



**COMSATS University Islamabad,
Park Road, Chak Shahzad, Islamabad Pakistan**

Department of Computer Science

Assignment-03

CLO-3

Software Design Specification

For

AI CHATBOT XOTRON

Submitted By:

ARSALAN AMIN SP21-BCT-005

HALEEMA SAADIA SP211-BCT-007

Supervisor

Tehseen Riaz Abbasi

Submission Date: (02-12-2022)

Original Version 1.0

Bachelor of Science in Cyber Security (2021-2025)

Table of Contents

1. Introduction	1
------------------------	----------

Contents

1. Introduction	1
2. Design methodology and Software Process Model	1
2.1 Software Process Methodology	1
2.2 Design Methodology	1
3. System overview	1
3.1 Architectural Design	2
4. Design Models	2
4.1 Activity Diagram:	2
4.2 Class Diagram	11
4.3 Sequence Diagram:	11
5. Data design	20
5.1 Data dictionary	20
6. Algorithm & Implementation	20
7. Software requirements traceability matrix	20
8. Human Interface Design	20
8.1 Screen images	20
8.2 Screen objects and actions	20
9. Conclusion	20
10. References	20
11. Appendix I	20
12. Plaragism Report	21

1. Introduction

This chapter provides the Scope of the project, in this chapter we discussed the scope of AI chat bot and its limitations, what we are going to deliver, tool and technologies which we used to build Chatbot.

Additionally, businesses that operate, gather a lot of data, and depend on making quick judgments in real time need to mix AI and intelligent bots. Therefore, a chatbot system will be quite useful in this setting. The ease with which those who dislike chatbots can be connected to actual personnel is one of their best features. The flexibility to switch between assisted and unassisted selling and the choice to engage a human sales assistant when necessary is one of a sales chatbot's most crucial capabilities. The goal of this chatbot system is to lower the work burden on the humans and provide a useful way to the consumers in their daily based task. Also, a check and balance on their health. Everyone needs a personal assistance, so this AI based chatbot is the best system for assessing a user without the interaction of other humans. User work will be easier and efficient.

2. Design methodology and Software Process Model

The Sentiment Analysis module will enable the bot to understand the mood of the user and provide better solutions. Bot will enable dynamic and agile conversation. Will keep the record of old chats to get a deeper understanding of the user's needs.

Health care module will take care of the user's daily bases health by generating a form regarding the daily health issues.it will recommend the exercises or medication according to the problem

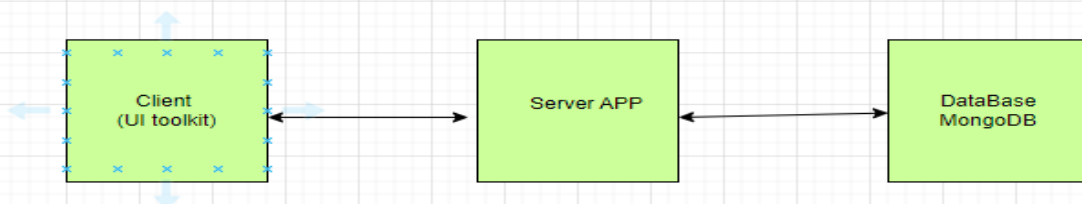
Chatbot analytics will check for the busiest period of the user .It will check when the user will be more active.

3. System overview

Below is the system architecture of our project:

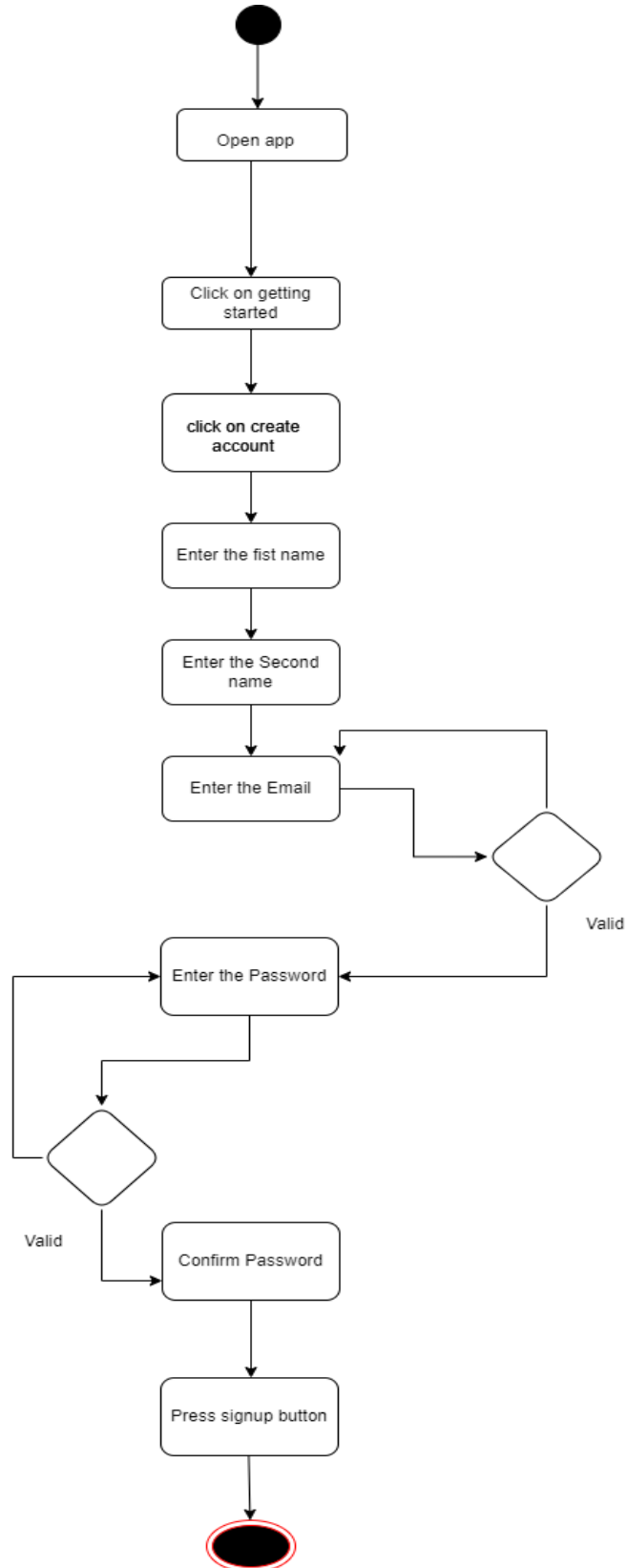
Architectural Design

The three-tier architecture will be used to construct the AI Chatbot System. The system's needs are met by this architecture. For the user interface, we need a database, a logic tier, and a client tier. The system will employ 3-tier architecture because it gives us these levels.

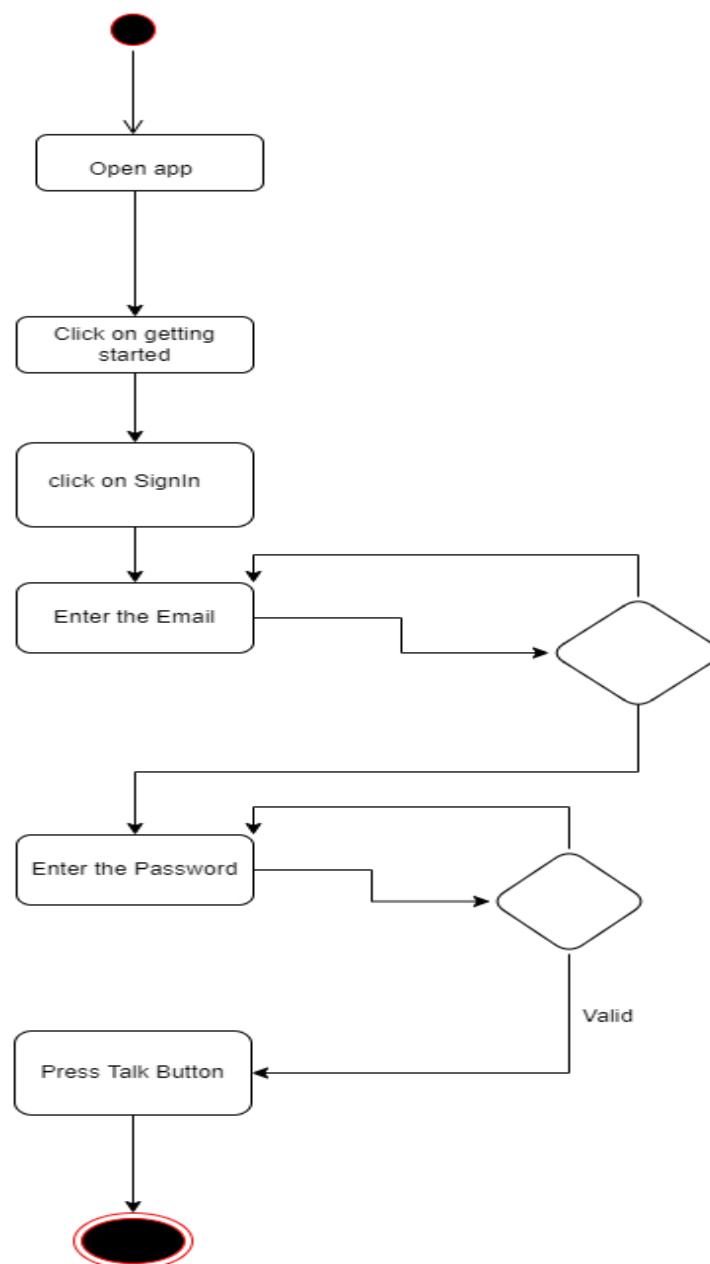


4. Design Models

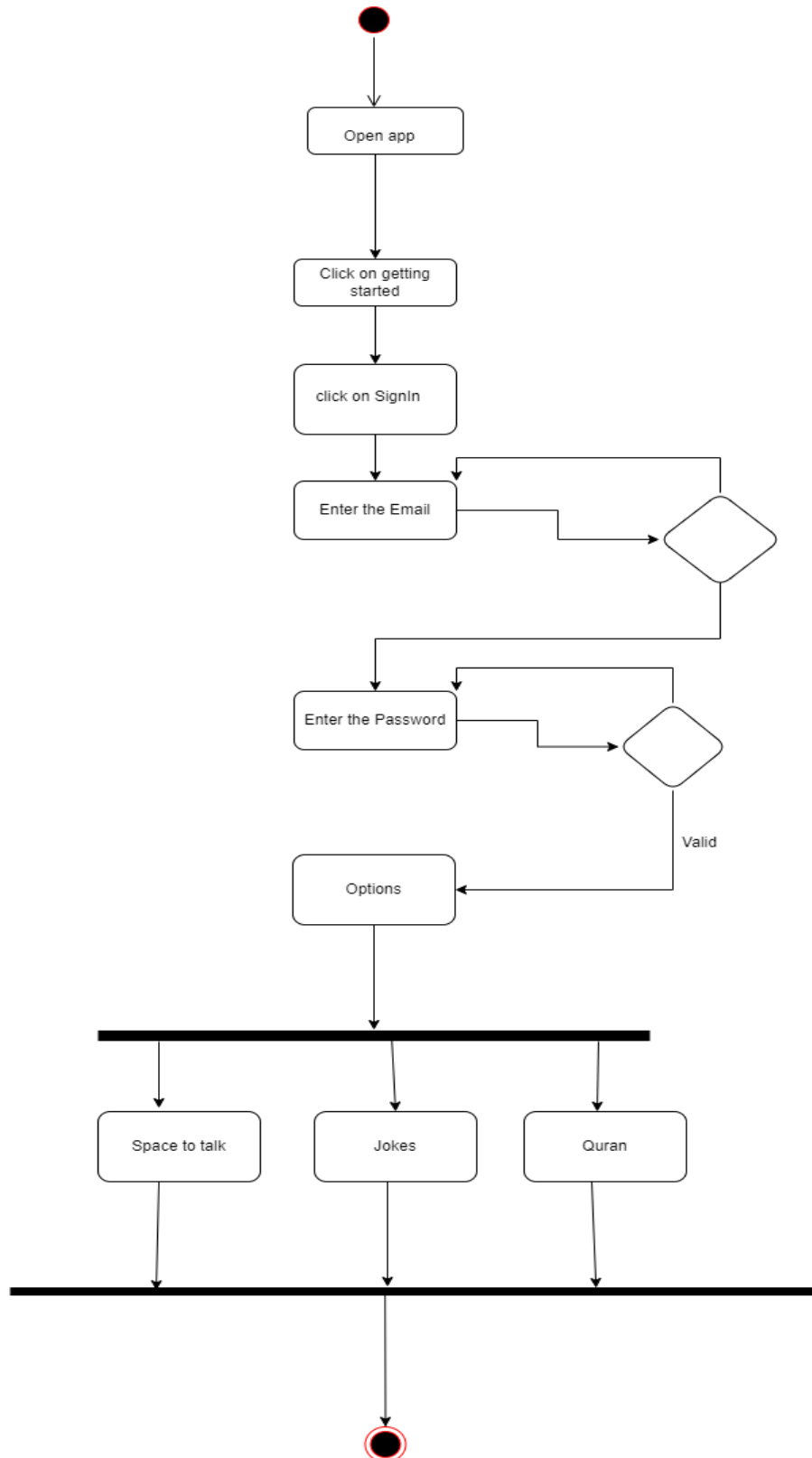
Activity Diagram:



