A
Mini-Project Report on
Automated Email Sender

Submitted in partial fulfillment of the requirements
for the degree of
BACHELOR OF ENGINEERING
IN
Computer Science & Engineering
Artificial Intelligence & Machine Learning

by

Mohit Suthar (22106056)
Vaishnavi Sonawane (22106125)
Disha Suryawanshi (22106112)
Aman Thakur(22106014)

Under the guidance of

Prof. Mahesh Pawaskar

Department of Computer Science & Engineering
(Artificial Intelligence & Machine Learning)
A. P. Shah Institute of Technology
G. B. Road, Kasarvadavli, Thane (W)-400615
University Of Mumbai
2023-2024

A. P. SHAH INSTITUTE OF TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled "Automated Email Sender**"** is a bonafide work of Mohit Suthar (22106056), Vaishnavi Sonawane (22106125), Disha Suryawanshi (22106112), Aman Thakur (22106014) submitted to the University of Mumbai in partial fulfillment of the requirement for the award of **Bachelor of Engineering** in **Computer Science & Engineering (Artificial Intelligence & Machine Learning).**

Prof. Mahesh Pawaskar	Dr. Jaya Gupta
Mini Project Guide	Head of Department

A. P. SHAH INSTITUTE OF TECHNOLOGY

Project Report Approval

This Mini project report entitled "**Automated Email Sender"** by **Mohit Suthar (22106056), Vaishnavi Sonawane (22106125), Disha Suryawanshi (22106112), Aman Thakur (22106014)** is approved for the degree of ***Bachelor of Engineering*** in***Computer Science &Engineering***, (AIML) ***2023-24***.

External Examiner: _____

Internal Examiner: _____

Place: APSIT,
ThaneDate:

Declaration

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Mohit Suthar	Vaishnavi Sonawane	Disha Suryawanshi	Aman Thakur
(22106056)	(22106125)	(22106112)	(22106014)

ABSTRACT

The "Automated Email Sender " project is a comprehensive exploration of the development and implementation of an intelligent email automation system. In today's digitally driven world, email communication plays a pivotal role in personal, professional, and business interactions. The need for efficient, time-saving, and error-free email management has led to the development of automated email sending solutions.

This project report outlines the design, development, and deployment of an automated email sender system that significantly enhances email productivity. The system leverages cutting-edge technologies and algorithms to streamline the process of sending emails, with a focus on improving efficiency, personalization, and scalability.

Key project components include the development of a user-friendly web-based interface, integration with email services, user management, email scheduling, and intelligent content personalization. The system is capable of sending a wide variety of emails, from newsletters and marketing campaigns to transactional and reminder emails.

The report highlights the technical architecture, design considerations, and the algorithms employed in the system, ensuring that automated emails are delivered with precision and timeliness. It discusses the challenges encountered during development and the corresponding solutions, making it a valuable resource for developers and businesses seeking to implement similar systems.

KEYWORD: mail Delivery, Email Automation, Email management.

Index

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

In an era where communication is paramount, the ability to efficiently manage and send emails has become a critical aspect of personal and professional life. As the volume of emails continues to rise, so does the demand for tools that simplify and automate this process.

Our project, Mail Bot , is an innovative solution designed to revolutionize the way we handle email communication. This automated email sender application is crafted to address the challenges of modern email management, making it easier and more effective than ever before.

In this introduction, we'll explore the need for such a solution in today's fast-paced world of digital communication, outline the objectives of our project, and provide a glimpse of the key features that make Mail Bot a game-changer in the realm of email management. Join us on this journey as we unlock the potential of streamlined and automated email communication.

CHAPTER 2

LITERATURE SURVEY

2. LITERATURE SURVEY

2.1-HISTORY

The history of the automated email sender project dates back to the early days of email, when early adopter businesses and organizations began using simple scripts to automate tasks such as sending out newsletters and marketing messages. However, it was not until the late 1990s and early 2000s that automated email sender projects began to become more sophisticated and widespread One of the key drivers of the growth of automated email sender projects was the development of more powerful and user-friendly email marketing platforms. These platforms made it possible for businesses of all sizes to create and send automated email campaigns without the need for specialized technical expertise Another key driver of the growth of automated email sender projects was the increasing popularity of cloud computing. Cloud-based automated email sender projects offer a number of advantages over traditional on-premises solutions, such as lower costs, greater scalability, and easier maintenance. Today, automated email sender projects are an essential part of the marketing and sales strategies of many businesses. They are used to send a wide variety of emails, including newsletters, promotional emails, welcome emails, abandoned cart emails, product recommendation emails, customer feedback emails, order confirmation emails, shipping notifications, account alerts, and more.

