

“Bulk SMS and Email Delivery System”

A Major Project Report Submitted to



**Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal
Towards Partial Fulfillment for the Award of**

**Bachelor of Technology
(Computer Science and Engineering)**

Submitted By

Aryan Dame (0827CS191054)

Anurag Chauhan (0827CS191048)

Anirudha Khode (0827CS191040)

Aditya Trivedi (0827CS191015)

Under the Supervision of

Prof. Shivshankar Rajput

Assistant Professor



**Department of Computer Science and Engineering
Acropolis Institute of Technology & Research, Indore
Jan-June 2023**

EXAMINER APPROVAL

The Project entitled ***“Bulk SMS and Email Delivery System”*** submitted by **Aryan Dame (0827CS191054), Anurag Chauhan (0827CS191048), Anirudha Khode (0827CS191040), Aditya Trivedi (0827CS191015)** has been examined and is hereby approved towards partial fulfillment for the award of ***Bachelor of Technology degree in Computer Science and Engineering*** discipline, for which it has been submitted. It is understood that by this approval the undersigned does not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approves the project only for the purpose for which it has been submitted.

(Internal Examiner)

Date:

(External Examiner)

Date:

GUIDE RECOMMENDATION

This is to certify that the work embodied in this project entitled “***Bulk SMS and Email Delivery System***” submitted by ***Aryan Dame (0827CS191054)***, ***Anurag Chauhan (0827CS191048)***, ***Anirudha Khode (0827CS191040)***, ***Aditya Trivedi (0827CS191015)*** is a satisfactory account of the bonafide work done under the supervision of ***Prof. Shivshankar Rajput***, is recommended towards partial fulfillment for the award of the Bachelor of Technology (Computer Science and Engineering) degree by Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal.

(Project Guide)

(Project Coordinator)

(Dean Academics)

STUDENTS UNDERTAKING

This is to certify that the project entitled ***“Bulk SMS and Email Delivery System”*** has been developed by us under the supervision of ***Prof. Shivshankar Rajput***. The whole responsibility of the work done in this project is ours. The sole intention of this work is only for practical learning and research.

We further declare that to the best of our knowledge; this report does not contain any part of any work which has been submitted for the award of any degree either in this University or in any other University / Deemed University without proper citation and if the same work is found, then we are liable for explanation to this.

Aryan Dame (0827CS191054)

Anurag Chauhan (0827CS191048)

Anirudha Khode (0827CS191040)

Aditya Trivedi (0827CS191015)

Acknowledgment

We thank the almighty Lord for giving us the strength and courage to sail out through the tough and reach shore safely.

There are several people without whom this project's work would not have been feasible. Their high academic standards and personal integrity provided us with continuous guidance and support.

We owe a debt of sincere gratitude, and a deep sense of reverence, and respect to our guide and mentor **Prof. Shivshankar Rajput** for their motivation, sagacious guidance, constant encouragement, vigilant supervision, and valuable critical appreciation throughout this project work, which helped us to complete the project on time.

We express profound gratitude and heartfelt thanks to **Dr. Kamal Kumar Sethi**, HOD CSE, AITR Indore for his support, suggestion, and inspiration for carrying out this project. I am very much thankful to other faculty and staff members of the CSE Dept, AITR, Indore for providing me with all support, help, and advice during the project. We would fail in our duty if we do not acknowledge the support and guidance from **Dr. S.C. Sharma**, Director, AITR, Indore whenever needed.

We are grateful to **our parents** and **family members** who have always loved and supported us unconditionally. To all of them, we want to say, "Thank you", for being the best family that one could ever have and without whom none of this would have been possible.

Aryan Dame (0827CS191054)

Anurag Chauhan (0827CS191048)

Anirudha Khode (0827CS191040)

Aditya Trivedi (0827CS191015)

Executive Summary

Bulk SMS and Email Delivery System

This project is submitted to Rajiv Gandhi Proudhyogiki Vishwavidhyalaya, Bhopal (MP), India for the fulfillment of the Bachelor of Engineering in Information Technology branch under the sagacious guidance and vigilant supervision of ***Prof. Shivshankar Rajput***.

The project is based on Python MySQL connectivity. Python has a lot of GUI frameworks, but Tkinter is the only framework that's built into the Python standard library. Tkinter has several strengths. It's cross-platform, so the same code works on Windows, macOS, and Linux. Visual elements are rendered using native operating system elements, so applications built with Tkinter look like they belong on the platform where they're run. MySQL Connector/Python enables Python programs to access MySQL databases, using an API that is compliant with the Python Database API Specification v2.0 (PEP 249). It is written in pure Python and does not have any dependencies except for the Python Standard Library.

Keywords: Python, MySQL, MySQL Connector, API, Tkinter

*“Where the vision is one
year, cultivate flowers;
Where the vision is ten years,
cultivate trees;
Where the vision is eternity,
cultivate people.”*

- Oriental Saying

List of Figures

Figure 1: Registration Flow Chart.....	13
Figure 2: Login Flow Chart.....	14
Figure 3: Message Flow Chart.....	15
Figure 4: Use Case Diagram.....	16
Figure 5: ER Diagram.....	17
Figure 6.1: Validation while Login Process	22
Figure 6.2: Successful Login Demonstration	22
Figure 7: Registration check.....	23
Figure 8.1: Password confirmation through JavaScript	24
Figure 8.2: New Registration (Successful).....	24
Figure 9: Splash Screen.....	25
Figure 10: Login Screen.....	25
Figure 11: Successful Login Popup	26
Figure 12: Registration Page	26
Figure 13: Successful Registration	26
Figure 14: Selection Menu	27
Figure 15: Email Client Page.....	28
Figure 16: Demonstration of using bulk Email service.....	28
Figure 17: SMS Client Page	29
Figure 18: Demonstration of using bulk SMS service	29
Figure 19: Project Table Description.....	30
Figure 20: Email_info Table Description.....	30

Figure 21: SMS Table Description.....	30
Figure 22: Entries in Project Table (For testing Purposes).....	30
Figure 23: Entries in email_info Table (For testing Purposes)	31
Figure 24: Entries in SMS Table (For testing Purposes).....	31

List of Tables

Table 1: Test Case 1 22

Table 2: Test Case 2 23

Table 3: Test Case 3 24

List of Abbreviations

Abbr1: SMS- Short Message Service

Abbr2: API – Application Programming Interface

Abbr3: SMTP- Simple Mail Transfer Protocol

Abbr4: RAM- Random Access Memory

Abbr5: SQL- Structured Query Language

Abbr6: GUI- Graphical User Interface

Abbr7: DND- Do Not Disturb

Abbr8: OTP- One Time Password

Abbr9: SMSC- Short Message Service Center

Abbr10: MMS- Multimedia Messaging Service

Abbr11: HCL- Hardware Compatibility List

Abbr12: GNU – General User License

Abbr13: FLOSS- Free/Libre and Open-Source Software

Abbr14: CGI- Computer-generated imagery

Abbr15: XML- Extensible Markup Language

Abbr16: FTP- File Transfer Protocol

Abbr17: REST- Representational State Transfer

Abbr18: HTTP- Hyper Text Transfer Protocol

Abbr19: TCP- Transmission Control Protocol

Abbr20: CRM- Customer Relationship Management

Table of Contents

CHAPTER 1. INTRODUCTION	1
1.1 Overview	1
1.2 Introduction	1
1.3 Background and Motivation	1
1.4 Types of SMS	2
1.3.1 Promotional SMS Service.....	2
1.3.2 Transactional SMS Service.....	3
1.5 Objectives	3
1.6 Scope of the Project	3
1.7 Project Description.....	4
1.8 Applications.....	4
1.9 Team Organization.....	7
1.10 Report Structure	8
CHAPTER 2. REVIEW OF LITERATURE	9
2.1 Introduction	9
2.2 Short Message Service (SMS) and Email.....	9
2.3 Bulk SMS and Email.....	9
2.4 Review	10
2.5 Survey of Existing Systems	10
2.5.1 Plivo.....	10
2.5.2 Atomic SMS Sender.....	11
2.5.3 My SMS Mantra.....	12

CHAPTER 3.	METHODOLOGY OF THE PROJECT WORK.....	13
3.1	Basic Planning	13
3.2	Information	13
3.3	Requirement Specification	14
	3.3.1 Hardware Specification.....	14
	3.3.2 Software Specification.....	14
3.4	Feasibility Study	15
	3.4.1 Technical	15
	3.4.2 Economical.....	15
	3.4.3 Operational.....	15
3.5	Diagrams and Flowcharts.....	16
	3.5.1 Flowcharts	16
	3.5.2 Use Case UML Diagram.....	19
	3.5.3 ER Diagram.....	20
CHAPTER 4.	IMPLEMENTATION.....	21
4.1	Tools Used	21
	4.1.1 Python Tkinter.....	21
	4.1.2 MySQL.....	21
4.2	Language Used	22
4.3	Modules Used	23
	4.3.1 Twilio.....	23
	4.3.2 SMTP LIB.....	23
4.4	Testing.....	24
	4.4.1 Strategy Used.....	24
	4.4.2 Test Case and Analysis.....	25
4.5	Application Screenshots	28
4.6	Database Screenshots	33

CHAPTER 5.	CONCLUSION.....	35
5.1	Conclusion	35
5.2	Advantages	36
5.3	Limitations of Project Work	37
5.4	Future Scope	38
CHAPTER 6.	BIBLIOGRAPHY.....	40
	GUIDE INTERACTION SHEET.....	42
CHAPTER 7.	SOURCE CODE.....	43

Chapter 1. Introduction

1.1 Overview

This project focuses on understanding the concept of bulk email and message sending and the process of sending a large number of emails and messages to organizations as a means of informing them. It is noted that bulk email and message sending are commonly used in large commercial organizations for sending numerous emails and messages. Logon Utility offers a bulk email and SMS gateway that enables the quick transmission of messages to mobile devices. In other words, it provides an efficient tool for sending bulk emails and SMS through the internet to handheld devices.

A bulk SMS or email service that can quickly convey important information to a large audience is necessary. This information could include important alerts, warnings, and news that must be delivered to all citizens of the country at once.

1.2 Introduction of the Project

Information sharing is a major factor for development of the country. It is a time taking process to share information with all the citizens of the country through media and newspapers. In this modern age, almost everyone has a mobile phone in their hand. We can use this opportunity to share information directly with them through SMS or email. This will save time and money as information can be shared with a larger community within a short time duration. Also, such a service can be used to repeatedly send SMS in case of emergencies.

1.3 Background and Motivation

Bulk SMS and email sending is a project that is designed to provide businesses and organizations with a means of communicating with their customers, employees, and stakeholders on a large scale. The motivation behind such a project has stemmed from several factors, including:

Increased Efficiency: The ability to send a large number of SMS or emails in one go can greatly increase the efficiency of communication for businesses and organizations. This is especially true when compared to sending individual messages to each recipient.

Cost-effectiveness: Sending bulk SMS and emails can be much more cost effective than traditional methods of communication, such as direct mail or telemarketing.

Targeted Marketing: The ability to segment and target specific groups of people with specific messages makes bulk SMS and email sending a valuable tool for marketing and advertising campaigns.

Time-saving: Automating the process of sending bulk SMS and emails can save businesses and organizations a significant amount of time and resources, which can then be redirected toward other areas of their operations.

Improved Customer Experience: By sending timely, relevant, and personalized messages to customers, businesses and organizations can improve their customer experience and build stronger relationships with their audience.

Overall, the motivation behind designing a project of bulk SMS and email sending is to provide businesses and organizations with an effective, efficient, and cost-effective way of communicating with their stakeholders on a large scale.

1.4 Types of Bulk SMS

1.4.1 Promotional SMS service:

Promotional SMS uses brands to register the non-registered mobile numbers to boost their reach and sales. These are SMSs that are sent to promote a product or a service. This category includes any sales & marketing messages which may or may not be solicited by the recipient. Promotional SMS can be sent only between 9 AM to 9 PM and only to numbers that are non-DND numbers. Bulk SMS is Widely used for promotional services. It is known as a crucial marketing tool. You can come up with a new offer for your store and inform your customers about the same in just a matter of minutes. SMS service is very useful for updating your customers with complete information related to your business. You could use this platform to entice your customers and get them to do more business with you. This is one of the biggest utilizations of sending bulk SMS. The benefits of

promotional SMS are that it is cost-effective and businesses can get quick responses from their customers.

