Faculty of Informatics Engineering
Department of Software Engineering



Online Psychological Consulting system

A senior 2 project report - submitted to complete the requirements

For obtaining a Bachelor's Degree in informatics engineering

Prepared by

Sdra Khange Hanan Rmman

Supervised by

Eng. Tharaa Aslan
Eng. Aya Alaswad

2024-2025

SUPERVISION CERTIFICATION

أشهد بأن إعداد هذا المشروع الموسوم

Online Psychological

Consulting system

والمعد من قبل الطلاب:

حنان عبد اللطيف رمان _سدرة سمير خانجي

تحت إشرافي في قسم هندسة البرمجيات ونظم المعلومات في كلية

هندسة الحاسوب والمعلوماتية والاتصالات وهو جزء من متطلبات نيل

الإجازة في هندسة البرمجيات ونظم المعلومات

المرتبة العلمية: .....

الاسم: .....

التوقيع: .....

التاريخ: .....

ABSTRACT

In today's fast-paced digital world, access to mental health services is increasingly crucial. Traditional mental health care faces challenges such as limited availability, geographic barriers, and patient reluctance due to stigma. To address these challenges, this report presents an **Online Psychological Consulting System** designed to facilitate seamless communication between patients and mental health professionals through a secure and user-friendly digital platform.

The system offers key features including video and voice consultations, instant messaging, and AI-based psychological assessments to help diagnose mental health conditions. Additionally, it enables therapists to create private support groups for patients and share educational content to promote mental well-being. The system ensures data security through advanced authentication and encryption methods while maintaining high performance and scalability to serve a large number of users effectively.

By providing a comprehensive and accessible platform, this system aims to bridge the gap between patients and mental health professionals, offering convenient, confidential, and personalized care to improve mental health outcomes

ملخص

في عالمنا الرقمي السريع، أصبح الوصول إلى خدمات الصحة النفسية أمرًا بالغ الأهمية. تواجه الرعاية النفسية التقليدية العديد من التحديات، مثل محدودية التوافر، الحواجز الجغرافية، وتردد المرضى بسبب الوصمة المرتبطة بالصحة النفسية. لمعالجة هذه التحديات، يقدم هذا التقرير **نظام العيادة النفسية الإلكترونية**، المصمم لتسهيل التواصل بين المرضى والمتخصصين في الصحة النفسية من خلال منصة رقمية آمنة وسهلة الاستخدام.

يوفر النظام ميزات أساسية تشمل الاستشارات عبر الفيديو والصوت، المراسلة الفورية، واختبارات نفسية مدعومة بالذكاء الاصطناعي للمساعدة في تشخيص الحالات النفسية. كما يُتيح للأطباء إنشاء مجموعات دعم خاصة بالمرضى ومشاركة محتوى تعليمي لتعزيز الوعي بالصحة النفسية. يضمن النظام حماية بيانات المستخدمين من خلال طرق مصادقة وتشفير متقدمة، مع الحفاظ على الأداء العالي والقدرة على استيعاب عدد كبير من المستخدمين بفعالية.

يهدف هذا النظام إلى سد الفجوة بين المرضى وأخصائي الصحة النفسية، من خلال توفير رعاية نفسية مريحة، سرية، وشخصية لتحسين النتائج الصحية النفسية.

TABEL OF CONTENT

Chapter1 Introduction *11*

1. Introduction: 12

2. Problem Definition: 12

3. Project objectives: 12

4. Report organization: 13

Chapter 2 ..... *14*

Basic Concepts and Literature Review *14*

1. Introduction: 15

2. Basic concepts: 15

3. Literature review: 16

Chapter3 Project Management *19*

1. Introduction: 20

2. Project charter: 20

3. The SOW document: 22

4. Project plan – Gantt chart: 24

5. Risk Management: 25

6. Summary: 26

Chapter4 System Analysis *27*

1. Introduction: 28

2. Requirements Elicitation: 28

3. SRS Document: 33

    1. Introduction: 33

        1. Purpose: 33

        2. Project scope: 33

        3. Document overview: 33

    2. Overall description and product features: 34

        1. Product feature: 34

2. User classes and characteristics: 36
3. Nonfunctional requirements: 37
4. Conclusion: 38
3. Requirements modeling: 38
Detailed use case diagram (Doctor): 39
Detailed use case diagram (Admin): 40
Detailed use case diagram (Patient): 41
System features – use cases: 42
- Sign Up (UC-01): 42
Use case specification: 42
Sequence diagram (Sign Up): 43
- Login (UC-02): 44
Use case specification: 44
Sequence diagram (Login): 45
- Manage Profile Settings (UC-03): 45
Detailed use case diagram: 45
Use case specification: 46
Sequence diagram (Change password): 47
Sequence diagram (Change photo): 47
- Manage Domain (UC-04): 48
Detailed use case diagram: 48
Use case specification: 48
Sequence diagram (Manage Domain): 50
Sequence diagram (Manage Domain- Add psychological test): 51
- Manage Doctor Info (UC-05): 51
Detailed use case diagram: 51
Use case specification: 52
Sequence diagram (Manage Doctor Info- Deactivate doctor account): 54
- View Dashboard (UC-06): 54
Use case specification: 55

- Sign up (As A Doctor):..... 55

Detailed use case diagram: 55

Use case specification: 56

- Manage account settings:..... 57

Detailed use case diagram: 57

Use case specification: 57

- Manage account settings (Set communication way):..... 58

Use case specification: 58

Sequence diagram (Set communication way):..... 59

- Manage Support Group:..... 59

Detailed use case diagram: 59

Use case specification: 60

Sequence diagram (Create Group):..... 61

- Manage support group (View created group): 61

Use case specification: 61

Sequence diagram (Manage support group):..... 62

- View list of appointment: 63

Detailed use case diagram: 63

- Manage about patient:..... 63

Detailed use case diagram: 63

Use case specification: 64

- Manage Content: 64

Detailed use case diagram: 64

Use case specification: 65

Sequence diagram (Manage Content):..... 66

- Manage Profile Setting:..... 67

Detailed use case diagram: 67

- Appointment booking management: 68

Detailed use case diagram: 68

- Create instant session:..... 68

Use case specification: 68
Sequence diagram (Create instant session): 69
- Add book an appointment: 70
Use case specification: 70
Sequence diagram (Add book an appointment): 71
- Manage about doctor: 72
Detailed use case diagram: 72
Use case specification: 72
- Manage Session: 73
Detailed use case diagram: 73
Use case specification: 73
Sequence diagram (Manage Session): 74
- Share memories: 75
Use case specification: 75
Sequence diagram (Share memories): 75
- Answer psychological test: 76
Use case specification: 76
Sequence diagram (Answer psychological test): 76
4. Analysis class diagram: 77
5. Class Diagram: 78
4. Initial Test Cases: 79
(Test cases for general user) 79
- *Login* 79
- *Sign-Up* 79
- *Manage Profile Setting* 80
- *Manage Profile Setting (Change password)* 80
- *Manage Profile Setting (Change Photo)* 81
- *Manage Profile Setting (Change Personal info)* 82
(Test cases for Admin): 82
- *Manage Domain (Add domain)* 82

- *Manage Domain (Edit domain)* 83
- *Manage Domain (Delete domain)* 84
- *Manage Domain (Add Psychological Test).* 85
- *Manage Doctor Info* 85
- *View Dashboard* 86

(Test cases for doctor): 87

- *Manage account setting (Set price for session).* 87
- *Manage account setting (Set communication way)* 87
- *Manage Support Group (Create Group)* 88
- *Manage Support Group (View created Group)* 89

5. Initial RTM Requirements Trackability Matrix: 104

Chapter4 System Design *107*

1. Introduction: 108
2. System Architecture: 108

System decomposition - Component diagram: 110

Components functionalities: 110

System Architecture: 112

Updating the RTM: 114

Chapter5 Practical Implementation *118*

1. Introduction: 119

2. Used Tools: 119

3. System Interface: 121

- Sign-up interface (Admin) : 122
- Domain interface (Admin): 123
- *Manage Tests interface (Admin):* 124
- *Specialists Interface (Doctor):* 126
- *Sign-up interface (Doctor):* 126
- *Login interface (Doctor):* 127
- *Patient Sign-up interface (Patient):* 133
- *Login interface (Patient):* 133

4. *Testing in insomnia:*..... 139

- *Sign up doctor:* 139
- *Sign Up Patient:* 139
- *Password is weak:*..... 140
- *login patient:*..... 140

..... 140

- *patient Login to doctor login interface* 141
- *Enter invalid email address:* 141
- *Create a user account already exists:* 142

Chapter6 Conclusion *152*

1. *Introduction:*..... 153
2. *References:* 153
3. *Appendices:* 154

List of abbreviations

Abbreviation	Abbreviation
API	Application Program Interface
AI	Artificial Intelligence
UML	Unified Modeling Language
NoSQL	Not only Structured Query Language.
SQL	Structured Query Language.
SOW	Statement of Work

Chapter1 Introduction

1. Introduction:

In this chapter, we'll introduce the project by discussing the key issues and reasons behind building this system. We will also outline the objectives and goals we aim to achieve with the system. Finally, we will provide an overview the report organization

2. Problem Definition:

In today's world, mental health is increasingly recognized as a critical aspect of overall well-being. However, many individuals still face barriers to accessing mental health services, such as feelings of embarrassment, living in remote areas, and limited availability of services. Online mental health consultation platforms can bridge this gap by providing accessible, convenient, and confidential support to those in need.

The challenges in this field include ensuring the quality and credibility of consultations, maintaining user privacy and data security, and creating a supportive and user-friendly environment. Additionally, the dynamic nature of mental health needs necessitates continuous updates and improvements to the platform to ensure it remains effective and responsive to users' evolving requirements

3. Project objectives:

The primary objective of this project is to develop an online platform for mental health consultations that provides users with high-quality, accessible, and confidential support.

To achieve this, the system will incorporate features such as secure video conferencing, chat-based consultations, and an extensive resource library. Additionally, the system will ensure the highest standards of privacy and data

security to protect user information. The platform will also provide regular updates and improvements based on user feedback and emerging trends in mental health care to ensure its continued effectiveness and relevance.

4. Report organization:

The report will be organized in the following chapters:

- Chapter 1: introduction.
- Chapter 2: Basic concepts and literature review.
- Chapter 3: project management.
- Chapter 4: System analysis.
- Chapter 5: design study for the system
- Chapter 6: practical implementation.
- Chapter 7: Conclusion

Chapter 2

Basic Concepts and Literature Review

1. Introduction:

In this chapter, we will provide an overview of the main concepts and tools utilized in the development of the Online Psychological consulting system. This includes the technologies and frameworks required to implement features such as video and voice consultations, AI-driven psychological tests, doctor-patient messaging, and group interactions. Additionally, we will present a reference study of similar mental health platforms and tools, analyzing their strengths and limitations. This foundational knowledge is crucial for understanding the technical and theoretical basis of our approach and how it has been shaped by existing solutions in the mental health domain.

2. Basic concepts:

▪ *Maintenance in Software Engineering:*

Maintenance in software engineering is essential to ensure that software remains functional, efficient, and secure over time. It involves regularly updating and improving system requirements to fix bugs, enhance performance, and adapt to changing user needs and technological advancements. In our project, maintenance is crucial to ensure the continuous and efficient operation of the Online Mental Health Clinic System, adding new features such as updated psychological tests or enhancing the user experience in doctor-patient communication

▪ *Requirements Engineering:*

Requirements engineering is the process of defining, documenting, and maintaining software requirements. The elicitation phase, in particular, involves gathering requirements from stakeholders through methods like interviews, surveys, and workshops to ensure a comprehensive understanding

of the project's needs. For our system, requirements engineering focuses on identifying features such as messaging systems, video calls, and mechanisms for recommending specialized doctors to ensure the system meets user expectations and project goals

- ***Software User Feedback:***

User feedback is vital for understanding how the system performs in real-world scenarios. It provides direct insights into user satisfaction, usability issues, and desired features. In the Online Mental Health Clinic System, feedback from patients and doctors will be systematically collected and analyzed to improve the platform and introduce features that enhance the user experience, such as better interface designs or additional educational content.

- ***Data Visualization:***

Data visualization involves using tools and techniques to present data visually in dashboards and reports. In our project, data visualization will be used to transform psychological analysis results into easy-to-understand charts and reports, such as displaying percentages of different psychological conditions or evaluating doctors' effectiveness based on patient feedback.

3. Literature review:

In this section, we review similar tools and platforms that were studied and utilized to inform the development of our system. These platforms provide insights into the functionalities and limitations of existing mental health services, which helped guide the requirements elicitation process for our proposed system:

BetterHelp

BetterHelp is a widely recognized online therapy platform that provides users with access to licensed therapists through live chat, phone sessions, video sessions, and messaging. It focuses on individual therapy and offers specialized counseling for teens, couples therapy, and psychiatry. However, the platform does not provide medication management. All services are paid.

TalkSpace

TalkSpace offers comprehensive online therapy services through live chat, phone sessions, video sessions, and messaging. It supports individual therapy, couples therapy, teen counseling, and psychiatry. Like BetterHelp, TalkSpace does not provide medication management. Its pricing structure is subscription-based, with paid plans offering various tiers of access to therapists

Labayh

Labayh is a mental health platform targeting Arabic-speaking users. It offers phone sessions, video sessions, messaging, and individual therapy. However, it lacks live chat functionality and does not provide couples therapy, teen counseling, medication management, or psychiatry services. Its services are also offered as paid plans.

Arztkonsultation

Arztkonsultation is an online health consultation platform focusing

primarily on individual therapy and general medical consultations. It supports live chat, phone sessions, and video sessions but does not offer messaging, couples therapy, teen counseling, medication management, or psychiatry services. Like other platforms, its services are offered on a paid basis.

Our System

Our system is designed to provide a more holistic approach to mental health services, including live chat, phone sessions, video sessions, and messaging. In addition to individual therapy, it offers teen counseling, couples therapy, and psychiatry. Furthermore, it allows for unique functionalities such as medication management and group creation for patient engagement by doctors, distinguishing it from the other platforms studied. All services will follow a paid structure.

The Summary of the reference study:

Table 1summary of reference study

	BetterHelp	TalkSpace	labayh	arztkonsultation	Our system
Live chat	✓	✓	✗	✗	✓
Phone session	✓	✓	✓	✓	✓
video session	✓	✓	✓	✓	✓
Messaging	✓	✓	✓	✗	✓
Individual Therapy	✓	✓	✓	✗	✓
Couples Therapy	✓	✓	✓	✗	✓
Teen counseling	✓	✓	✓	✗	✓
Medication management	✗	✓	✗	✓	✓
Psychiatry	✗	✓	✓	✓	✓
price	paid	paid	paid	paid	paid

Chapter3 Project Management

1. Introduction:

In this chapter, we will explore the project management phase, a fundamental part of ensuring the project's success. We will explore the project charter, project plan, Statement of Work (SOW) document, and risk management strategies to effectively steer and control the project from initiation to completion.

2. Project charter:

The project charter is a formal document that grants the official approval to initiate a project. It serves as a reference point throughout the project's duration, providing a clear understanding of the project's objectives and establishing a foundation for decision-making and project management

Project title	*Online Psychological consulting system*
Project start date	Tue 4/27/25
Project finish date	Thu 6/19/25
Project manage	Eng. Tharaa Aslan

Project objectives

1. Develop a user-friendly website where patients can easily access psychologists.
2. Create a platform to provide continuous psychological support through articles, videos
3. Offer online therapeutic sessions in multiple formats such as video, text chat, and voice calls.
4. Increase awareness of mental health by publishing educational and awareness content.

Approach

1. Requirements Analysis: collect requirements from patients and psychologists to understand their needs.
2. Design and Development: Design and develop the website with a user-friendly interface.
3. System Testing: Conduct thorough testing to ensure the system is error-free and ready for use.
4. Launch and Promotion: Launch the website and promote it through social media platforms and media outlets.
5. Evaluation and Improvement: Collect feedback from users and continuously work on improving the system.

Roles and responsibilities

NAME	ROLE	RESPONSIBILITY
ENG. THARAA ASLAN	Project manager – supervisor	Project management and work monitoring
SDRA KHANGE & HANAN RMMAN	Engineer	-Backend development -Frontend development -documentation -analysis and design and implementation and testing phase team leader.

3. The SOW document:

The Statement of Work (SOW) is a comprehensive document that defines the scope of work for the "Online Psychological consulting system project". It outlines the specific tasks, deliverables, timeline, and responsibilities associated with the development of the system. This SOW provides a clear understanding of what needs to be accomplished, including the project's objectives of delivering efficient and secure mental health services. It also highlights the criteria for success, such as implementing AI-based psychological tests, enabling effective communication between patients and doctors, creating patient groups for shared discussions, and offering educational resources to support mental well-being.

Project Title: Online Psychological consulting system project

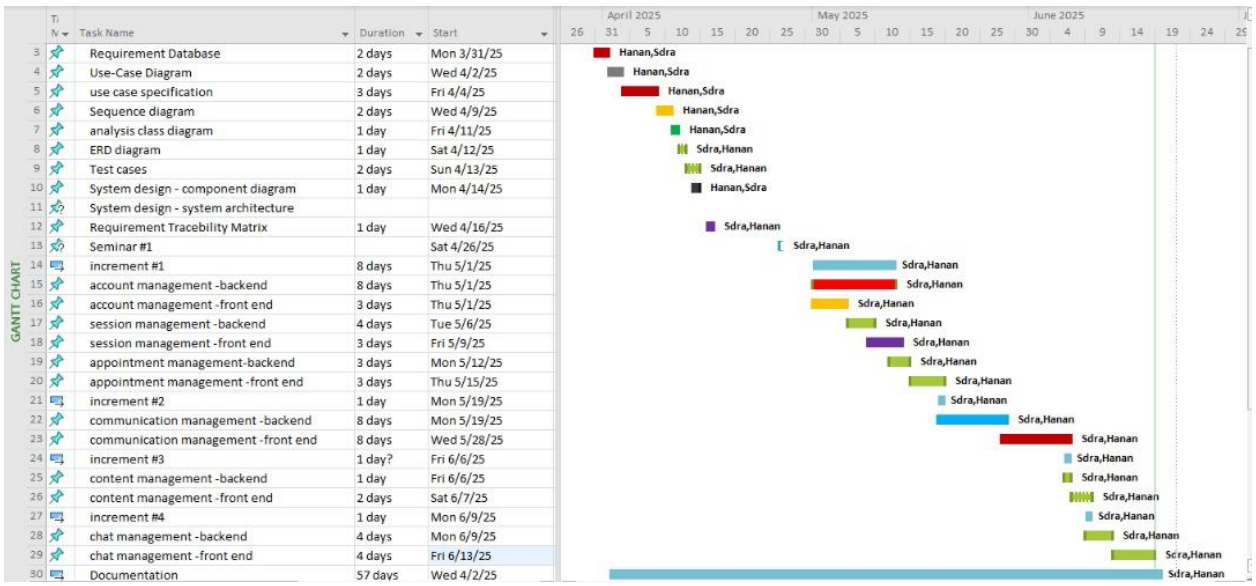
Project Description and Objectives:	*This project aims to develop a digital mental health clinic system that provides patients with comprehensive mental health services. The system will include video and voice call consultations, text messaging with doctors, psychological tests to diagnose patients' issues, and AI-based recommendations for specialized doctors. Additionally, doctors can create private groups for their subscribed patients to facilitate group discussions, as well as upload awareness videos for patients to view.*
Project Scope:	*The project involves building a digital system using AI and advanced technologies to provide a complete mental health service platform. The key features include:* *Direct Communication with Doctors: Video calls, voice calls, and text messaging.* *Integrated Psychological Test: Helps diagnose patients' mental health issues through AI-driven analysis of their responses.* *Doctor Recommendations: Based on the test results, the system suggests the most suitable specialist doctors.*

	Patient Groups: Doctors can create private groups for their subscribed patients to discuss and share insights. ○ Awareness Video Library: Doctors can upload educational and awareness videos that patients can access
Project goals:	○ *Develop a digital platform to provide secure and private mental health services to patients.* ○ *Incorporate AI-driven tools for analyzing psychological tests and recommending appropriate doctors.* ○ *Enable doctors to create patient groups for community support and discussion.* ○ *Provide a platform for uploading and accessing mental health awareness content*
Project Deliverables:	○ *Project Plan.* ○ *Software Requirements Specification (SRS).* ○ *Final Project Report.* ○ *AI Algorithms for analyzing psychological tests and recommending specialists.* ○ *Complete System (Frontend and Backend):* ○ *Patient interface.* ○ *Doctor dashboard for managing groups, uploading videos, and scheduling consultations.*
Technology and Tools:	*Programming Languages: Python, JavaScript, HTML, and CSS.* ○ *Frameworks:* ○ *Django (Backend).* ○ *React (Frontend).* ○ *Database: MySQL, MongoDB.* ○ *Additional Technologies:* ○ *AI for psychological test analysis.*

	○ *WebRTC for video and voice call services*
Project Resources	*Human Resources:* ○ *Eng. Tharaa Aslan- Project Manager.* ○ *Hanan Rmman - Software Developer.* ○ *Sdra Khange - Software Developer.*
Schedules:	*Project Start Date: September 17, 2024.* *First Project Seminar: October 13, 2024.* *Second Project Seminar: December 1, 2024.* *Project Completion Date: March 1, 2025*

4. Project plan – Gantt chart:

A document that outlines tasks, deadlines, and resources needed to achieve project objectives. This plan serves as a roadmap for successful project execution, making it easier to track progress and ensure all parties adhere to the set schedule



	Task Name	Duration	Start
3	Requirement Database	2 days	Mon 3/31/25
4	Use-Case Diagram	2 days	Wed 4/2/25
5	use case specification	3 days	Fri 4/4/25
6	Sequence diagram	2 days	Wed 4/9/25
7	analysis class diagram	1 day	Fri 4/11/25
8	ERD diagram	1 day	Sat 4/12/25
9	Test cases	2 days	Sun 4/13/25
10	System design - component diagram	1 day	Mon 4/14/25
11	System design - system architecture		
12	Requirement Tracebility Matrix	1 day	Wed 4/16/25
13	Seminar #1		Sat 4/26/25
14	increment #1	8 days	Thu 5/1/25
15	account management -backend	8 days	Thu 5/1/25
16	account management -front end	3 days	Thu 5/1/25
17	session management -backend	4 days	Tue 5/6/25
18	session management -front end	3 days	Fri 5/9/25
19	appointment management-backend	3 days	Mon 5/12/25
20	appointment management -front end	3 days	Thu 5/15/25
21	increment #2	1 day	Mon 5/19/25
22	communication management -backend	8 days	Mon 5/19/25
23	communication management -front end	8 days	Wed 5/28/25
24	increment #3	1 day?	Fri 6/6/25
25	content management -backend	1 day	Fri 6/6/25
26	content management -front end	2 days	Sat 6/7/25
27	increment #4	1 day	Mon 6/9/25
28	chat management -backend	4 days	Mon 6/9/25
29	chat management -front end	4 days	Fri 6/13/25
30	Documentation	57 days	Wed 4/2/25

5. Risk Management:

Is the process of identifying, analyzing, and addressing risks that may impact the success of a project.

This process aims to minimize the impact of risks by implementing preventive and proactive measures to ensure project objectives are effectively achieved.

Risk_ID	RK-01	RK-01	RK-01	RK-01
Risk title	Learning New Technologies	Team of Two Students	Misunderstanding Project Scope	Server Load
Risk Description	The new team of two members needs to learn front-end development as they previously worked on the back-end only.	The team consists of only two students; if one stops for any reason, the entire project will be affected.	The project's vastness might lead to misunderstandings about its scope among team members.	Increased number of users may lead to server overload.
Raised Date	4/27/2025	4/27/2025	19/6/2025	19/6/2025
Tracking Frequency	Weekly	Weekly	Daily	Weekly
State	Active	Active	Active	Active
Impact	High	High	High	Medium
Mitigation Plan	Provide appropriate training and educational resources, and divide tasks to facilitate learning.	Assign tasks so that each member can assist the other when necessary.	Regular meetings and reviews to ensure clarity among team members.	Enhance server capacity and improve performance.

6. Summary:

Ultimately, effective project management is crucial for the successful development of software systems. It provides the structure needed to ensure projects are completed within scope, time, and budget. Strong project management leads to the delivery of high-quality software systems that meet project objectives.

Chapter4 System Analysis

1. Introduction:

In this chapter we will introduce the analytical study of the system using the needed UML diagrams for system requirements modeling

2. Requirements Elicitation:

During the process of requirements elicitation, we took meticulous steps to gain an in-depth understanding of the project. We conducted a thorough study and analysis of similar systems, while also analyzing and anticipating the needs of the application's users, ensuring that our solution aligns perfectly with user expectations and industry standards.

The resulted Requirement database:

REQ ID	Requirements title	category	priority
RE-FR-AM-01	the system should provide all user (admin, doctor, patient) the ability to create a new account	accounts management	high
RE-FR-AM-02	the system should provide all user the ability to log to their accounts using their credentials	accounts management	high
RE-FR-AM-03	the system should provide all user ability to logout their accounts	accounts management	high
RE-FR-AM-04	the system should provide all user the ability to change their account password	accounts management	high
RE-FR-AM-05	The system should provide admin the ability to accept adding a new doctor in the system	accounts management	high
RE-FR-AM-06	The system should provide admin the ability to reject adding a new doctor in the system	accounts management	high
RE-FR-AM-07	The system should be able to display the number of site users statistics	accounts management	low
RE-FR-AM-08	The system should be able to display the number of doctors on site	accounts management	low

RE-FR-AM-09	The system should provide admin the ability to deactivate a doctor account	accounts management	high
RE-FR-AM-10	the system should be able to display the number of activated accounts for admin	accounts management	low
RE-FR-AM-11	the system should be able to display all activated accounts with their information	accounts management	low
RE-FR-AM-12	the system should provide admin the ability to add domain(category) for video and text article	accounts management	high
RE-FR-AM-13	the system should provide admin the ability to delete an existence domain	accounts management	high
RE-FR-AM-14	the system should provide the admin the ability to edit domain	accounts management	high
RE-FR-AM-15	the system should provide admin the ability to add psychological test	accounts management	high
RE-FR-AM-16	The system should provide all doctors, life coach the ability to enter the information for doctors, life coach by uploading files of these information (files: "csv").	accounts management	high
RE-FR-AM-17	the system should provide all doctors, life coach the ability to display their account information	accounts management	medium
RE-FR-AM-18	The system should provide all doctors, life coach the ability to delete their accounts.	accounts management	high
RE-FR-AM-19	the system should provide all doctors, life coach the ability to change their profile photo by uploading local photo	accounts management	medium
RE-FR-AM-20	The system should provide all doctors, life coach the ability to add their username of the account.	accounts management	medium
RE-FR-AM-21	the system should provide all doctors, life coach the ability to add information	accounts management	high
RE-FR-AM-22	the system should provide all doctors, life coach the ability to change information	accounts management	high

RE-FR-SM-23	the system should provide all doctors, life coach the ability to add schedule	schedule management	high
RE-FR-SM-24	the system should provide all doctors, life coach the ability to change schedule	schedule management	high
RE-FR-SM-25	the system should provide all doctors, life coach the ability to delete schedule	schedule management	high
RE-FR-APM-26	the system should provide all doctors, life coach the ability to choose a communication method	appointment management	high
RE-FR-APM-27	the system should provide all doctors, life coach the ability to create new video call	appointment management	high
RE-FR-APM-28	the system should provide all doctors, life coach the ability to create new voice call	appointment management	high
RE-FR-APM-29	the system should provide all doctors, life coach the ability to create new message	appointment management	high
RE-FR-APM-30	the system should provide all doctors, life coach the ability to create new group support	appointment management	high
RE-FR-APM-31	the system should provide all doctors, life coach the ability to add information group support	appointment management	high
RE-FR-APM-32	the system should provide all doctors, life coach the ability to change information group support	appointment management	high
RE-FR-APM-33	the system should provide all doctors, life coach the ability to delete information group support	appointment management	high
RE-FR-APM-34	the system should provide all doctors, life coach the ability to add therapy programs	appointment management	high
RE-FR-APM-35	the system should provide all doctors, life coach the ability to send of medical prescription	appointment management	high
RE-FR-APM-36	the system should provide all doctors, life coach the ability to select price for each one	appointment management	high
RE-FR-APM-37	the system should provide all doctors, life coach the ability to add a new text article within the domain	appointment management	high

RE-FR-APM-38	the system should provide all doctors, life coach the ability to add a new video support within the domain	appointment management	high
RE-FR-APM-39	the system should provide all doctors, life coach the ability to search for patients	appointment management	high
RE-FR-APM-40	the system should provide all doctors, life coach the ability to browse the list of participating patients in the group support	appointment management	high
RE-FR-APM-41	the system should provide all doctors, life coach the ability to browse patient appointment schedule	appointment management	high
RE-FR-APM-42	the system should provide all doctors, life coach the ability to display status for doctor	appointment management	high
RE-FR-APM-43	the system should provide all patient the ability to add information	accounts management	high
RE-FR-APM-44	the system should provide all patient the ability to modification information	accounts management	high
RE-FR-APM-45	the system should provide all patient the ability to add book an appointment	appointment management	high
RE-FR-APM-46	the system should provide all patient the ability to change book an appointment	appointment management	high
RE-FR-APM-47	the system should provide all patient the ability to delete book an appointment	appointment management	high
RE-FR-APM-48	the system should provide all patient the ability to choose the doctor, life coach	appointment management	high
RE-FR-APM-49	the system should provide all patient the ability to change the doctor, life coach	appointment management	high
RE-FR-APM-50	the system should provide all patient the ability to browse the list of the doctor, life coach are exist on the system	appointment management	high
RE-FR-APM-51	the system should provide all patient the ability to browse the list of online doctors, life coach are exist	appointment management	high

RE-FR-APM-52	the system should provide all patient the ability to search for doctors, life coach are exist	appointment management	high
RE-FR-APM-53	the system should provide all patient the ability to share their memories	appointment management	high
RE-FR-APM-54	the system should provide all patient the ability to answer psychological test	appointment management	high
RE-FR-APM-55	the system should provide all patient the ability to do instant sessions	appointment management	high
RE-FR-APM-56	the system should provide all patient the ability to add rate the doctors, session	appointment management	high
RE-FR-APM-57	the system should provide all patient the ability to add comment for system and doctors	appointment management	high
RE-FR-APM-58	the system should provide all patient the ability to record the video call with their doctor, life coach	appointment management	high
RE-NF-59	The system should be able to give the result of an operation or event any user chooses in less than 5 minutes.	non-functional	high
RE-NF-60	The system should be able to protect its all-users' data successfully.	non-functional	high
RE-NF-61	the system should provide easy to use interfaces for the all users to control their process	non-functional	high
RE-NF-62	The system should be maintainable ensuring ease of modification, updating, and maintenance over time	non-functional	high
RE-NF-63	The system should be extensible allowing for easy addition of new features	non-functional	high

3. SRS Document:

1. Introduction:

1. Purpose:

The purpose of this Software Requirements Specification (SRS) is to outline the functional and non-functional requirements for our system. This document will serve as a foundational reference to ensure alignment between stakeholders and the development team throughout the project lifecycle.

2. Project scope:

Our system is designed to provide accessible, high-quality, and confidential mental health consultations. To achieve this, the platform includes key functionalities such as secure video conferencing, chat-based consultations, and an extensive resource library. The system aims to bridge the gap in mental health services by offering support to users in remote areas and ensuring user privacy and data security. Additionally, the platform will include regular updates and improvements based on user feedback to stay effective and relevant.

3. Document overview:

The software requirements specification (SRS) document will be structured as follows:

1. Introduction
2. Overall description and product features
3. Non-functional requirements
4. Conclusion

2. Overall description and product features:

1. Product feature:

The product features derived from the system's use cases are:

- Sign Up (UC-01): This feature allows all users (admin, doctor, life coach, patient) to create a new account on the system

- Login (UC-02): This feature provides all users the ability to log into their accounts using their credentials.

- Manage Profile Settings (UC-03): This feature allows users to update and manage their profile settings like password and profile photo updating, or other account information

- Manage Domain (UC-04): This feature allows admin to manage domains, including adding, deleting, and editing domains for video and text articles

- Manage Doctor Info (UC-05):
  This feature allows admin to manage doctor information, including accepting doctor, rejecting, deactivating accounts
- View Dashboard (UC-06): This feature enables the system admin to view a dashboard with system-wide metrics and information about the system usage

- Manage Schedule (UC-07): This feature allows doctors and life coaches to manage their schedules, including adding, changing, and deleting schedules

- Manage Support Group (UC-08): This feature allows doctors and life coaches to create and manage support groups, including adding, changing, and deleting group support information

- View List of Appointments (UC-09): This feature provides doctors and life coaches the ability to view their list of appointments

- Manage about Patient (UC-10): This feature allows doctors and life coaches to manage information about patients, including searching, viewing schedules, and browsing patients

- Manage Content (UC-11): This feature allows doctors and life coaches to manage content, including adding text articles and video support within the domain.

- Appointment Booking Management (UC-12): This feature allows patients to manage their appointments, including booking, changing, and deleting appointments.

- Manage about Doctors (UC-13): This feature provides patients the ability to manage information about doctors and life coaches, including searching and browsing lists.

- Share Memories (UC-14): This feature provides patients the ability to share their diaries and memories.

- Answer Psychological Test (UC-15): This feature provides patients the ability to answer psychological tests.

- Manage Session (UC-16): This feature provides patients the ability to manage their sessions, including participating in instant sessions, rating doctors and sessions, and adding comments.

2. User classes and characteristics:

Our system has the following main types of users:

- **Admin:** A high-privilege user responsible for overseeing the entire system's operations. The admin has full administrative privileges, including user management and system configuration. They are responsible for managing user accounts and monitoring system usage

- **Doctor/Life Coach:** These users leverage the system to provide support and consultations. They manage their schedules, patient information, and content. Additionally, they can create and manage support groups, and handle appointments

- **Patient:** The primary beneficiaries of the system's functionalities, patients use the system to manage their appointments, view and interact with doctors or life coaches, and access various resources. They can also participate in psychological tests and share their diaries.

- **General User**: Any individual who uses the system for accessing content, creating accounts, and managing their profile settings.

3. Nonfunctional requirements:

The nonfunctional requirements derived from the system's needs are:

- **RE-NF-01 (High Performance for Responding):** The system should be able to give the result of an operation or event any user chooses in less than 5 minutes. This ensures efficient and timely responses for all users.

- **RE-NF-02 (Security):** The system should be able to protect its users' data successfully. This includes implementing strong security measures, such as token-based authentication, to ensure the safety and privacy of user data.

- **RE-NF- 03(User-Friendly Interfaces):** The system should provide easy-to-use interfaces for all users to control their processes. Users should be able to navigate and understand all parts and pages of the system with ease.

- **RE-NF-04 (Maintainability):** The system should be maintainable, ensuring ease of modification, updating, and maintenance over time. This allows for continuous improvement and adaptation to new requirements.

- **RE-NF-05 (Extensibility):** The system should be extensible, allowing for easy addition of new features. This ensures the system can grow and adapt to new functionalities as needed.

4. Conclusion:

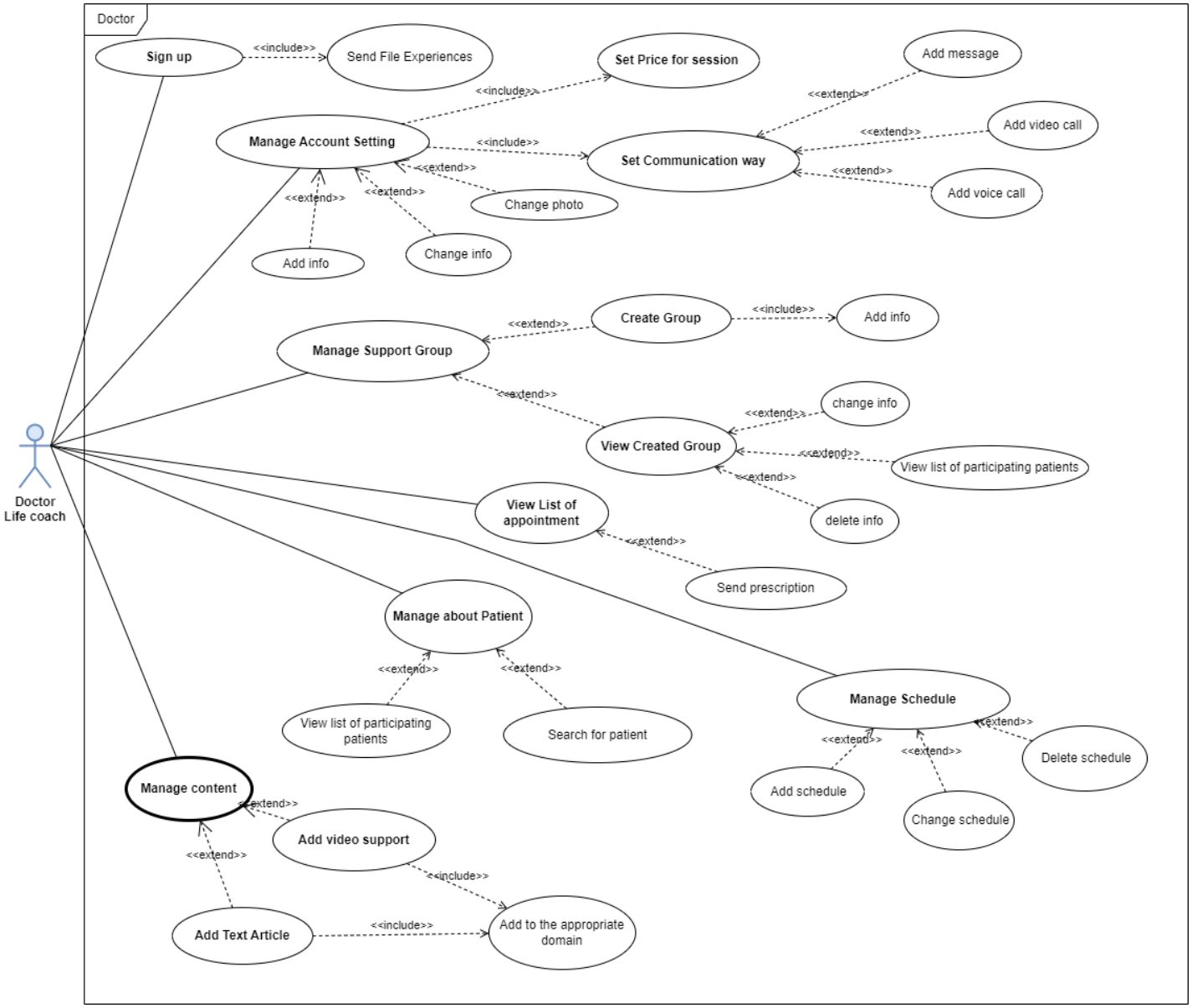
This Software Requirements Specification (SRS) document outlines the critical functionalities, user roles, and assumptions essential for developing a robust and user-friendly system. By adhering to these requirements, the system will effectively support users (admin, doctors, life coaches, patients) in enhancing their interaction with the platform to meet their needs and streamline various processes, from appointment management to content sharing and psychological support.

