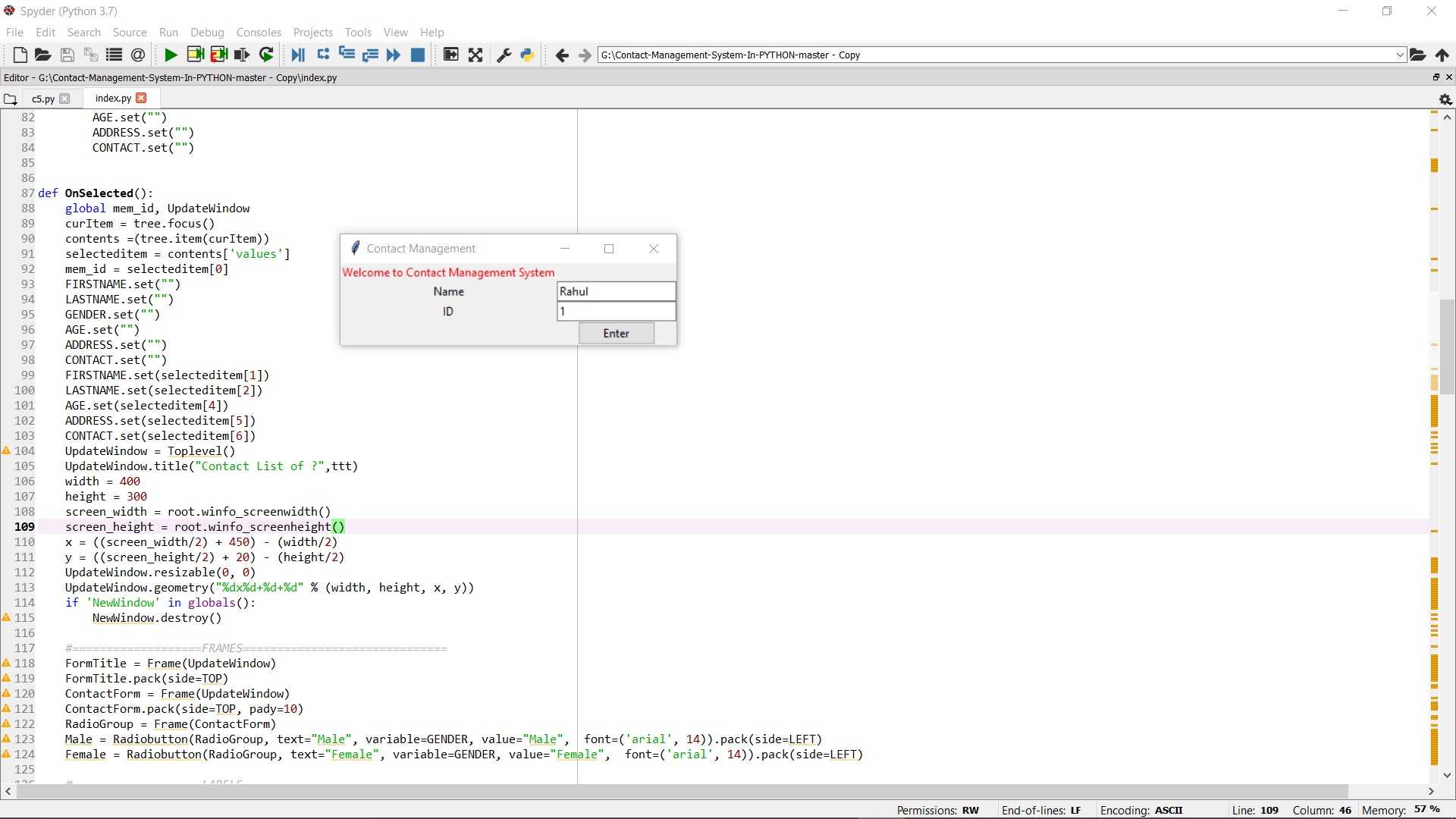
The Member id is the unique field which auto increments when contact is added acting as the primary key of the contacts table, user enters string values for First name, Last Name,Gender,City and int value for Contact, entering the submit button. Upon submitting, user can see the information he displayed. User can also update his/her record or delete records.