2.2-LITERATURE REVIEW

1. The Proliferation of Email Communication:

As the volume of business-critical e-mails continues to grow, the need to automate the management of e-mails increases for several reasons, such as spam e-mail classification, phishing e-mail classification, and multi-folder categorization, among others [1]

2. Enhancing Email Security:

Email is a critical Internet application, and its security is important. The Sender Policy Framework (SPF), DomainKeys Identified Mail (DKIM), and Domain-based Message Authentication, Reporting, and Conformance (DMARC) were developed to enable mail servers to detect and reject email coming from fraudulent sources. [2]

3. Navigating the Digital World:

The advancement of technology and its infinite possibilities have made it unavoidable for current generations to fully utilise Internet technology. Email, as one of the most extensively used features of the Internet, serves as a basic requirement. Despite the availability of various screen readers, visually impaired users face challenges when using the internet.[3]

4. Analysing Email Transmission :

A mail spoofing attack is a harmful activity that modifies the source of the mail and trick users into believing that the message originated from a trusted sender whereas the actual sender is the attacker. Based on the previous work, this paper analyzes the transmission process of an email. Our work identifies new attacks suitable for bypassing SPF, DMARC, and Mail User Agent's protection mechanisms.[4]

5. Addressing the Challenge of Email Sorting and Spam Filtering :

Emails that seem valid are received in the inbox and sometimes relevant emails are directed to spam. Another aspect of the problem is that, due to very high number of incoming emails, it is very difficult to identify the required ones easily. In this process, user wastes so much of their time, energy and efforts by sifting through irrelevant emails in which they have no interest.[5]

CHAPTER 3

Problem Statement

3. Problem Statement

The primary objective of the "Mail Bot" is to provide a comprehensive solution to these challenges by creating an automated email sender that offer Efficiency and Time Savings: Automating common email tasks, such as scheduling, template management, and personalized responses, to save users time and reduce manual effort Enhanced Organization: Intuitive features for categorizing and prioritizing emails, ensuring efficient email management and reducing email overload Security and Data Protection: Implementing robust security measures, including encryption and secure connections, to safeguard user data and privacy Scalability and Performance: Designing the application to handle a high volume of emails without compromising performance, ensuring it's suitable for both individual users and organizations. User-Centric Design: Developing a user-friendly interface with comprehensive user training and support to enhance user adoption and satisfaction In today's digital era, email communication is a cornerstone of personal, professional, and business interactions. However, as the volume of emails continues to rise, managing email campaigns, newsletters, and transactional emails manually is increasingly challenging and time-consuming. This challenge is further exacerbated by the need for personalization and segmentation to engage recipients effectively. In this context, the problem statement revolves around the need for an efficient and robust automated email sender solution. The existing tools often lack the ability to seamlessly integrate with popular email services, provide sophisticated scheduling and personalization, and ensure compliance with evolving email marketing regulations. Addressing these issues is vital for businesses and individuals to streamline their email communication, enhance user engagement, and deliver personalized, relevant content efficiently while adhering to legal and ethical email marketing practices.

CHAPTER 4

Experimental Setup

4.1 Hardware Setup

CPU and RAM:

- The processing power (CPU) and memory (RAM) of your server should be chosen based on the expected workload and the number of emails you plan to send. For small to medium-scale systems, a server with at least 2-4 CPU cores and 4-8 GB of RAM should suffice.

Storage:

- Allocate sufficient storage space for your application and data. SSDs are preferable for faster data access. You may need additional storage for email templates, logs, and user data.

Network Connectivity:

- Ensure your server has a stable and high-speed internet connection. This is essential for sending emails through SMTP or APIs.

4.2 Software Setup

Operating System:

Choose an operating system for your server. Common choices include Linux distributions (e.g., Ubuntu, CentOS, or Debian) due to their stability and cost-effectiveness. Select the OS that aligns with your team's expertise.

Web Server:

Install a web server to host the application. Popular web servers include Apache, Nginx, or lighttpd. Make sure to configure the web server properly for security and performance.

Database System:

Select a database management system (DBMS) to store user data and email templates. Options include MySQL, PostgreSQL, or NoSQL databases like MongoDB, depending on your data structure and scalability needs.