3. Requirements modeling:

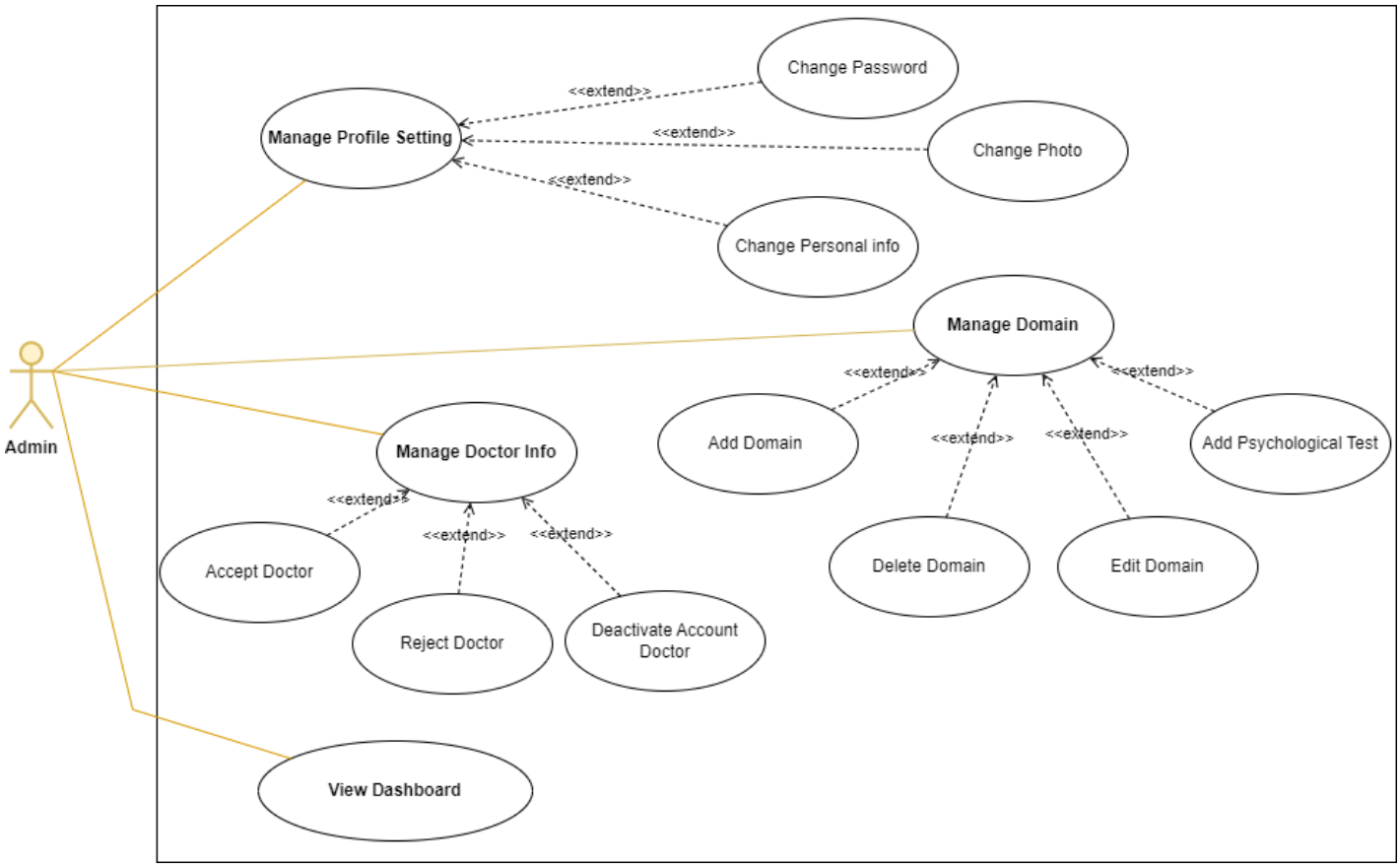
Use Case Diagram: Use-case diagrams model the behavior of a system and help to capture the requirements of the system. These diagrams provide a visual representation of the various interactions between users (admin, doctors, life coaches, patients) and the system, illustrating how different user roles interact with the system to achieve specific goals.



Detailed use case diagram (Doctor):



Detailed use case diagram (Admin):



Detailed use case diagram (Patient):



System features – use cases:

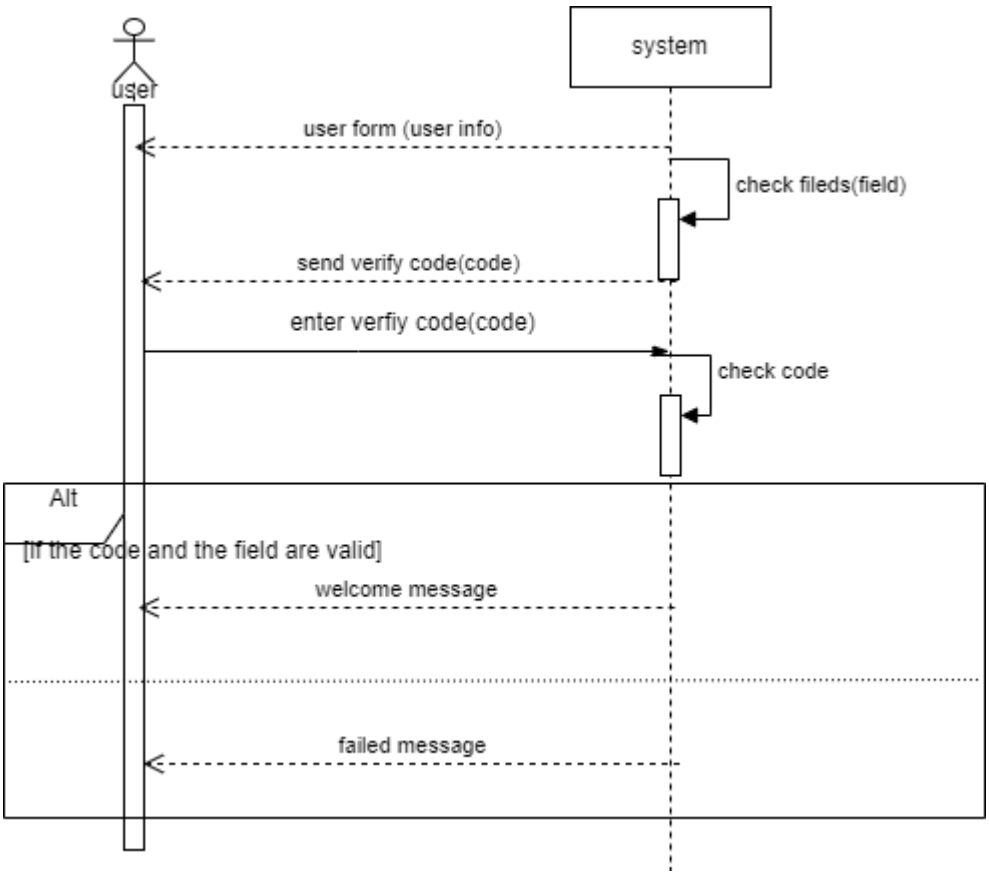
- ### Sign Up (UC-01):

  This feature allows all users (admin, doctor, and life coach, patient) to create a new account on the system

Use case specification:

Use Case name	Sign-Up
Participating Actors	Initiated by: Doctors, Life coach, Patient
The flow of events	1. The users enter the website on the sign page 2. The system shows a form 3. The user completes the form and chooses "create account" 4. The system checks if all fields are complicated 5. The system will make defined validations and send to the user a verify code on the entered email If the users enters the correct code the system send, the account will be added, and the system will open
Alternative flows	First alternative flows A1: start at step 4 in the main flow, there is a missing field: 5: The system will show an error message "there is a missing field" 6: The user will complete the fields and the flow will return to step 3 Third alternative flows A3: if the password the user chose is weak start at 5: 6: The system will tell the user that the password must be at least 8 characters 7: The user will reenter the password, and the flow will return to the step5. First exception flow E1: if the verify code is not correct the system will cancel the whole process and the use case will fail.
Entry condition	The user opens the system website.
Exit condition	The user has an account.

Sequence diagram (Sign Up):



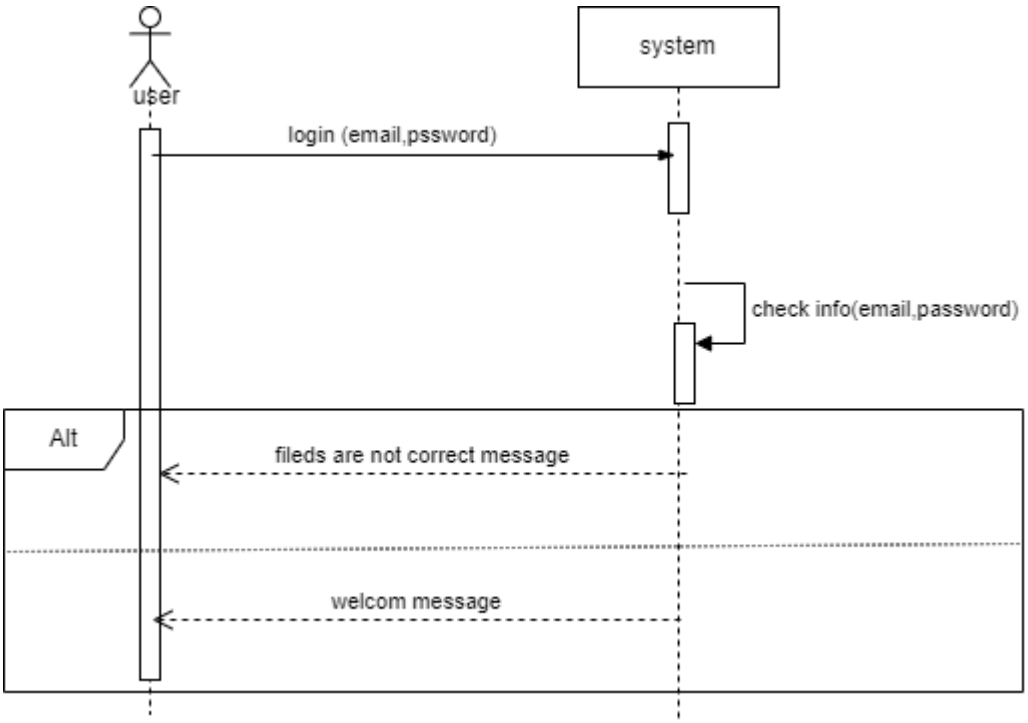
Login (UC-02) :

This feature provides all users the ability to log into their accounts using their credentials

Use case specification:

Use Case name	Log in
Participating Actors	Initiated by: admin
The flow of events	1. The user enters the website on the main page 2. The system shows a log-in form 3. The user completes the form by entering his information. 4. The system checks if all fields are complicated 5. The system will make defined validations 6. If the user belongs to the system, the system will be open on the main page for each user
Alternative flows	First alternative flows A1: start at step 4 in the main flow, there is a missing field: 5:The system will show an error message "there is a missing field" 6:the user will complete the fields and the flow will return to step 3 Second alternative flows A2: start at step 5 from the alternative flow: the email is not existing 7. The system will ask the user to reenter the email 8. The user will reenter the email filed, and the flow will return to step5.
Entry condition	The user has an account.
Exit condition	The user logged in.

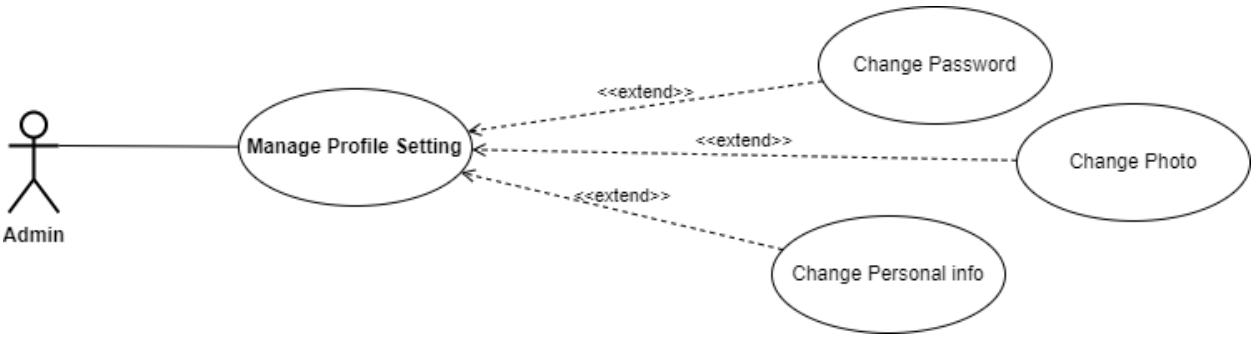
Sequence diagram (Login):



- ## Manage Profile Settings (UC-03):

This feature allows users to update and manage their profile settings like password and profile photo updating, or other account information

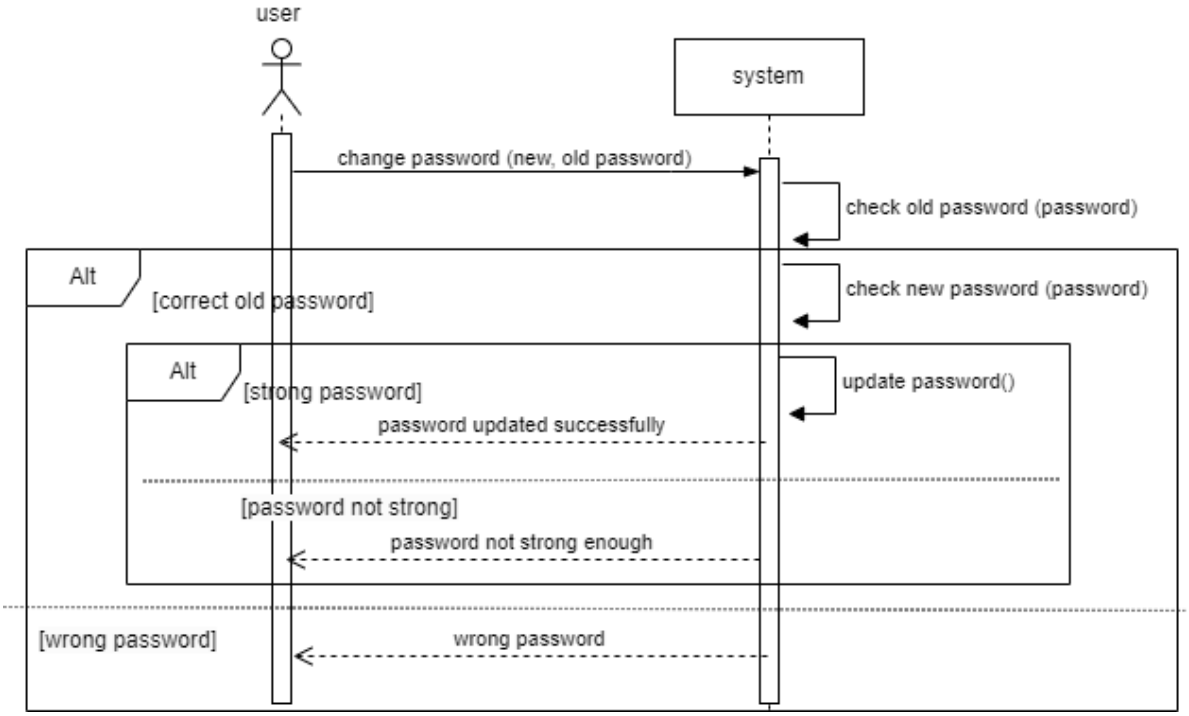
Detailed use case diagram:



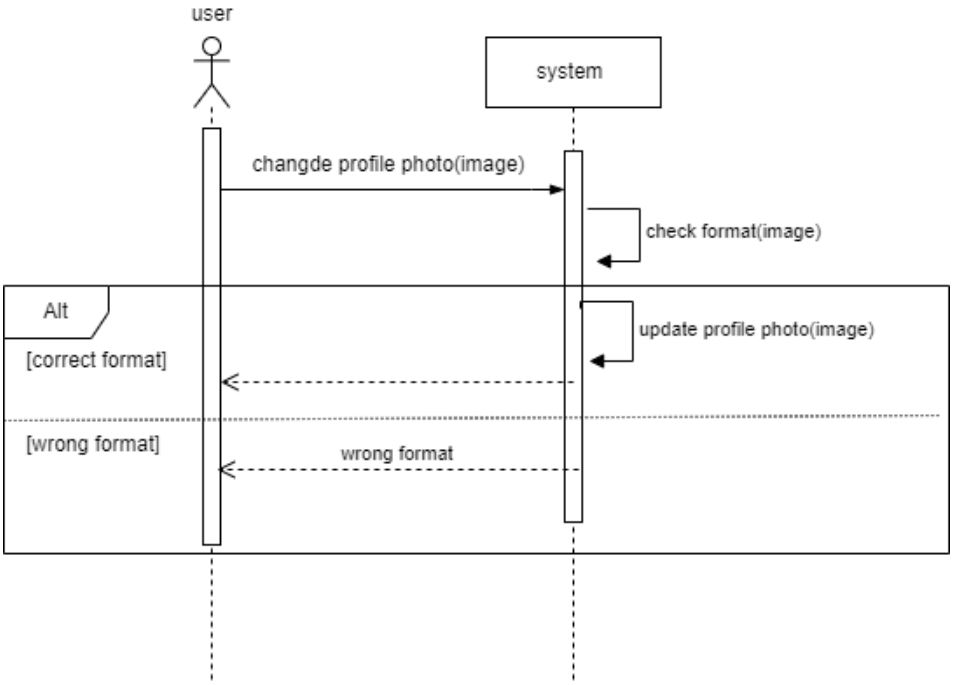
Use case specification:

Use Case name	Manage Profile Settings
Participating Actors	Initiated by: admin
The flow of events	1. The admin enter the "profile settings" page 2. If the user chooses to change their profile photo 3. The system will ask to upload the new photo 4. The user will upload the new photo 5. The system will check the format of the image 6. If the user ask to change their password 7. The system will ask them to add the old password and the new password of the user 8. The system will check if the old password matches the password of the user 9. The system check if the new password matches the user strength conditions 10. If the password is a strong enough, the system will save the new password. 11. If the admin choose change user name 12. The system will ask to write new user name 13. The admin write new user name 14. The system saves the new user name in database
Alternative flows	First alternative flows A1: start at step 7in the main flow, incorrect old password: 5:The system will show an error message "wrong old password" 6:the user will reenter the password and the flow will return to step 7 Second alternative flows A2: start at step 8 from the alternative flow: the password in not strong enough: 10: The system will ask the user to reenter the password. 11:the user will reenter the password filed, and the flow will return to step8
Entry condition	The user has an account.
Exit condition	The user changes his profile data.

Sequence diagram (Change password):



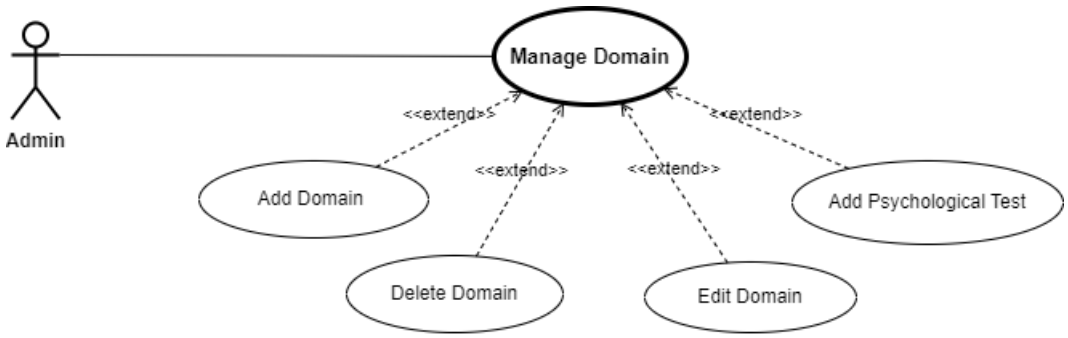
Sequence diagram (Change photo):



■ Manage Domain (UC-04):

This feature allows admin to manage domains, including adding, deleting, and editing domains for video and text articles

Detailed use case diagram:

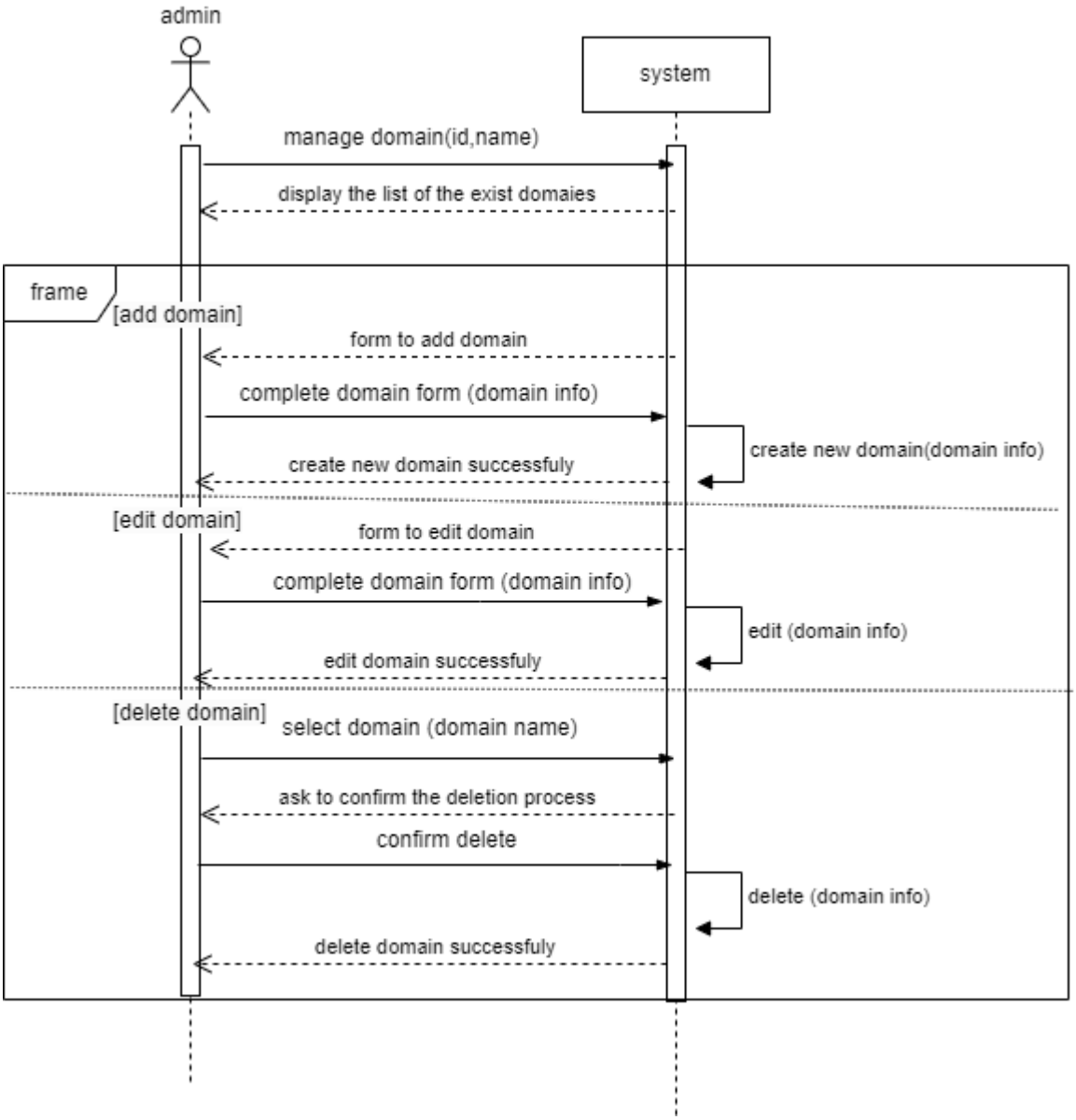


Use case specification:

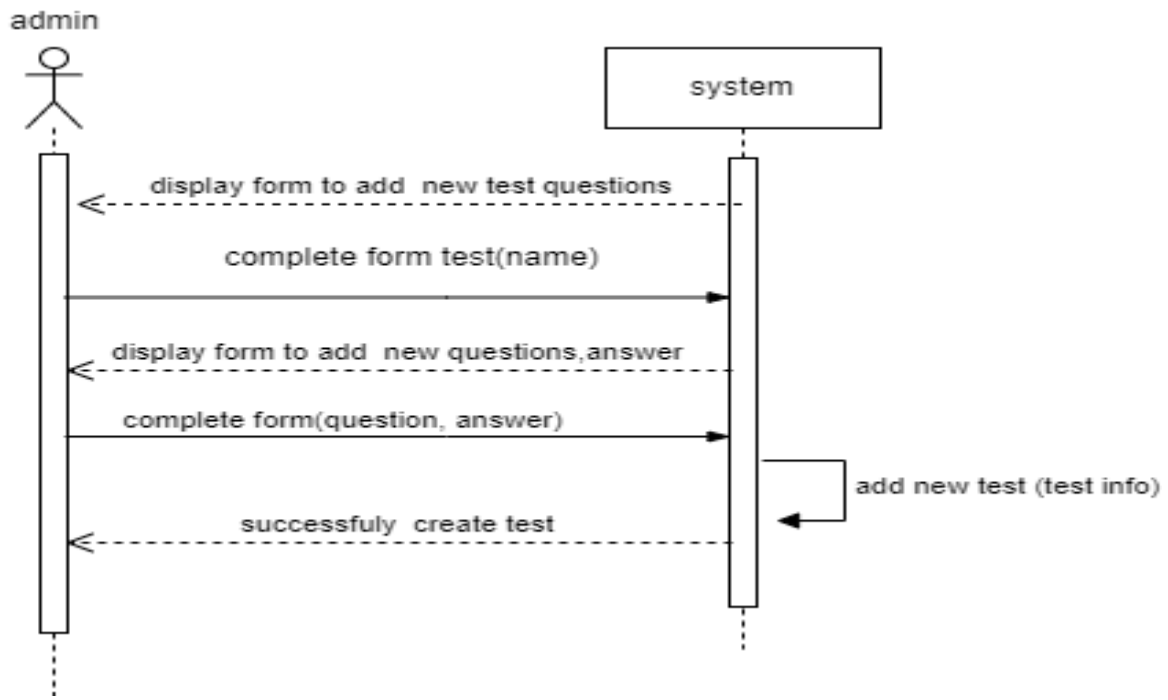
Use Case name	Manage domain
Participating Actors	Initiated by :admin
The flow of events	1. The admin enter "manage domain" page 2. If the admin choose add domain 3. The system display form to add domain and (name) 4. The admin enter the information about domain(name) 5. The system make defined validation 6. If the admin choose edit domain 7. The admin choose the domain that he want to edit 8. The system shows form to add edits 9. The admin write the edit 10. The system make defined validation 11. If the admin choose delete domain 12. The admin choose the domain that want to delete

	13. The system want to delete the domain from db 14. If the admin choose add psychological test 15. The system display fields to add question and the answer The system saves the test in db.
Alternative flows	First alternative flows A1: start at step 2in the main flow, The domain already exist: 3: The system will show an error message "The domain already exists" 4: the admin will reenter the new name of the domain fields and return to step 3 Second alternative flows A2: start at step 7in the main flow the domain is not exist in db 8: The system will show an error message "The domain not exists in system" 4: the admin will reenter the correct name of the domain and return to step 7
Entry condition	The admin logged in
Exit condition	Changes made to domain are saved in the system

Sequence diagram (Manage Domain):



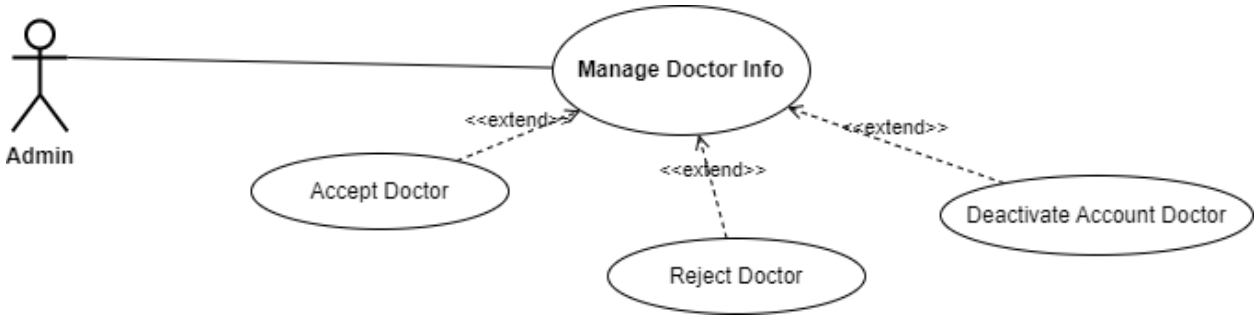
Sequence diagram (Manage Domain- Add psychological test):



- ## Manage Doctor Info (UC-05):

  This feature allows admin to manage doctor information, including accepting doctor, rejecting, deactivating accounts

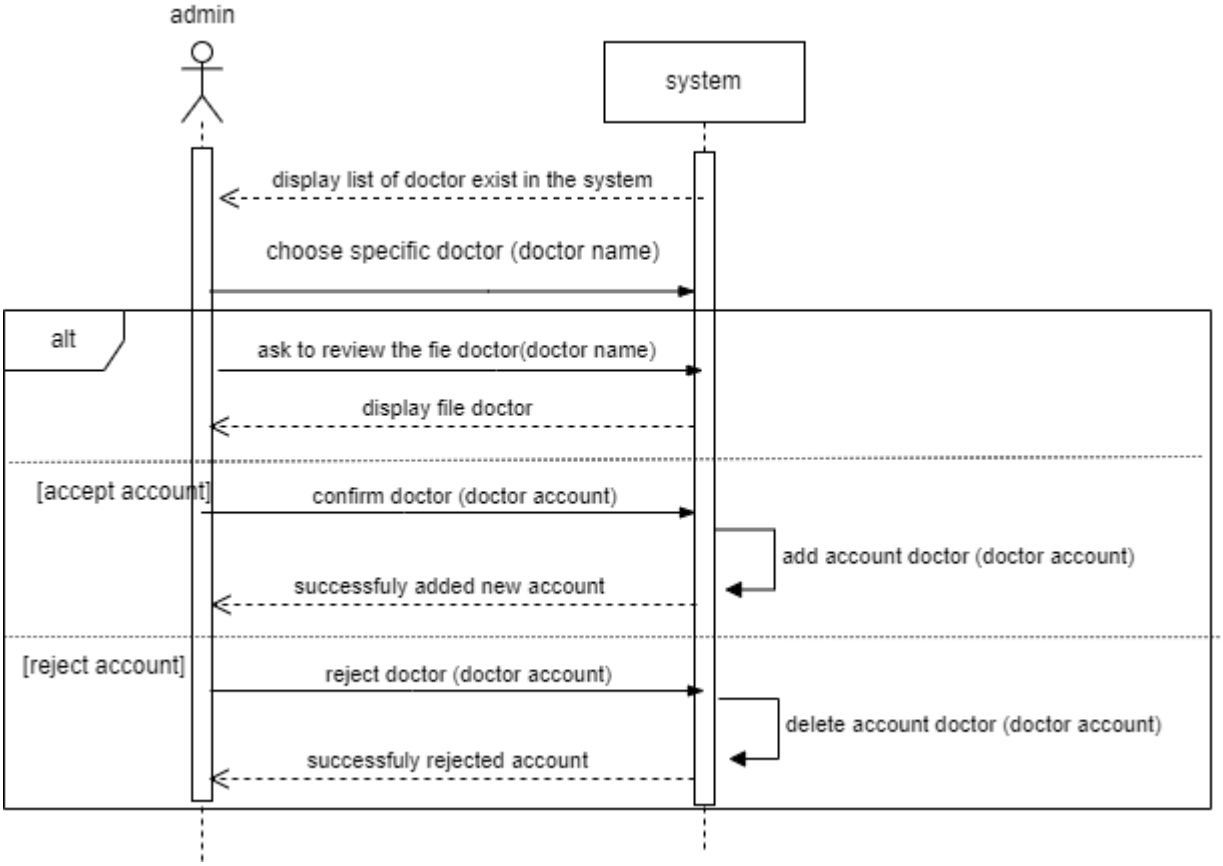
Detailed use case diagram:



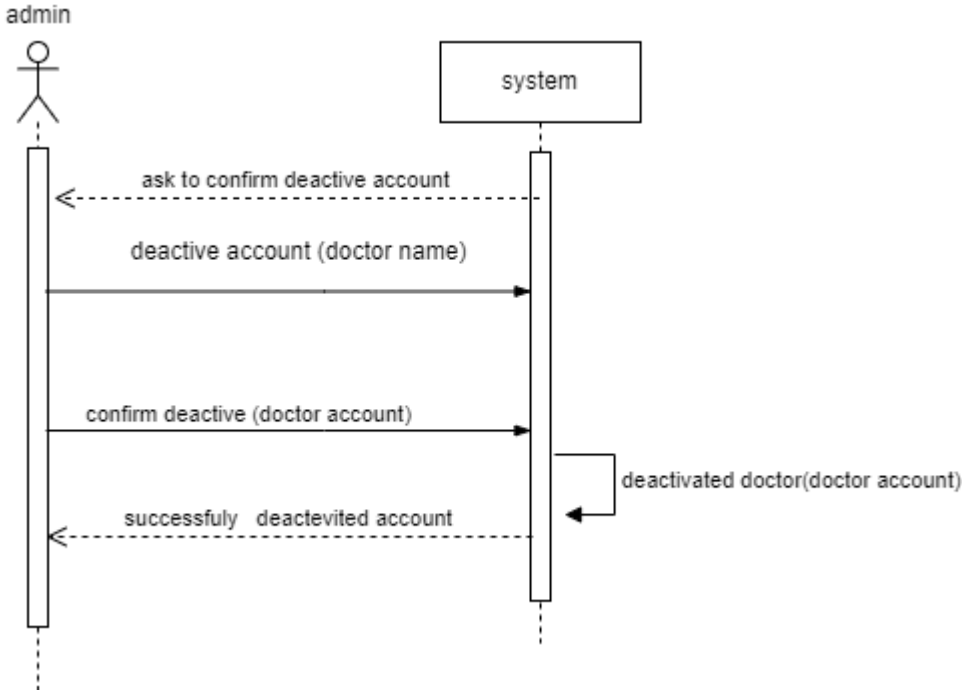
Use case specification:

Use Case name	Manage doctor information
Participating Actors	Initiated by: admin
The flow of events	1. The admin enter “Manage doctor information” page 2. The system displays page “view list doctor” 3. The admin choose the specific doctor 4. The system display doctor info and options (deactivate, accept, reject) 5. The admin the admin makes sure of the doctor's information sent the (experience file and certificate...) 6. If the admin choose accept doctor in the system 7. The system add doctor to the system 8. If the admin choose reject doctor 9. The system reject the doctor 10. If the admin choose deactivate account doctor 11. The system show message “confirm this process” 12. The system saves process in db
Entry condition	The admin logged in
Exit condition	changes made to info doctor saved to the system

Sequence diagram (Manage Doctor Info- Accept doctor):



Sequence diagram (Manage Doctor Info- Deactivate doctor account):



- ## View Dashboard (UC-06):

This feature enables the system admin to view a dashboard with system-wide metrics and information about the system usage

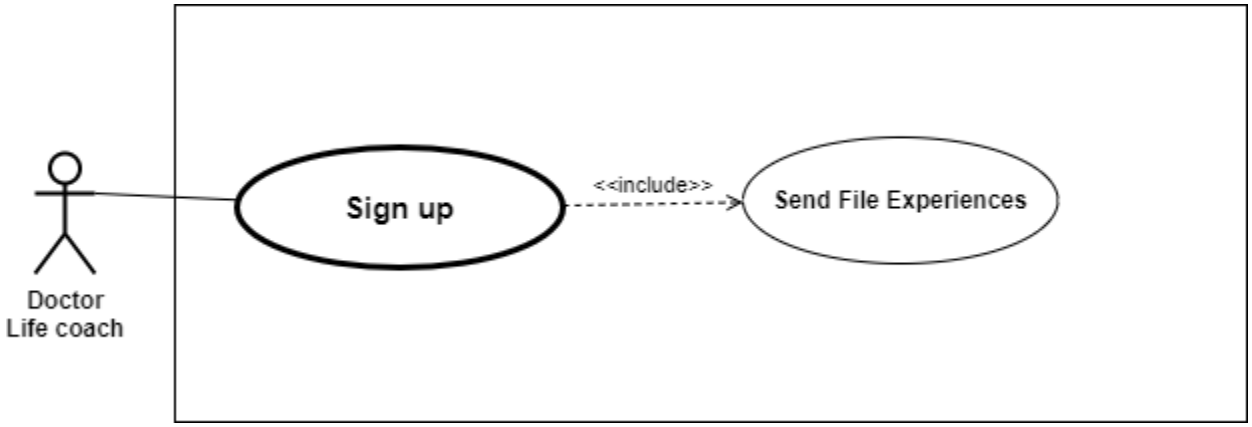
- The system should be able to display the number of site user's statistics
- The system should be able to display the number of doctors on site

Use case specification:

Use Case name	View dashboard
Participating Actors	Initiated by: Admin
The flow of events	1. Admin selects the "View Dashboard" option from the main menu 2. The system displays the Dashboard containing statistics of the number of doctors and the number of users 3. Admin can view the statistics and interact with the displayed data
Entry condition	The admin logged in
Exit condition	Display statistics of analysis result for the reviews

- ## Sign up (As A Doctor):

Detailed use case diagram:



Use case specification:

Use Case name	Sign-Up
Participating Actors	Initiated by : Doctors ,Life coach
The flow of events	1. The doctor enter the website on the sign page 2. The system shows a form 3. The doctor completes the form and chooses "create account" 4. The system asks the doctor to send the experience and certificate in "csv" 5. The doctor upload file experience... 6. The system checks if all fields are complicated 7. The system will make defined validations and send to the user a verify code on the entered email 8. If the users enter the correct code the system send, the account will be added, and the system will open
Alternative flows	First alternative flows A1: start at step 3in the main flow, the email of doctor there exist in the system: 5: The system will show an error message "the email is already exist" 6: The doctor rewrite the new email and the flow will return to step 3 Second alternative flows A3: if the file format is not required: 6: The system will show an error message" format of file not matches with csv" 7: The doctor will resend the file, and the flow will return to the step6.
Entry condition	The doctor opens the system website.
Exit condition	The user has an account.

- ## Manage account settings:

Detailed use case diagram:



Use case specification:

Use Case name	Manage account Settings
Participating Actors	Initiated by: doctor
The flow of events	1. The doctor enter the "manage account settings" page 2. The system shows page include "set price for one session, set communication way" 3. If the doctor choose set price for session 4. The system display a page the determine the price of doctor session along with a range of prices that should be chosen 5. The doctor enters the price for his session 6. The system saves the price for doctor
Alternative flows	First alternative flows A1: start at step 5in the main flow, price the session is high or less the determine the prices: 5: The system will show an error message "the price not within the range "

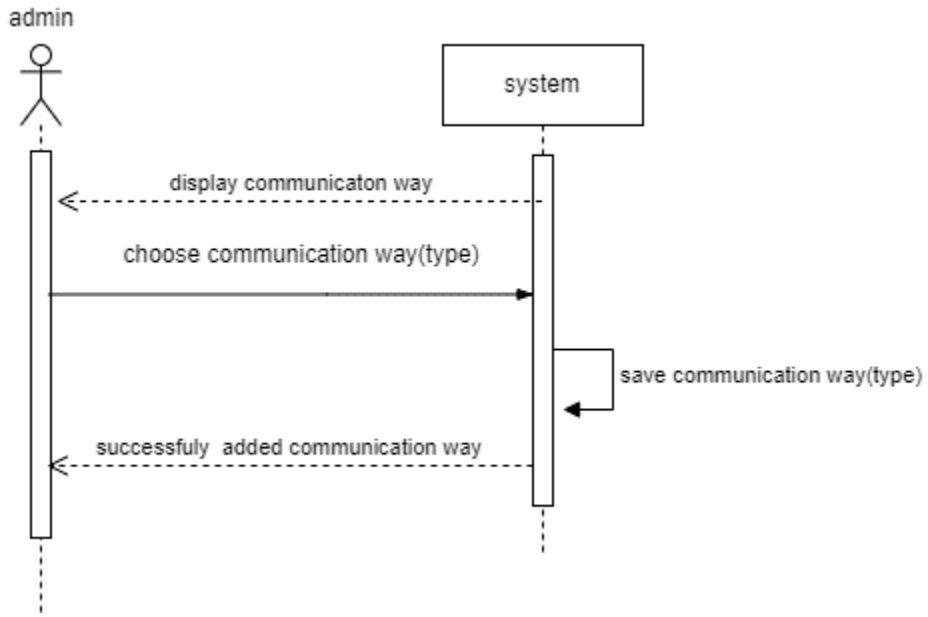
	6: The doctor will reenter the new price within the domain and the flow will return to step 5
Entry condition	The doctor logged in
Exit condition	The doctor changes his profile data.

