Introduction

Today, psychological treatments are still very popular, people may need these therapies because of their nature. In these therapies, sometimes the therapists cannot analyze the client well enough and the duration of the therapy may be longer. Today, digitalization also contributes to every aspect of our lives. We are developing a project, thinking that applying this digitalization in the field of health will make people's lives better. MAI Therapist is a Project designed to help Psychologists analyze therapy sessions. Therapists will use this app in their clinics. When using the MAI Therapist, therapists will have an analysis of how the client is feeling in a particular conversation. These feelings can be happiness, tension, anxiety, anger, calmness, feeling peaceful. To report these feelings, we will use a video camera to record the session and artificial intelligence that analyzes the client's emotions, gestures, facial expressions and tone of voice. At the end of the session, the therapist will have an analysis result that includes dialogues. Therapists can improve their approach by comparing their analysis with the MAI Therapist analysis with the help of frontend and backend development tools and see if there is a point they missed.

Proposed System

First of all, let's talk about the theoretical aspects of our system, we have a therapist and a client will come today. Our therapist has set up his system for the MAI Therapist, and the therapy session will begin when his client arrives. Our therapists will continue their therapy without any external influence. When it reaches the end of therapy, the MAI Therapist will stop video recording via the system. They will go over his own analysis and will also review the analysis of the MAI Therapist system whenever they need it. The client will seek and find answers to questions such as how did clients react to which question, what answer did they give, what kind of emotional changes they experienced.

If we come to more technical systems, we will make the artificial intelligence program with the python scripting program. We will import the required libraries externally into our own project. We will develop an admin page for therapists using VueJs, one of the frontend technologies. Here, the information of all the clients of that therapist will be registered in an encrypted manner, and only the therapist giving the therapy will be able to access this information. When video recording is stopped, we will store this video on a server. For this, we will use the Huawei Cloud Database Storage system. By storing it on a server, we will get rid of too much load on our local computer. We will also be able to use some systems provided by Huawei Cloud for future developments (speech to text, face recognition etc.). Again in this interface, the therapists will see the analysis of the video they uploaded in detail and will take their notes. When the therapists are finished, they will log out of their account in the interface. In this way, therapists will be able to log in from their personal accounts whenever they want and review their clients' information and analyzes again. We will use microservices for the backend service. We will use RestAPI for these as well, thanks to these services, therapists will be able to login, logout, create clients, and store client videos to the cloud.

Requirements

1.Functional Requirements

1.Users shall register with their email and password.

2.The system must send a verification link to user’s email to verify it.

3.Users shall log in with their credentials (email and password)

4.The system must send a link to user’s mail to renew user’s password.

5.The user shall see patient list

6.The user shall see patients detailed data.

7.The user shall update their patient’s data

8.The user shall add or remove patient

9.The user shall download transcript of the session

10.The user shall Download analysis of the session

11. The user shall pay for the service

12.The user shall see analysis, transcript and download them

13.The user shall refund the payment

14.The user shall see the payment history and download the receipt

15.The user shall see the refund history

16. The user shall buy a subscription plan

17. The user shall see the subscription history

18. The user shall contact the support

19. The user shall send feedback

20. The user shall add a payment method (credit card)

21. The system must automatically charge the user’s credit card when the subscription is expired

22. The system must send an email to the user when the subscription is expired

23. The system must send an email to the user when the subscription is renewed

24. The system must send an email to the user when the subscription is canceled

25. The system must send an email to the user when an analysis and transcript is ready

26. The system must send an email to the user when a payment is made

27. The system must send an email to the user when a refund is made

28. The user shall submit a video of the session to the system

29. The system must analyze the video and generate a transcript and analysis

30. The system must send an email to the user when the video is analyzed

31. The system must delete video records after 30 days.

**Non-functional Requirements**

**Safety Requirements:**

1.System shall back-up all data in safe. This requirement prevents any data loss.

2.Due to any possible hardware problem, data must be saved in a cloud

3.All data must be encrypted

4. Request must be authenticated and authorized before processing them to prevent any unauthorized access

5. Requests to the backend service must be from a trusted source which is the frontend service

6.The system must be safe for sql injection attacks, cross site scripting attacks.

7.High amount of request from one source must be prevented to prevent any denial-of-service attack

**Performance Requirements:**

1.In every request except submitting the session video, server must send a response in five seconds maximum.

2.System should run 7/24 without any error

3.Multiple users can use the system at the same time.

4.Frontend service's lighthouse score must be 85 or higher

**Security Requirements**

1.Since we store very sensitive data, there should not be a security leakage.

2.Key management: Patient’s sensitive information must be encrypted.

3.Backup: We must have a back-up database in case of any data loss.

4.Captcha: System must prevent unnecessary traffic by a captcha if we get too many requests from the same IP.

5.Log system: System must log every action of the users.

**Software Quality Attributes**

6.Reliable: System must function properly.

7.Simplicity and usability: System shall be designed with these attributes. They provide easy maintenance for next users.

8.Efficient: System must be efficient that not unnecessarily use of memory, ram, cpu.

9.Portable: System must be cross platform. Desktop, mobile, iOS, android etc.

10.Integrity: Any unauthorized operations must be prevented.

11.Flexibility: Additional features can be easily added to the current system.

12.Reusability: The System designed reusable for any car rental system.

13.Maintainability: System should be designed as maintainable due to creating a continuity system.

14.Adaptability: System shall be designed as open to changes and updates. When an update is needed, it must be adaptable to the system easily.

**Pseudo requirements**

Front-end must be created by JavaScript especially the Vue framework.

All dependencies that will be included in the project, must be widely used already.

Code base must be clear and well documented.

Server must be implemented by python.

Database must be implemented by PostgreSQL.

Uploading video must be implemented by a cloud service.

**SYSTEM MODELS**

To visualize and explain our project, we used some of the UML models given below.

