

A Software requirements specification  
On

# Secure Email Sender

Submitted in Partial Fulfillment for the Award of Degree of Bachelor of Technology in  
Computer Science and Engineering from Rajasthan Technical University, Kota



**MENTOR:**

**Mr. Sumit Mathur**

(Dept. of Computer Science & Engineering)

**SUBMITTED BY:**

**Aditya Mishra (19ESKCS017)**

**COORDINATOR:**

**Mr. Kailash Soni**

(Dept. of Computer Science & Engineering)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

---

**SWAMI KESHWANAND INSTITUTE OF  
TECHNOLOGY, MANAGEMENT & GRAMOTHAN  
Ramnagar (Jagatpura), Jaipur – 302017**

**SESSION 2023-24**

# Table of Contents

1. Introduction
  - 1.1 Purpose
  - 1.2 Motivation
  - 1.3 Methodology
  - 1.4 Implementation
  - 1.5 Technologies to be used
  - 1.6 Overview
2. Overall Description
  - 2.1 Software Interface
  - 2.2 Hardware Interface
  - 2.3 Communication Interface
  - 2.4 Constraints
  - 2.5 E-R Diagram
  - 2.6 Use-Case Model Survey
  - 2.7 Architecture Design
  - 2.8 Database Design
3. Specific Requirements
  - 3.1 Use-Case Reports
  - 3.2 Activity Diagrams
4. References

## **Purpose**

To build a web app named **securegmail** which will help users to send emails using Gmail but in encrypted format and the receiver decrypts it back using our web app. We implemented the message encryption by Advanced Encryption Standard (AES) using python cryptography library

## **Motivation:**

While sending emails on Gmail we cannot be sure whether the information remains confidential since Google can still access the mails. We wanted to make sure that the content is only accessible to the desired recipient and no one else has access to the message shared.

## **Methodology**

1. Built UI using React Js
2. Used cryptography library provided by python for encryption and decryption purposes.
3. Used Flask as the backend framework.
4. Achieved secure key exchange through asymmetric cryptography.

## **Implementation:**

1. Sender provides his gmail email and password (google's third party authentication password)
2. Then write his message
3. Create a password for his message then send the mail
4. Sender sends the generated password to the receiver
5. Receiver enters the password provided by the sender
6. Enters the encrypted mail that he received on gmail
7. Finally after the key and encrypted message is verified the real message is displayed.

## **Tech stack used**

**React Js, Tailwind CSS, CSS :** For frontend

**Python, Flask:** For backend

**Cryptography Library:** For encrypting and decrypting the messages and for key exchange

# Overall Description

## 2.1 Software Interface

- **Client on Internet**  
Web Browser, Operating System (any)
- **Client on Intranet**  
Web Browser, Operating System (any)
- **Web Server**  
WASCE, Operating System (any)
- **Data Base Server**  
DB2, Operating System (any)
- **Development End**  
React, Flask, MongoDB

## 2.2 Hardware Interface

### Minimum Requirements:

| Client Side           |                                      |        |            |
|-----------------------|--------------------------------------|--------|------------|
|                       | Processor                            | RAM    | Disk Space |
| Internet Explorer - 6 | Intel Pentium III or<br>AMD -800 MHz | 128 MB | 100 MB     |

| Server Side |                                      |        |                                    |
|-------------|--------------------------------------|--------|------------------------------------|
|             | Processor                            | RAM    | Disk Space                         |
| RAD         | Intel Pentium III or<br>AMD -800 MHz | 1 GB   | 3.5 GB                             |
| DB2 - 9.5   | Intel Pentium III or<br>AMD -800 MHz | 256 MB | 500 MB<br>(Excluding Data<br>Size) |

