MENTAL HEALTH CHAT BOT

Software Engineering
Project Report

Aryabhatta College
University of Delhi
2023
Department of Computer Science
B.Sc(H) Computer Science, Semester IV

Submitted by:
Ritesh (21059570036)
Pinki Kumari(21059570029)
Riya Saini (21059570039)

Supervisor:

Ms. Neha Kumari

Mr. Raj Kumar Sharma

Acknowledgement

In performing our Project We would like to show our gratitude to our professors Mr. Rajkumar Sharma and Ms. Neha Kumari for their even willingness to give us valuable advice and direction under which we executed this project. Their constant guidance and willingness to share their vast knowledge made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks. We like to extend special thanks to friends, classmates and team members, have made valuable comment suggestions on this proposal which gave us inspiration to improve our assignment, without their support and coordination we would not have been able to complete this project.

Mrs. Neha Kumari

Ritesh
(21059570036)

Mr. Raj Kumar Sharma

Pinki
(21059570029)

Riya
(21059570039)

Certificate

This is to certify that **Ritesh Kumar**, **Riya Saini , Pinki Kumari** successfully carried out the completion of the project entitled **"Mental Health Chat Bot"** under my supervision. The Project has been submitted as per the requirement of Lab based on Software Engineering of B.Sc. (H) Computer Science, IV Semester.

Ritesh
(21059570036)

Pinki
(21059570029)

Riya Saini
(21059570039)

Ms. Neha Kumari

Mr. Raj Kumar Sharma

Contents

TABLE OF FIGURES

Fig: Figure DFD: Data Flow Diagram ER: Entity Relation

Problem Statement/Objective:

According to the World Health Organization (WHO), mental health conditions are increasing worldwide. 14% of all adolescents in the world (almost 1 billion) experienced mental health problems in 2019 due to the Corona virus Disease 2019 (COVID-19) pandemic. Over one in every hundred deaths is due to suicide, and nearly 60% of all suicides happen to those under 50. The isolation because of lockdown, fear and uncertainty due to job cuts and general discomfort due to the inability to control several aspects of life has triggered severe mental trauma in people. Increased use of social media have been implicated which encourages the isolation in youth. It is crucial at this time to take care of one's mental health. There are hardly any steps being taken at the scale required to manage this increasing number of people with mental health issues. There is a huge gap in the treatment that should be available and of the help available at hand, easily and cost-effectively. Even in developed countries, the ratio of psychiatrists, psychologists, psychiatric social workers, and mental health nurses to patients is 1: 10,000. The lacuna in the system ensures that most people suffering from mental health issues are never able to get the help they need.

Web based chat bots can be a solution to help users manage their mental health. It will interact with the users through text chatting. The users will type in their questions and send them to the Chat bots which will reply with best answer. This application also includes a psychological testing gives a breadth of information in a fairly short duration of time regarding numerous facets of a person’s life, including their overall cognitive ability, personal behaviors, traits, and personality functioning. It gives score obtained through psych testing which helps in providing the individual insight into their world through their unique perspective. This could assist in providing timely care to individuals who may be unwilling or unable to consult a healthcare provider.

Process Model

Incremental Model will be best suited for this software. The Software satisfies the following requirements for selection of Incremental model:

- The requirements are superior.
- Errors are easy to be recognized.
- Simple to manage risk because it handled during its iteration.
- Software application is large.

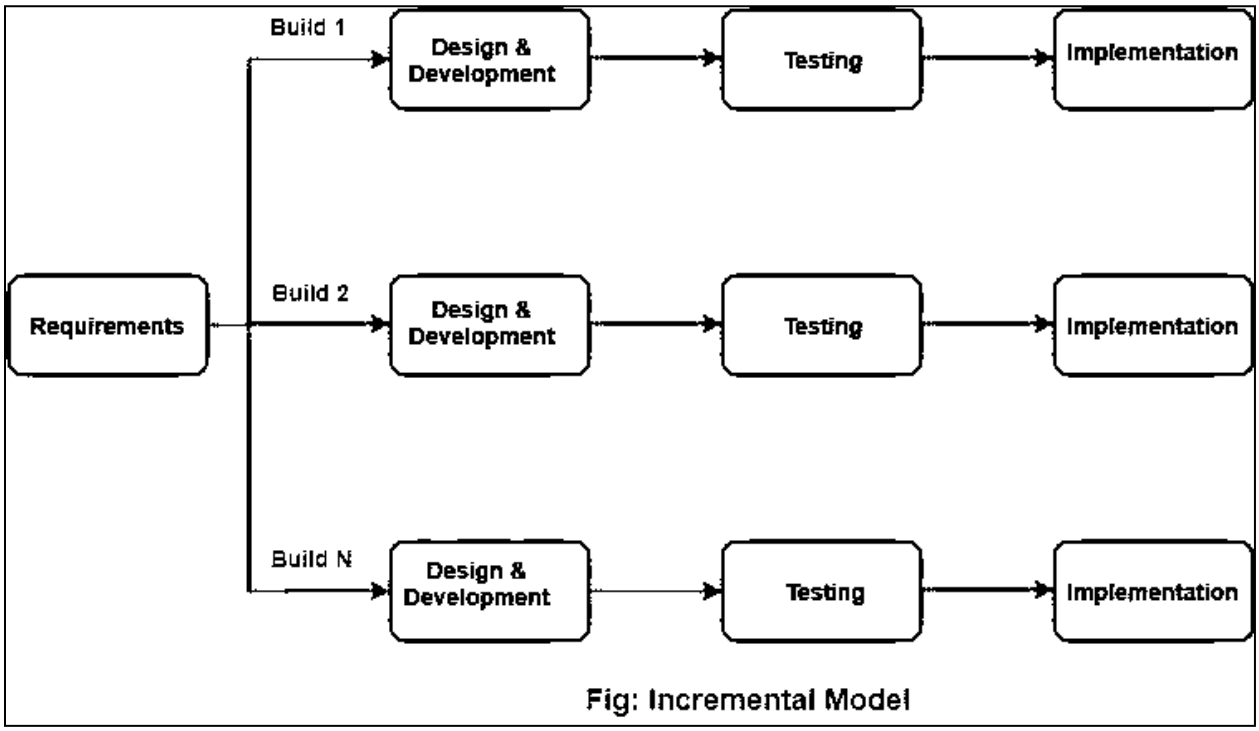


Fig: Incremental Model

Ref : Software Engineering (3rd ed.), By K.K Aggarwal & Yogesh Singh

CHAPTER 1: Software Requirement Specification (SRS)

1.1 Overall Description

Due to the fact that people are not as forthcoming about their emotions these days, psychologists face difficulties in diagnosing mental health problems. Approximately one in four adults, or 26% of those aged 18 and above, are affected by a diagnosable mental disorder each year. Many individuals experience multiple mental health issues at any given moment. Depression is often linked with anxiety and substance abuse disorders. Approximately 9.5% of adults over 18 year's old experience major depression, bipolar disorder, or dysthymia each year.
This Project offers a solution to this issue by having conversations with individuals, analyzing their data to determine if they have a mental illness, and suggesting helpful activities to overcome it. If needed, the chatbot will also recommend reputable psychologists in the person's vicinity to ensure they receive the proper treatment.

1.1.1 Product Functions

Our Chatbot performs the following functions:

1. **Screening**: Our chatbot is able to conduct mental health screenings, such as assessments for depression, anxiety, and other mental health conditions.
2. **Providing Information**: The chatbot provides an accurate and reliable information about mental health conditions, treatment options, and coping strategies.
3. **Guided Meditations and Relaxation Techniques**: Our chatbot offers guided meditations, breathing exercises, and other relaxation techniques to help users manage stress and anxiety.
4. **Connection to Professional Help:** Our chatbot recommend users with licensed mental health professionals if necessary, and provide information about available resources in their community.
5. **Encouragement and Support:** Our chatbot is trained in such a way that it offer encouragement, support, and a safe and non-judgmental space for users to discuss their mental health.
6. **Personalized Insights and Recommendations:** Our chatbot is cable of analyzing the user's data and provide personalized insights and recommendations based on their specific needs and symptoms.

By performing these functions, our chatbot can help individuals to better understand and manage their mental health, and provide support and resources to help them overcome mental illness.

1.1.2 User Characteristics

User1: Admin – Admin will be able to track the user's progress, and Admin will have access to or authorization to amend or update the user's information. Admin will also have the authority to remove a user if any inappropriate behavior is observed.
User2: User – The user will be able to register once and login using the same userid and password that they provided when they registered. The user must be at least 15 years old and be able to read and write in English.

1.2 External Interface Requirements

1.2.1 User Interfaces

1. **Ease of Use:** The chatbot is designed with a user-friendly interface that is easy to navigate and understand, with clear navigation and engaging design, as well as straightforward language.
2. **Customizable:** Users can personalize their experience by adjusting the background color or font size.
3. **Multiple Modes of Interaction:** To accommodate different user preferences and needs, the chatbot offers various modes of interaction, including text-based conversations, voice recognition, and touch gestures.
4. **Quick Responses:** The chatbot provides immediate and concise feedback to user inputs, letting users know when it is processing or waiting for input.
5. **Inclusive Access:** The chatbot is designed to be accessible at all times for all users, including those with disabilities.

1.2.2 Hardware Interfaces

- **Laptop/Desktop PC**-Purpose of this is to be able to access the website from any ware and anytime, Minimum requirement of 2 GB RAM and i7 processor.
- **Laser Printer (B/W) -** This device is required for printing the recommended activities. It could be an optional requirement.
- **Wi-Fi router –** Wi-Fi router or a stable internet connection is need to run the website without any problem.

1.2.3 Software Interfaces

1. **APIs, or application programming interfaces:** The chatbot analyzes the conversation using machine learning algorithms to determine whether the individual has a mental health issue or not.
2. **Interfaces for data management:** The chatbot stores, retrieves, and analyzes data about the user's mental health history through data management interfaces.
3. **Interface for use:** A user interface may be used by the chatbot to display information and receive user inputs, such as voice commands or text-based conversations.
4. **Browser:** A browser application is need to run the website smoothly.

1.3 Functional Requirements

1.3.1 Secure Login: Our chatbot will feature a secure login system that requires users to provide personal information, such as a username and password, to access the Chabot's services.

1.3.2 Tailored Conversation: The chatbot will personalize the conversation by asking users about their mental health history, current symptoms, and preferences.

1.3.3 Emotional Support: The chatbot will be able to offer emotional support, understanding, and empathy to users who are facing mental health issues. This includes recognizing signs of distress and providing a responsive response.

1.3.4 Assessment and Guidance: The chatbot will have the ability to assess users' mental health symptoms and provide a set of recommended activities based on standardized criteria, in collaboration with a licensed mental health professional.

1.3.5 Personalized Recommendations: Based on the assessment, the chatbot will provide personalized treatment recommendations and connect users with relevant resources and support services. If needed, the chatbot will also recommend a reputable psychologist.

1.3.6 Progress Monitoring: The chatbot will monitor users' mental health progress and provide follow-up support and encouragement.

1.4 Performance Requirements

There are no particular extra performance requirement at this point of time.

1.5 Design Constraints

Software Restrictions: The application must adhere to the usual web application standards.
Hardware Constraints: At this time, no hardware constraints have been identified.
Acceptance Restrictions: In order to validate the system, the developers must accomplish the following tasks:
1. Upon request, demonstrate the operational system and its features.
2. Demonstrate that all important functional requirements have been met.
3. Provide enough test cases to demonstrate that the system is comprehensive and correct.
The system must be designed to be web-accessible. That is, the system must be constructed in such a way that it is simple to use and visible on the majority of browsers.

1.5.1 Standard Compliance Report Format:

All reports produced for this project adhere to the standard templates provided by the adviser in class.

Naming Conventions: All documents shall be named in accordance with IEEE standards.