With behavioral models -in this case activity model, use-case model, and the sequence diagrams- we model the dynamic behavior of the MAI Therapist system and how it responds to events. In the other hand, with structural models -which are object and class model in this case- we model the organization of the MAI Therapist system

Diagram

Description automatically generated

**Scenario**

* User opens the application
* There are Sign-in and Sign-up buttons displaying on the screen
  + User clicks the Sign-in button
    - User enters their E-mail and password then clicks to login button
    - Main page appears where the user can see the statistics, analyses, and patient lists.
  + User clicks the Sign-up button.
    - Register form appears on the screen.
    - User fills the identity form with valid information.
    - Registration is successful, payment page appears
    - User makes payment to access the app’s features.
* User chooses the patient list from main page to add new patients.
* Fills the new patient information and add saves it.
* Patient list appears including the new patient’s information.
* User goes to main page with clicking the main page button.
* Chooses the analyses.
* Analyses list page opens, user choose one of the sessions.
* Analysis including session’s transcript and video appears on the screen.

Activity Model of the system when user access the platform to sign-in/sign-up

Diagram

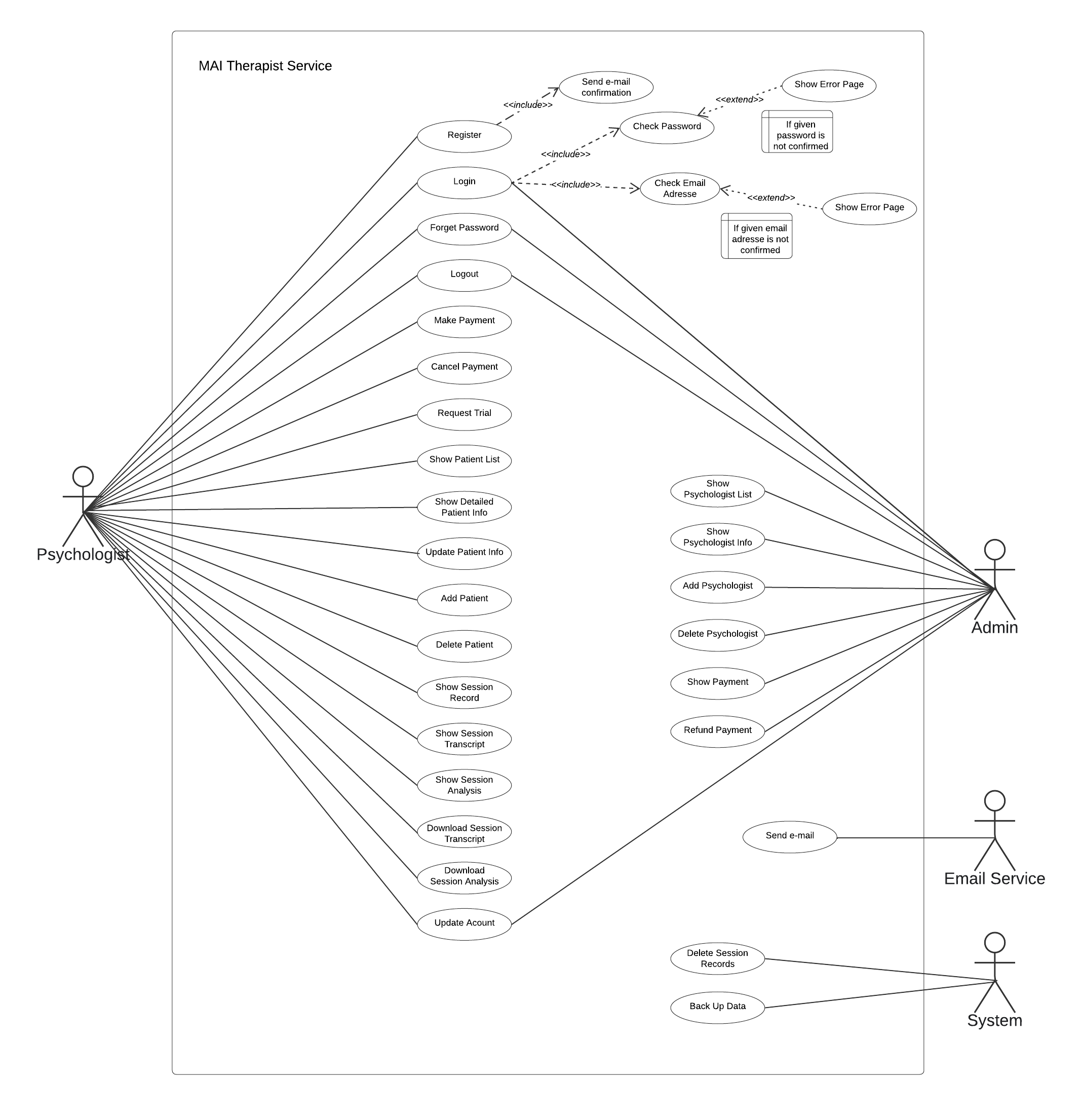
Description automatically generated

Object and Class Model

A picture containing text

Description automatically generated

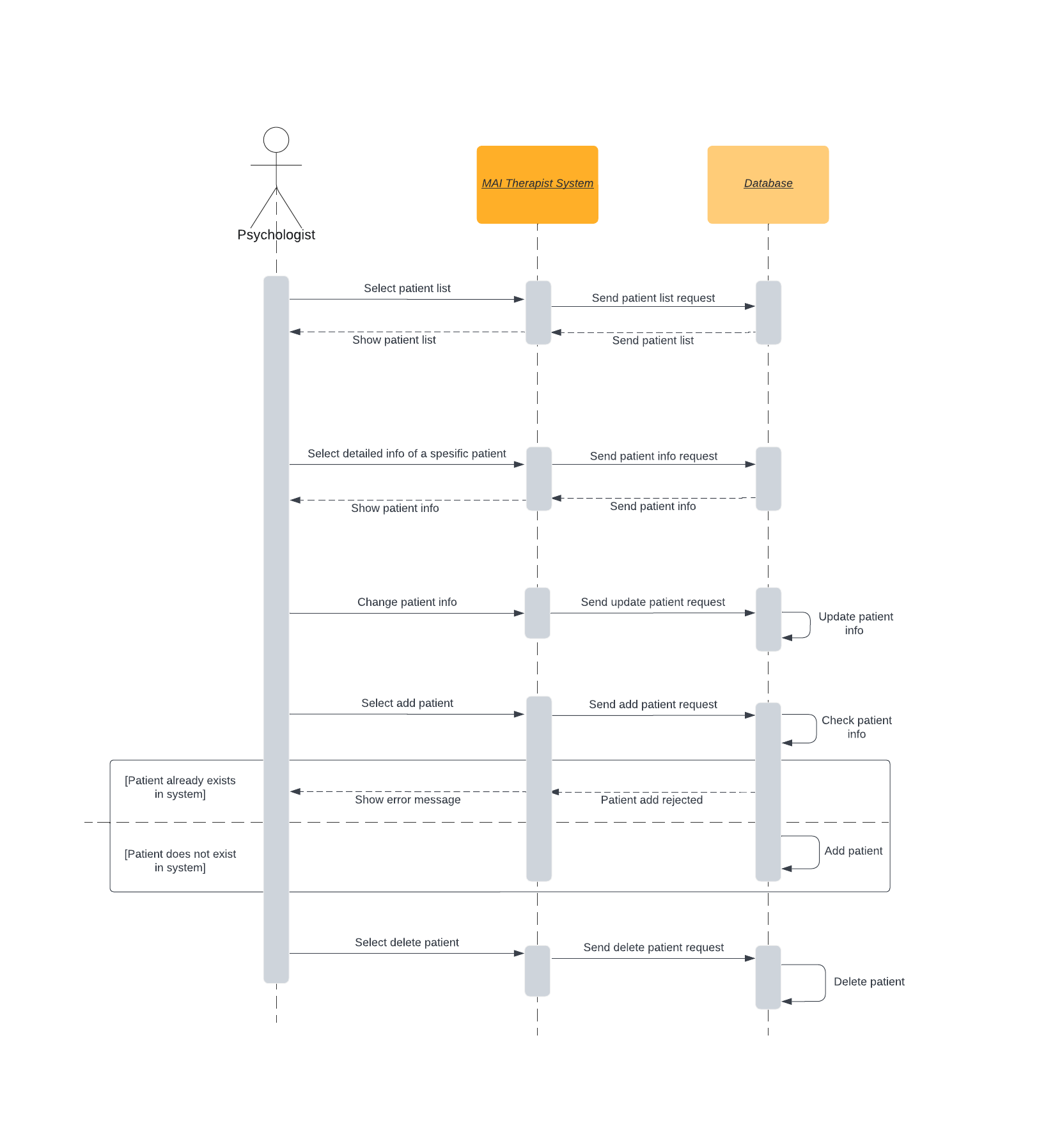
**USE CASE MODEL**

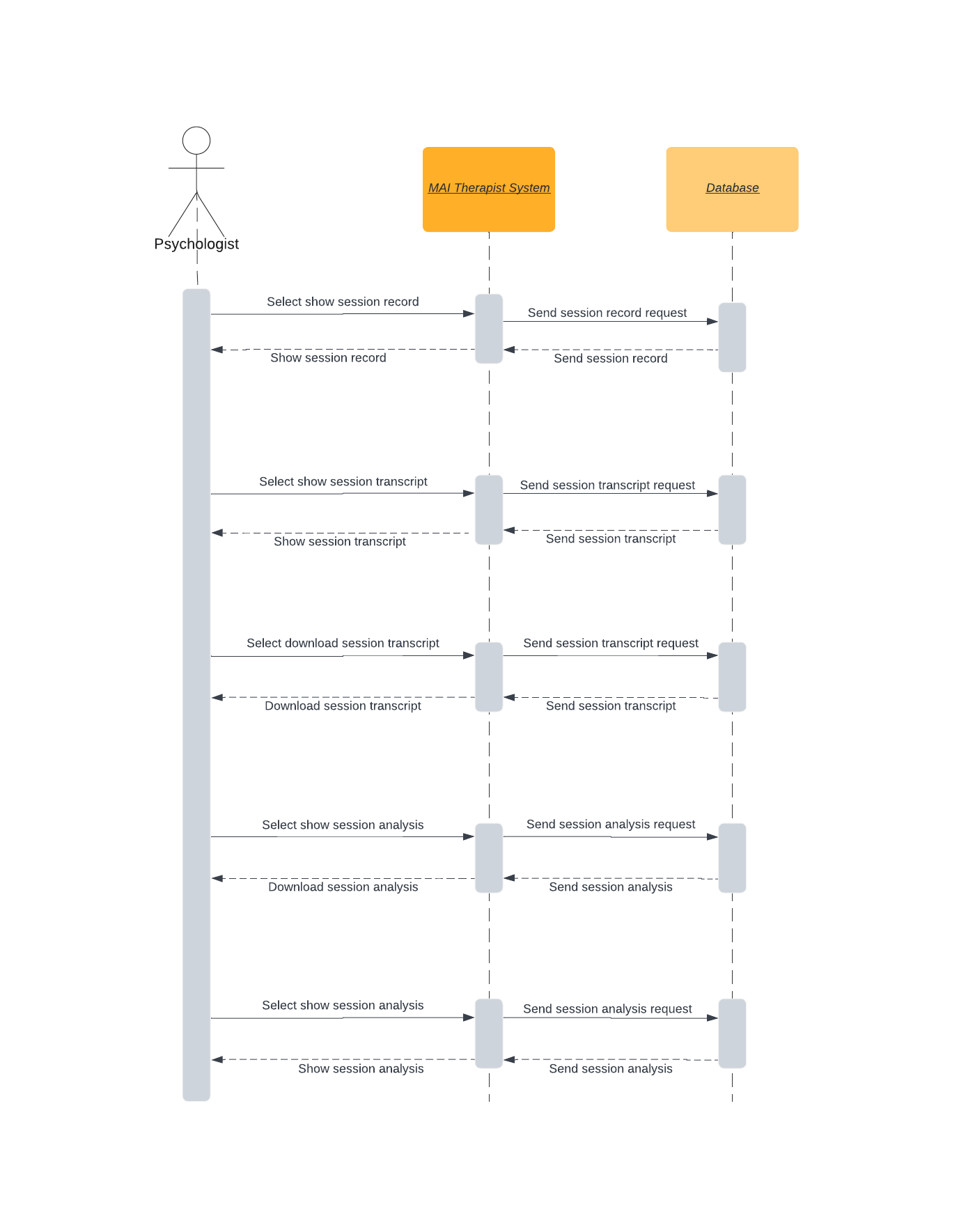


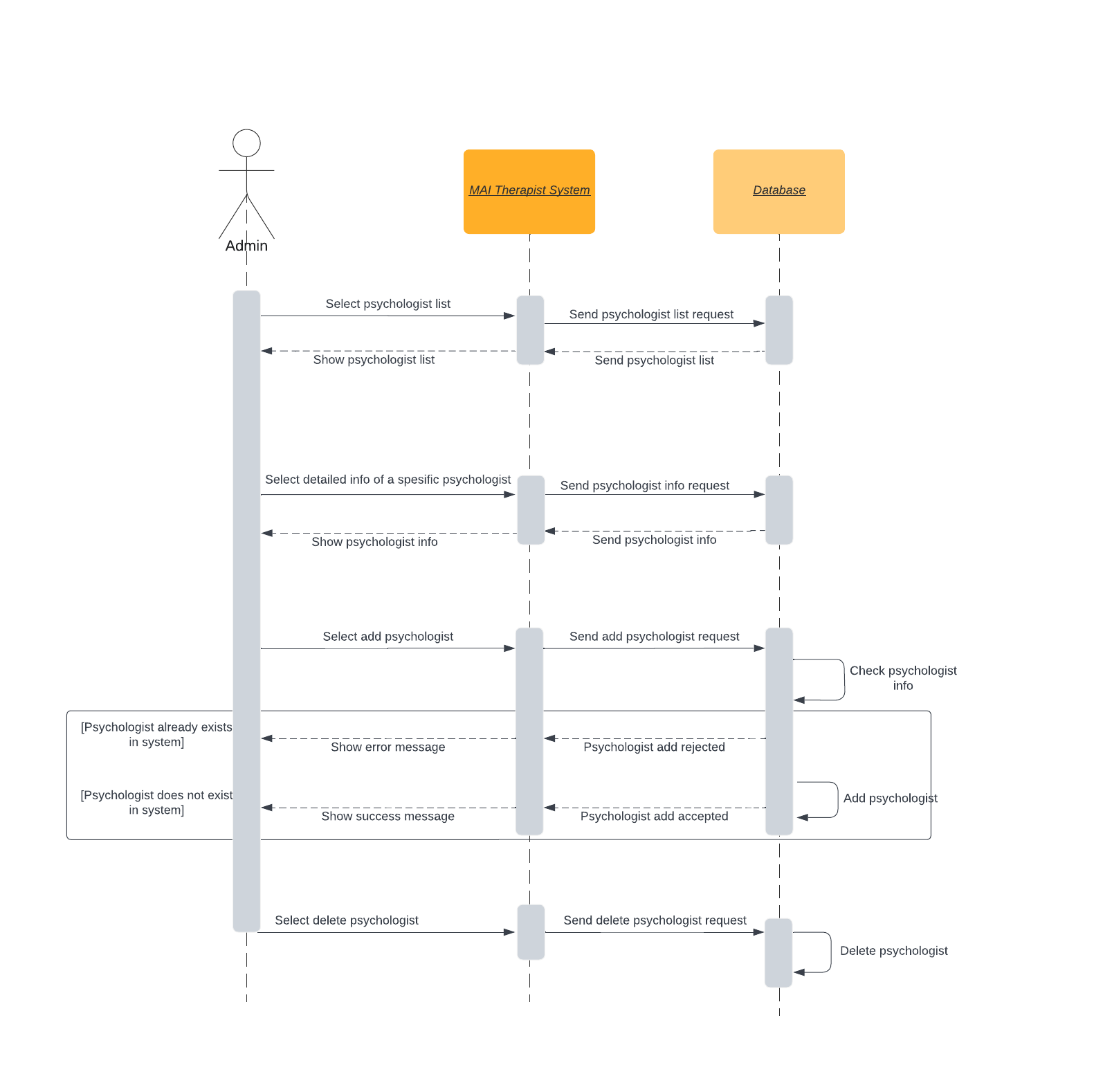
**DYNAMIC MODELS**

Sequence Diagrams

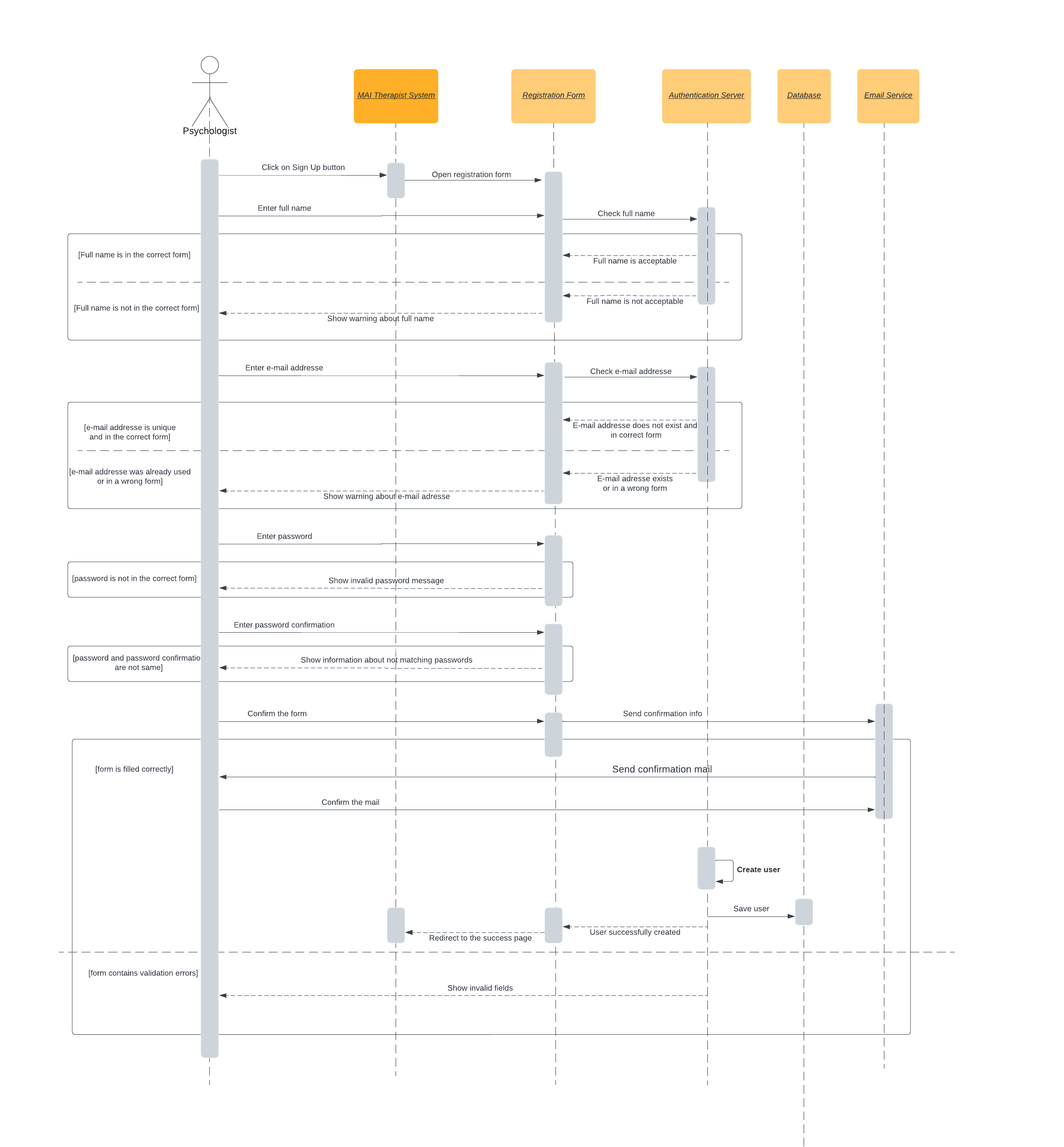
* Patient Related Sequence Diagram for Psychologist