- ## Manage account settings (Set communication way):

Use case specification:

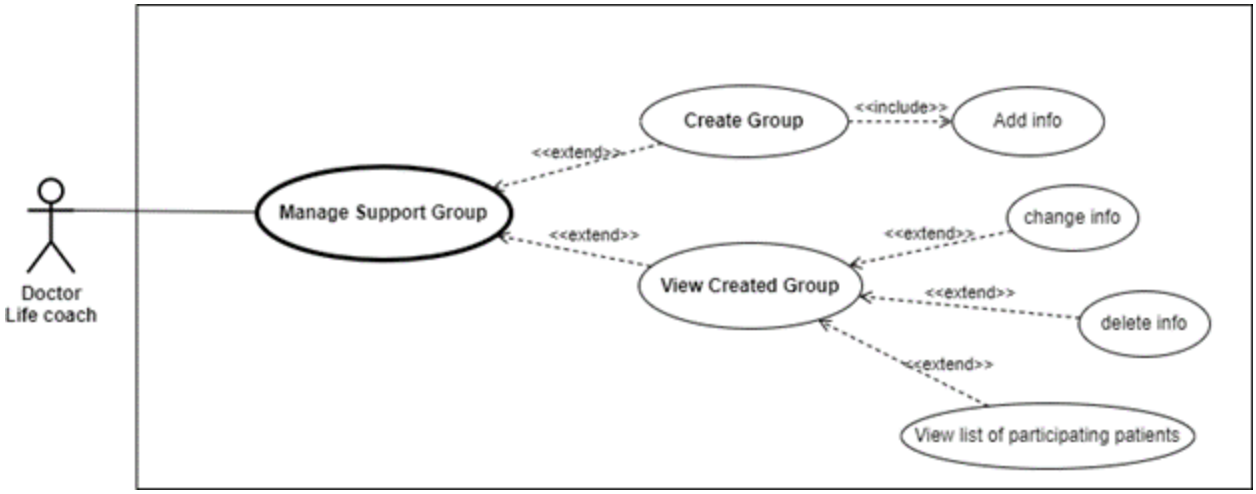
Use Case name	Set communication way
Participating Actors	Initiated by: doctor
The flow of events	1. If the doctor choose "Set communication way" page 2. The system shows page include "add message, add video call, add voice call" 3. The doctor choose one or all this above options 4. The system saves the communication way chosen by the doctor and includes them in his account info
Alternative flows	First alternative flows A1: start at step 3in the main flow, the doctor not set any one of the communication way: 4: The system will show an error message "you must set communication way" 6: The doctor will choose the communication way and the flow will return to step 4
Entry condition	The doctor logged in
Exit condition	The user changes his profile data.

Sequence diagram (Set communication way):



- ## Manage Support Group:

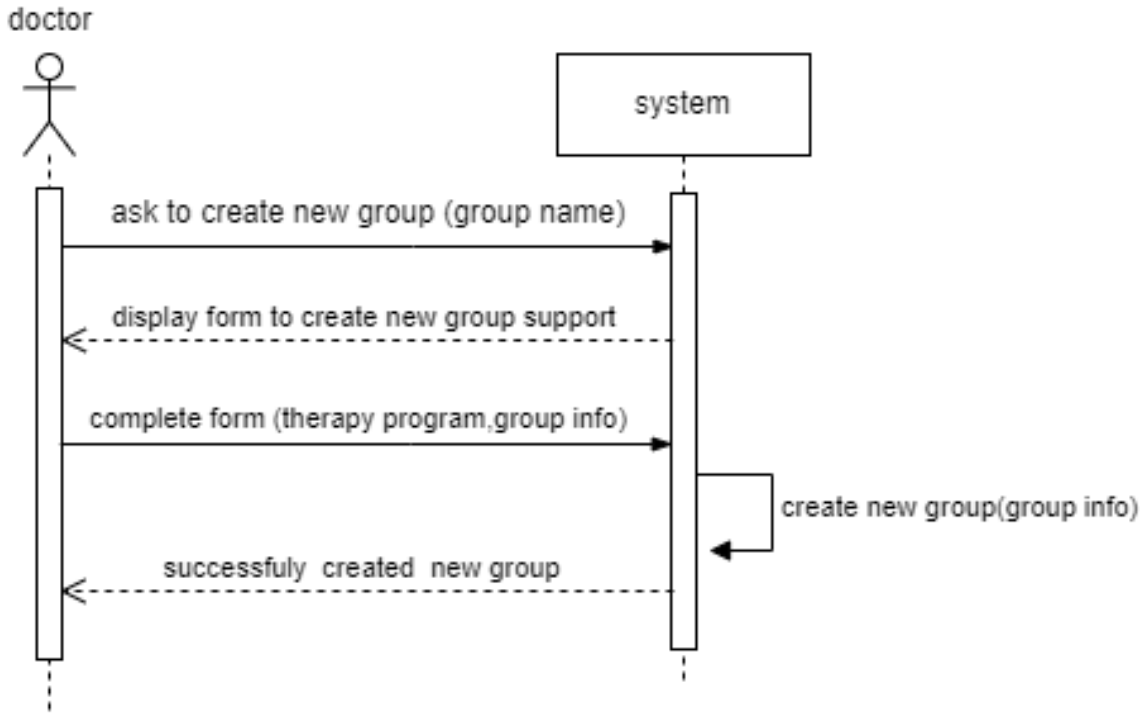
Detailed use case diagram:



Use case specification:

Use Case name	Manage support group
Participating Actors	Initiated by: doctor
The flow of events	7. The doctor enter the "Manage support group" page 8. The system shows page include "create new group ,view created group" 9. If the doctor choose "create group" 10. The system shows for the doctor form to create new group 11. The system asks the doctor to add "therapy programs, info about group" 12. If the doctor add info 13. The system display form include (title, description, domain) 14. The system saves the group 15. If the doctor choose add therapy program 16. The system display form to add therapy program 17. The doctor enter therapy programs 18. The system save this therapy program
Alternative flows	First alternative flows A1: start at step 3in the main flow, create group already exist: 4:The system will show an error message "the group already exist " 6:the doctor will recreate the new group and the flow will return to step 3 Second alternative flows A2: start at step 11in the main flow 12: The system will show an error message "the fields therapy program missing" 13: the doctor will add therapy program and the flow will return to step11.
Entry condition	The doctor logged in
Exit condition	Changes made to group are saved in the system

Sequence diagram (Create Group):



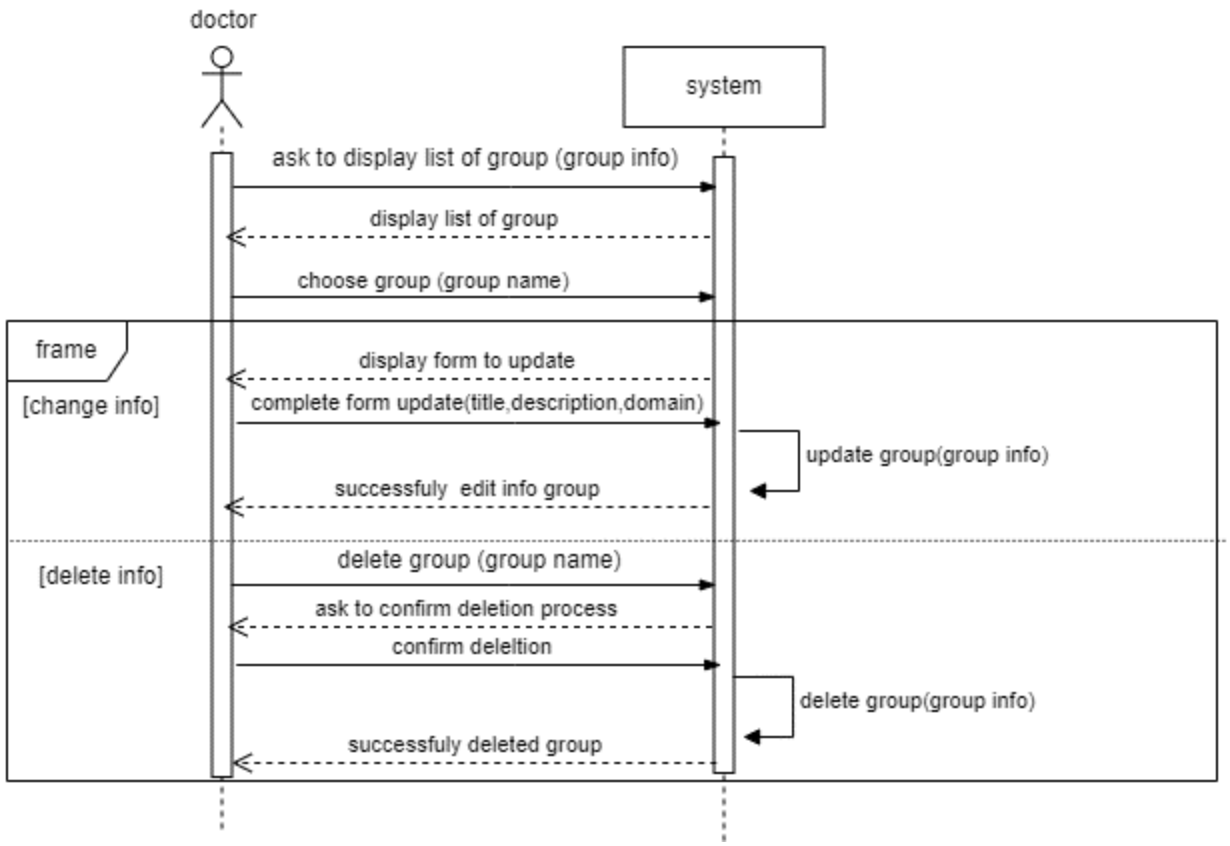
- Manage support group (View created group):

Use case specification:

Use Case name	View created group
Participating Actors	Initiated by: doctor
The flow of events	1. The doctor enter the "View created group" page 2. The system shows page include "change info, delete info, view list of participle patients" 3. If the doctor choose "change info" 4. The system shows form new (title, description, domain) 5. The doctor enters the new info 6. The system saves the new info in db 7. If the doctor choose "delete info"

	8. The system display list of group that existing in the system 9. The doctor choose the group he wants to delete 10. The system delete the group from db 11. If the doctor choose "view list of participle patients" 12. The system display page include list of participle patients and their info
Entry condition	The doctor logged in
Exit condition	Changes made to group are saved in the system

Sequence diagram (Manage support group):



- ## View list of appointment:

Detailed use case diagram:

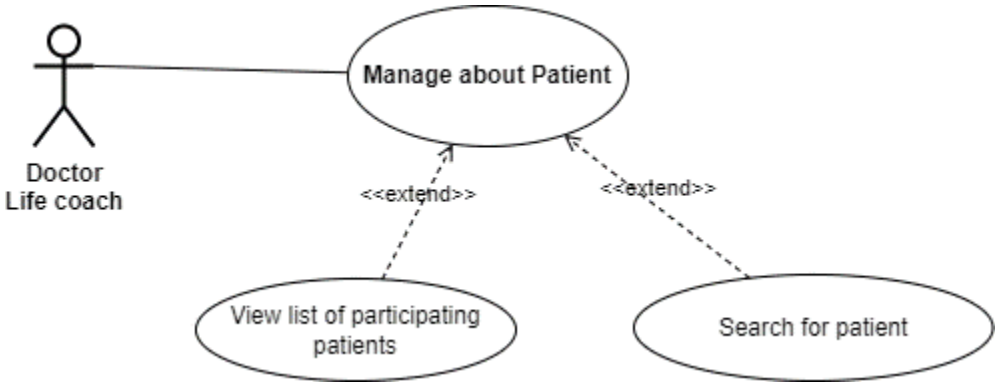


Use case specification:

Use Case name	View list of appointment
Participating Actors	Initiated by: doctor
The flow of events	1. The doctor enter the "View list of appointment" page 2. The system display page include "send prescription" 3. the doctor after done the session he send prescription for patient by messaging
Entry condition	The doctor logged in
Exit condition	The doctor review patient's appointments

- ## Manage about patient:

Detailed use case diagram:

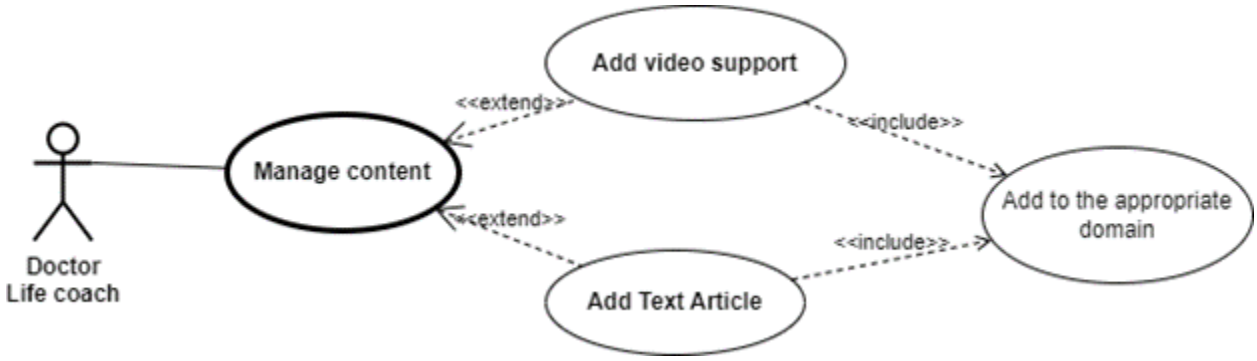


Use case specification:

Use Case name	Manage about patient
Participating Actors	Initiated by: doctor
The flow of events	1. The doctor enter the "Manage about patient" page 2. The system shows page include "view list of participating patients, search for patient" 3. If the doctor choose "search for patient" 4. The system display a text form to enter the patient's name 5. The doctor enter patient name 6. The system display the account of patient and his info and his medical record 7. If the doctor choose "view list of participating patients" 8. The system shows the list of patient that participating and info time and appointments
Alternative flows	First alternative flows A1: start at step 4in the main flow, search for patient not exist: 5:The system will show an error message "patient is not exist " 6:the doctor will re write the correct name and the flow will return to step 6
Entry condition	The doctor logged in
Exit condition	The doctor review patient's info

- ## Manage Content:

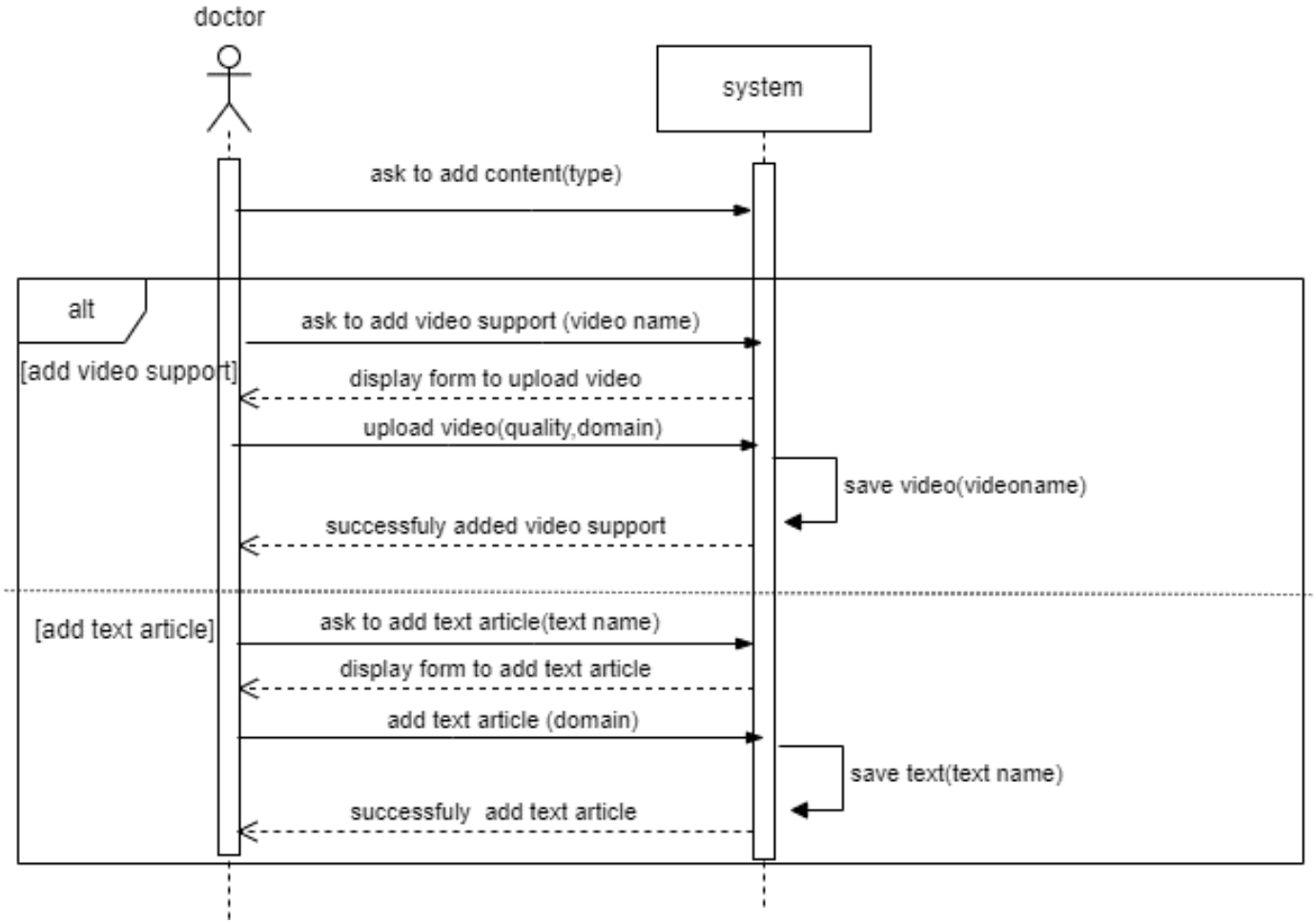
Detailed use case diagram:



Use case specification:

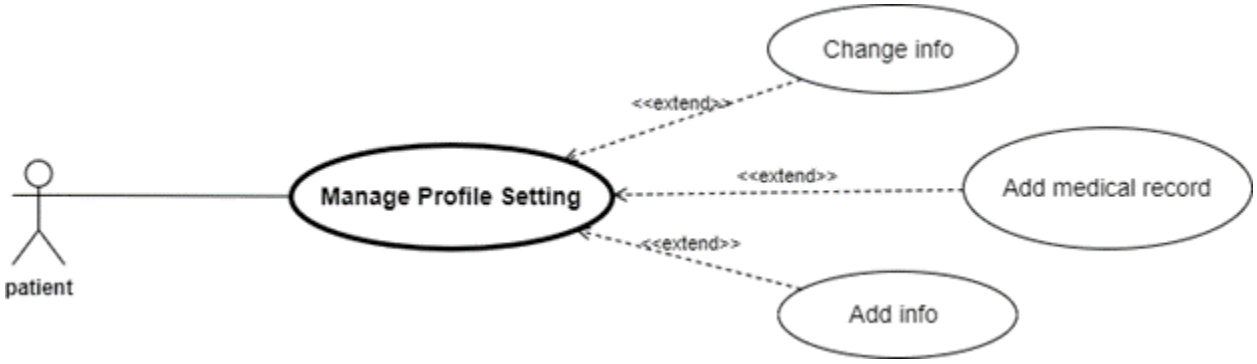
Use Case name	Manage content
Participating Actors	Initiated by: doctor
The flow of events	1. The doctor enter the "Manage content " page 2. The system shows page include "add video support, add text article " 3. If the doctor choose "add video support" 4. The system displays form to add new video with options for quality and asks doctor to which domain 5. The doctor upload the video with his domain and quality 6. The system saves the video in his domain in the db 7. If the doctor choose "add text article" 8. The system displays form to add new article and asks doctor to which domain 9. The doctor add the article with his domain 10. The system saves the article in his domain in the db
Alternative flows	First alternative flows A1: start at step 5in the main flow, : 6:The system will show an error message "the video in low quality" 7:the doctor will re upload the video and the flow will return to step 5
Entry condition	The doctor logged in
Exit condition	Changes made to content are saved in the system

Sequence diagram (Manage Content):



Manage Profile Setting:

Detailed use case diagram:

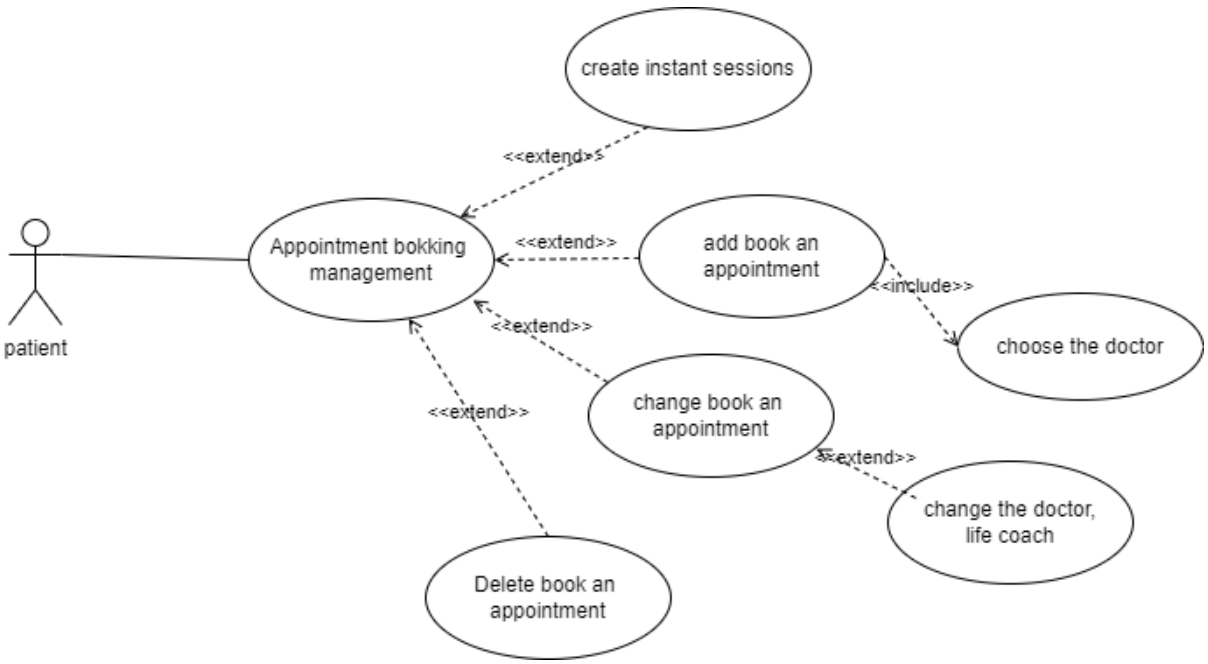


Use case specification:

Use Case name	Manage Profile Settings
Participating Actors	Initiated by: Patient
The flow of events	1. The patient enter the "profile settings" page 2. The system display page include "change info, add medical record , add info" 3. If the patient choose the "add info" 4. The system shows form to add info 5. The patient add info 6. The system saves patient info in db 7. If the patient choose "change info" 8. The system shows form to change his info 9. The patient change info 10. The system shows message to confirmation this process 11. The patient confirm this process 12. The system saves this changes in db 13. If the patient choose "add medical record " 14. The system shows form to add medical record 15. The patient upload his file 16. The system saves the file in db and add to patient account.
Entry condition	The patient logged in
Exit condition	The patient changes his profile data

Appointment booking management:

Detailed use case diagram:



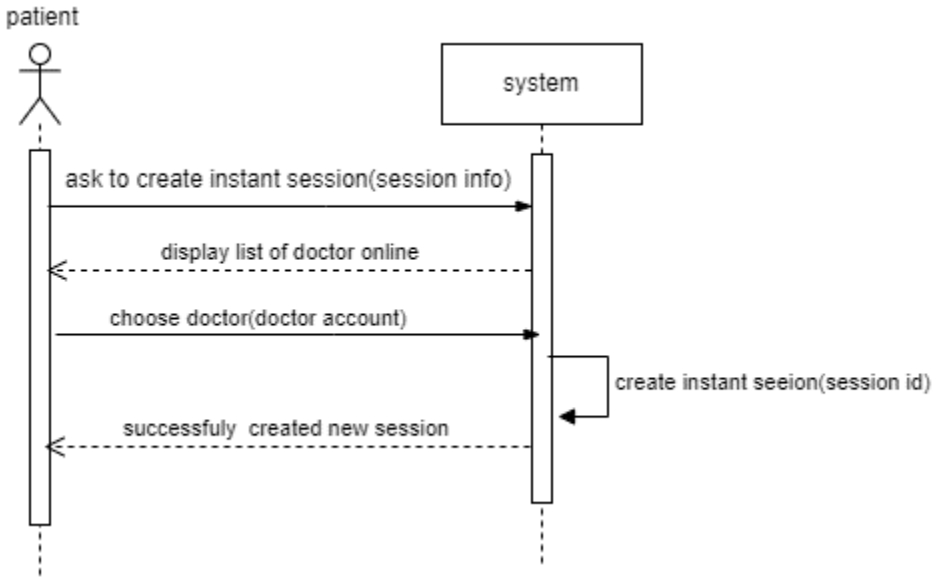
Create instant session:

Use case specification:

Use Case name	Appointment booking management
Participating Actors	Initiated by: patient
The flow of events	1. The patient enter the "Appointment booking management" page 2. The system displays page include "create instant session, add an appointment, change an appointment, delete an appointment" 3. If the patient choose "create instant session" 4. The system displays the list of the doctor is online in the system

	5. The patient chose the prefer doctor 6. The system asks the patient to set communication way 7. The patient choose the prefer way 8. The system conduct the session between the doctor and the patient.
Entry condition	The patient logged in
Exit condition	Changes made to appointment are saved in the system

Sequence diagram (Create instant session):

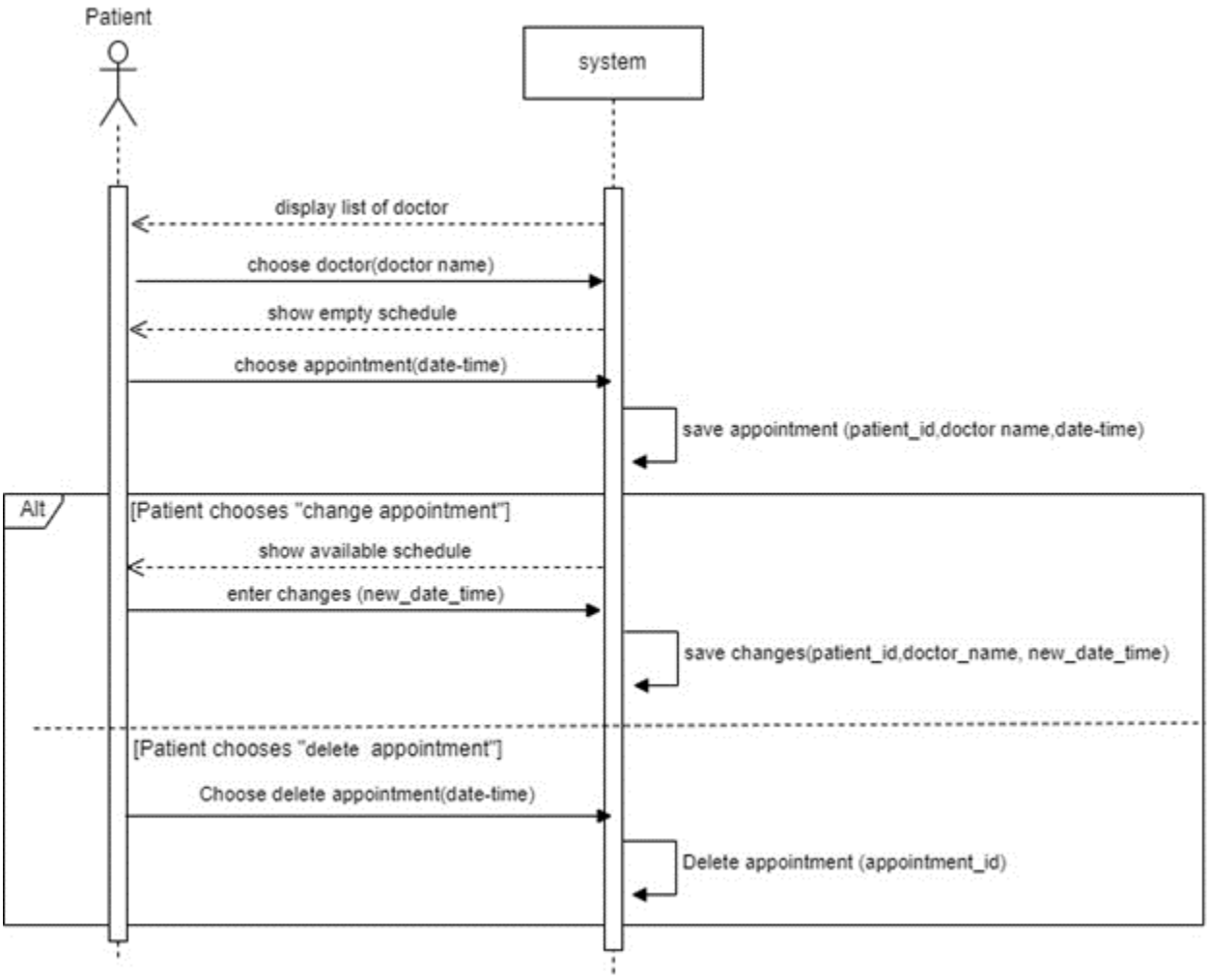


Add book an appointment:

Use case specification:

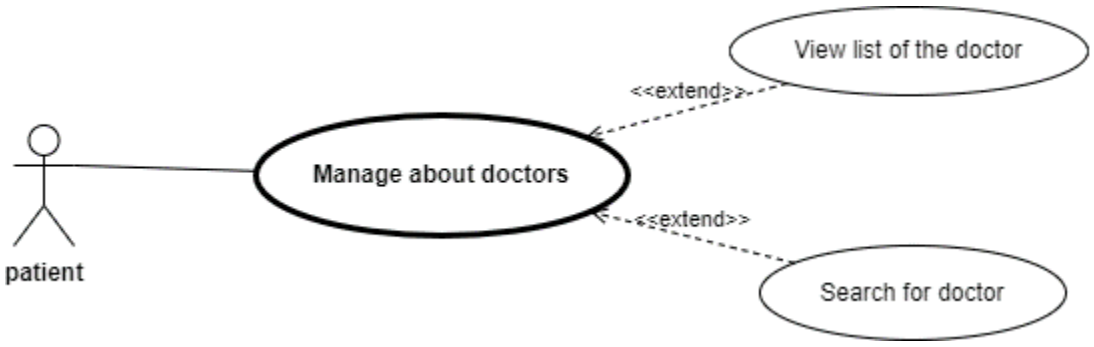
Use Case name	Add book an appointment
Participating Actors	Initiated by: patient
The flow of events	1. The patient enter the "Add book an appointment" page 2. The system displays list of the doctor to choose the best option for patient 3. The patient choose the doctor 4. The system shows the patient empty schedule for the doctor and to chooses the appropriate date 5. The patient choose the appointment 6. The system saves the appointment in db 7. If the patient choose "change the appointment " 8. The system shows the available schedule 9. The patient enter changes 10. The system saves this changes 11. If the patient choose "delete appointment" 12. The system delete from db
Alternative flows	First alternative flows A1: start at step 5in the main flow,: 5:The system will show an error message "you must add an appointment " 6:the patient will enter an appointment and the flow will return to step 6
Entry condition	The patient logged in
Exit condition	Changes made to appointment are saved in the system

Sequence diagram (Add book an appointment):



■ Manage about doctor:

Detailed use case diagram:

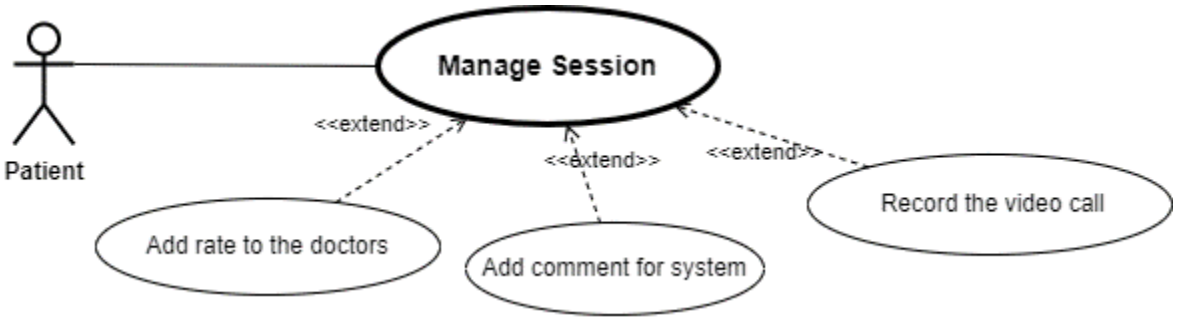


Use case specification:

Use Case name	Manage about doctor
Participating Actors	Initiated by: patient
The flow of events	1. The patient enter the "Manage about doctor" page 2. The system displays page include "view list of the doctor, search for doctor" 3. If the patient choose " view list of the doctor" 4. The system displays list of available doctor on the system 5. If the patient choose "search for doctor" 6. The system displays form to search 7. The patient enter the name of doctor 8. The system displays account of doctor with his info
Alternative flows	First alternative flows A1: start at step 7in the main flow, doctor not exist: 8:The system will show an error message " not fount the account doctor " 9:the user will reenter the name and the flow will return to step 6
Entry condition	The patient logged in
Exit condition	The doctor review patient's info

Manage Session:

Detailed use case diagram:

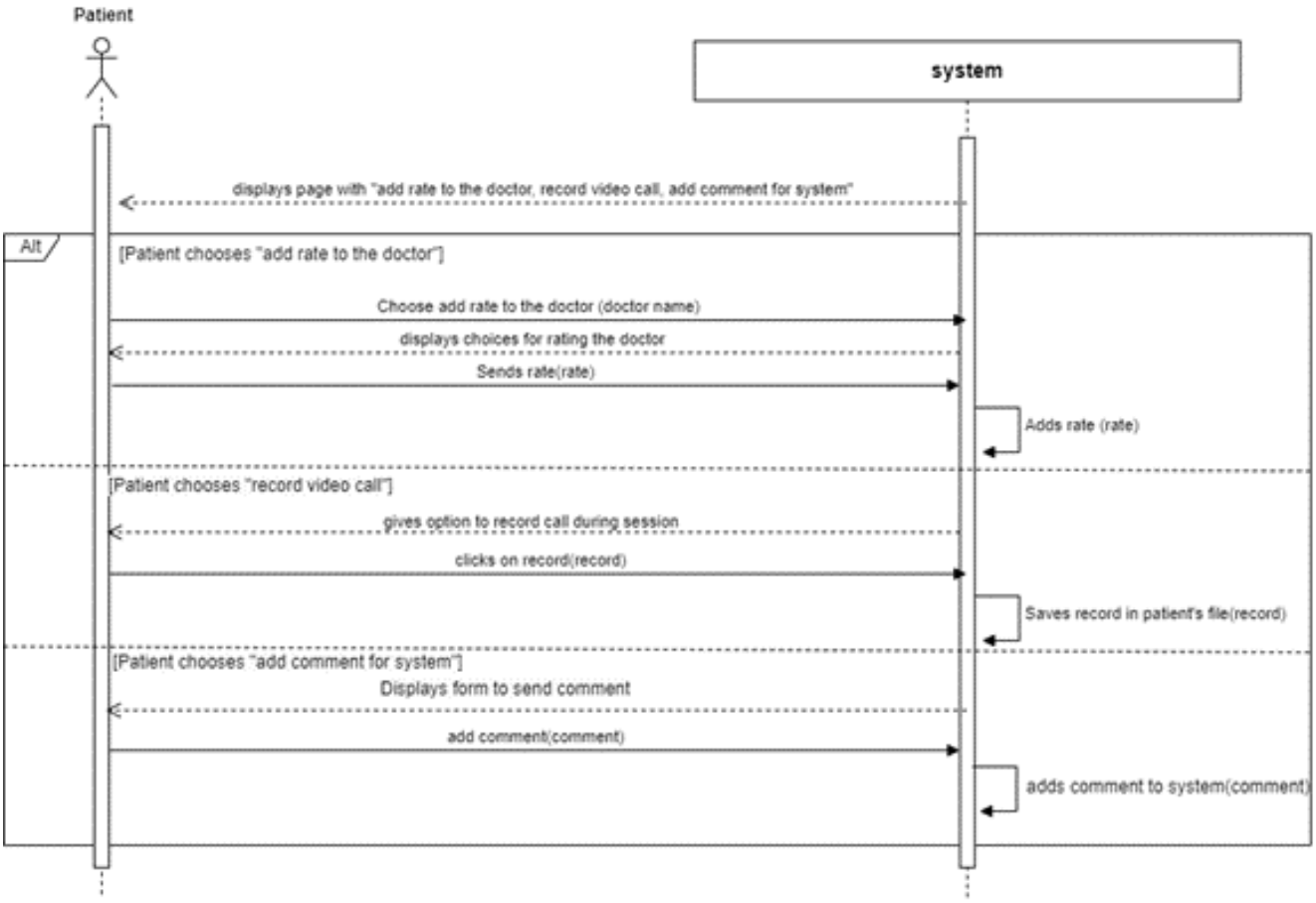


Use case specification:

Use Case name	Manage sessions
Participating Actors	Initiated by: patient
The flow of events	1. The patient enter the "Manage sessions" page. 2. The system displays page include "add rate to the doctor, record video call, add comment for system" 3. If the patient choose "add rate to the doctor" 4. The system displays choices for patient for rate the doctor 5. The patient send his rate 6. The system add this rate to db for sum this rate with another rates 7. If the patient choose "record video call" 8. The system when the patient is having the session he is given the option to record the call 9. The patient click on record 10. The system saves this record in file patient

	11. If the patient choose “add comment for system” 12. The system display form to send patient comment 13. The patient write the comment and add to the system
Entry condition	The patient logged in
Exit condition	The patient comment ,rate to the doctor

Sequence diagram (Manage Session):

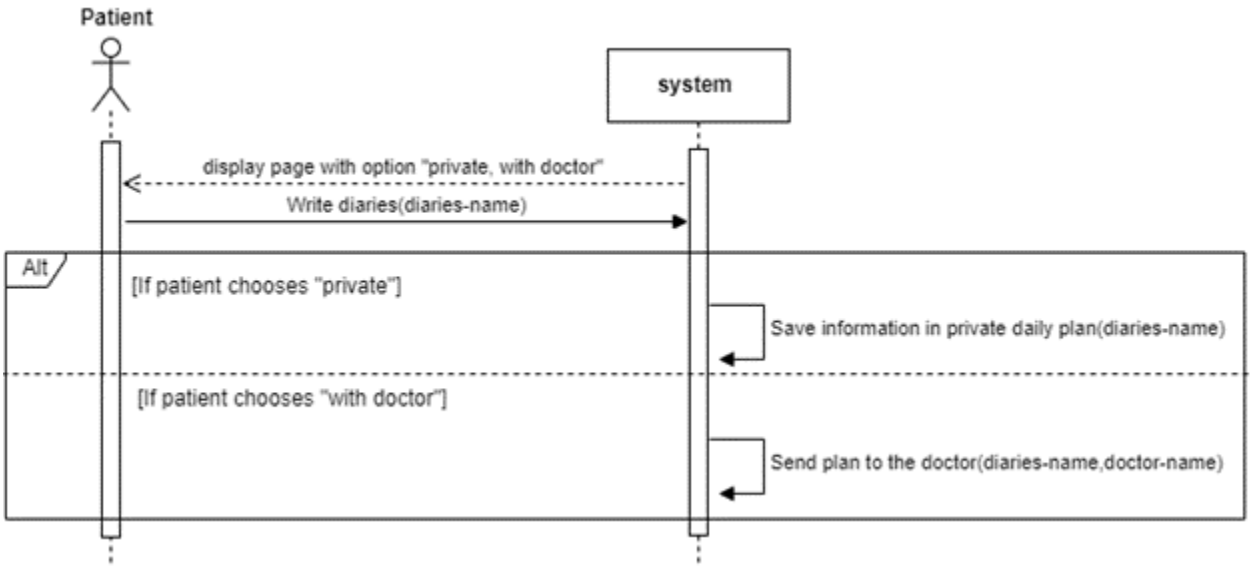


Share memories:

Use case specification:

Use Case name	Share memories
Participating Actors	Initiated by: patient
The flow of events	1. The patient enter the "Share memories" page 2. The system displays page with option "private, with doctor" 3. The system displays form to patient writes his diaries and plan 4. If the patient choose private 5. The system saves this in private plan daily info 6. If the patient choose with doctor 7. The system send the plan to the doctor
Entry condition	The patient logged in
Exit condition	The patient shared memories

Sequence diagram (Share memories):

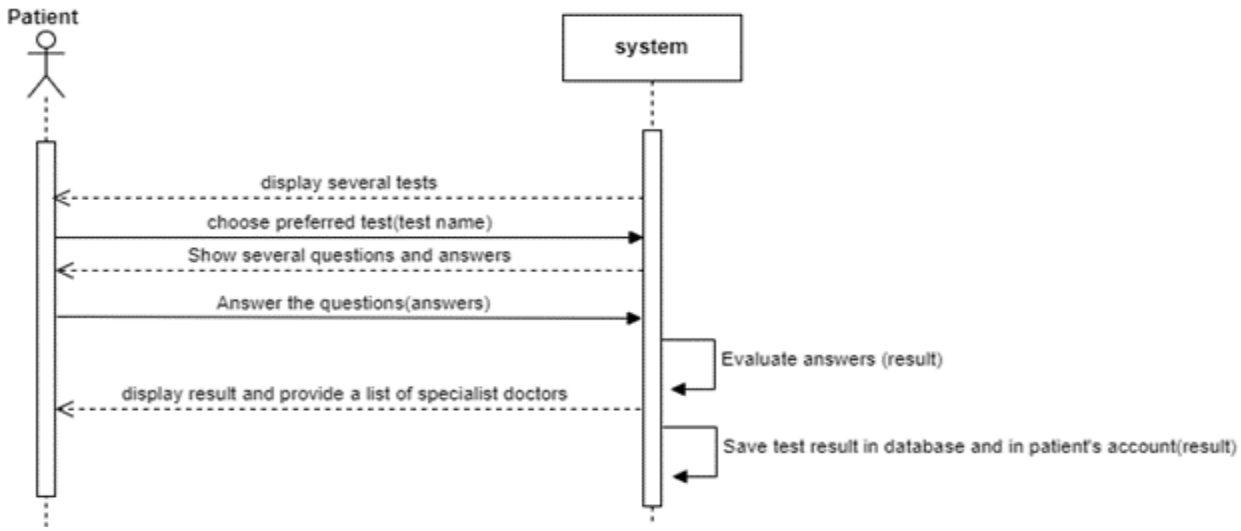


Answer psychological test:

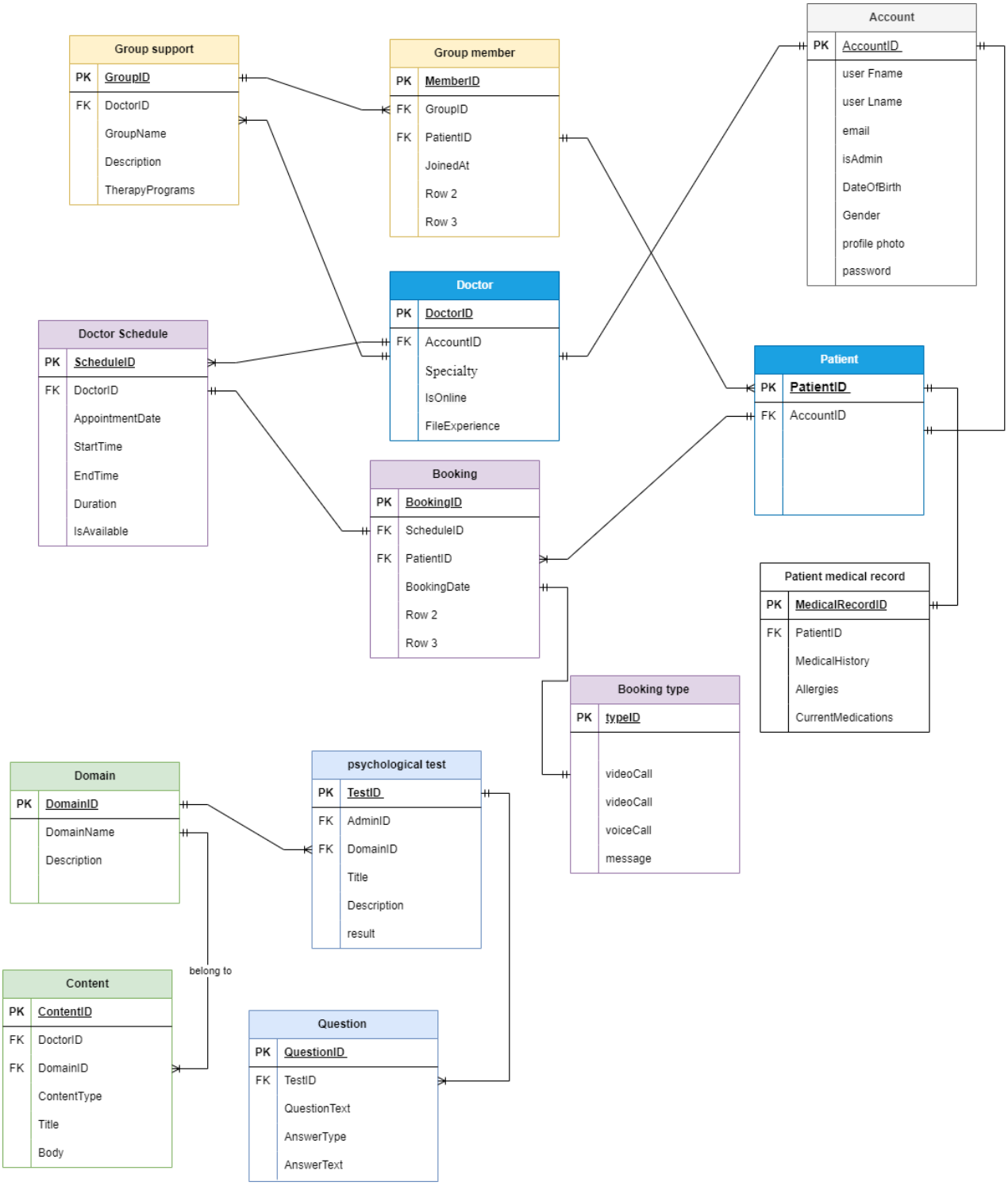
Use case specification:

Use Case name	Answer psychological test
Participating Actors	Initiated by: patient
The flow of events	1. The system displays to the patient several test 2. The patient choose the test that prefer 3. The system show a several of question and her answer 4. The patient answers the questions presented to him 5. The system evaluates the answers, saves the test result and displays it to the patient and provides him with a list of specialist doctors in the field in which he has a problem 6. The system saves the result test in db and in the account patient
Alternative flows	First alternative flows A1: start at step 4in the main flow: 5: The system will show an error message "you must answer to all question" 6: The patient will re answer the question and the flow will return to step 5.
Entry condition	The patient logged in
Exit condition	The patient answered all the question and the result was saved in system

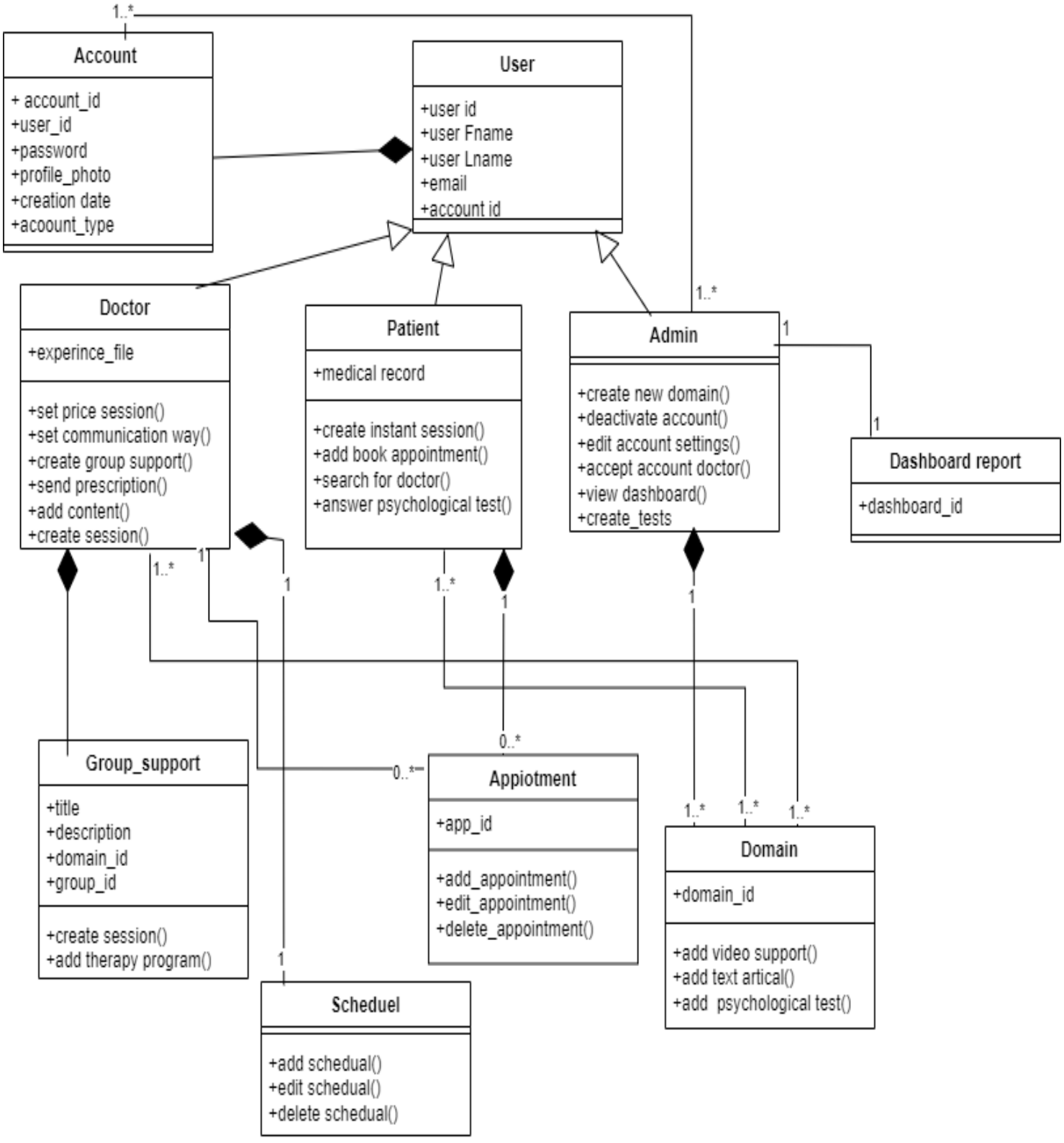
Sequence diagram (Answer psychological test):



4. Analysis ERD Table diagram:



5. Class Diagram:



4. Initial Test Cases:

(Test cases for general user)

- ***Login***

Test Case Scenario:		Login	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-01	check results on entering a valid email and password	1. launch the app on the login page 2. Enter your email and password 3. enter login	The login should be successful
TC-02	check result on entering invalid email or password	1. launch the app on the login page 2. Enter your email and password 3. enter login	Error message "the email or password is invalid"
TC-03	check result on empty fields	1. launch the app on the login page 2. Enter your email and password 3. enter login	Error message "the email or password is empty"

- ***Sign-Up***

Test Case Scenario:		Sign-Up	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-04	check results on entering a valid field	1. launch the app on the Sign-up page 2. fill in the fields with information 3. enter Sign up	The login should be successful
TC-05	Check result when entering an existing email	1. launch the app on the login page	Error message "the email is already existed "

		2. Fill in the fields with information. 3. enter Sign up	
TC-06	Check result when entering invalid password	1. launch the app on the login page 2. Fill in the fields with information. 3. enter Sign up	Error message "the password is weak "
TC-07	check result on empty fields	1. launch the app on the login page 2. Fill in the fields with information. 3. enter Sign up	Error message "the field is empty"

- ***Manage Profile Setting***

Test Case Scenario:		Manage Profile Setting	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-08	Check results on display user info for user	1. Launch the web app. 2. Launch the personal profile setting interface	The system should display the user info

- ***Manage Profile Setting (Change password)***

Test Case Scenario:		Manage Profile Setting (Change password)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-09	Check results on entering the correct old password and a strong new password	1.launch the app on the personal profile setting page 2. Choose to “change password” 3.Enter the old password	The password must be changed successfully

		4. Enter a new password	
TC-10	Check results on entering an incorrect old password	1.launch the app on the personal profile setting page 2. Choose to "change password" 3.Enter the old password 4. Enter a new password	Error message "incorrect old password"
TC-11	Entering a new password that is not strong enough	1.launch the app on the personal profile setting page 2. Choose to "change password" 3.Enter the old password 4. Enter a new password	Error message "New password is not strong enough"

- ***Manage Profile Setting (Change Photo)***

Test Case Scenario:		**Manage Profile Setting (Change Photo)**	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-12	Check results on uploading the correct format for changing profile pictures	1.launch the app on the personal profile setting page 2. Choose to change your profile photo 3. Upload a photo	The photo must be changed successfully
TC-13	Check results on uploading incorrect format	1.launch the app on the personal profile setting page 2. Choose to change your profile photo 3. Upload a photo	Error message "uploaded format is not supported"

- *Manage Profile Setting (Change Personal info)*

Test Case Scenario:		Manage Profile Setting (Change Personal info)	
Test case id	Test case title	Test steps	Expected result
TC-14	Check results on updating the username with strings	1.launch the app on the personal profile setting interface 2. Enter new username 3. Press "enter"	The username updated
TC-15	Check results on updating the username with numbers only	1.launch the app on the personal profile setting interface 2. Enter new username 3. Press "enter"	Error message "please enter strings"

(Test cases for Admin):

- *Manage Domain (Add domain)*

Test Case Scenario:		Manage Domain (Add domain)	
Test case id	Test case title	Test steps	Expected result
TC-16	Check results on adding a new domain with valid info	1. Launch the app on "Manage Domain" page 2. Choose "Add Domain" 3. Complete a form with the information 4. Save the domain	The domain must be added successfully

TC-17	Check results on adding a new domain with an invalid name	1. Launch the app on “Manage Domain” page 2. Choose “Add Domain” 3. Complete a form with the information 4. Save the domain	Error message "invalid domain name"
TC-18	Check results on adding a new domain with empty name	1. Launch the app on “Manage Domain” page 2. Choose “Add Domain” 3. Complete a form with the information 4. Save the domain	Error message "domain name cannot be empty"
TC-19	Check results on adding a new domain but is already exists	1. Launch the app on “Manage Domain” page 2. Choose “Add Domain” 3. Complete a form with the information 4. Save the domain	Error message “The domain already exists”

- ***Manage Domain (Edit domain)***

Test Case Scenario:		**Manage Domain (Edit domain)**	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-20	Check results on editing an existing domain with valid info	1. Launch the app on “Manage Domain” page 2. Choose “Edit Domain” 3. Select domain to edit 4. Make edits 4. Save changes	The domain must be edited successfully

TC-21	Check results on editing with an invalid name	1. Launch the app on “Manage Domain” page 2. Choose “Edit Domain” 3. Select domain to edit 4. Make edits 4. Save changes	Error message "invalid domain name”
TC-22	Check results on editing a domain with empty name	1. Launch the app on “Manage Domain” page 2. Choose “Edit Domain” 3. Select domain to edit 4. Make edits 4. Save changes	Error message "domain name cannot be empty”

- ***Manage Domain (Delete domain)***

Test Case Scenario:		Manage Domain (Delete domain)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-23	Check results on deleting a domain	1. Launch the app on “Manage Domain” page 2. Choose “Delete Domain” 3. Select domain to delete 4. Confirm deletion	The domain must be deleted successfully
TC-24	Check results on deleting a non-existent domain	1. Launch the app on “Manage Domain” page 2. Choose “Delete Domain” 3. Select domain to delete 4. Confirm deletion	Error message "Domain not found"

- *Manage Domain (Add Psychological Test)*

Test Case Scenario:		Manage Domain (Add Psychological Test)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-25	Check results on adding a psychological test with valid values	1. Launch the app on "Manage Domain" page 2. Choose "Add psychological test" 3. Enter question and answer 4. Save the test	The test must be saved successfully
TC-26	Check results on adding a psychological test with invalid data	1. Launch the app on "Manage Domain" page 2. Choose "Add psychological test" 3. Enter question and answer 4. Save the test	Error message "invalid test data"

- *Manage Doctor Info*

Test Case Scenario:		Manage Doctor info	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-27	Check results on accepting a doctor with valid info	1.Launch the app on "Manage Doctor Information" interface 2. Select a specific doctor 3. Verify the doctor's information (experience file and certificate) 4. Select "Accept Doctor"	The doctor must be added to the system successfully
TC-28	Check results on accepting a doctor with incomplete info	1. Launch the app on "Manage Doctor Information" interface 2. Select a specific doctor	Error message "Doctor information is incomplete"

		3. Verify the doctor's information (experience file and certificate) 4. Select "Accept Doctor"	
TC-29	Check results on rejecting a doctor	1. Launch the app on “Manage Doctor Information” interface 2. Select a specific doctor 3. Verify the doctor's information (experience file and certificate) 4. Select "Reject Doctor"	The doctor should be rejected successfully
TC-30	Check results on deactivating a doctor	1. Launch the app on “Manage Doctor Information” interface 2. Select a specific doctor 3. Select "Deactivate Doctor" 4. Confirm the deactivation	The doctor’s account must be deactivated successfully
TC-31	Check results on deactivating a doctor without confirmation	1. Launch the app on “Manage Doctor Information” interface 2. Select a specific doctor 3. Select "Deactivate Doctor"	The doctor's account must remain active

- ***View Dashboard***

Test Case Scenario:		View Dashboard	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-32	Check results on display the dashboard of the admin	1. Launch the application by the admin 2. Launch the dashboard interface	Display dashboards

(Test cases for doctor):

- ***Manage account setting (Set price for session)***

Test Case Scenario:		Manage account setting (Set price for session)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-33	Check results on setting a valid price for session	1. Launch the app on “Manage Account Setting” interface 2. Select "Set Price Session" 3. Enter price 4. Save the price	The price must be set successfully
TC-34	Check results on setting an invalid price for session	1. Launch the app on “Manage Account Setting” interface 2. Select "Set Price Session" 3. Enter price 4. Save the price	Error message "Invalid price entered"
TC-35	Check results on leaving the price field empty	1. Launch the app on “Manage Account Setting” interface 2. Select "Set Price Session" 3. Save the price	Error message "Price field cannot be empty"

- ***Manage account setting (Set communication way)***

Test Case Scenario:		Manage account setting (Set communication way)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-36	Check results on setting a valid communication way	1. Launch the app on “Manage Account Setting” interface 2. Select "Set Communication Way" 3. Select communication way 4. Save the communication way	The communication way must be set successfully

TC-37	Check results on leaving the communication way field empty	1. Launch the app on “Manage Account Setting” interface 2. Select "Set Communication Way" 3. Save the communication way	Error message "Communication way field cannot be empty"

- ***Manage Support Group (Create Group)***

Test Case Scenario:		Manage Support Group (Create Group)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-38	Check results on creating a new group with valid info	1. Launch the app on “Manage Support Group” interface 2. Select "Create Group" 3. Complete the form with information 4. Save the group	The group must be created successfully
TC-39	Check results on creating a group that already exists	1. Launch the app on “Manage Support Group” interface 2. Select "Create Group" 3. Complete the form with information 4. Save the group	Error message "The group already exists"
TC-40	Check results on creating a group with missing info	1. Launch the app on “Manage Support Group” interface 2. Select "Create Group" 3. Complete the form with information 4. Save the group	Error message "Some fields are missing"

- ***Manage Support Group (View created Group)***

Test Case Scenario:		Manage Support Group (View created Group)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-41	Check results on changing info of a created group	1. Launch the app on “View created group” interface 2. Select "Change info" 3. Complete the form with information 4. Save the new info	The new info must be saved successfully
TC-42	Check results on changing info with missing fields	1. Launch the app on “View created group” interface 2. Select "Change info" 3. Complete the form with information 4. Save the new info	Error message "Some fields are missing"
TC-43	Check results on deleting a group successfully	1. Launch the app on “View created group” interface 2. Select "Delete info" 3. Choose the group to delete 4. Confirm the deletion	The group must be deleted successfully
TC-44	Check results on deleting a non-existent group	1. Launch the app on “View created group” interface 2. Select "Delete info" 3. Choose the group to delete 4. Confirm the deletion	Error message "Group not found"
TC-45	Check results on viewing the list of participle patients	1. Launch the app on “View created group” interface 2. Select "View list of participle patients" 3. View the list displayed	The system must display the list of participle patients and their info successfully

Upload Doctor Information (CSV Files)

Test Case Scenario:		*Upload Doctor Information*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-46	Verify successful upload of a valid CSV file	1. Log in as a doctor 2. Navigate to " Profile" "Upload Information" 3. Click "ok".	System confirms: "File uploaded successfully."

Delete Account

Test Case Scenario:		*Delete Account*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-47	Verify successful Delete account	1. Log in as a doctor 2. Navigate to "Profile" 3. Delete Account 4. Confirm deletion	The account is deleted successfully and the user is redirected to the home page.

Add info

Test Case Scenario:		*Change Doctor Information*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-48	Verify successful added of doctor information	1. Log in as doctor 2. Go to "Profile" 3. Click “add Information” 4. Complete fields 5. Click "Save"	System displays: "Information added successfully."

Change Doctor Information

Test Case Scenario:		*Change Doctor Information*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-48	Verify successful update of doctor information	1. Log in as doctor 2. Go to "Profile" 3. Click “Edit Information” 4. Update fields 5. Click "Save"	System displays: "Information updated successfully."
TC-49	Verify update with empty required fields	1. Log in as doctor 2. Go to "Profile" 3. Click “Edit Information” 4. Click "Save"	Error message: "fields cannot be empty."

Manage schedule (Add schedule):

Test Case Scenario:		*Add schedule*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-50	Verify adding a valid schedule.	1. Log in as doctor 2. Go to "Manage Schedule" section 3. add the information of the session 4. click “confirm”	System displays: “created session successfully”
TC-51	Verify adding overlapping session	1. Log in as doctor 2. Go to "Manage Schedule" section 3. add the information of the session 4. click “confirm”	Error message: "the session overlaps with another session "

Manage schedule (Change Schedule):

Test Case Scenario:		*Change Schedule*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-52	Verify rescheduling a booked session	1. Log in as doctor 2. Go to " view appointment schedule " section 3. Edit the information of the session 4. click “save”	System updates the session

Manage schedule (Delete Schedule):

Test Case Scenario:		*Change Schedule*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-53	Verify deleting session	1. Log in as doctor 2. Go to " view appointment schedule " section 3. Click “delete” 4. Confirm deletion	System displays: "session deleted successfully.

Create Video Call

Test Case Scenario:		*Create Video Call*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-54	Verify creating a new video call session successfully	1. Log in as doctor 2. Go to "Appointments" 3. Choose patient 4. Click "New Video Call"	System displays: "Video call session created successfully."

Create Voice Call

Test Case Scenario:		*Create Voice Call*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-55	Verify creating a new voice call session successfully	1. Log in as doctor 2. Go to "Appointments" 3. Choose patient 4. Click "New voice Call"	System displays: "Voice call session created successfully."

Create Message Session

Test Case Scenario:		*Create Message Session*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-56	Verify creating a message session successfully	1. Log in as doctor 2. Go to "Appointments" 3. Choose patient 4. Click "message session"	System displays: " Message session started successfully"

Manage Support Grouping(Add Support Group Information)

Test Case Scenario:		*Add Support Group Information*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-57	Verify adding a new support group with valid info	1. Log in as doctor 2. Go to "Support Groups" then "Add New Group" 3. Enter group info 4. Click "Save"	System displays: "Support group created successfully"
TC-58	Verify adding a group with empty field info	1. Log in as doctor 2. Go to "Support Groups" then "Add New Group" 3. Click "Save"	Error message: "Group info cannot be empty"

Add Therapy Programs

Test Case Scenario:		*Add Therapy Programs*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-59	Verify adding a new therapy program with valid details	1. Log in as doctor 2. Go to "Support Groups" 3."select group 4. Click " adding a new therapy " 5. Click “ok”	System displays: "Therapy program added successfully."

Send Medical Prescription

Test Case Scenario:		*Send Medical Prescription*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-60	Verify sending a prescription to a selected patient	1. Log in as doctor 2. Go to "manage patient" 3. Select patient 4. Click "Send Medical Prescription" 5. Click “ok”	System displays: "Prescription sent successfully."

Add personal information (patient)

Test Case Scenario:		Add personal information (patient)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-62	Verify adding personal information successfully	1. Log in as doctor 2. Go to "Profile" 4. Enter valid personal information 5. Click “save”	System displays: “Information saved successfully."

Modifying personal information (patient)

Test Case Scenario:		Modifying personal information (patient)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-63	Verify modifying personal information successfully	1. Log in as patient 2. Go to " edit Profile" 3. Modify existing information 4. Click “save”	System displays: “Information updated successfully."

Booking a new appointment

Test Case Scenario:		Booking a new appointment	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-64	Verify booking a new appointment successfully	1. Log in as patient 2. Go to “Appointments" 3. Click “Book New Appointment” 4. Choose doctor/life coach and time 5. Click "Confirm"	System displays: "Appointment booked successfully."

Changing an existing appointment

Test Case Scenario:		Changing an existing appointment	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-65	Verify changing an existing appointment	1. Log in as patient 2. Go to “Appointments" 3. Select “an Appointment” 4.Edit time or details 5. Click "Save"	System displays: Appointment updated successfully."

Deleting an appointment

Test Case Scenario:		*Deleting an appointment*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-66	Verify deleting an existing appointment	1. Log in as patient 2. Go to “Appointments" 3. Select “an Appointment” 4. Click "Save"	System displays: Appointment deleted successfully."

Choosing a doctor or life coach

Test Case Scenario:		*Choosing a doctor or life coach*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-67	Verify choosing a doctor or life coach	1. Log in as patient 2. Go to “Book Appointments" 3. Browse list of doctor 4. Select one	System displays: “Doctor/Life coach selected successfully."

Edit Choosing a doctor or life coach

Test Case Scenario:		*Edit Choosing a doctor or life coach*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-68	Verify changing selected doctor or life coach	1. Log in as patient 2. Go to “manage Appointments" 3.select an appointment 4.change assigned doctor/life coach 5. Click “Save”.	System displays: “Assigned doctor/life coach updated successfully."

Browsing all available doctors

Test Case Scenario:		Browsing all available doctors	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-69	Verify browsing all available doctors and life coaches	1. Log in as patient 2. Go to “Browse Professionals" 3. View full list of doctors/life coaches	System displays: “list of all doctors and life coaches."

Browsing online doctors

Test Case Scenario:		Browsing all available doctors	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-70	Verify browsing online doctors and life coaches	1. Log in as patient 2. Go to “Browse Professionals" 3. Apply filter: “Online Only”.	System displays: “list of currently online doctors and life coaches."

Search for doctors

Test Case Scenario:		Search for doctors	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-71	Verify search functionality for doctors/life coaches	1. Log in as patient 2. Go to “Browse Professionals" 3. Enter search term (name, specialty, etc.)	System displays: relevant search results based on input

Sharing daily diary

Test Case Scenario:		Sharing daily diary	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-72	Verify sharing daily diary	1. Log in as patient 2. Go to “Diaries" 3.add or write a diary entry 4.click: “Share”	System displays: “Diary shared successfully”

Starting an instant session

Test Case Scenario:		starting an instant session	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-73	Verify starting an instant session	1. Log in as patient 2. Go to “instant session " 3.click: “Start Session”	System displays: “starts the instant session successfully”

Rating doctor or session

Test Case Scenario:		Rating doctor or session	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-74	Verify rating a doctor or session	1. Log in as patient 2. Go to “Rating" 4.add a rating and submit 5. Click: “Save”.	System displays: “Rating submitted successfully”

Taking psychological a test

Test Case Scenario:		Taking psychological a test	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-75	Verify Taking psychological a test	1. Log in as patient 2. Go to "psychological test " 3.Select and complete a test 4.Submit answers	System displays: "records the test results successfully"

Commenting on the system

Test Case Scenario:		Commenting on the system	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-76	Verify commenting on the system or doctors	1. Log in as patient 2. Go to "Feedback" 3.choose system or doctor 4.write and submit a comment	System displays: "comment added successfully"

Recording a video call with doctor/life coach

Test Case Scenario:		*Recording a video call with doctor/life coach*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-77	Verify recording a video call with doctor/life coach	1. Log in as patient 2. Join a video call session with doctor/life coach 3.click: "Record"	System displays: "records and saves the video call successfully"

Searching for a domain by name (Admin)

Test Case Scenario:		Verify searching for a domain by name	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-78	Verify searching for a domain by name (Admin)	1. Log in as admin 2. Go to: "Domain management" 3. Enter domain name in the search field 4. click "Search"	System displays: "search results matching the domain name"

Managing domain status (Admin)

Test Case Scenario:		Managing domain status(Admin)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-79	Verify managing domain status	1. Log in as admin 2. Go to: "Domain management" 3.Select a domain 4. Change the status (Active, Inactive, Pending) 4. click "Save"	System displays: "Domain status updated successfully"

Sorting domains by date or name (Admin)

Test Case Scenario:		Searching for a domain by name (Admin)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-80	Verify sorting domains by date or name	1. Log in as admin 2. Go to: "Domain management" 3. choose sorting criteria (Newest, Oldest, A-Z, Z-A)	System displays: "domains sorted based on selected criteria"

Displaying the number of activated accounts (Admin)

Test Case Scenario:		Displaying the number of activated accounts(Admin)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-81	Verify displaying the number of activated accounts	1. Log in as admin 2. Go to: “Dashboard” 3. View activated accounts counter	System displays: “the total number of activated user accounts”

Logout

Test Case Scenario:		Logout	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-81	Verify logout functionality for any user	1. Log in as any user (patient, doctor, or admin) 2. Go to: “Logout” 3. Select “Logout”	System displays: “logs the user out and redirects to the login page”

Content Management

Test Case Scenario:		Content Management(add a new text article)	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-82	Verify adding a new text article	1. Log in as doctor 2. Navigate to "Content Management" 3. Click "Add New Article" 4. Enter title and content 5. Click "Save"”	System displays: "Article added successfully"

Content Management

Test Case Scenario:		Content Management(adding a new video support)	
Test case id	Test case title	Test steps	Expected result
TC-83	Verify adding a new video support	1. Log in as doctor 2. Navigate to "Content Management" 3. Click "Add New Video" 4. Upload a video file 5. Click "Save"	System displays: "Video support added successfully"

Manage patient:

Test Case Scenario:		*Manage patient*(search for patients)	
Test case id	Test case title	Test steps	Expected result
TC-84	Verify searching for patients	1. Log in as doctor 2. Navigate to "Appointment Management" 3. Enter patient name or ID in the search field 4. Click "Search"	System displays: "Search results matching patient details"

Manage Support Group:

Test Case Scenario:		*Manage Support Group* (search for patients)	
Test case id	Test case title	Test steps	Expected result
TC-85	Verify browsing the list of participating patients in a support group	1. Log in as doctor 2. Navigate to "Group Support" 3. Open a specific group 4. Click "View Participants"	System displays: "List of participating patients"

Manage appointments of patient:

Test Case Scenario:		*Appointments of patient*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-86	Verify browsing patient appointment schedule	1. Log in as doctor 2. Navigate to "Appointment Management" 3. Click "View Appointment Schedule"	System displays: "List of scheduled appointments"

Manage Doctor:

Test Case Scenario:		*Doctor status*	
Test case id	**Test case title**	**Test steps**	**Expected result**
TC-87	Verify displaying doctor status (online/offline)	1. Log in as doctor 2. Navigate to "Account Settings" 3. View status indicator	System displays: "Doctor status: Online/Offline"

5. Initial RTM Requirements Tractability Matrix:

REQ ID	Use cases	analysis	design	coding	Test cases
RE-FR-AM-01					TC-04 TC-05 TC-06 TC-07
RE-FR-AM-02					TC-01 TC-02 TC-03
RE-FR-AM-03					Tc-82
RE-FR-AM-04					TC-09 TC-10 TC-11
RE-FR-AM-05					TC-27 TC-28
RE-FR-AM-06					TC-29
RE-FR-AM-07					TC-32
RE-FR-AM-08					TC-32
RE-FR-AM-09					TC-30 TC-31
RE-FR-AM-10					TC-81
RE-FR-AM-11					TC-32
RE-FR-AM-12					TC-16 TC-17 TC-18 TC-19
RE-FR-AM-13					TC-23 TC-24
RE-FR-AM-14					TC-20 TC-21 TC-22

RE-FR-AM-15					TC-25 TC-26
RE-FR-AM-16					TC-46
RE-FR-AM-17					TC-08
RE-FR-AM-18					TC-47
RE-FR-AM-19					TC-12 TC-13
RE-FR-AM-20					TC-14 TC-15
RE-FR-AM-21					TC-61
RE-FR-AM-22					TC-48 TC-49
RE-FR-SM-23					TC-50 TC-51
RE-FR-SM-24					TC-52
RE-FR-SM-25					TC-53
RE-FR-APM-26					TC-36 TC-37
RE-FR-APM-27					TC-54
RE-FR-APM-28					TC-55
RE-FR-APM-29					TC-56
RE-FR-APM-30					TC-38 TC-39 TC-40
RE-FR-APM-31					TC-57 TC-58
RE-FR-APM-32					TC-41 TC-42
RE-FR-APM-33					TC-43 TC-44
RE-FR-APM-34					TC-59
RE-FR-APM-35					TC-60
RE-FR-APM-36					TC-33 TC-34 TC-35
RE-FR-APM-37					TC-82

RE-FR-APM-38					TC-83
RE-FR-APM-39					TC-84
RE-FR-APM-40					TC-85
RE-FR-APM-41					TC-86
RE-FR-APM-42					TC-87
RE-FR-APM-43					TC-62
RE-FR-APM-44					TC-63
RE-FR-APM-45					TC-64
RE-FR-APM-46					TC-65
RE-FR-APM-47					TC-66
RE-FR-APM-48					TC-67
RE-FR-APM-49					TC-68
RE-FR-APM-50					TC-69
RE-FR-APM-51					TC-70
RE-FR-APM-52					TC-71
RE-FR-APM-53					TC-72
RE-FR-APM-54					TC-75
RE-FR-APM-55					TC-73
RE-FR-APM-56					TC-74
RE-FR-APM-57					TC-76
RE-FR-APM-58					TC-77
RE-FR-APM-64					TC-78
RE-FR-APM-65					TC-79
RE-FR-APM-66					TC-80

Chapter4 System Design

1. Introduction:

This chapter provides a comprehensive overview of the design phase steps, including system design, architectural determination, and detailed modeling. It highlights the importance of these steps in creating a robust and efficient mental health platform that facilitates communication between patients and therapists. The system incorporates features such as video and audio sessions, messaging, mental health assessments, and the ability for therapists to create patient groups and share educational videos. These design considerations aim to ensure an intuitive and effective platform for users.

2. System Architecture:

In this section, we will discuss the system architecture, which involves defining the overall structure and design of the mental health platform. The architecture ensures seamless interaction and integration of all system components, creating a robust and efficient platform for users. This phase establishes the blueprint for efficient communication, data flow, and functionality across the system.

When building the architecture for a software system, the primary constraints to consider are the non-functional requirements (quality attributes). Below are the key non-functional requirements of our system:

- ***Extensibility:***

  The system should demonstrate high extensibility to accommodate new features such as updated diagnostic tools or new communication functionalities with minimal impact on the existing codebase or system integrity.

- ***Security:***

  The system must ensure the protection of all sensitive user data, including medical records, chat history, and video call logs. This will involve robust authentication, authorization, secure passwords, and a well-designed architecture to safeguard data.

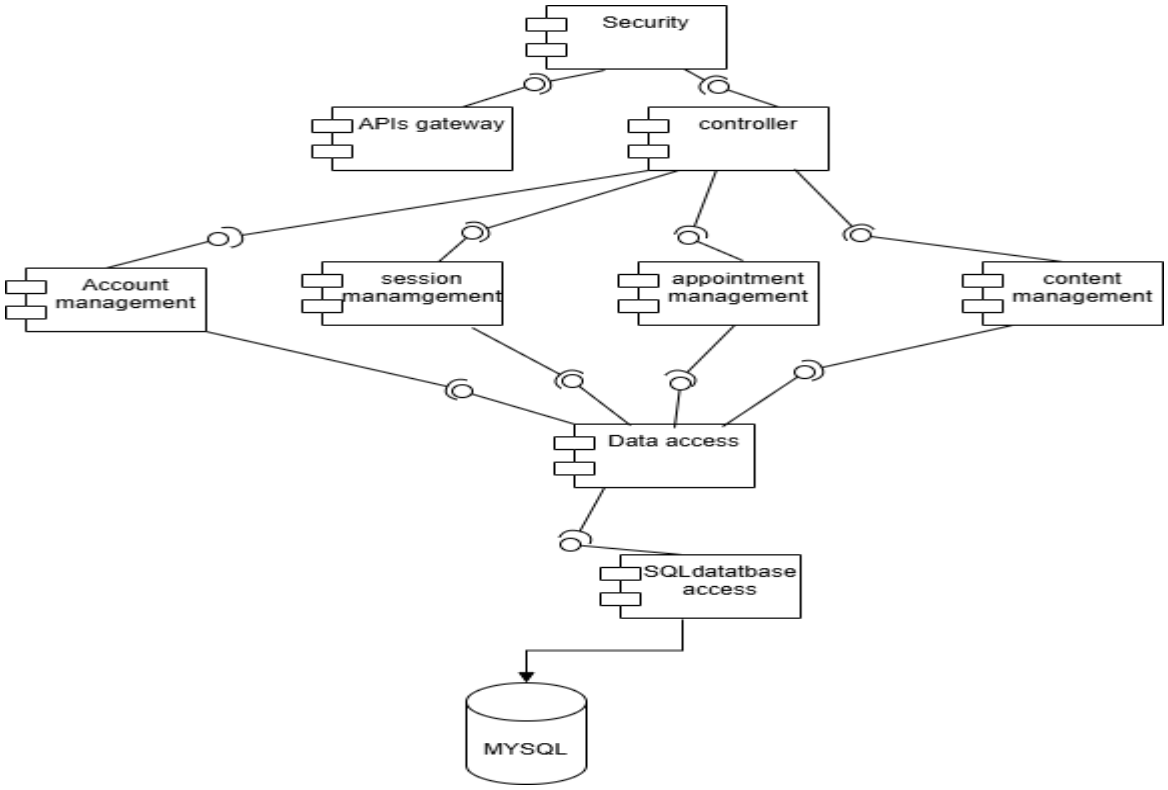
- ***Scalability:***

  The system should support up to 5,000 active users simultaneously, ensuring smooth performance for video and audio sessions, messaging, and other system features.

- ***Performance:***

  The system should deliver results for user requests within an average of 3 seconds, except for complex operations such as generating detailed mental health assessments, which may take up to 1 minute.

System decomposition - Component diagram:



Components functionalities:

1. **APIs Gateway :**

   Responsible for routing all external API calls to the appropriate controller, ensuring smooth communication between external systems and the platform.

2. **Controllers:**

   Acts as the interface for APIs, handling incoming requests and coordinating actions between services and other components to ensure proper functionality.

3. **Session Management:**

   Handles all aspects of user sessions, including authentication, maintaining active sessions, and ensuring smooth transitions between different system functionalities.

4. **Appointment Management:**

   Manages booking, canceling, and rescheduling appointments. Ensures proper allocation of time slots between patients and therapists, improving user experience.

5. **Content Management:**

   Enables therapists to upload, edit, and manage educational content such as videos, documents, and guides to improve patient care.