1.6 Data Flow Diagram

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination

1.6.1 Context Level DFD

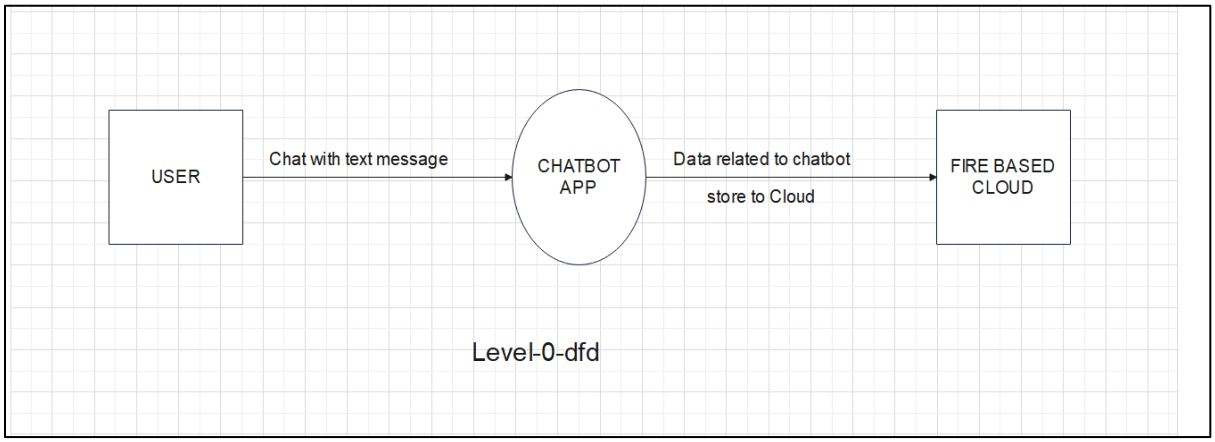


Figure 1.6.1 Context Level DFD

1.6.2 Data Flow Diagram Level 1

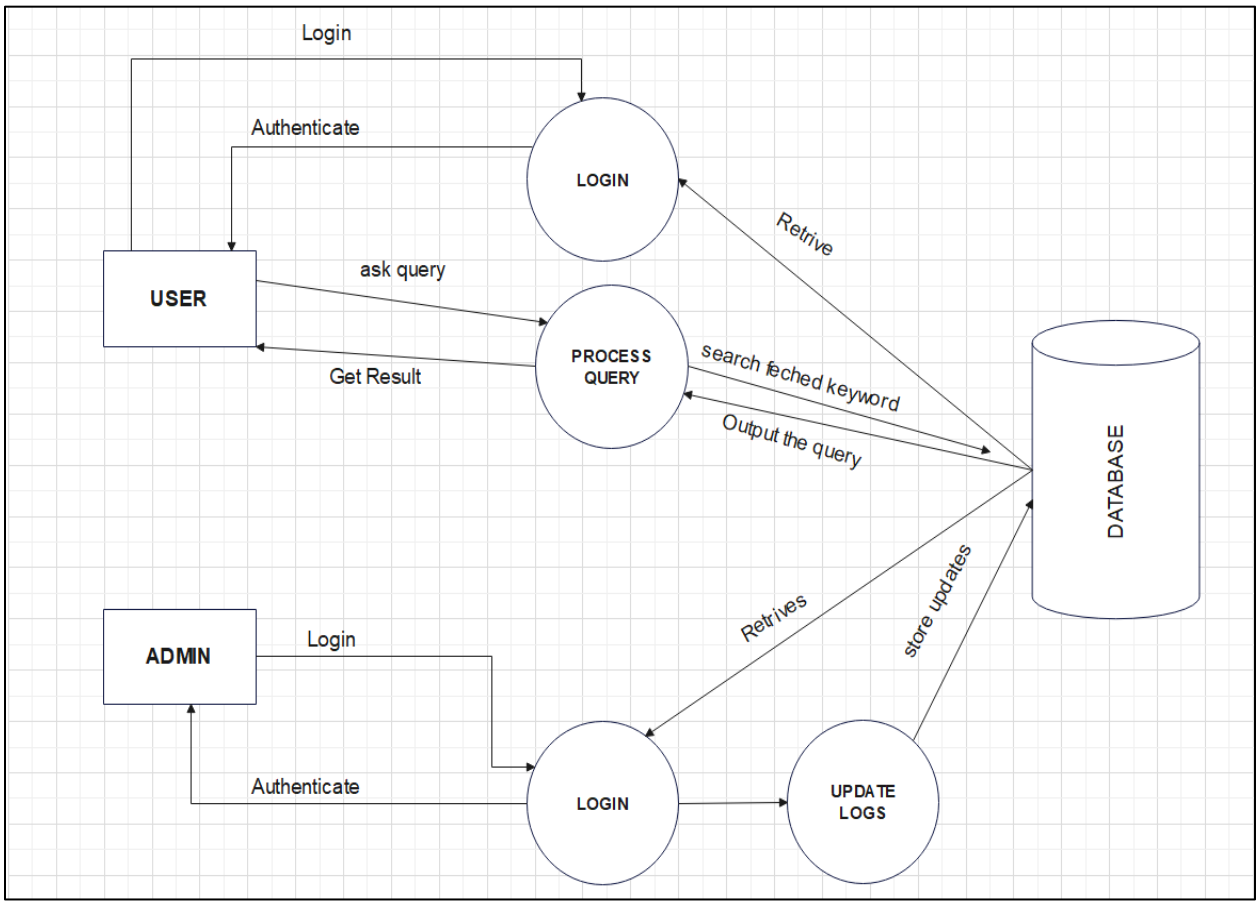


Fig 1.6.2 DFD Level 1

1.6.3 Data Flow Diagram Level 2

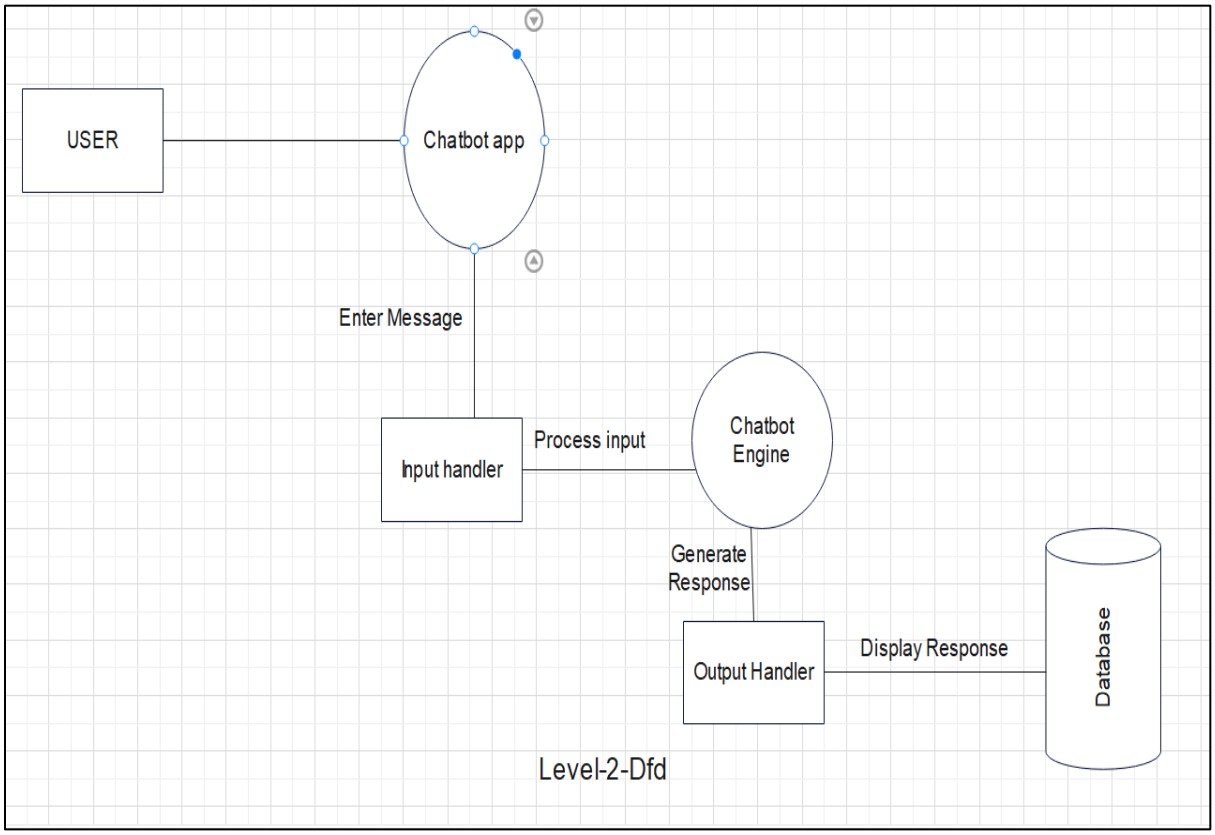


Fig 1.6.3 DFD Level 2

1.7 Data Dictionary

1.7.1 Range of values

S.No	Name	Values
1.	legal_character	[a-z| A-Z]
2.	Digit	[0-9]
3.	special_ch	[@|$|#|+|-]

Table 1.7.1 Range of values

1.7.2 Data Dictionary

S. No.	Name	Description	Range of Value
1.	User_Id	It is the id using which the user will be able to access the chatting section.	{legal_character + digit}*
2.	Password	Password is required for every user to verify that it logged in by authentic user.	{legal_character + digit + special_ch}*
3.	User_dob	Dob contains information about the user age.	
4.	Admin_name	It has the admin name	{legal_character}*
5.	Admin_id	Admin_id contains the id of admin which is unique.	{legal_character + digit}*
6.	Email_id	Contains information about email.	{ digit + special_ch +legal_character + digit+ special_ch+@+legal_character}*
7.	Mobile_no	Mobile no	{ digit }*
8.	Chat_id	Every user has a unique chat id to track their progress individually.	{legal_character + digit}*
9.	Act_id	Contains id of the activity.	{legal_character + digit}*
10.	Act_name	Contains name of activity.	{legal_character}*
11.	Act_type	Contains the type of activity recommended by the chatbot.	{legal_character}*

Table 1.7.2 Data Dictionary

1.8 Use Cases

1.8.1 Login

1.8.1.1 Brief Description

This use case describes how a user logs into the Mental Health Chat Bot.

1.8.1.2 Actors

Actor interact and participate in this use case is User.

1.8.1.3 Flow of Events

- **Basic Flow**

  This use case starts when the actor wishes to Login to the Chabot:

  1. The system requests that the actor enter his/her name, password.
  2. The actor enters his/her name, password.
  3. The system validates the entered name, password and logs the actor into the system.

- **Alternative Flow**

  If in the basic flow, the actor enters an invalid name, password, the system displays an error message. The actor can choose to either return to the beginning o the basic flow or cancel the login at which point the use case ends.

1.8.1.4 Special Requirements

None

1.8.1.5 Pre-conditions

All users must have User Account created for them in the system, prior to executing the use case.

1.8.1.6 Post Conditions

If the use case was successful, the actor is logged into the system. If not, the system state to unchanged.

1.8.1.7 Extension Points

None

1.8.2 Chat With Bot

1.8.2.1 Brief Description

This use case describes how the user chats with the Bot.

1.8.2.2 Actors

Actor interact and participate in this use case is User.

1.8.2.3 Flow of Events

- **Basic Flow**

This use case starts when the user wishes to chat with the Bot:

1. The actor enters his/her questions.
2. The system will respond with the best results and tries to eliminate the problem of the actors.

- **Alternative Flow**

None

1.8.2.4 Special Requirements

None

1.8.2.5 Pre-conditions

None

1.8.2.6 Post Conditions

If the actor wants any recommendation of the doctor, the system can provide them.

1.8.2.7 Extension Points

None

1.8.3 Psych Testing

1.8.3.1 Brief Description

This use case describes how the actor will take the test.

1.8.3.2 Actors

Actor interact and participate in this use case is User and Admin.

1.8.3.3 Flow of Events

- **Basic Flow**

This use case starts when the actors wishes to give the psychological test :

1. The system will give some set of the multiple choice questions which will help to evaluate the user.
2. The user will fill the best suited answer.
3. The system checks the answer given by the user and then it evaluates the mental health of the user by giving them grade out of five.

- **Alternate Flow**

None

1.8.3.7 Special Requirements

None

1.8.3.8 Pre-conditions

None

1.8.3.9 Post Conditions

If the user's mental health is severe. Then, the system will give the recommendations of the doctor for the consultancy**.**

1.8.3.10 Extension Points

None

1.8.4 Progress

1.8.4.1 Brief Description

This use case will show the progress of the user.

1.8.4.2 Actors

Actor interact and participate in this use case is User and Admin.

1.8.4.3 Flow of Events

- **Basic Flow**

  The user can check his/her progress report.

- **Alternate Flow**

  If the actor wants any recommendation of the doctor, the system can provide them.

