CHAPTER ONE: **INTRODUCTION.**

Note: Remove anything on red underline and include everything in green

The term “email management system” is generally referred to as a system which supports the creating, sending and receiving of messages through a computer system. It enables messages to be sent online to general or private directories and electronic mail boxes by the use of a unique system address [11]. [1] It also involves the systematic control of the quality and quantity of electronic messages that are sent from within, and received by corporate organizations which is are an entity ies, comprises ing multiple people, such as an institution or an association that has a collective goal and is linked to an external environment.

The existing email management system works in such a way that a new user submits his/her registration form and the details in the form are stored in the database. Any time the user wants to enter into his account; email address and password used to register the account are required. There are some actions that logged in users can perform which is to compose email in which during the process the user will be asked for the email address of the recipient, the subject of the message, the body to of the message and files to be attached attachment files if any. The user can click on send button or save the email as a draft. Another action a user can perform is to view receive emails which resides in the inbox folder and can also can delete emails.

The problem with the existing system is that a user does not have the freedom to schedule email to be sent or received at a time and date specified by a user.

The main potential of email scheduling management system is to improve communication and also make sure that message is delivered appropriately within and outside the organization. This need to improves the productivity of both clerical and managerial office employees. In [6] the paper estimated that email conversation generally costs 75% less than a phone conversation and the time it takes to respond to an email is significantly less than phone call. Another reason for interest in email management systems is that it will increase the organizational decision making and information needs. The aim of this work is to develop an email scheduling management system for corporate organizations. The email system in addition to routine functions of email management systems will have a sub-system in it which the system will be used to schedule emails to be sent automatically at a specified time and date. In the future, this email management system may be the only feasible way to deal with information processing in increasing complexity and rapidly, thus changing organizational environments. As stated in [2] that email policies of the past decade have tended to make individual users responsible for managing their own emails. Recent studies have highlighted the shortcomings of this approach in guaranteeing organization wide compliance with records and other requirements. In large organizations, email is managed inconsistently if left to end users, because individuals exercise various levels of discipline and use their email accounts differently. [1] Published their study on email use within Lotus. Like [3], the paper found that email was being used for several tasks in addition to basic communication, a phenomenon called “email overload”. They paper also studied how people handled email overload when it comes to filing messages and classified classifying people as no filers(people who don’t clean up their inbox but use searching tools to manage it), frequent filers (people who constantly clean up their inbox), and spring cleaners (people who cleaned up their inbox once every few months).

The reason why the proposed system is needed is that with scheduling system it will help the directors or managers of corporate organizations to plan ahead of events, by effective scheduling of a mail which will be sent before or on the day of the event. Also members of such organisations can take advantage of the system to prepare their mails containing their opinions or suggestions regarding important issues in advance and schedule it to be sent at on the date and time of events such as board or executive meetings. The only thing that the user needs to do is to compose a mail by giving it time and date then save it in the database. Once the date and time is due, the mail will be sent appropriately to the specified receiver recipient.

* 1. **Statement of Problem**

1. Loss of mails in the organization due to inappropriate documentation/routing.
2. Registration of mail parcel takes much time.
3. Time is wasted in managing mails without a management system

Often important mails are received in advance and forgotten at the time it is needed

* 1. **Aim of the project**

The aim of the project is to develop an email scheduling system that will help the manager and staff of a corporate organizations to send email at a specified time, which will help to increase organizations productivity.

* 1. **Objectives**

Almost all the organizations in the country are now using modern computer technologies to deliver services to both members and non-members of the organization, Using this computer based technologies is saving substantial amount of man hour in record preservation, tracking and tracing.

The objectives of this work are:

1. To develop effective email scheduing system insuring rapid handling and accuracyte in delivery of emails to both members and non-members of the corporate organizations.
2. To create email reminder subsystem to enable members of organizations remember event dates and time and send emails to his staff on the appropriate dates in respect of the events. The system will help the directors to remember event dates and send emails to his staff on the appropriate dates in respect of the events.
3. Desgn a dtatabase of emails which will help to manage current emails and archive old but important mails. Delivering scheduled email to the receiver at appropriate time.
4. Present a statistical report of mails according to dates and time Preventing important emails from getting lost from the database.
   1. **Significance of the project**

The system will be of great importance to directors or managers of corporate organization, Staff of corporate organization and the Government.

The System will help the directors or managers of corporate organization to improve email security and data retention policies through centralized administration. Improve the end user experience by delivering uninterrupted access to emails in other words making sure that emails are delivered to users at appropriate time. The system will help top level staff such as director to remember event dates and send emails to his staff on the appropriate dates in respect of the events.

The system will help the staff of corporate organization to organize their daily activities, boost efficiency, find business opportunities and work more closely with customers and suppliers. The system will serve as a good means of record-keeping without actually record-keeping and having a physical to-do list.

To the government, it permits officers to send and receive personal e-mail messages using the government accounts provided via the government email system.

CHAPTER TWO: **LITERATURE REVIEW**

**2.0 INTRODUCTION**

This chapter presents the theoretical background of the study and the technologies used. The literature review is also discussed.