Application Server:

Depending on your programming language and framework, set up an application server or runtime environment. Common choices are Node.js, Ruby on Rails, Django, Flask, Java (with Apache Tomcat), or PHP (with PHP-FPM).

Programming Language and Framework:

Choose the programming language and web framework for your application. For example, you can use Python with Django or Flask, Ruby with Ruby on Rails, JavaScript with Node.js, or PHP.

CHAPTER 5

Proposed System & Implementation

5.1 BLOCK DIAGRAM

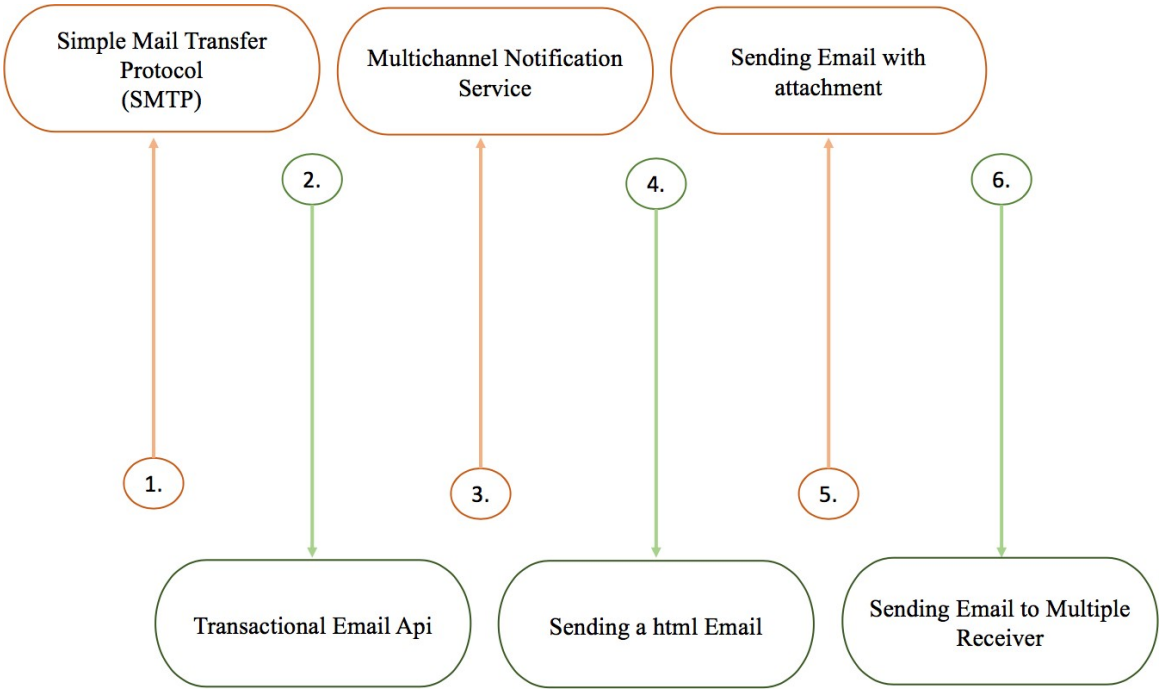


Fig 5.1: block diagram of proposed system

1. **Simple Mail Transfer Protocol(SMTP)**: SMTP is a text-based protocol that defines how email issent and received between mail servers.

Here's how it works:

i. You write an email and hit "send."
ii. SMTP picks up your email and delivers it to the recipient's email service.
iii. The recipient's email service then puts the email in their inbox.

SMTP is important because it makes sure your emails get to the right place, even if the recipient is usinga different email service than you.

2.Transaction Email Api: Transactional Email APIs are tools that allow developers to integrateemail sending functionality into their applications or websites. These APIs are used to send emails automatically in response to specific actions or events, such as user sign-ups, password resets, purchase confirmations, or delivery notifications.

