-Smart-Contact-Manager-
A Smart Contact Management System

A MINOR PROJECT I REPORT

Submitted in partial fulfillment of the

requirements for the degree of

BACHELOR OF TECHNOLGY

In

COMPUTER SCIENCE & ENGINEERING

By

NAME	*ENROLLMENT NO.*
Abhinav Tiwari	0112CS231004
Bhoomi Gupta	0112CS221041
Abhishek Tripathi	0112CS231011
Chitranshu Yadav	0112CS231147

-

Under the guidance of

Prof. Virendra

Khatarkar (Professor,

CSE)

July-Dec 2024

Department of Computer Science & Engineering

Bansal Institute of Science and Technology ,Bhopal (M.P.)

Approvedby AICTE,New Delhi&Govt.of M.P.

Affiliated to Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal (M.P.)

Bansal Institute of Science and Technology, Bhopal

Department of Computer Science & Engineering Bhopal (M.P.)

CERTIFICATE

We hereby certify that the work which is being presented in the B.Tech MinorProject-I Report entitled ***Smart Contact Manager,*** in partial fulfillment of the requirements for the award of the degree of ***Bachelor of Technology*** in ***Computer Science & Engineering*** and submitted to the Department of Computer Science & Engineering, ***Bansal institute of Science & Technology,*** Bhopal (M.P.) is an authentic record of my our work carried out during the period from July 2024 to Dec. 2024 under the supervision of ***Prof. Virendra Khatarkar*** The content presented in this project has not been submitted by us for the award of any other degree else where.

NAME	***ENROLL***	***SIGNATURE***

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

Date:

Prof. Virendra Khatarkar.	Dr. Kailash Patidar	Dr. Damodar Prasad Tiwari
Project Guide	***HOD-CSE, BIST***	***Director, BIST***

TABLE OF CONTENTS

ABSTRACT

Smart Contact Manager 2.0 (SCM 2.0) is a modern, secure, and scalable web-based application designed to efficiently manage personal and professional contacts in a digital environment. The project aims to replace traditional contact-keeping methods with an intelligent, user-friendly system that allows users to store, organize, search, and maintain their contacts with ease. Built using **Spring Boot** for the backend and modern frontend technologies, SCM 2.0 follows a full-stack development approach and adheres to industry-standard software engineering practices.

The application provides robust **user authentication and authorization**, including support for **OAuth-based login using Google and GitHub**, ensuring secure access and a seamless user experience. Each registered user has a personalized dashboard where they can perform complete **CRUD (Create, Read, Update, Delete)** operations on contacts. The system supports features such as profile images, contact photos, email and phone number storage, and advanced search functionality for quick retrieval of information.

SCM 2.0 integrates **database management** for persistent data storage and uses **Spring Data JPA** to simplify data access and manipulation. To enhance usability and visual appeal, the project incorporates **responsive UI design** with theme-based customization, including light and dark modes. Additionally, the application is designed to be **Docker-enabled**, allowing easy deployment and environment consistency across different systems.

The project demonstrates the practical implementation of key software development concepts such as MVC architecture, RESTful APIs, authentication mechanisms, and containerization. Smart Contact Manager 2.0 serves as an excellent learning platform for understanding full-stack Java web development while also offering a real-world solution for effective contact management. The system is scalable, maintainable, and suitable for future enhancements such as cloud deployment and advanced analytics.

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of my project. All that we have done is only due to such supervision and assistance and we would not forget to thank them.

We thank ***Prof. Virendra Khatarkar*** for providing us an opportunity to do the project work and giving us all support and guidance, which made us complete the project duly. We are extremely thankful to him/her for providing such a nice support and guidance, although he/she had busy schedule managing the corporate affairs.

We are thankful to and fortunate enough to get constant encouragement support and guidance from all teaching staffs CSE department which helped us in successfully completing my project work. Also,we would like to extend our sincere esteems to all staff in laboratory for their timely support.

.