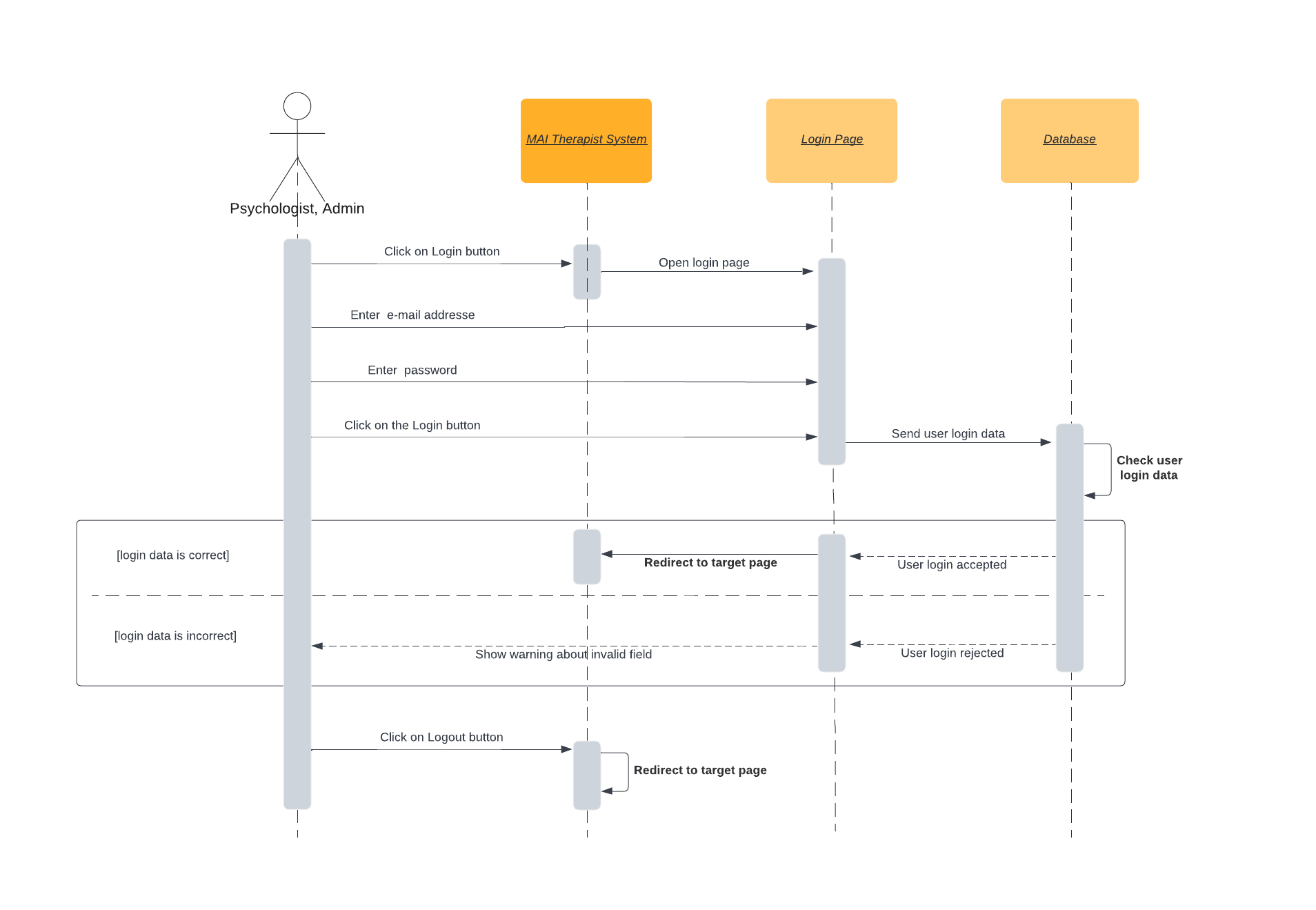


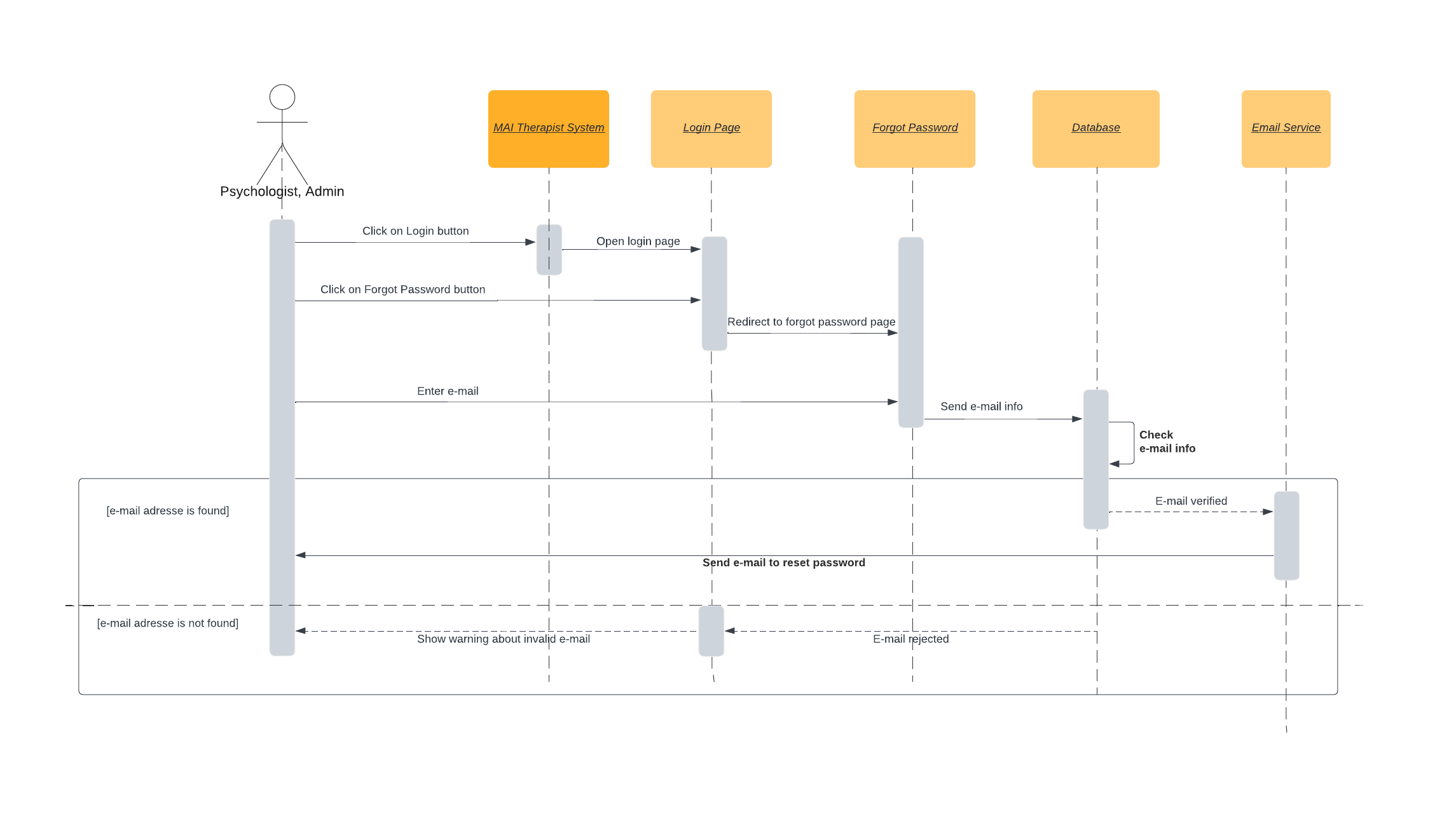
* Session Related Sequence Diagram
* Psychologist Related Sequence Diagram for Admin