1.8.4.4 Special Requirements

None

1.8.4.5 Pre-conditions

None

1.8.4.6 Post Conditions

None

1.8.4.7 Extension Points

None

1.8.5 Activities

1.8.5.1 Brief Description

This use case describes some of the activities like games, books, exercises to reduce the stress of the user.

1.8.5.2 Actors

Actor interact and participate in this use case is User.

1.8.5.3 Flow of Events

- **Basic Flow**

  The user can chooses one of the activities to reduce his/her stress.

- **Alternate Flow**

  None

1.8.5.4 Special Requirements

None

1.8.5.5 Pre-conditions

None

1.8.5.6 Post Conditions

None

1.8.5.7 Extension Points

None

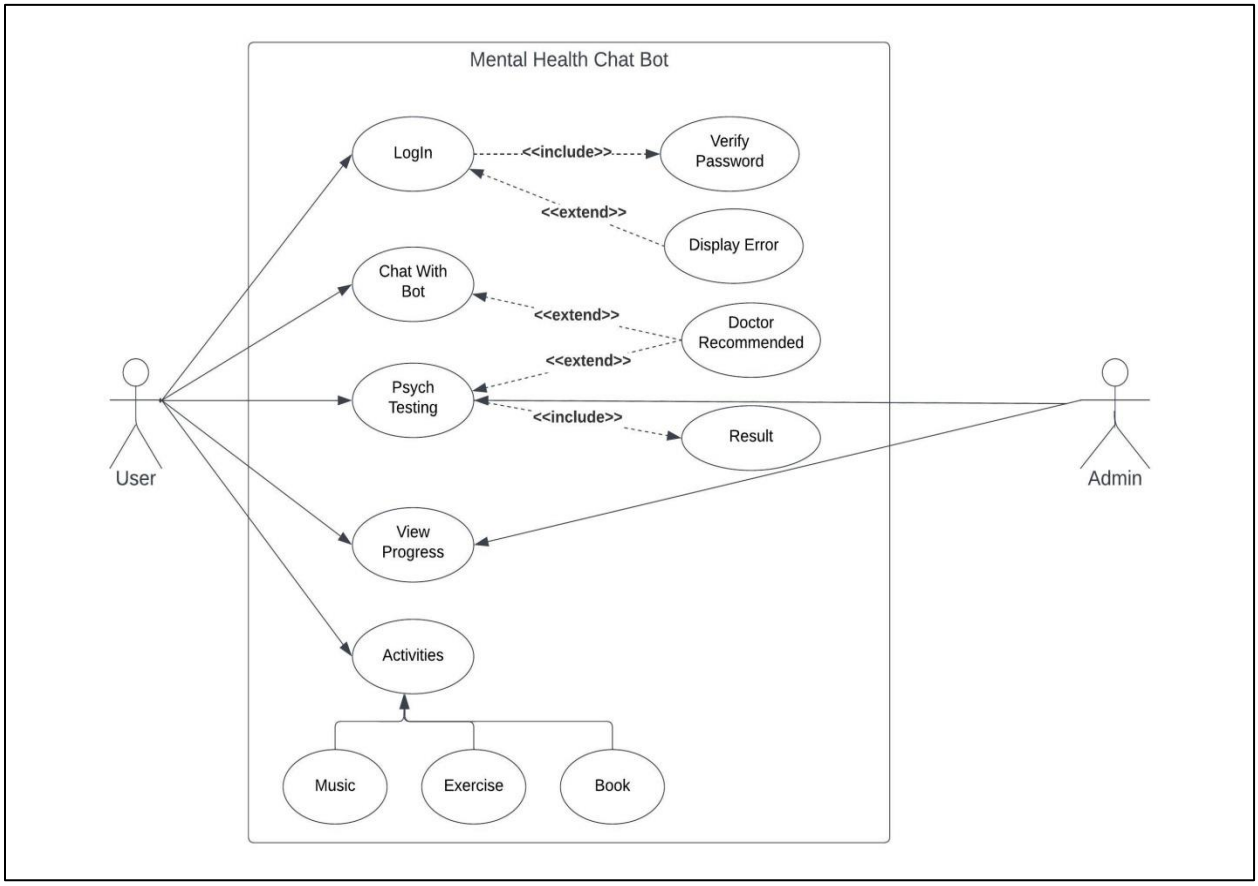


Fig 1.8 Use case Diagram

Chapter 2: Estimations

2.1 Function Point:

Information domain values are defined in the following manner:

• **Number of external inputs (EIs)** - Each external input originates from a user or is transmitted from another application and provides distinct application-oriented data or control information. Inputs are often used to update internal logical files (ILFs). Inputs should be distinguished from inquiries, which are counted separately.
• **Number of external outputs (EOs) -** Each external output is derived data within the application that provides information to the user. In this context external output refers to reports, screens, error messages, etc. Individual data items within a report are not counted separately.
• **Number of external inquiries (EQs) -** An external inquiry is defined as an online input that results in the generation of some immediate software response in the form of an online output (often retrieved from an ILF).
• **Number of internal logical files (ILFs) -** Each internal logical file is a logical grouping of data that resides within the application's boundary and is maintained via external inputs.
• **Number of external interface files (EIFs**). - Each external interface file is a logical grouping of data that resides external to the application but provides information that may be of use to the application.

Measurement parameters	**Count**	**Weighing factor (Average)**	**FP Count = Count x Weighing factor**
Number of User Inputs	4	4	4x4=16
Number of User Outputs	7	5	7x5=35
Number of User Inquires	1	4	1x4=4
Number of Internal Files	2	10	2x10=20
Number of External Interfaces	0	7	0x7=0
Count Total/Unadjusted Function Point Count (UFP)			16+35+4+20+0 = **75**

Table 2.1.1 Function Point

Therefore, **UFP (Unadjusted Function Point) = 258**

S.no	Questions	Value (F_i)
1	Does the system require reliable backup and recovery?	2
2	Are specialized data communications required to transfer information to or from the application?	0
3	Are there distributed processing functions?	0
4	Is performance critical?	4
5	Will the system run in an existing, heavily utilized operational environment?	2
6	Does the system require online data entry?	4
7	Does the online data entry require the input transaction to be built over multiple screens or operations?	3
8	Are the ILFs updated online?	3
9	Are the inputs, outputs, files, or inquiries complex?	2
10	Is the internal processing complex?	3
11	Is the code designed to be reusable?	3
12	Are conversion and installation included in the design?	2
13	Is the system designed for multiple installations in different organizations?	2
14	Is the application designed to facilitate change and ease of use by the user?	4
	$\Sigma(F_i)$	**34**

Table 2.1.2 Complexity Count

The **value adjustment factor (VAF)** (also called **Complexity adjustment factor**) is calculated as follows:

$$\mathbf{VAF = [0.65 + 0.01 * \Sigma (F_i)]}$$

Given, **Complexity adjustment factor = Value Adjustment Factor (VAF)**

= [0.65 + 0.01 * Σ (F_i)] = [0.65 + 0.01 * 34]

= 0.99

Adjusted FP Count = Unadjusted FP Count × VAF

= 75 * 0.99 = 74.25

Hence, Adjusted FP Count or FP $_{estimated}$ = 74 (approx.)

2.2 EFFORT

COnstructive COst MOdel (COCOMO II) is a more comprehensive estimation model.

COCOMO II is actually a hierarchy of estimation models that address the following areas:

- **Application composition model** - Used during the early stages of software engineering, when prototyping of user interfaces, consideration of software and system interaction, assessment of performance, and evaluation of technology maturity are paramount.
- **Early design stage model -** Used once requirements have been stabilized and basic Software architecture has been established.
- **Post-architecture-stage model** - Used during the construction of the software.

The COCOMO II model require sizing information. Three different sizing options are available as part of model hierarchy:

1. **Object points**
2. **Function points**
3. **Lines of source code**

Like function points the object point is an indirect software measure that is computed using counts of the number of screens, reports and components likely to be required to build the application. Each object instance is classified into one of the three complexity levels (simple, medium or difficult) as shown in the following table:

Object Type	**Weighing Factor**		
	Simple	**Medium**	**Difficult**
Screen	1	2	3
Report	2	5	8

3GL Component			10

Table 2.2.1 Complexity Level

The object count is determined by multiplying the total number of object instances by weighing factor in the figure and summing to obtain a total object point count. When component based development or general software reuse is to be applied, the percent of reuse is estimated

(%reuse) and the object point count is adjusted.

NOP = (object points) * [(100 - % reuse)/100]

Where NOP is defined as new object points.

To derive an estimate of effort based on the computed NOP value, a "productivity rate" must be

Derived.

PROD = NOP / person-month

Productivity rate for different level of developer experience and development environment

Maturity:

Developer's experience/capability	Very low	Low	Nominal	High	Very High
Environment maturity/capability	Very low	Low	Nominal	High	Very High
PROD	4	7	13	25	50

Table 2.2.2: Productivity Rate

Once the productivity rate has been determined, an estimate of project effort is computed using –

Estimated effort = NOP / PROD

COCOMO estimation for our project:

1.) Number of screens = 11
2.) Number of reports = 4

3.) Number of 3GL components used = 0

Considering all of the above having medium complexity, 0% reuse of the components and taking the developer experience and environment maturity as low we get the following estimates :-

1. Object Points = (11 * 2) + (4 * 5) = 22 + 20
= 42

Object Points = 42

2. NOP = (object points) * [(100 - % reuse)/100]

= (42) * [(100 – 0)/100] = 42

New Object Points (NOP) = 42

3. Effort Estimate

Estimated effort = NOP / PROD

Taking the developer experience and environment maturity as low we get
PROD = (7 + 7) / 2 = 7

Estimated effort = 42 / 7 = 6 Person-Month

Chapter 3: Scheduling

3.1 Project Scheduling

TASK	Planned Start	Actual Start	Planned complete	Actual Complete	Person Assigned
Problem Statement	Jan W1	Jan W1	Jan W1	Jan W1	Ritesh, Pinki
Software Model	Jan W1	Jan W1	Jan W1	Jan W1	Riya
SRS	Jan W2	Jan W2	Feb W2	FebW2	Ritesh, Pinki,Riya,
DFD level-1	Jan W3	Jan W3	Jan W3	Jan W3	Ritesh
DFD level-2	Jan W4	Jan W4	Jan W4	Jan W4	Ritesh
Data Dictionary	Feb W1	Feb W1	Feb W1	Feb W1	Pinki
Use Case	Feb W2	Feb W2	Feb W2	Feb W2	Riya
Use Case Description	Feb W3	Feb W3	Feb W3	Feb W3	Riya
Estimations – Function Point	Mar W1	Mar W1	Mar W1	Mar W1	Ritesh
Estimations – COCOMO II	Mar W2	Mar W2	Mar W2	Mar W2	Pinki
Project Scheduling	Mar W2	Mar W2	Mar W2	Mar W2	pinki
Risk Management	Mar W3	Mar W3	Mar W3	Mar W3	Pinki
Design – System Design	Mar W4	Mar W4	Mar W4	Mar W4	Ritesh
Design – Screen Design	Mar W4	Mar W4	Mar W4	Mar W4	Ritesh
ER Diagram	Apr W1	Apr W1	Apr W1	Apr W1	Ritesh
Design – Database Design	Apr W2	Apr W2	Apr W2	Apr W2	Ritesh,Pinki
Testing	Apr W2	Apr W2	Apr W3	Apr W3	Riya,Ritesh

Table 3.1: Project Scheduling

Jan - January Feb – February Mar – March Apr – April W – Week

3.2 Timeline-chart

Month	**January**				**February**			**March**				**April**		
Week	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**1**	**2**	**3**	**4**	**1**	**2**	**3**
1.1 Problem Statement														
2.1 Software Model														
3.1 SRS **3.2** DFD level1 **3.3** DFD level2 **3.4** Data Dictionary **3.5** Use Case **3.6** Use Case Description														
4.1 Estimations –FP **4.2** Estimations –COCOMO II														
5. Project Scheduling														
6. Risk Management														
7.1 Design – System **7.2** Design – Screen Design **7.3** ER Diagram														
8. Testing														

Table 3.2 Time Line Chart

CHAPTER 4: RISK MANAGEMENT

Risk is an expectation of loss, a potential problem that may or may not occur in the future. It is generally caused due to lack of information, control or time. A possibility of suffering from loss in software development process is called a software risk. Loss can be anything, increase in production cost, development of poor quality software, not being able to complete the project on time.