### Recommended Requirements:

| Client Side           |                          |        |            |
|-----------------------|--------------------------|--------|------------|
|                       | Processor                | RAM    | Disk Space |
| Internet Explorer - 6 | All Intel or AMD - 1 GHZ | 256 MB | 100 MB     |

| Server Side |                          |        |                                 |
|-------------|--------------------------|--------|---------------------------------|
|             | Processor                | RAM    | Disk Space                      |
| RAD         | All Intel or AMD - 2 GHZ | 2 GB   | 3.5 GB                          |
| DB2 - 9.5   | All Intel or AMD - 2 GHZ | 512 MB | 500 MB<br>(Excluding Data Size) |

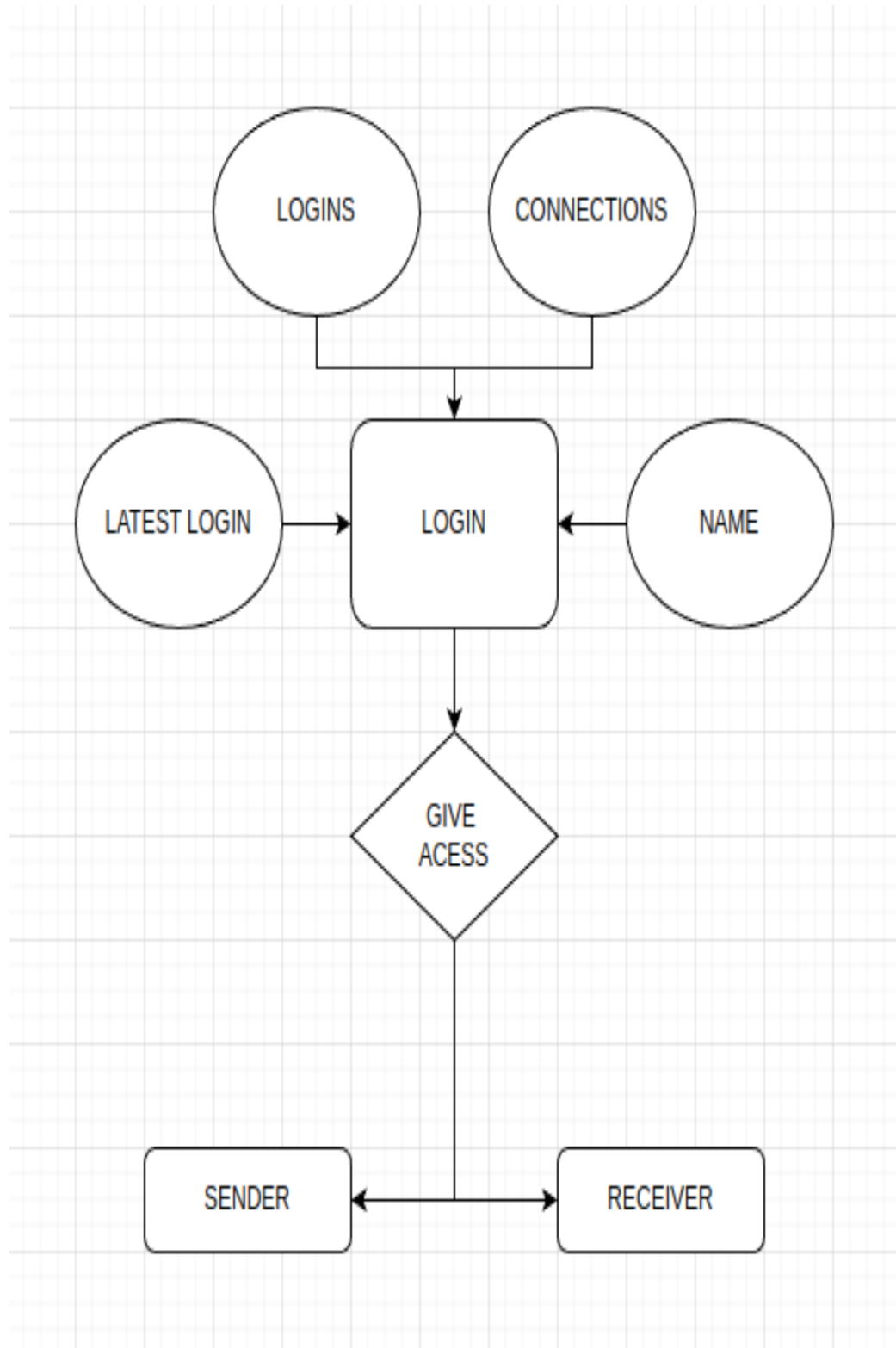
## 2.3 Communication Interface

- Client (customer) on Internet will be using HTTP/HTTPS protocol.
- Client (system user) on Internet will be using HTTP/HTTPS protocol.

