Project Report - Contact Management System

1. Introduction

The Contact Management System is a simple Python console application that enables users to store, view, search, update, and delete contact information. It demonstrates core Python programming concepts such as data structures, user input handling, loops, and conditionals.

2. Objective

The objective of this project is to:

- Build a functional contact manager using Python.
- Practice basic programming concepts like lists, dictionaries, and functions.
- Develop an interactive console menu system for user interaction.

3. Features

The system provides the following functionalities:

1. Add Contact
2. View Contacts
3. Search Contact
4. Update Contact
5. Delete Contact
6. Exit

4. Technologies Used

Technologies:

- Python 3.13.5: Programming language

- IDLE: Development and testing environment

5. Code Structure

Data Structure:

- A Python list named 'contacts' holds all contact entries.

- Each contact is a dictionary with keys: Name, Phone, and Email.

Functions:

- add_contact()

- view_contacts()

- search_contact()

- update_contact()

- delete_contact()

- menu()

6. Sample Output

Add Contact:

Enter Name: abc

Enter Phone Number: 12345678

Enter Email: abc@gmail.com

Contact added successfully!

Search Contact:

Project Report - Contact Management System

Enter name to search: abc

Found: Name: abc, Phone: 12345678, Email: abc@gmail.com

7. Limitations

- Contacts are stored in memory only; data is lost when the program exits.

- No validation for duplicate contacts or format of phone/email.

- Search is exact-match only.

8. Future Enhancements

- Add file/database persistence.

- Validate user inputs.

- Implement GUI using Tkinter.

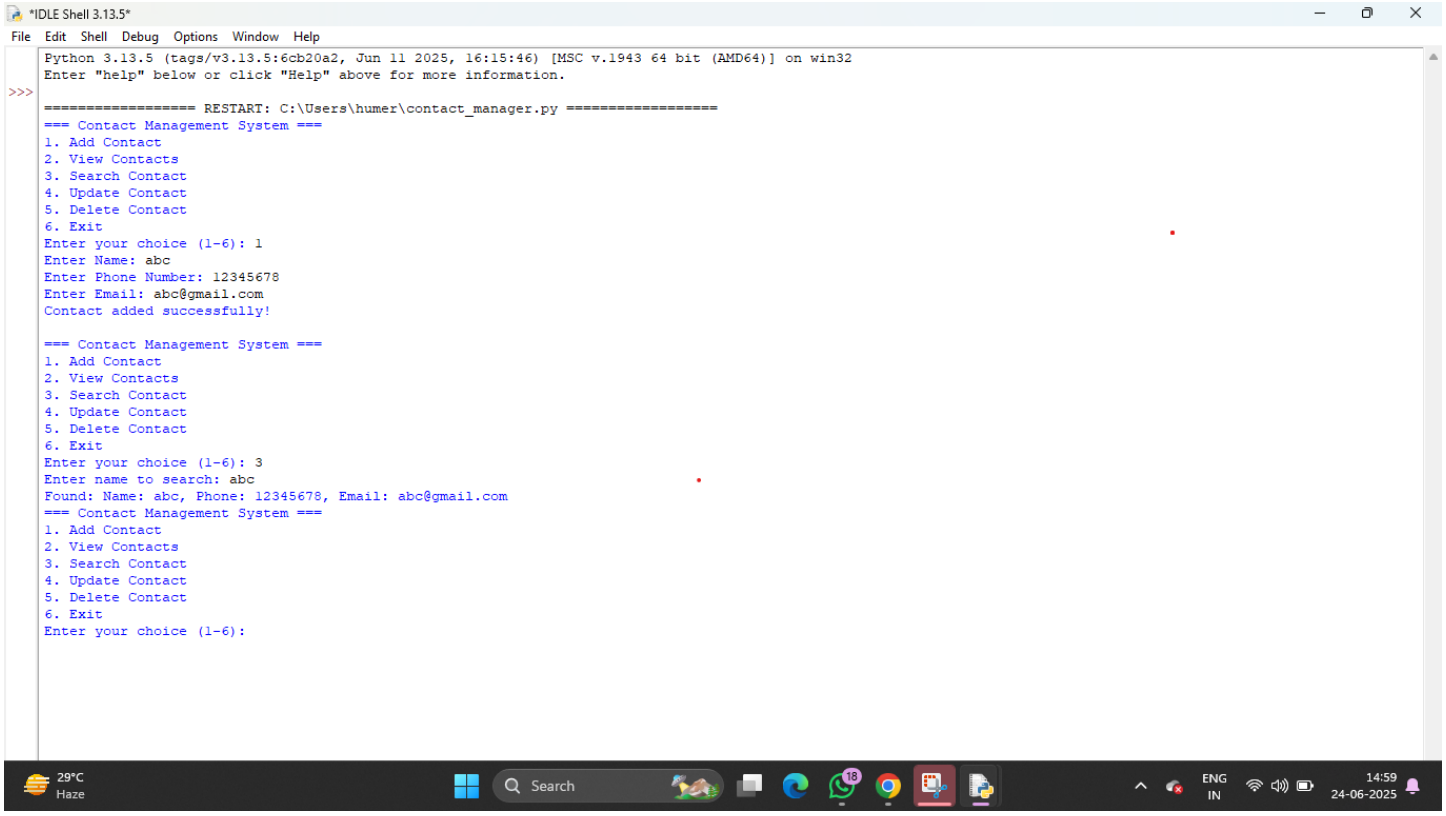
- Add advanced search and sort options.