Examples Of Promotional SMS: Dear Travel Agents, NOW Get Flat 10% Off on Flight Tickets, same has been updated on the portal. Login now and start booking!

1.4.2 Transactional SMS Service:

Transactional SMS is informative SMS. People like to receive these SMS, like banking, school, colleges, alerts, bill payment reminders, OTP, etc. These SMS get delivered to all DND and Non-DND numbers and it works 24 hours. Messages can be sent 24/7 from your 6-character Sender ID. These are messages which are sent to the customer to pass on information necessary for using the product or service.

Examples of Transactional SMS:

- 1) A message sent by a bank to an account holder regarding his/her available account balance.
- 2) University sends messages to the parents of their students about their marks.

1.5 Problem Statement and Objective

The requirement is for a mobile or web-based application that can send bulk emails and SMS (approximately 10 million) to users simultaneously using SMTP or other technology and notify them through Artificial Intelligence techniques. The app should also have features such as log generation, notifications for failed delivery of SMS and/or email, the ability to import contact details from Excel sheets, and the option to customize emails with the recipient's name.

1.6 Scope of Project

Bulk SMS is utilized across various industries to enhance communication efficiency and resolve business problems. There is a significant demand for bulk SMS in India, a developing country. Utilizing technology, companies can create a strong communication channel to bridge the gap between brands and customers. Bulk SMS services provide a direct and immediate means of reaching people, resulting in increased sales and two-way communication. Unlike TV and print advertising, the delivery of the message cannot be controlled and is dependent on

various factors. However, with SMS marketing, the potential or existing customer can receive the message at a specific time and day of your choice, making it a more beneficial approach.

1.7 Project Description

The end product will be an application developed using Python Tkinter for the front end and MySQL for the back end. This project serves as a valuable study of Python and the interaction between SMS and computer programs. The project report encompasses the basic concept of receiving and sending bulk SMS through a personal computer. It aims to assist busy individuals by saving time. The user interface is user-friendly for adding new contact information, such as name and number, and storing it. A MySQL database has been implemented to store the information of the contacts.

1.8 Applications

1.8.1 Airlines:

Airlines companies send bulk SMS for several special offers on flight tickets. In India, the aviation sector has witnessed drastic growth over the past few years as the least expensive tickets were introduced, as a result more per-middle-class families started using flights to commute rather than trains.

In India, the aviation industry is striving hard to offer the best airfare to all possible consumers across business and economy classes. Improving customer satisfaction is the main motive of any airline business. So, to achieve their customer satisfaction goal, airline companies are adopting SMS mobile technology to increase their customer satisfaction. With the help of bulk SMS services, airline companies can update their passenger's data and provide accurate information to them about flight details, several offers and seasonal discounts.

1.8.2 Automobile sector:

Almost every automobile has established its own identity to increase sales and customer service. These automobile dealers opt for varied modes of communication to keep healthy sales figures and to improve customer relationship in the form of better sales figure. One of the effective media is to use bulk SMS marketing which will help them to target a large group of audience and

increase their market boundaries, as bulk SMS marketing is a two-way interactive service it benefits both the dealer and the customer by providing better sales and better after-sales services.

1.8.3 Banking and financial service sector:

Text messaging your customers in the banking and financial sector will help your business to gain a competitive advantage in the market because, with the help of this tool, you can keep the customers updated and connected with the services. SMS marketing can be used by banks to notify customers of their financial status, any bank transactions and alert them about any upcoming schemes or offers. This simple communication will make your banking life easier and safer.

1.8.4 Courier and logistics:

SMS messaging means bridging a gap in transaction processes at every level, which is very important for the smooth flow of business transactions. Bulk SMS API will surely create new opportunities for delivery businesses.

Courier service uses SMS messaging for dealing transactions with their customers. This process can be made more effective by utilizing bulk SMS in the business as it will give a professional touch to the business and customer can get an immediate response of the status of their courier.

1.8.5 E-commerce services:

Bulk SMS in the online business portals comes in handy to keep the customer updated at a lower operating cost because this works with the help of software that sends predefined texts to a large group of people regarding any sale alerts or password reset alerts or various types of business promotional offers. This is beneficial to the customer in the sense of customer satisfaction and also for the business in the sense of business growth.

1.8.6 Educational sector:

Bulk SMS is regarded as an instant alert method because everyone can't be updated via email because of certain network problems neither everyone can be updated via calls. After all, it will be time-consuming as well as non-economical. Bulk messages, on the other hand, are quite easy to be sent and read because of no

network errors. As parents have a busy schedule, so it's not possible to call every single parent regarding any event or program. Thus, for any last-minute alert, bulk texting is the only solution.

1.8.7 Tours & Travel:

Bulk SMS service is playing a major role in the tourism business, especially in several specific countries. Traveling is the latest trend in the world and everyone likes to explore new places but they run out of money and that is the place where the tourism business sets in with their latest offers and schemes that ring the notification bell now and then and keeps you updated with all the discounts being put up by several hotels and restaurants or discounts in flight bookings and several other reasons to travel more. This information is needed to be sent to a large group of people every single day. This monotonous extensive task can only be completed with the help of bulk SMS.

1.8.8 Marketing for Real estate Business:

Bulk SMS marketing is quite good when it comes to handling the advertisements of any real estate business because the basic call for any real estate business handler is to display as many houses and plots as possible. Every broker would charge high fees to show the plot to customers. This task can be handled much economically by a bulk SMS tool because it can send links to customers regarding new plots based on the information filled in by them, thus lowering down the human effort and increasing the technical advancement in the market.

1.8.9 Insurance sector:

Bulk SMS Services is showing its capabilities in the insurance sector ensuring speedy deliveries of messages and timely feedback from the customer with the help of links. There is a huge advantage of using this as it is cost-effective and most importantly it is seen that SMS has a chance of 65% more feedback than emails and calls. Thus, the insurance sector can ensure speedy feedback from their customers and easier renewals in a few clicks via the internet.

1.8.10 Media sector:

The media sector has been the ever-growing sector in the market since its inception in India. As this industry has flourished it has adapted to technological changes. Now bulk SMS is being used as an important tool in this sector to conduct polls and send various offers by several service providers or to spread any thought across public for major participation. Thus, bulk messaging tool is helping the media industry to get closer to the customer and be more responsive to the customer.

1.9 Team Organization

Aryan Dame: Along with doing a preliminary investigation and understanding the limitations of the current system, he studied the topic and its scope and surveyed various research papers related to bulk messaging and the technology that is to be used. For the implementation of the project, he collected the data and designed the tool that fulfills the purpose of project. Implementation logic for the project objective and coding of internal functionalities is also done by him. He worked on creating a database for storing the contacts.

Anirudha Khode: He took care of the presentation by researching the various other bulk email and SMS services that already exist. By this, he concluded the basic features that were lacking in the old systems and needed to be covered in this project. He helped in the front-end development part of the project. Along with it, he also helped in the testing of the project.

Aditya Trivedi: He did the complete research work and helped in the back-end development of the project. He helped his teammates connect the developed system with the database in MySQL. Also, he helped in the later part of the documentation.

Anurag Chauhan: He was responsible for searching for the journal where we can publish the research paper based on this project. Also, he created the UML diagrams of the project, by understanding the complete functioning of the project.

In all, all of us worked hard to implement the solution proposed and also took care of the documentation part, as it is the most crucial part of the project, without

which the end users of the system will not be able to use the system for solving the purpose.

1.10 Report Structure

Chapter 1: Introduction- The chapter describes the overview, problem statement elaboration, objectives, and scope of the project, followed by a brief description of the project, and its applications. Further, the chapter gives the details of the people involved and their contribution to the development of the project which is then subsequently ending with a report outline.

Chapter 2: Review of Literature- explores the work done in the area of the Project undertaken. It gives a review of the proposed system along with the explanation of keywords and also discusses the limitations of the existing system and highlights the issues and challenges of the project area. The chapter finally ends up with the main features that should be inculcated in the present project work.

Chapter 3: Methodology of the Project Work - starts with the project proposal based on the requirements identified, followed by the benefits of the project. The chapter also illustrates the software engineering paradigm used along with different design representations. The chapter also includes block diagrams and details of the major modules of the project. It also gives details of the different deployment requirements for the developed project.

Chapter 4: Implementation - includes the details of different Tools/ Programming Languages used in developing the Project. The chapter also includes the different user interfaces designed for the project along with their functionality. The chapter ends with an evaluation of the project on different parameters like accuracy and efficiency.

Chapter 5: Conclusion - Concludes with objective-wise analysis of results and limitations of present work which is then followed by suggestions and recommendations for further improvement. It also describes the advantages of the system proposed and shows its significance in solving real-world issues.

Chapter 2. Review of Literature

2.1 Introduction

A review of the literature for a bulk email and SMS-sending project would likely cover several key topics, including the benefits and challenges of these communication methods, the technical requirements and considerations for implementation, and best practices for designing and executing successful project.

We understand the bulk message sender and how to send a large number of messages to any organization for information alerts. Textlocal says that the bulk message sender is most useful in big commercial agencies for sending a large number of messages. Logon Utility has a Bulk SMS gateway that pushes the simple message to mobile phones.

2.2 Short Message Service (SMS) and Email

SMS stands for Short Message Service. It is a technology that enables the sending and receiving of messages between mobile phones. SMS first appeared in Europe in 1992. One SMS can contain at most 140 bytes of data, so one SMS message can contain up to 160 characters. Once a message is sent, it is received by an SMSC (Short Message Service Center) which must then get it to the appropriate mobile device. Electronic mail, commonly shortened to "email," is a communication method that uses electronic devices to deliver messages across computer networks. "Email" refers to both the delivery system and individual messages that are sent and received.

2.3 Bulk SMS and Email

Bulk SMS and Email service is very popular way to send a bulk number of messages and emails in the organization & used in Marketing and Banking. It is often used by Big & small business owners. The e-commerce industry, media companies, and Banks for Alerts, Marketing, and Fraud control alerts. It is also commonly used between staff of a company or bank and their clients. One

advantage of Bulk Messaging is that it delivers your message directly to mobile handsets anywhere around World.

2.4 Review:

In the digital world, finding the best bulk SMS service provider can be a challenging task with so many options available on the internet. Logon Utility, an IT company, offers a solution to this problem. With a robust SMS reseller network of over 1200 providers and serving over 15000 clients in India, Logon Utility has proven its success.

Bulk emails and SMSs are more effective than traditional forms of advertising, such as television and print, because they reach the audience directly. A message on a phone can be easily referred to later, making it more likely to be remembered compared to ignored marketing calls. Additionally, bulk email and SMS reseller providers can reach thousands of people at a lower cost as compared to conventional marketing methods. While traditional advertising cannot guarantee to reach the intended target audience, bulk SMSs provide a deeper reach into the market without incurring high costs.

2.5 Study of Existing Systems

2.5.1 Plivo

2.5.1.1 Description: Plivo's cloud communication platform offers a simple, fast, and scalable way for companies to modernize their business communications. Plivo's API platform lets businesses quickly incorporate voice and text messaging capabilities (both SMS and MMS) into their applications, and it offers SIP truncating and phone number rental services.

2.5.1.2 Advantages:

- Ease Of Integration
- API Call Speed
- Communication Varieties
- Documentation Available

2.5.1.3 Disadvantages: It is a bit expensive and majorly meant for business purposes.

2.5.1.4 Reference link: <https://www.plivo.com>

2.5.2 Atomic SMS Sender

2.5.2.1 Description: This is an online application that allows users to send bulk SMS from their phones or computers. The application can be used to send SMS to over 200 countries and supports over 700 mobile operators.