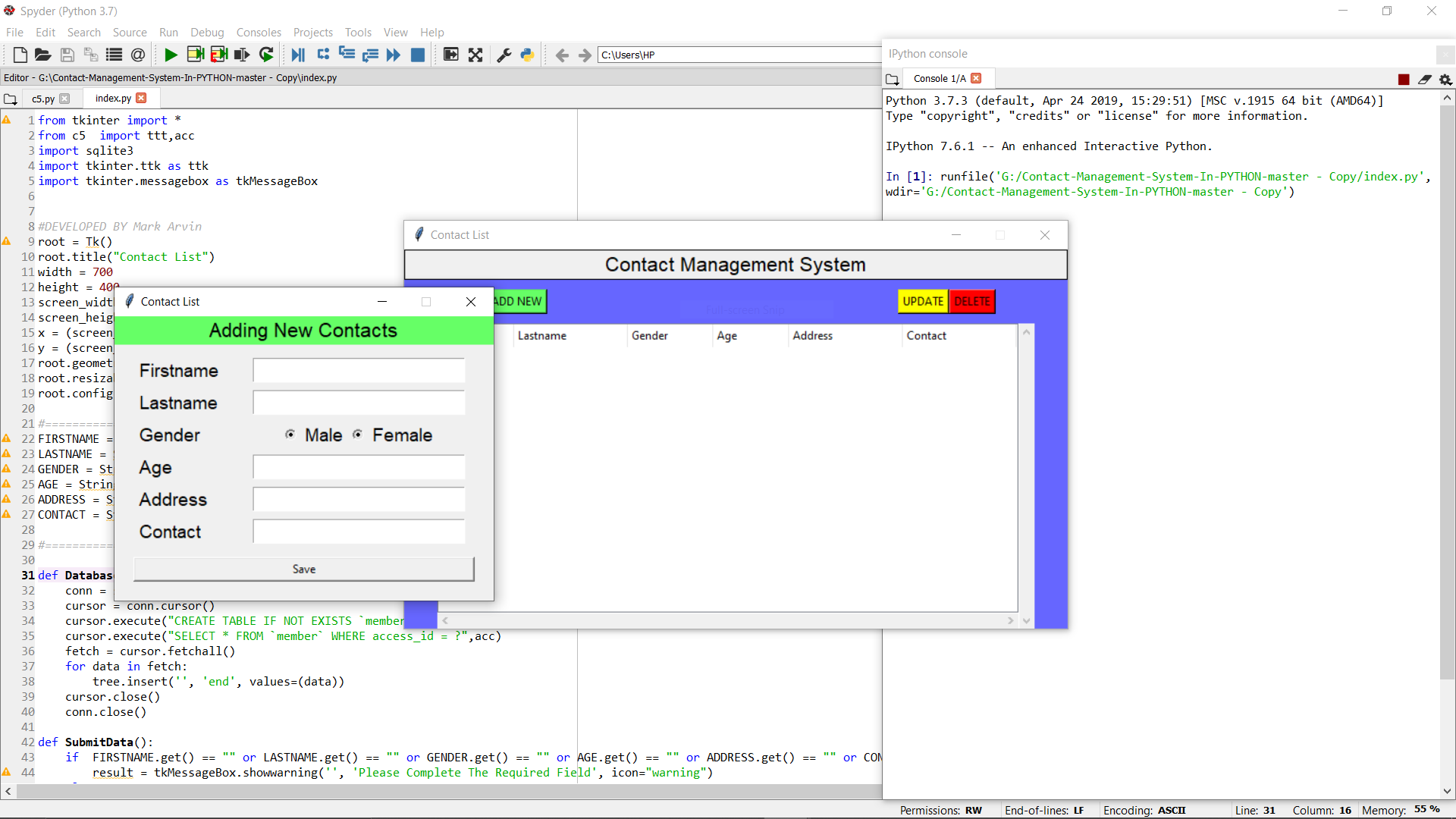
The functional flow of the project is

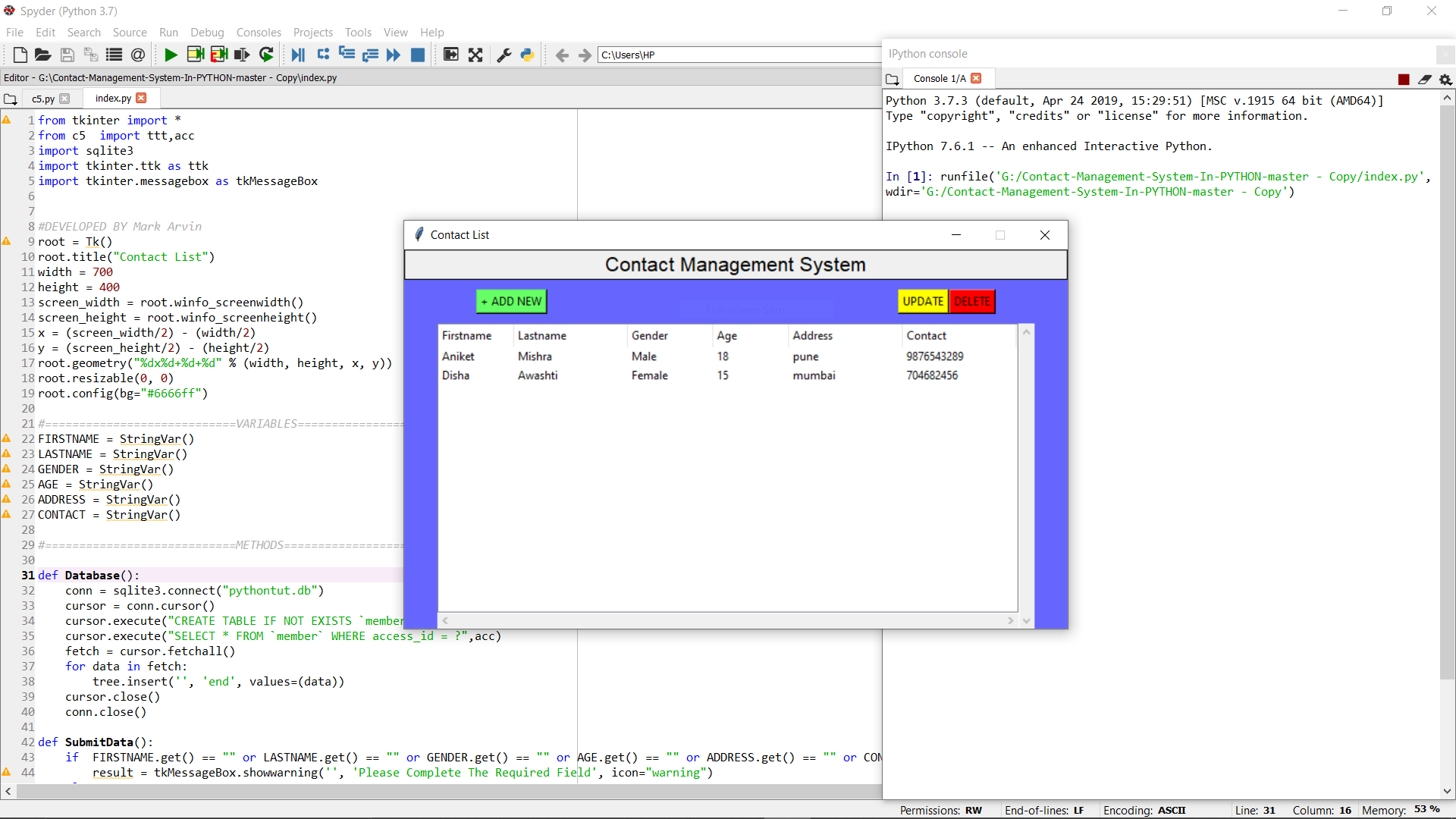
|  |  |
| --- | --- |
| **FUNCTION** | **DESCRIPTION** |