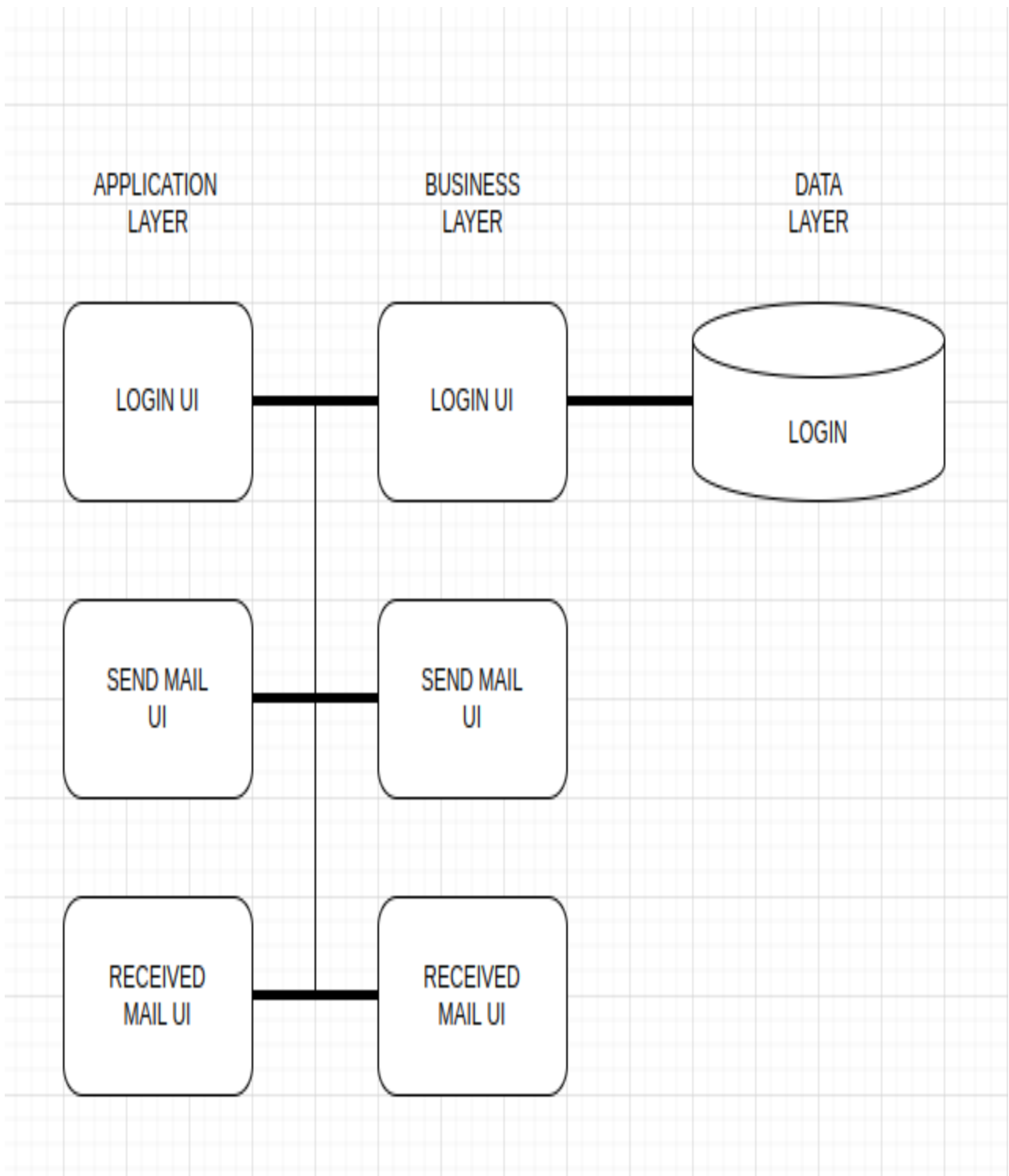
## 2.4 Constraints

- GUI is only in English.
- Login and password is used for the identification of users.
- Limited to HTTP/HTTPS.
- This system is working for single server.