2.5.2.2 Advantages:

- Easy contact management
- Can schedule messages at anytime
- Live online training and documentation are readily available
- Offers unsubscribe feature for the audience to make them feel privileged

2.5.2.3 Disadvantages:

- 2-way messaging is not possible
- MMS and mobile coupons are also not possible
- Sometimes SMSs don't get delivered on time

2.5.2.4 Gaps Identified: Atomic SMS sender uses demand-based pricing. Their pricing is dependent on the monthly traffic of SMS sent. They offer additional discounts in case there is a large SMS volume to be sent and ten free SMS for the user to their systems.

2.5.2.5 Reference Link: <https://sourceforge.net/software/product/Atomic-SMS-Sender/>

2.5.3 My SMS Mantra

2.5.3.1 Description: My SMS Mantra is a free software for sending bulk SMS that enables the user to send SMS online using an Excel plug-in. All you need is to download and install the Excel plug-in and then stay connected to your clients/customers via SMS.

2.5.3.2 Advantages:

- Seamlessly interfaces with other applications through its API
- Has free technical support and easy setup
- Allows 2-way messaging, thanks to its Mantra Communicator

2.5.3.3 Disadvantages:

- The word limit for SMS messages is too short
- Might lose data put in the software if you face network issues

2.5.3.4 Gaps Identified: The pricing depends on the kind of SMS to be sent – whether it is a transactional or a promotional SMS.

2.5.3.5 Reference Link: <https://www.mysmsmantra.com>

Chapter 3: Methodology of the Project Work

3.1 Basic Planning

First, we will create a splash screen for the application followed by a login page. The login page will also have the option for registration. Next, we will code the registration page for the new users. It will ask for all the necessary basic information for registration.

After logging in, the user can choose from two options: SMS or Email

- SMS

The SMS screen will have a Browse option to attach the contact number of all receivers. Next, it will contain the message box followed up by a “Send” button.

- Email

The email screen will have the Browse option to attach the email addresses of all the receivers. Next, it will contain the Subject box and Message box followed by a “Send” button.

Next, the implementation of the main code will be done followed by a confirmation screen showing if the message was sent or not.

3.2 Information

A bulk email and SMS Gateway service may be a software application or web-based application that is used for purpose of sending emails and messages to several people from the computer via a mobile network. Its bulk email and SMS service allow one email and SMS to be sent to a large number of people or a great number of emails and messages with different contents to be delivered to various people. Large companies, Offices, colleges, and, corporations send Email and messages in a bulk gateway for the sake of efficiency and to save time and money. Bulk Emails and SMS service supports sending quick and short emails to many users. The amount may vary from hundreds to thousands. The identical message is shipped to all or any individuals contained in a list. This service is usually used for sales promotion and for advertising services and products. As an example,

informing people about a future event, a bit like a concert or even the launching of a new gadget. However, generally, these messages are sent to consumers who volunteered to be among the list of clients of any organization or business.

3.3 Requirement Specifications

To be used efficiently, all computer software needs certain hardware components and the software resources to be present on a computer. These pre-requirements are known as system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements - minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time.

3.3.1 Hardware Specification: The most common set of requirements defined by any operating system for designing a software application is the physical computer resources also known as a hardware requirement list that is often accompanied by a hardware compatibility list (HCL). Especially in the case of operating systems, HCL lists tested compatibility of hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

Hardware Requirements:

- Processor: Intel core i3
- RAM: 2GBs
- Hard Disk: 80 GBs (Memory Consumption: 5 MB while in use)

3.3.2 Software Specification: Software specification deals with defining software resource requirements and pre-requirements that need to be installed on a computer to provide optimal functioning of an application. These pre-requirements are generally not included in the software installation package and are needed to be installed separately before the software is installed.

Software Requirements:

- Operating System: Windows 7 (or latest)
- Front-End: Python Tkinter (Python version 3.10.10)
- Back-End: MySQL (version 8)

3.4 Feasibility Study

A feasibility study is an analysis of how successfully a system can be implemented, accounting for factors that affect it, such as economic, technical, and operational factors to determine its potential positive and negative outcomes before investing a considerable amount of time and money into it.

3.4.1 Technical

For this project, the technical requirements are basic. All that is needed for building an application used for sending bulk SMS and Emails can be easily configured on a computer. The framework that is used here is Tkinter. For storage purposes and maintaining the stored information, MySQL is used. The project is developed with minimal pre-requirements and pre-installations.

3.4.2 Economical

This project does not require any extra cost as it can be executed on any local machine with a decent configuration. The cost of designing as well as using this application is negligible. Hence, it is highly economical.

3.4.3 Operational

The main motto of our system is to reduce the manual efforts of sending the same SMS or Email again by making it possible in a single click.

The system can do that quickly and properly. This proves the system to be operationally feasible.²

3.5 Diagrams/Flow Charts:

3.5.1 Flowcharts

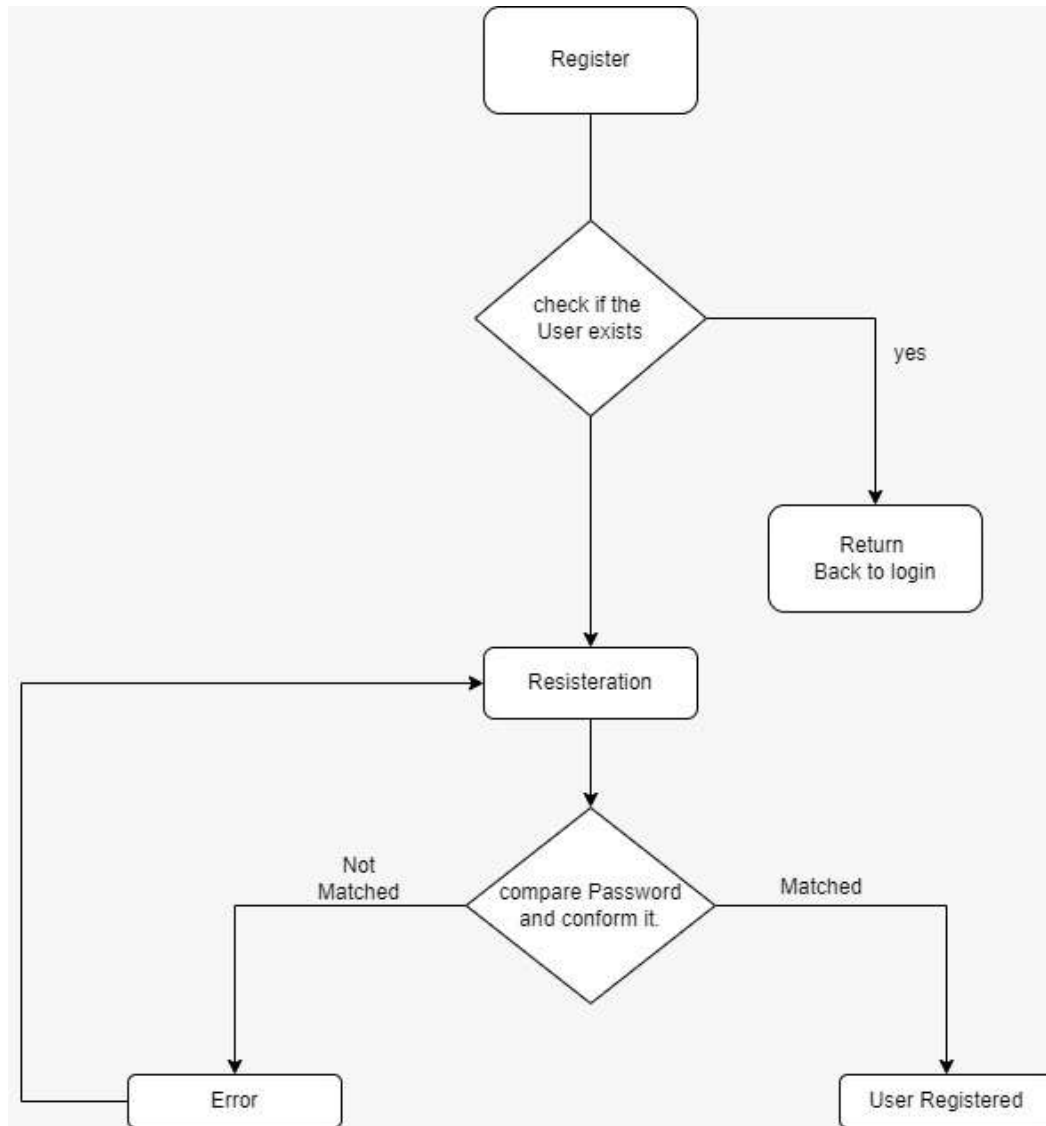
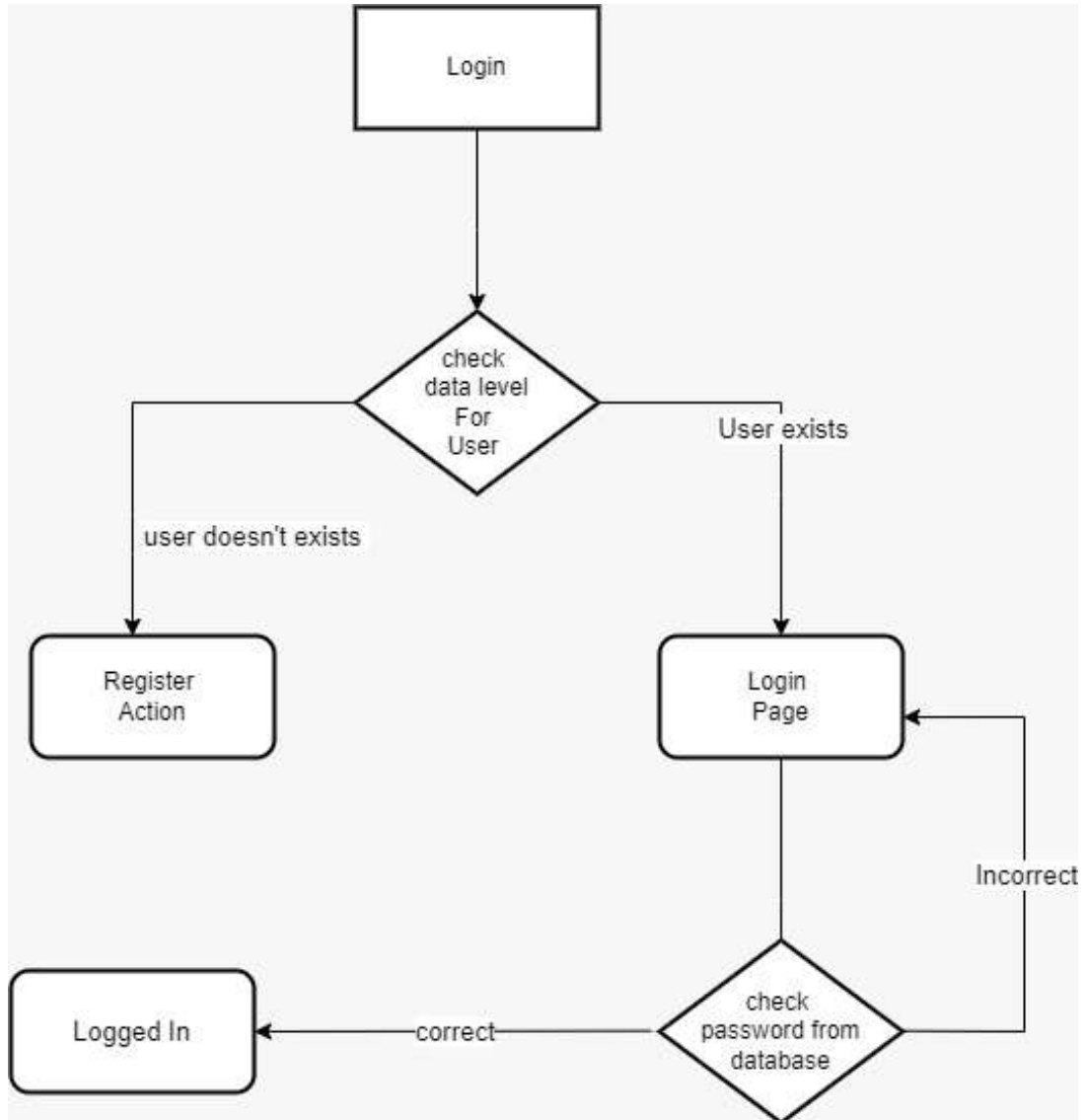