| Database() | It connects to Sqlite3 database  Creates the table Members if it doesn’t exist  Selects all the records and copies it to the tree |
| Submit Data() | Inserts the new record to the members table |
| Update Data() | Updates the selected record from the table |
| On Selected() | After selecting the record to be updated, it pops out new window for changing the record as per the user  Then calls the update Data() function |
| Delete Data() | Deletes the selected record from the table |

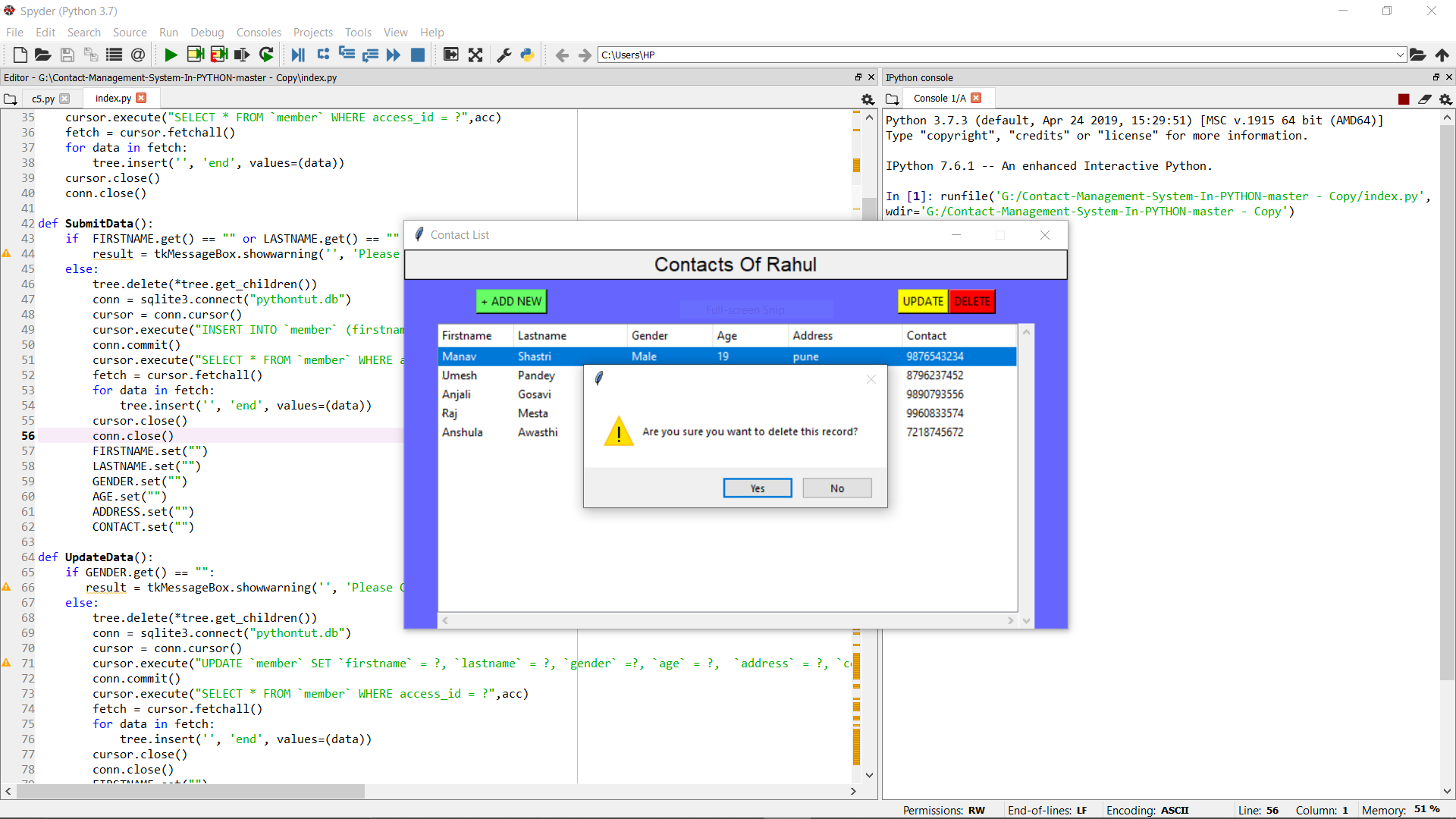