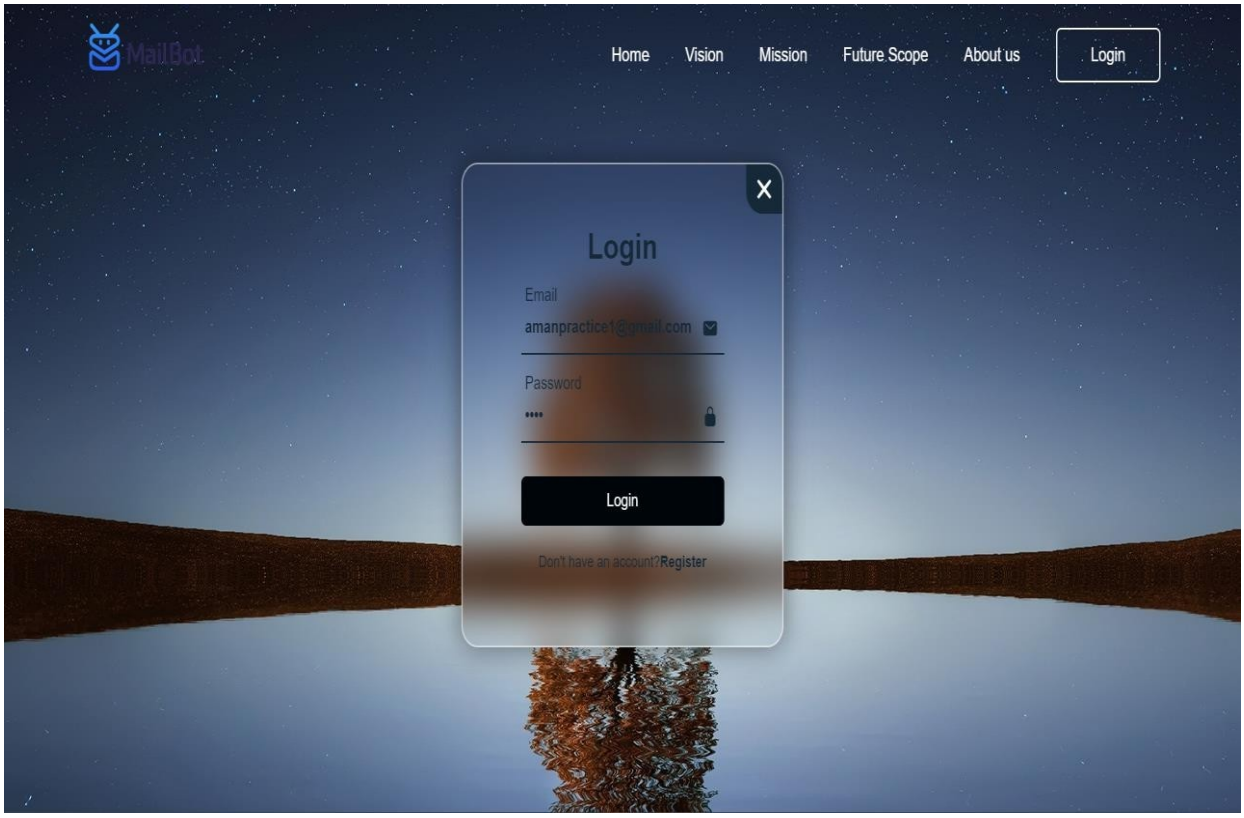
3.Multichannel Notification Service: Multichannel Notification Service is a tool that helps businesses send messages to their customers through various communication channels such asemail, SMS, push notifications, and more. It enables businesses to reach their customers on different platforms, ensuring that important information or updates are delivered effectively.

4.Sending a html Email: Sending an HTML email means sending a more attractive and visually enhanced email that can include images, colors, and different layouts to make it moreengaging and interesting for the recipient.

5.Sending Email with attachment: Sending an email with an attachment is like you can include a file,like a picture or a document, along with your email message. This allows you to share additional information or materials with the recipient that can't be easily included in the main body of the email.

6.Sending email to multiple receiver: Sending an email to multiple recipients is like sending thesame message to many people at once. It's similar to writing one letter and making copies to send to different friends or colleagues. This way, you can easily share the same information or messagewith multiple people without having to write separate emails for each person