Risk always involves the following two traits:

- **Uncertainty:** The risk may or may not occur; there are no risks that are 100% likely to occur.
- **Loss:** If the risk becomes a reality, undesirable consequences or losses will occur.

When analyzing risks, it's critical to put each risk's level of uncertainty and degree of loss into numerical terms.

A software risk can be of two types:

- **Internal risks:** Risks that are within the control of the project manager.
- **External risks:** Risks that are beyond the control of project manager.

Broadly we have to deal with the risks arising from three possible cases:

1. **Technical risks**: Technical risk is related with the design, implementation, and maintenance of software. Complex software architecture, programming language constraints, software defects, or hardware incompatibility can all create problems.
2. **Project management risk:** Risks related with project management include those linked with project planning, execution, monitoring, and control. Poor project planning, a lack of resources, insufficient communication, or unclear needs can all lead to these hazards.
3. **Business Risk:** Business risks are related with the influence of software on a company or organization. These risks can occur as a result of market competition, shifting customer needs, or financial instability.

4.1 RISK IDENTIFICATION

- **Product size** - Risks associated with the overall size of the software to be built or modified.
- **Development environment** - Risks associated with the availability and quality of the tools to be used to build the product.
- **Technology to be built** - Risks associated with the complexity of the system to be built and the “newness” of the technology that is packaged by the system.

- **Team size and experience** - Risks associated with the overall technical and project experience of the software engineers who will do the work.
- **Incorrect Responses** - The chatbot may provide you with incorrect or insufficient advice.
- **Privacy and security** - Attacks via the internet or hacking may be targeted at user data.
- **Inadequate personal** - Interactive Users may lose out on opportunities for personal interaction.
- **Bias and discrimination** - Algorithms in ChatBot have the ability to encourage bias and discrimination. A chatbot may be unable to handle all sophisticated mental health conditions.

4.2 RISK ASSESMENT OF OVERALL PROJECT

A project risk assessment entails finding, analyzing, and evaluating potential risks to the project's success. Here are some questions which had been derived from risk data obtained by surveying experienced software project managers for conducting a project risk assessment:

1. Have top software and customer managers formally committed to support the project?
Ans. Yes
2. Are end users enthusiastically committed to the project and the system/product to be built?
Ans. Yes
3. Are requirements fully understood by the software engineering team and its customers?
Ans. Yes
4. Have customers been involved fully in the definition of requirements?
Ans. Yes
5. Do end users have realistic expectations?
Ans. Yes
6. Is the project scope stable?
Ans. Yes
7. Does the software engineering team have the right mix of skills?
Ans. Yes
8. Are project requirements stable?
Ans. No
9. Does the project team have experience with the technology to be implemented?
Ans. Yes
10. Is the number of people on the project team adequate to do the job?
Ans. Yes
11. Do all customer/user constituencies agree on the importance of the project and on the requirements for the system/product to be built?
Ans. Yes

4.3 RISK PROJECTION

Risk projection, also called risk estimation, attempts to rate each risk in two ways—

(1) The likelihood or probability that the risk is real and will occur and
(2) The consequences of the problems associated with the risk, should it occur.

4.3.1 RISK TABLE FORMATION

In software engineering, a risk table is a document that highlights potential risks or difficulties that could arise during a software project, as well as their related probability and impact. A risk table's goal is to assist project teams in anticipating and preparing for probable problems, as well as prioritising risk reduction measures.

RISKS	CATEGORY	PROBABILITY	IMPACT
Development environment	Project Risk	15%	Medium
Team size and experience	Project Risk	30%	Low
Privacy and security	Techical Risk	20%	High
Inadequate personal	Techincal Risk	10%	Medium
Incorrect Responses	Techincal Risk	30%	High
Bias and discrimination	Technical Tisk	15%	Medium
Hardware Failure	Techincal Risk	40%	Critical

TABLE 4.3.1

According to this table, the most serious risks associated with a mental health chatbot are incorrect responses and privacy/security issues, both of which have a high chance and impact. While the risk of bias and discrimination is modest, it could have a significant impact if it occurs. Although the risks of a lack of human connection and limited treatment scope are minimised in both probability and impact, they must still be considered when developing and implementing a mental health chatbot.

4.4 RISK MITIGATION, MONITORING, AND MANAGEMENT (RMMM) PLAN

The RMMM plan is a risk management plan that specifies the strategies and procedures for detecting, analyzing, mitigating, and monitoring risks throughout a project's life cycle. Risk Management, Monitoring, and Mitigation is abbreviated as RMMM.If a software team adopts a proactive approach to risk, avoidance is always the best strategy. This is achieved by developing a plan for risk mitigation

RISKS	**RMMM**
Development environment	Using a development environment in which every member of the team is experienced and able to work properly.
Team size and experience	Appointing adequate number of teams members to build the project, the team members should be experienced with the technology and have the will to learn more.
Privacy and security	To protect user data, utilize encryption, secure authentication systems, and frequent security audits. Comply with applicable data privacy laws.
Inadequate personal	Include resources that will connect users to professional mental health care providers, such as a referral system or a live chat option. Encourage users to seek extra assistance outside of the chatbot.
Incorrect Responses	To increase accuracy and keep the Chabot’s knowledge base up to date, use natural language processing (NLP) and machine learning methods. Implement quality assurance and testing techniques to detect faults and inaccuracies.
Bias and discrimination	Review chatbot data on a regular basis to discover and correct any prejudice or discrimination in the Chabot’s responses. Make sure the chatbot is created with inclusivity and diversity in mind.
Hardware Failure	Invest in good quality hardware.