GUI(Screen shots)

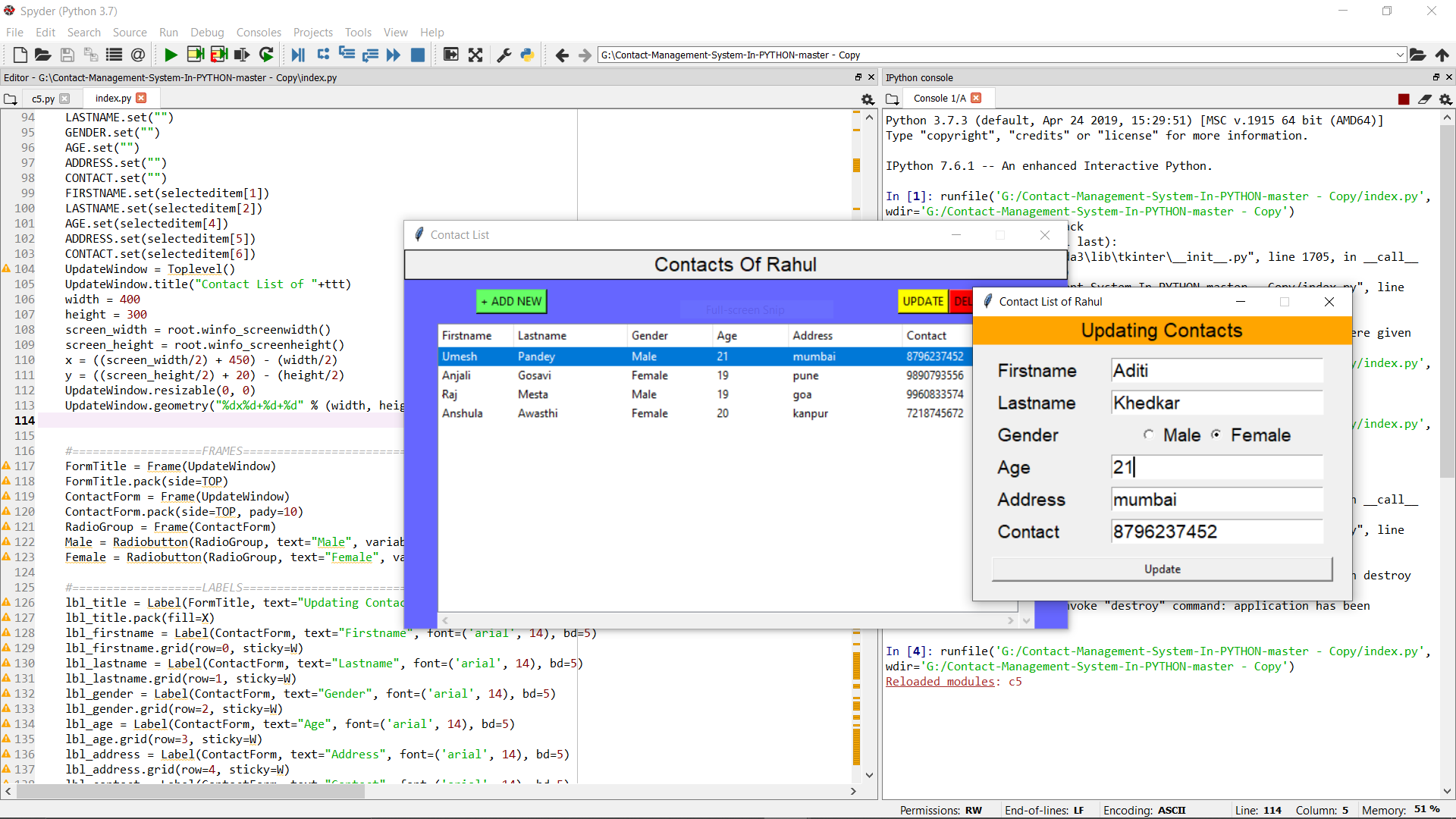


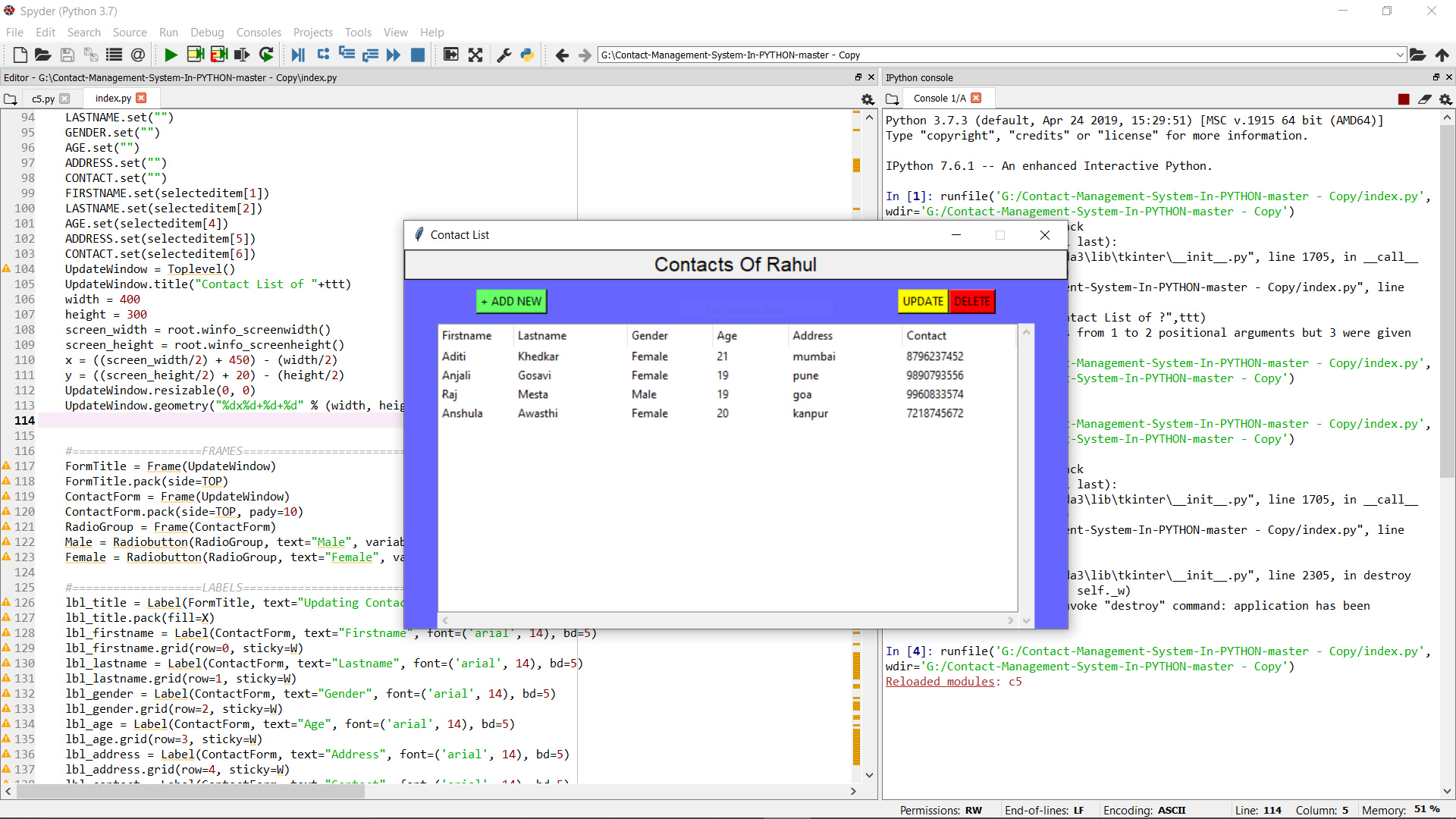


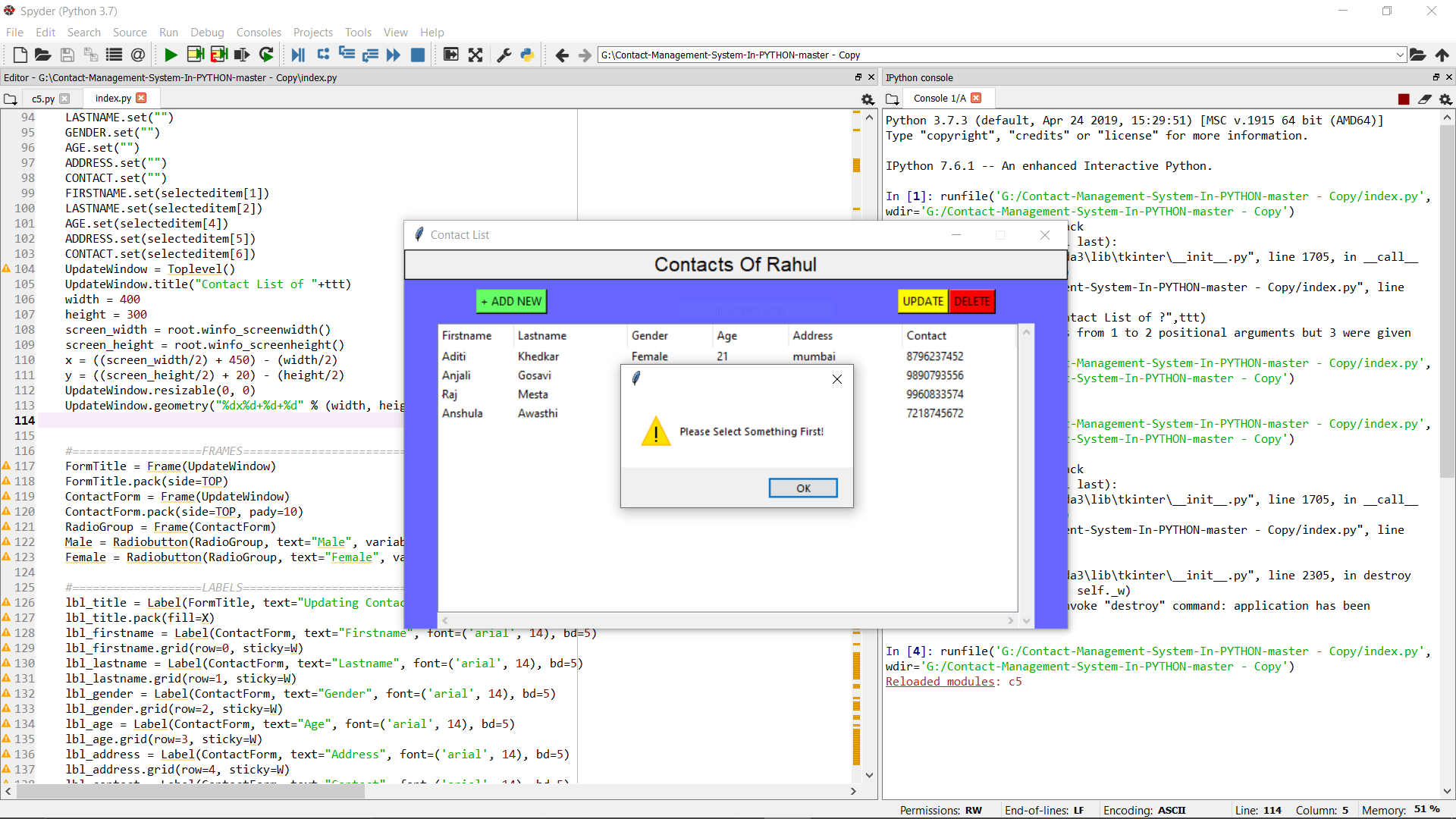