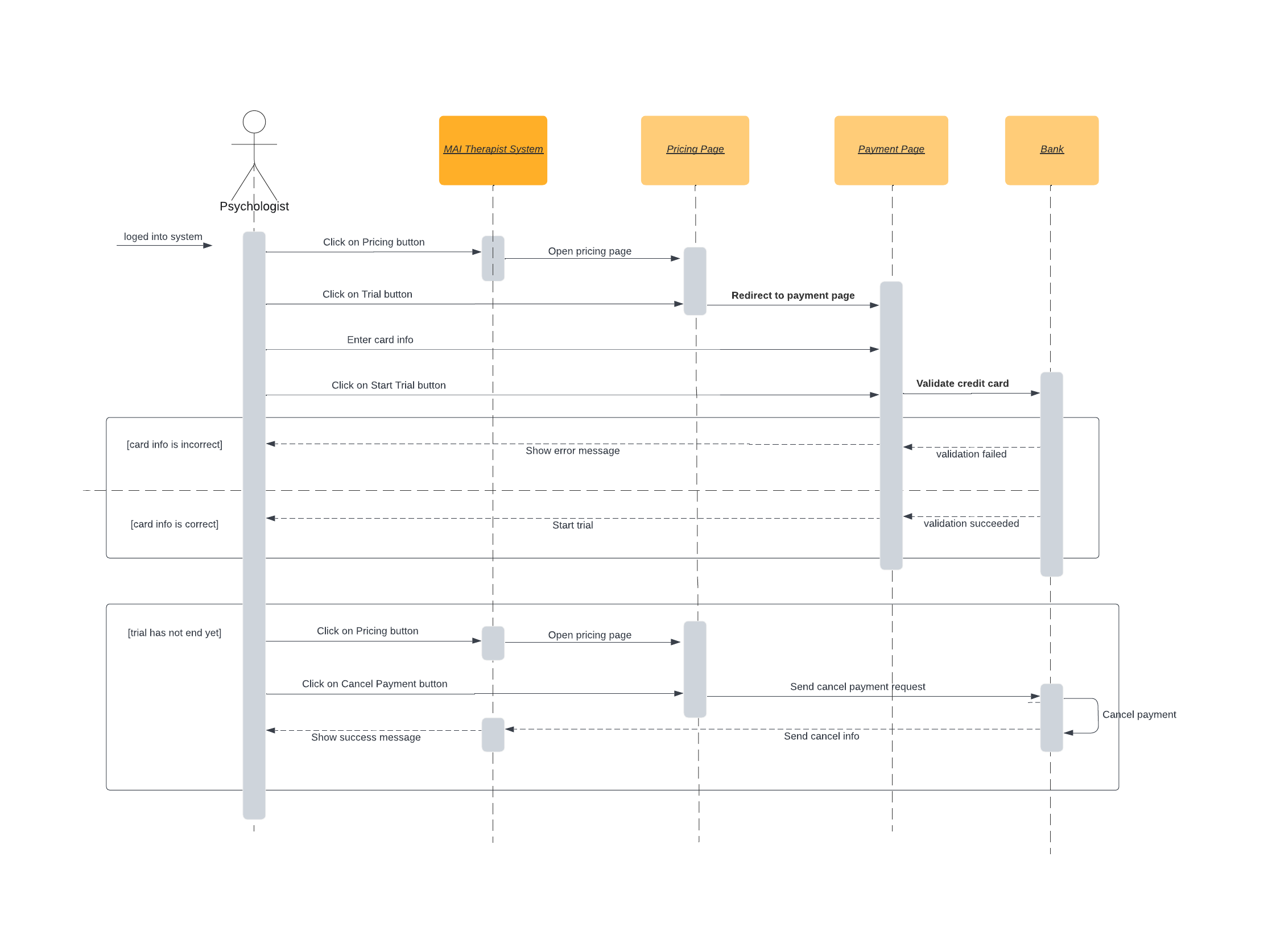
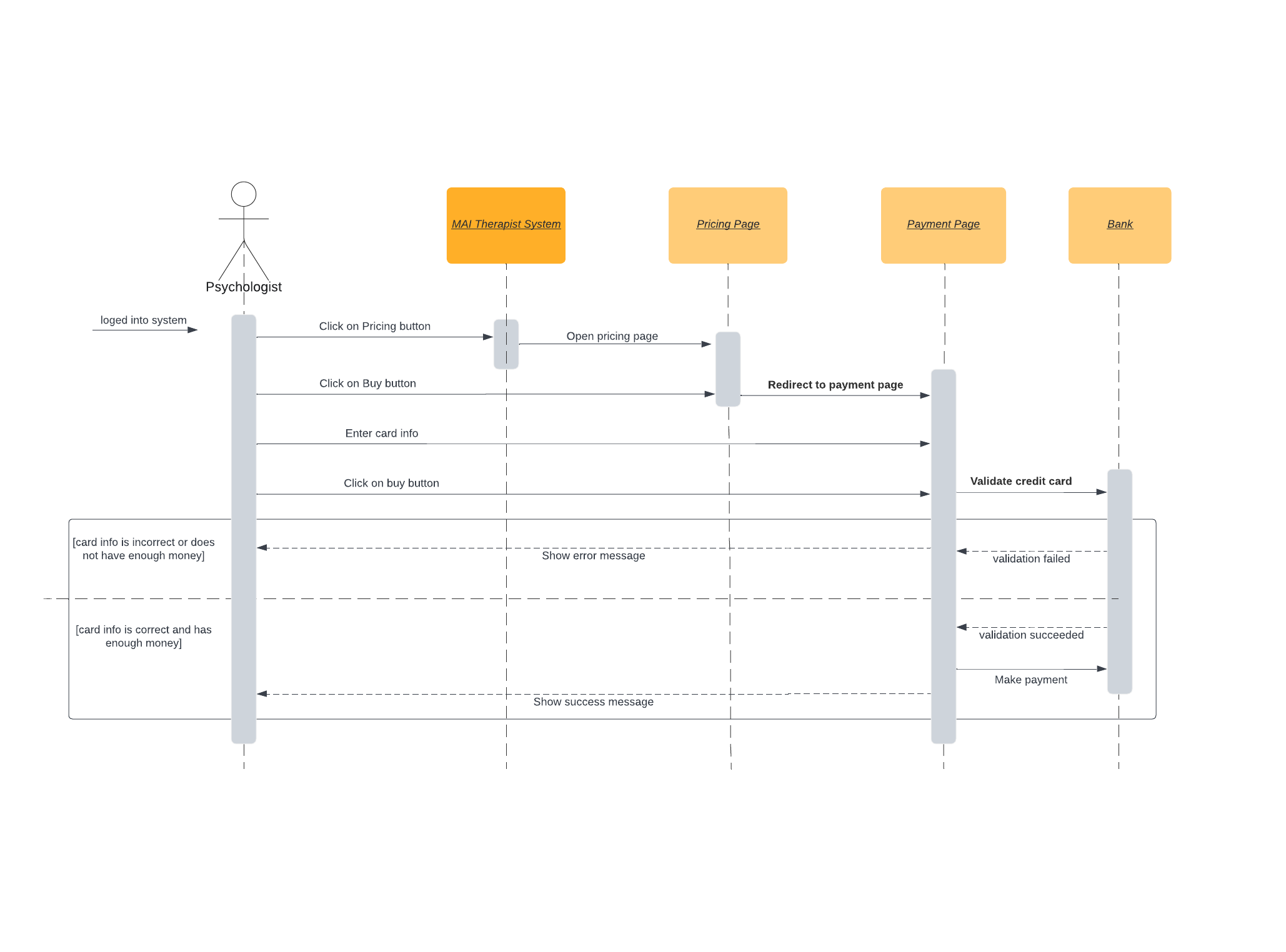
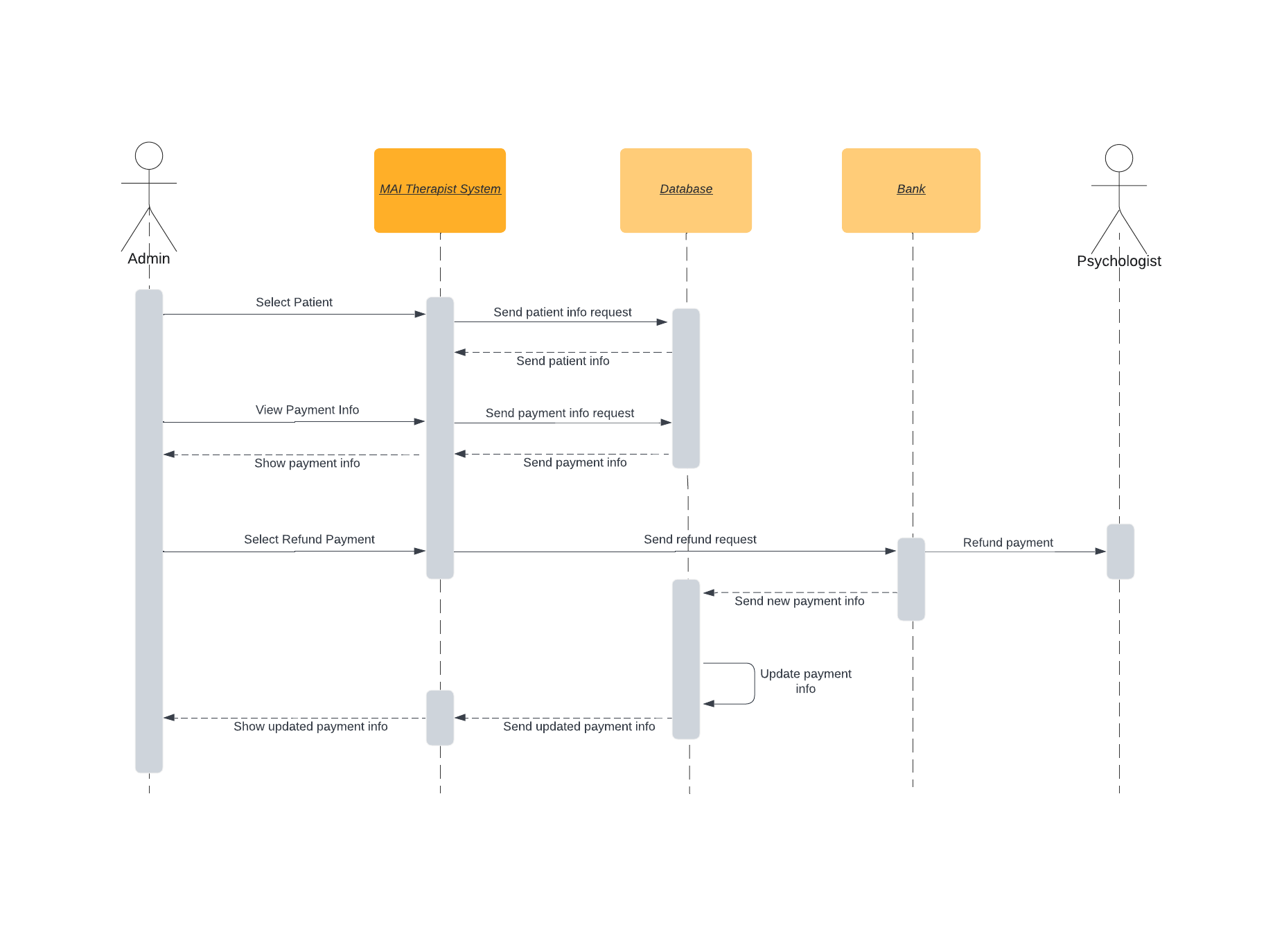


* Register Sequence Diagram



* Login/Logout Sequence Diagram
* Forgot Password Sequence Diagram



* Request Trial/Cancel Payment Sequence Diagram
* Make Payment Sequence Diagram
* Show Payment/Refund Payment Sequence Diagram

Glossary

VueJS : Progressive Web Application for developing webdev related projects. It is a framework of JavaScript and it is a frontend development tool to make its users to interact the Project easily.

Python : Scripting language for backend development.

Cross-Platform : Applications that can be run on both phones, computers and tablets are called cross-platform applications.

Server : Server is a general name given to computer units that run resources or some services (FTP, E-Mail, Web Site) that clients (users) can access, use and share in computer networks.

Captcha : A security measure called query-response validation, also known as CAPTCHA (Fully Automatic General Turing Test for Human and Computer Discrimination), is used. Protection against spam and encryption is aided by CAPTCHA.