Figure 1: Registration Flow Chart

**Figure 2: Login Flow Chart**

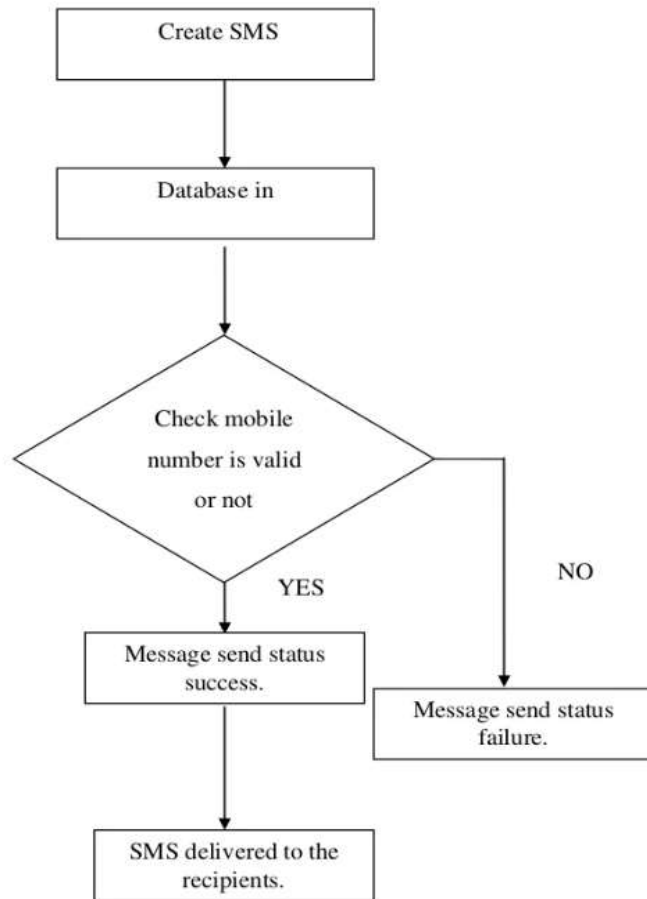


Figure 3: Message Flow Chart

3.5.2 Use Case UML Diagram

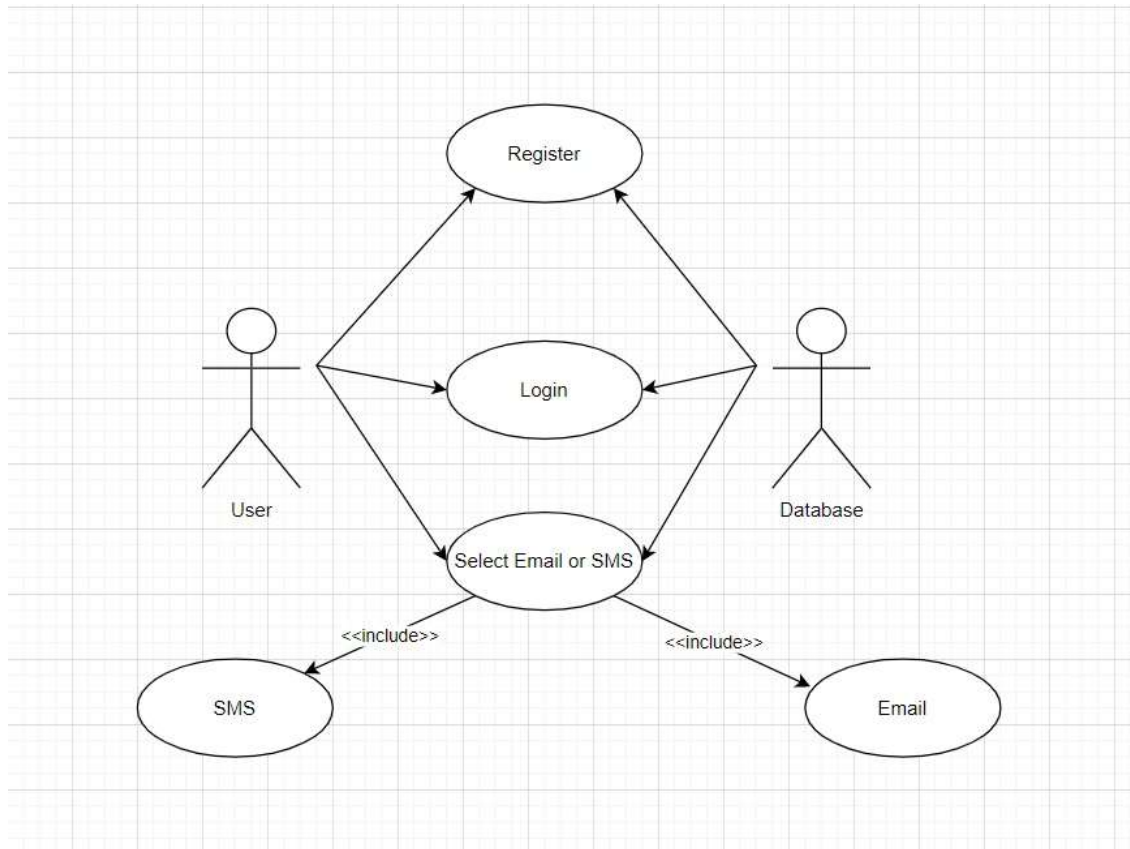


Figure 4: Use Case Diagram

3.5.3 ER Diagram

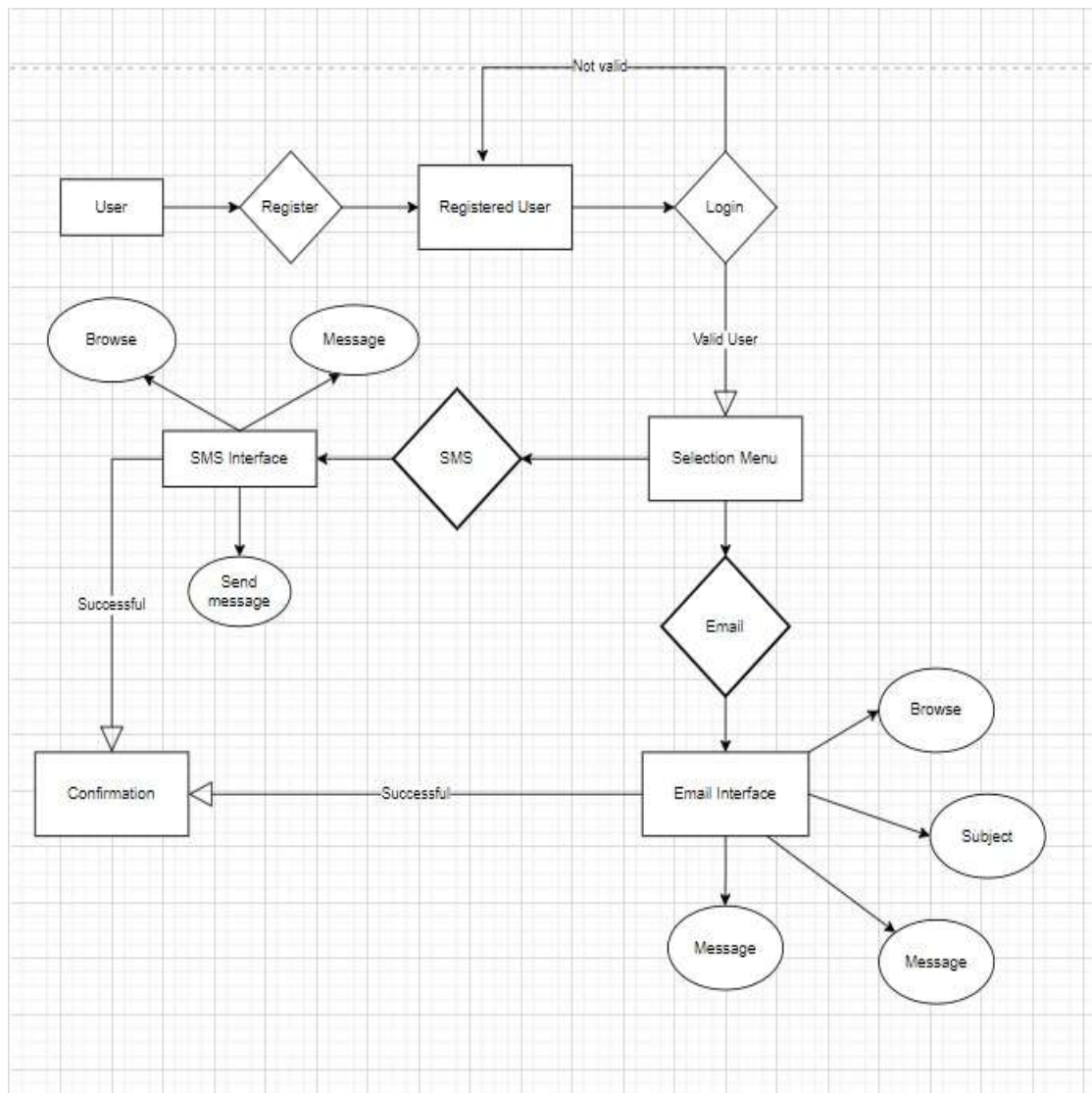


Figure 5: ER Diagram

Chapter 4. Implementation

4.1 Tools Used

4.1.1 Python Tkinter:

Tkinter is the de facto way in Python to create Graphical User interfaces (GUIs) and is included in all standard Python Distributions. It's the only framework built into the Python standard library. This Python framework provides an interface to the Tk toolkit and works as a thin object-oriented layer on top of Tk. The Tk toolkit is a cross-platform collection of 'graphical control elements', aka widgets, for building application interfaces.

4.1.2 MySQL:

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter "My", and "SQL", the abbreviation for Structured Query Language. A relational database organizes data into one or more data tables in which data may be related to each other. These relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access, and facilitates testing database integrity and creation of backups. MySQL is free and open-source software under the terms of the GNU (General Public License) and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB. MySQL has stand-alone clients that allow users to interact directly with a MySQL database using SQL, but more often, MySQL is used with other programs to implement applications that need relational database capability. MySQL is a component of the LAMP web application software stack (and others), which is an acronym for Linux, Apache, MySQL, Perl/PHP/Python. MySQL is used by many database-driven web applications, including Drupal, Joomla, phpBB, and WordPress. MySQL is also used

by many popular websites, including Facebook, Flickr, MediaWiki, Twitter, and YouTube.

4.2 Language Used

Python language (version 3.10.10) is used in the system due to the following characteristics:

4.2.1 Simple:

Python is a simple and minimalistic language. Reading a good Python program feels almost like reading English. This pseudo-code nature of Python is one of its greatest strengths. It allows you to concentrate on the solution to the problem rather than the syntax.

4.2.2 Free and Open Source:

Python is an example of FLOSS (Free/Libre and Open-Source Software). In simple terms, you can freely distribute copies of this software, read the software's source code, make changes to it, use pieces of it in new free programs. FLOSS is based on the concept of a community that shares knowledge. This is one of the reasons why Python is so good - it has been created and improved by a community that just wants to upgrade programming language.

4.2.3 Object Oriented:

Python supports procedure-oriented programming as well as object-oriented programming. In procedure-oriented languages, the program is built around procedures or functions that are nothing but reusable pieces of programs. In object-oriented languages, the program is built around objects which combine data and functionality. Python has a powerful but simple way of implementing object-oriented programming, especially, when compared to languages like C++ or Java.

4.2.4 Extensive Libraries:

The Python Standard Library is huge indeed. It can help you do various things involving regular expressions, documentation generation, unit testing, threading, databases, web browsers, CGI, FTP, email, XML XML-RPC, HTML, WAV files, cryptography, GUI (graphical user interfaces) using Tk, and also other system-

dependent stuff. Remember, all this is always available wherever Python is installed. This is called the "batteries included" philosophy of Python.