**2.1 Theoretical background**

2.2.1 Conceptual Background

The idea of mail started in other to enhance communication between two people, organizations, etc. It is done in the sense that the sender writes a letter which will contain the address of the receiver together with his own address is its informal letter, after which he/she proceeds to poll station to drop it in the mail box, letter is appropriately delivered to the receiver address. This had been a traditional way of sending a mail of it takes much time to send and receive mails, this lead to developing and email systems, which is a means by which sender sends a mail to the receiver using an electronic mail which is very fast means of sending mails. But still there is a need to classify the mails. According to [10] MailCat which is an intelligent assistant that helps usersorganize their e-mail into folders. It uses a text classifier to learn each user’s mail-filing habits. MailCat uses what it learns to predict the three folders in which the user is most likely to place each incoming message. It then provides shortcut buttons to file each message into one of these three folders. When one of MailCat’s predictions is correct, the effort required to file a message is reduced to a single button click.When people use electronic mail, they can communicate even when they are not physically or close [9]. Thus, it is not surprising that most studies report that the use of electronic mail increases organizational communication. [12] Provides a method for displaying information that represents a quantity of received email messages that have been held for processing by a specific entity for a predefined time period in an email response management system. In the implementation, the method includes determining first value associated with a first quantity of received email messages that have been held for processing by the specific entity for a first predefined time period, wherein each email message has been sent from end user. According to [13] Managing email over the long term is challenging because of email’s essential characteristics.An email originates in an electronic format, but it can also exist in a specific type of computer file or file format. Email tends to reside in a proprietary file format in an email system; although there is interaction between email software systems, email exists as a file format that is owned and controlled by a single software company and is not easily exportable to another environment. For these reasons, long term or permanent emails must be created in or converted to non-proprietary formats for preservation.

2.2.1 Technologies used

The technology used in this project is server/client technology. These include (Xampp, PHP, CSS, MySQL, and HTML) and relational database technology.

Xampp is used in this system as a local host that help the developer to test the application locally to know how the system will behave when deployed online. It is an integrated server package of MySQL, and PhpMyAdmin, that run from removable drive. It is a database, File Transfer Protocol (FTP) and mail server package for windows. It requires no configuration and has control panel starting/stopping mode.

HTML is a language for describing web pages. It stands for Hypertext Mark-up **Language** and it is the language upon which all web pages on the "www" (World Wide Web) are based. The HTML enables your web browser to be viewed and also enhance user computer interface which add interaction and relationship attributes to build and organize its codes.

PHP (Hypertext Pre-processor) is a server side scripting language that can be written into your HTML scripts and used to create dynamic web pages.

**CSS** (Cascading Style Sheets) is a collection of formatting rules that control the appearance of content in a web page. Using CSS styles to format a page separates content from presentation.

MySQL is an Open Source relational database management system (RDBMS) that uses Structured Query Language (SQL). In English, MYSQL is a server-side program (which means it is installed on a server, not onto your computer) where you can store your databases and access them in the internet.

**2.2 Review of related literature**

According to [2] Email based communication over the course of globalization in recent years has transformed into an all-encompassing form of interaction and requires automatic processes to control email correspondence in an environment of increasing email database. Relevance characteristics defining class of email in general includes the topic of the mail and the sender of the email along with the body of email. Intelligent reply algorithms can be employed in which machine learning methods can accommodate email content using probabilistic methods to classify context and nature of email. This helps in correct selection of template for email reply. Still redundant information can cause errors in classifying an email. Natural Language Processing (NLP) possess potential in optimizing text classification due to its direct relation with language structure. An enhancement is presented in this research to address email management issues by incorporating optimized information extraction for email classification along with generating relevant dictionaries as emails vary in categories and increases in volume. The open hypothesis of this research is that the underlying concept to fan email is communicating a message in form of text. It is observed that NLP techniques improve performance of Intelligent Email Reply algorithm enhancing its ability to classify and generate email responses with minimal errors using probabilistic methods. Improved algorithm is functionally automated with machine learning techniques to assist email users who find it difficult to manage bulk variety of emails.

Email has been one of the most commonly used tool for communication in the recent years and email management has evolved as a major challenge due to prevailing situation of online email congestion. [5] Presents a novel algorithm for automatic email response methodology in an Email Management System to minimize email overload. The proposed model uses Bayes classifier to categorize emails into classes and generate suitable replies to these classes using information extraction and template filling. Our research aims to intelligently automate email response using Naïve Bayesian classification and formulate probabilistic dictionaries for accurate information extraction. This research will help in reducing email overload and unavoidable congestion by employing novel email response architecture for an email management system.

The Semantic Web was designed to represent the enormous data that is existing on the World Wide Web in a machine readable format. [4] Research shows the long period of time that was spent on the Emails for communication and information exchange. Adding the semantics to the existing Email systems could not only provide for the valuable usage of time and resources, but also refreshes the meaning of Email communication. The presented research work examines the ontology extraction process from the Email systems adopting scalable pattern rules that is based on the extracted techniques. The proposed architecture is designed to handle the unstructured Emails and the ontologies that are extracted from the Email which is divided into four main components as follows: the Ontology Learning Component, the Management Component, the Semantic Email Component and the Client Side Plugin.

[7] Develops a ripper classification algorithm which is used in automatic filtering of email. Its architecture is based on rule based structure to sort email. Ripper has the ability to automatically generate rules for selecting keywords instead of manual selection. Its advantage is that it is fast and able to deal with a large set of email attributes. However keyword extraction rules have to be constructed for every possible class and it is easy for emails to be mixed up or irrelevant attribute information is extracted. These extraction rules only makes binary decisions, so there predictions are not fully deterministic as strict rules may cause emails to get mixed up in wrong classes. This is caused by attributes competing against each other for possession of an email messages. Such a system is also unable to learn adaptively. Whenever the attributes of the email are changed, the rules will be recreated from ground up in order accurately assign an email class. This task needs time to complete and other dependent task like automatic email reply is directly affected.