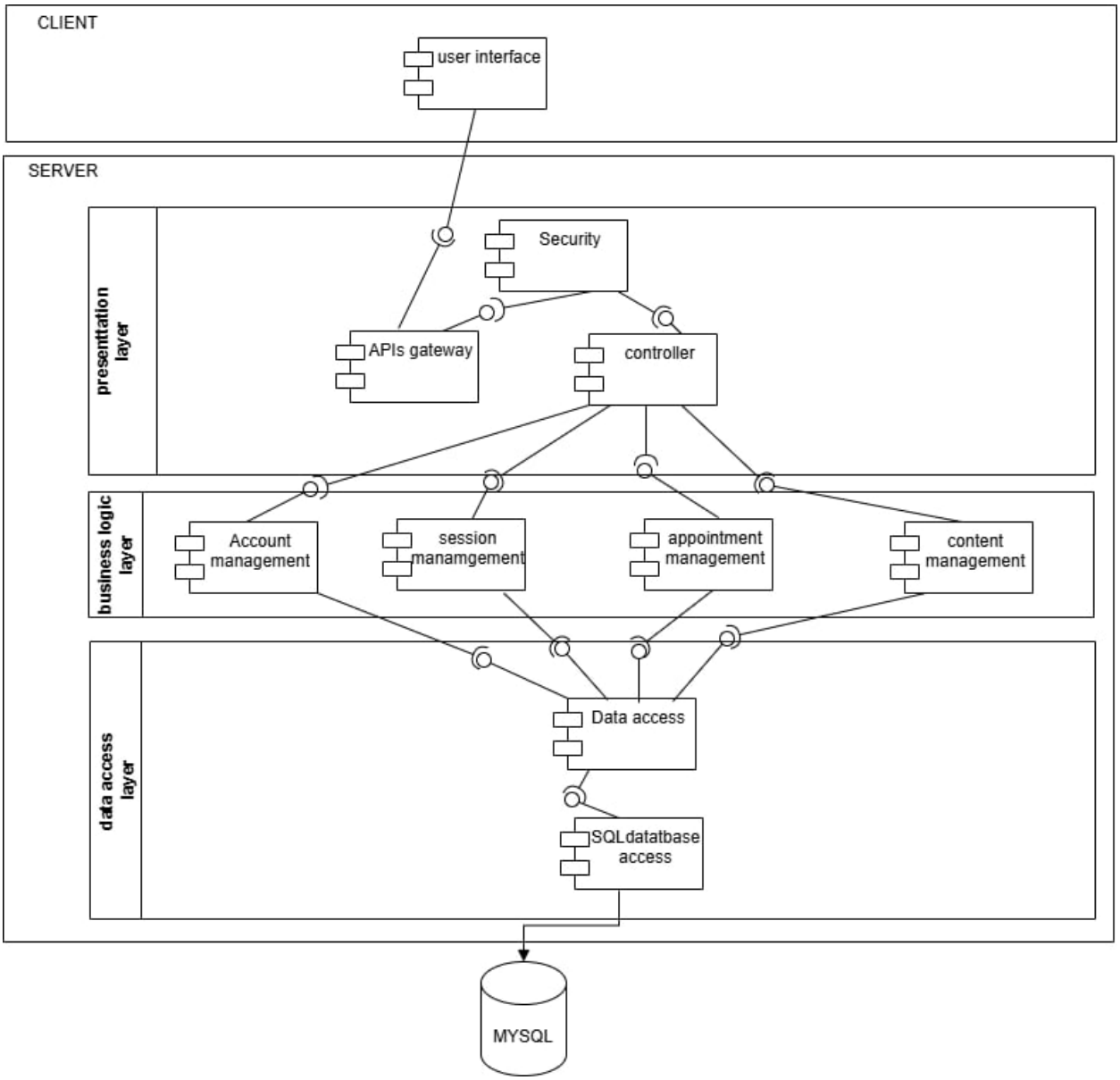
6. **Data Access :**

   Facilitates communication with both SQL and NoSQL databases, ensuring seamless data retrieval and storage while maintaining system integrity.

7. **SQL Database Access:**

   Focuses on accessing structured data, such as patient records, session logs, and transaction history stored in relational tables.

System Architecture:



- We used layered architecture with client-server architecture.

- The client side is the front-end application building with the React framework.

- The server side is the backend application building with Django framework, and the server side will be divided into multi layers for more security and less complexity of the system.

- The layered of the server side:

  Presentation layer: serves as a presentation for the backend APIs.
  Business logic layer: contain the main logic of the system.

- Data access layer: responsible for accessing and communicating with the two databases we have.

Updating the RTM:

Req-id	analysis	System design	Detailed design	coding	Test cases
RE-FR-AM-01	AM-AN	AM-BLL-04	DD-DP	source	TC-04 TC-05 TC-06 TC-07
RE-FR-AM-02	AM-AN	AM-BLL-04	DD-DP	source	TC-01 TC-02 TC-03
RE-FR-AM-03	AM-AN	AM-BLL-04	DD-DP	source	Tc-08
RE-FR-AM-04	AM-AN	AM-BLL-04	DD-DP	source	Tc-09
RE-FR-AM-05	AM-AN	AM-BLL-04	DD-DP	source	Tc-10 Tc-11 Tc-12
RE-FR-AM-06	AM-AN	AM-BLL-04	DD-DP	source	Tc-13 Tc-14
RE-FR-AM-07	AM-AN	AM-BLL-04	DD-DP	source	Tc-15 Tc-16
RE-FR-AM-08	AM-AN	AM-BLL-04	DD-DP	source	Tc-04 Tc-05 Tc-06 Tc-07
RE-FR-AM-09	AM-AN	AM-BLL-04	DD-DP	source	Tc-09
RE-FR-AM-10	AM-AN	AM-BLL-04	DD-DP	source	Tc-17
RE-FR-AM-11	AM-AN	AM-BLL-04	DD-DP	source	Tc-18
RE-FR-AM-12	AM-AN	AM-BLL-04	DD-DP	source	Tc-19 Tc-20

RE-FR-AM-13	AM-AN	AM-BLL-04	DD-DP	source	Tc-21 Tc-22
RE-FR-AM-14	AM-AN	AM-BLL-04	DD-DP	source	Tc-17
RE-FR-AM-15	AM-AN	AM-BLL-04	DD-DP	source	Tc-17
RE-FR-AM-16	AM-AN	AM-BLL-04	DD-DP	source	Tc-18
RE-FR-AM-17	AM-AN	AM-BLL-04	DD-DP	source	Tc-18
RE-FR-PM-18	AM-AN	AM-BLL-04	DD-DP	source	Tc-23
RE-FR-AM-19	AM-AN	AM-BLL-04	DD-DP	source	TC-12 TC-13
RE-FR-AM-20	AM-AN	AM-BLL-04	DD-DP	source	TC-14 TC-15
RE-FR-AM-21	AM-AN	AM-BLL-04	DD-DP	source	TC-61
RE-FR-AM-22	AM-AN	AM-BLL-04	DD-DP	source	TC-48 TC-49
RE-FR-SM-23	AM-AN	AM-BLL-04	DD-DP	source	TC-50 TC-51
RE-FR-SM-24	AM-AN	AM-BLL-04	DD-DP	source	TC-52
RE-FR-SM-25	AM-AN	AM-BLL-04	DD-DP	source	TC-53
RE-FR-APM-26	AM-AN	AM-BLL-04	DD-DP	source	TC-36 TC-37
RE-FR-APM-27	AM-AN	AM-BLL-04	DD-DP	source	TC-54
RE-FR-APM-28	AM-AN	AM-BLL-04	DD-DP	source	TC-55
RE-FR-APM-29	AM-AN	AM-BLL-04	DD-DP	source	TC-56
RE-FR-APM-30	AM-AN	AM-BLL-04	DD-DP	source	TC-38 TC-39 TC-40
RE-FR-APM-31	AM-AN	AM-BLL-04	DD-DP	source	TC-57 TC-58
RE-FR-APM-32	AM-AN	AM-BLL-04	DD-DP	source	TC-41 TC-42

RE-FR-APM-33	AM-AN	AM-BLL-04	DD-DP	source	TC-43 TC-44
RE-FR-APM-34	AM-AN	AM-BLL-04	DD-DP	source	TC-59
RE-FR-APM-35	AM-AN	AM-BLL-04	DD-DP	source	TC-60
RE-FR-APM-36	AM-AN	AM-BLL-04	DD-DP	source	TC-33 TC-34 TC-35
RE-FR-APM-37	AM-AN	AM-BLL-04	DD-DP	source	TC-82
RE-FR-APM-38	AM-AN	AM-BLL-04	DD-DP	source	TC-83
RE-FR-APM-39	AM-AN	AM-BLL-04	DD-DP	source	TC-84
RE-FR-APM-40	AM-AN	AM-BLL-04	DD-DP	source	TC-85
RE-FR-APM-41	AM-AN	AM-BLL-04	DD-DP	source	TC-86
RE-FR-APM-42	AM-AN	AM-BLL-04	DD-DP	source	TC-87
RE-FR-APM-43	AM-AN	AM-BLL-04	DD-DP	source	TC-62
RE-FR-APM-44	AM-AN	AM-BLL-04	DD-DP	source	TC-63
RE-FR-APM-45	AM-AN	AM-BLL-04	DD-DP	source	TC-64
RE-FR-APM-46	AM-AN	AM-BLL-04	DD-DP	source	TC-65
RE-FR-APM-47	AM-AN	AM-BLL-04	DD-DP	source	TC-66
RE-FR-APM-48	AM-AN	AM-BLL-04	DD-DP	source	TC-67
RE-FR-APM-49	AM-AN	AM-BLL-04	DD-DP	source	TC-68
RE-FR-APM-50	AM-AN	AM-BLL-04	DD-DP	source	TC-69
RE-FR-APM-51	AM-AN	AM-BLL-04	DD-DP	source	TC-70
RE-FR-APM-52	AM-AN	AM-BLL-04	DD-DP	source	TC-71
RE-FR-APM-53	AM-AN	AM-BLL-04	DD-DP	source	TC-72
RE-FR-APM-54	AM-AN	AM-BLL-04	DD-DP	source	TC-75
RE-FR-APM-55	AM-AN	AM-BLL-04	DD-DP	source	TC-73

RE-FR-APM-56	AM-AN	AM-BLL-04	DD-DP	source	TC-74
RE-FR-APM-57	AM-AN	AM-BLL-04	DD-DP	source	TC-76
RE-FR-APM-58	AM-AN	AM-BLL-04	DD-DP	source	TC-77
RE-FR-APM-64	AM-AN	AM-BLL-04	DD-DP	source	TC-78
RE-FR-APM-65	AM-AN	AM-BLL-04	DD-DP	source	TC-79
RE-FR-APM-66	AM-AN	AM-BLL-04	DD-DP	source	TC-80

Chapter5

Practical Implementation

1. Introduction:

In this chapter, we will present the practical implementation of the system, detailing the technologies and tools employed. Additionally, we will showcase the system interfaces and conclude by executing test cases to verify the system's functionalities across different scenarios.

2. Used Tools:

- Django

  Is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source. Especially we use DRF: Django REST Framework is a widely used, full-featured API framework designed for building RESTful APIs with Django. At its core, DRF integrates with Django's core features “models, views, and URLs” making it simple and seamless to create a RESTful API.

❖ React

React is a popular JavaScript library for building user interfaces. It was created by Facebook and is widely used in web development. React allows developers to build reusable UI components that can efficiently update and render changes to the user interface when the underlying data changes.

Reacts primary focus is on building user interfaces, and it excels in creating interactive and dynamic web applications.

❖ Insomnia

Insomnia is an open-source desktop application that takes the pain out of interacting with and designing, debugging, and testing APIs, we use it to test our APIs.

❖ My SQL Database

MySQL is an open-source relational database management system (RDBMS) that is widely used for storing, managing, and retrieving data. It is one of the most popular and widely adopted databases in the world, known for its reliability, scalability, and ease of use.

❖ MongoDB

MongoDB is a NoSQL database that stores data in flexible, JSON-like documents, making it ideal for handling unstructured data and large volumes of information. It offers high scalability and performance, enabling efficient data management for various applications.

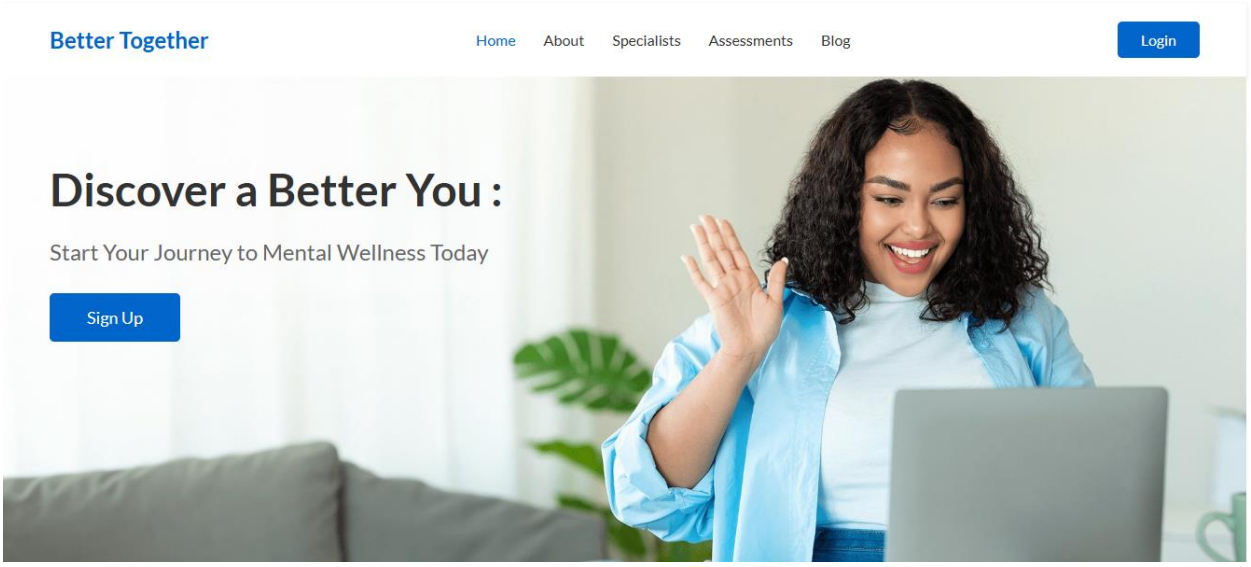
❖ Visual studio code (VS code):

Visual Studio Code combines the simplicity of a source code editor with powerful developer tooling, like IntelliSense code completion and debugging.
We use it to develop the whole project (frontend, backend).

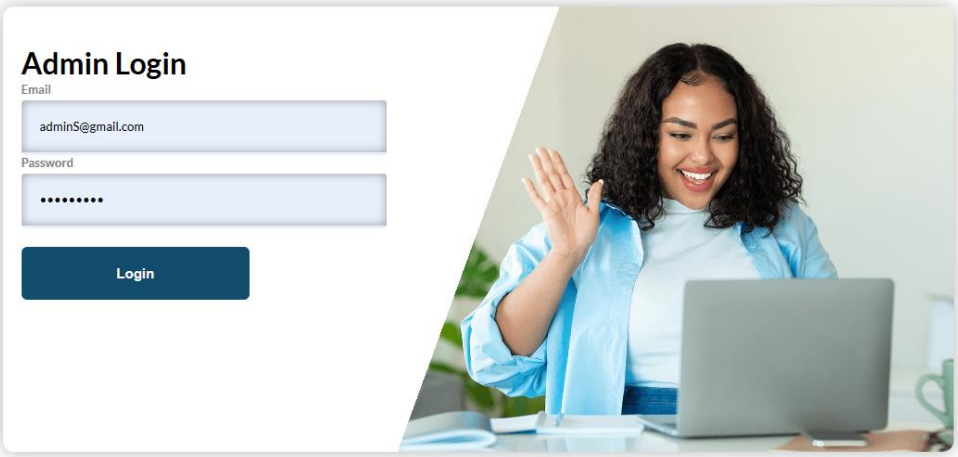
❖ Bitbucket:

Bitbucket is a Git-based source code repository hosting service that facilitates version control and collaborative development. It supports continuous integration and deployment, providing tools for managing code changes and streamlining the software development process "The project repository".

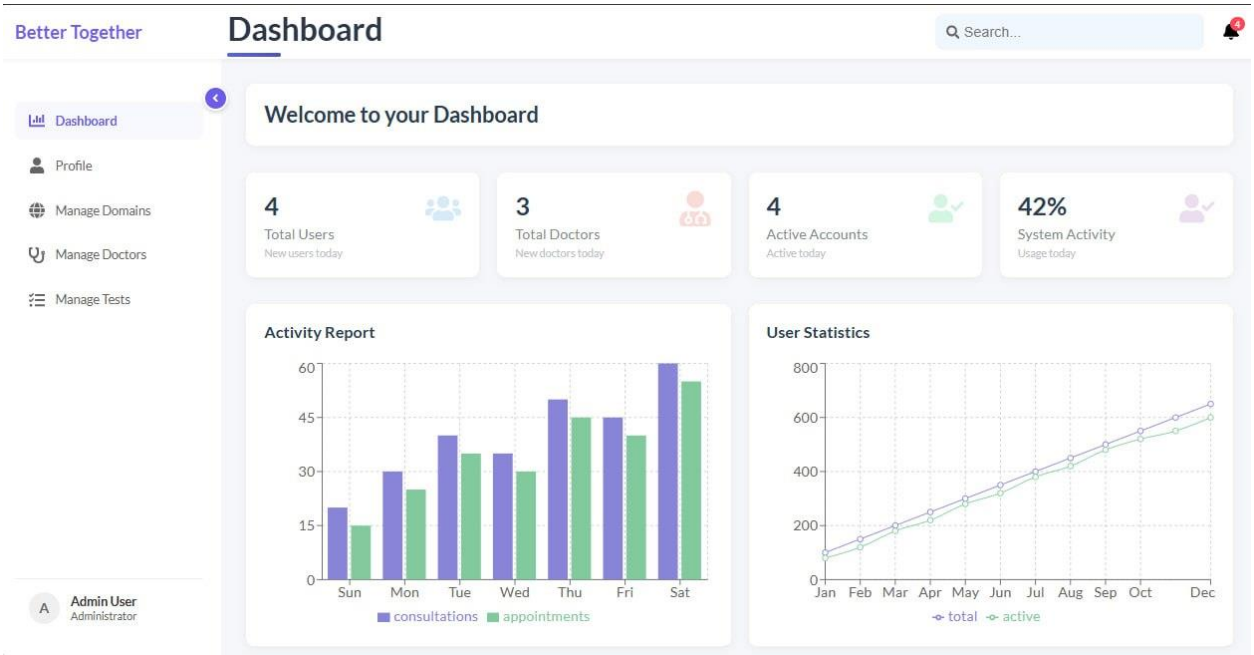
3. System Interface:



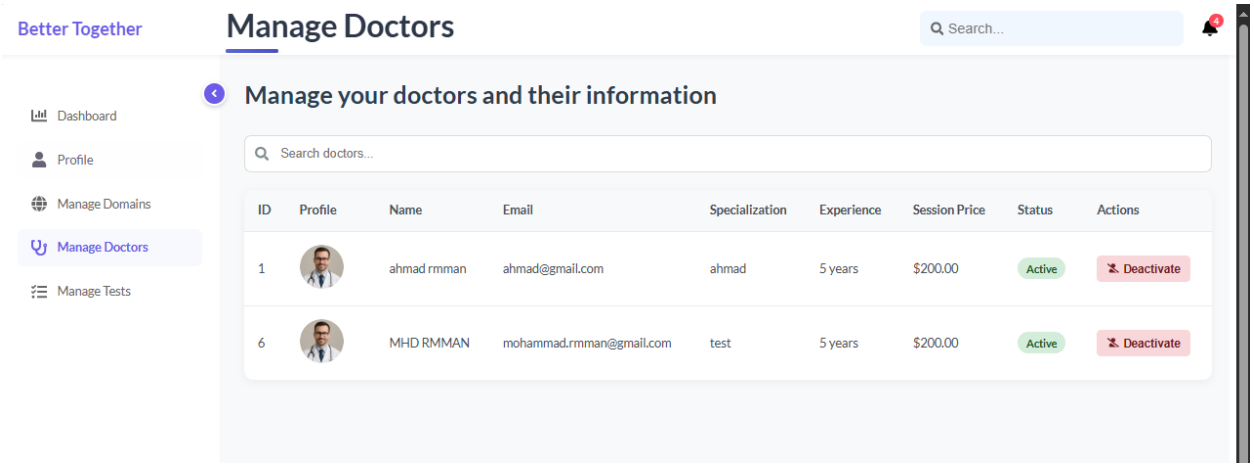
- ## Sign-up interface (Admin) :



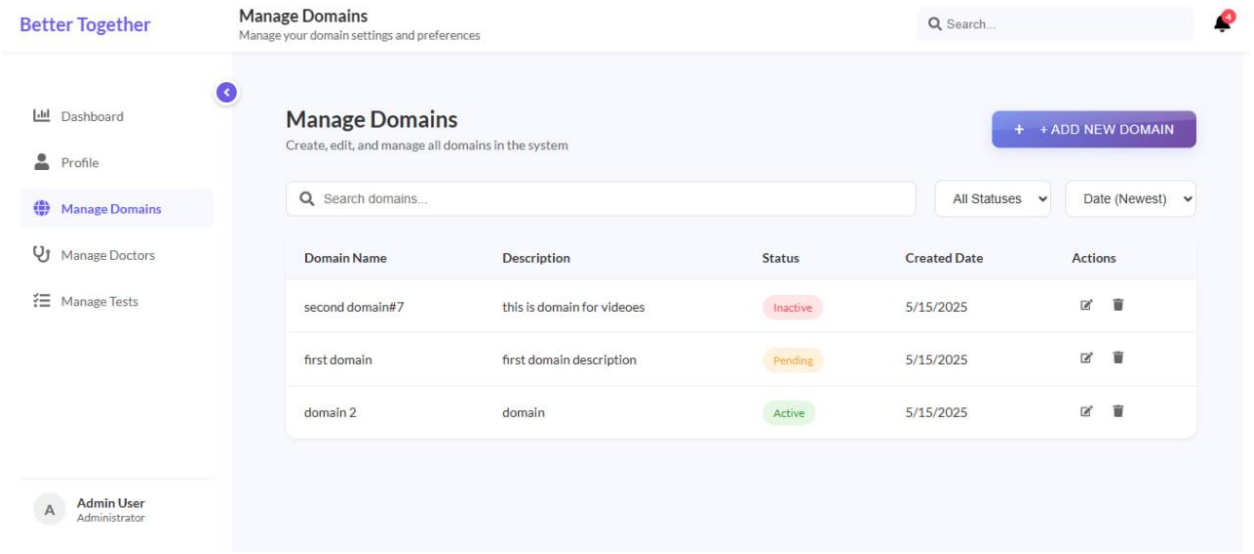
Dashboard admin:

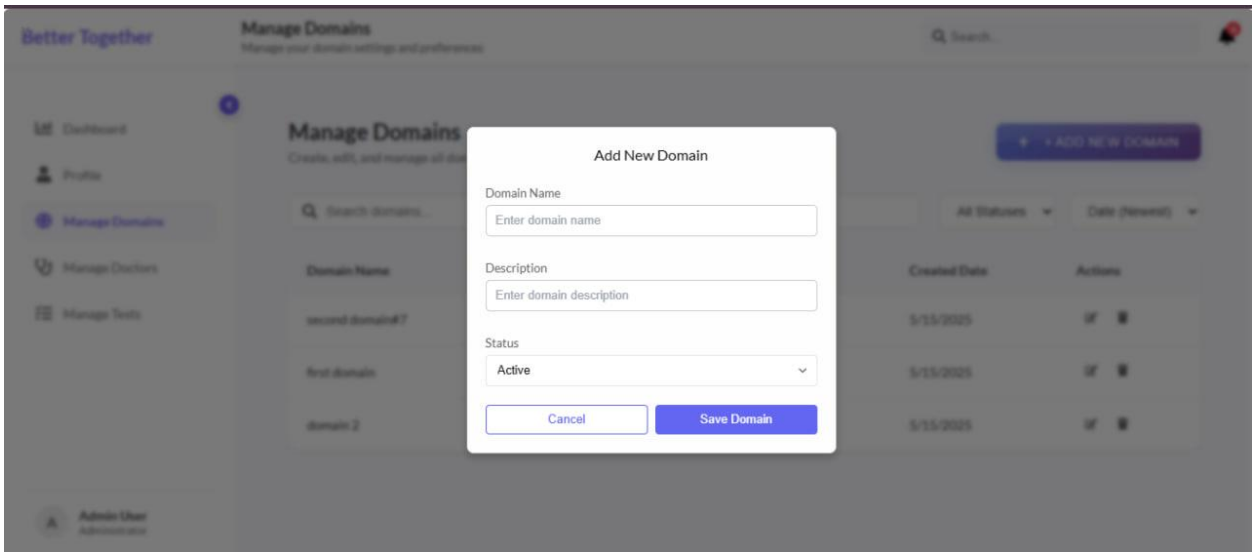


Manage Doctors:

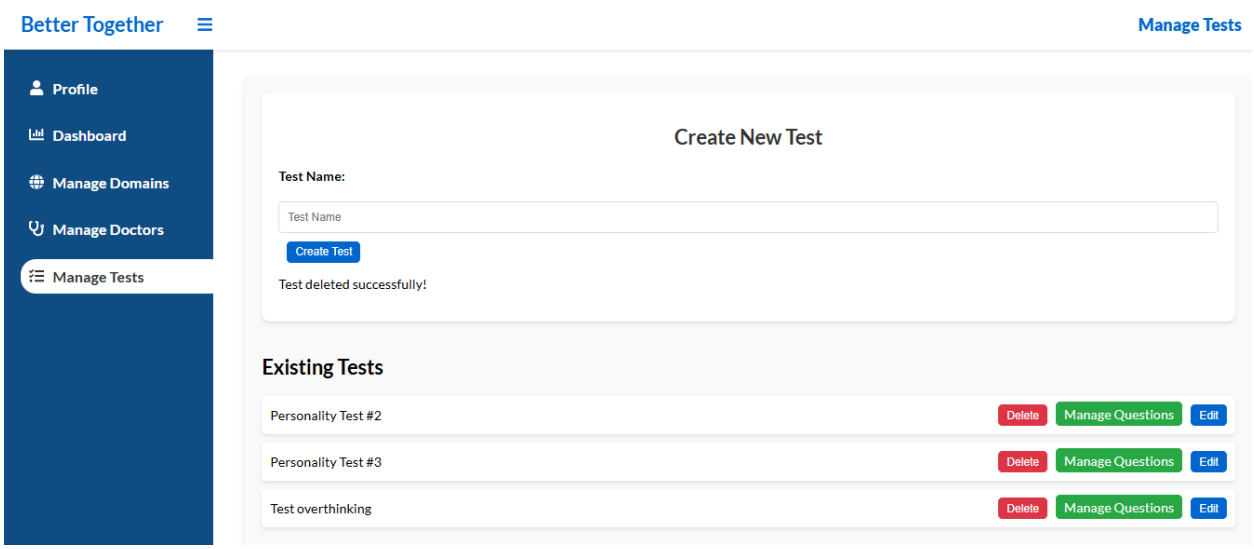


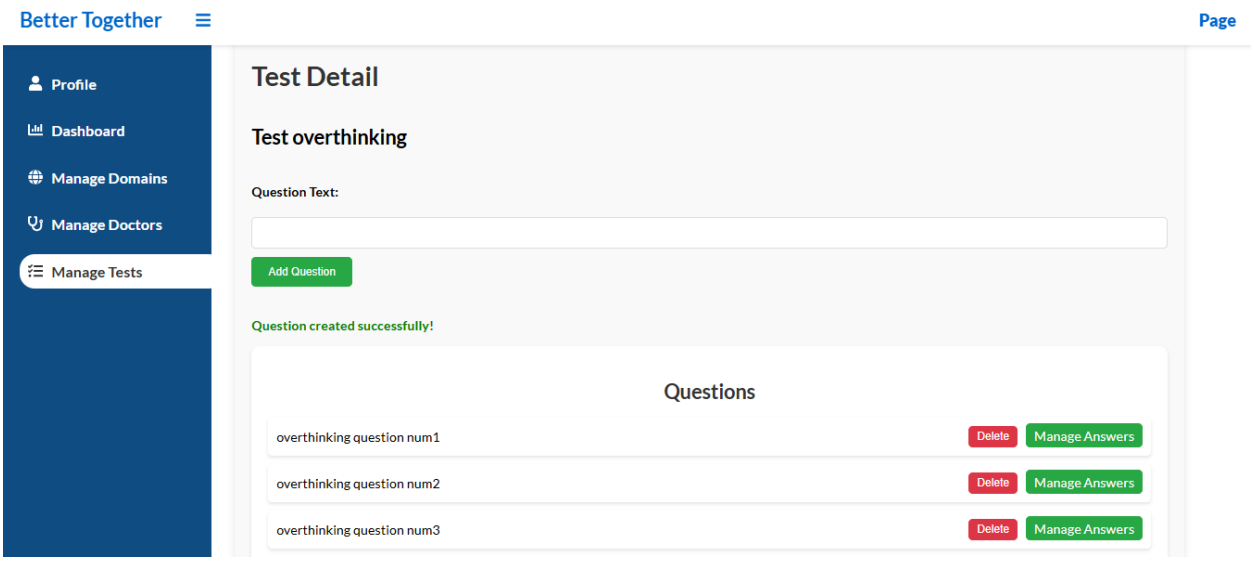
- ## Domain interface (Admin):

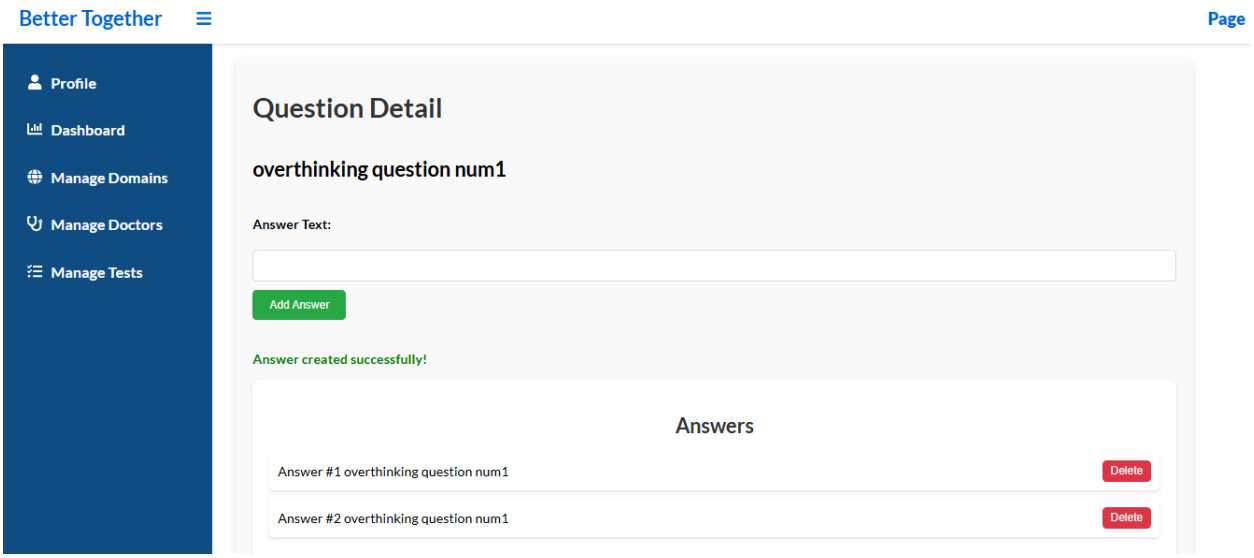




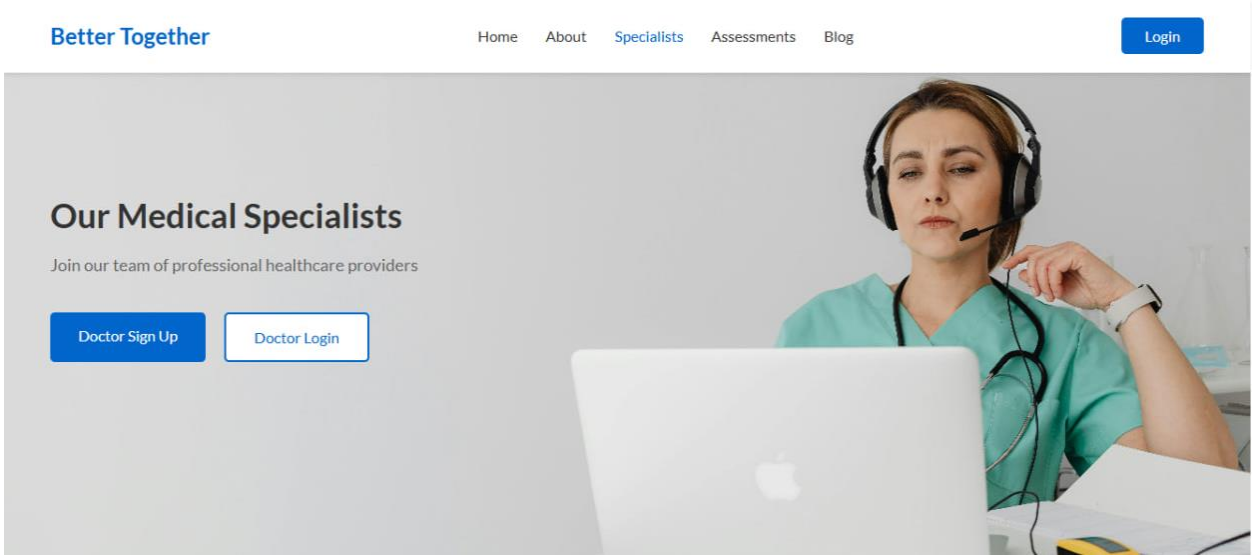
- *Manage Tests interface (Admin):*



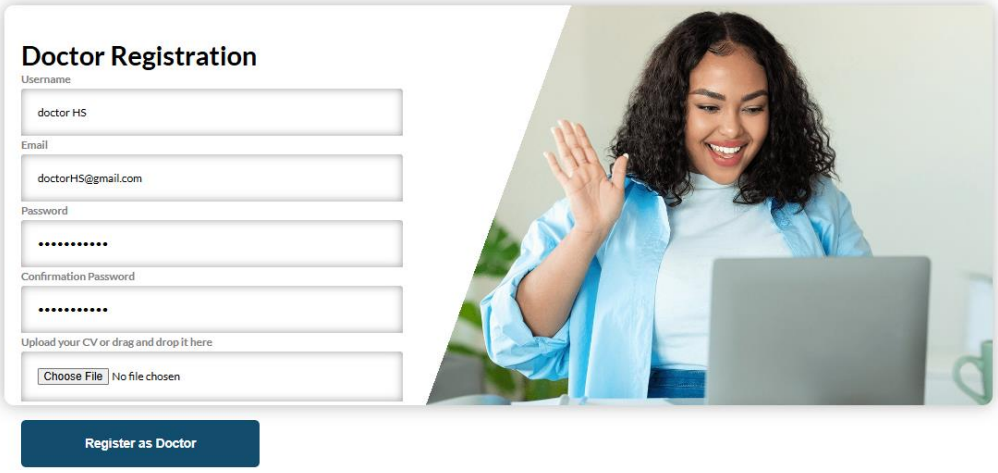

Better Together
Page
Profile
Dashboard
Manage Domains
Manage Doctors
Manage Tests
Test Detail
Test overthinking
Question Text:
Add Question
Question created successfully!
Questions
overthinking question num1
Delete
Manage Answers
overthinking question num2
Delete
Manage Answers
overthinking question num3
Delete
Manage Answers


Better Together
Page
Profile
Dashboard
Manage Domains
Manage Doctors
Manage Tests
Question Detail
overthinking question num1
Answer Text:
Add Answer
Answer created successfully!
Answers
Answer #1 overthinking question num1
Delete
Answer #2 overthinking question num1
Delete