4.3 Modules Used

4.3.1 Twilio:

Twilio's APIs (Application Programming Interfaces) power its platform for communications. Behind these APIs is a software layer connecting and optimizing communications networks around the world to allow the users to call and message anyone, globally. API is short for 'Application Programming Interface'. An API is a set of rules that lets programs talk to each other, exposing data and functionality across the Internet in a consistent format. REST stands for 'Representational State Transfer'. This is an architectural pattern that describes how distributed systems can expose a consistent interface. When people use the term 'REST API', they are generally referring to an API accessed using the HTTP protocol at a predefined set of URLs. Twilio's Programmable SMS API helps you add robust messaging capabilities to your applications. Using this REST API, you can send and receive SMSs, track the delivery of sent messages, schedule SMS messages to send at a later time, and retrieve and modify message history.

4.3.2 SMTP LIB:

Simple Mail Transfer Protocol (SMTP) is a protocol, which handles sending and routing e-mail between mail servers. Python provides a `smtplib` module, which defines an SMTP client session object that can be used to send mail to any Internet machine with an SMTP or ESMTP listener daemon. An SMTP instance encapsulates an SMTP connection. It has methods that support a full repertoire of SMTP and ESMTP operations. If the optional host and port parameters are given, the SMTP `connect()` method is called with those parameters during initialization. If specified, `local_hostname` is used as the FQDN of the local host in the HELO/EHLO command. Otherwise, the local hostname is found using `socket.getfqdn()`. If the `connect()` call returns anything other than a success code, a `SMTPConnectError` is raised. The optional timeout parameter specifies a timeout in seconds for blocking operations like the connection attempt (if not specified, the global default timeout setting will be used). If the timeout expires, `TimeoutError` is raised. The optional `source_address` parameter allows binding to some specific source address in a

machine with multiple network interfaces, and/or to some specific source TCP port. It takes a 2-tuple (host, port), for the socket to bind to its source address before connecting. If omitted (or if host or port are "" and/or 0 respectively) the OS default behavior will be used.

4.4 Testing

Testing is the process of evaluation of a system to detect differences between given input and expected output and also to assess the feature of the system. Testing assesses the quality of the product. It is a process that is done during as well as after the development process.

4.4.1 Strategy Used:

Tests can be conducted based on two approaches –

- Functionality testing
- Implementation testing

The testing method used here is Black Box Testing. It is carried out to test the functionality of the program. It is also called 'Behavioral' testing. The tester, in this case, has a set of input values and respective desired results. On providing input, if the output matches with the desired results, the program is tested 'ok', and problematic otherwise.

4.4.2 Test Case and Analysis:

TEST-CASE 1:

Test Case ID	TC001
Summary	It will check if the user logging in already exists or not
Procedure	Just simply login
Expected Result	If the user exists, login, if it doesn't, ask to register first
Actual Result	User existed: logged in, User didn't exist: asked to register
Status	Pass

Table 1: Test Case 1

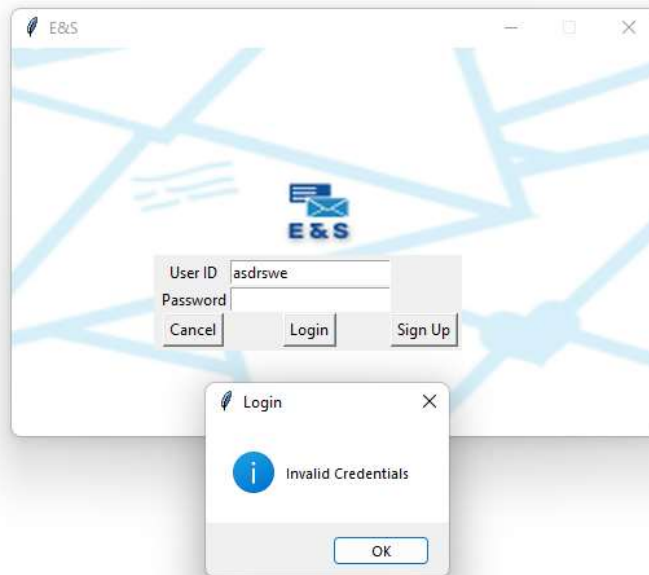


Figure 6.1: Validation while Login Process

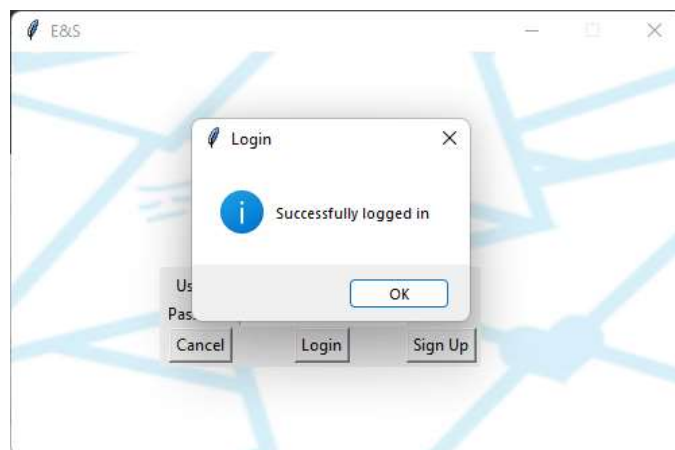
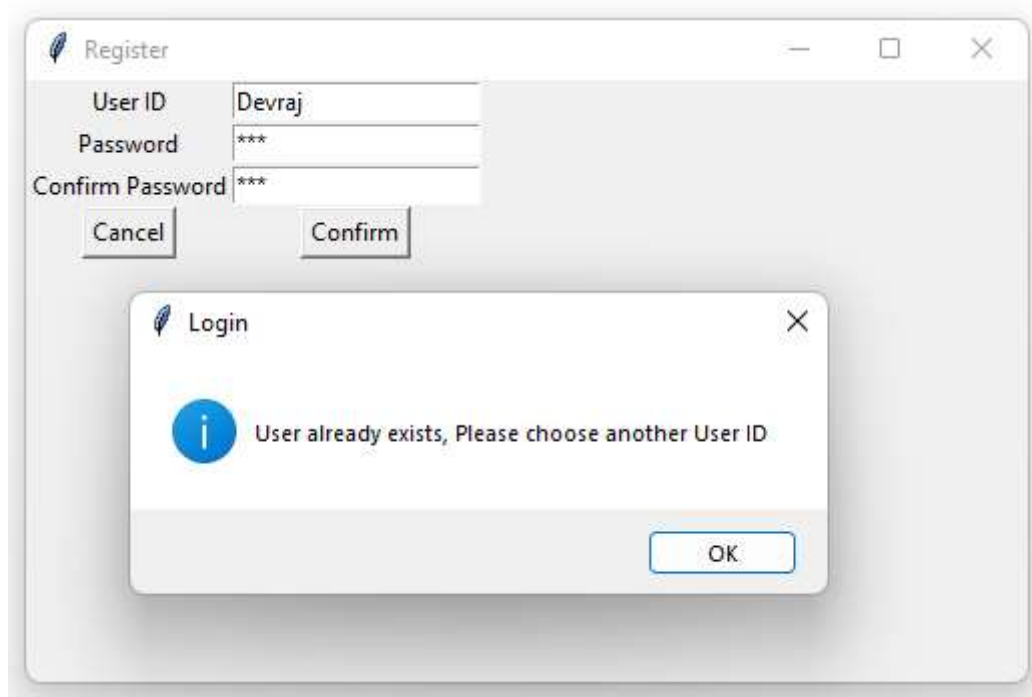


Figure6.2: Successful Login Demonstration

TEST-CASE 2:

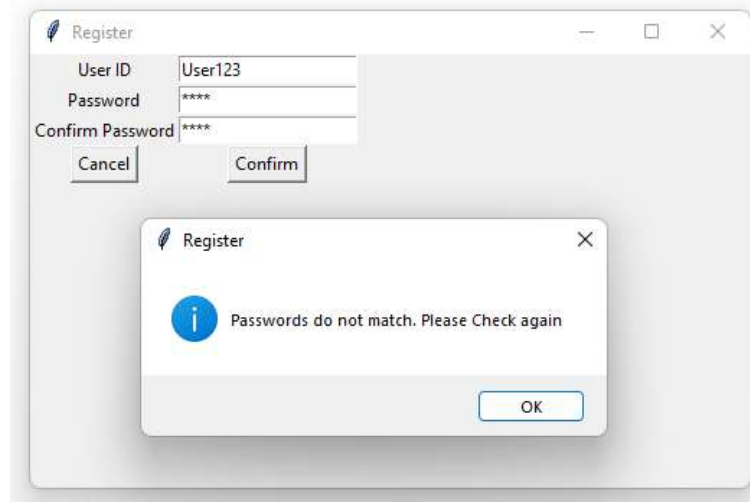
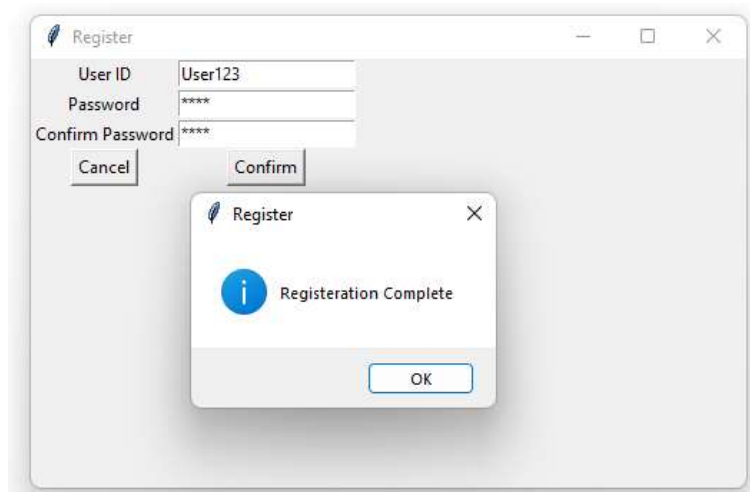
Test Case ID	TC002
Summary	It will check if a user is registered or not
Procedure	Just simply register
Expected Result	If the user exists, throw an error saying “user exists”, if it doesn’t, register
Actual Result	User existed: error, User didn’t exist: Registration
Status	Pass

Table 2: Test Case 2**Figure 7: Registration check**

TEST-CASE 3:

Test Case ID	TC003
Summary	It will check if the user has entered the password and confirm password same or not
Procedure	Just simply enter the password in both fields
Expected Result	If the password matches: registration is complete, if they don't match: throw an error
Actual Result	Password matched: registered, Didn't Match: error
Status	Pass

Table 3: Test Case 3

**Figure 8.1: Password confirmation through JavaScript****Figure 8.2: New Registration (Successful)**

4.5 Application Screenshots



Figure 9: Splash Screen

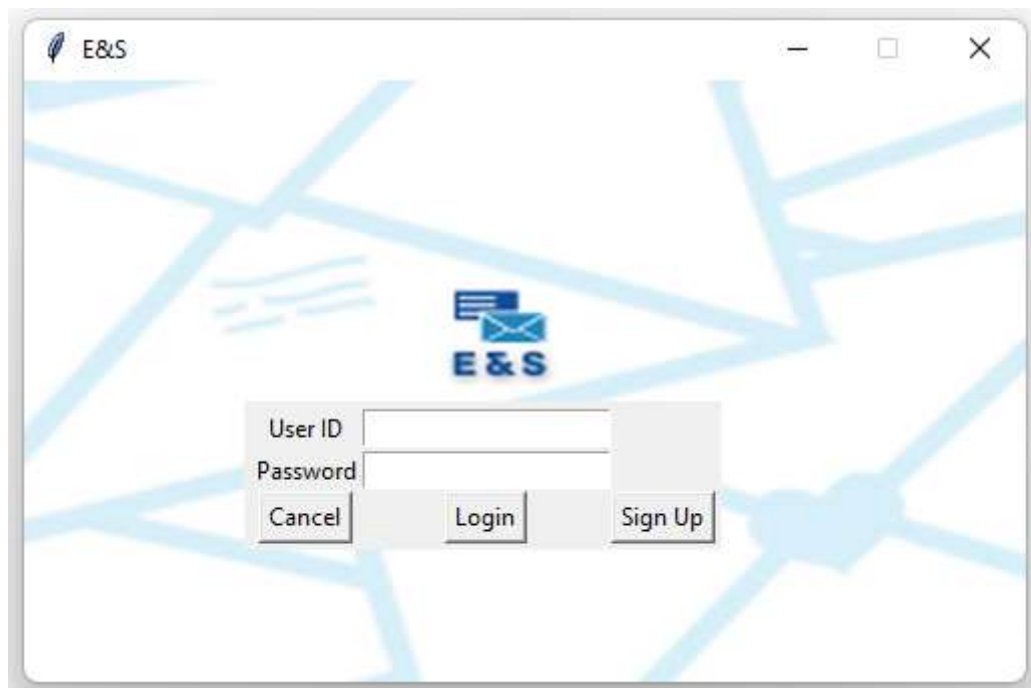


Figure 10: Login Screen



Figure 11: Successful Login Popup

A screenshot of a 'Register' form window. The window has a title bar with a feather icon and the text 'Register'. Inside, there are three input fields: 'User ID', 'Password', and 'Confirm Password'. Below the fields are two buttons: 'Cancel' and 'Confirm'. The window has standard minimize, maximize, and close buttons in the title bar.