Table 4.4.1: RMMM Plan Table

CHAPTER 5: DESGIN

5.1 System Design

5.1.1 Architecture Design

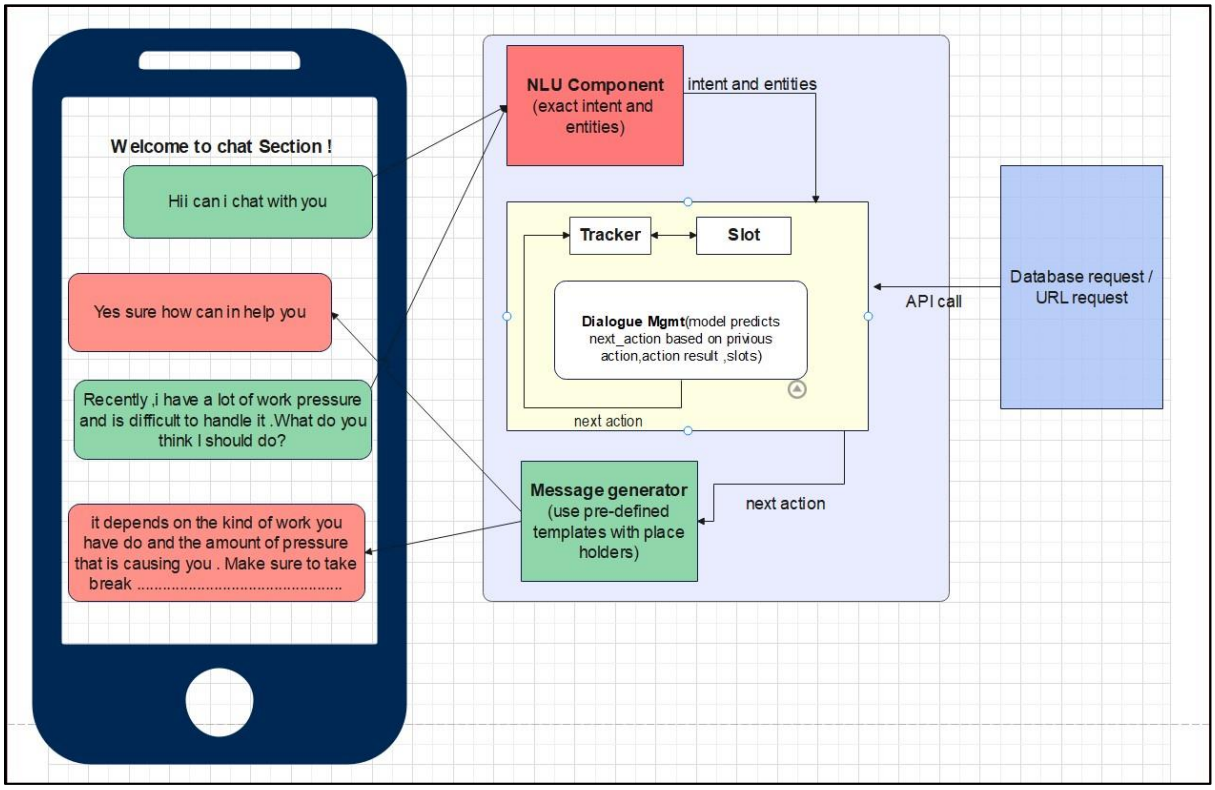


Fig 5.1.1 Overall Chatbot Architecture

5.1.2 Login Architecture

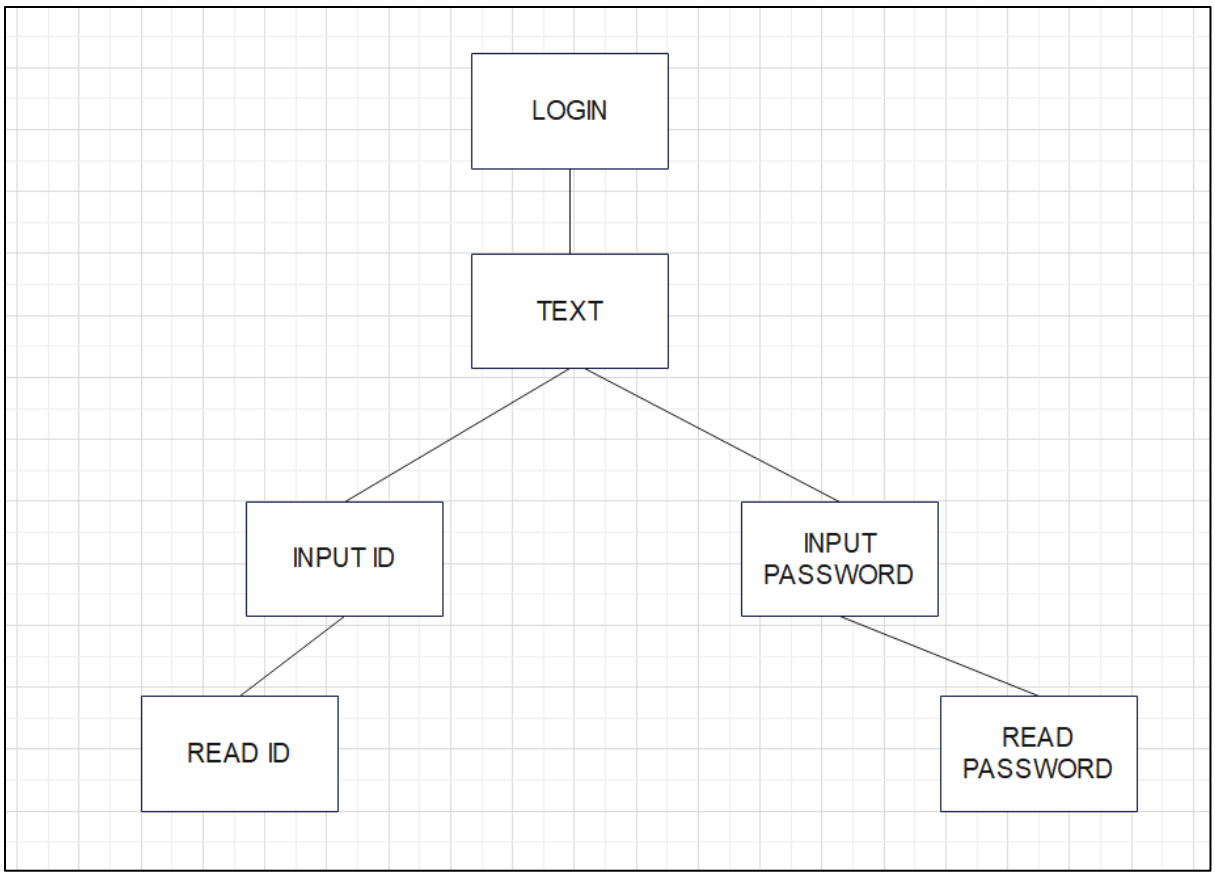


Fig 5.1.2 Login Architecture

5.1.3 Sign up Architecture

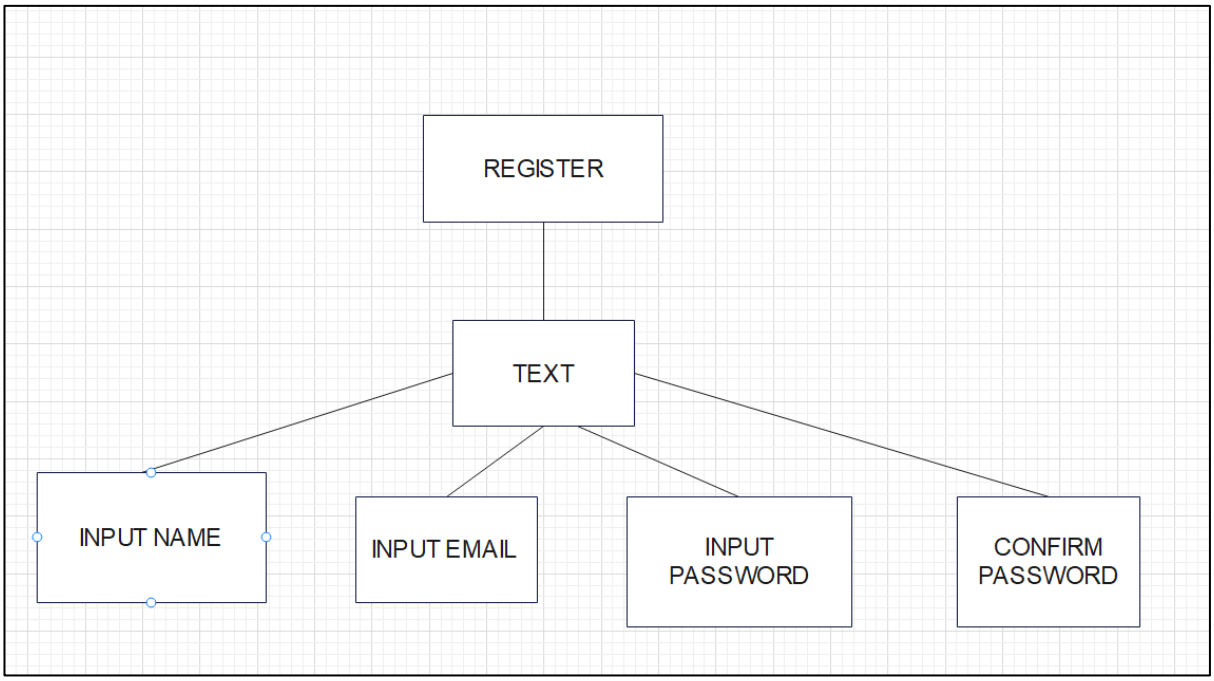
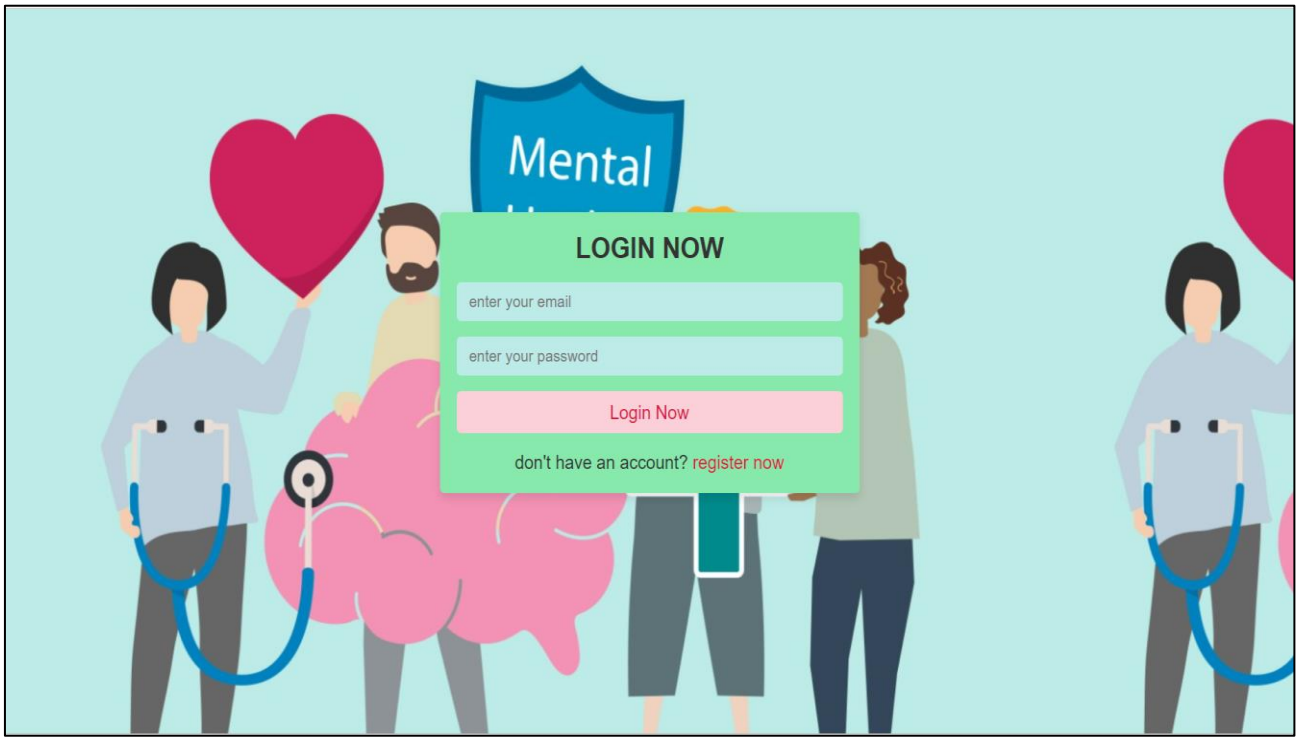