[3] Describes a series of interviews that examine the ways that professional office workers use electronic mail to manage their daily work. The purpose is to generate hypotheses for future research. A number of implications for the design of flexible email systems are discussed. Two principal claims are made, first, the use of electronic mail in strikingly diverse, although not infinitely so; Individuals vary both in objective measures of email use and in preferred strategies for managing work electronically. Feelings of control are similarly diverse and are related to the size of the user’s inbox, number of folders, and subscriptions to distribution lists. This diversity implies that one’s own experiences with electronic mail are unlikely to provide sufficient understanding of other’s uses of email. According to the literature, email designers should thus seek flexible primitives that capture the important dimensions of use and provide flexibility for a wide range of users. The second claim is that electronic mail is more than just a communication system. Users archive messages for subject retrieval, prioritize messages to sequence work activities, and delegate tasks via email. Taxonomy of work management is proposed in which email is used for information management, time management, and task management activities.

In this paper [8], the connection between the distinctive association techniques and the time important to utilize a specific methodology is shown by a numerical model in light of keystroke-level examination. The model gauges that time utilization for filing and recovering email messages for singular clients. Other than clarifying why clients create diverse techniques to sort out email messages for singular clients. Other than clarifying why clients create different techniques to sort out email messages, the model can likewise be utilized to prompt clients separately when to begin utilizing envelopes, clean messages, take in the inquiry functionality and utilizing channels to store messages.

The development of the proposed system will help the directors or managers of corporate organization to plan ahead of events, by effective scheduling of a mail which that will be sent before or on the day of the event. The only thing that the user needs to do is to compose a mail by giving it time and date then save it in the database. Once the date and time is due, the mail will be sent appropriately to the specified receiver.

CHAPTER THREE:**SYSTEM ANALYSIS AND DESIGN**

**3.0 INTRODUCTION**

The knowledge of system analysis is to establish precisely what the users of the system want. The main part of this stage is communication between the users and the software developers or engineer. System analysis and design relates to shaping organizations, improving performance and achieving objectives for profitability and growth. The emphasis is on system in action, the relationships among subsystems and their contribution to meeting a common goal.

State appropriate research methodology and research method. Refer to the guideline. Be reminded that OOAD is a design tool and not research methodology

The proposed system is built with an object-oriented analysis and design (OO  
AD) methodology. The purpose of Object oriented analysis and design can be described as:

* Identifying the objects of the system.
* Identifying the relationship of the objects
* Make a design which can be converted to executable using OO languages.

**3.1 Analysis of Existing System**

In the present fast growing world, new email domains are being developed and hosted on the web, to simulate the process of sending and receiving emails. The existing system was developed in which new user’s accesses the email domain provided online. Through the registration form provided one can be able to provide details according to the requirements of the form, after which submit button is clicked and the details provided is being sent to the database which store the information of the user. Already registered users have to provide their email address and password which must exist in the database before they can be able to login into the homepage of the system. In the homepage, the user can be able to send email by first filling the form for composing email which include the receiver email address, the subject of the message, the body of the message, attachment file if any and click on send button. Each sent email is being saved in sent folder. A user can decides to compose a message and save it as a draft in which later he/she will login and send the email to the specified receiver.

Each registered users have an inbox folder which contains messages received from other users. In the folder the most recent messages resides at the top of the inbox. The user can decide to view the message by clicking on the message, or delete messages. The problem with the existing system is that there is no provision for email scheduling which causes time wasted in sending saved email;

The drawbacks observed after detailed analysis of the system includes the following:

* Time Wasting: It takes much time visiting the system to send saved email.
* Low Productivity: Sometime the director may forget to send an email at specified time which causes low productivity of the organization.

**3.2 Analysis of the Proposed System**

This chapter analyses the email management system with a detailed understanding of the software being designed as well as detailed description of the system’s design. The knowledge of system analysis is to establish precisely what the users of the system want. At this stage we try to establish the requirements for the system, making sure that the system does what its prospective organization wants.

The design tool used in this work is Unified Modeling Language (UML) because it is a graphical language for modeling software. It is a set of diagrams that can be used to specify, construct, visualize, and document software design. UML has diagrams to assist in every part of the application development process, from requirement gathering through design, and into coding, testing and deployment. The UML diagrams used in this software are USE CASE diagram and CLASS diagrams.

The proposed system has been designed to overcome all the draw backs and also has enhanced feature, which is to develop a scheduling system that will all allow user to save email with time and date to send it.

Use Case Diagram

The main purpose why use case diagram in UML is used is to demonstrate the different ways that the user might interact with the system and also to represent the functionality of the system. A use case diagram can summarize the details of the users of the system and their interactions with the system. An effective use case diagram can help to discuss and represent:

* Scenarios in which the system interact with people, organization.
* The goals that the system will help the actors (users) to accomplish.
* Scope of the system.

The use case diagram of the proposed system is shown in fig 3.1

<<include>>

<<extend>>

Compose Mail

Attach files to Mail

Registration

Login

Schedule Mail

Send Mail

Manage all Users

Manage Database

View Mail

Delete Mail

Edit Mail



End User

Admin

Figure 3.1: Use case diagram  
Make the lines to align with the actors eg admin

Class Diagram

The class diagram is a static diagram. It represents the static vies of the device. Class diagram is not most effective used for visualizing, describing and documenting distinct factors of a device but additionally for constructing executable code of the software program application. Class diagram describes the attributes and operations of a class and additionally the limitations imposed on the gadget. The class diagrams are extensively used in the modeling of object-orientated structures because they are the most effective UML diagrams, which may be mapped at once with object-orientated languages.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints.