Figure 12: Registration Page

A screenshot of a 'Register' form window with the fields filled in: 'User ID' is 'User123', 'Password' is '****', and 'Confirm Password' is '****'. The 'Confirm' button is highlighted. A 'Registration Complete' popup window is overlaid on top of the form. The popup has a title bar with a feather icon and the text 'Register'. Inside, there is a blue information icon followed by the text 'Registration Complete'. At the bottom right, there is an 'OK' button.

Figure 13: Successful Registration



Figure 14: Selection Menu

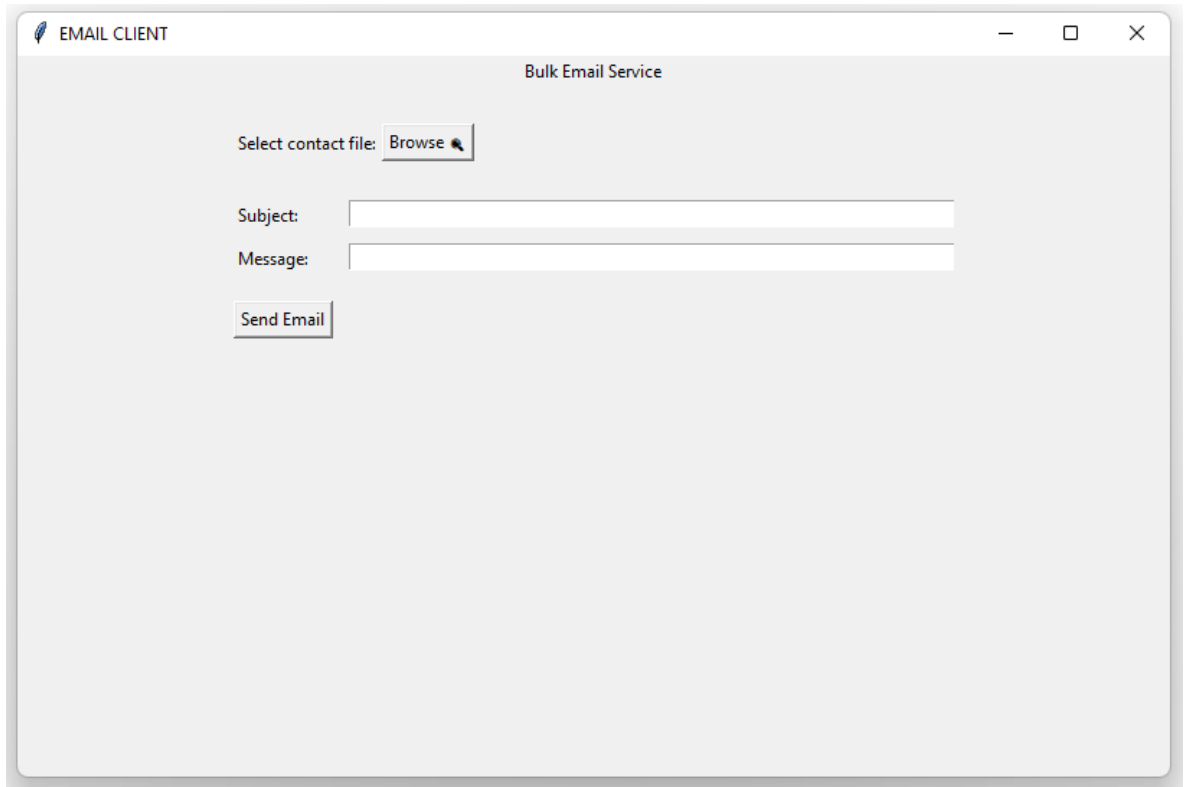


Figure 15: Email Client Page

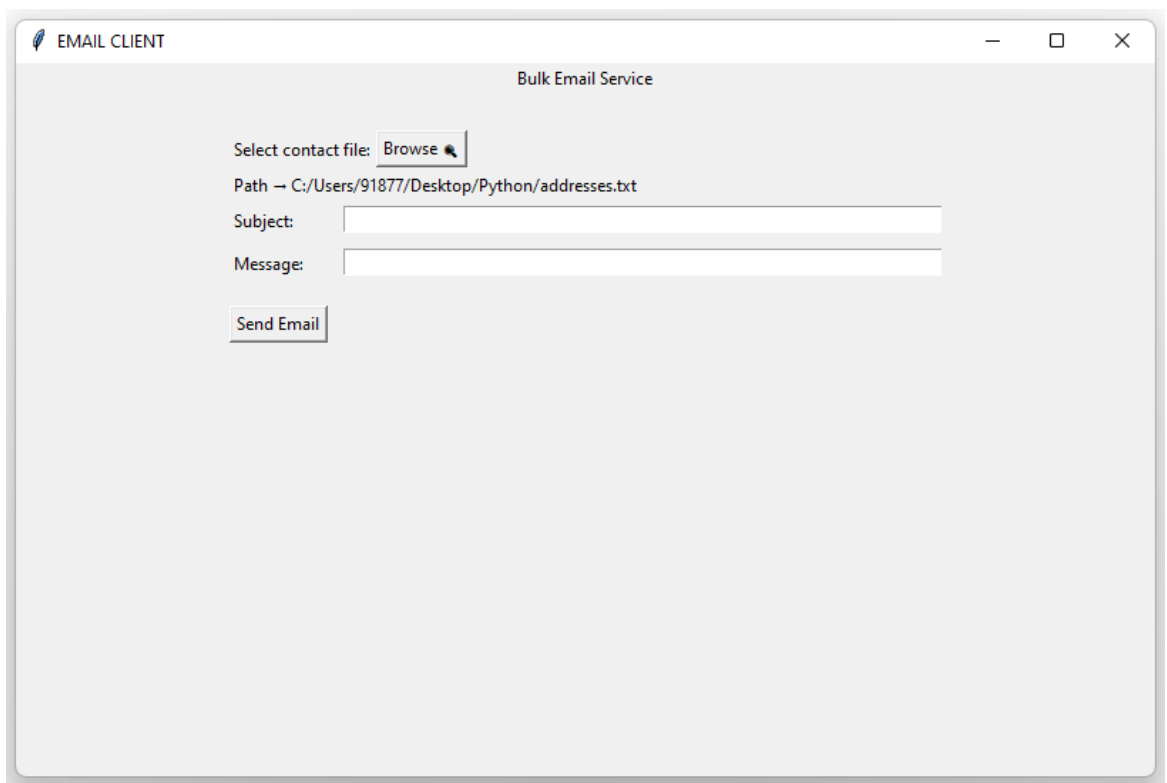


Figure 16: Demonstration of using bulk Email service

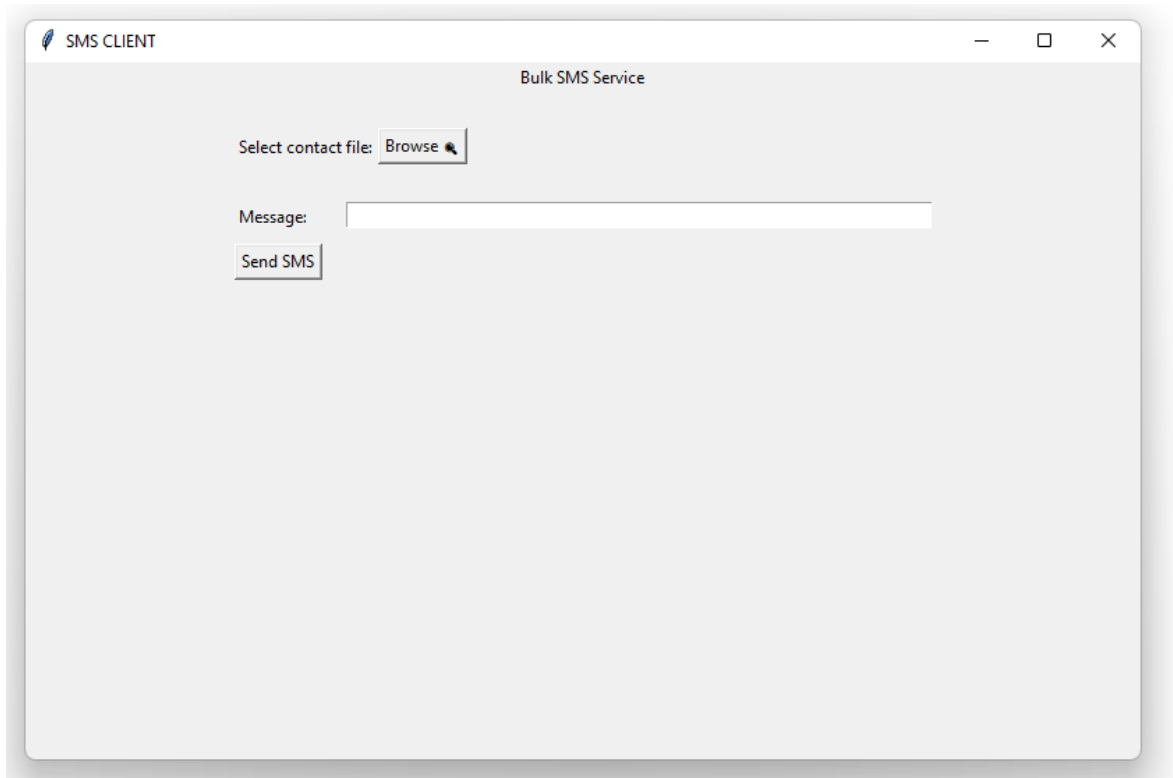


Figure 17: SMS Client Page

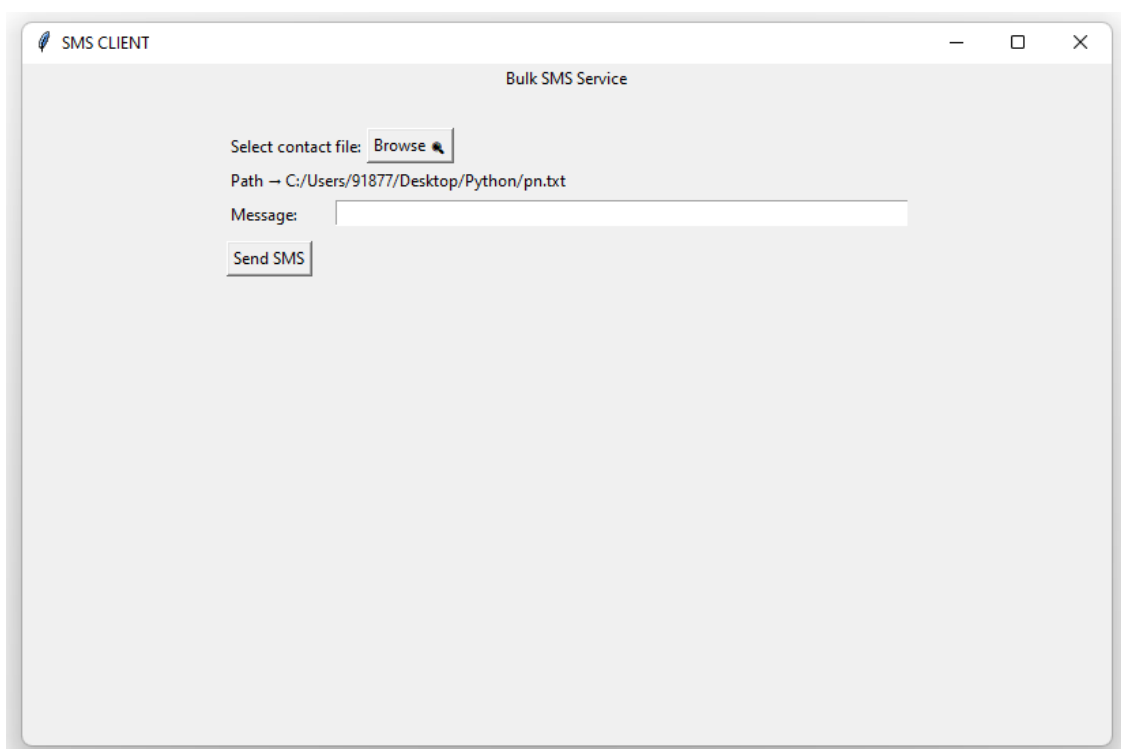


Figure 18: Demonstration of using bulk SMS service

4.6 Database Screenshots

```
mysql> desc project;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| User_ID    | varchar(20)   | YES  |     | NULL    |       |
| Password   | varchar(20)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.03 sec)
```

Figure 19: Project Table Description

```
mysql> desc email_info;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Sender_Mail    | varchar(50)   | YES  |     | NULL    |       |
| Receiver_Mail  | varchar(50)   | YES  |     | NULL    |       |
| Mail_Subject   | varchar(200)  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

Figure 20: Email_info Table Description

```
mysql> desc sms;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Contact_Number | varchar(15)   | YES  |     | NULL    |       |
| Time           | varchar(20)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Figure 21: SMS Table Description

```
mysql> select * from project;
+-----+-----+
| User_ID | Password |
+-----+-----+
| User 1  | 123      |
| User 2  | 1234     |
| Grashal | abc@123  |
| Goutam  | maha@123 |
+-----+-----+
4 rows in set (0.00 sec)
```

Figure 22: Entries in Project Table (For testing Purposes)

```
mysql> select * from email_info;
+-----+-----+-----+
| Sender_Mail | Receiver_Mail | Mail_Subject |
+-----+-----+-----+
| bulkproject1@gmail.com | devrajsinghamb123@gmail.com | Project Mail |
| bulkproject2@gmail.com | legentamb123@gmail.com | Project Mail |
| bulkproject1@gmail.com | devraj123@gmail.com | Project Mail |
| bulkproject3@gmail.com | goshenmasih19@gmail.com | Project Mail |
| bulkproject2@gmail.com | goutammahajan@gmail.com | Project Mail |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

Figure 23: Entries in email_info Table (For testing Purposes)

```
mysql> select * from sms;
+-----+-----+
| Contact_Number | Time |
+-----+-----+
| +916964256401 | 13:54:50 |
| +917989479294 | 14:02:45 |
| +919203902063 | 22:15:25 |
+-----+-----+
3 rows in set (0.00 sec)
```

Figure 24: Entries in SMS Table (For testing Purposes)

Chapter 5. Conclusion

5.1 Conclusion

This project is widely useful in various organizations, colleges, banking, companies, e-commerce, market news, government and public utilities, logistics, media and entertainment, travel and tourism industries, etc. It helps organizations to alert their customers/users about new updates. In logistics, it can be used to send shipping updates, invoices, bills, tracking details, and web URLs via SMS. Media and entertainment companies can use it to invite the audience to a TV or FM show or ask the audience to vote for their favorite contestants in a reality show. The travel and tourism industry can use it as a travel buddy by sending itineraries and travel routes via SMS to make the journey more hassle-free.

Sending SMS is easy, even school-going children are capable of doing it. No special training is required to learn the skill of texting. Every day people send hundreds of messages to their friends, office colleagues, and even strangers. This feature is now being exploited for commercial purposes, making bulk SMS marketing popular among small and medium-sized companies. Bulk SMS marketing comes with several advantages, such as no advertising waste as you have a clear idea of whom you are sending the message to, and complete flexibility in sending the desired message. Companies spend thousands of rupees to attract customers, but they are unable to retain even a few of them. By taking advantage of technology, a company can create a strong communication bridge to shrink the gap between brands and customers.

Bulk SMS services offer instant reach and increase sales and two-way communication. In contrast, with TV and print advertising, the time of message delivery is not restricted. With SMS, you can decide the time and day you want to send the message, ensuring its delivery. Furthermore, the cost of implementing a bulk SMS marketing plan is low, and you can easily send bulk SMS with the right software. When sending bulk SMS, it's important to follow the right means to acquire the phone numbers of prospective buyers.

In conclusion, bulk SMS is the perfect choice for marketers. There is no doubt that bulk SMS services will be the right choice for marketers to easily promote their products and services, target a huge population, and attract more potential customers.

5.2 Advantages

5.2.1 Speed Marketing:

In today's times, when most businesses are on the go, you get much lesser time to be in touch with your promising customers. So, it's necessary whatever you are trying to convey should be transmitted to all the concerned users.

Customer's Attention:

Getting customers' attention is the key to opening the possibility for your business or service. Here, the necessary aspect that counts to be working for the customers is the period spent by the customers in reading an e-mail or SMS. It also includes the effort that is made by the customer while reading the text. SMS stands more effective than an email because it asks for lesser effort from the customers.

5.2.2 Effective Communication:

Bulk SMS is a unidirectional communication channel, a contact number, and a website link can be easily shared through the message. Such information can be used for customers' further reference. In addition to this, a personalized message can be sent to the respective set of customers according to the details of demographics and their interests.

5.2.3 Easy reach:

Nowadays, even the simplest mobile phone has an SMS feature. People do not need any internet connection to access the SMS feature (that is the best bulk SMS benefit, it has very few dependencies). That means you can reach more users with SMS than any other platform.

5.2.4 Low Cost:

Newspaper ads, TV commercials and almost all forms of outdoor advertising cost prohibitive for most small to medium-sized businesses. Bulk SMS marketing, on the other hand, has such low setup and running costs.

5.2.5 Delivery Report:

SMS can be easily tracked using delivery reports. You can see which messages were delivered to which mobile numbers.

5.3 Limitations of the Project Work

While many bulk email and SMS systems are highly regarded and widely used, there are still limitations to these systems. Some common limitations include:

5.3.1 Delivery rate limitations:

The bulk SMS and Email Delivery System has limitations on the number of messages that can be sent and the delivery rate. This can result in delays or failed deliveries if the volume of messages exceeds the system's capacity.

5.3.2 Content limitations:

There are certain restrictions on the content that can be sent through bulk emails and SMS systems, such as restrictions on the use of certain keywords or images.

5.3.3 Spam filters:

Email providers have implemented spam filters that can prevent bulk emails from reaching the inbox of the recipient. Similarly, some SMS providers have implemented filters that can prevent SMS messages from being delivered to the recipient.

5.3.4 Cost:

Depending on the system and the volume of messages being sent, bulk email and SMS systems can be quite expensive.

5.3.5 Integration limitations:

Some bulk email and SMS systems may have limited integration with other systems, such as customer relationship management (CRM) systems or e-commerce platforms.

5.3.6 Technical difficulties:

The use of bulk email and SMS systems can be technically complex and requires a certain level of technical knowledge to set up and use effectively.

Despite these limitations, many organizations still find bulk email and SMS systems to be an effective tool for communicating with large numbers of people quickly and efficiently.

5.4 Future Scope:

The scope below demonstrates how Bulk SMS and Email Delivery System is utilized in various industries to achieve business and communication efficiency and solve critical business problems. There is scope for Bulk SMS service, especially in India. As you know that India still is considered a country under development, therefore, such smart, automatic, and time-saving services are a requirement for us.

Sending SMS is quite easy and even a school-going kid can learn to use it. You do not need experience or any specialized training to learn the skill of texting. Every day people send hundreds of messages to their friends, office colleagues, and even strangers. This feature is now exploited for commercial purchases. Bulk SMS marketing is now quite popular with many small and medium-sized companies. Bulk SMS marketing comes with quite a good number of advantages. First and foremost, there is no advertising wastage as you have a clear idea of to whom you are sending the message. Also, you have complete flexibility in sending the desired message. Companies spend thousands of rupees just to attract customers, yet they are unable to retain even a few of them. By taking advantage of technology, companies can create a strong communication bridge to shrink the gap between brands and customers. Bulk SMS services are just another way to reach people but what makes this different is its instant reach as it increases sales and two-way communication instantly.

The TV and print advertising, you cannot control the time of the message delivery since it is governed by many different factors. However, we can decide at which

time the SMS is to be sent according to the availability of the user, to make sure that the SMS is being read by him. Hence, it is more beneficial. The cost of implementing a bulk SMS marketing plan is quite low. You can easily send bulk SMS in India with the right software. When you have to send bulk SMS, ensure that you follow the right means to acquire the phone numbers of prospective buyers. The Bulk SMS System proposed is the perfect choice for marketers.

With proper knowledge and using certain techniques, we can integrate this system with applications such as WhatsApp and Telegram so that the same message can be floated among all the contacts in one go as these are the widely used communication platforms in today's world.

Chapter 6. Bibliography

- [1] <https://www.sih.gov.in/sih2022PS>
- [2] <https://www.google.comhttps://landing.sendinblue.com/bulk-email/>
- [3] <https://www.g2.com/products/bulk-SMS/bulkmails/competitors/alternatives>
- [4] <https://www.textlocal.in>
- [5] Arnaud Henry-Labordere, Vincent Jonack- "SMS and MMS Interworking in Mobile Networks"
- [6] Paul DuB0is- "MySQL (4th Edition)"
- [7] Dotson Creative publishers- "Bulk SMS Business Blueprint"
- [8] Alan Moore – "Python GUI Programming with Tkinter: Develop responsive and powerful GUI applications with Tkinter"
- [9] "A Study on the Effectiveness of Bulk SMS Marketing in India" by Amruta R. Pawar, published in the International Journal of Emerging Research in Management & Technology.
- [10] "The Impact of SMS Marketing on Consumer Behavior" by E. Bamidele and A. Adeyemo, published in the Journal of Marketing Management.
- [11] "Bulk SMS Marketing: An Effective Marketing Tool for Small and Medium Enterprises" by P. K. S. Bhadoria and P. K. S. Bhadoria, published in the Journal of Emerging Trends in Computing and Information Sciences.

[12] "Bulk Email Marketing: A Study of Best Practices and Success Factors"
by R. K. Bhowmick and M. K. Bhowmick, published in the International
Journal of Marketing, Financial Services & Management Research.

[13] "The Impact of Bulk SMS Marketing on Consumer Purchasing Behavior"
by O. A. Adebisi, published in the Journal of Business and Management.

Guide Interaction Sheet

Date	Discussion	Action Plan
03/01/2023	Discussed the title of the project	Bulk SMS and Email Delivery System was decided as the title
17/01/2023	Discussion on the technology to be used for sending bulk SMS and Email in one go	Tkinter, MySQL and other tools were finalized
31/01/2023	Discussion of the creation of synopsis of the project	Gathering of information for synopsis creation
07/02/2023	Suggestions on how to do a literature survey and preliminary investigation on the topic	Many research papers were read, understood, and then the abstract was to be written.
14/02/2023	Discussion on the implementation of the project	Using tkinter and other tools, we decided to implement project.
28/02/2023	Discussion on the objective of the project (sending bulk email or SMS to several users in a single go)	Decided to Include the logic of email sending in the program
07/03/2023	Suggestion for counting the number of contacts to whom the message is being sent	Took steps for adding and modifying the program for counting contacts also
21/03/2023	For the generation of log files and storing the contacts, the database was advised to be added	For each SMS/ Email sent, an entry must be made in the database so that count can be made easy
11/04/2023	Discussion on project documentation	Decided to write the content and integrate it into the proper format of the report

Chapter 7: Source Code