Fig 5.1.3 Sign up Architecture

5.2 Screen Design

5.2.1 Login Screen Design



No. of External Inputs: 4

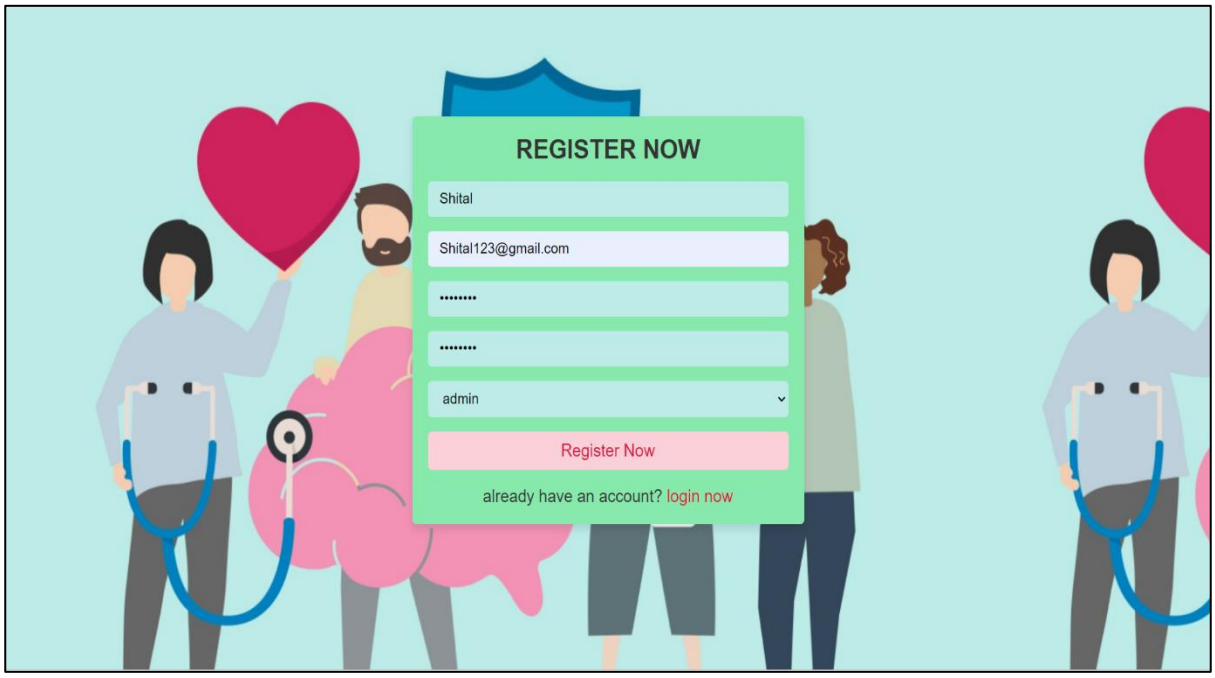
No. of External Outputs: 2

No. of External Inquiries: 0

No. of Internal Logical File: 2

No. of External Interface files: 0

5.2.2 Admin Login



No. of External Inputs: 7

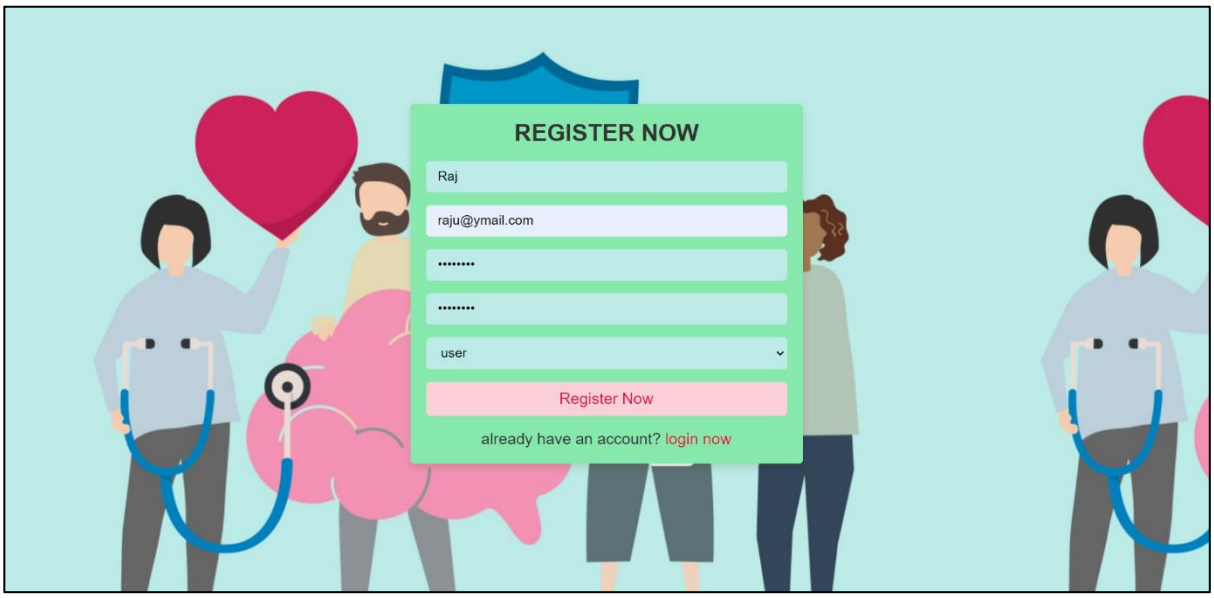
No. of External Outputs: 2

No. of External Inquiries: 0

No. of Internal Logical File: 2

No. of External Interface files: 0

5.2.3. USER LOGIN



No. of External Inputs: 7

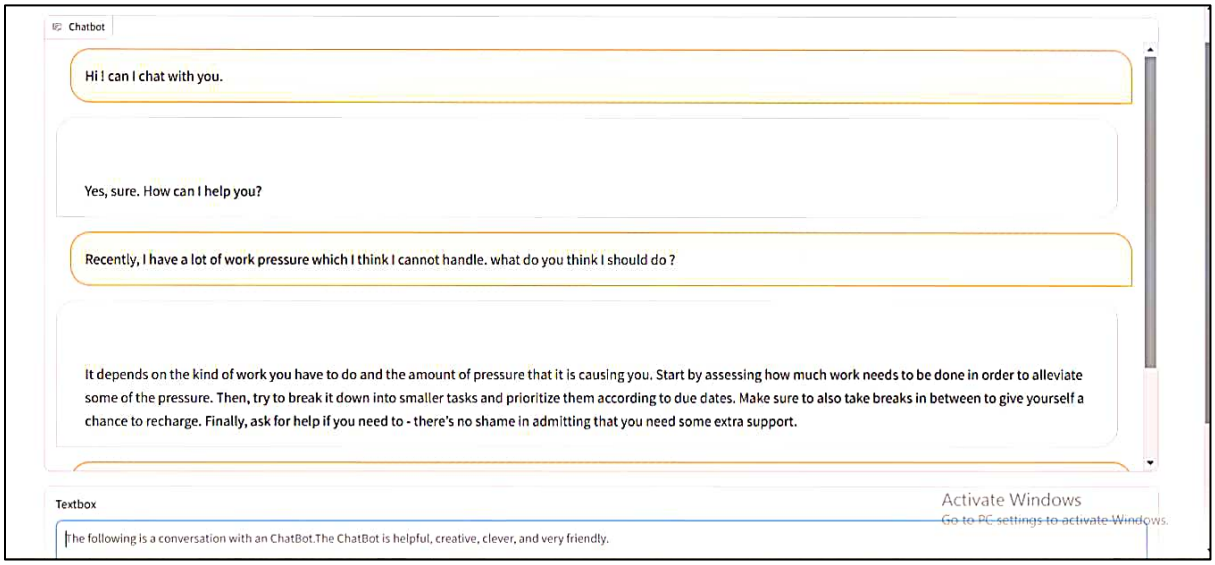
No. of External Outputs: 2

No. of External Inquiries: 0

No. of Internal Logical File: 2

No. of External Interface files: 0

5.2.4 Chatbot Discussion



No. of External Inputs: 1

No. of External Outputs: 1

No. of External Inquiries: 0

No. of Internal Logical File: 0

No. of External Interface files: 0

5.3 ER Diagram

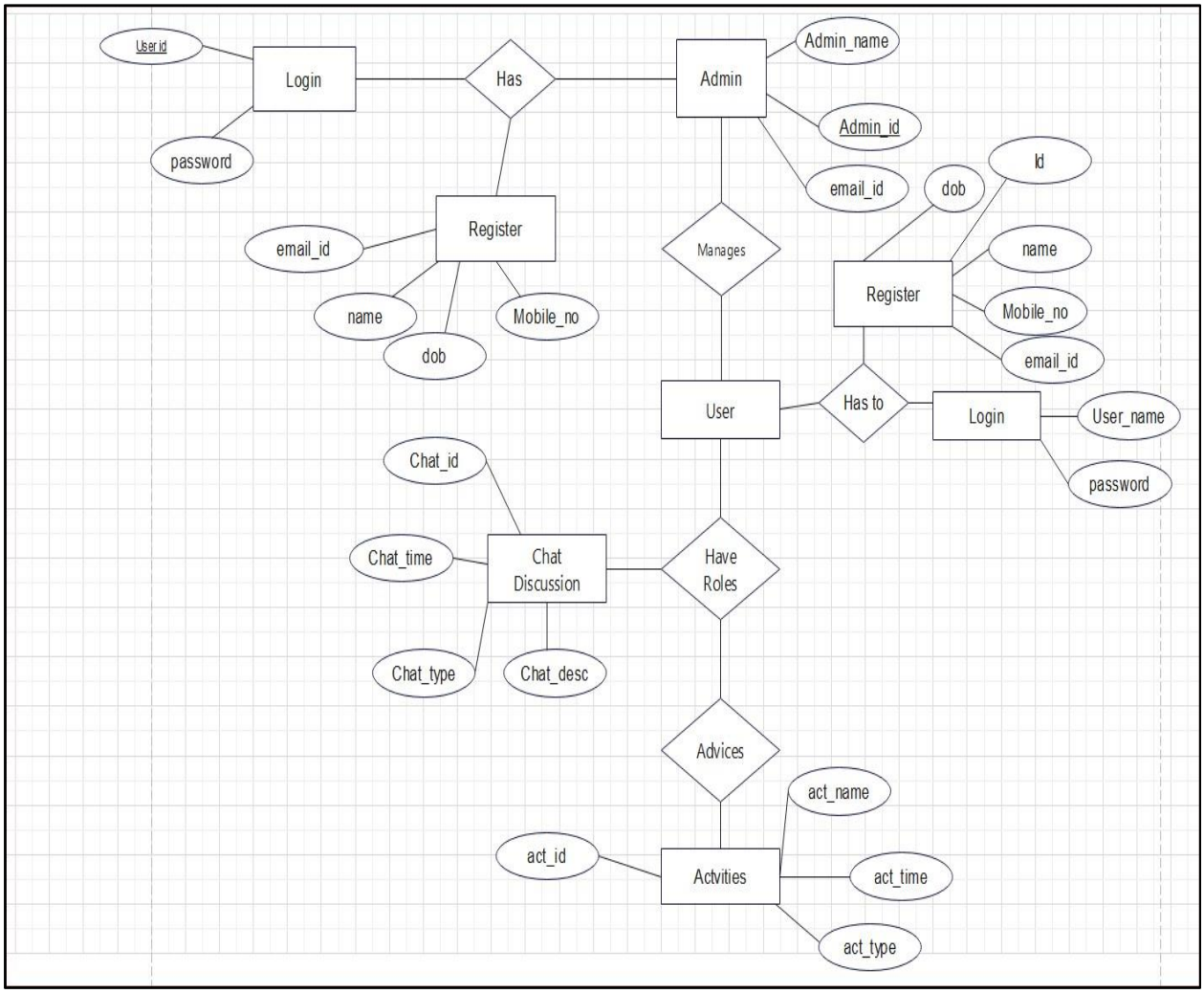


Fig 5.3.1 ER Diagram

5.4 Database Design

5.4.1 Login

Field	Type	Description
User_id	**Varchar(20)**	It is the id using which the user will be able to access the chatting section.
Password	**Varchar(20**	Password is required for every user to verify that it logged in by authentic user.

5.4.2 Register

Field	Type	Description
User_id	**Varchar(20)**	It is the id using which the user will be able to access the chatting section.
Password	**Varchar(10)**	Password is required for every user to verify that it logged in by authentic user.
Email_id	**Varchar(50)**	Contains information about email.
Name	**Char(20)**	Contains information about Name.
Dob	**Date**	Contains date of birth of user.
Mobile_No	**Int(10)**	Contains Mobile number of user.

5.4.3 Admin

Field	Type	Description
Admin_Name	**char(30)**	Contains information about name of admin.
Admin_id	**Varchar(10)**	It is the id using which the user will be able to access the website.
Email_id	**Varchar(50)**	Contains information about email of admin

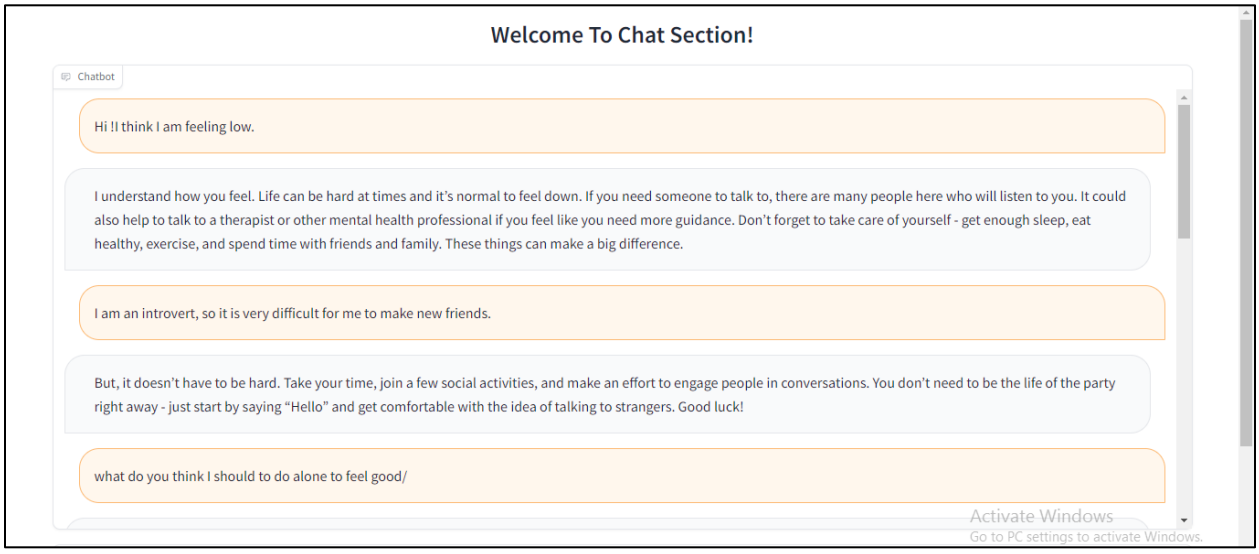
5.4.4 Activities

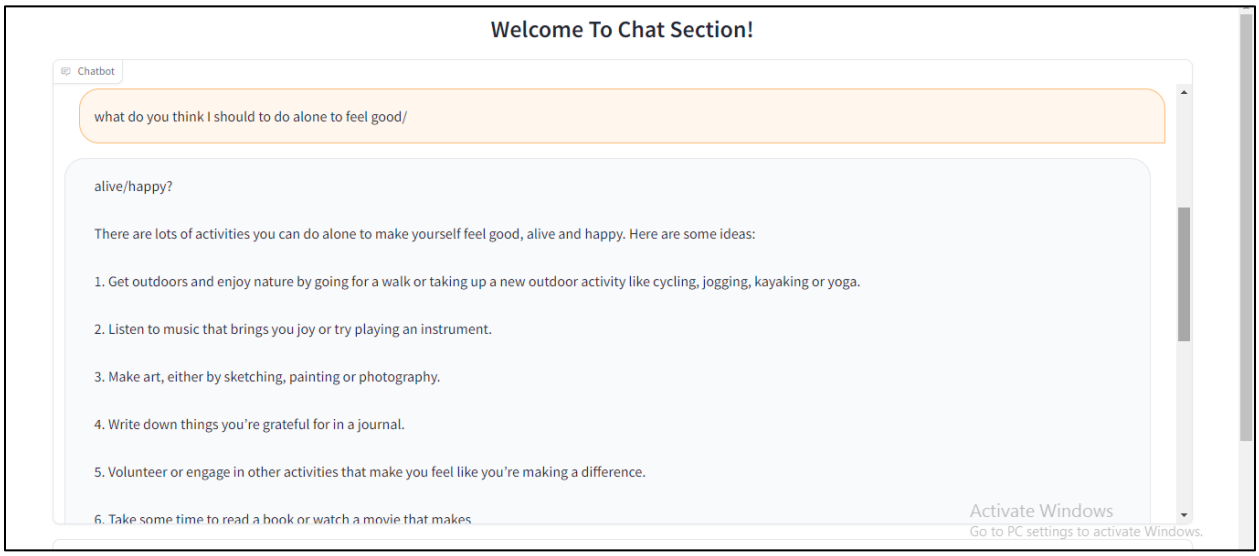
Field	Type	Description
Activities_Name	**char(30)**	Contains information about name of activities.
Activities _id	**Varchar(10)**	It is the id using which the user will be able to identify the activity uniquely.
Activities _type	**char(50)**	Contains information about the type of Activities