5.2 Implementation



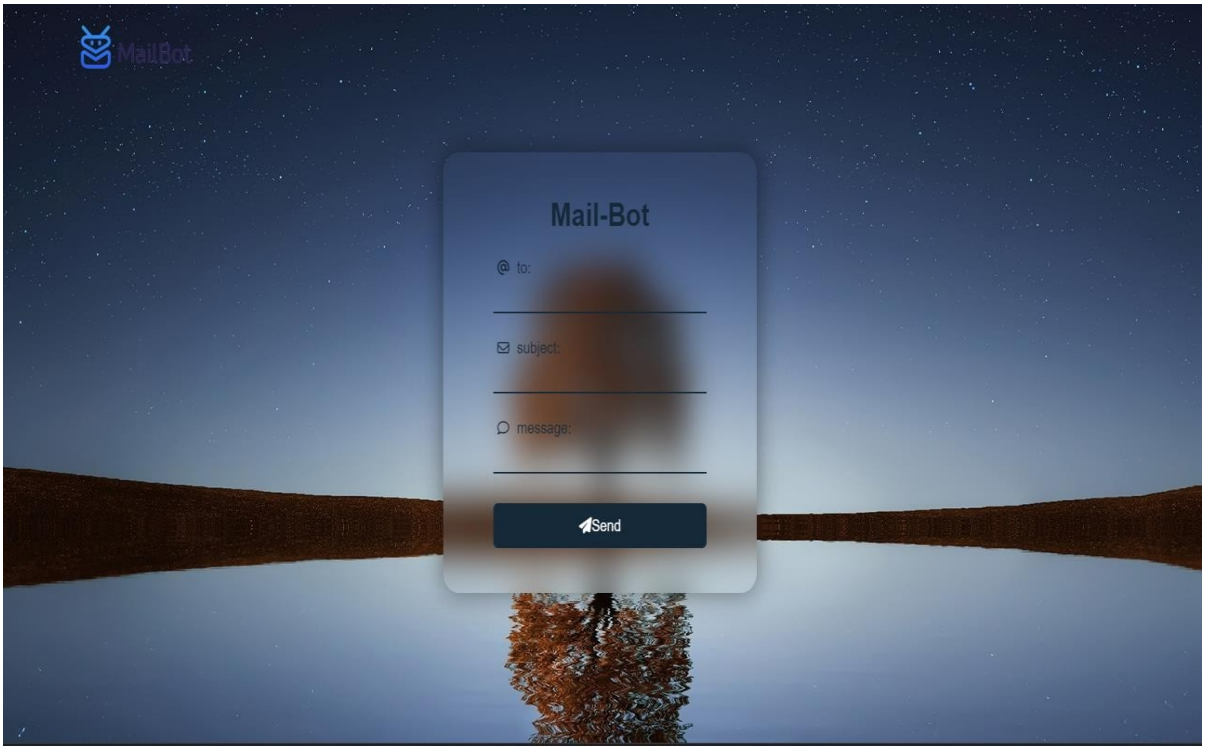


Fig 5.2,5.3: Implementation of the proposed system.

CHAPTER 6

Conclusion

Conclusion

The "Automated Email Sender " Automated email senders have become invaluable tools in our fast-paced digital age, streamlining communication and enhancing productivity for individuals and businesses alike. Through this technology, we can efficiently deliver important messages, updates, and notifications, ensuring that recipients receive timely information. With their ability to schedule emails, personalize content, and track engagement, automated email sendersoffer a versatile solution for a wide range of applications For example, you can use Python withDjango or Flask, Ruby with Ruby on Rails, JavaScript with Node.js, or PHP.

REFERENCES

Research paper

[1 Mujtaba, Ghulam, Liyana Shuib, Ram Gopal Raj, Nahdia Majeed, and Mohammed Ali Al-Garadi. "Email classification research trends: review and open issues." *IEEE Access* 5 (2017): 9044-9064.

[2]. Deccio, Casey, Tarun Yadav, Nathaniel Bennett, Alden Hilton, Michael Howe, Tanner Norton, Jacob Rohde, Eunice Tan, and Bradley Taylor. "Measuring email sender validation in the wild." In *Proceedings of the 17th International Conference on emerging Networking EXperiments and Technologies*, pp. 230-242. 2021.

[3Patil, Bhavani V., and K. Sreelakshmi. "Implementation of Voice Based E-Mail System for Visually Challenged." In *2022 International Conference on Futuristic Technologies (INCOFT)*, pp. 1-9. IEEE, 2022.

[4]. Yu, Beiyuan, Pan Li, Jianwei Liu, Ziyu Zhou, Yiran Han, and Zongxiao Li. "Advanced analysis of email sender spoofing attack and related security problems." In *2022 IEEE 9th International Conference on Cyber Security and Cloud Computing (CSCloud)/2022 IEEE 8th International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pp. 80-85. IEEE, 2022.

[5]. Savaliya, Bansi R., and C. George Philip. "Email fraud detection by identifying email sender." In *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pp. 1420-1422. IEEE, 2017.

Links

https://stackoverflow.com/

https://www.youtube.com/watch?v=MV-Aqkjju64

https://app.elasticemail.com/login

https://smtpjs.com/

https://tailwindcss.com/