LIST OF TABLES

LIST OF FIGURES

CHAPTER - 1
INTRODUCTION

CHAPTER 1
INTRODUCTION

1.1 Introduction

Smart Contact Manager 2.0 is a application designed to securely manage personal and professional contacts. Built using Spring Boot, it enables authenticated users to create, update, search, and organize contacts efficiently through a responsive interface, OAuth login, database integration, and scalable deployment support with security and usability features.

1.2 Purpose

The purpose of Smart Contact Manager 2.0 is to provide a secure, efficient, and user-friendly platform for managing personal and professional contacts using modern web technologies and authentication mechanisms..

1.3 Objectives

- To provide a secure and reliable system for storing and managing personal and professional contacts.
- To enable user authentication using traditional login and OAuth services.
- To support complete CRUD operations for efficient contact management.
- To offer a responsive and user-friendly interface with modern design features.
- To ensure scalability and easy deployment using modern tools like Docker.

CHAPTER - 2
LITERATURE SURVEY

CHAPTER 2
LITERATURE SURVEY

SURVEY

2.1 LITERATURESURVEY : EXISTING SYSTEM

Existing contact management applications generally focus on storage of contact details and lack advanced, integrated features. Most traditional systems allow users to save names and phone numbers but do not provide secure access, management, or cross-platform availability. Users often rely on multiple tools such as phone contacts, spreadsheets, email lists to manage their contacts efficiently.

Many existing systems also lack proper security and personalization. Contacts are often stored locally without authentication, making data vulnerable to loss or unauthorized access. Additionally, such applications do not provide user-specific dashboards, role-based access, or login options through trusted platforms like Google or GitHub, which limits both security and convenience.

Another major limitation is poor usability and scalability. Several contact management systems offer limited search and filtering capabilities, making it difficult to handle a large number of contacts. The absence of a modern, responsive interface and theme customization reduces user experience. Furthermore, many applications are not designed for scalable deployment or easy maintenance. Smart Contact Manager 2.0 addresses these limitations by providing a secure, user-centric, and scalable web-based solution with modern features and technologies.

CHAPTER - 3 PROBLEM DESCRIPTION

CHAPTER 3
PROBLEM DESCRIPTION

3.1 PROBLEM DESCRIPTION OVERVERVIEW

In recent years, the rapid growth of digital communication and professional networking has increased the need for efficient contact management. Individuals interact with a large number of people through emails, phone calls, and social platforms, making it difficult to organize and maintain contact information manually. The lack of a system often results in lost contacts, outdated information, and reduced productivity.

Traditional methods of contact management, such as phone address books, or spreadsheet files, are inefficient and error-prone. These methods do not provide secure access, centralized storage, or advanced search capabilities. Manual record-keeping makes it difficult to manage large volumes of contacts, update information consistently, or retrieve specific details quickly. Additionally, such data is vulnerable to accidental deletion, device loss, or unauthorized access.

Although several digital contact management applications exist, many of them suffer from significant limitations. Most available systems offer only basic contact storage features and lack proper authentication and personalization. Users often cannot access their data across devices securely, and many applications do not provide modern login mechanisms such as OAuth-based authentication.

.

Problem Identified

- Absence of a centralized and secure system for managing professional contacts
- Inefficient and error-prone traditional methods such as manual lists or basic phone storage
- Lack of secure authentication and controlled access to contact information
- No effective search, filtering, or organized representation of large contact datasets
- Limited personalization and scalability in existing contact management applications

CHAPTER - 4
SOFTWARE AND HARDWARE REQUIREMENTS

CHAPTER 4
SOFTWARE AND HARDWARE REQUIREMENTS

4.1 Software Requirements

- **Operating System:** Windows / Linux
- **Frontend:** HTML, CSS, JavaScript, Tailwind CSS, React Js
- **Backend:** Node.js, Express.js
- **Database:** MongoDB
- **Browser:** Chrome / Edge

4.2 Hardware Requirements

- **Processor:** Intel i3 or higher
- **RAM:** Minimum 4 GB
- **Storage:** 10 GB free space
- **Internet connection**

CHAPTER - 5 SOFTWARE REQUIREMENT SPECIFICATION

CHAPTER 5
SOFTWARE REQUIREMENT SPECIFICATION

5.1 Introduction

Software Requirement Specification (SRS) defines the functional and non-functional requirements of the Smart Contact Manager 2.0 system. It describes how the system should operate to meet user requirements.
Smart Contact Manager 2.0 is a web-based application designed for secure, and user-friendly management of personal and professional contacts.