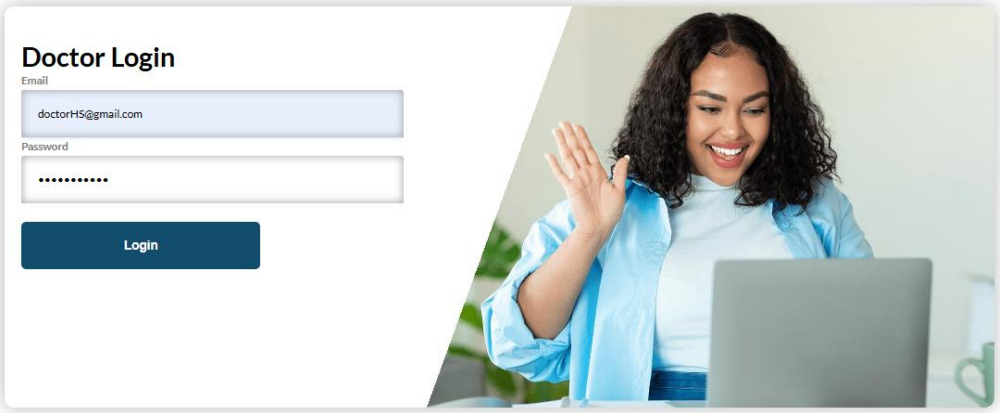
- *Specialists Interface (Doctor):*



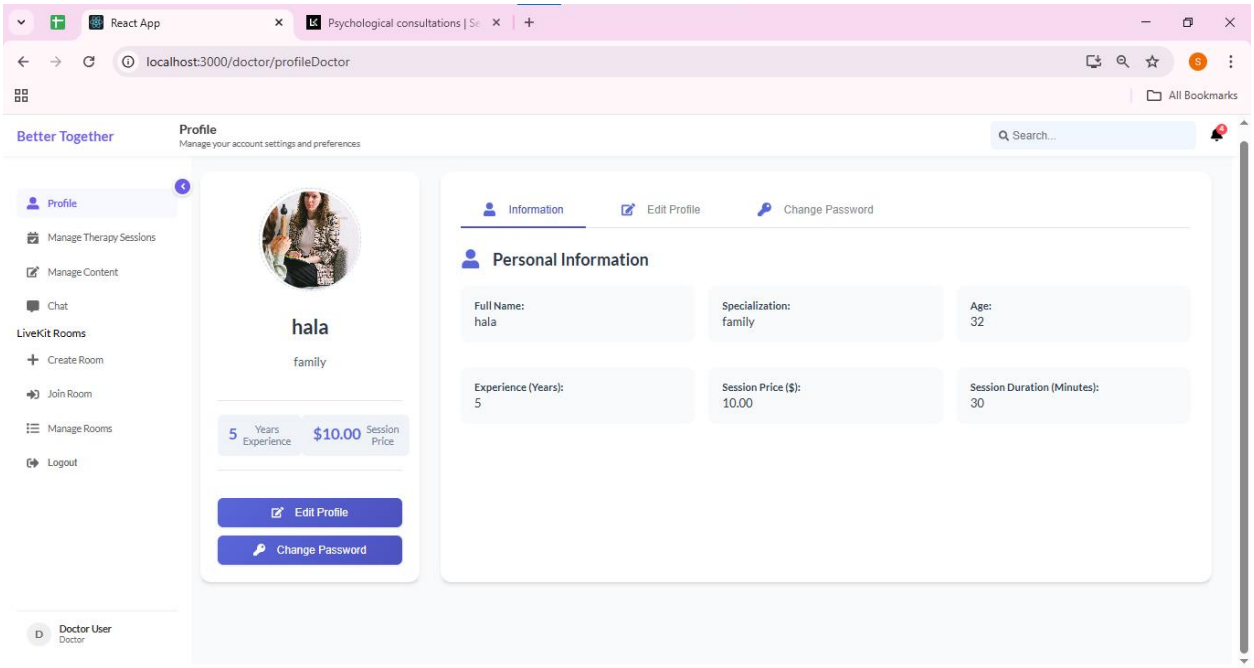
- *Sign-up interface (Doctor):*

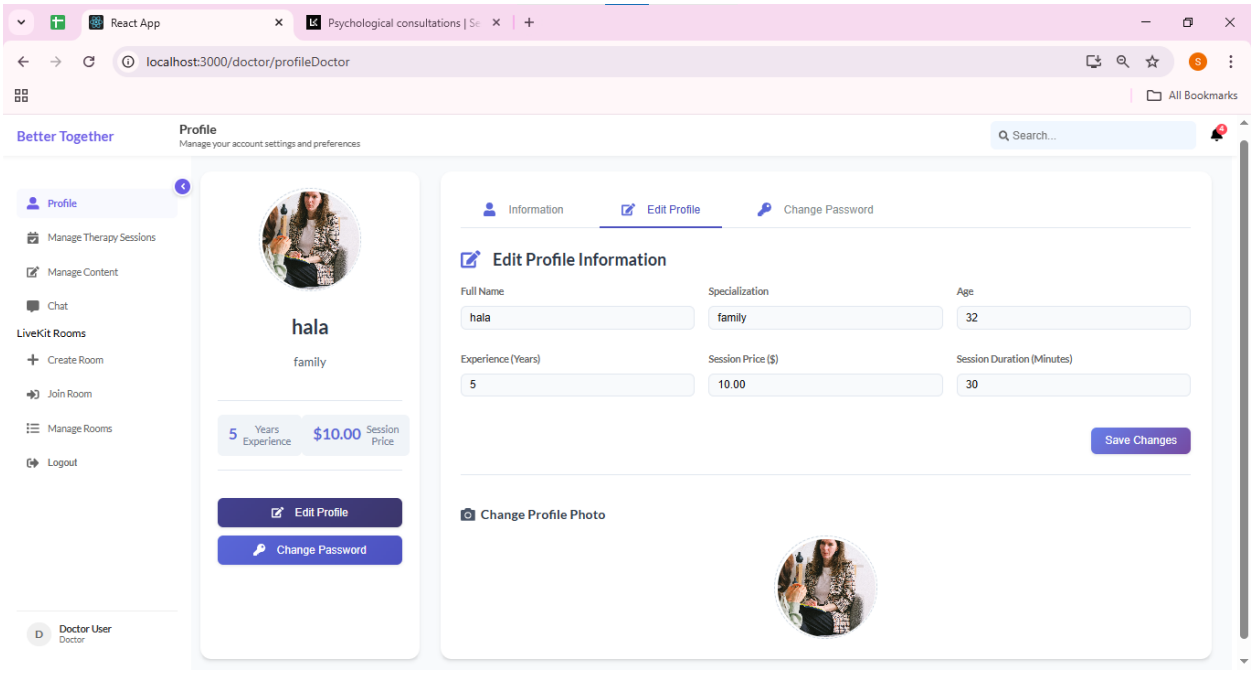


- *Login interface (Doctor):*

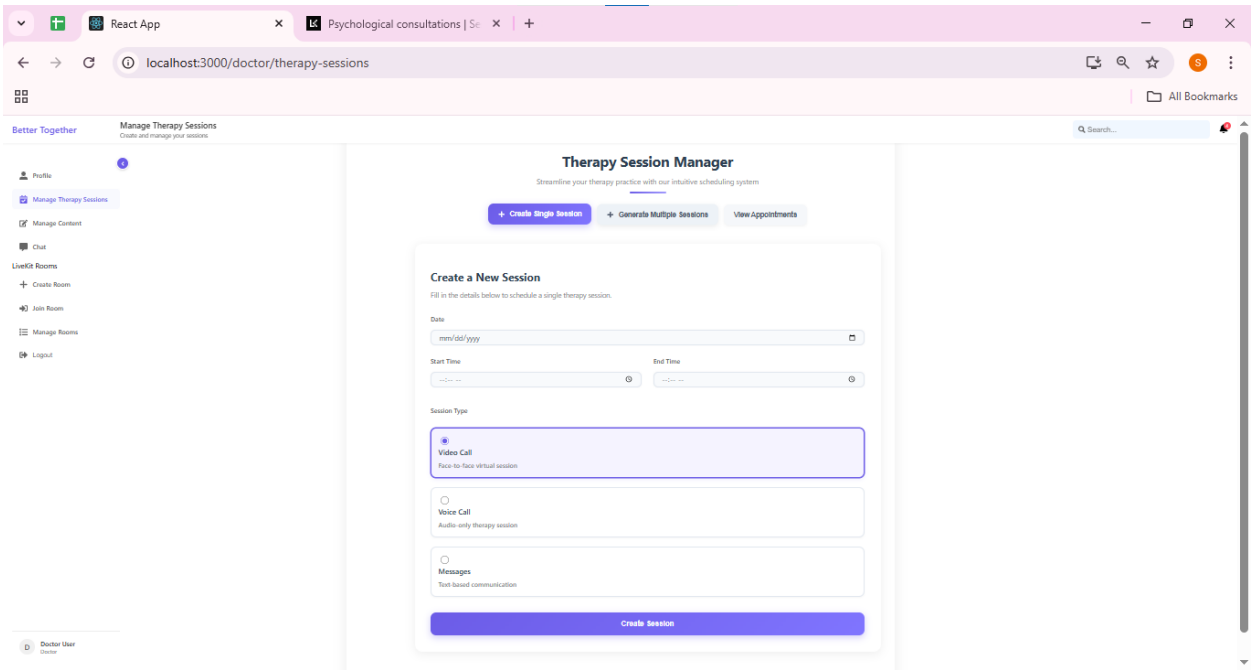


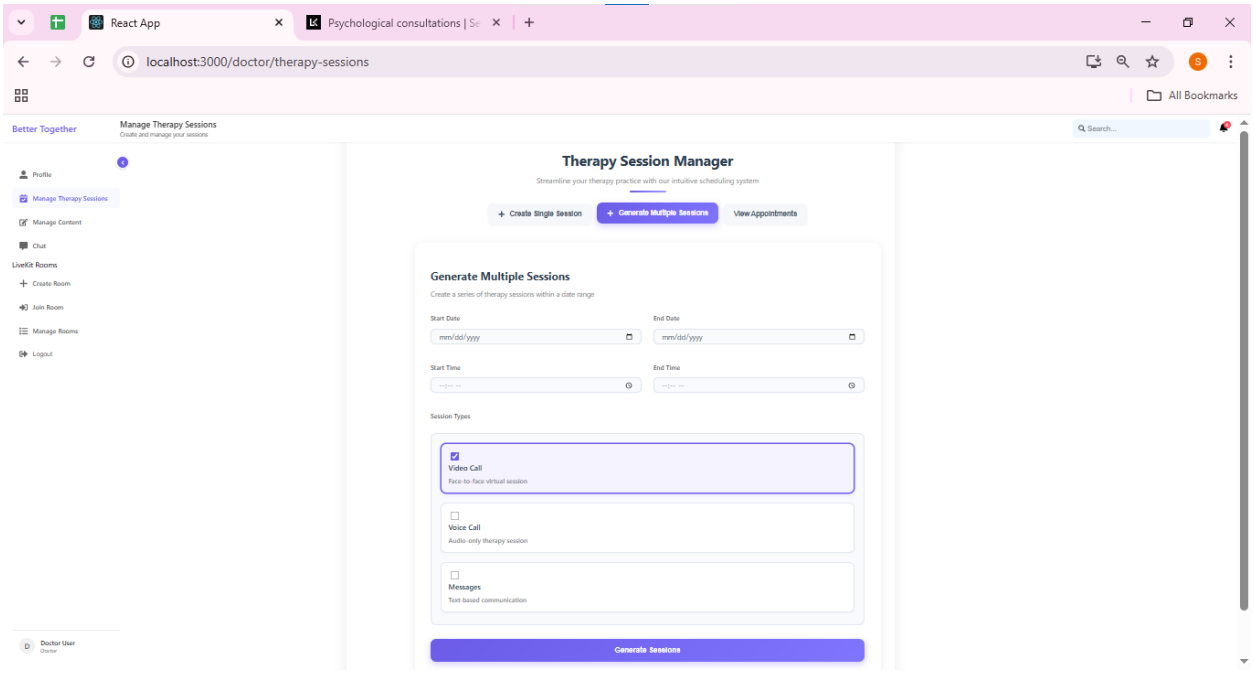
Profile doctor :

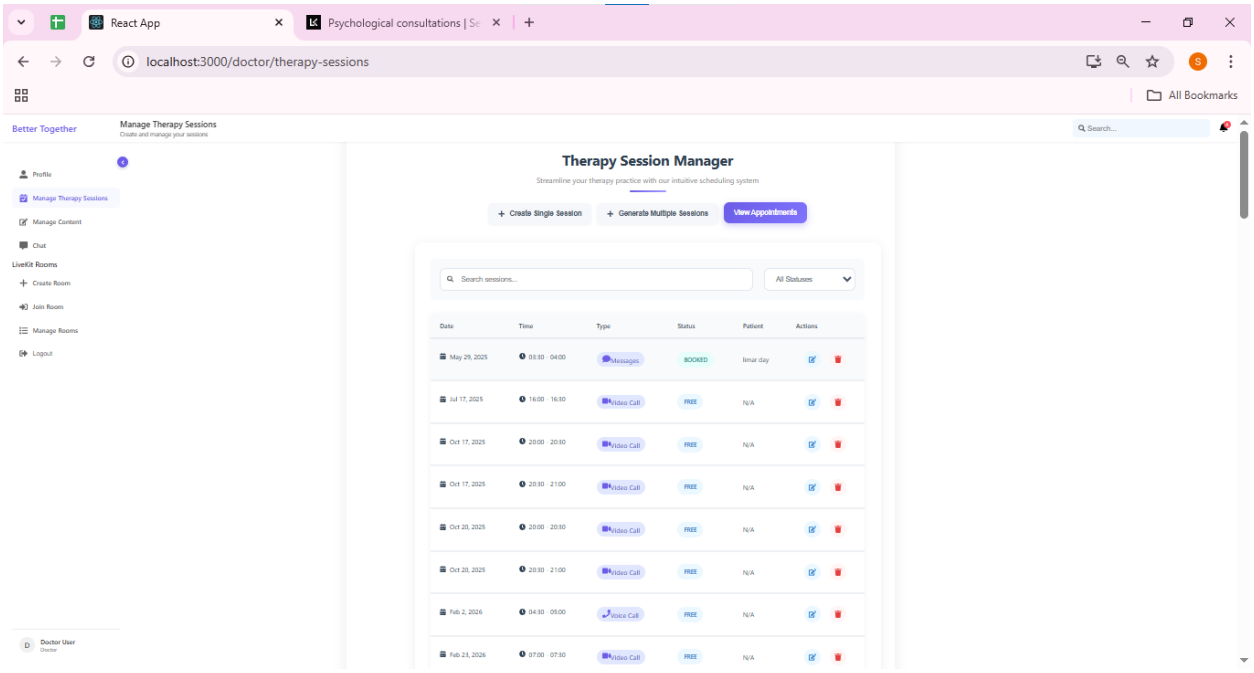


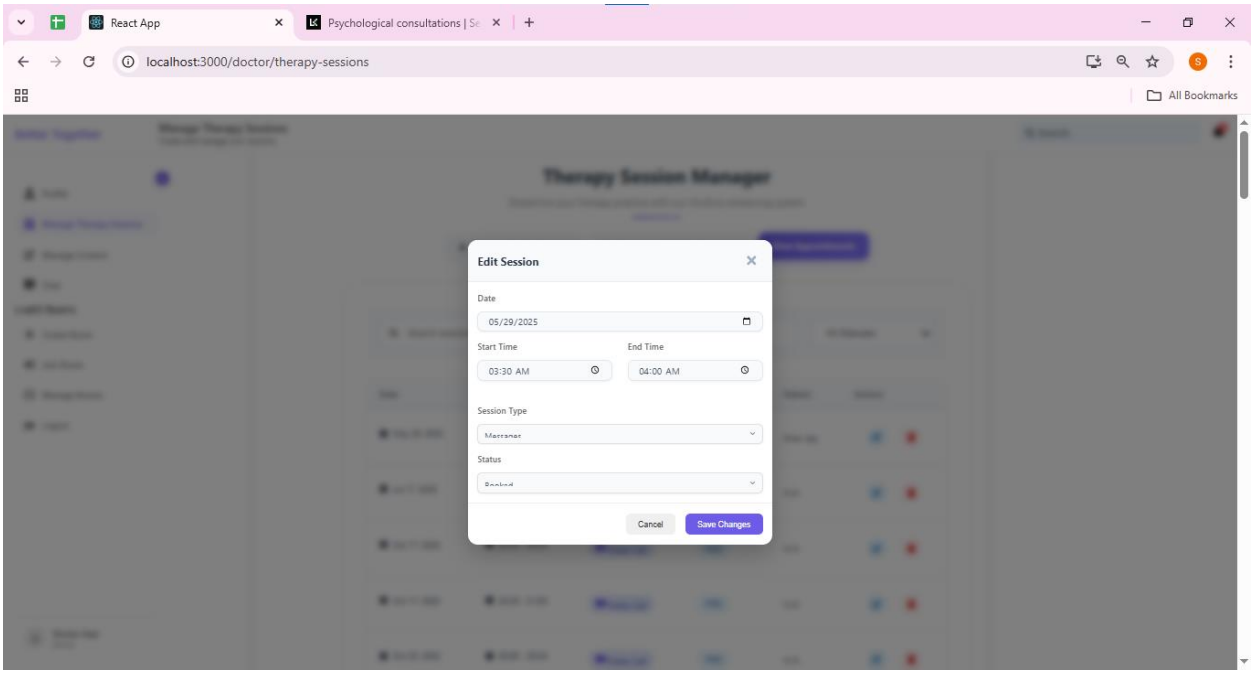


Therapy Session Manager:

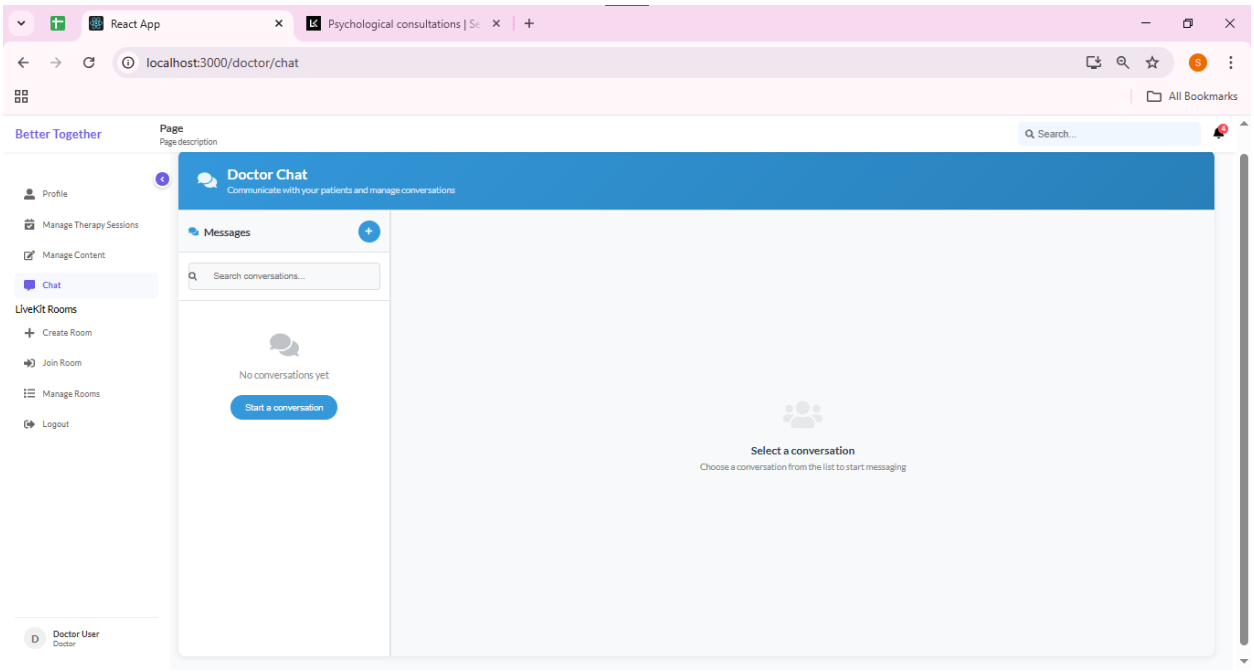



React App
Psychological consultations | Se
localhost:3000/doctor/therapy-sessions
All Bookmarks
Better Together
Manage Therapy Sessions
Create and manage your sessions
Profile
Manage Therapy Sessions
Manage Content
Chat
LiveKit Rooms
Create Room
Join Room
Manage Rooms
Logout
Doctor User
Doctor
Therapy Session Manager
Streamline your therapy practice with our intuitive scheduling system
+ Create Single Session
+ Generate Multiple Sessions
View Appointments
Generate Multiple Sessions
Create a series of therapy sessions within a date range
Start Date
mm/dd/yyyy
End Date
mm/dd/yyyy
Start Time
End Time
Session Types
Video Call
Face-to-face virtual session
Voice Call
Audio-only therapy session
Messages
Text-based communication
Generate Sessions

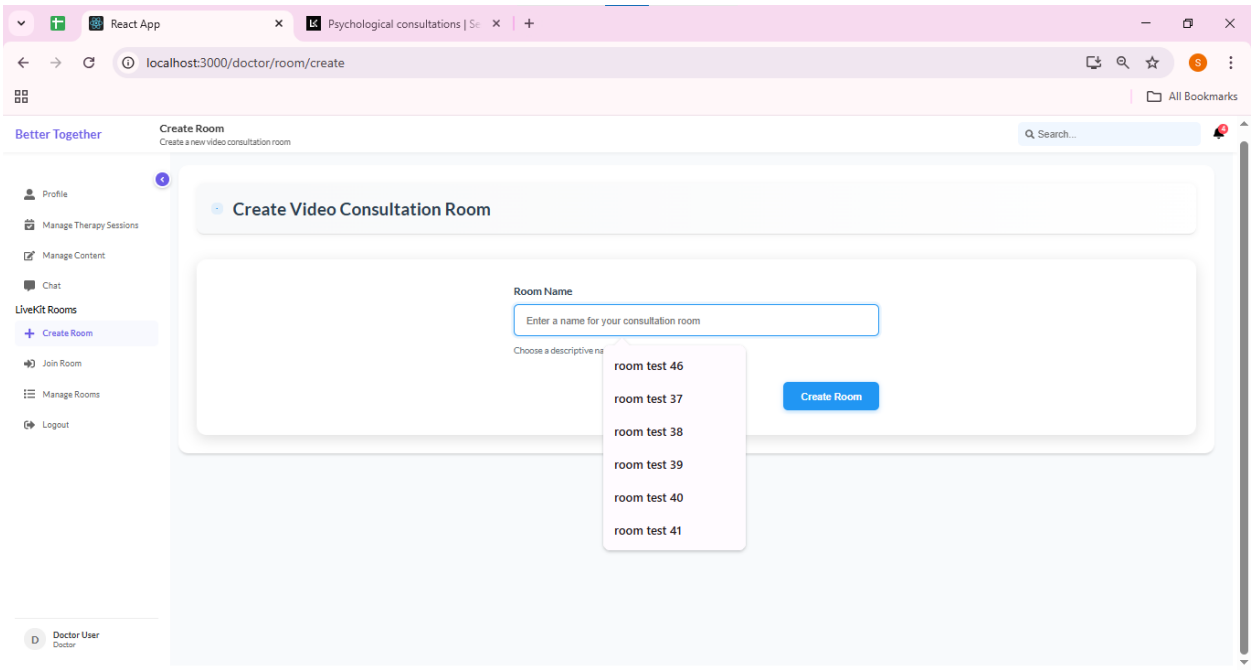

React App
Psychological consultations | Se
localhost:3000/doctor/therapy-sessions
All Bookmarks
Better Together
Manage Therapy Sessions
Create and manage your sessions
Profile
Manage Therapy Sessions
Manage Content
Chat
LiveKit Rooms
Create Room
Join Room
Manage Rooms
Logout
Doctor User
Doctor
Therapy Session Manager
Streamline your therapy practice with our intuitive scheduling system
+ Create Single Session
+ Generate Multiple Sessions
View Appointments
Search sessions...
All Statuses
Date
Time
Type
Status
Patient
Actions

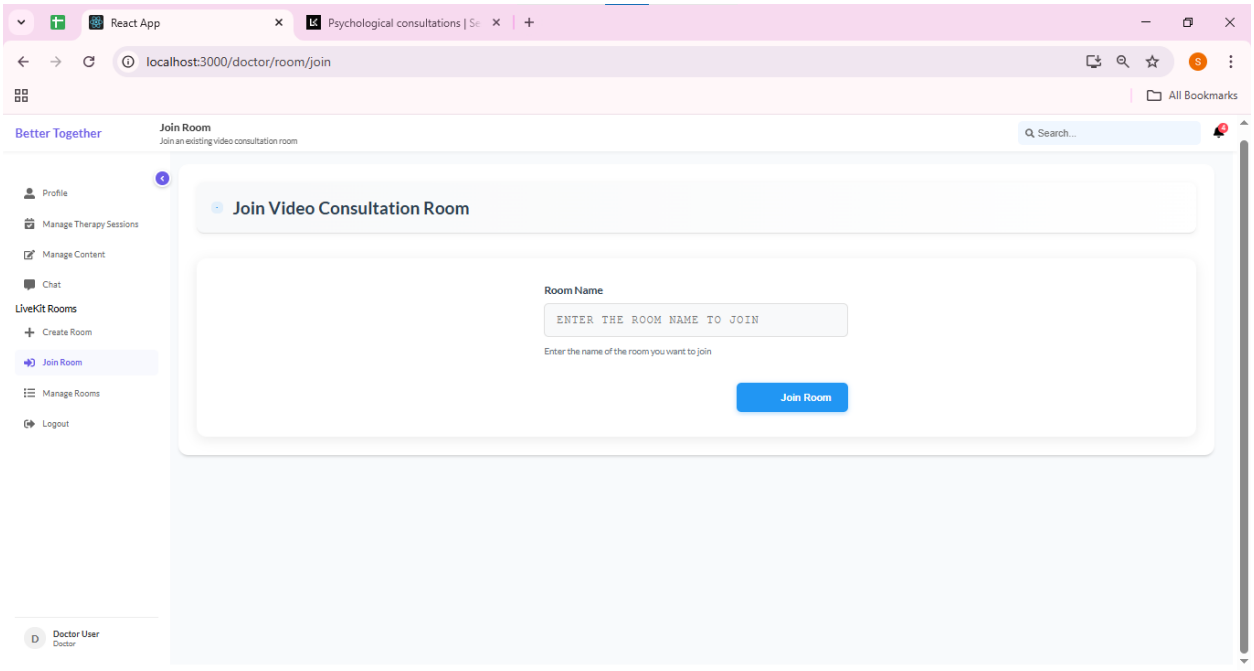

React App
Psychological consultations | Se
localhost:3000/doctor/therapy-sessions
All Bookmarks
Therapy Session Manager
Edit Session
Date
05/29/2025
Start Time
03:30 AM
End Time
04:00 AM
Session Type
Status
Cancel
Save Changes

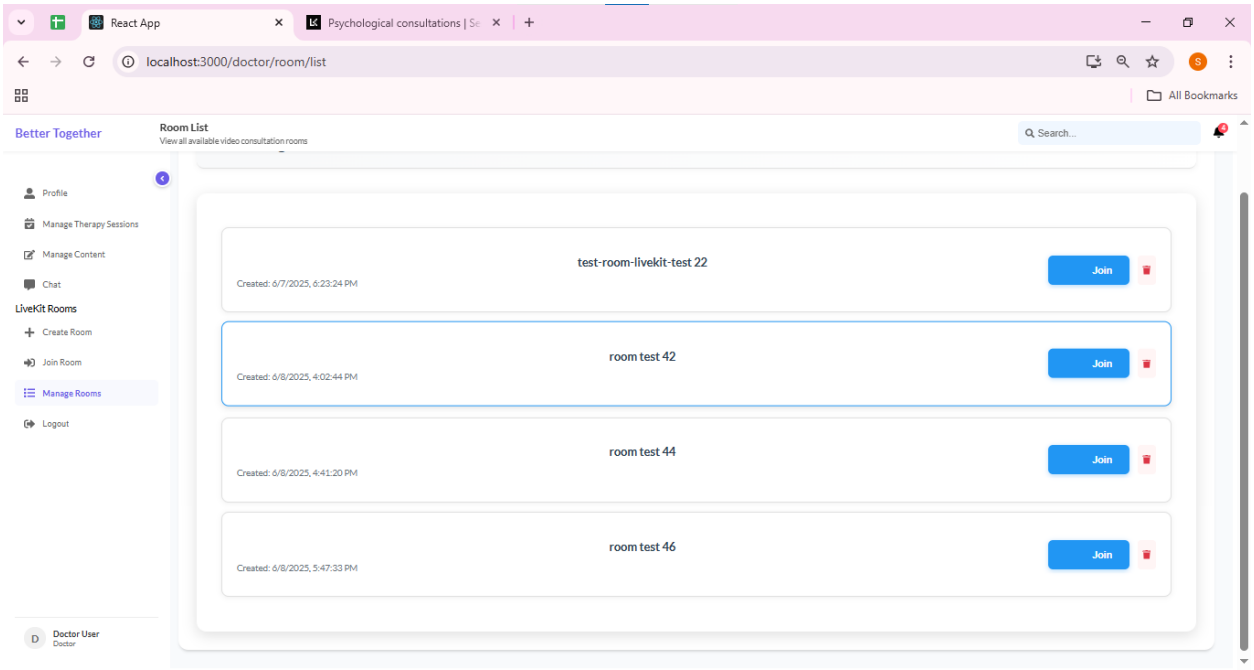
Chat doctor

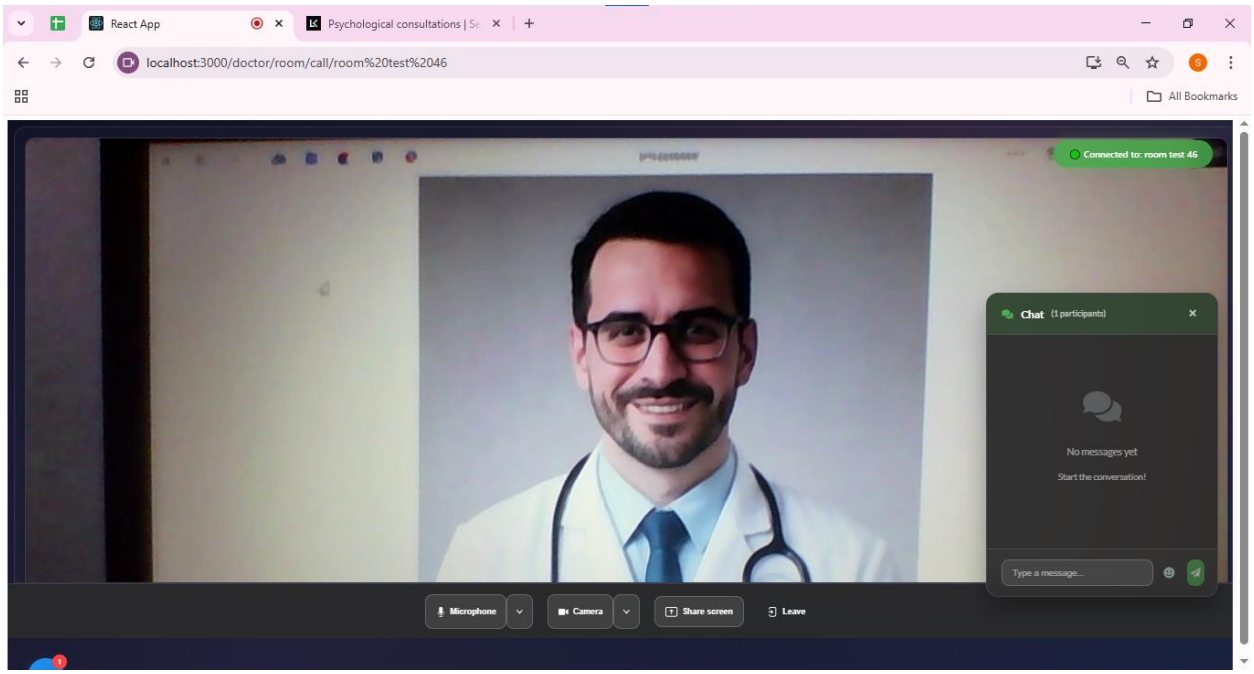

React App
Psychological consultations | Se
localhost:3000/doctor/chat
All Bookmarks
Better Together
Page
Page description
Search...
Profile
Manage Therapy Sessions
Manage Content
Chat
LiveKit Rooms
Create Room
Join Room
Manage Rooms
Logout
Doctor User
Doctor
Doctor Chat
Communicate with your patients and manage conversations
Messages
Search conversations...
No conversations yet
Start a conversation
Select a conversation
Choose a conversation from the list to start messaging

Room doctor:

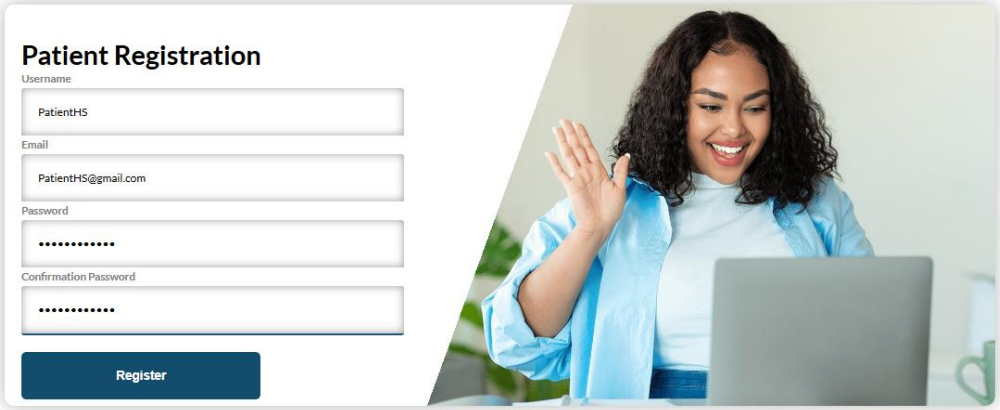

React App
Psychological consultations | Se
localhost:3000/doctor/room/create
All Bookmarks
Better Together
Create Room
Create a new video consultation room
Search...
Profile
Manage Therapy Sessions
Manage Content
Chat
LiveKit Rooms
Create Room
Join Room
Manage Rooms
Logout
Doctor User
Doctor
Create Video Consultation Room
Room Name
Enter a name for your consultation room
Choose a descriptive na
room test 46
room test 37
room test 38
room test 39
room test 40
room test 41
Create Room


React App
Psychological consultations | Se
localhost:3000/doctor/room/join
All Bookmarks
Better Together
Join Room
Join an existing video consultation room
Search...
Profile
Manage Therapy Sessions
Manage Content
Chat
LiveKit Rooms
Create Room
Join Room
Manage Rooms
Logout
Doctor User
Doctor
Join Video Consultation Room
Room Name
ENTER THE ROOM NAME TO JOIN
Enter the name of the room you want to join
Join Room

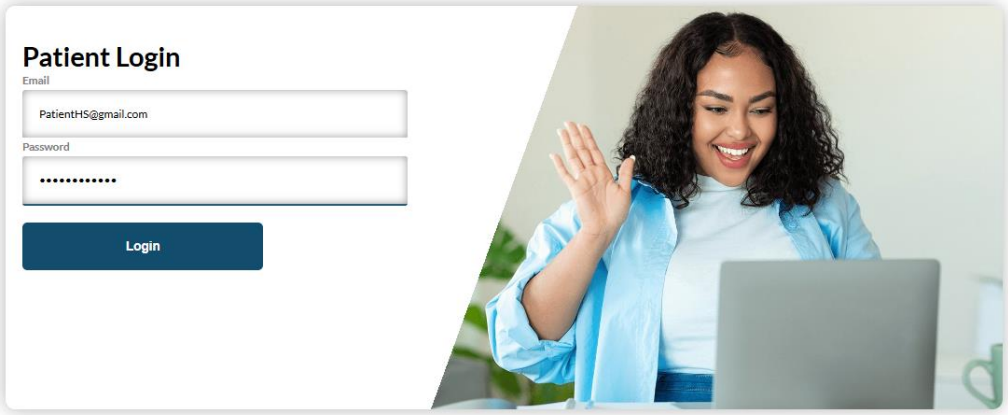

React App
Psychological consultations | Se
localhost:3000/doctor/room/list
All Bookmarks
Better Together
Room List
View all available video consultation rooms
Search...
Profile
Manage Therapy Sessions
Manage Content
Chat
LiveKit Rooms
Create Room
Join Room
Manage Rooms
Logout
Doctor User
Doctor
test-room-livekit-test 22
Created: 6/7/2025, 6:23:24 PM
Join
room test 42
Created: 6/8/2025, 4:02:44 PM
Join
room test 44
Created: 6/8/2025, 4:41:20 PM
Join
room test 46
Created: 6/8/2025, 5:47:33 PM
Join


React App
Psychological consultations | Se
localhost:3000/doctor/room/call/room%20test%2046
All Bookmarks
Connected to: room test 46
Chat (1 participants)
No messages yet
Start the conversation!
Type a message...
Microphone
Camera
Share screen
Leave

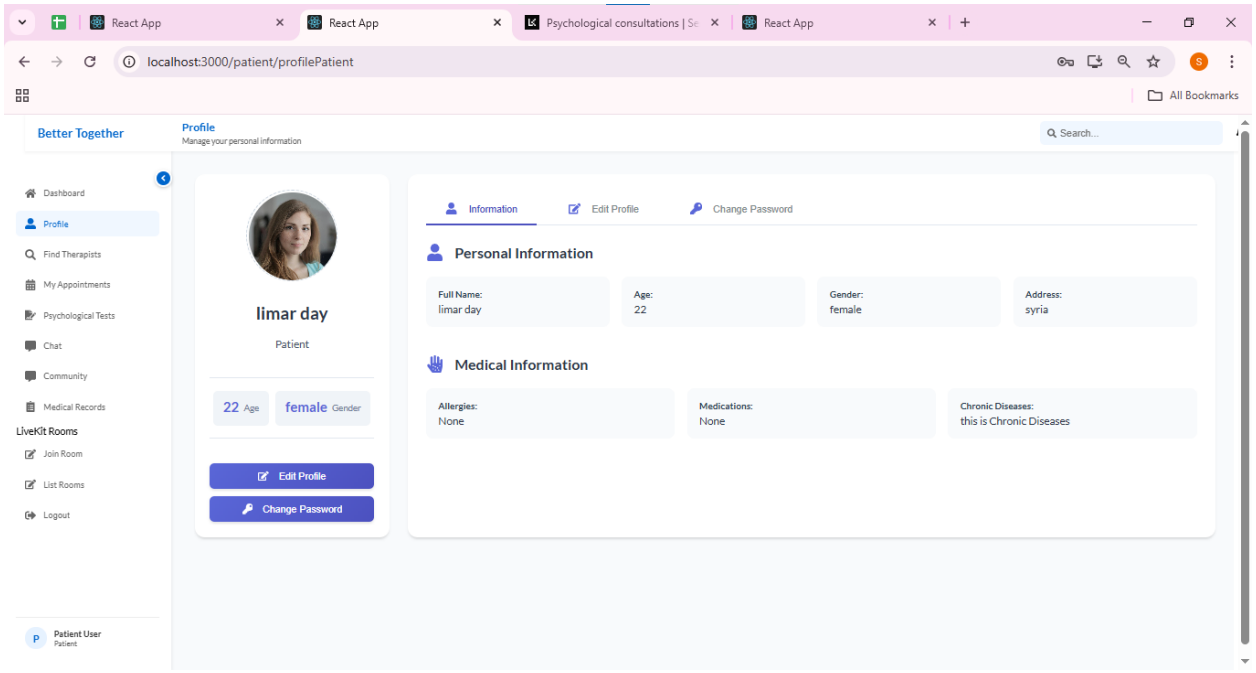
- *Patient Sign-up interface (Patient):*



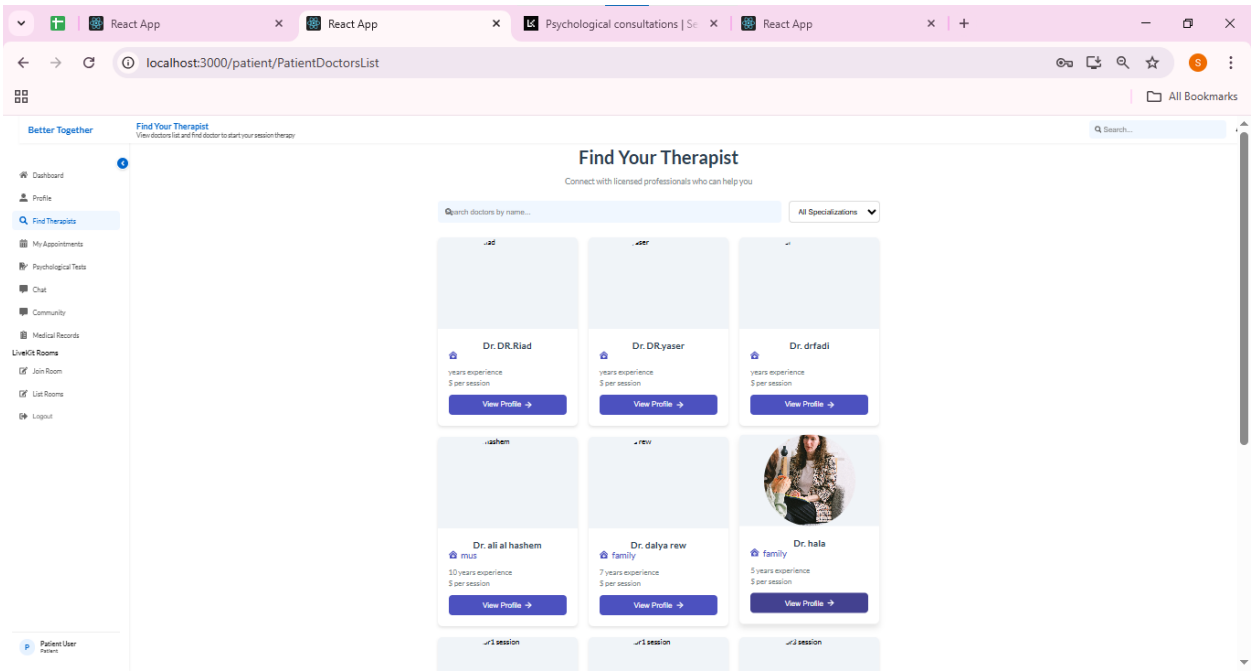
- *Login interface (Patient):*

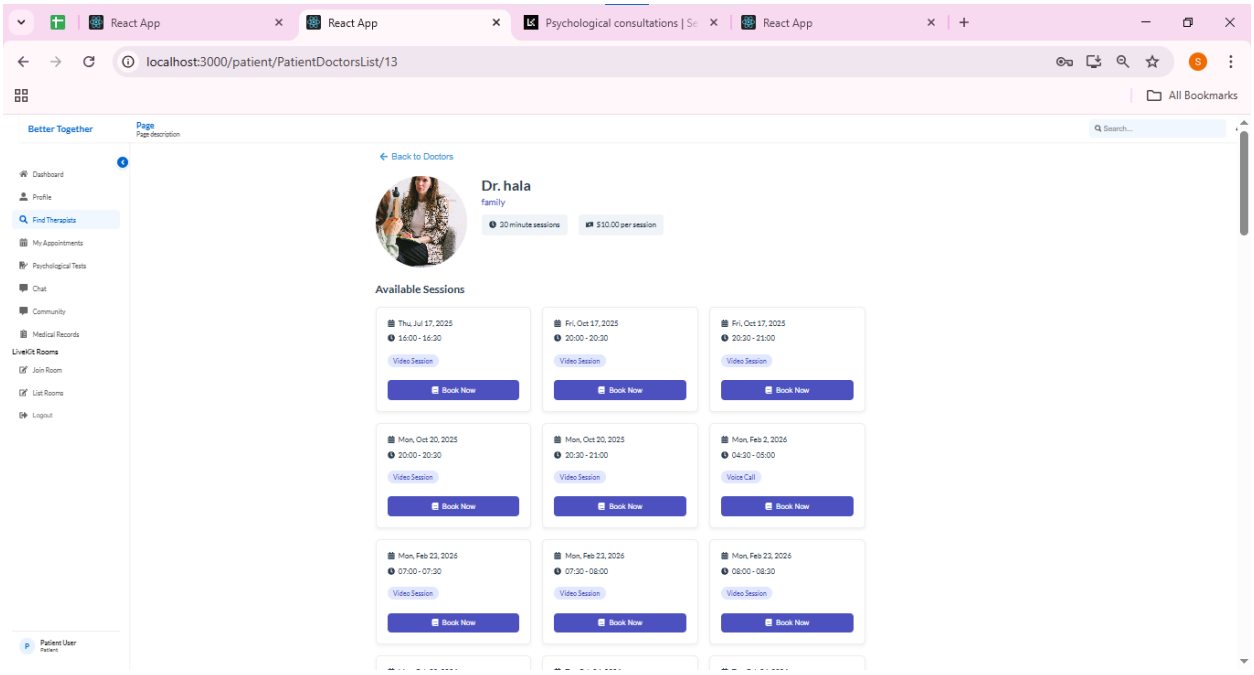


Profile patient :