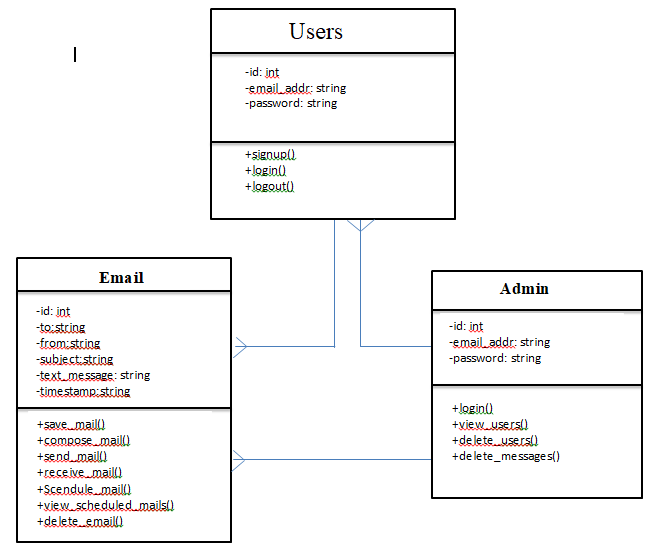


Figure 3.2: Class diagram

**3.3 Design of the Proposed System**

The proposed system is a web-based system that will authenticate the user with the information supplied. Before sending message and scheduling of message, it makes sure that senders and recipient’s details are collect before sending the message and attachment file. It enhances appropriate delivery of email to members of the organization at a specified time. The following are used in the design and implementation of the proposed system: Input design, output design, database design, algorithm design etc.

**3.3.1 Input Design**

The input design is the connection between the system and the user. It comprises the developing specification and procedures for data preparation. The input design describe the necessary information the user are required to enter into the system Graphic User Interface (GUI), before establishing connection with the network tier.

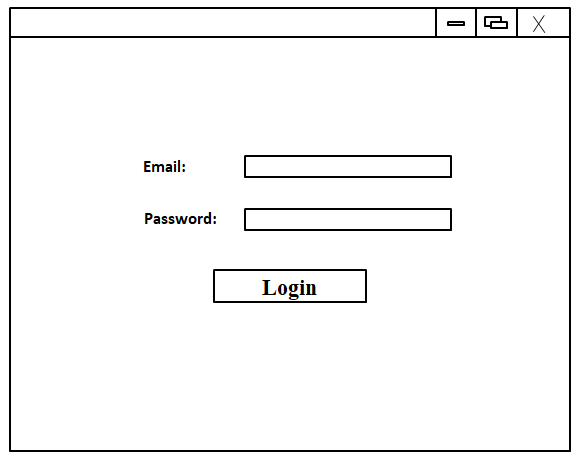
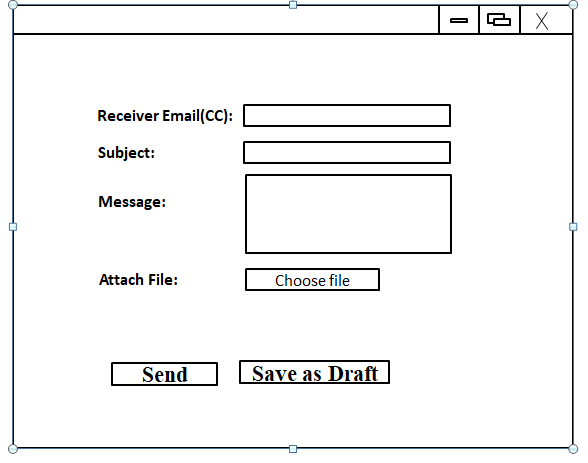


Figure 3.3: login form

 Figure 3.4: Compose mail form

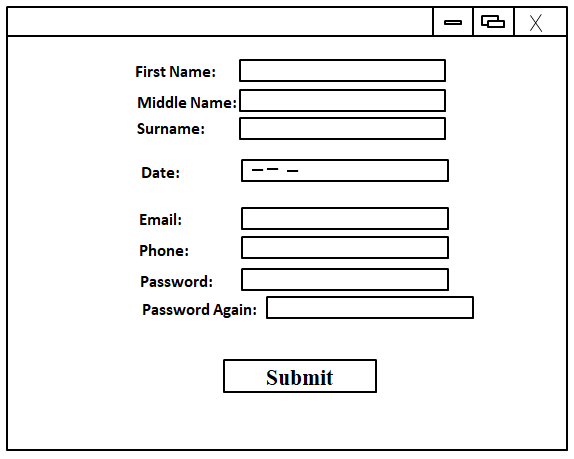
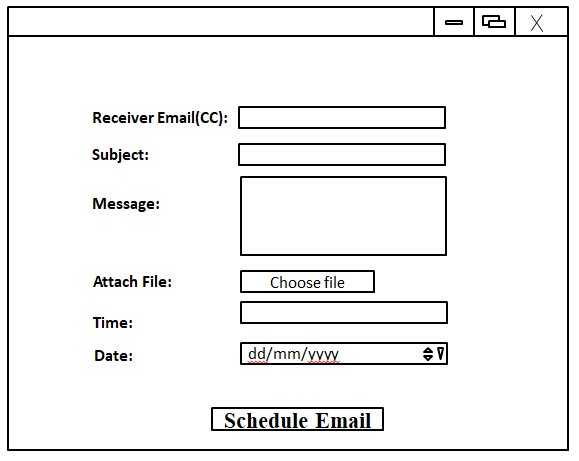