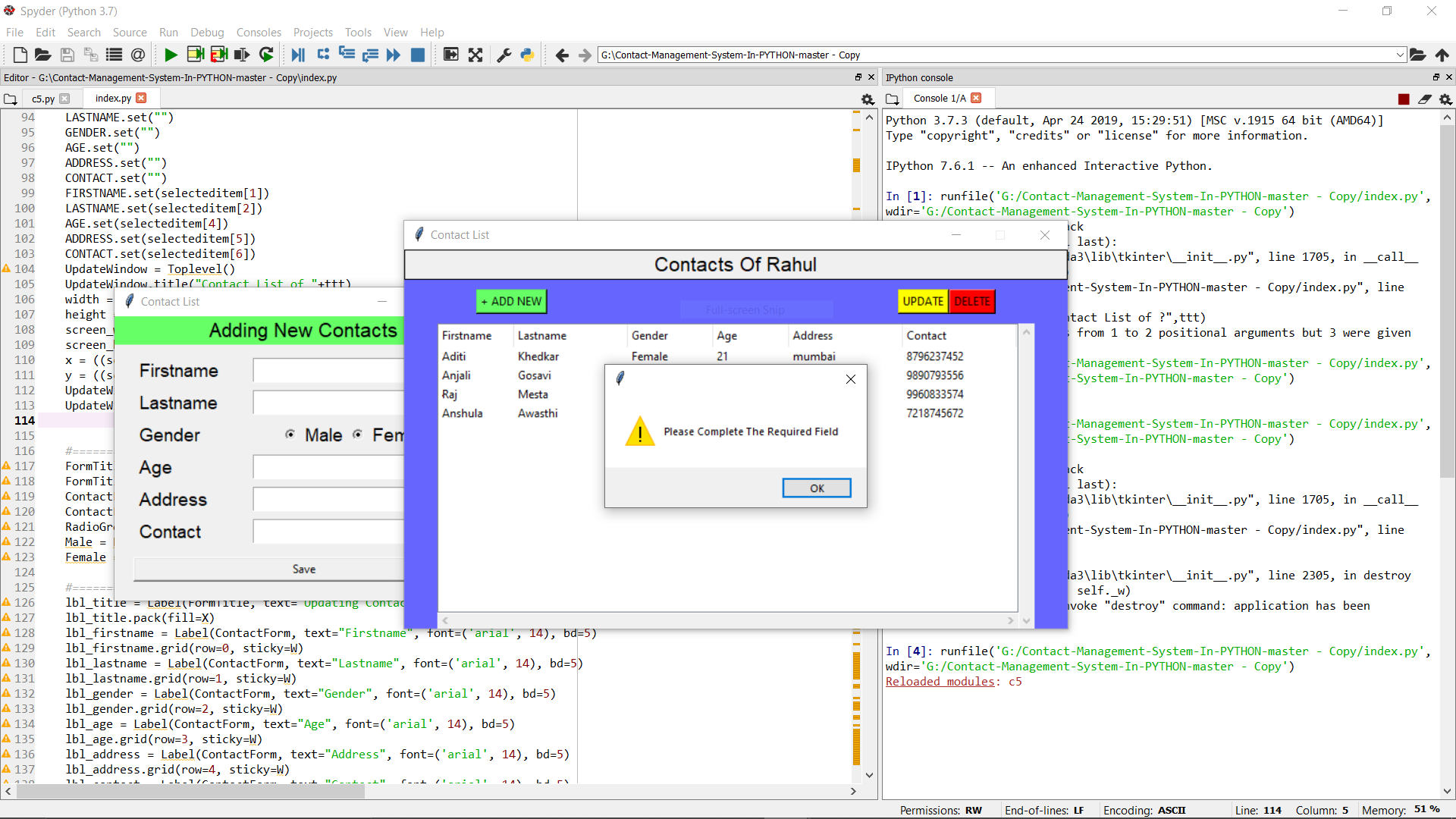












Conclusion

In this project we have created one application which is easy to access and user friendly.The application keeps a backup of the contact management data which includes their details. A contact management is a professional database who assists with the design, planning and management of a client's contacts.