5.2 Functional Requirements

- The system allows users to register, log in, and log out securely.
- Users can manage their profile details and personal information.
- Users can add, and delete details such as name, email, phone number, and profile image.
- The system provides a dashboard to view, search, and organize contacts efficiently.
- Users can view and manage their previously saved contact records securely.

.

5.3 Non-Functional Requirements

- The system should be secure, fast, and user-friendly.
- It should be available at all times.
- The system should be easy to maintain.

5.4 System Constraints

- Requires internet connection.
- Requires modern web browser.
- Requires MongoDB database.

CHAPTER - 6
SOFTWARE DESIGN

CHAPTER 6
SOFTWARE DESIGN

6.1 ER DIAGRAM

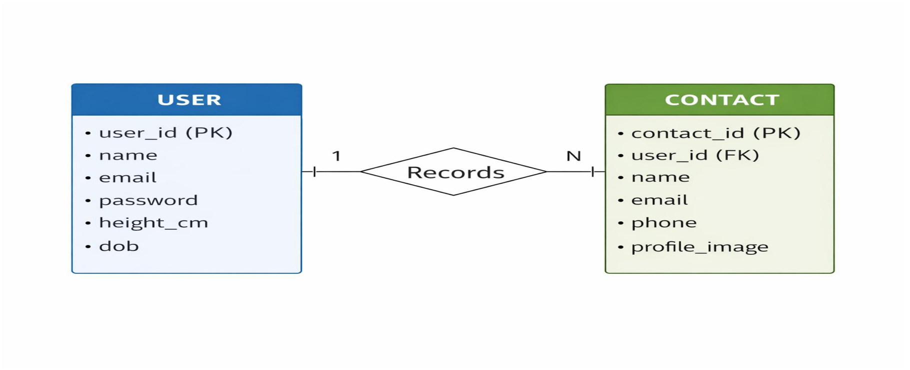
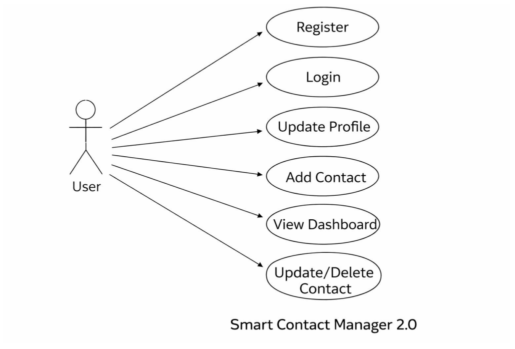