Figure 3.5: Users registration form

 Figure 3.6: Schedule mail form

**3.3.2 Output Design**

The results of processing are communicated to the users and to other system through output. A quality output is one, which meets the requirements of the end user and presents the information clearly. The output of this system is a response message that may take any of the form:

* Success response.
* Failure response due to:
  + Internet unavailability
  + User entering email address and password that does not exist in the database
  + Invalid recipient email address

**3.3.3 Database Design**

In any good database design, effort should be made to remove completely or at best reduce redundancy. The database used in this project is MySQL which is a relational database that uses tables to store information about the various entities in the system. The tables used for the system include the following:

**Table 1: Users**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Data type | Size | Null | Action | Extra |
| Id | Int | 10 | No | Primary key | Auto\_increment |
| First\_name | Varchar | 20 | No |  |  |
| Middle\_name | Varchar | 20 | No |  |  |
| Surname | Varchar | 20 | No |  |  |
| Category | Varchar | 15 | No |  |  |
| Date\_of\_birth | DATE |  | No |  |  |
| Email | Varchar | 50 | No |  |  |
| Phone | Varchar | 11 | No |  |  |
| Password | Text |  | No |  |  |
| Secrete\_question | Varchar | 50 | No |  |  |

**Table 2: messages**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Data type | Size | Null | Action | Extra |
| Id | Int | 10 | No | Primary key | Auto\_increment |
| Sender\_email | Varchar | 50 | No |  |  |
| reciever\_email | Varchar | 50 | No |  |  |
| Subject | Varchar | 100 | No |  |  |
| message | Text |  | No |  |  |
| File\_name | Varchar | 200 | No |  |  |
| Date\_sent | DATE |  | No |  |  |
| Time\_stamp | Varchar | 35 | No |  |  |
| open | int | 11 | No |  |  |
| Sent | Int | 11 | No |  |  |
| Not\_sent | Int | 11 | Yes |  |  |
| Sender\_deleted | Int | 11 | No |  |  |
| Receiver\_deleted | Int | 11 | No |  |  |

**Table 3: saved**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Data type | Size | Null | Action | Extra |
| Id | Int | 10 | No | Primary key | Auto\_increment |
| from | Varchar | 50 | No |  |  |
| Sender\_email | Varchar | 50 | No |  |  |
| Subject | Varchar | 100 | No |  |  |
| message | Text |  | No |  |  |
| Date\_recieved | DATE |  | No |  |  |
| Time\_recieved | Varchar | 35 | No |  |  |
| File\_path | Text |  | No |  |  |
| Open | Int | 11 | No |  |  |

**Table 4: Admin**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Data type | Size | Null | Action | Extra |
| Id | Int | 10 | No | Primary key | Auto\_increment |
| email | Varchar | 50 | No |  |  |
| password | Varchar | 50 | No |  |  |

**Activity Diagram of Schedule Email**

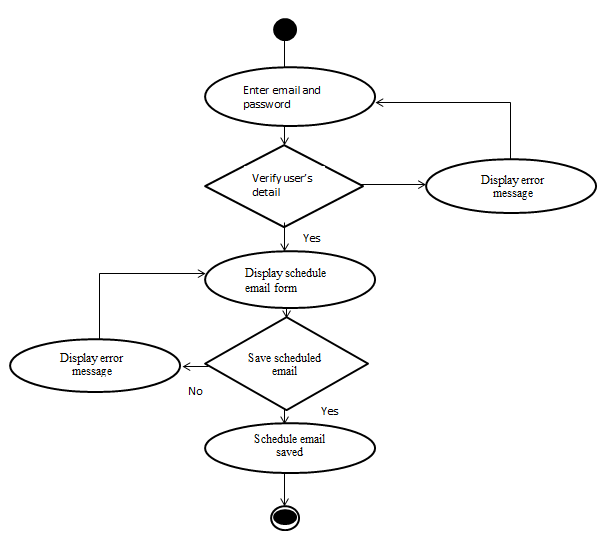


Figure 3.7: Activity diagram of email scheduler

**3.3.4 System Architecture**

The architecture of the system is designed in three-tier. The tier is presentation tier, middle tier and data tier. The presentation tier is the user interface and it is designed using HTML and CSS. The middle tier connects the presentation tier and the data tier together. It is designed using PHP. The data tier is the part of the system that is responsible for storage of data. The data tier is designed with MySQL DBMS server. The figure below shows the system architecture.

Server

Middle Tier Data Tier

HTML andCSS

PHP

MySQL DBMS (Model)

Presentation Tier

Figure 3.8: System architecture

CHAPTER FOUR:**SYSTEM IMPLEMENTATION**

**4.0 Introduction**

System implementation is the construction of the new system and delivery of that system into production that is, the day to day business. It can also be define as the practice of creating or modifying a system to create a new business process or replace an existing system.

**4.1 Choice of development Environment**

There are three development environment choices and these include: System platform, Integrated Development Environment and programming language.

System Platform is the underlying computer system on which application programs can run. Computer system includes both the hardware and software part of the system. The system platform of my web-based application program is personal computer and window operating system. An example of personal computer include: laptops, desktops, palmtops, notebooks etc. While example of windows operating system are window 98, window 2000, window XP, window 7 and window 8.