## 2.5 E-R Diagram



## 2.6 Architecture Diagram





## 2.7 Database Design

|              |
|--------------|
| USER LOGIN   |
| NAME         |
| CONNECTIONS  |
| LOGINS       |
| LATEST LOGIN |

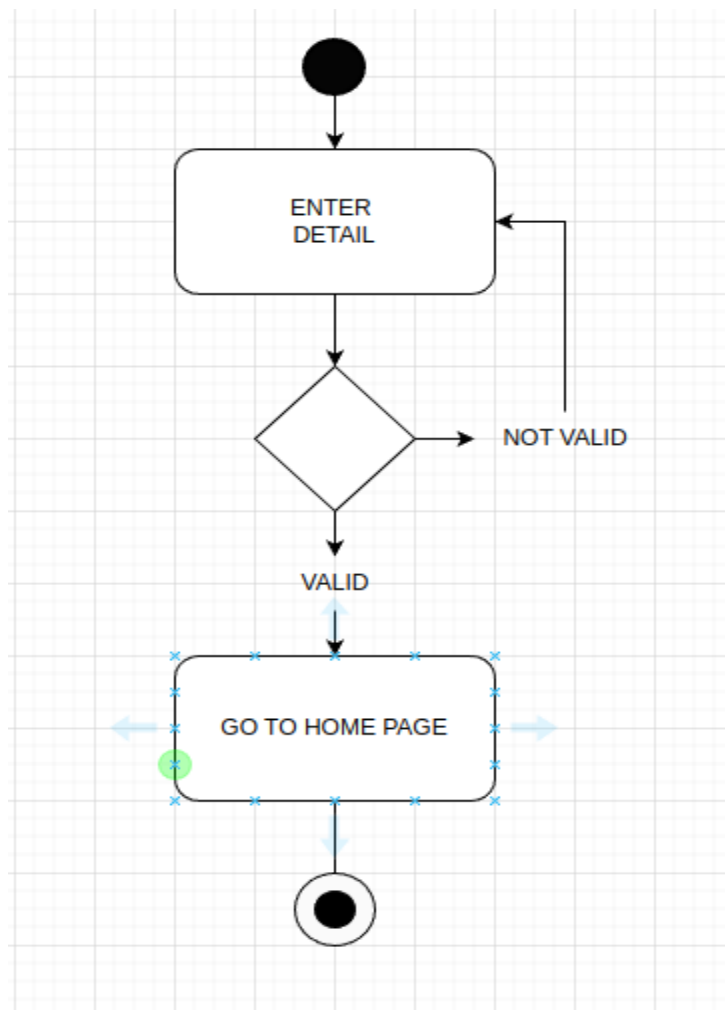
## Specific Requirements

### 3.1 Use Case Reports

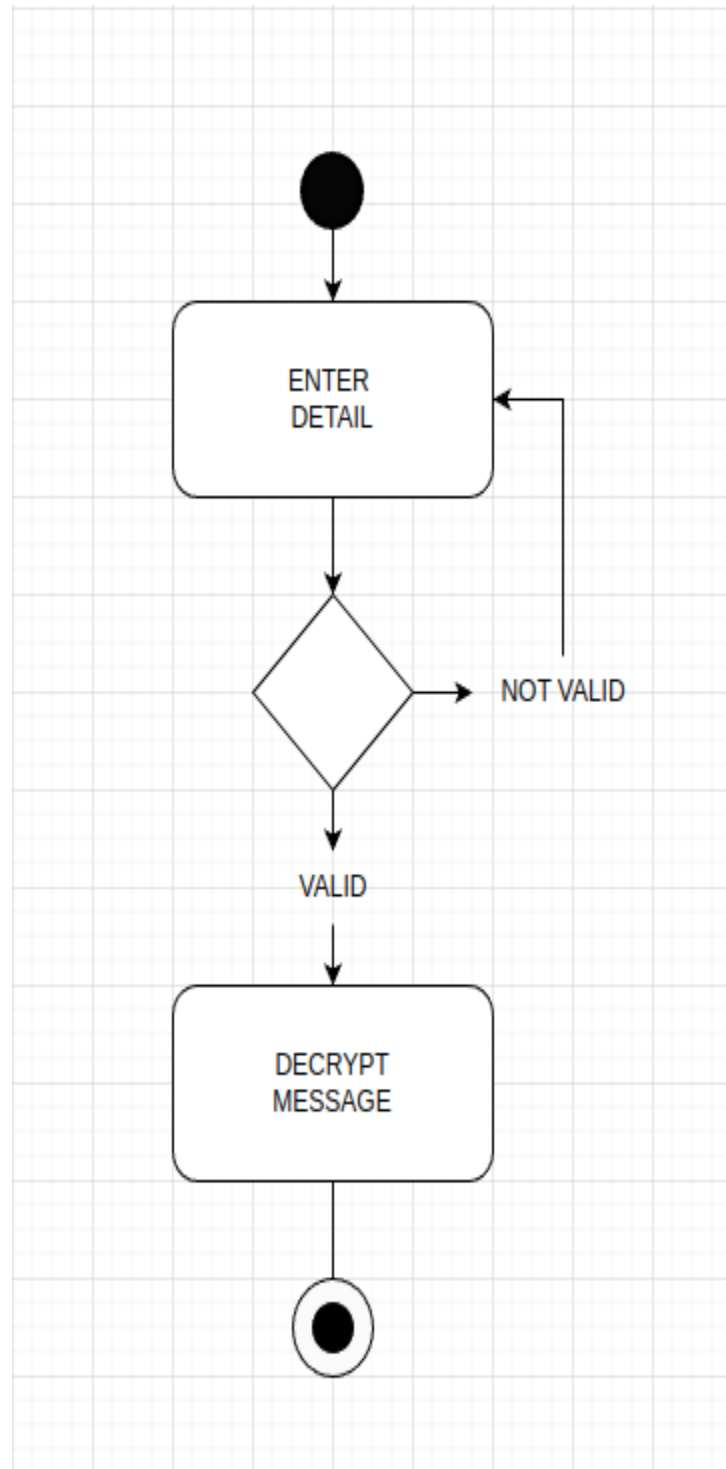
| Use Case     | Description   |
|--------------|---|
| SignIn       | The user has to signIn in order to access the application |
| Send Mail    | Every registered user can send encrypted mails.           |
| Decrypt Mail | Every registered user can decrypt received message        |
| Logout       | The user can logout if they wish to                       |
|              |   |