Figure 6.1.1:ER diagram

6.2 SEQUENCE DIAGRAM



6.2.1 USECASE DIAGRAM FOR ADMIN/FCULTY, STUDENT

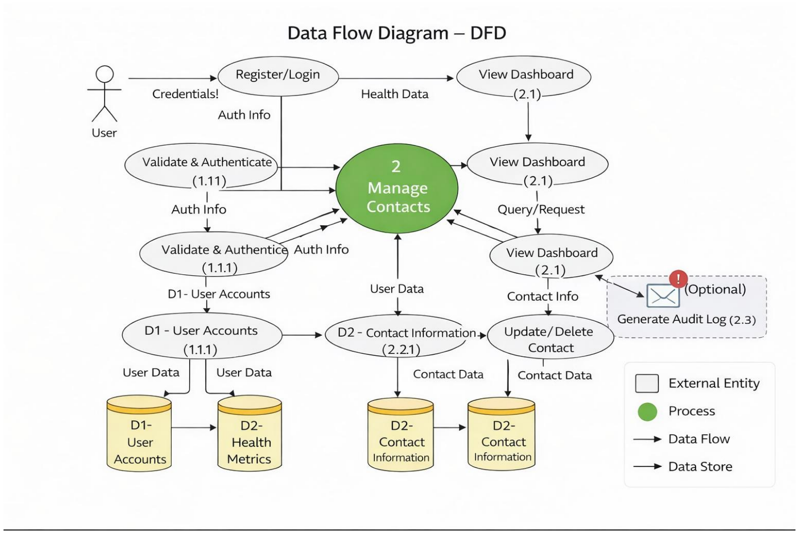
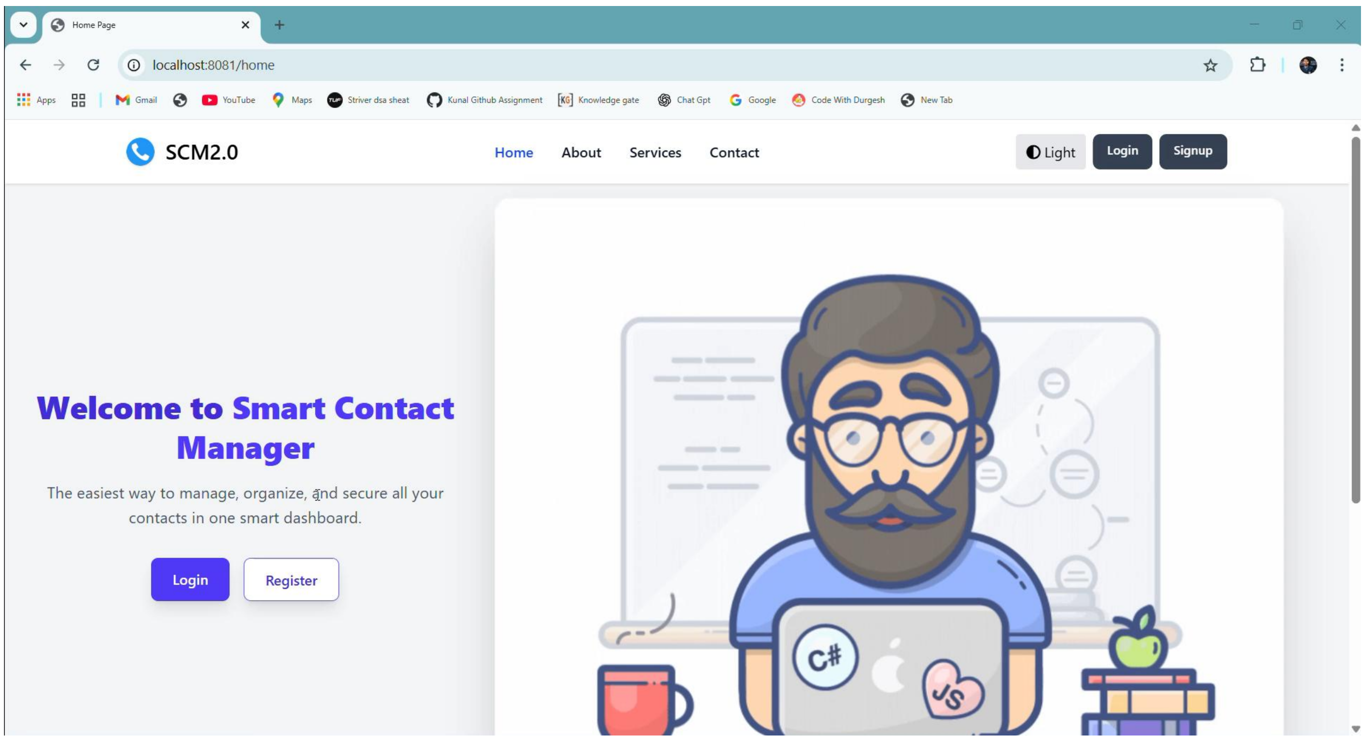