The integrated Development Environment (IDE) is a programming environment that has been packaged as an application program typically consisting of a code editor, a compiler, a debugger and a graphic user interface (GUI) builder. My project application uses the following IDE: sublime text, xamp server and system browsers such as Google chrome, Mozilla Firefox etc.

The programming languages used in my project are HTML, CSS, PHP, MySQL and Bootstrap classes. These languages are collectively called web technology. The web technology is used because it has the following features:

* Easy to learn.
* PHP support Object Oriented Programming (OOP)
* Easy to understand
* Speed of development
* Supported platform environment
* Portability

**4.2 Implementation Architecture**

Sent

draft

View Schedule

Schedule Email

View users

Delete users

Delete Emails

Logout

User Login

Forgotten Password

Admin Login Login

Register

**Email Management System**

inbox

Logout

Figure 4 Implementation Architecture

**4.3 Software testing**

Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results.

The screen shots of my web-based program are shown below:

Figure 4.2 Show the form which enables the user to login into the system

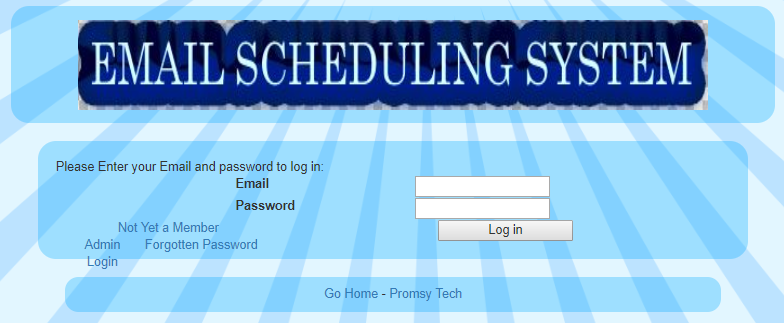


Figure 4.2 Screen shot for member’s login form

Figure 4.3 is a registration form that enables new users to register.

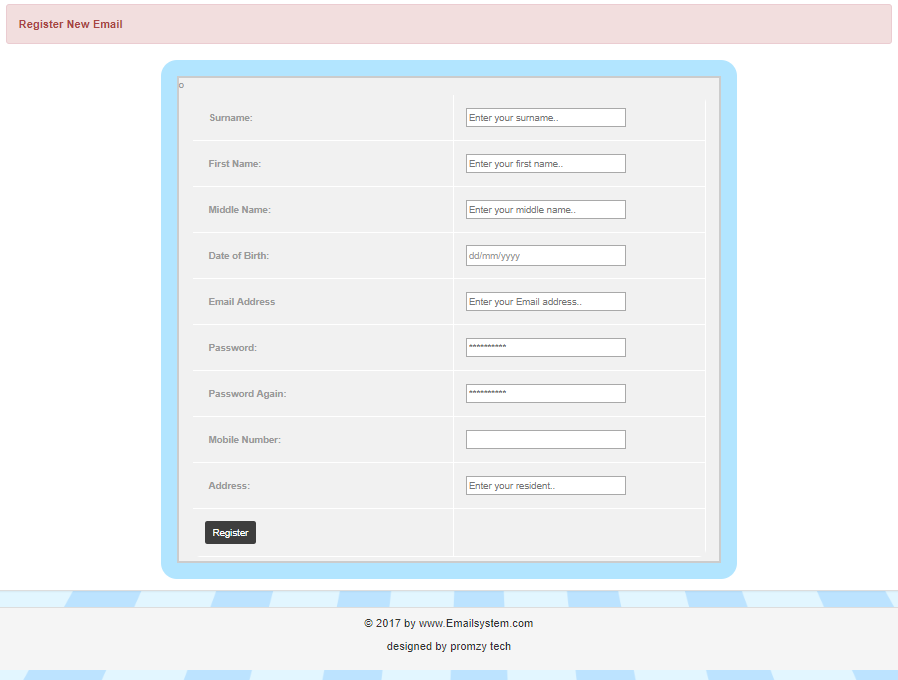


Figure 4.3 Screen shot of member’s Registration form

Figure 4.4 is the homepage of the system through which the users selects actions to perform.