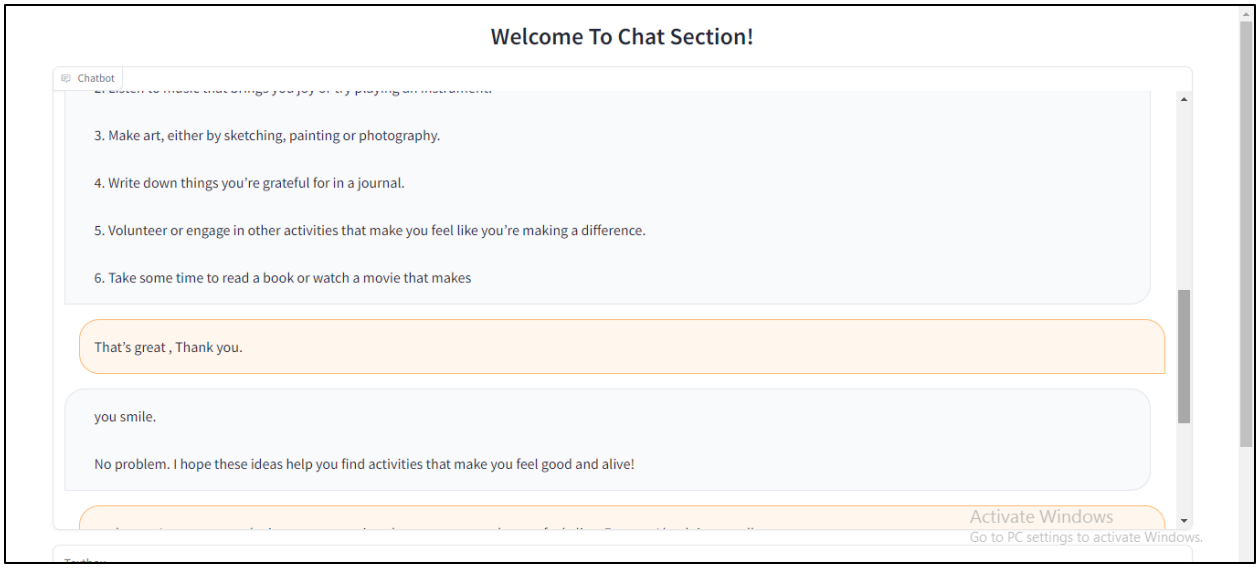
5.4.5 Chat

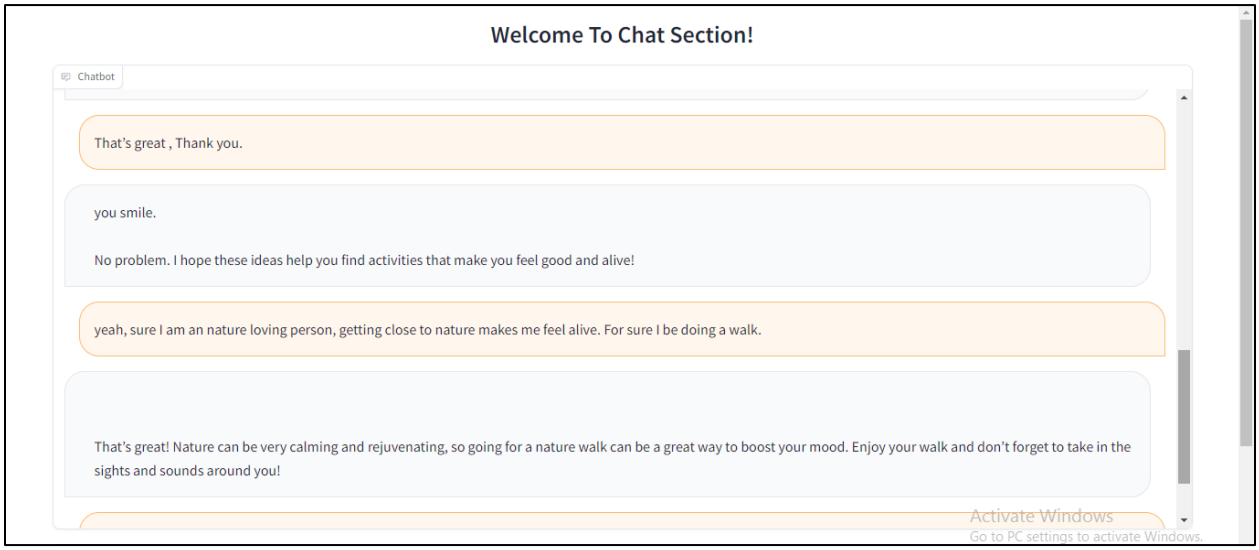
Field	Type	Description
User_id	**Varchar(20)**	It is the id using which the user will be able to access the chatting section.
Password	**Varchar(20)**	Password is required for every user to verify that it logged in by authentic user.
Chat_id	**Varchar(20)**	Every user has a unique chat id to track their progress individually.

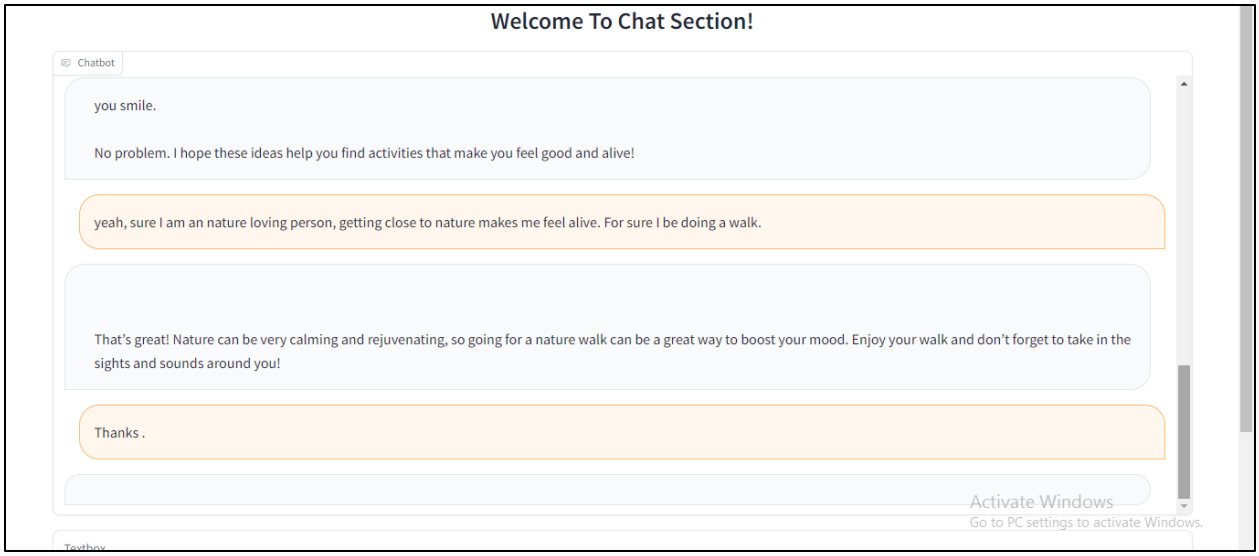
CHAPTER 6: SAMPLE SCREEN SHOTS






Welcome To Chat Section!
Chatbot
3. Make art, either by sketching, painting or photography.
4. Write down things you're grateful for in a journal.
5. Volunteer or engage in other activities that make you feel like you're making a difference.
6. Take some time to read a book or watch a movie that makes
That's great , Thank you.
you smile.
No problem. I hope these ideas help you find activities that make you feel good and alive!
Activate Windows
Go to PC settings to activate Windows.


Welcome To Chat Section!
Chatbot
That's great , Thank you.
you smile.
No problem. I hope these ideas help you find activities that make you feel good and alive!
yeah, sure I am an nature loving person, getting close to nature makes me feel alive. For sure I be doing a walk.
That's great! Nature can be very calming and rejuvenating, so going for a nature walk can be a great way to boost your mood. Enjoy your walk and don't forget to take in the sights and sounds around you!
Activate Windows
Go to PC settings to activate Windows.


Welcome To Chat Section!
Chatbot
you smile.
No problem. I hope these ideas help you find activities that make you feel good and alive!
yeah, sure I am an nature loving person, getting close to nature makes me feel alive. For sure I be doing a walk.
That's great! Nature can be very calming and rejuvenating, so going for a nature walk can be a great way to boost your mood. Enjoy your walk and don't forget to take in the sights and sounds around you!
Thanks .
Activate Windows
Go to PC settings to activate Windows.

CHAPTER 7: TESTIING

7.1 Flow graph

A Control Flow Graph (CFG) is the graphical representation of control flow or computation during the execution of programs or applications. Control flow graphs are mostly used in static analysis as well as compiler applications, as they can accurately represent the flow inside of a program unit.

Pseudo code for flow graph

1. ChatBot

2. Select login or register or logout

3. if (login selected){

4. Enter username

5. Enter password

6. Click Login

7.}

8. else{

9. Enter Name

10. Enter email

11. Enter Password

12. Enter Password

13. Select user or admin

14. Click Register

15.}

16. Select chatting or psychological test or activities

17. if (chatting selected){

18. Chat with the Bot

19.}

20. else if(psychological test selected){

21. Fill the answers of given questions

22. Click submit

23. Result will be generated

24.}

25. else {

26. Select Exercises or games or music or yoga

27. if (exercise selected){

28. Perform recommended exercises

29.}

30. else if (games selected){

31. Play games

32. }

33. else if (yoga selected){

34. Perform recommended yoga

35. }

36. else {

37. Listen music

38.}

39.}

40. End ChatBot

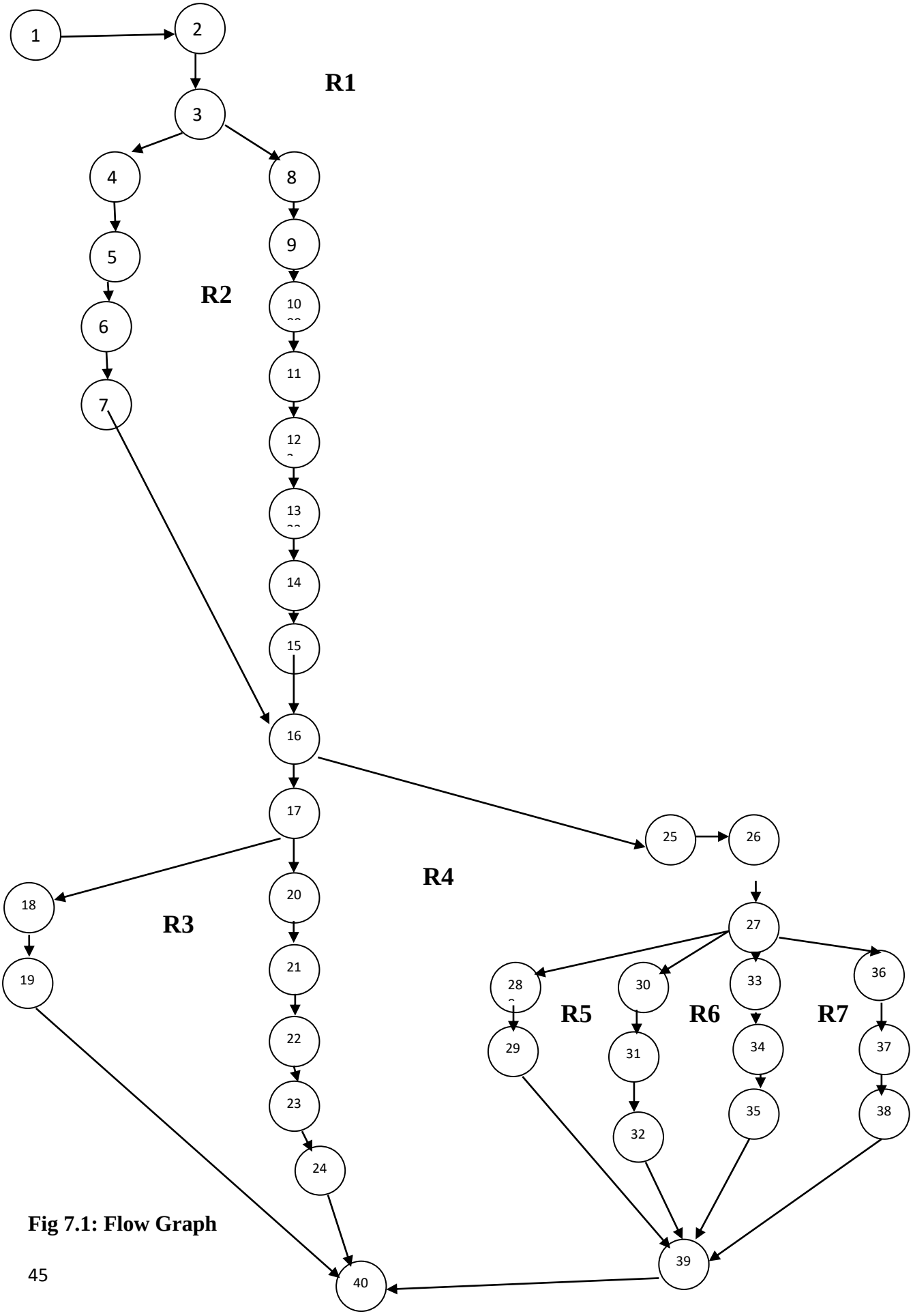


Fig 7.1: Flow Graph

7.2 Mapping table for DD Path Graph

Flow Graph Nodes	DD Path graph corresponding nodes	Remarks
1,2	A	Sequential Nodes
3	B	Decision Nodes
4 to 7	C	Sequential Nodes
8 to 15	D	Sequential Nodes
16	E	Two edges are joined
17	F	Decision Nodes
18,19	G	Sequential Nodes
20 to 24	H	Sequential Nodes
25,26	I	Sequential Nodes
27	J	Decision Nodes
28,29	K	Sequential Nodes
30 to 32	L	Sequential Nodes
33 to 35	M	Sequential Nodes
36 to 38	N	Sequential Nodes
39	O	Four Edges are joined
40	P	Three edges are joined