Figure 6.2: Use Case Diagram

CHAPTER - 7 OUTPUT SCREEN

CHAPTER 7
OUTPUT SCREEN

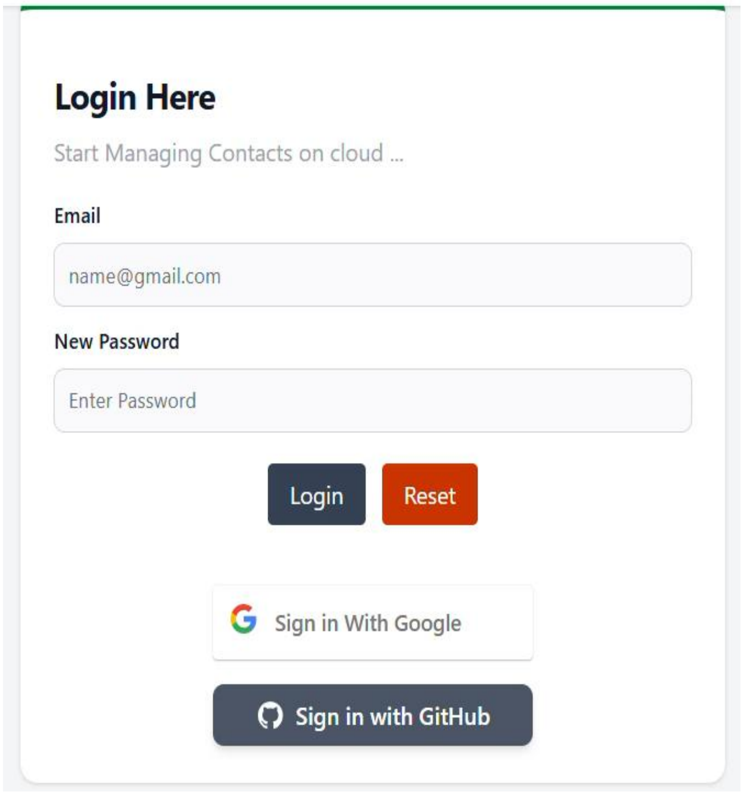
7.1 HomePage



Welcome to Smart Contact Manager

The easiest way to manage, organize, and secure all your contacts in one smart dashboard.

7.2 Result Page


Login Here

Start Managing Contacts on cloud ...

Email

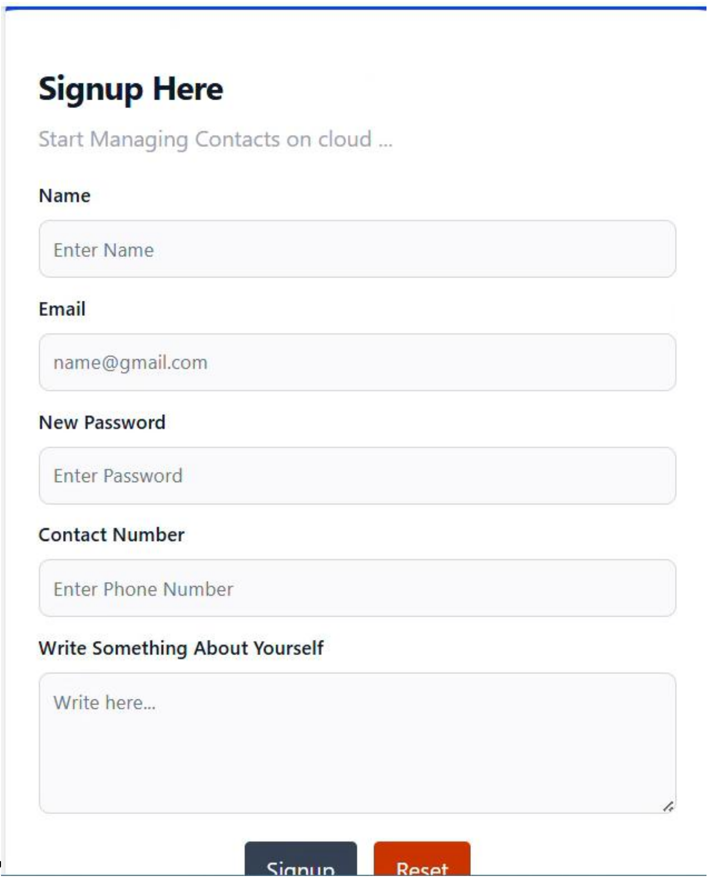
name@gmail.com

New Password

Enter Password

7.2 Contact Page

7.3 Admin Page


Signup Here

Start Managing Contacts on cloud ...