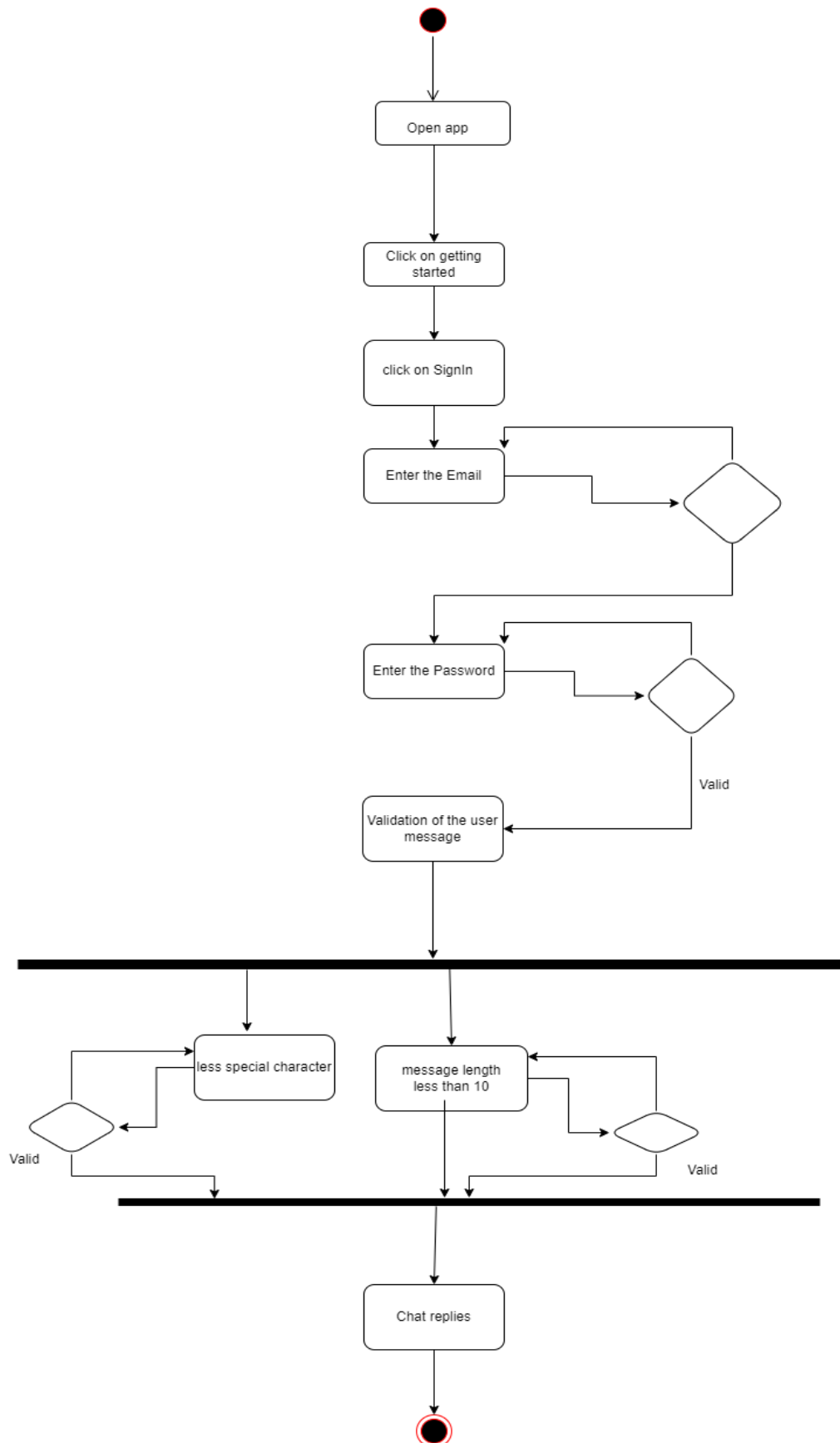
Activity Diagram:



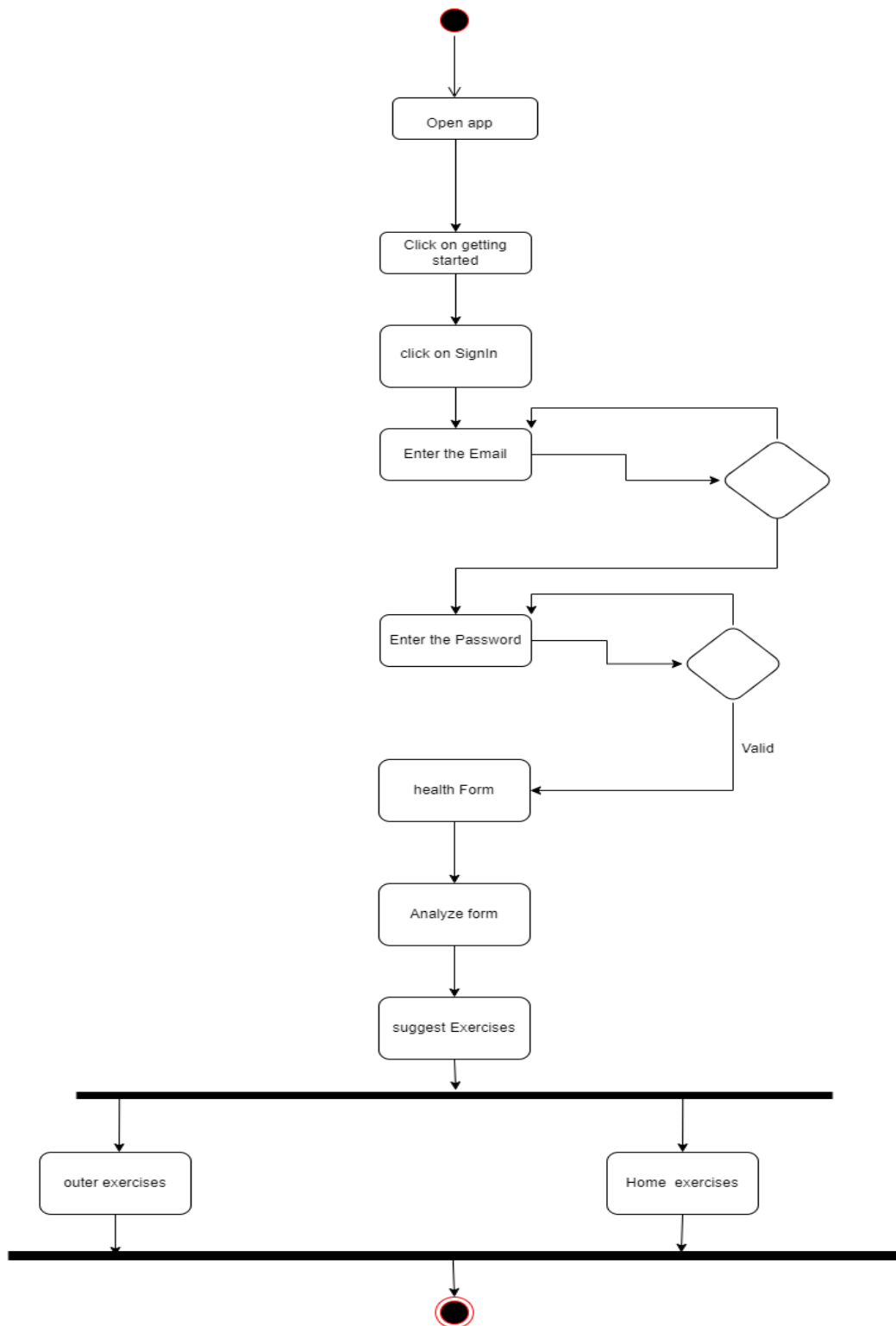
Activity Diagram:



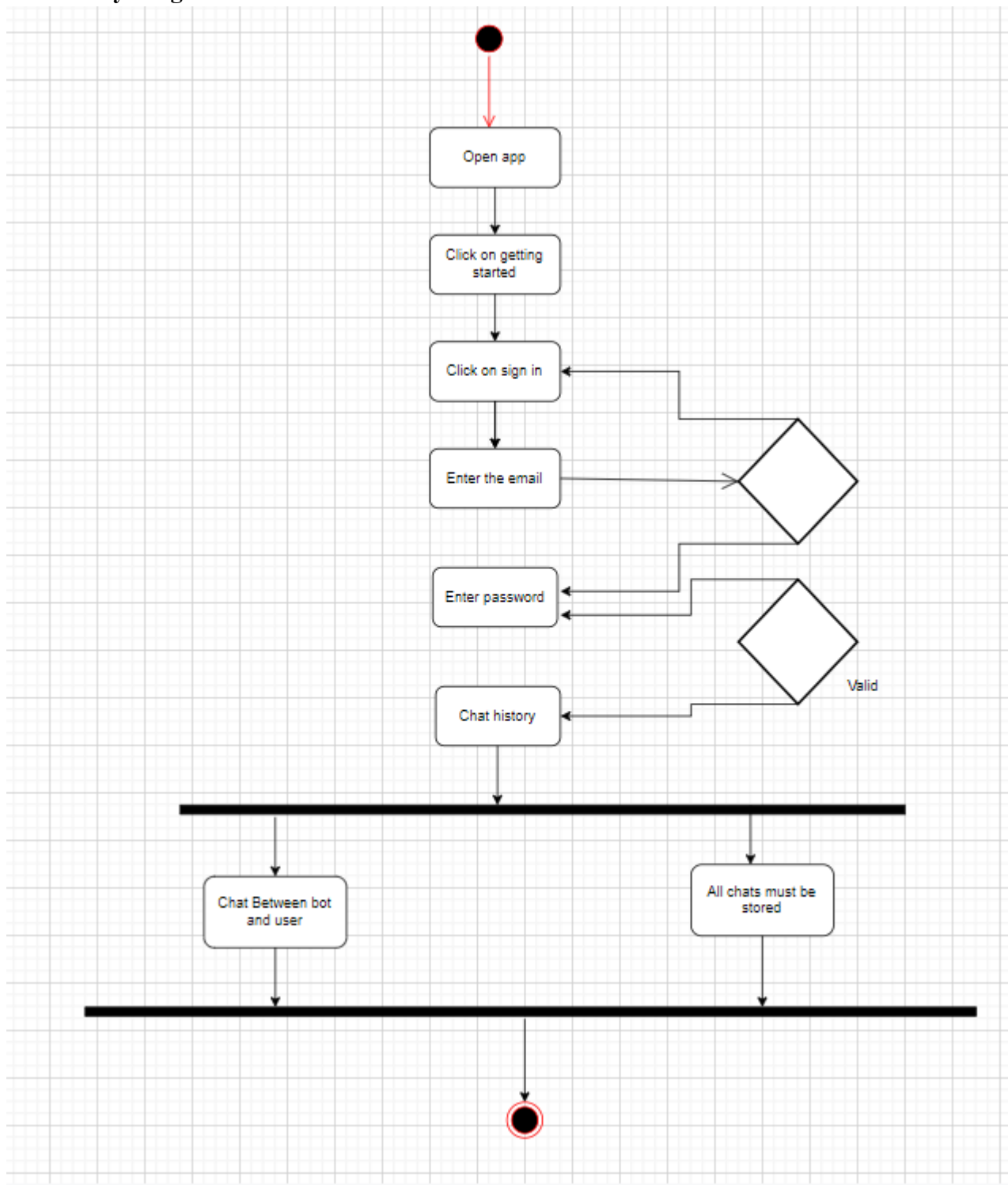
Activity Diagram:



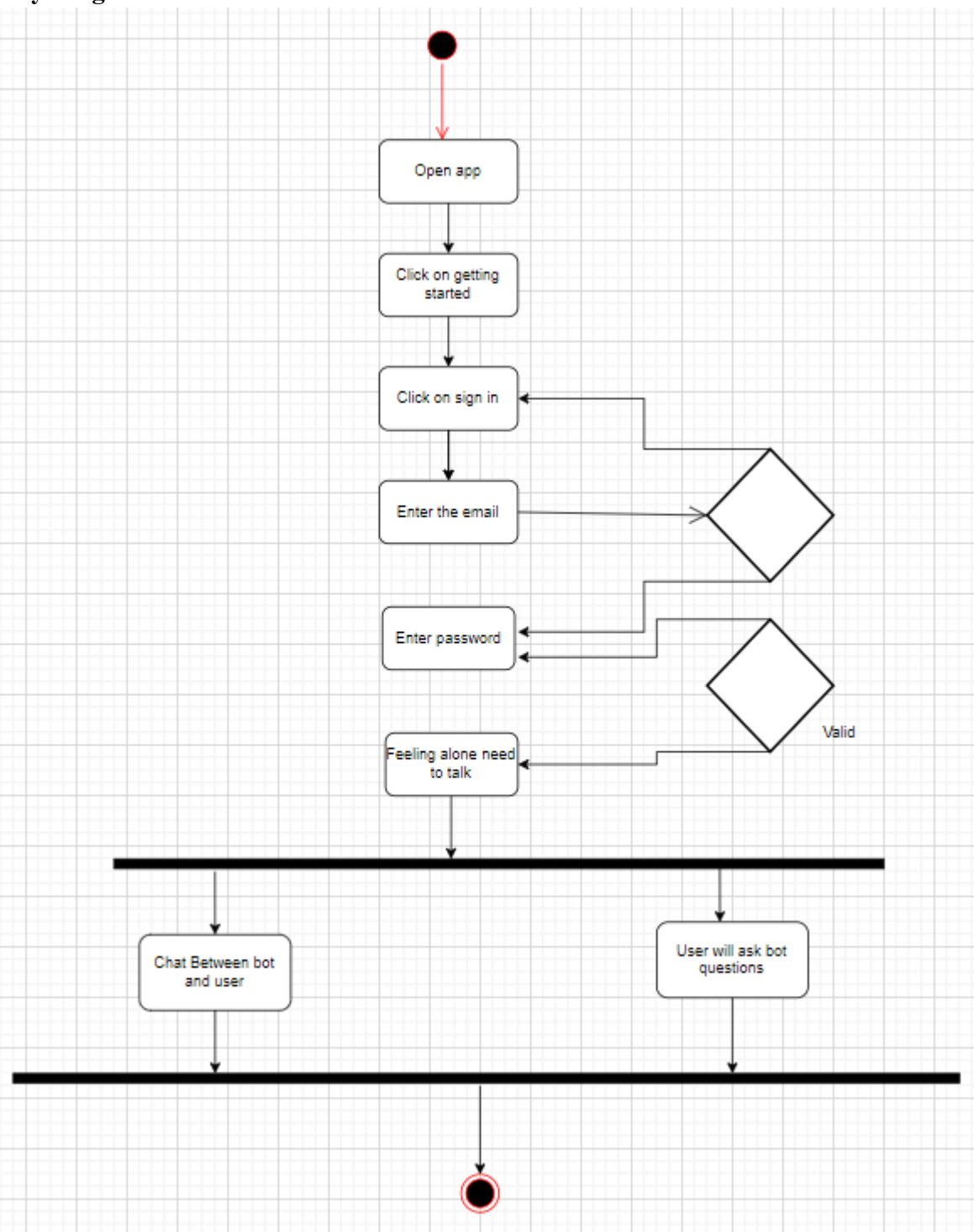
Activity Diagram:



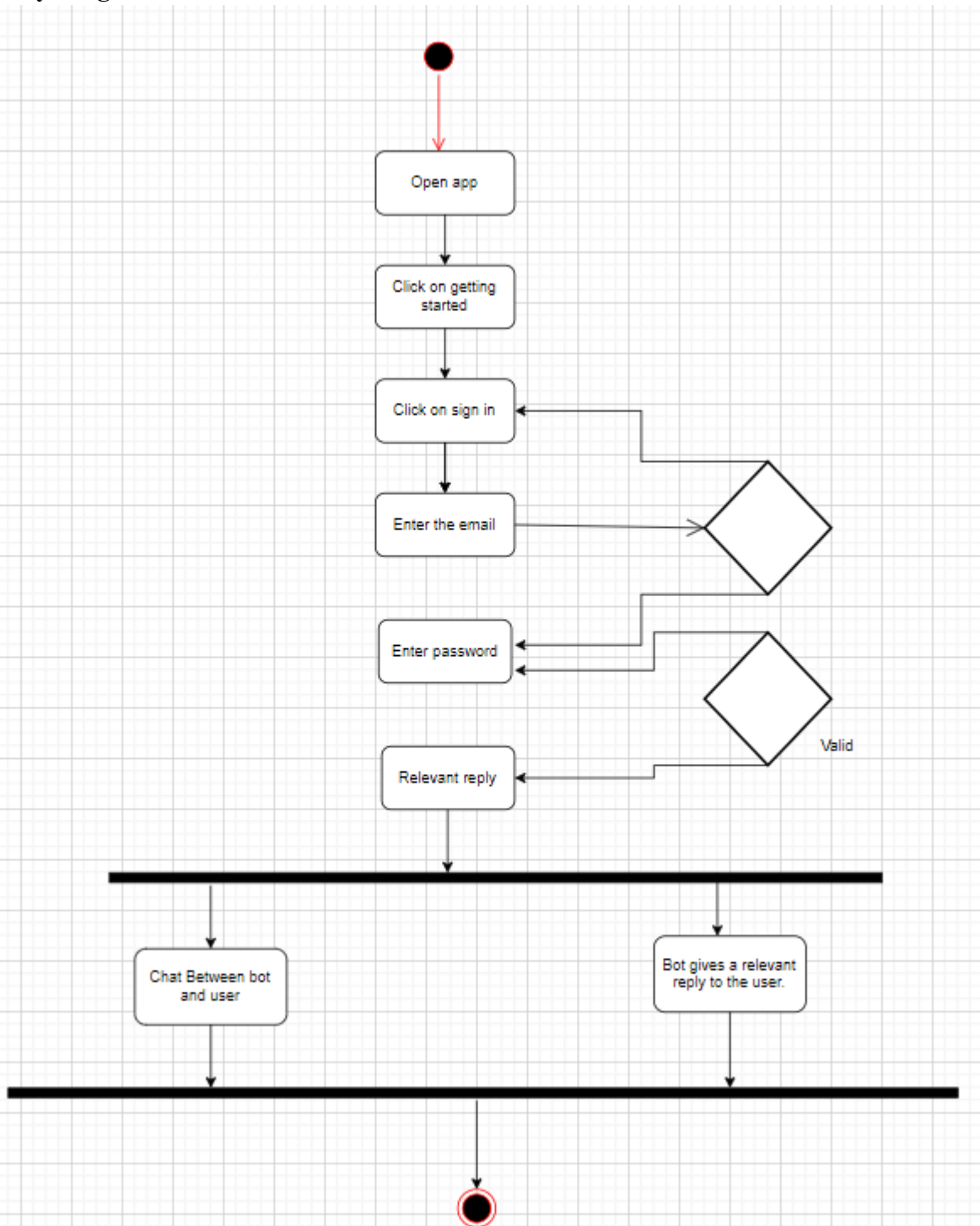
4.6 Activity Diagram:



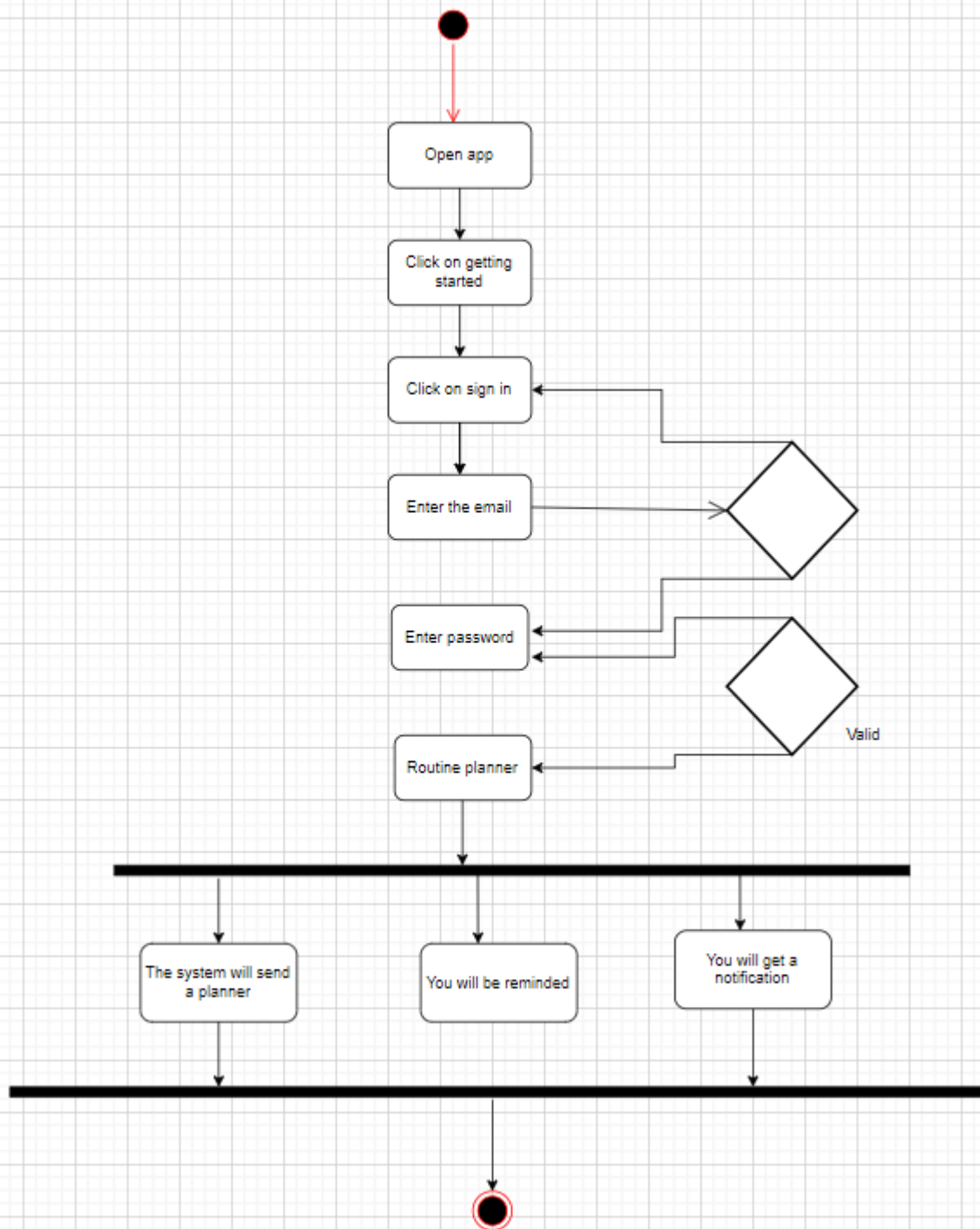
4.7 Activity Diagram:



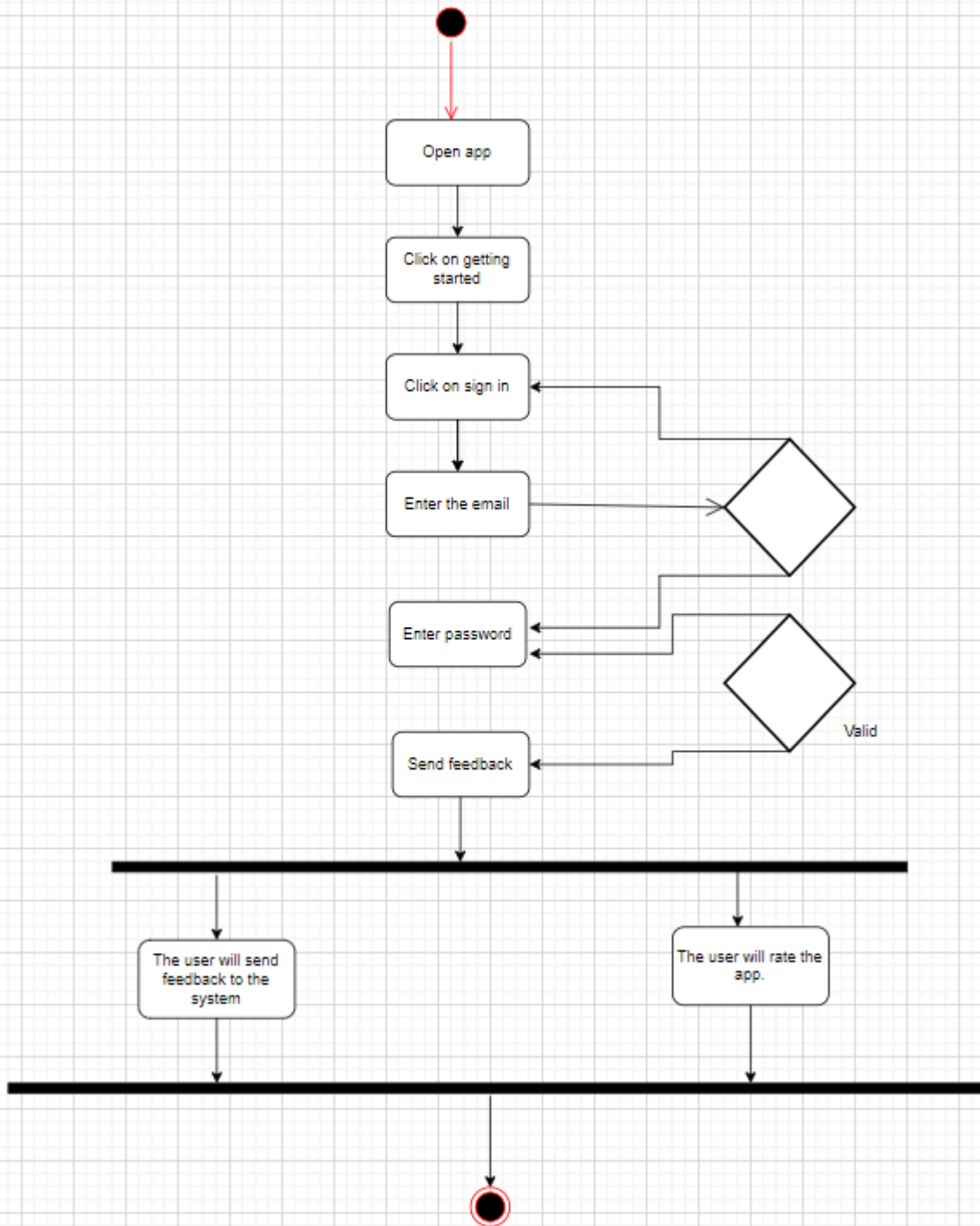
4.8 Activity Diagram:



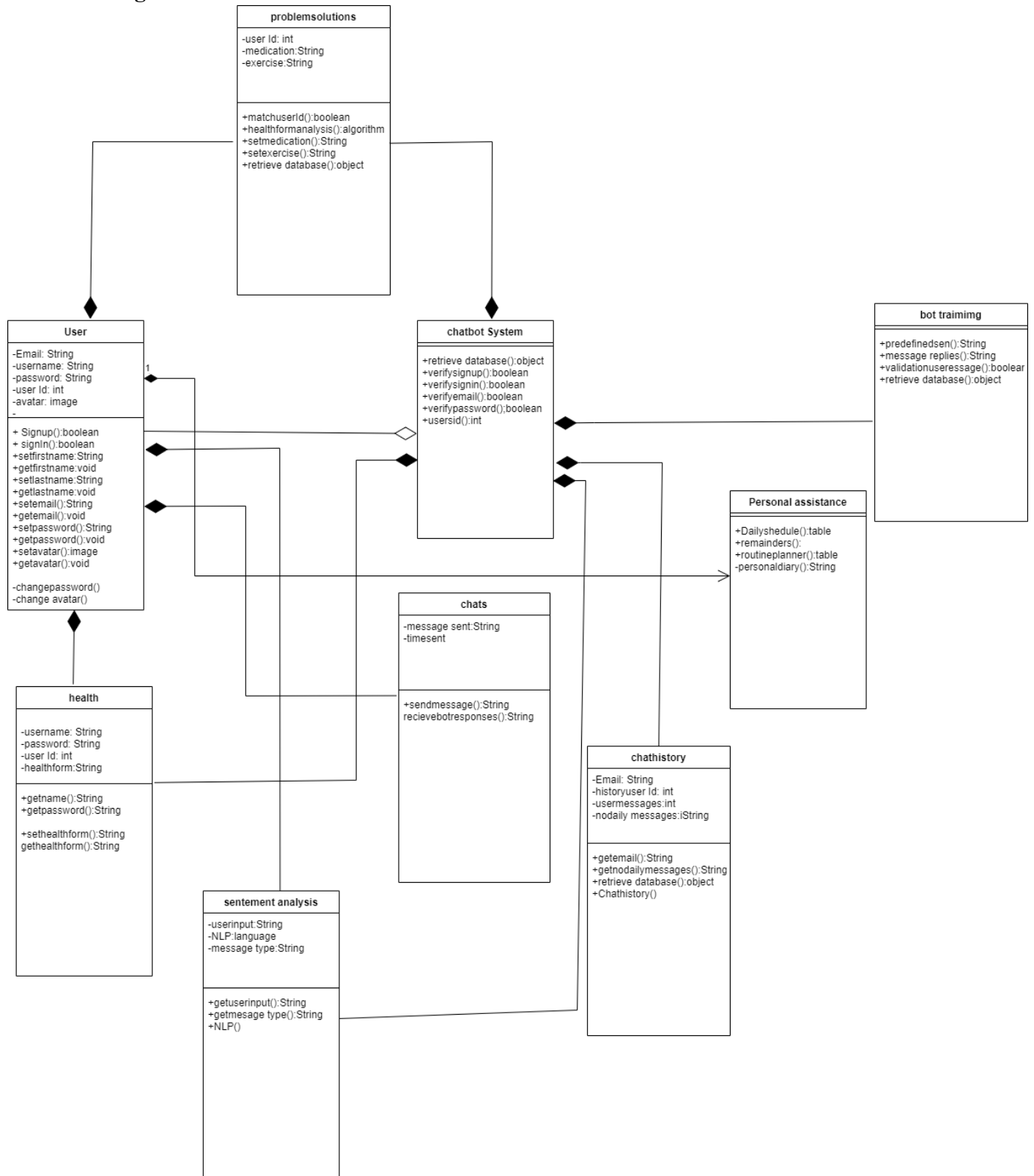
4.9 Activity Diagram:



4.10 Activity Diagram:

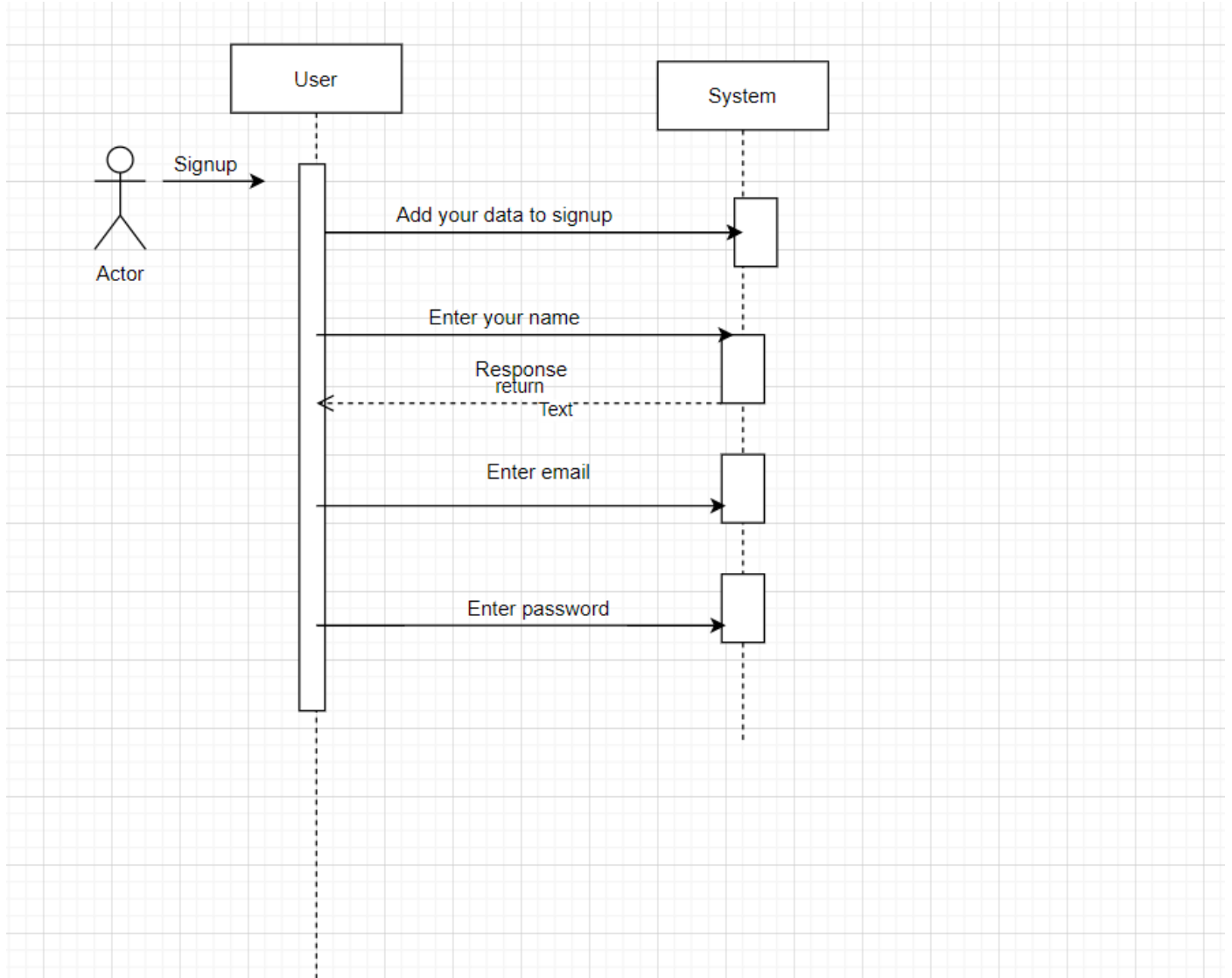


4.6 Class Diagram:

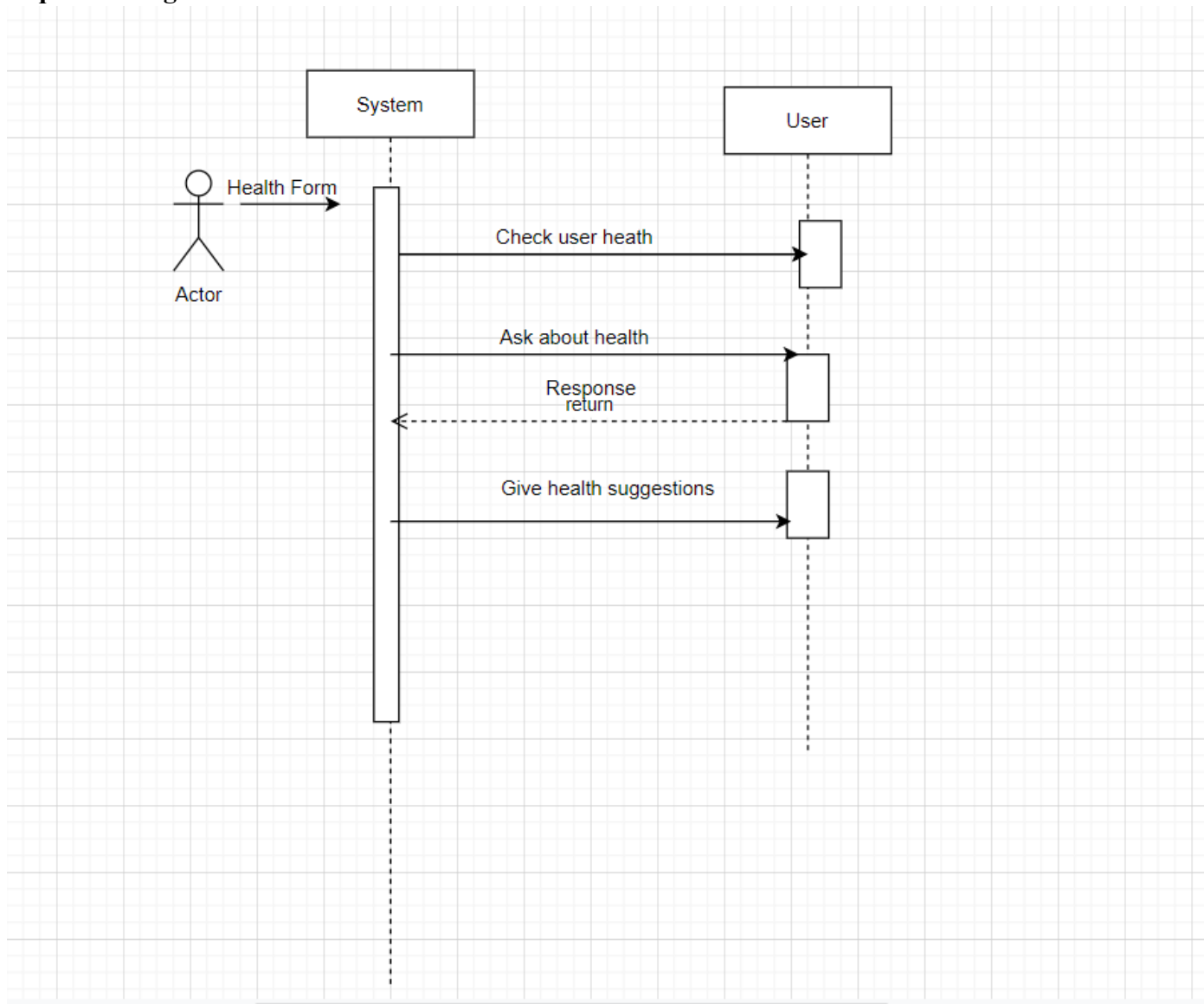


4.7 Sequence Diagrams.

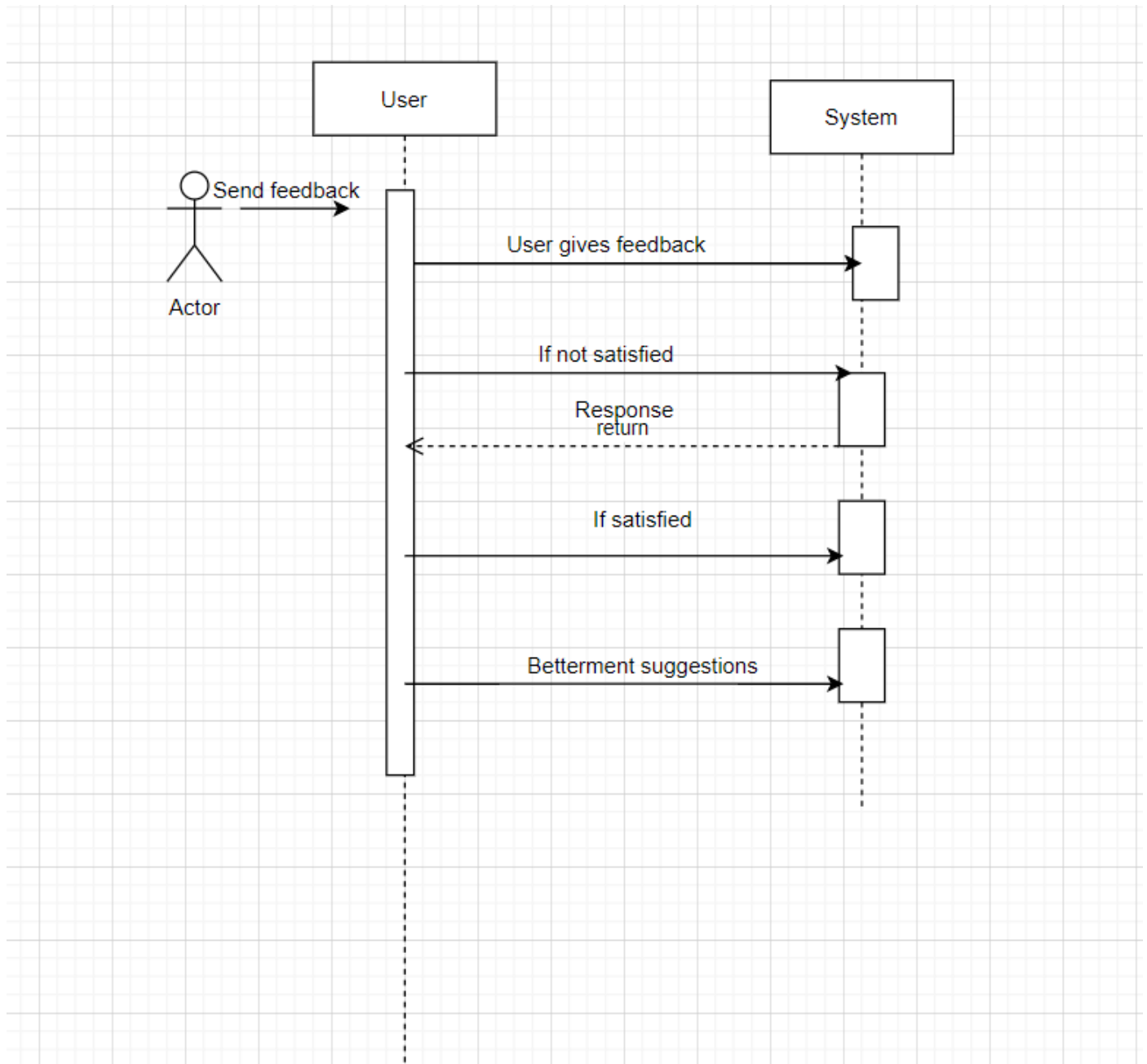
4.7.1 Sequence Diagram 1:



4.7.2:
Sequence Diagram 2:

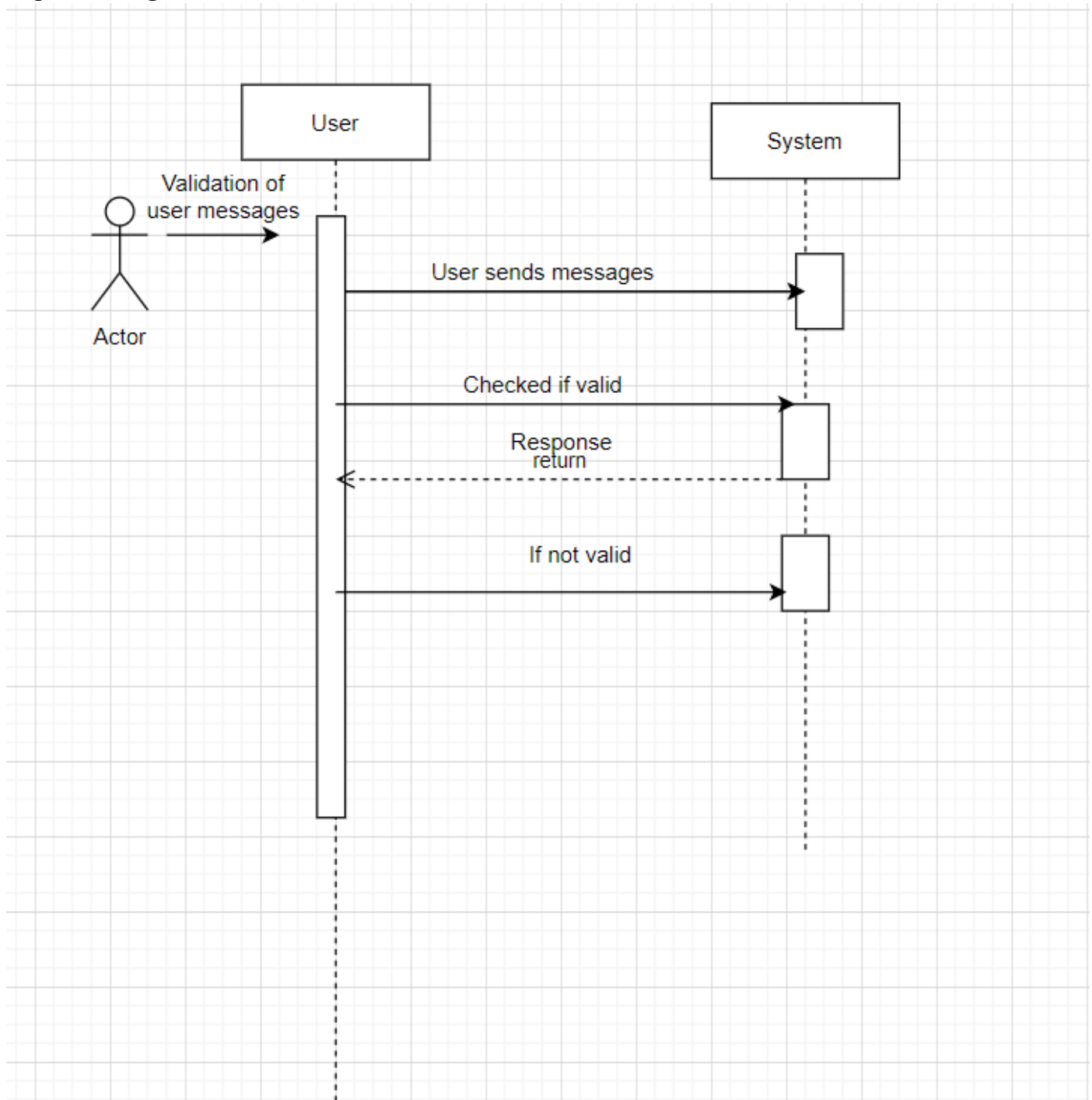


4.7.3:
Sequence Diagram 3:

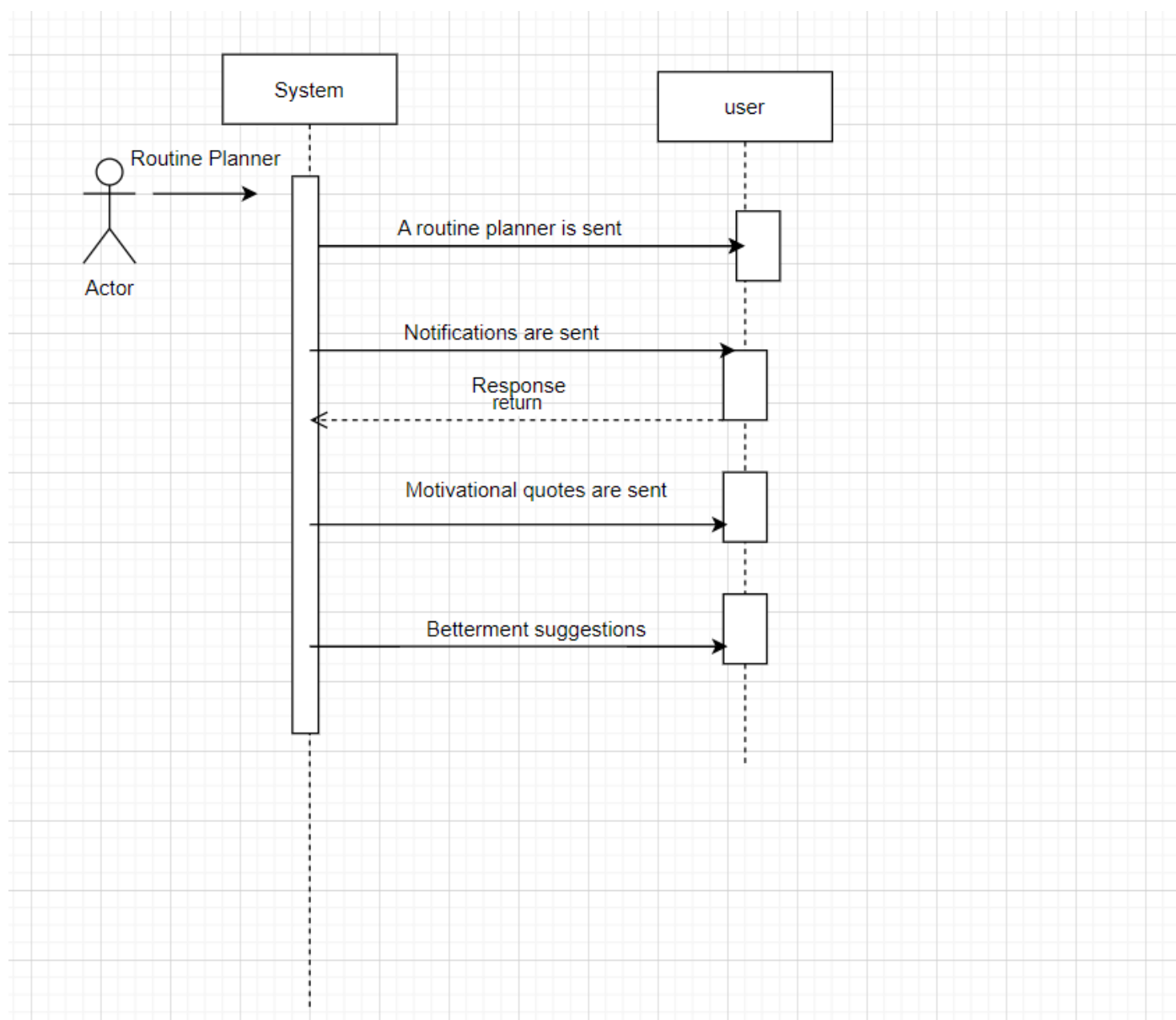


4.7.4:

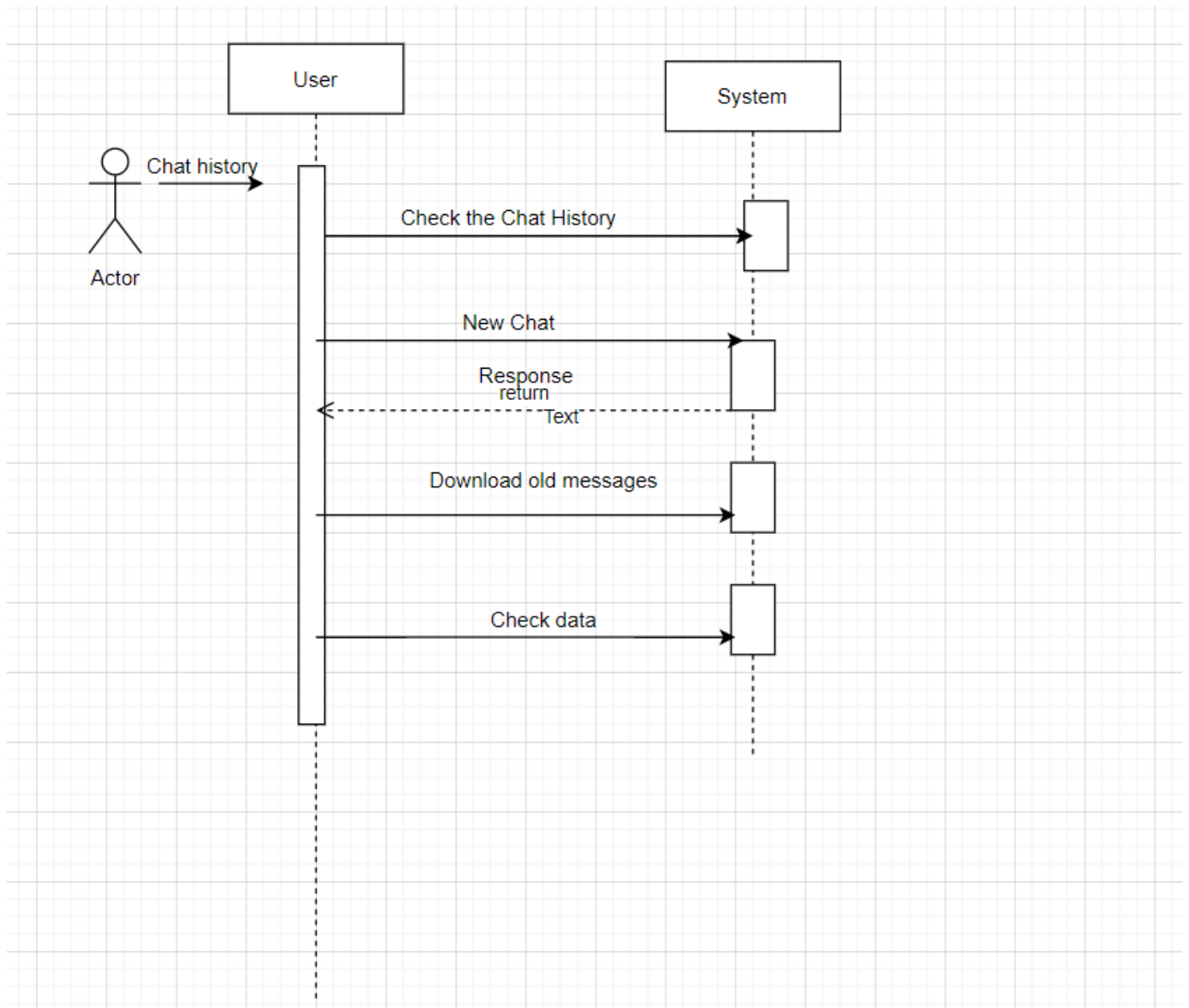
Sequence Diagram 4:



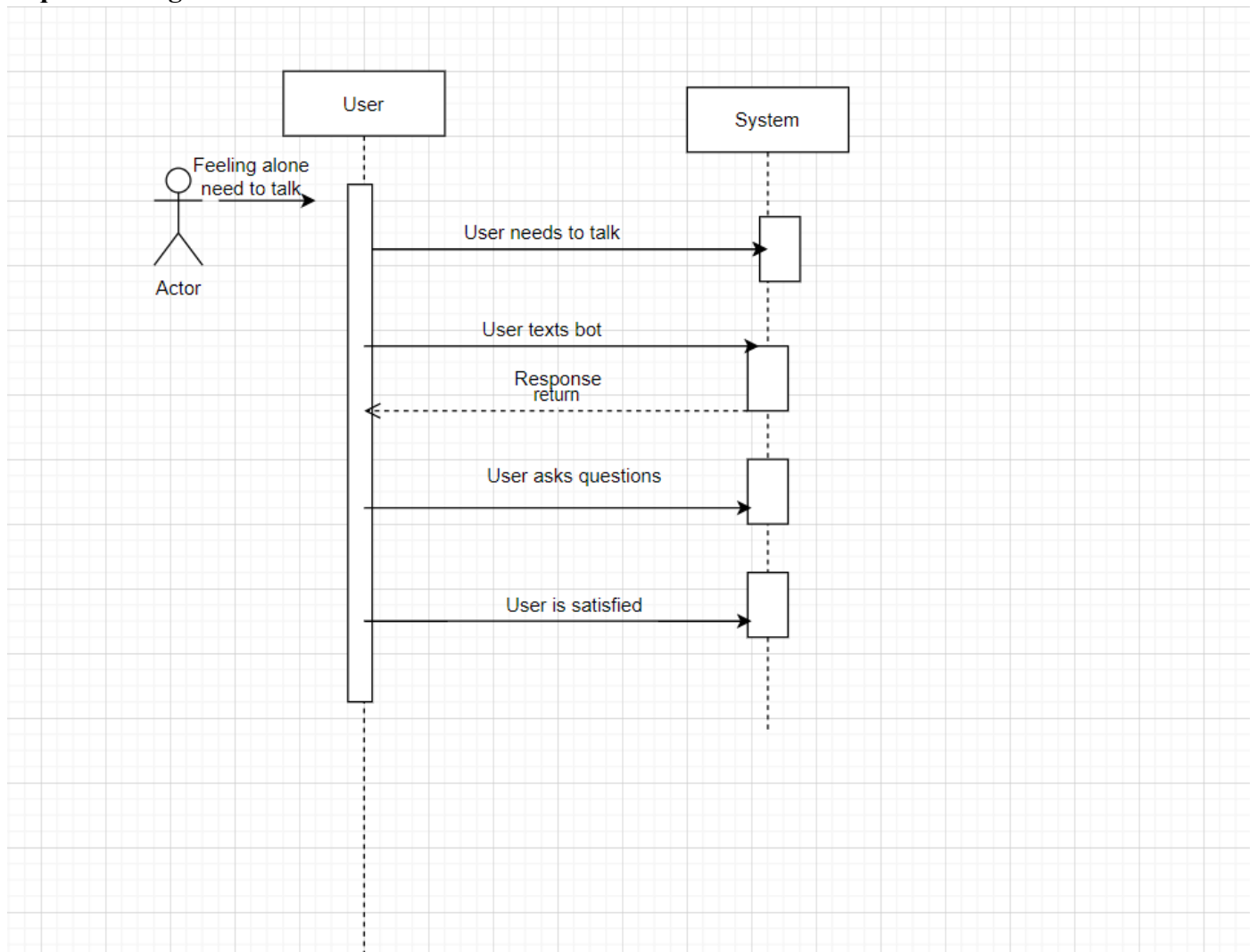
4.7.5:
Sequence Diagram 5:



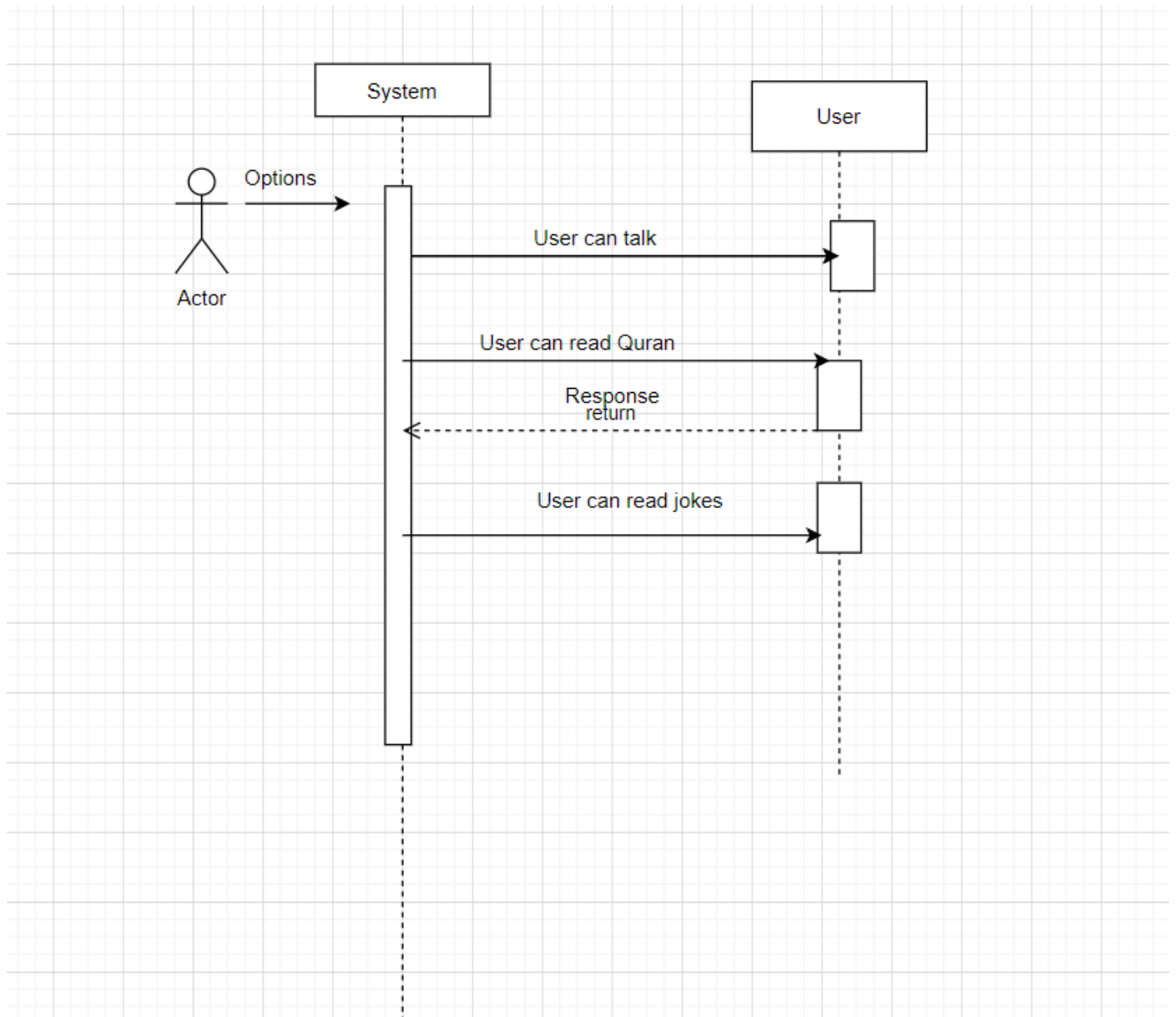
4.7.6:
Sequence Diagram 6:



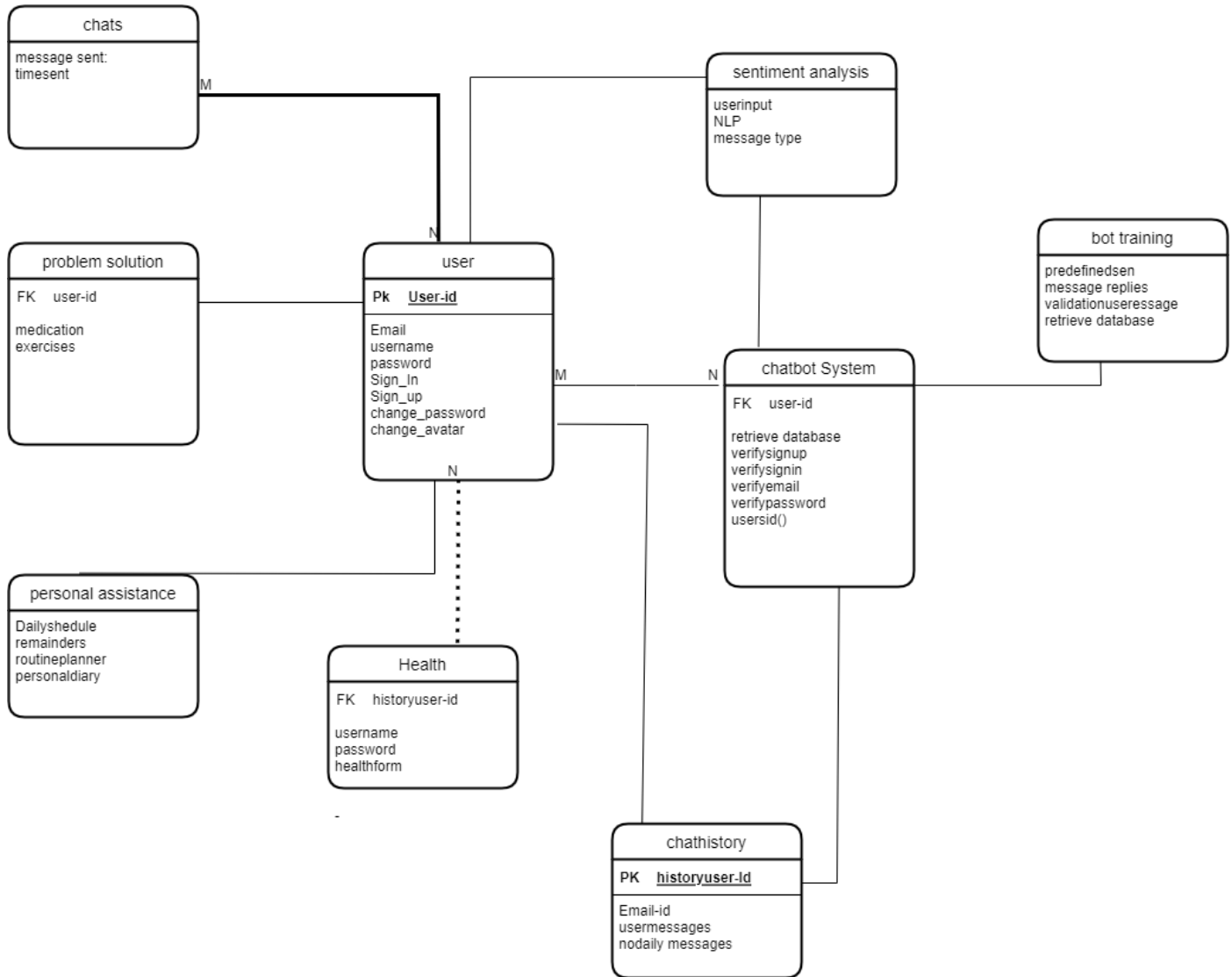
4.7.7:
Sequence Diagram 7:



4.8.8:



5. Data design



Following is the JSON schema of RPC [Renter-Proprietor Coordination] System:

```

{
  "title": "Payment",
  "type": "object",

```

```

    "properties": {
      "Payment_ID": {
        "type": "String",
        "description": "It will contain the ID of payment"
      },
      "Payment_Method": {
        "type": "String",
        "description": "It will contain the method of online payment".
      },
      "Payment_Account_Number": {
        "type": "int",
        "description": "It will contain account number".
      },
      "CVV": {
        "type": "int",

        "description": "It will contain the CVV code of account".

      },
      "Country": {

        "type": "String",

        "description": "It will contain the country of account holder".

      },
      "Account_Expiry": {
        "type": "Date",
        "description": "It will contain the date of expiry of bank card".
      },
      "Billing_Address": {
        "type": "String",
        "description": "It will contain the address of account holder".
      },
      "Date": {
        "type": "Date",
        "description": "It will contain the date of payment".
      },
      "Payment_Amount": {
        "type": "float",
        "description": "It will contain the amount of payment".
      },
    }
  }

  {
    "title": "Complaint",
    "type": "object",
    "properties": {

```



```

"Complaint_ID": {
  "type": "String",
  "description": "It will contain the ID of complaint"
},
"Type": {
  "type": "String",
  "description": "It will contain the type of complaint".
},
"Description": {
  "type": "String",
  "description": "It will contain description of complaint".
},
>Status": {
  "type": "boolean",
  "description": "It will contain the progress status of complaint".
},
>Date": {
  "type": "Date",
  "description": "It will contain the date of complaint".
},
>Time": {
  "type": "Time",
  "description": "It will contain the time of complaint".
},
}

```

```

{

```

```

  "title": "User",

```

```

  "type": "object",

```

```

  "properties": {

```

```

    "ID": {

```

```

      "type": "int",

```

```

      "description": "It will contain the ID of user"

```

```

    },

```

```
“Profile_Image”: {  
  
    “type”: “image”,  
  
    “description”: “It will contain the profile picture of user”.  
  
    },
```

```
“CNIC”: {  
  
    “type”: “String”,  
  
    “description”: “It will contain the CNIC of user”.  
  
    },
```

```
“First_Name”: {  
  
    “type”: “String”,  
  
    “description”: “It will contain the first name of user”.  
  
    },
```

```
“Last_Name”: {  
  
    “type”: “String”,  
  
    “description”: “It will contain the last name of user”.  
  
    },
```

```
“Contact”: {  
  
    “type”: “int”,
```

“description”: “It will contain the contact number of user”.

},

“Email”: {

“type”: “String”,

“description”: “It will contain the email of user”.

},

“Password”: {

“type”: “String”,

“description”: “It will contain the password of user”.

},

“isPremiumAccount”: {

“type”: “boolean”,

“description”: “It will contain whether account is premium or not.

},

“isRenter”: {

“type”: “boolean”,

“description”: “It will contain whether user is renter.

```
    },  
    "isProprietor": {  
        "type": "boolean",  
        "description": "It will contain whether user is proprietor."  
    },  
}
```

```
{  
    "title": "Feedback",  
    "type": "object",  
    "properties": {
```

```
        "ID": {  
            "type": "String",  
            "description": "It will contain the ID of feedback"  
        },
```

```
        "Description": {  
            "type": "String",  
            "description": "It will contain description of feedback".  
        },
```

```
        "Rating": {
```

“type”: “int”,

“description”: “It will contain the rating from 1-5”.

},

“Date”: {

“type”: “Date”,

“description”: “It will contain the date of feedback”.

},

“Time”: {

“type”: “Time”,

“description”: “It will contain the time of feedback”.

},

}

{

“title”: “Notification”,

“type”: “object”,

“properties”: {

“ID”: {

“type”: “String”,

“description”: “It will contain the ID of notification”

},

“Type”: {

“type”: “String”,

“description”: “It will contain the type of notification”.

},

“Message”: {

“type”: “String”,

“description”: “It will contain message of notification”.

},

“Title”: {

“type”: “String”,

“description”: “It will contain the Title of notification”.

},

}

{

```
“title”: “Notification_Settings”,

“type”: “object”,

“properties”: {

    "Chat_Notification_State": {

        “type”: “boolean”,

        “description”: “It will contain whether chat notification is on or off”

    },

    "Rent_Notification_State": {

        “type”: “boolean”,

        “description”: “It will contain whether rent notification is on or off”

    },

    “Ringtone”: {

        “type”: “File”,

        “description”: “It will contain the ringtone”.

    },

    “Intensity”: {

        “type”: “int”,

        “description”: “It will contain volume of notification”.

    },
```

“Vibration_Frequency”: {

“type”: “float”,

“description”: “It will contain the frequency of vibration”.

},

}

{

“title”: “Rent”,

“type”: “object”,

“properties”: {

“ID”: {

“type”: “String”,

“description”: “It will contain the ID of rent”

},

“Due_Amount”: {

“type”: “float”,

“description”: “It will contain the amount of due rent”.

},

“Due_Date”: {

“type”: “Date”,

“description”: “It will contain Date of due rent”.

},

“Rent_Status”: {

“type”: “boolean”,

“description”: “It will contain the progress status of rent payment”.

},

}

{

“title”: “ChatBot”,

“type”: “object”,

“properties”: {

“Message_Sent”: {

“type”: “String”,

“description”: “It will contain the message sent to bot”

},

```
“Time”: {  
  
    “type”: “Time”,  
  
    “description”: “It will contain the time of message sent to bot”.  
  
    },
```

```
}
```

```
{
```

```
    “title”: “Lease”,  
    “type”: “object”,  
    “properties”: {  
  
        “ID”: {  
  
            “type”: “String”,  
  
            “description”: “It will contain the ID of lease”  
  
            },
```

```
        “Description”: {  
  
            “type”: “String”,  
  
            “description”: “It will contain description of lease”.  
  
            },
```

```
        “Renter_Signature”: {
```

“type”: “image”,

“description”: “It will contain the verification signature of renter”.

},

“Proprietor_Signature”: {

“type”: “image”,

“description”: “It will contain the verification signature of Proprietor”.

},

“Creation_Date”: {

“type”: “Date”,

“description”: “It will contain the date of lease finalization”.

},

}

{

“title”: “Property”,

“type”: “object”,

“properties”: {

"ID": {

 "type": "String",

 "description": "It will contain the ID of property"

},

"Property_Name": {

 "type": "String",

 "description": "It will contain name of property".

},

"Property_Address": {

 "type": "String",

 "description": "It will contain address of property".

},

"Property_Rooms": {

 "type": "int",

 "description": "It will contain number of rooms in the property".

},

"Property_Allocation_Area": {

 "type": "float",

 "description": "It will contain area of the property".

```

    },
    "Property_Floors": {

        "type": "int",

        "description": "It will contain number of floors in the property".

    },
    "Number_of_Renters": {

        "type": "int",
        "description": "It will contain number of renters allowed to rentthe property".
    },
}

{
    "title": "Chat",
    "type": "object",
    "properties": {
        "Sender_ID": {
            "type": "String",
            "description": "It will contain the ID of message sender"
        },
        "Receiver_ID": {
            "type": "String",
            "description": "It will contain the ID of message receiver"
        },
        "Sender_Message": {
            "type": "String",
            "description": "It will contain the sender message"
        },
        "Receiver_Message": {
            "type": "String",
            "description": "It will contain the receiver message"
        },
        "Date_Sent": {
            "type": "Date",
            "description": "It will contain the date of message sent".
        },
        "Time_Sent": {
            "type": "Time",
            "description": "It will contain the time of message sent".
        },
    },
}

```

}

Data dictionary

Collections	Properties	Data Type	Description
email	Address	String	will contain the email address of the user
chat history	history	String	will have the chat history of the user
daily routine	users	Table	will have the daily routine table
ID	User1 id	String	It will contain user1 ID
ID	user2id	String	It will contain users2 ID
ID	user3id	String	It will contain user3 ID
name	First name	String	It will contain first name of the user
name	Last name	String	It will contain last name of the user
Password1	Password1 User1	String	will contain the password of user1
Password2	Password2 User2	String	will contain the password of user2
Password3	Password3 User3	String	will contain the password of user3
Health form	users health	algorithms	will contains an algo or table containing the information about user health
description	feedback Description	String	will contain the feedback description
description	complaint Description	String	will contain the complaint description

6. Algorithm & Implementation

```
INPUT email
IF(verifyEmail NOT EQUAL true)
    DISPLAY "Invalid Email"
WHILE (verifyEmail EQUAL false)
INPUT password
IF(account NOT found)
    DISPLAY "Invalid login credentials! Please try again!"
ELSE
    DISPLAY "Successfully logged in!"
    GENERATE ads
    DISPLAY ads
IF(login)
    DO
        INPUT message
        INPUT receiverID
        WHILE(message NOT null OR valid receiverID)
            SEND message request to receiver end
ELSE
    DISPLAY "Please login"
```

Object Detection:

7. Software requirements traceability matrix

8. Human Interface Design

Screen images

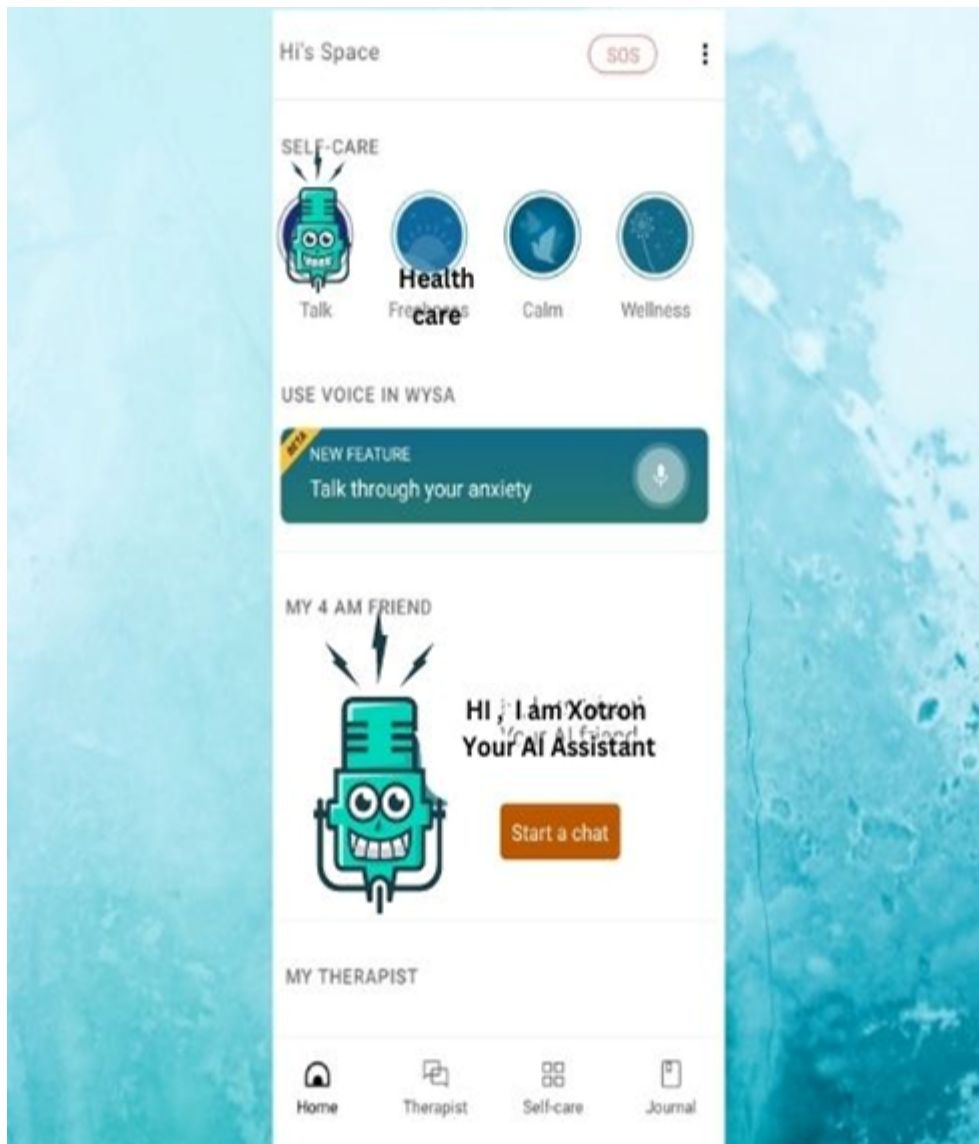
FOR Mobile:

MOCKUP 1:

Front page

MOCKUP 2:

Home Page

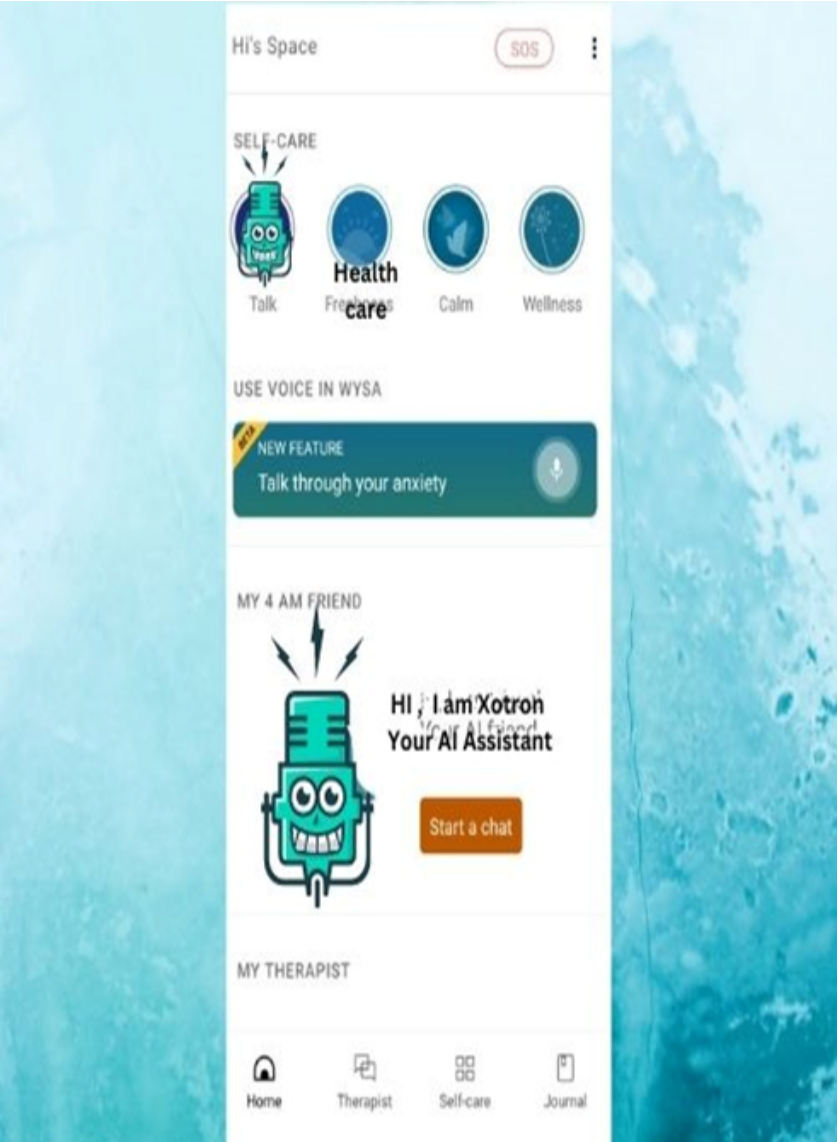


Mockup 3:
Security



Mockup 4:

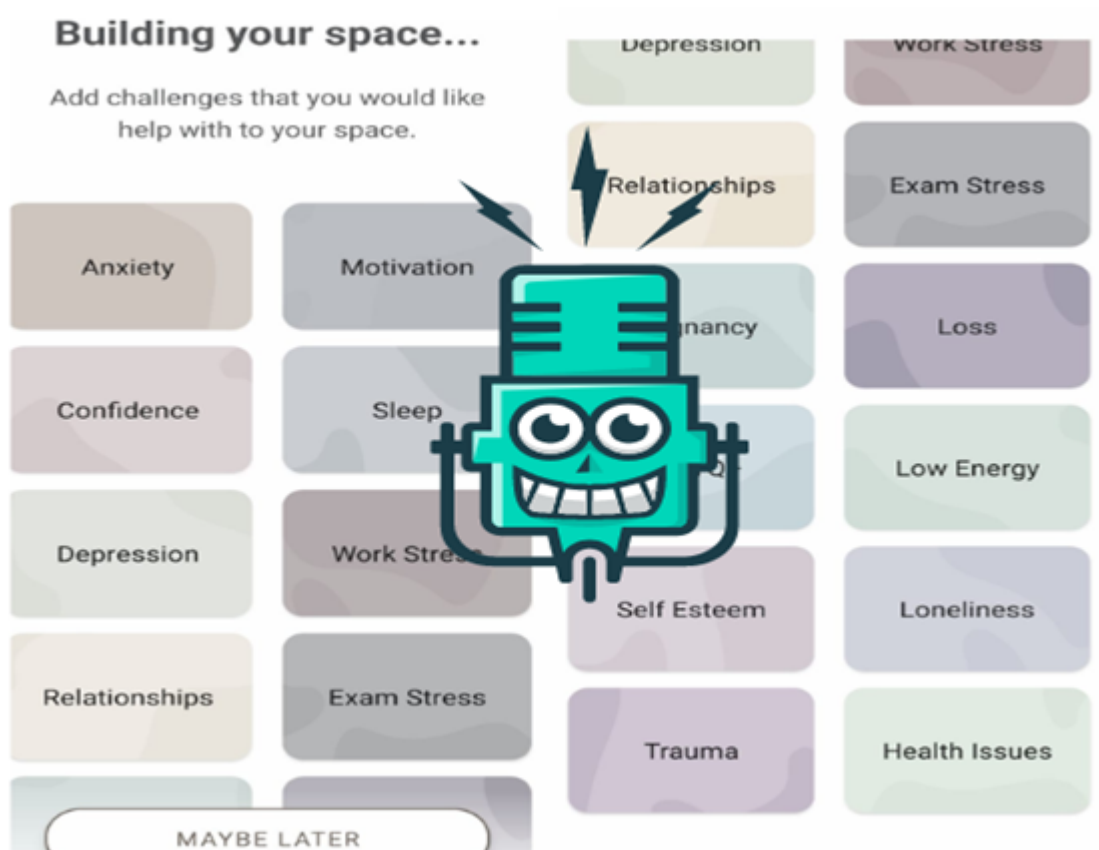
Choose from Options

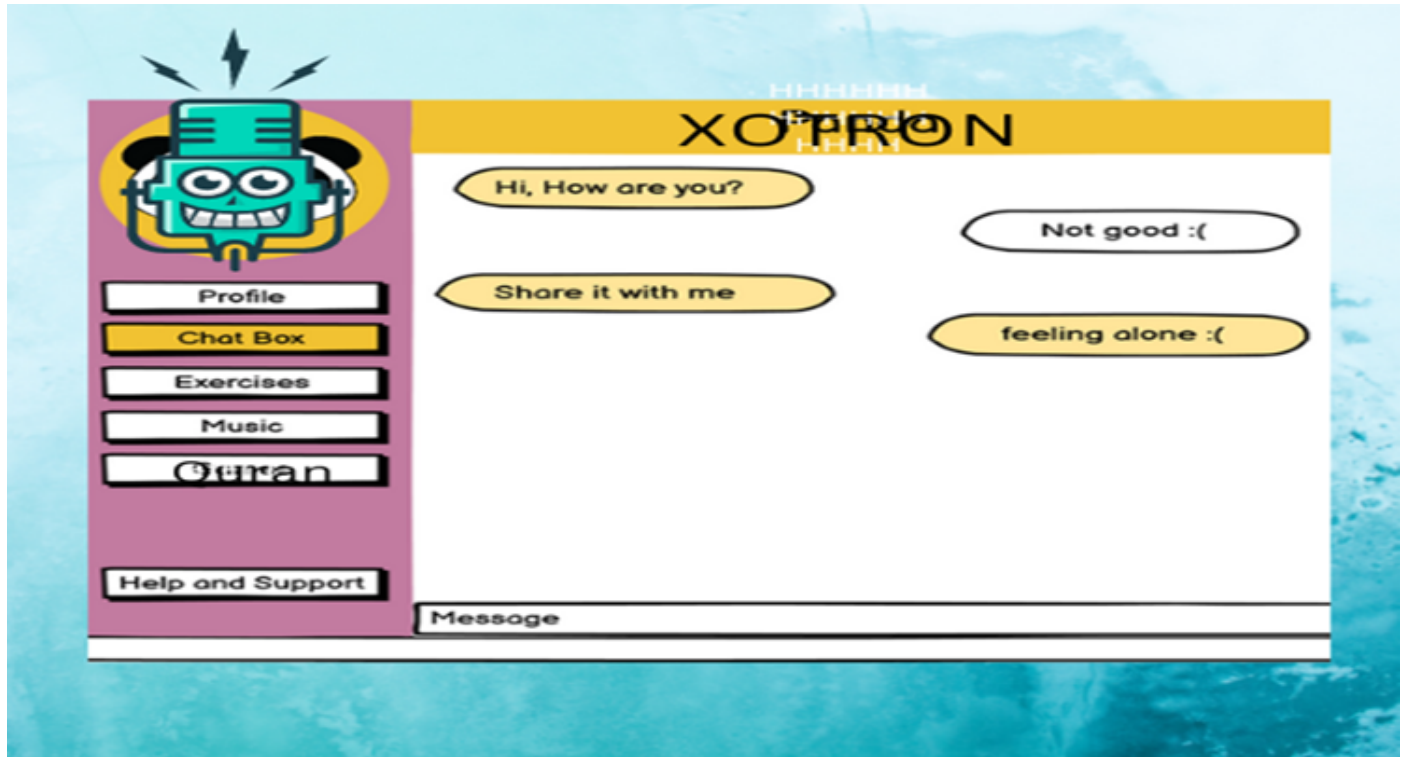


FOR WEB

Mockup 5

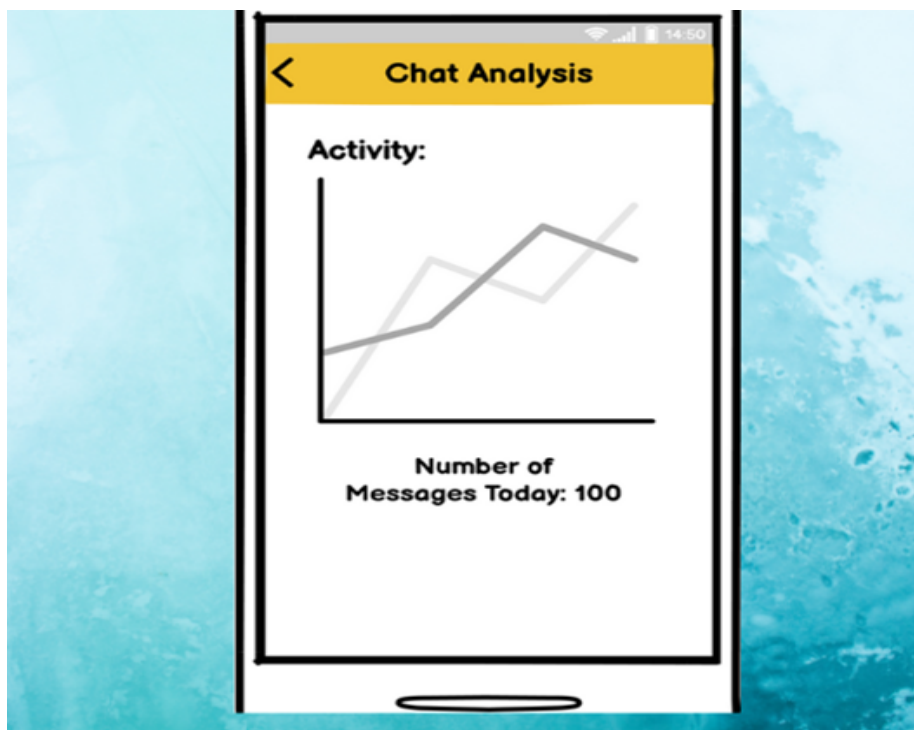
Chat screen for Web





MOCKUP 6

Chat analysis



9. Conclusion

The system's design is fully described in this document. Additionally provided are the abstract implementation details and the logical flow. The user and system requirements have been met. This document will contain the application's design. How it operates and how it is put into practise. Additionally, the approaches necessary for the application to function correctly will be described.

10. References

- · <https://ui8.net/the-madbrains/products/madbrainsui--chatbot--ai-startup-agencyux>
- · <https://www.analyticsvidhya.com/blog/2021/10/complete-guide-to-build-your-ai-chatbot-with-nlp-in-python/>
- · <https://vilmate.com/blog/how-to-develop-a-chatbot/>
- · <https://youtu.be/1XHp8WrZzoc>
- · <https://www.ameyo.com/blog/key-advantages-and-use-cases-of-healthcare-chatbot/>
- · <https://geekflare.com/create-chatbot/>
- · https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_natural_language_processing.htm

11. Appendix I

12. Plaragism Report

Attach the Plaragism report of your project requirement document from library staff of Turnitin tool (<http://turnitin.com>)