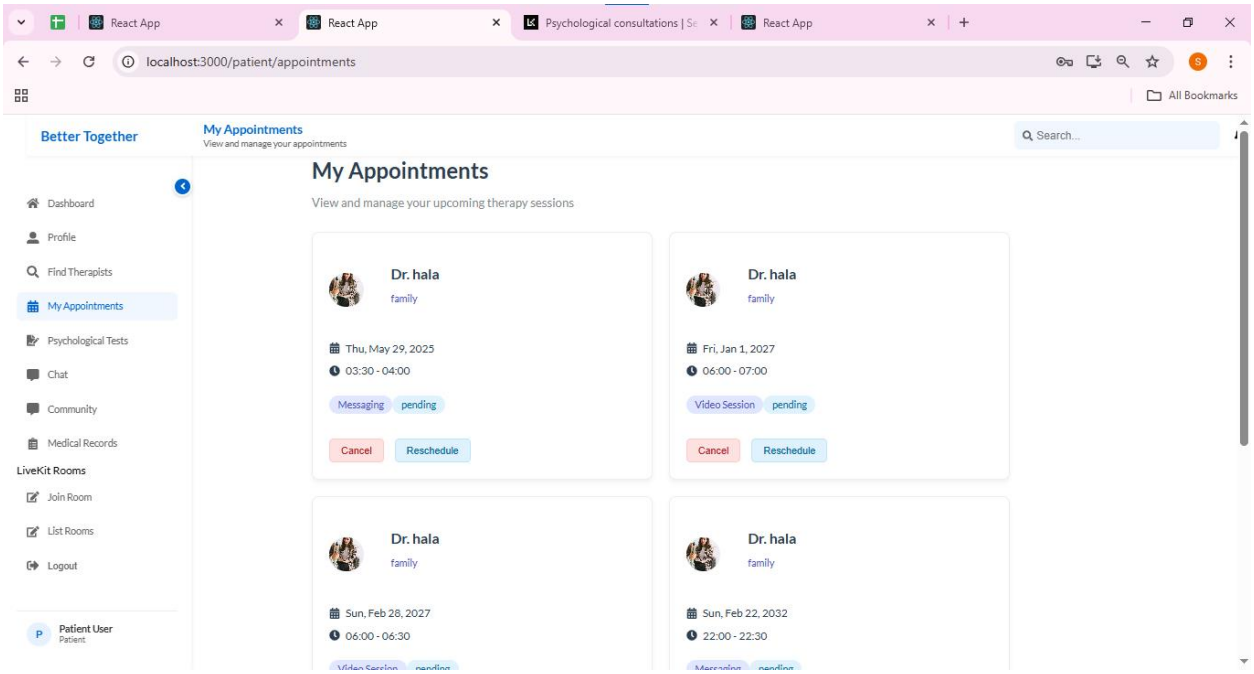


Find Therapy :

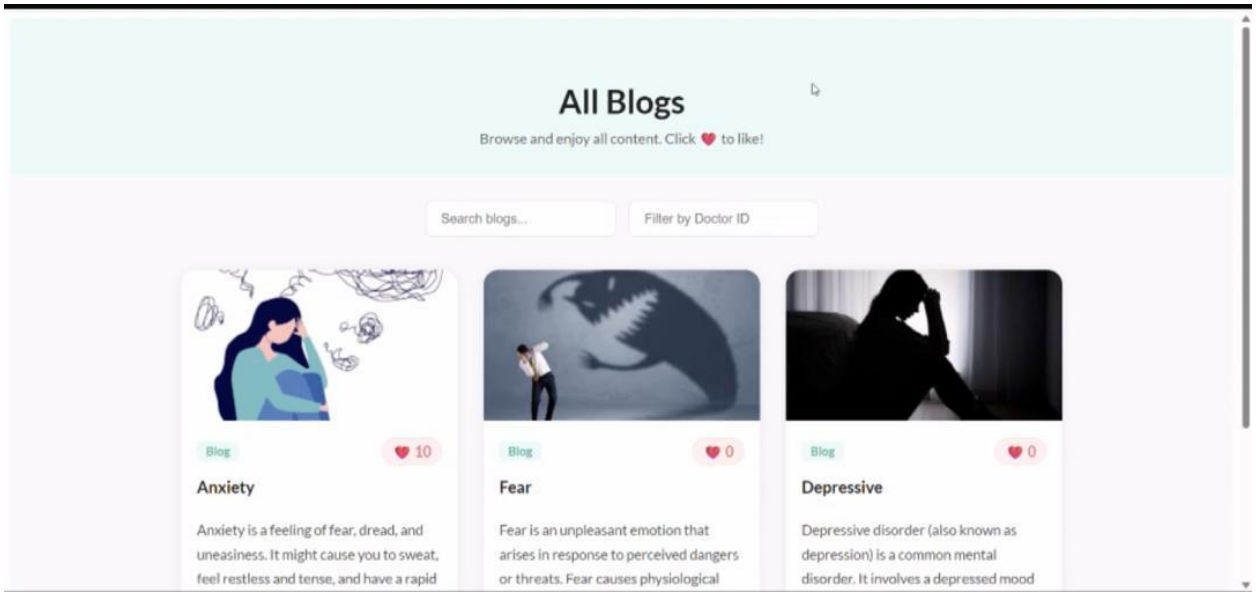




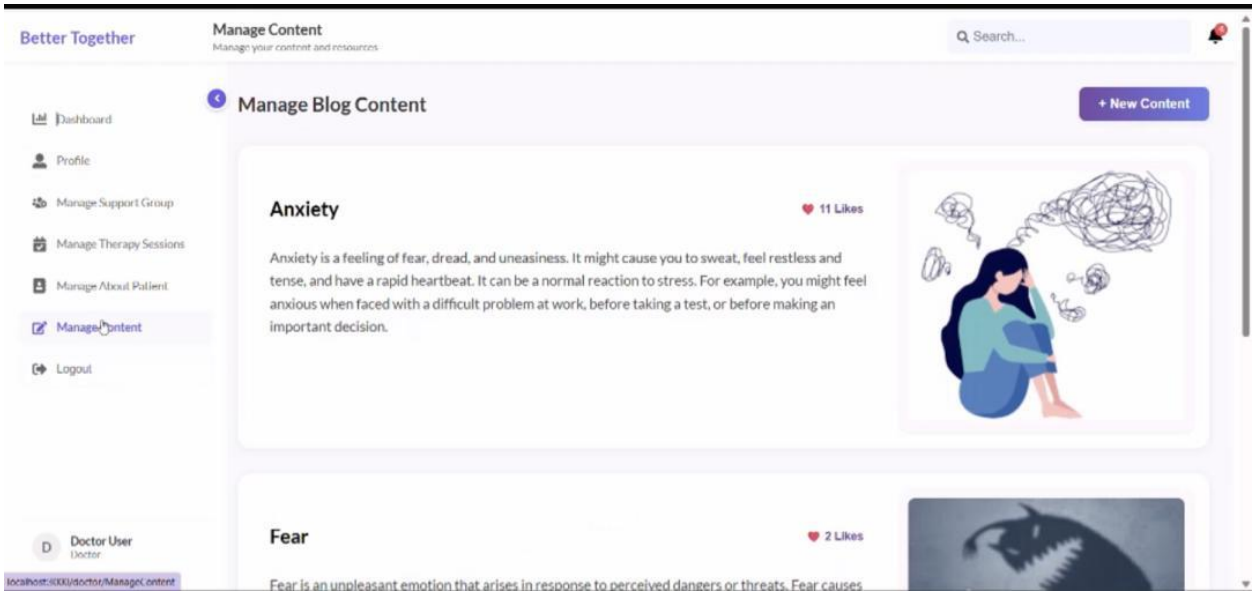
My Appointments:

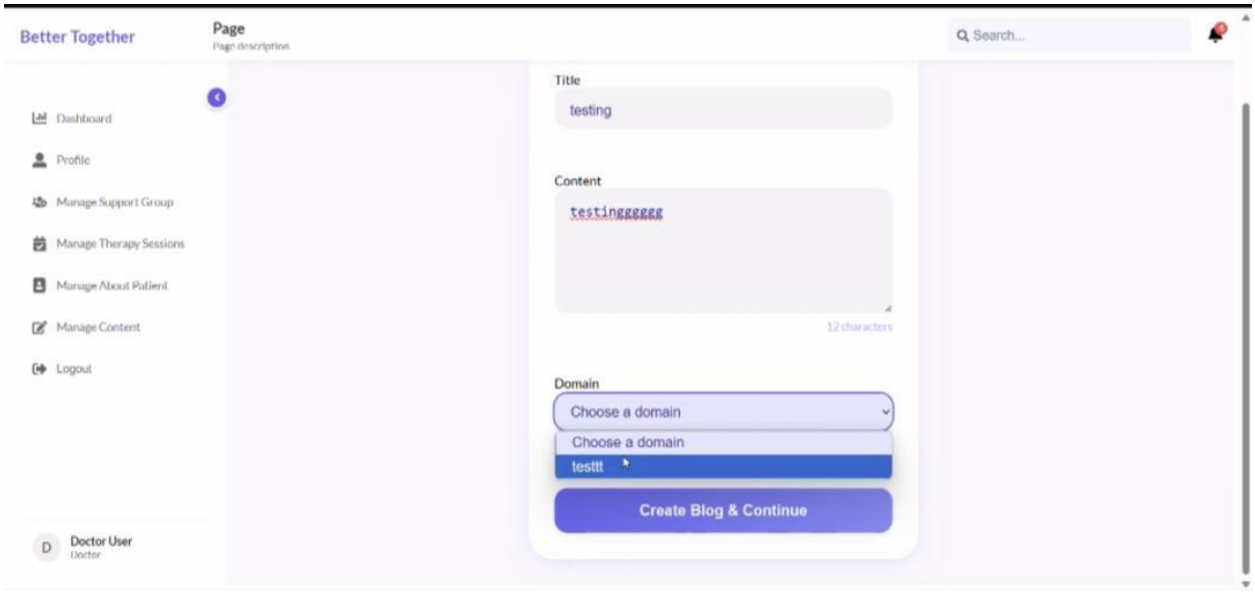


Blog :



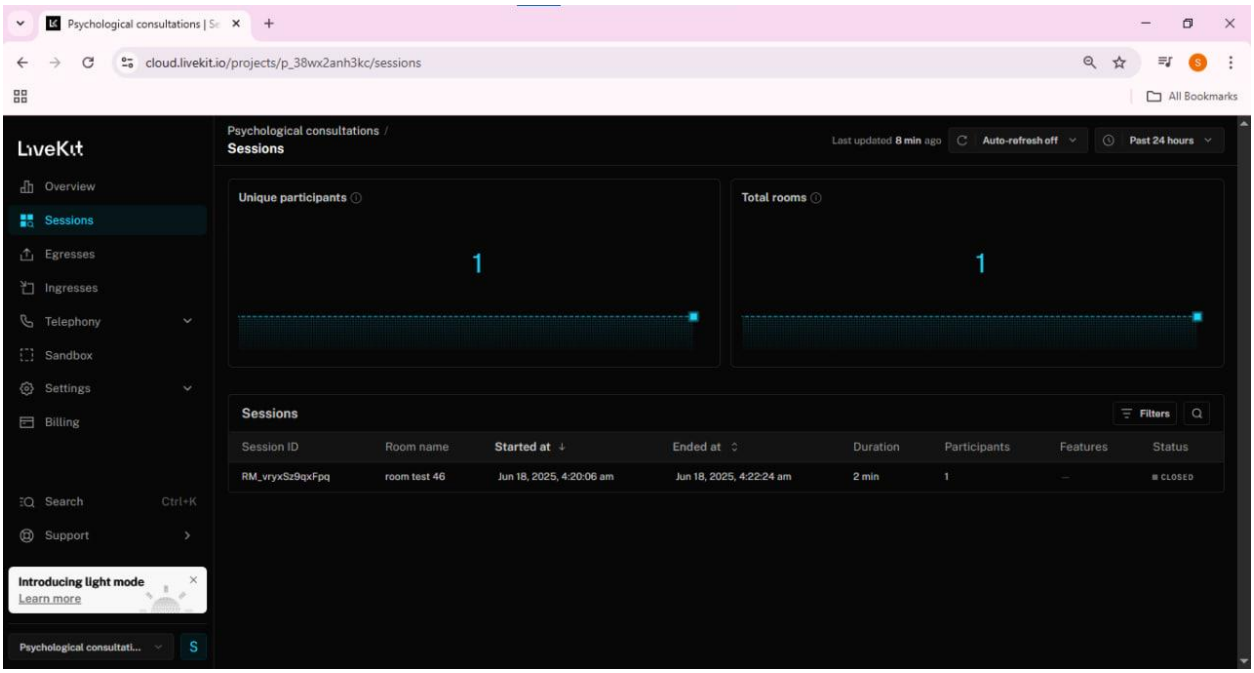
Mange Blog content doctor:





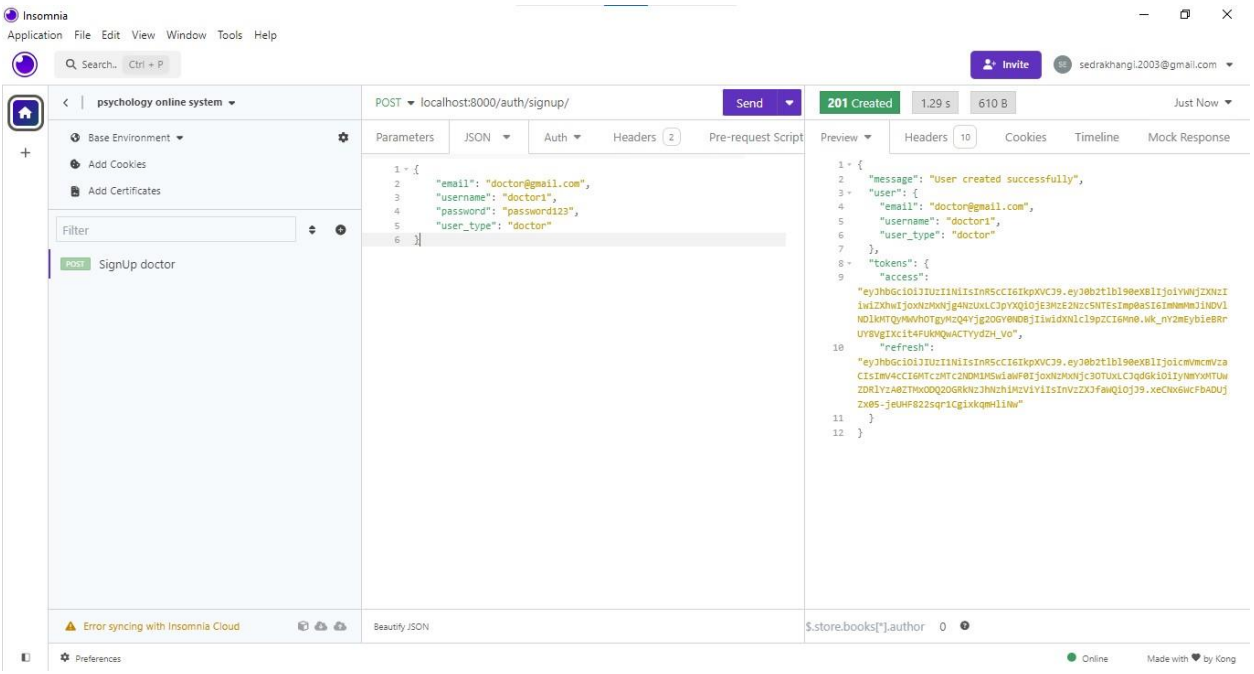
Livekit cloud interface:



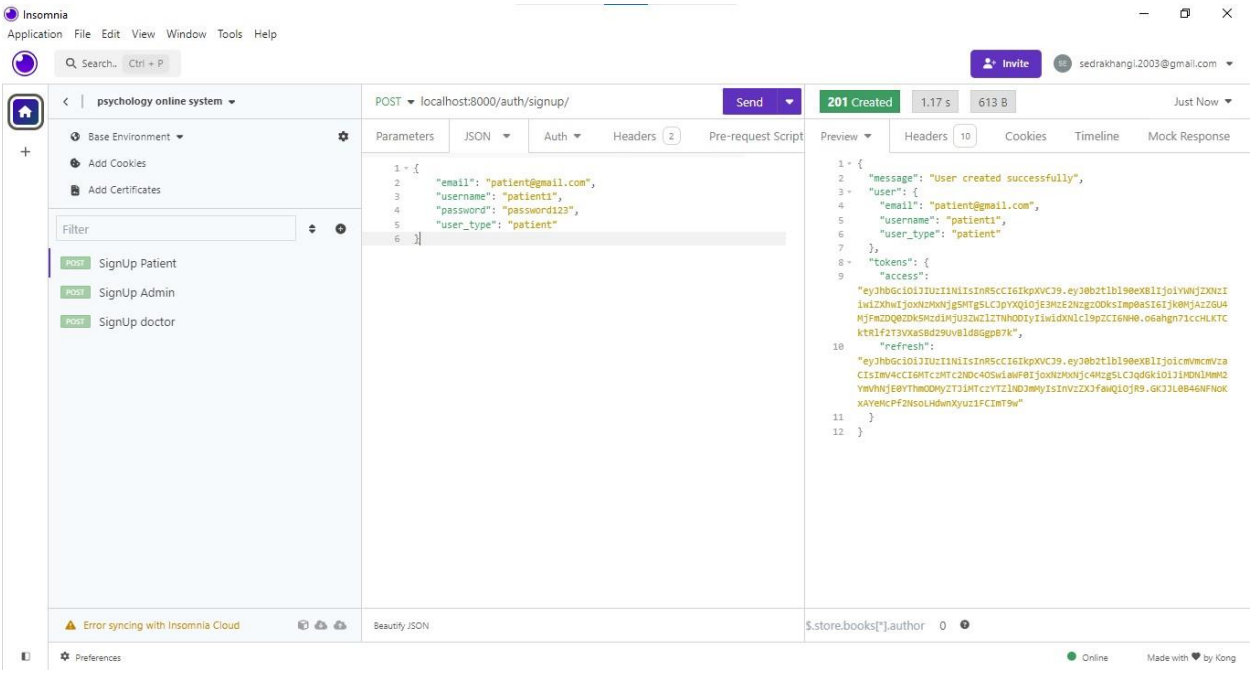

Psychological consultations | Sc
cloud.livekit.io/projects/p_38wx2anh3kc/sessions
All Bookmarks
LiveKit
Overview
Sessions
Egresses
Ingresses
Telephony
Sandbox
Settings
Billing
Search Ctrl+K
Support
Introducing light mode
Learn more
Psychological consultati...
S
Psychological consultations /
Sessions
Last updated 8 min ago
Auto-refresh off
Past 24 hours
Unique participants
1
Total rooms
1
Sessions
Filters
Session ID
Room name
Started at
Ended at
Duration
Participants
Features
Status
RM_vryxSz9qxFpq
room test 46
Jun 18, 2025, 4:20:06 am
Jun 18, 2025, 4:22:24 am
2 min
1
CLOSED

4. Testing in insomnia:

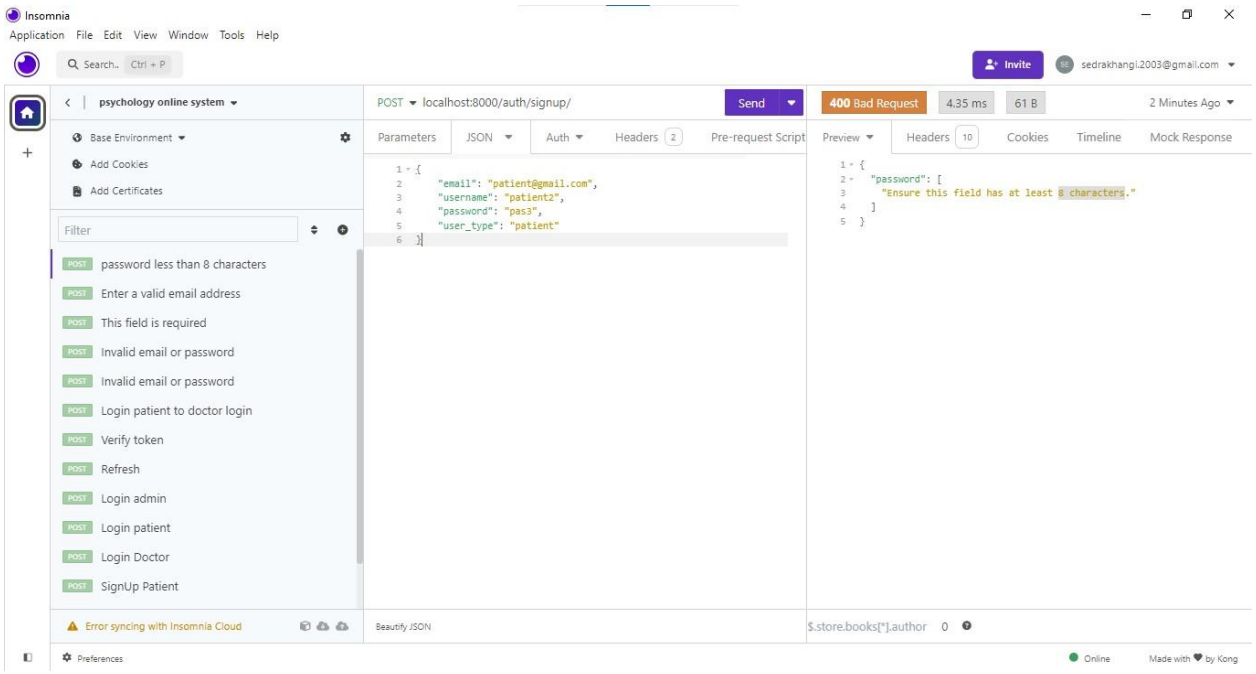
- ## Sign up doctor:



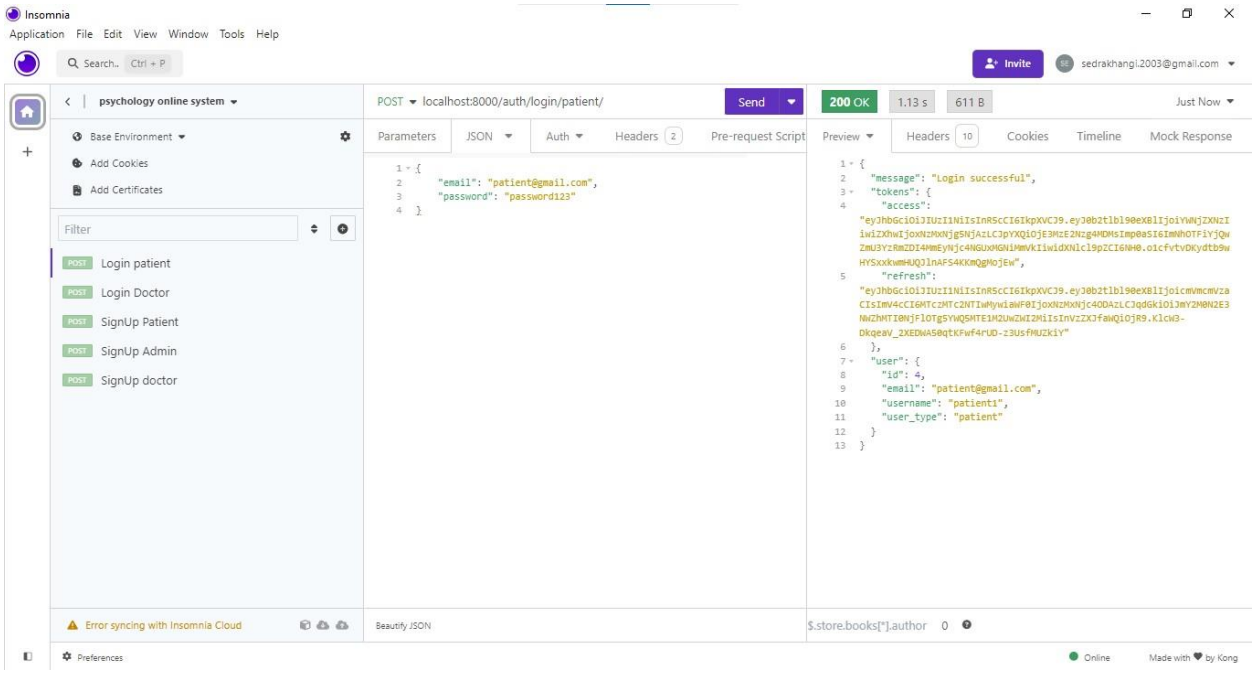
- ## Sign Up Patient:



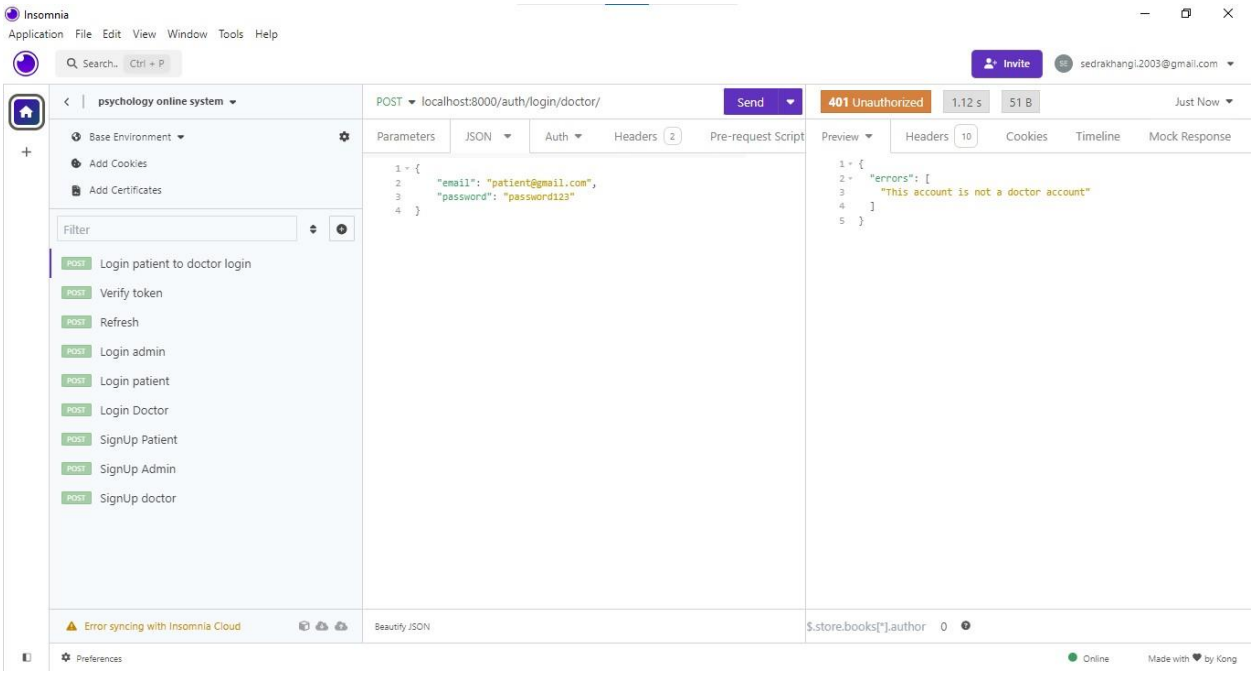
- ***Password is weak:***



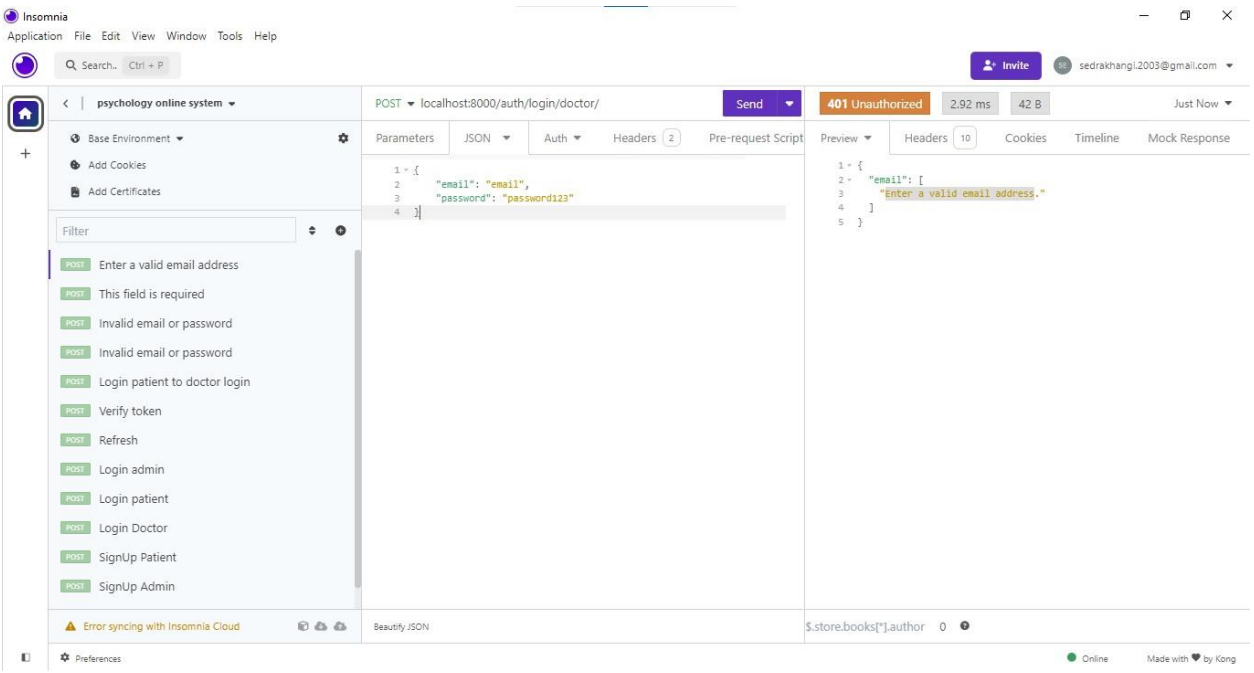
- ***login patient:***



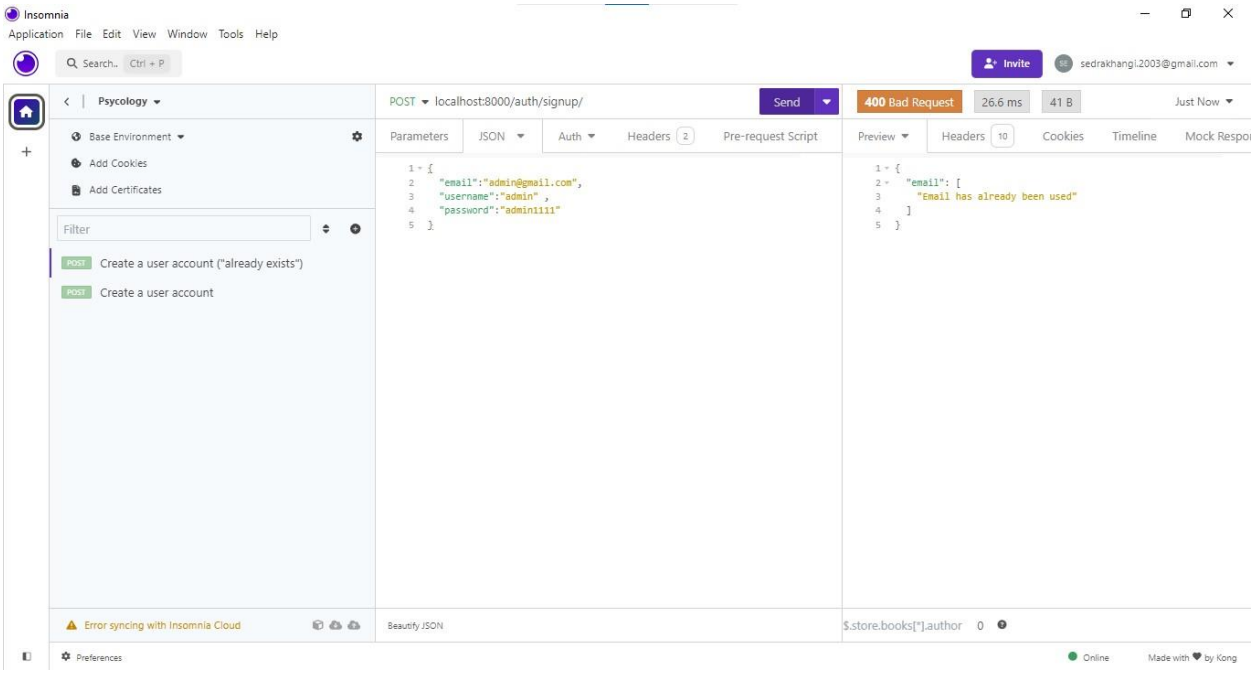
- *patient Login to doctor login interface*



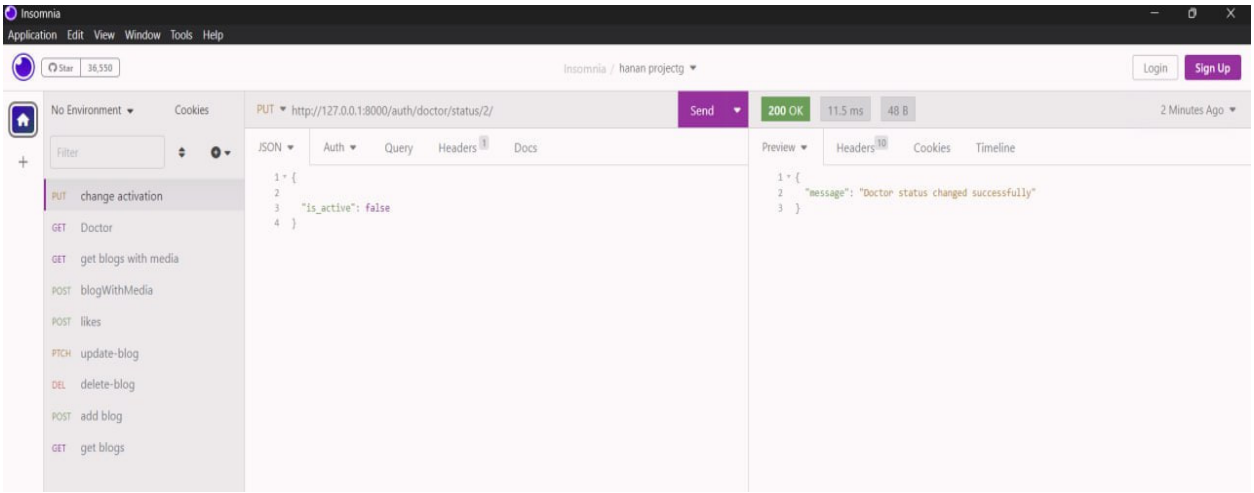
- *Enter invalid email address:*



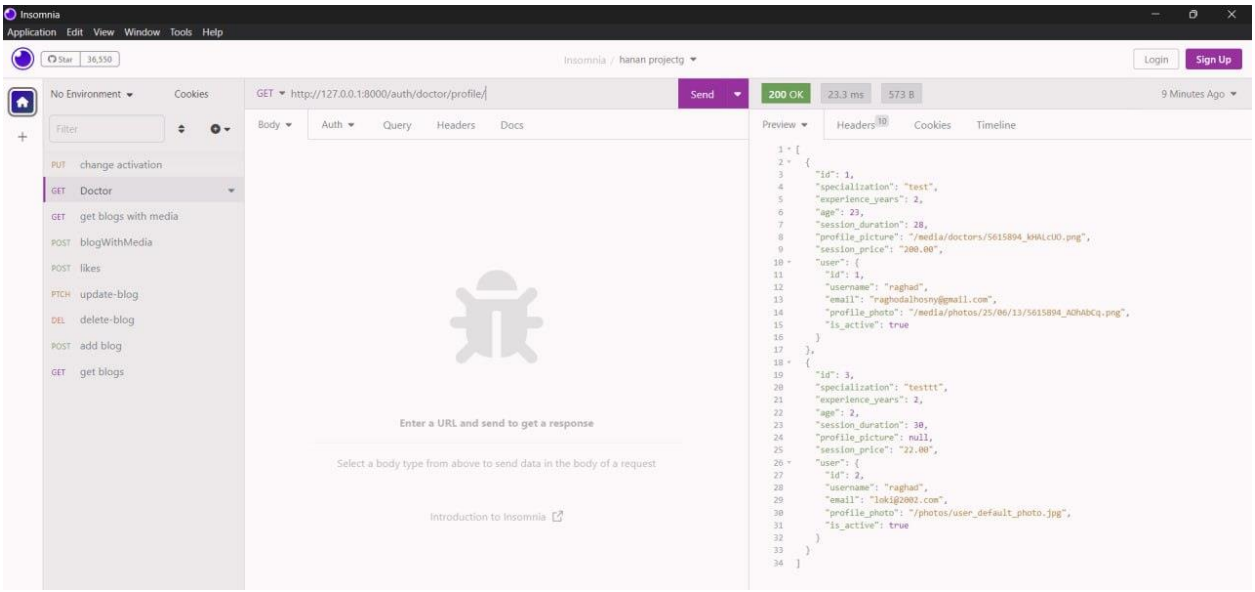
- *Create a user account already exists:*



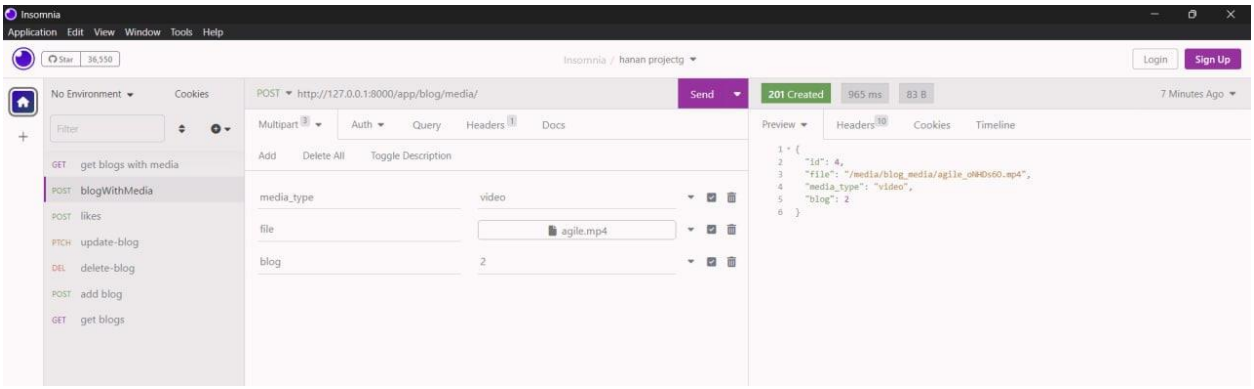
- *Update status account doctor by the admin*