Name

Enter Name

Email

name@gmail.com

New Password

Enter Password

Contact Number

Enter Phone Number

Write Something About Yourself

Write here...

7.4 Admin Panel Page

CHAPTER - 8
DEPLOYMENT

CHAPTER 8
DEPLOYMENT

8.1 DEPLOYMENT OF WEBSITE

8.1.1 INTRODUCTION

The **Smart Contact Manager 2.0** application is currently deployed on a **local Spring Boot server** for development and testing purposes. The backend is built using **Spring Boot**, which handles user authentication, contact management operations, and communication with the database.

MongoDB is used as the backend database to securely store user information and contact details. The database runs on a local MongoDB server during development and can be managed using tools such as **MySQL Workbench** or similar database clients.

The frontend of the application is rendered using server-side templates along with static resources such as **HTML, CSS, JavaScript, React js and Tailwind CSS**, which are served by the Spring Boot application itself.

For future deployment, the application can be hosted on cloud platforms such as **AWS, Render, or Railway**. The system also supports **Docker-based deployment**, enabling easier setup and scalability. A managed cloud database service can be used to improve availability, performance, and security, allowing multiple users to access the application simultaneously from different locations.

CHAPTER - 9 CONCLUSION AND WORK

CHAPTER 9
CONCLUSION AND WORK

9.1 Conclusion

Smart Contact Manager 2.0 is a comprehensive and user-friendly contact management system that effectively addresses the need for organized and secure handling of personal and professional contacts. The system enables users to store, update, and manage essential contact details such as names, phone numbers, email addresses, and profile images in a structured manner.

By providing features such as secure authentication, OAuth-based login, efficient search and filtering, and a responsive dashboard, the application improves productivity and simplifies contact management. The use of modern web technologies such as Spring Boot, HTML, CSS, JavaScript, and database integration ensures scalability, data security, and reliable performance.

Overall, Smart Contact Manager 2.0 demonstrates how modern web technology can be effectively utilized to streamline contact organization, enhance data security, and provide a practical solution for everyday digital communication management.

9.2 Future Work

Although **Smart Contact Manager 2.0** fulfills its current objectives, several enhancements can be implemented in the future to increase its functionality and real-world applicability:

Integration with cloud-based contact synchronization to enable real-time access across multiple devices.

- Advanced search and AI-based contact suggestions to improve organization and retrieval efficiency.
- Mobile application development to provide better accessibility and portability.
- Contact sharing and collaboration features for controlled sharing with teams or family members.
- Enhanced security features, including two-factor authentication and activity monitoring.

These future enhancements can transform Smart Contact Manager 2.0 into a more intelligent, secure, and widely usable contact management platform.

REFERENCES

WEBSITES

1. https://www.w3schools.com
2. https://nodejs.org
3. https://www.mongodb.com

APPENDIX
GLOSSARY OF TERMS

JWT (JSON Web Token): A secure token used for user authentication and session management in web applications.

API (Application Programming Interface):A set of rules that allows communication between the frontend and backend of an application.

UI (User Interface): The visual part of the application through which users interact with the system.

DB (Database): A structured collection of data used to store and manage application information.

MongoDB: A NoSQL database used to store user data and health records in JSON-like format.

Node.js: A server-side JavaScript runtime used to build scalable backend applications.

Express.js: A lightweight Node.js framework used to handle routing and server logic.

Authentication: The process of verifying the identity of a user.

Authorization: The process of determining what actions a logged-in user is allowed to perform.

Dashboard: A visual interface that displays contact data, details, and insights.