Table 7.2 Mapping table for DD Path Graph

7.2.3 DD Path Graph

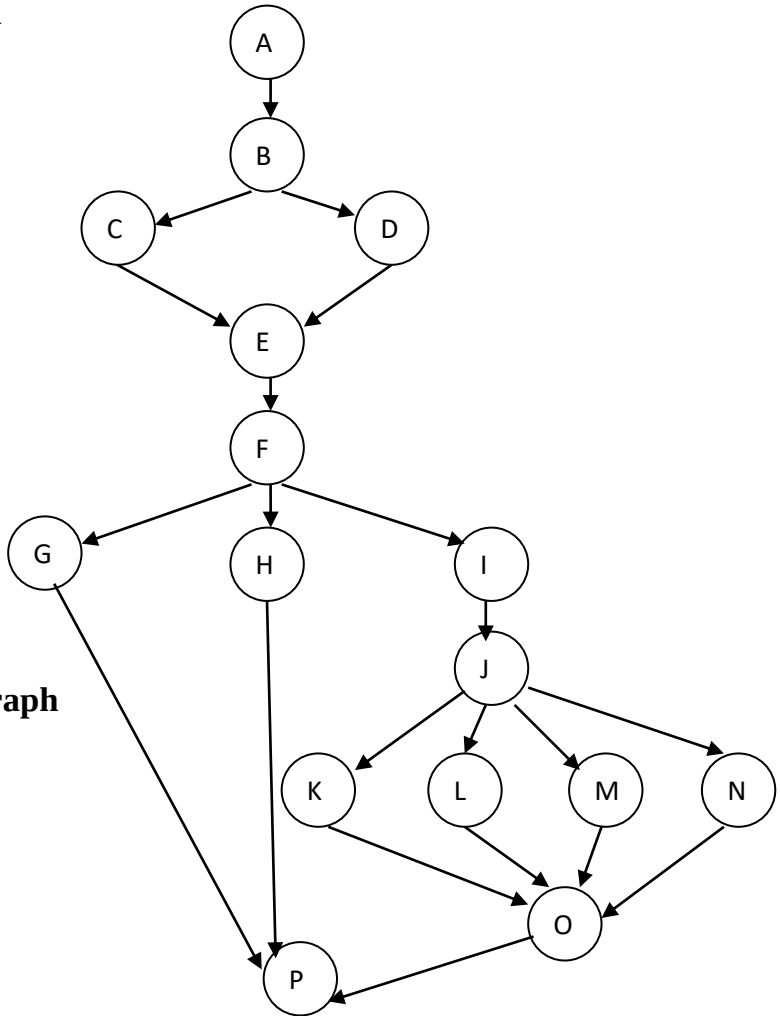


Fig 7.2.3 DD Path Graph

7.3 CYCLOMATIC COMPLEXITY V (G)

1. Cyclomatic complexity V (G) = Total number of Regions.

V (G) = 7

2. Cyclomatic complexity V (G) = (E – N) + 2

Where E = the number of flow graph edges. i.e. 21

N = the number of flow graph nodes. i.e. 16

V (G) = (21 – 16) + 2 = 7

3. Cyclomatic complexity V (G) = P + 1

Where P = the number of predicate nodes contained in the flow graph G = 6

V (G) = 6 + 1 = 7

There will be 7 independent Paths.

7.3.1 Independent Paths

Path 1: ABCEFGP

Path 2: ABDEFGP

Path 3: ABCEFHP

Path 4: ABDEFHP

Path 5: ABCEIJKOP

Path 6: ABDIJKOP

Path 7: ABCIJLOP

7.3.2 Regions

There are 7 regions in the as shown in the flow graph R1, R2, R3, R4, R5, R6, R7.

7.4 Test Cases

7.4.1 Test Case1

Project Name	Mental Health Chatbot								
Module Name	Login								
Created by:	Riya								
Created Date:	11-04-2023								
Executed by:	Riya								
Executed Date:	11-04-2023								
Test Case Id		Test Case Description	Pre Step	Test Step	Preconditions	Test Data	Expected Result	ActualResult	Status
Login Functionality		Verify Login functionality with valid username and password.		Navigate to Login Page.			Able to see the login Page.	As expected.	Pass
				Enter use name	Valid Username	raju@gmail.com	Credentials can be tered.	As expected.	Pass
				Enter Password.	Valid Password	XXXXXX@1	Credentials can be entered.	As expected.	Pass
				Click on Login Button.			User Logged	User Logged in Succesfully.	Pass
Login Functionality		Verify Login functionality with valic sername invalid password.		Navigate to Login Page.			Able to see the login Page.	As expected.	Pass
				Enter use name	Valid Username	shital@gmail.com	Credentials can be tered.	As expected.	Pass
				Enter Password.	Invalid Password	XXXXX@2	Credentials can be entered.	As expected.	Pass
				Click on Login Button.			User Logged	Unsuccesfully Logged	Fail
Login Functionality		Verify Login functionality with inval sername nd valid password.		Navigate to Login Page.			Able to see the login Page.	As expected.	Pass
				Enter use name	Invalid Username	pinki@gmail.com	Credentials can be tered.	As expected.	Pass
				Enter Password.	Valid Password	XXXXX@3	Credentials can be entered.	As expected.	Pass
				Click on Login Button.			User Logged	Unsuccesfully Logged	Fail
Login Functionality		Verify Login functionality with inval sername nd invalid password.		Navigate to Login Page.			Able to see the login Page.	As expected.	Pass
				Enter use name	Invalid Username	ritesh@gmail.com	Credentials can be tered.	As expected.	Pass
				Enter Password.	Invalid Username	XXXXX@4	Credentials can be entered.	As expected.	Pass
				Click on Login Button.			User Logged	Unsuccesfully Logged	Fail
Login Functionality		Verify Login functionality with blan sername nd invalid password.		Navigate to Login Page.			Able to see the login Page.	As expected.	Pass
				Enter use name	Blank		Credentials can be tered.	As expected.	Pass
				Enter Password.	Invalid Username	XXXXX@5	Credentials can be entered.	As expected.	Pass
				Click on Login Button.			User Logged	Unsuccesfully Logged	Fail

7.4.2 Test Case 2

Project Name	Mental Health Chatbot								
Module Name	Chatting								
Created by:	Riya								
Created Date:	14-03-2023								
Executed by:	Riya								
Executed Date:	14-03-2023								
Test Case Id		Test Case Description	Pre Step	Test Step	Preconditions	Test Data	Expected Result	ActualResult	Status
Chatting Functionality		Verify chatting functionality with wrong spelling.		Navigate to chatting page.				As expected.	Pass
				Enter the chat.	Wrong Spelling.	Helo	Chat should be entered.	As expected.	Pass
				Click on send button.			Response should be generated.	No response.	Fail
Chatting Functionality		Verify chatting functionality with number as input.		Navigate to chatting page.				As expected.	Pass
				Enter the chat.	Numbers as input.	1234	Chat should be entered.	As expected.	Pass
				Click on send button.			Response should be generated.	No response.	Fail
Chatting Functionality		Verify chatting functionality with no wrong spelling and numbers.		Navigate to chatting page.					Pass
				Enter the chat.	Chat as input.	Hello	Chat should be entered.	As expected.	Pass
				Click on send button.			Response should be generated.	As expected.	Pass

CHAPTER 8: FUTURE SCOPE

The future potential of a mental health conversation bot is rather promising. As technology advances, more people are embracing digital tools to manage their mental health. A mental health chat bot can provide a number of benefits, including:

- **Accessibility:** Mental health chat bots are available at any time and from any location. This is especially helpful for people who live in remote areas who have difficulty receiving typical mental health care.
- **Anonymity:** Users of mental health chat bots can remain anonymous. This can be beneficial for persons who are ashamed or stigmatized about obtaining mental health therapy.
- **Personalization:** Mental health chat bots can cater to the needs of individual users. Using natural language processing, they can comprehend a user's symptoms and provide customized recommendations and help.
- **24 hours a day, seven days a week**: Mental health chat bots may provide help 24 hours a day, seven days a week, which is extremely beneficial for people who are facing mental health emergencies outside of normal business hours.
- **Cost-effective:** Compared to traditional mental health services, which can be expensive and are frequently not covered by insurance, mental health chat bots can be a less expensive option.

Mental health chat bots are projected to grow increasingly complicated and useful as the technology powering those progresses. Chat bots, for example, may be able to detect changes in a user's mood or conduct over time and intervene or provide support early. They can also be used in conjunction with other digital tools, such as wearable devices or virtual reality platforms, to provide more customized and effective help.

While there are numerous potential benefits to using a mental health chat bot, there are some drawbacks to consider:

- **Limited empathy:** Chat bots can provide nonjudgmental support and simulate empathy, but they cannot provide the same level of empathy and understanding as a real therapist or counsellor. This may limit chat bots' effectiveness as a mental health support tool for some people.
- **Lack of personalization:** While chat bots can be tailored, they may not be able to provide the same level of individualized assistance as a human therapist or counsellor. They may be unable to appreciate the full depth of a user's mental health difficulties or to provide individualized support based on specific needs.

Overall, mental health chat bots can be a useful tool for boosting mental health, but they have limits. Before implementing mental health chat bots as a primary support tool for mental health difficulties, it is critical to thoroughly analyze the potential benefits and cons. Furthermore, if

you are experiencing severe or complex mental health problems, you should seek professional help.

CHAPTER 9: REFRENCES

1. Pressman, R. S., & Maxim, B. R. (2015). Software Engineering: A Practitioner's Approach.

2. Aggarwal, K. K., & Singh, Y. (2007). Software Engineering. 3rd edition. New Age

3. Mental health: a state of well-being, World Health Organization, 2014.

4. T. Chey, J.W. Jackson, V. Patel, D. Silove, Z. Steel, C. Marnane, C. Iranpour Athorough review and metal analysis of the global prevalence of prevalent mental diseases Int J Epidemiol, 43(4), 476-493, 1980-

5. The global burden of disease: an update for 2004, World Health Organization,Geneva, 2008.

6. The global burden of mental, neurological, and drug use disorders: an analysis from the Global Burden.

7. H.A. Whiteford, A.J. Ferrari, L. Degenhardt, V. Feigin, and T. Vos, Plos one, 10

8. V. Patel, S. Saxena, N. Radcliffe, S. Ali Al-Marri, A. Darzi, S.P. Jones Health Affairs,33 (2014) 1603-1611,

9. "Study on Worker Behaviour in the Construction Industry to Improve Production

Efficiency," International Journal of Civil, Structural, Environmental, and Infrastructure

Engineering Research and Development (IJCSEIERD), Vol. 3, Issue 1, Mar 2013,

59-66

10. www.irjmets.com

11. www.freeproject.com

12. www.irjmets.com

13. Emanuela Haller and Traian Rebedea, "Designing a Chatbot that Simulates an

Historical Figure", IEEE Conference Publications, July 2013.

14. Maja Pantic, Reinier Zwitserloot, and Robbert Jan Grootjans, "Teaching Introductory

Artificial Intelligence Using Asimple Agent Framework", IEEE Transactions on

Education, Vol. 48, No. 3, August 2005.

15. www.iotforall.com

16. https://formative.jmir.org/2020/11/e17065/

17.https://www.researchgate.net/publication/336054143_An_overview_of_the_features_of_chatbots_in_mental_health_A_scoping_review

18. https://iopscience.iop.org/article/10.1088/1742-6596/2161/1/012039/pdf

19. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3768304

20.https://www.academia.edu/79840155/Sentiment_Analysis_using_Chatbot_and_Mental_Health_Tracker

21. Ian Sommerville. *Software Engineering* (Seventh Edition). Addison-Wesley, 2004.

22.Hans van Vliet. *Software Engineering: Principles and Practice* (Second Edition). Wiley, 1999.

23.David Budgen. *Software Design*. Pearson Addison Wesley; 2nd edition (May 15, 2003). 1st Edition: 1994.

24.Steve McConnell. *Software Project Survival Guide*. Microsoft Press; (November 1997).

25.Fundamentals of Database system by Elmensari, Navathe Sixth Edition.