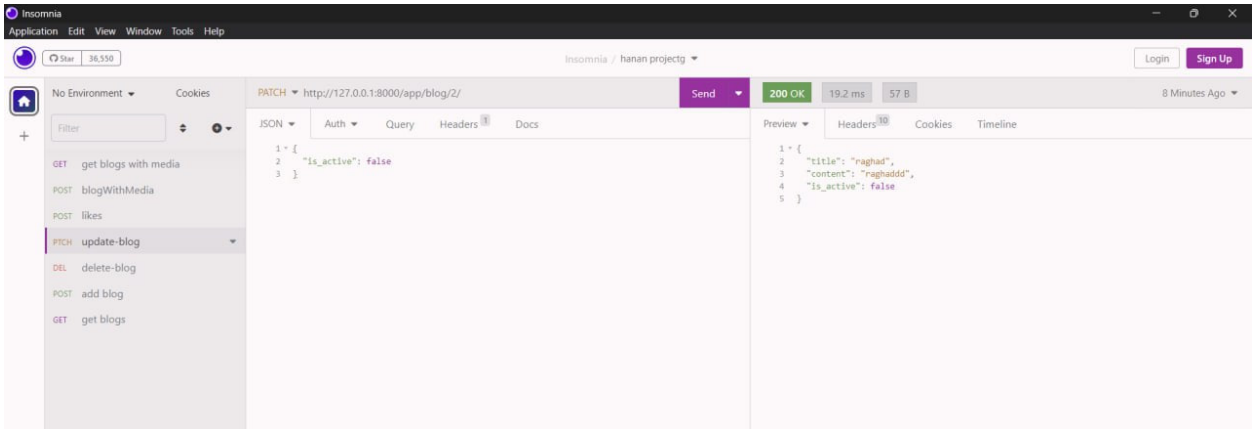
- *Get all doctors in the system*



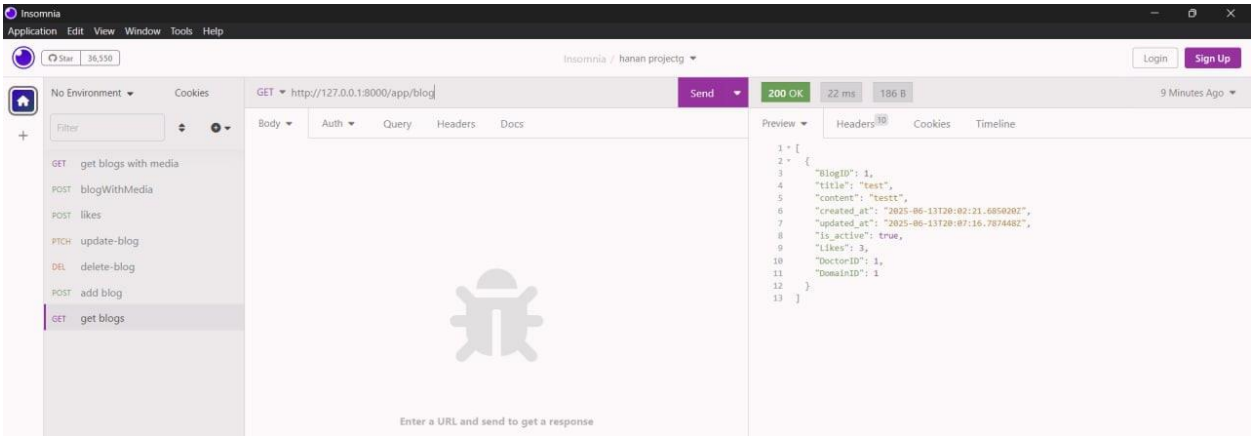
Post blog with media



- *Update blog by the doctor*

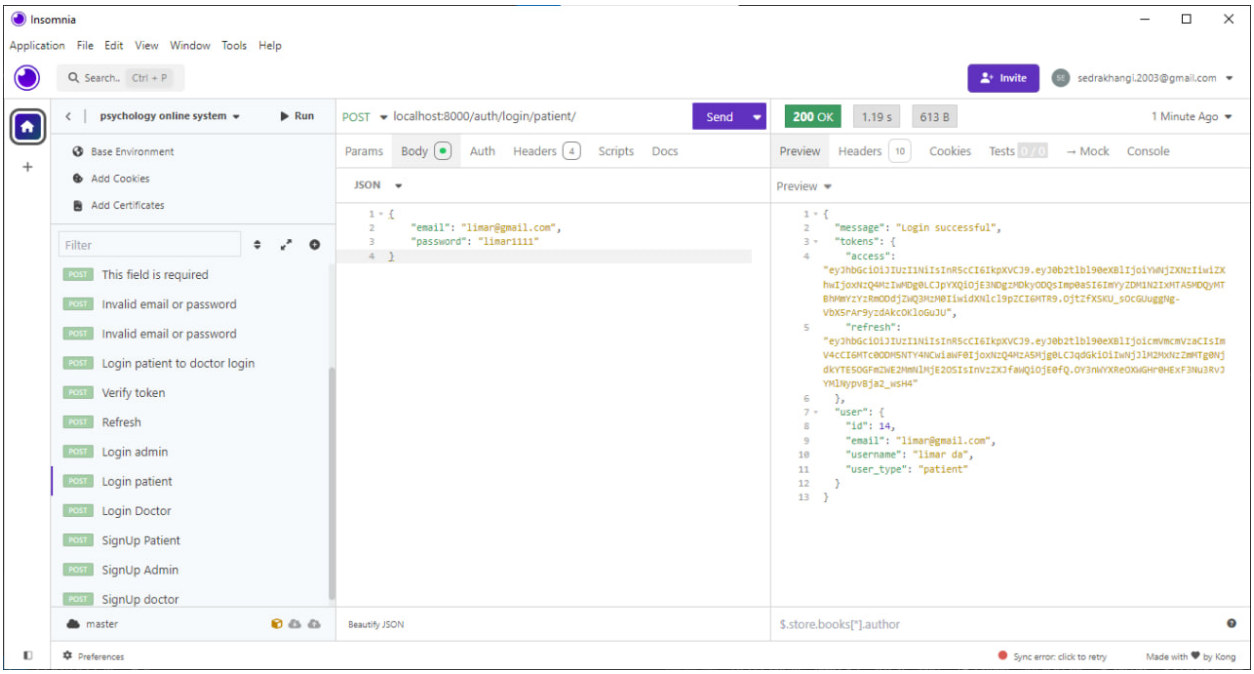


- *get blogs by the doctors*

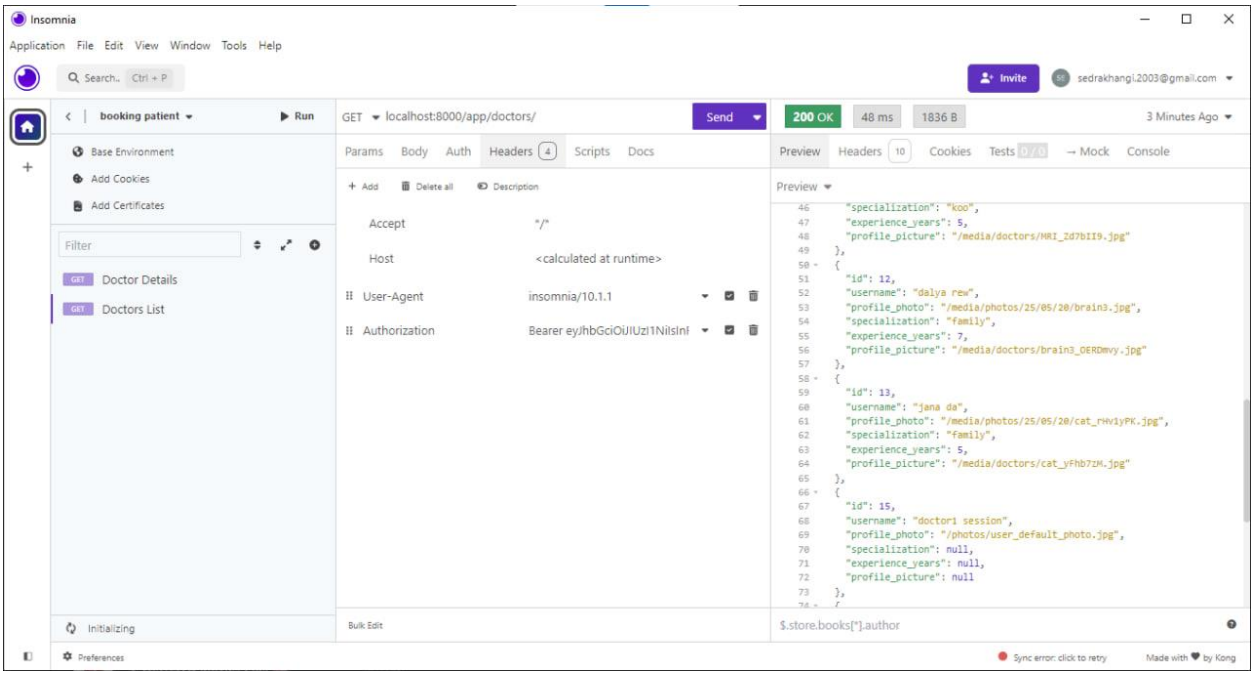


- *Patient appointment booking process sequence:*

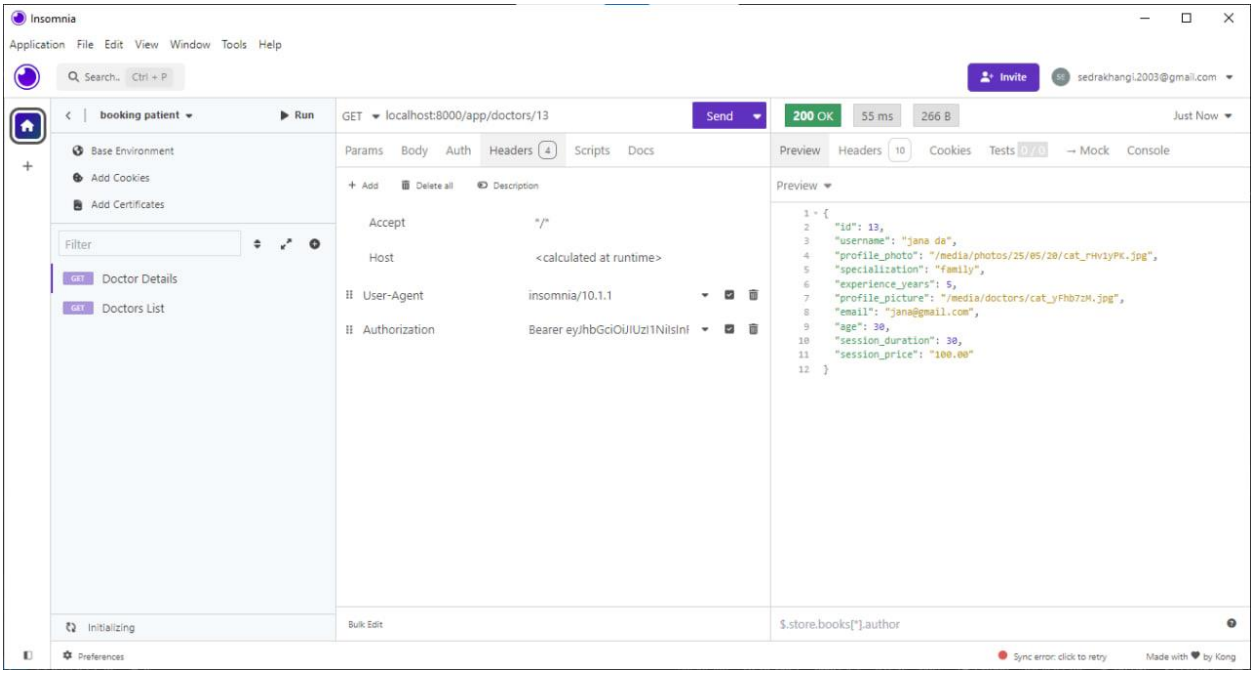
1. *Patient Login :*



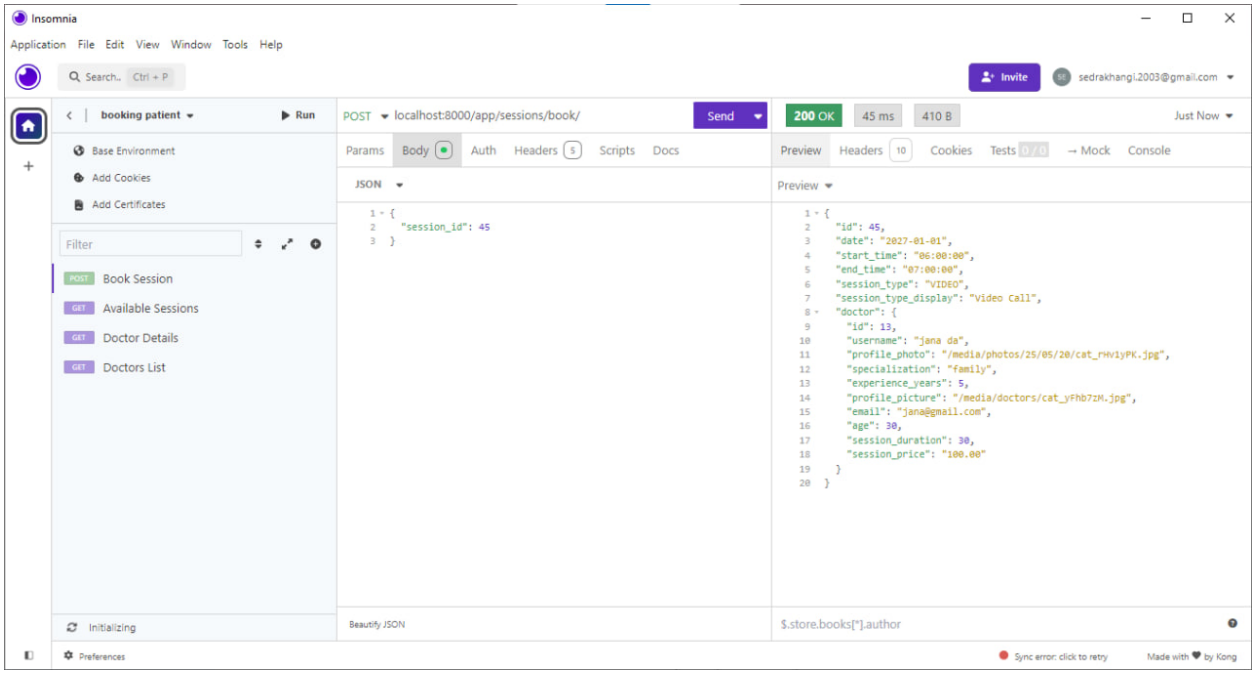
2. Doctors List:



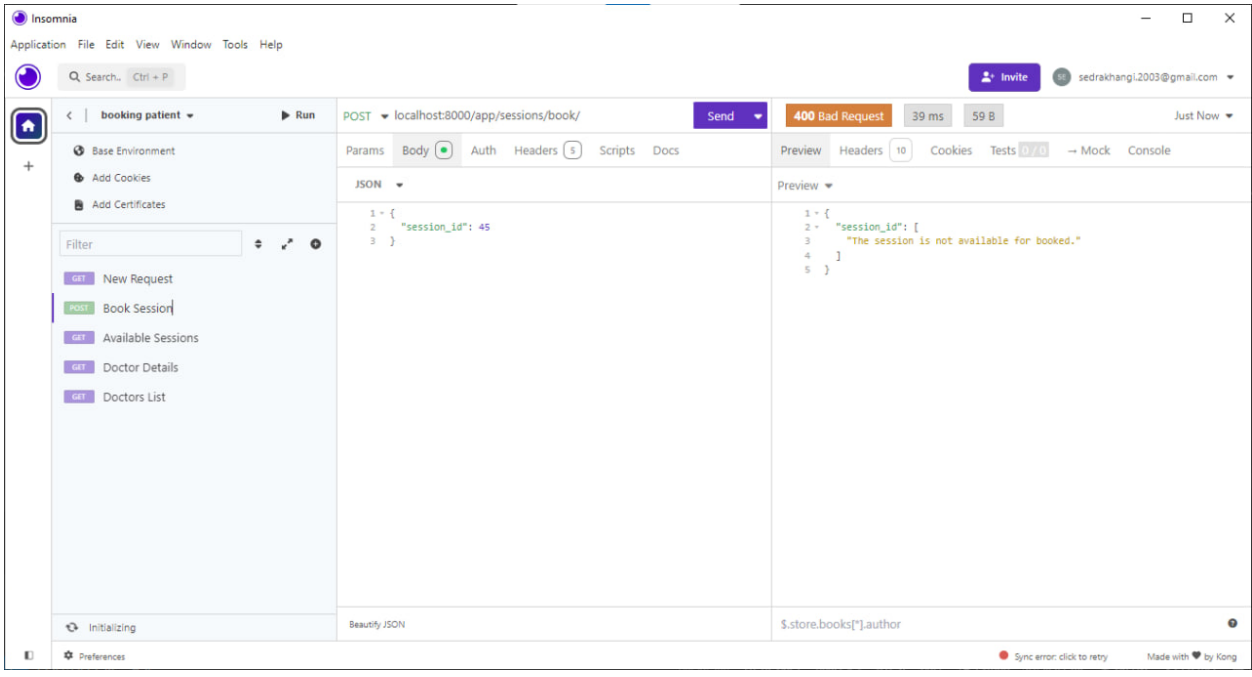
3. Doctor Details.



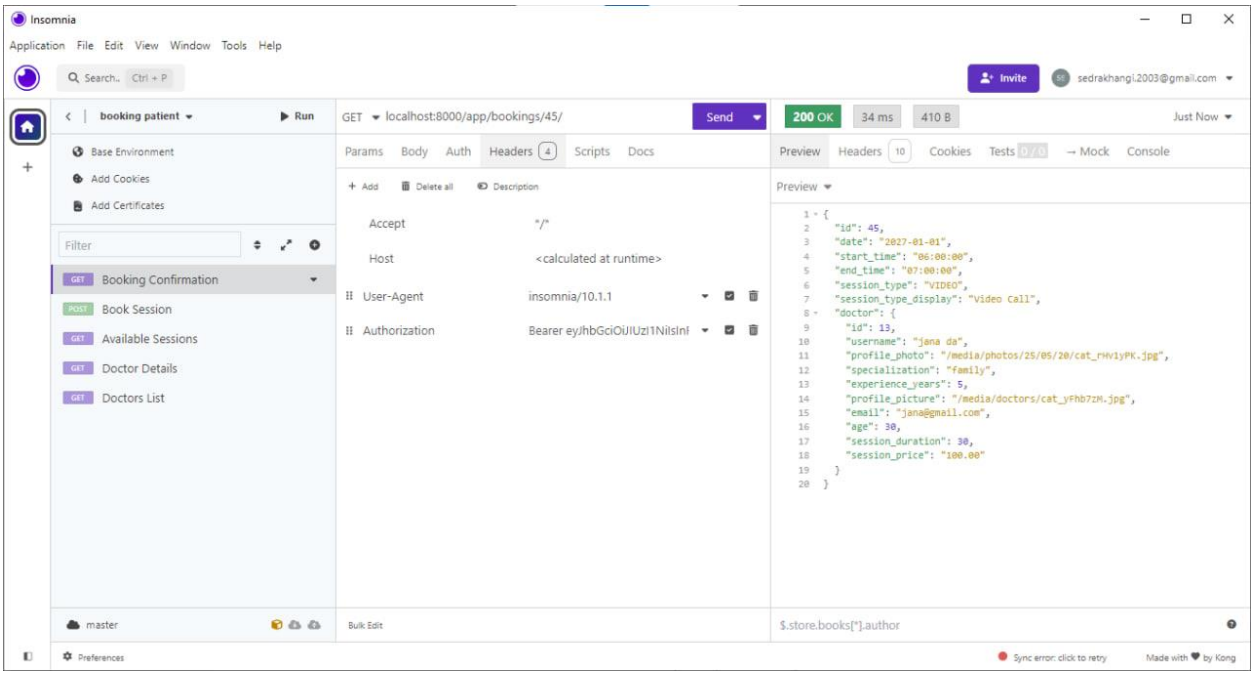
4. Book Session (Successfully booked):



Book Session (if booked the same session should return message 400 Bad request):

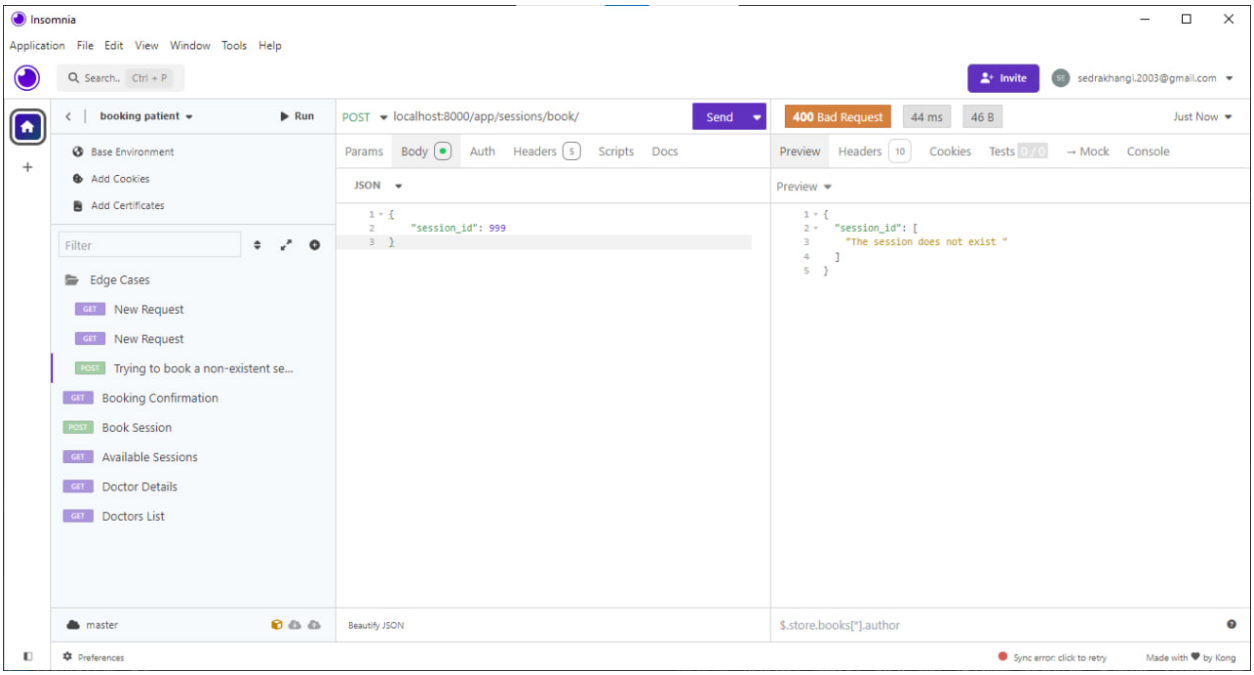


5. Booking Confirmation:

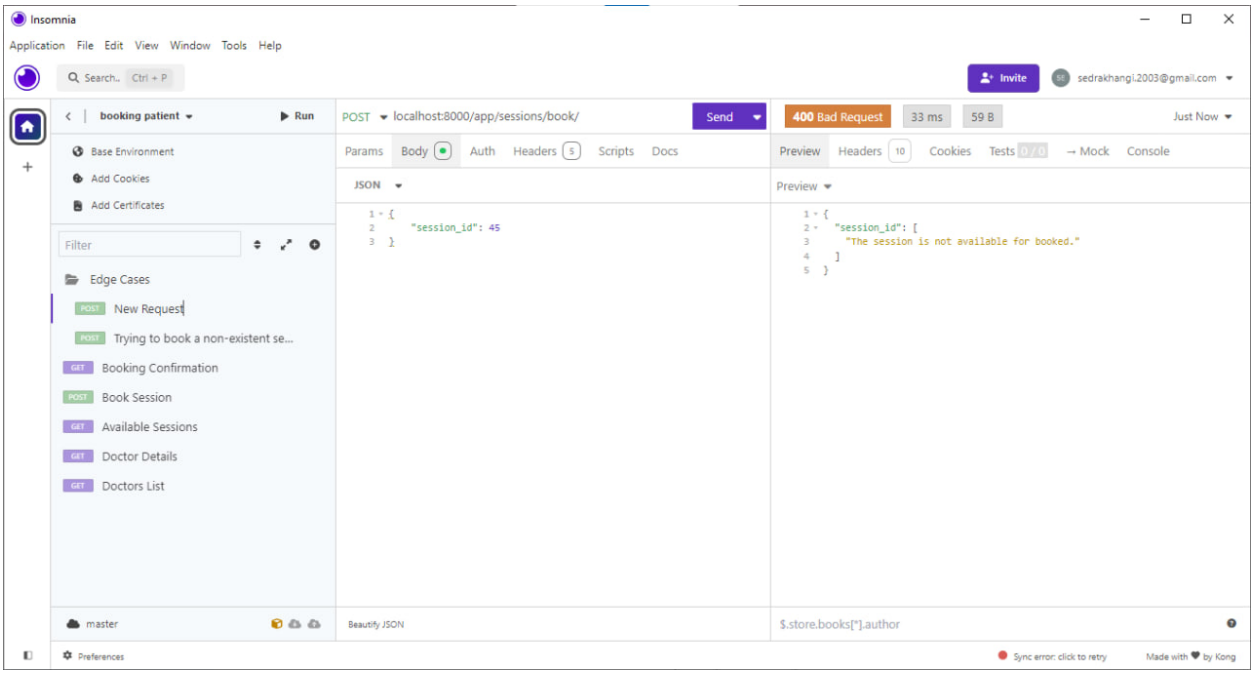


Edge Cases:

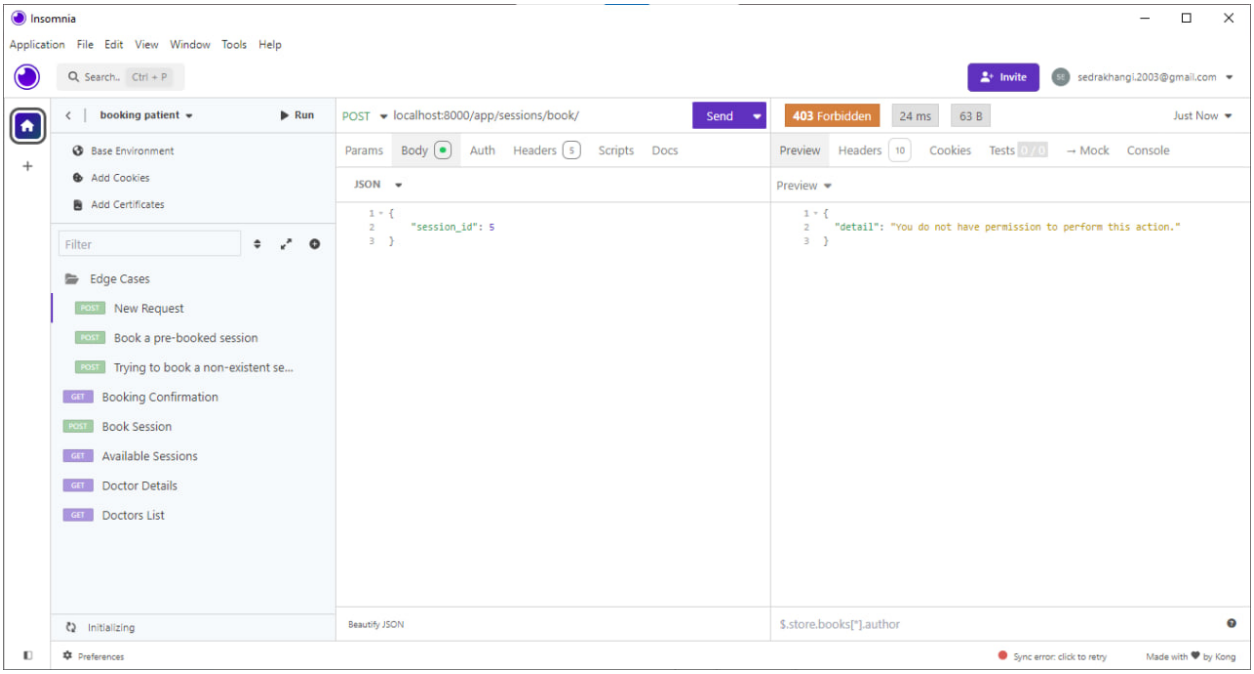
1. Trying to book a non-existent session:



2. Book a pre-booked session:

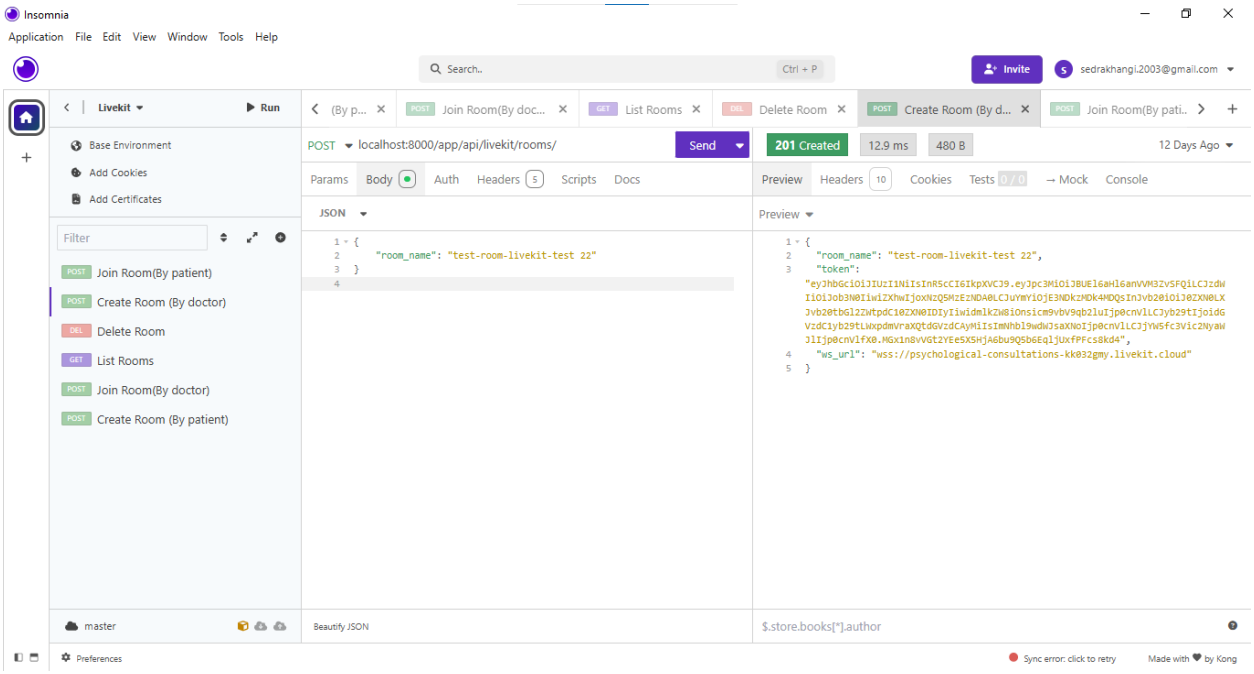


3. Trying to book a session with a doctor (instead of a patient):

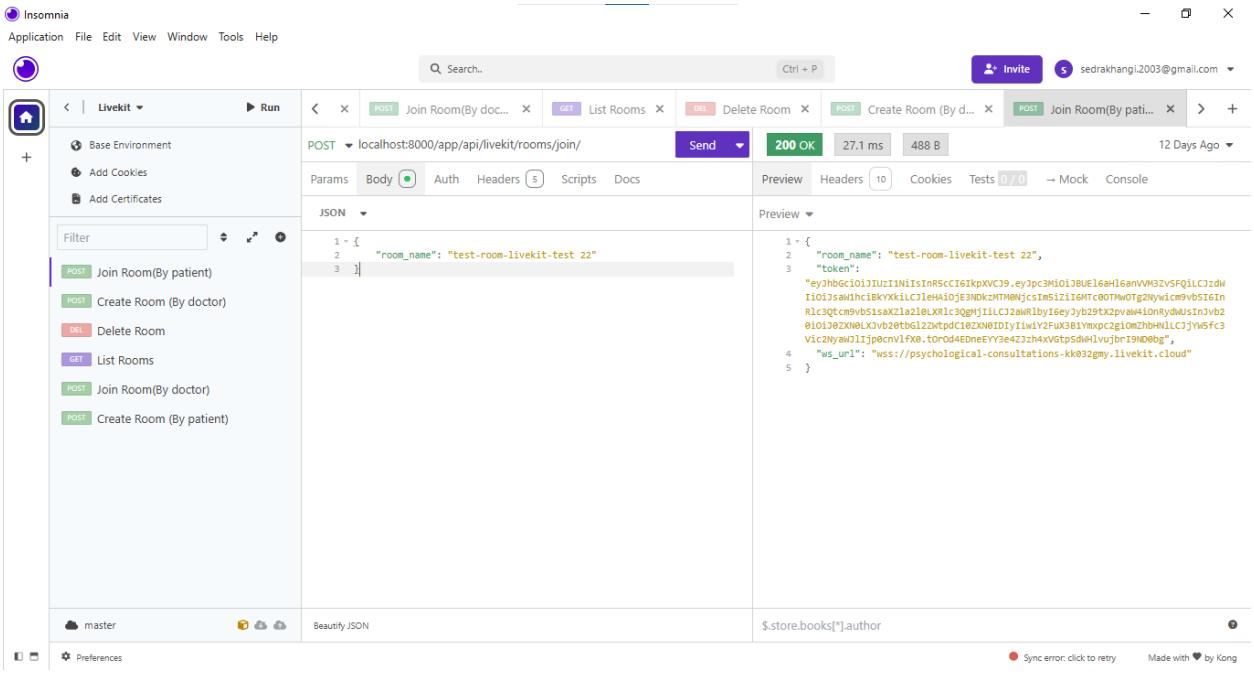


Livekit :

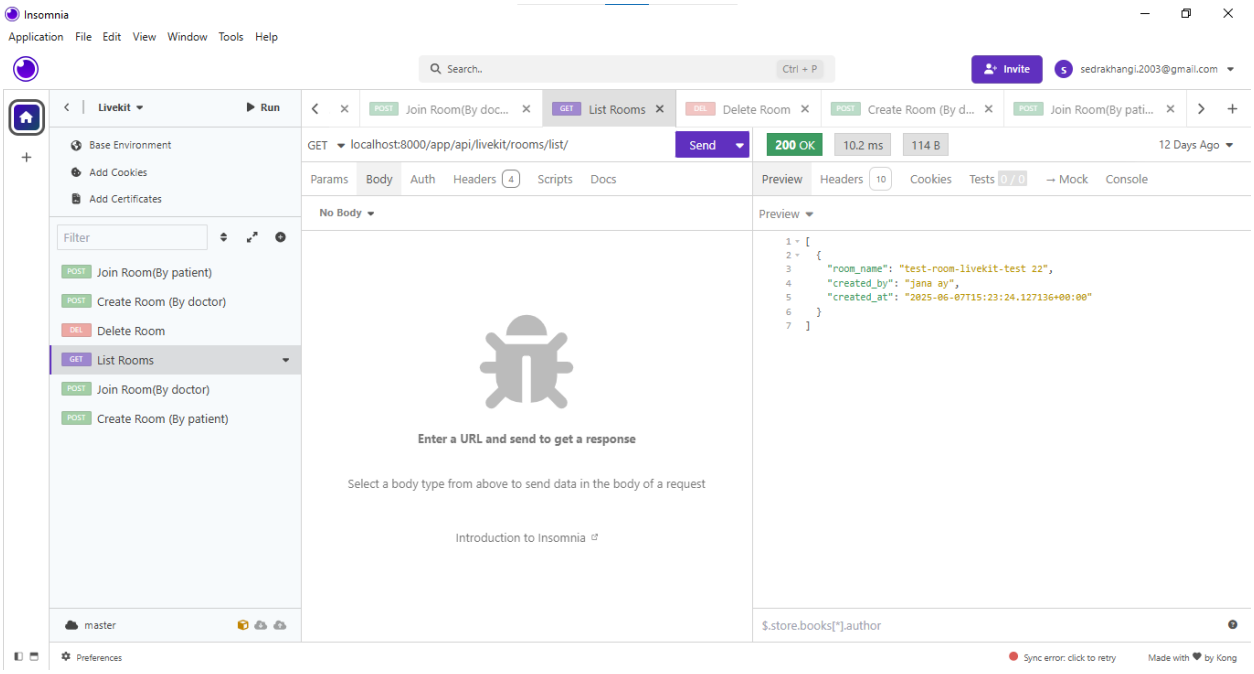
1. *Create Room (By doctor):*



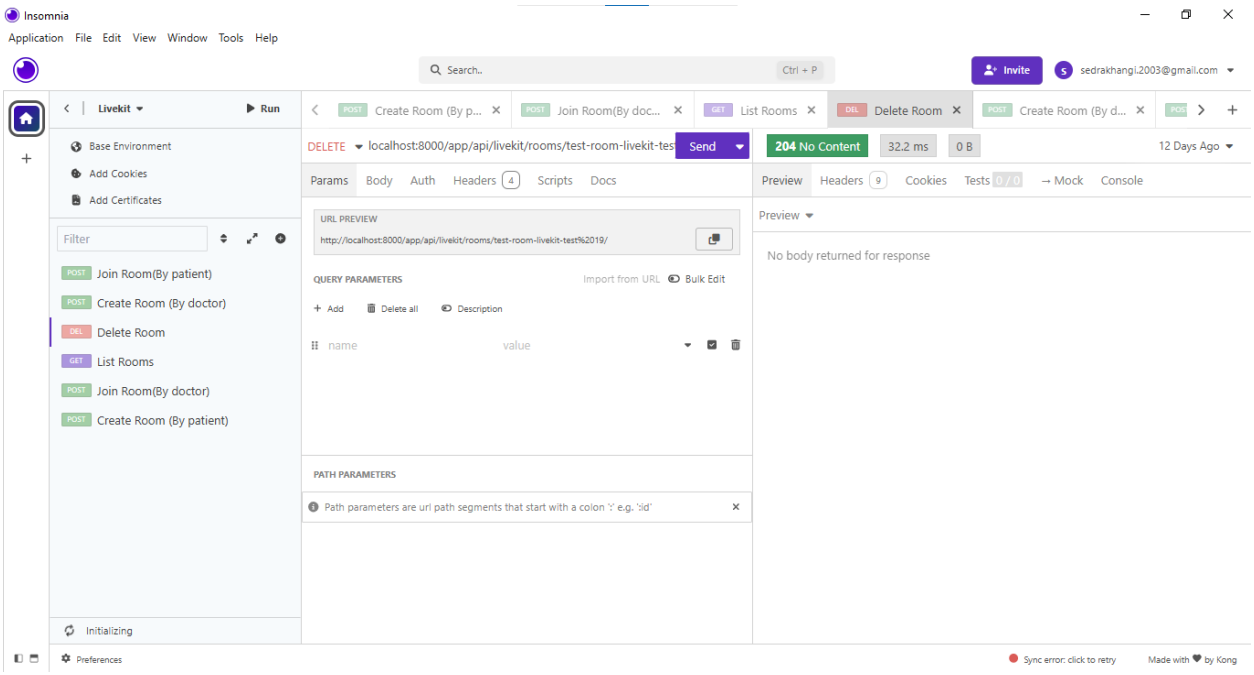
2. *Join Room(By patient):*



3. List Rooms:



4. Delete Room:



Chapter6 Conclusion

1. Introduction:

As a result of our project, we developed an Online Psychological Consulting system that helps mental health professionals enhance their services efficiently, ensuring that improvements align with patients' needs for maximum satisfaction and accessibility. To achieve this, we utilized modern AI technologies to analyze patient feedback, extracting valuable nsights that assist in refining therapy approaches, improving user experience, and tailoring mental health services to better serve patients.

2. References:

-https://docs.djangoproject.com/en/5.0/

https://www.google.com/search?q=react+documentation&oq=react+documentation&gs_lcrp=EgZjaHJvbWUyBggAEEUYOTIGCAEQRRg7MgYIAhBFGDsyBggDEEUYO9IBCDQzMTdqMGo3qAIAsAIA&sourceid=chrome&ie=UTF-8

-https://docs.python.org/3/

-https://www.mongodb.com/docs/

-https://github.com/sdra-khange/Senior-project

3. Appendices:

-Git repository