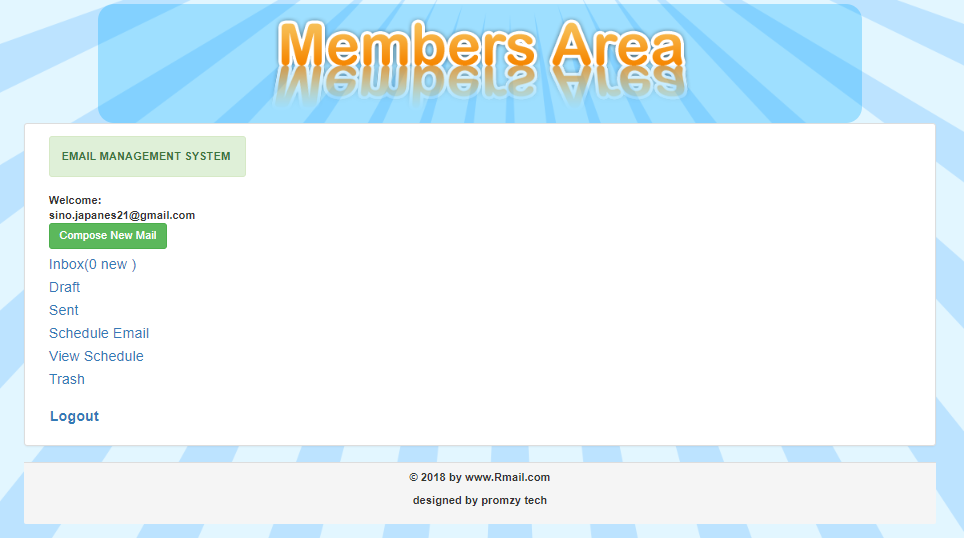


Figure 4.4 Screen shot of member’s homepage

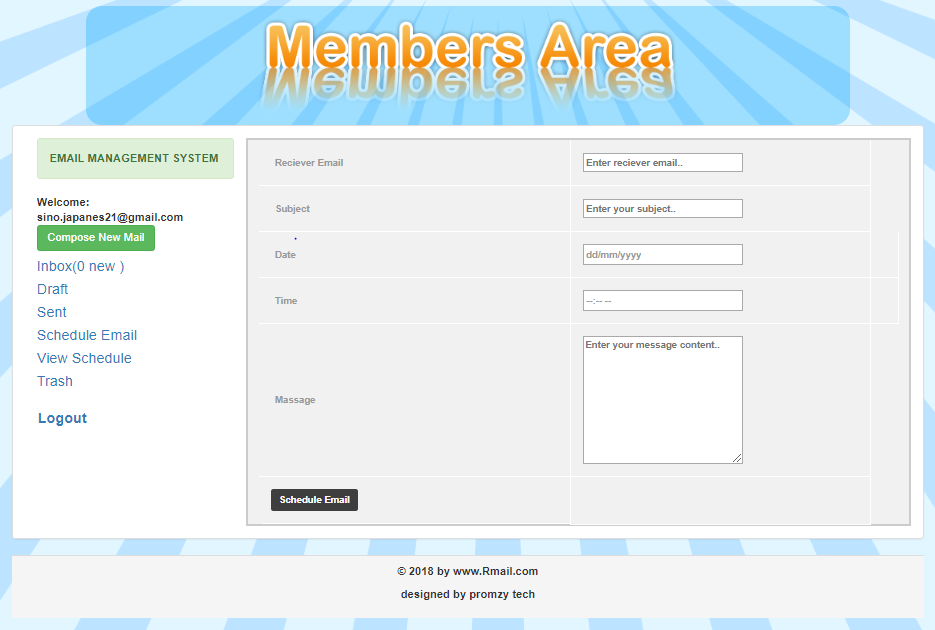


Figure 4.5 Screen shot of Schedule form

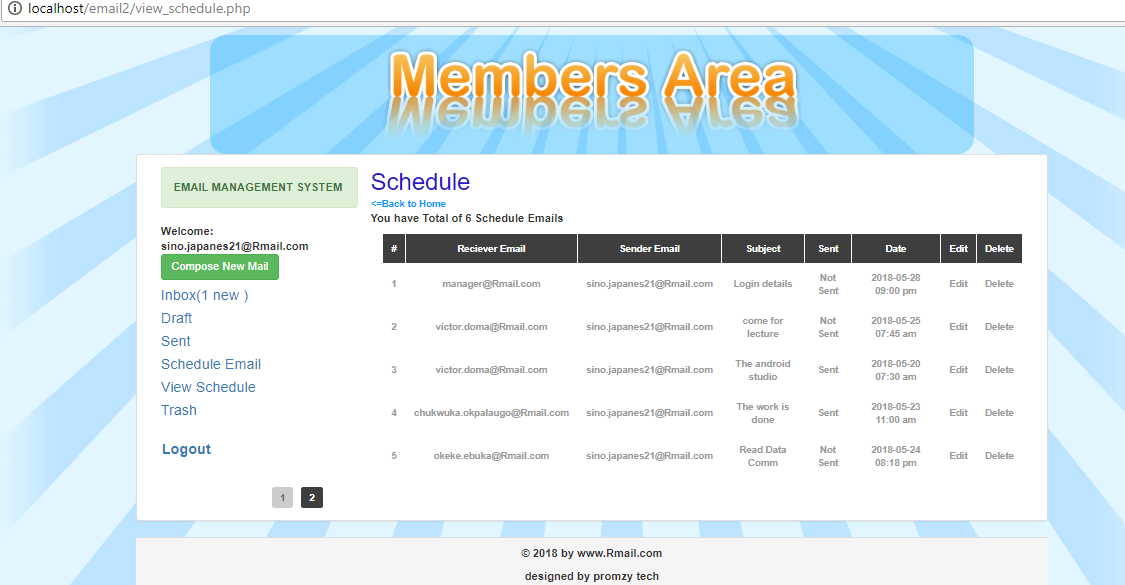
****

Figure 4.6 Screen shot for Schedule Emails both sent and unsent emails.