### 3.2 Activity Diagrams

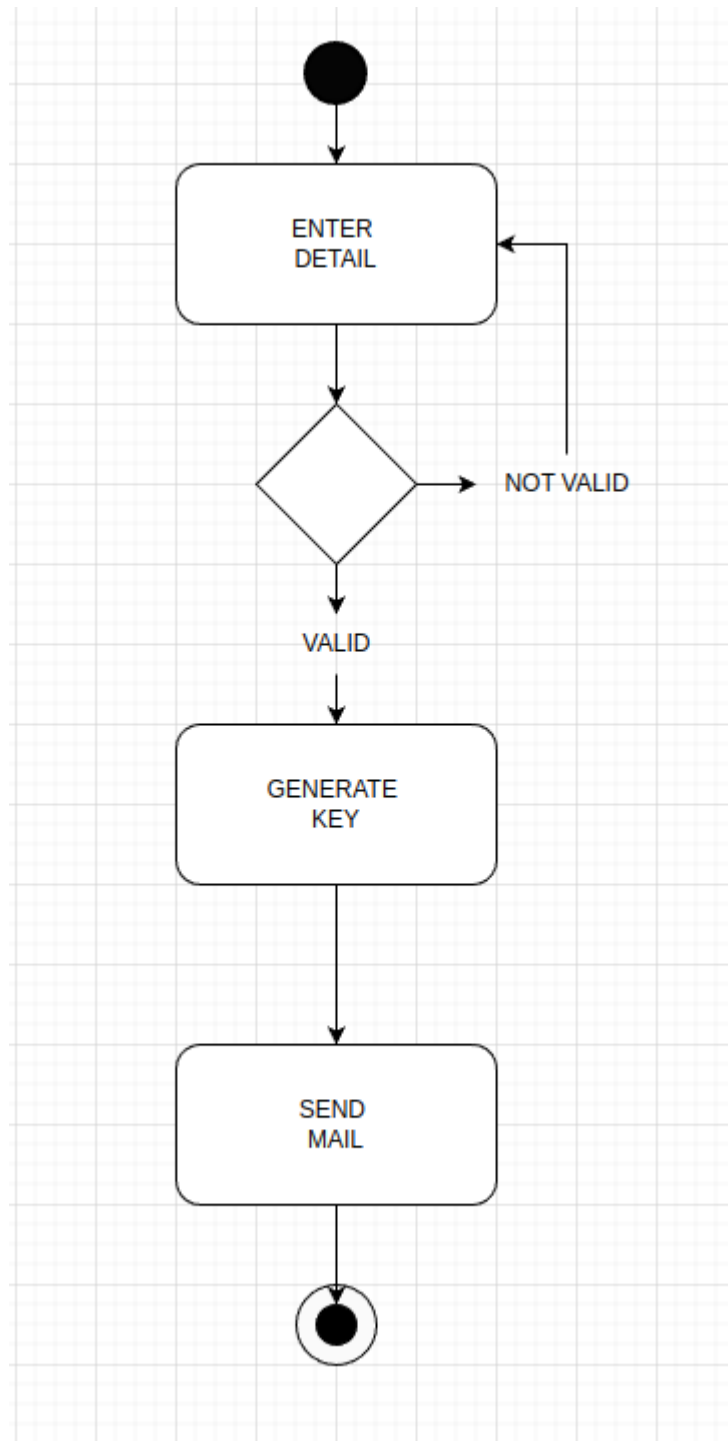
#### 3.2.1 User Login Activity



### 3.2.2 User Send Mail Activity Diagram



### **3.2.3 User Receive Mail Activity Diagram**



## 4. References

- Object Oriented Modeling and Design with UML-Michael Blaha, James Rumbaugh.
- Software Engineering, Seventh Edition, Ian Sommerville.
- IBM Red Books.

- IBM TGMC Sample Synopsis.
- IBM – [www.ibm.in/developerworks](http://www.ibm.in/developerworks).
- Java - [www.sun.com](http://www.sun.com)
- Wikipedia - [www.wikipedia.com](http://www.wikipedia.com)
- Database Management Systems - Navathe.
- Complete Reference - J2EE - Keogh.