```
import tkinter as tk
import tkinter.ttk
import sys
from tkinter import messagebox
from tkinter import filedialog
from tkinter import *
import PIL
from PIL import ImageTk, Image
from twilio.rest import Client
import random
from random import randrange
import mysql.connector as sql
import smtplib
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
from datetime import datetime
now = datetime.now()
current_time = now.strftime("%H:%M:%S")

con=sql.connect(host="localhost",user="root",passwd="root",database="project")
cursor1=con.cursor()

inimain=tk.Tk()
inimain.title("E&S")
inimain.geometry("500x300")
inimain.resizable(height=0,width=0)

mf=Frame(inimain,background="white", highlightthickness=0, width=500,
height=300)

welimg=Image.open("logo_small.png")
welimage1=welimg.resize((1000,820),Image.ANTIALIAS)
imgwel=ImageTk.PhotoImage(welimage1)

lbimg=Label(inimain,image=imgwel,highlightthickness=0,bg="white")
lbimg.image=imgwel
lbimg.place(x=-265,y=-260)

mf.grid(pady=0)

def Sign():
    iniframe=Frame(inimain,highlightthickness=3,width=1366,height=40,
    bd= 0)
```

```
iniframe.place(x=110,y=160)
iniframe.tkraise()

lb1=Label(iniframe,text="User ID")
lb1.grid(row=0,column=0)
lb2=Label(iniframe,text="Password")
lb2.grid(row=1,column=0)
e1=Entry(iniframe)
e1.grid(row=0,column=1)
e2=Entry(iniframe,show="*")
e2.grid(row=1,column=1)

def loginfunc():

    cursor2=con.cursor(buffered=True)
    cursor3=con.cursor(buffered=True)
    cursor2.execute("SELECT User_ID from project WHERE User_ID
=%s",(e1.get(),))
    cursor3.execute("SELECT Password from project WHERE User_ID
=%s",(e1.get(),))
    userverify=cursor2.fetchone()
    passverify=cursor3.fetchone()

    if (e1.get(),)==userverify:
        if (e2.get(),)==passverify:
            messagebox.showinfo("Login","Successfully logged in")
            inimain.destroy()
            select=tk.Tk()
            select.title("E&S")
            select.geometry("500x300")
            select.resizable(height=0,width=0)

            mycolor='#FFFFFF'
            frame1 = Frame(select, highlightbackground=mycolor,
background="grey", highlightthickness=3, width=1366, height=100, bd= 0)

            frame1.grid(pady=0)
            image=Image.open("logo_small.png")
            image1=image.resize((1100,900),Image.ANTIALIAS)
            img=ImageTk.PhotoImage(image1)

logo=Label(frame1,image=img,highlightthickness=0,bd=0)
logo.image=img
logo.place(x=-310,y=-385)
mycolor2='#177565'
frame2 = Frame(select, highlightbackground="white",
background="white", highlightthickness=3, width=1366, height=400, bd= 0)
```

```

frame2.grid(pady=0)
onglb=Label(frame2,text="Select
Service",bg="white",fg='Black',relief=FLAT,font=(("FranklinGothic"),14))
onglb.place(x=0,y=0)
frame2.grid_propagate(False)

def email():
    class emailGui:
        def __init__(self):
            self.root = tkinter.Tk()
            self.root.geometry("800x500")
            self.root.title("EMAIL CLIENT")
            self.top_frame =
tkinter.Frame(self.root,width=500,height=100)
            self.bottom_frame =
tkinter.Frame(self.root,width=500,height=100)
            self.top_frame.pack_propagate(0)
            self.bottom_frame.pack_propagate(0)
            self.header = tkinter.Label(self.top_frame, text =
"Bulk Email Service")
            self.header.pack(side = 'top')
            self.to = tkinter.Label(self.top_frame, text = "")
            self.to.place(x=0,y=75)
            self.message = tkinter.Entry(self.bottom_frame,
width = 100)
            self.subject = tkinter.Entry(self.bottom_frame, width
= 100)
            self.send = tkinter.Button(self.bottom_frame, text =
"Send Email", command = self.sendEmail)
            self.msg = tkinter.Label(self.bottom_frame, text =
"Message:")
            self.sub = tkinter.Label(self.bottom_frame, text =
"Subject:")
            self.sub.place(x=0,y=0)
            self.msg.place(x=0,y=30)
            self.message.place(x=80,y=30)
            self.subject.place(x=80,y=0)
            self.send.place(x=0,y=70)
            self.top_frame.pack()
            self.bottom_frame.pack()
            self.browse = tkinter.Label(self.top_frame, text =
"Select contact file:")
            self.browse.place(x=0,y=50)
            self.button_explore =
tkinter.Button(self.top_frame,text = "Browse ,command = self.browseFiles)
            self.button_explore.place(x=103,y=47)
            tkinter.mainloop()
            def browseFiles(self):

```

```

self.filename = filedialog.askopenfilename(initialdir
= "/",
title = "Select a File",
filetypes = (("Text files",
               "*.txt*"),
              ("all files",
               "*.*")))
self.to.configure(text="Path → "+self.filename)

def sendEmail(self):
    with open(self.filename) as f:
        contents = f.read()
        email = contents.splitlines()
        mail_content = str(self.message.get())
        subject_mail=str(self.subject.get())

    with open("email.txt") as f:
        contents = f.read()
        lst = contents.splitlines()
        dict1 = {lst[i]: lst[i + 1] for i in range(0, len(lst),
2)}}

        emails_lst=[lst[i] for i in range(0,len(lst),2)]

    for address in email:
        random_index = randrange(len(emails_lst))
        i=emails_lst[random_index]
        sender_address=i
        sender_pass = dict1[i]
        receiver_address = address
        print(sender_address,sender_pass)
    #Setup the MIME
    message = MIMEMultipart()
    message['From'] = sender_address
    message['To'] = receiver_address
    message['Subject'] = subject_mail #The subject
line
    #The body and the attachments for the mail
    message.attach(MIMEText(mail_content, 'plain'))
    #Create an SMTP session for sending the mail
    session = smtplib.SMTP('smtp.gmail.com', 587)

    #use gmail with port
    session.starttls() #enable security
    session.login(sender_address, sender_pass)

    #login with mail_id and password
    text = message.as_string()
    session.sendmail(sender_address,
receiver_address, text)
    session.quit()

```

```
print('Mail Sent')

val=(sender_address,receiver_address,subject_mail)

cursor_info=con.cursor(buffered=True)
cursor_info.execute('INSERT INTO
email_info(Sender_Mail,Receiver_Mail,Mail_Subject) VALUES (%s,%s,%s)' , val)
con.commit()
```

```
GUI = emailGui()
```

```
def sms():
    class SMSGUI:
        def __init__(self):
            self.root = tkinter.Tk()
            self.root.geometry("800x500")
            self.root.title("SMS CLIENT")
            self.top_frame =
tkinter.Frame(self.root,width=500,height=100)
            self.bottom_frame =
tkinter.Frame(self.root,width=500,height=100)
            self.top_frame.pack_propagate(0)
            self.bottom_frame.pack_propagate(0)
            self.header = tkinter.Label(self.top_frame, text =
"Bulk SMS Service")
            self.header.pack(side = 'top')
            self.to = tkinter.Label(self.top_frame, text = "")
            self.to.place(x=0,y=75)
            self.message = tkinter.Entry(self.bottom_frame,
width = 100)
            self.send = tkinter.Button(self.bottom_frame, text =
"Send SMS", command = self.sendSms)
            self.msg = tkinter.Label(self.bottom_frame, text =
"Message:")
            self.msg.place(x=0,y=0)
            self.message.place(x=80,y=0)
            self.send.place(x=0,y=30)
```

```

        self.top_frame.pack()
        self.bottom_frame.pack()
        self.browse = tkinter.Label(self.top_frame, text =
"Select contact file:")
        self.browse.place(x=0,y=50)
        self.button_explore =
tkinter.Button(self.top_frame,text = "Browse ",command = self.browseFiles)
        self.button_explore.place(x=103,y=47)
        tkinter.mainloop()
    def browseFiles(self):

        self.filename = filedialog.askopenfilename(initialdir
= "/",
                                                    title = "Select a File",
                                                    filetypes = (("Text files",
                                                                "*.txt*"),
                                                                ("all files",
                                                                "*.*")))
        self.to.configure(text="Path → "+self.filename)

    def sendSms(self):
        account_sid =
'AC9aeff01229521f230da5030c4b9378'
        auth_token =
'96649a54f3bde8014bb8eedc543fcd87'
        client = Client(account_sid, auth_token)
        msg = str(self.message.get())
        with open(self.filename) as f:
            contents = f.read()
            phone_num = contents.splitlines()

        msg = str(self.message.get())

        for rec in phone_num:
            message =
client.messages.create(body=msg,from_='+18176464494',to= rec)
            print(message.sid)
            tkinter.messagebox.showinfo('Success', 'Message
sent successfully')

            val=(rec,str(current_time))

            cursor_sms=con.cursor(buffered=True)
            cursor_sms.execute('INSERT INTO
sms(Contact_Number,Time) VALUES (%s,%s)' , val)
            con.commit()

```

```
gui = SMSGUI()
```

```
        Button(frame2,text="E-Mail",height= 2,  
width=10,command=email).place(x=100,y=70)  
        Button(frame2,text="SMS",height= 2,  
width=10,command=sms).place(x=300,y=70)
```

```
        else:  
            messagebox.showinfo("Login","Invalid Credentials")  
    else:  
        messagebox.showinfo("Login","Invalid Credentials")
```

```
def exitfunc():  
    inimain.destroy()
```

```
def register():  
    regis=tk.Tk()  
    regis.title("Register")  
    regis.geometry("500x300")  
  
    lb3=Label(regis,text="User ID")  
    lb3.grid(row=0,column=0)  
    lb4=Label(regis,text="Password")  
    lb4.grid(row=1,column=0)  
    lb7=Label(regis,text="Confirm Password")  
    lb7.grid(row=2,column=0)  
    e3=Entry(regis)  
    e3.grid(row=0,column=1)  
    e4=Entry(regis,show="*")  
    e4.grid(row=1,column=1)  
    e5=Entry(regis,show="*")  
    e5.grid(row=2,column=1)
```

```
def register1():
```

```
        cursor4=con.cursor(buffered=True)
        cursor4.execute("SELECT User_ID from project WHERE
User_ID =%s",(e3.get(),))
        usercheck=cursor4.fetchone()

        if (e3.get(),)==usercheck:
            messagebox.showinfo("Login","User already exists, please
choose another User ID")
        else:
            if e4.get()==e5.get():
                messagebox.showinfo("Register","Registration
Complete")

                l=e3.get()
                m=e4.get()

                val=(l,m)
                cursor1.execute('INSERT INTO
project(User_ID>Password) VALUES (%s,%s)' , val)
                con.commit()
                print(cursor1.rowcount, "record inserted.")
                regis.destroy()

            else:
                messagebox.showinfo("Register","Passwords do not
match. Please Check again")

    def exitfunc2():
        regis.destroy()

    Button(regis,text="Confirm",command=register1).grid(row=3,column=1)

    Button(regis,text="Cancel",command=exitfunc2).grid(row=3,column=0)

    Button(iniframe,text="Cancel",command=exitfunc).grid(row=3,column=0)

    Button(iniframe,text="Login",command=loginfunc).grid(row=3,column=1)
        Button(iniframe,text="Sign
Up",command=register).grid(row=3,column=2)

    Button(inimain,text="Sign In/Sign Up",command=Sign).place(x=190,y=200)
```