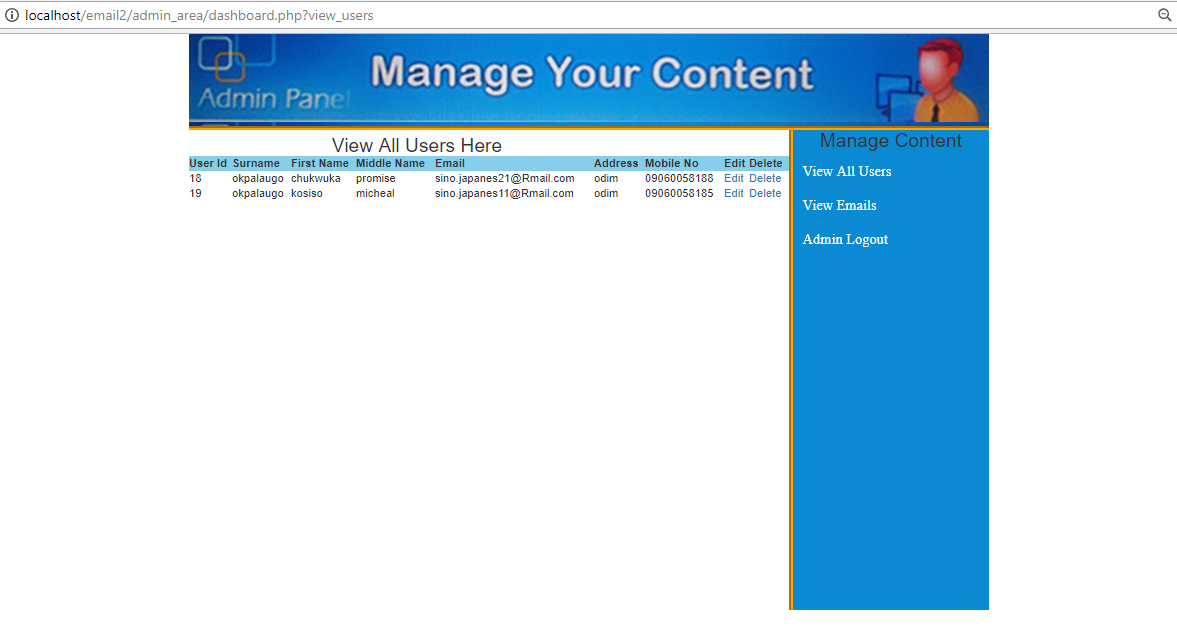


Figure 4.Admin Dashboard

**4.4 Documentation**

Documentation is a set of documents provided on paper, or online or other devices.

Software documentation is written text that accompanies computer software. It either explains how it operates of how to use it.

For the purpose of my project, the documentation is for the end users.

4.4.1 User Manual

In order to use this web application, the following steps are required:

* Open web browser, example internet Google chrome and type localhost/email2/index.php in the address bar.
* On the Login form provided, enter your email address and password if already registered or click register.
* After login to the email system; you can view receive emails by clicking inbox, view sent email by clicking sent, schedule email by clicking schedule email, view schedule email by clicking view schedule.
* There is a logout button though which logged in users can quit the system.

4.4.1 Source code listing

The source code of my project work is in Appendix A.

CHAPTER FIVE:**SUMMARY AND CONCLUSION**

**5.0 Summary**

Email management system is the use of integrated computer and communication systems to support information dissemination in an organization. This represents structured methods of handling communications through an integrated system that may include electronic message systems for communication, electronic filing systems. The system ensures delivery of scheduled mail to the recipient any time the service is through.

**5.1 Achievement of the work**

The system was able to schedule mails and send scheduled mails to the recipient based on the time and date assigned to the mail.

**5.2 Conclusion**

In harmony with the objective of this project which is to schedule mails and send scheduled mails on the specified timestamp. I feel satisfied to say that the defined objectives of this project have been actualized.

The system first checks the email requirements before scheduling the mail which is saved in the database. If it encounters error like incorrect recipient email address, it will display an error message to the user. Once the current timestamp corresponds with the timestamp saved in the database, the mail is appropriately delivered to the recipient email address. So, this system enables the manager/directors and staff of corporate organization to planning and communication enhancement.

**5.3 Recommendations**

Due to time constraint, the system has been developed as a standalone system. Further work can still be made on the email system by incorporating it in the organization system, also making the system to be available on a shared host. This will enable every member and non-member of the organization to register from a distance area.

**REFERENCES**

[1] Whittaker, S. and Sidner, C. Email Overload: Exploring Personal Information *Computing Systems:* CHI, 1996

[2] A. Al-Alwani, "Improving Email Response in an Email Management System Using Natural Language Processing Based Probabilistic Methods", *Journal of Computer Science*, vol. 11, no. 1, 2014.

[3] W. Mackay, "Diversity in the Use of Electronic Mail: A Preliminary Inquiry", *PRACTICE AND EXPERIENCE*, vol. 6, no. 4, 1998.

[4] M. Beseiso, A. Ahmad and R. Ismail, "A New Architecture for Email Knowledge Extraction", *Web & Semantic Technology*, vol. 3, no. 3, 2012.

[5] A. Al-Alwani, "A NOVEL EMAIL RESPONSE ALGORITHM FOR EMAIL MANAGEMENT SYSTEMS", *Journal of Computer Science*, vol. 10, no. 4, 2014.

[6] Deerfield Communications, Inc., "Email Management – Opportunities and Challenges", Michael Ryan.

[7] N. Ducheneaut and V. Bellotti, "Email as a habitat", 3333 Coyote Hill Road, PALO ALTO, CA 94304 - USA.

[8] O. Balter, *Keystroke Level Analysis of Email Message Organization*. Stockholm, Sweden, 2000.

[9] M. Sarbaugh-Thompson and M. Feldman, "Electronic Mail and Organizational Communication", *Organization Science*, vol. 6, no. 9, 1996.

[10]R. Segal and J. Kephart, *MailCat: An Intelligent Assistant for Organizing E-Mail*. Yorktown: Heights, 1999.

[11] "What is Email Management?", *Aiim.org*, 2018. [Online]. Available: http://www.aiim.org/What-is-Email-Management-EMM. [Accessed: 06- May- 2018].