9. Conclusion

The Contact Management System is a foundational project demonstrating structured data management in Python through a console interface. It serves as a starting point for more advanced CRUD-based applications.

10. Screenshot

Project Report - Contact Management System



Project Report - Contact Management System

11. Source Code

```
# Contact Management System - Console Version

contacts = []

def add_contact():
    name = input("Enter Name: ")
    phone = input("Enter Phone Number: ")
    email = input("Enter Email: ")
    contact = {"Name": name, "Phone": phone, "Email": email}
    contacts.append(contact)
    print("Contact added successfully!\n")

def view_contacts():
    if not contacts:
        print("No contacts found.\n")
        return
    print("\nAll Contacts:")
    for i, contact in enumerate(contacts, start=1):
                    print(f"{i}.  Name:  {contact['Name']},  Phone:  {contact['Phone']},  Email:
{contact['Email']}")
    print()

def search_contact():
    search_name = input("Enter name to search: ")
    found = False
    for contact in contacts:
        if contact['Name'].lower() == search_name.lower():
                     print(f"Found:  Name:  {contact['Name']},  Phone:  {contact['Phone']},  Email:
{contact['Email']}")
            found = True
            break
    if not found:
        print("Contact not found.\n")

def delete_contact():
    del_name = input("Enter name to delete: ")
    for contact in contacts:
        if contact['Name'].lower() == del_name.lower():
            contacts.remove(contact)
            print("Contact deleted successfully!\n")
            return
    print("Contact not found.\n")

def update_contact():
    upd_name = input("Enter name to update: ")
    for contact in contacts:
```

Project Report - Contact Management System

```
        if contact['Name'].lower() == upd_name.lower():
            contact['Phone'] = input("Enter new phone number: ")
            contact['Email'] = input("Enter new email: ")
            print("Contact updated successfully!\n")
            return
    print("Contact not found.\n")

def menu():
    while True:
        print("=== Contact Management System ===")
        print("1. Add Contact")
        print("2. View Contacts")
        print("3. Search Contact")
        print("4. Update Contact")
        print("5. Delete Contact")
        print("6. Exit")
        choice = input("Enter your choice (1-6): ")

        if choice == '1':
            add_contact()
        elif choice == '2':
            view_contacts()
        elif choice == '3':
            search_contact()
        elif choice == '4':
            update_contact()
        elif choice == '5':
            delete_contact()
        elif choice == '6':
            print("Exiting... Goodbye!")
            break
        else:
            print("Invalid choice. Please enter a number between 1 and 6.\n")

# Start the program
menu()
```